Imperial College London
Department of Electrical and Electronic Engineering

Model and Data Driven Approaches to Wireless Image Transmission

by
Can Karamanli

June 2023

Supervised by
Prof. Deniz Gündüz

A thesis submitted in fulfilment of the requirements for the degree of Doctor of Philosophy in Electrical and Electronic Engineering of Imperial College London and the Diploma of Imperial College London

Abstract

This dissertation introduces novel wireless image/video transmission schemes designed to optimize end-to-end reconstruction quality under limited available resources such as memory, bandwidth and power, over static, time-varying, as well as multi-user channels. The proposed approaches address the challenges associated with conventional wireless image transmission by integrating traditional techniques with deep neural network architectures.

One of the contributions is the development of ZZCast, a robust low-complexity solution for wireless image transmission. By leveraging the strengths of both traditional techniques and state-of-the-art deep learning methods, ZZCast achieves highly effective image transmission. Extensive experiments have been conducted to validate its effectiveness, and the results demonstrate its promising performance.

The rest of the thesis focuses on neural network aided joint source-channel coding schemes combined with model-driven design components. In particular, DeepJSCC-MMSE is proposed for transmitting images over a time-varying fading channel when channel state information (CSI) is available only at the receiver. Model-driven channel equalization is employed together with neural network based DeepJSCC scheme for reduced complexity, and is shown to provide improved performance compared to advance neural network architectures employing channel attention modules. Next, we consider a variable-length DeepJSCC scheme which provides adaptation to the source sample and channel conditions. We also study JSCC over a multiple access channel, and propose a model-driven non-orthogonal multiple access scheme. Through numerical experiments, we show that the proposed approach outperform time-division approach by providing a learned access methodology.

Furthermore, the thesis explores advanced techniques including variational autoencoder and diffusion models within the context of wireless image transmission. Through experimentation, the potential of these methods has been demonstrated, showcasing the ability to enhance the performance and capabilities of wireless image transmission systems.

Finally, we develop vision transformer(ViT)-based architectures specifically tailored for image transmission tasks. By leveraging the strengths of ViT, we further enhance the efficiency and effectiveness of image transmission systems.

Overall, we believe that this thesis contributes to the fast developing literature on neural network aided JSCC design, particularly focusing on image transmission applications. We hope that our results motivate the practical value of this novel approach, highlight the potential of integrating deep learning architectures with traditional techniques to revolutionize conventional wireless image transmission. The proposed approaches offer promising performance improvements, and pave the way for further advancements in the field. Continued development and refinement of these methodologies, as well as their exploration in various application domains, hold great promise for the future of wireless image transmission.

Acknowledgements

First of all, I would like to sincerely thank my supervisor, Prof. Deniz Gündüz for his guidance, support, and expertise throughout the course of this research. His insightful feedback, valuable suggestions, and constant encouragement have been instrumental in shaping this dissertation. I am truly grateful for his mentorship and the knowledge I have gained under their supervision.

Secondly, I would like to express my heartfelt gratitude to my examiners, Dr. Imad Jaimoukha and Dr. Iñaki Esnaola, for graciously accepting the responsibility of evaluating my work and providing invaluable feedback to enhance my thesis and guide me in the right direction.

I am deeply indebted to my wife Seda for her understanding, and endless patience during the entire duration of this research journey. Her constant support, encouragement, and belief in my abilities have been a constant source of motivation.

I would also like to extend my gratitude to my brother, Cem for his continuous support, encouragement, and invaluable discussions. His technical expertise, insightful inputs, and willingness to lend a helping hand have greatly contributed to the success of this research. I am grateful for his unwavering support and camaraderie.

I would like to acknowledge the financial support provided by the Republic of Turkey, Ministry of National Education for funding my Ph.D. studies, and for enabling me to study with great researchers at one of the most distinguished universities.

I would also like to extend my heartfelt appreciation to my esteemed colleagues, including Bayram, Mehmet, Okan, Selim Firat, Tze-Yang, Aybars, and all other members of the lab. Their collaboration and collective efforts have significantly enriched my research experience.

©

Copyright

The copyright of this thesis rests with the author. Unless otherwise indicated, its contents are licensed under a Creative Commons Attribution-Non Commercial 4.0 International Licence (CC BY-NC).

Under this licence, you may copy and redistribute the material in any medium or format. You may also create and distribute modified versions of the work. This is on the condition that: you credit the author and do not use it, or any derivative works, for a commercial purpose.

When reusing or sharing this work, ensure you make the licence terms clear to others by naming the licence and linking to the licence text. Where a work has been adapted, you should indicate that the work has been changed and describe those changes.

Please seek permission from the copyright holder for uses of this work that are not included in this licence or permitted under UK Copyright Law.

Declaration of Originality

I, Can Karamanli, declare that this thesis titled, 'Model and Data Driven Approaches to Wireless Image Transmission' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at Imperial College of Science, Technology and Medicine.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.

Signed: Can Karamanli

Date: 7 June 2023

Contents

Abstract 1

Acknowledgements 4

Copyright 5

Statement of Originality 6

Contents 7

List of Figures 10

List of Tables 12

Abbreviations 13

1 Introduction 15
- 1.1 Motivation . . . 15
  - 1.1.1 Current trends . . . 16
    - 1.1.1.1 Network Design . . . 16
    - 1.1.1.2 Efficient Compression Algorithms . . . 18
  - 1.1.2 Joint Source-Channel Coding (JSCC) . . . 18
- 1.2 Uncoded Image/Video Transmission . . . 19
  - 1.2.1 Distinctive Characteristics of Image Processing . . . 22
  - 1.2.2 Particular Constraints Imposed by Image Transmission . . . 23
  - 1.2.3 Problem Definition . . . 25
  - 1.2.4 Deep Joint Source-Channel Coding for Wireless Image Transmission 26
- 1.3 Objectives . . . 26
  - 1.3.1 Contributions . . . 26
- 1.4 Outline . . . 28

2 Preliminaries 30
- 2.1 Wireless Image Transmission . . . 30
- 2.2 Compressive Sensing (CS) . . . 32
- 2.3 Deep Learning . . . 35
- 2.4 JSCC Problem . . . 37
- 2.5 Deep-JSCC Improvements . . . 38

2.6 Deep Learning based Compressive Sensing . . . 39
2.7 Deep Learning for Communications . . . 40
2.8 Image Compression / Denoising . . . 41
2.9 Nonlinear Transform for Image Transmission . . . 41
2.10 Others . . . 42
2.11 Conclusion . . . 42

3 Uncoded Image/Video Transmission 44
3.1 Introduction . . . 44
3.1.1 A new uncoded transmission scheme: ZZCast . . . 46
3.2 System Model . . . 48
3.2.1 ZZCast Algorithm . . . 49
3.3 Numerical Results . . . 53
3.4 Reducing Metadata . . . 58
3.5 Video Transmission . . . 59
3.5.1 Depth ZigZag Algorithm . . . 59
3.6 ZZCast Extensions . . . 61
3.6.1 Discrete Wavelet Transform(DWT) . . . 61
3.6.2 AMP Sparsity Level Information . . . 62
3.7 DL-aided Uncoded Image/Video Tranmission . . . 66
3.7.1 ZZCast with DNN-Decoders . . . 66
3.7.2 ZZCast with Convolutional Decoder . . . 67
3.8 Conclusion . . . 69

4 Model-Driven Deep Joint Source-Channel Coding over Time-Varying Channels 71
4.1 Introduction . . . 71
4.2 Problem Definition . . . 73
4.3 Proposed Solution . . . 74
4.4 Performance Evaluation . . . 77
4.4.1 Experimental Setup . . . 77
4.4.2 Baseline . . . 78
4.4.3 Numerical results . . . 78
4.5 Extensions . . . 82
4.5.1 Metadata-Aided Image Recovery . . . 82
4.5.2 Unsupervised Image Enhancement Recovery . . . 85
4.5.3 Multi-Image Transmission . . . 87
4.6 Conclusion . . . 89

5 Variable Bandwidth ratio Deep Joint Source Channel Coding 93
5.1 Introduction . . . 93
5.2 Related Work . . . 94
5.3 System Model . . . 96
5.4 Initial Proposal . . . 96
5.5 Proposed Solution . . . 97
5.6 Numerical Results . . . 98
5.7 Conclusion . . . 100

6 Model-Driven Distributed Deep Joint Source Channel Coding over a Multiple Access Channel 102
6.1 Introduction . . . 102
6.2 System Model . . . 103
6.3 Complex Attempt . . . 104
6.4 Novelties . . . 104
6.5 Numerical Results . . . 105
6.6 Conclusion . . . 108

7 Deep Generative Models in Deep Joint Source Channel Coding Wireless Image Transmission 109
7.1 Introduction . . . 110
7.2 VAE . . . 110
7.3 Normalizing Flows . . . 115
7.4 GAN . . . 116
7.5 Diffusion Models . . . 119
7.6 Conclusion . . . 121

8 Transformers in Deep Joint Source Channel Coding Wireless Image Transmission 123
8.1 Introduction . . . 124
8.2 Preliminaries . . . 125
8.3 Classification . . . 127
8.4 Image Transmission . . . 130
8.5 Conclusion . . . 133

9 Conclusion 136

List of Figures

1.1 Wireless image transmission 20

2.1 Simple Neural Network 36
2.2 Autoencoder 37
2.3 U-Net 42

3.1 ZigZag encoding 47
3.2 Effect of Chunk Size 51
3.3 Block diagram of the proposed ZZCast transmission scheme 53
3.4 Lenna - 131K Symbol Length 55
3.5 Lenna - 75K Symbol Length 56
3.6 Lenna - 50K Symbol Length 56
3.7 Lenna - 15K Symbol Length 57
3.8 Fading Channel 57
3.9 Comparison between metadata-free schemes 59
3.10 Akiyo - ZZCast vs SoftCast 60
3.11 Akiyo - 100K 61
3.12 Effect of Transmission 62
3.13 Comparison between DCT and DWT (50K Lenna ZZCast) 62
3.14 Exact recovery performance of AMP algorithms for synthetic data 65
3.15 Baseline pipeline of DL-aided ZZCast decoder (No improvement) 67
3.16 Pipeline of U-Net aided ZZCast decoder 67
3.17 Pipeline of Conv aided ZZCast decoder 68
3.18 AWGN BW=128, simple-DCT, Conv, WGAN-GP and U-Net comparison. 70

4.1 Architecture of the proposed method 76
4.2 DeepJSCC-Attention Decoder Architecture 76
4.3 Performance of DeepJSCC-MMSE and DeepJSCC-Attention for R = 1/6 and R = 1/12 AWGN channel. Uniform distribution of SNR from 0dB to 15dB. 79
4.4 Performance of DeepJSCC-MMSE and DeepJSCC-Attention for R = 1/6 and R = 1/12 Rayleigh channel. Uniform distribution of SNR from 0dB to 15dB. 79
4.5 Performance of DeepJSCC-MMSE and DeepJSCC-Attention for R = 1/6 in multiple channel uses. 80
4.6 Channel estimation mismatch performace of DeepJSCC-MMSE and DeepJSCC-Attention in case noise in the noise power 81
4.7 Fixed channel estimation performance of DeepJSCC-MMSE and DeepJSCC-Attention 82

4.8 Proposed Model . . . 83
4.9 CGAN-inspired DeepJSCC-MMSE and DeepJSCC-MMSE tcn=16 . . . 84
4.10 CGAN-inspired DeepJSCC-MMSE and DeepJSCC-MMSE tcn=8 . . . 85
4.11 Proposed Model . . . 88
4.12 InfoGAN-inspired DeepJSCC-MMSE and CGAN-inspired DeepJSCC-MMSE tcn=16 . . . 89
4.13 InfoGAN-inspired DeepJSCC-MMSE and CGAN-inspired DeepJSCC-MMSE tcn=8 . . . 90
4.14 Proposed Model . . . 90
4.15 Multiple image transmission DeepJSCC-MMSE and DeepJSCC-MMSE tcn=16 . . . 91
4.16 Multiple image transmission DeepJSCC-MMSE and DeepJSCC-MMSE tcn=8 . . . 92

5.1 Proposed solution for adaptive bandwidth . . . 97
5.2 Proposed solution for adaptive bandwidth . . . 100

6.1 Initial Proposal . . . 104
6.2 Proposed Model . . . 106
6.3 Proposed Method . . . 106
6.4 Performance of DeepJSCC-NOMA-MMSE and DeepJSCC-TDMA for AWGN. 107
6.5 Performance of DeepJSCC-NOMA-MMSE and DeepJSCC-TDMA for Rayleigh Fading. . . . 108

7.1 VAE . . . 111
7.2 Proposed Method . . . 112
7.3 Comparison . . . 113
7.4 Comparison . . . 114
7.5 Comparison . . . 115
7.6 Block diagram of WGAN. . . . 117
7.7 Block diagram of the proposed method. . . . 120
7.8 Results . . . 121

8.1 Block diagram . . . 128
8.2 Pipeline of the Model . . . 130
8.3 Focal Attention . . . 130
8.4 Proposed Pipeline (CIFAR-10) . . . 131
8.5 Comparison . . . 132
8.6 Comparison . . . 133

List of Tables

1.1 Usual requirements for video delivery. 17

3.1 Amount of Metadata that must be transmitted by ZZCast for different bandwidth constraints. 55
3.2 The amount of metadata used in bits. 55
3.3 Kodak-24 Dataset. 62
3.4 Lenna. 62
3.5 MSE performance when perfect channel case 64
3.6 Performance of methods without transmission 69

4.1 Number of parameters for the compared methods 78

8.1 CoAtNet-based encoder and CNN-based encoder(intermediate dim = 64) . 130

Abbreviations

AR	Augmented Reality
CV	Computer Vision
DNN	Deep Neural Networks
MAC	Multiple Access Channel
MSE	Mean Square Error
OAC	Over-the-Air Communication
QoS	Quality of Service
VR	Virtual Reality

To my family.

Chapter 1

Introduction

1.1 Motivation

Wireless image/video transmission is an important technology that offers a range of benefits in terms of mobility, convenience, real-time feedback, cost reduction, and safety. Wireless image transmission allows for the transfer of images without being tethered to a specific location. This is particularly important for applications such as security cameras, drones, and mobile devices, where the ability to move around is essential. It eliminates the need for cables and other physical connections, making it more convenient and easier to use. It also allows for more flexible installation options, as there are no limitations imposed by physical connections. This can be beneficials for applications such as projectors or virtual reality (VR)/ augmented reality (AR) headsets. Wireless image/video transmission allows for real-time feedback, which is particularly important in applications such as live video streaming, video conferencing, and remote monitoring. The importance of these applications has multiplied in the post-pandemic world due to increasing demand for remote work and education arrangements. Wireless image transmission eliminates the need for expensive cabling and other infrastructure, reducing costs and making it a more cost-effective option. Last but not least, in hazardous environments, such as oil rigs, mines, or disaster areas, wireless image/video transmission can provide remote viewing capabilities without endangering human lives.

Video content already comprises a significant majority of Internet traffic, video streaming accounts for approximately 82% of global Internet traffic now. This trend is expected to grow even further with the increasing use of AR/VR technology for online gaming and immersive metaverse applications. This heavy volume of video traffic can put pressure on networks, especially when video content is considered a high priority.

1.1.1 Current trends

In his seminal paper [1], Shannon introduces the fundamental problem of communications as transmitting a signal from one point to another in a reliable manner. He then proves his separation theorem, which shows that this problem can be divided into two subproblems without loss of optimality: the first problem is known as source coding, and deals with the efficient representation of the input signal with as few bits as possible, while the second problem is known as channel coding, and deals with the reliable transmission of compressed bits over the channel by adding as little redundancy as possible. He then continues to prove his two famous results, known as the source coding theorem and the channel coding theorem, which provide the fundamental limits for these two problems in the asymptotic infinite blocklength regime.

A major consequence of Shannon's separation theorem has been the modular network architecture, where the source compression is dealt with in the application layer, while the communication over the channel is taken care of by the physical layer. Almost all current communication systems and standards follow this layered architecture. Therefore, efforts to sustain the growing demand for video delivery are also done in parallel by trying to improve both the communication capacity and the compression efficiency.

1.1.1.1 Network Design

Wireless data transmission is primarily accomplished through two major methods: Wi-Fi and cellular systems (4G/5G). The wireless transmission of video presents a significant market opportunity, and multiple solutions have been developed by the designers of both to tackle this issue. A study has identified the typical needs for video delivery, which are outlined in Table 1.1.

There are three primary factors that constrain the performance of Wi-Fi: insufficient bandwidth, interference from other users and standards that share the same spectrum, and fluctuating channel conditions resulting from the receiver's mobility. The limited bandwidth is problematic because Wi-Fi operates in unlicensed spectrum, and this spectrum is shared with other users and standards, resulting in interference. Furthermore, the mobility characteristic of Wi-Fi can cause frequent signal attenuation, which further constrains performance. The technical team responsible for developing and implementing Wi-Fi standards is the 802.11 workgroup, with the most recent standard being 802.11ac. One of the primary goals for designers is to address challenges related to video transmission. Numerous approaches have been developed to tackle the challenges with Wi-Fi performance, particularly for video transmission, and improve it. One such approach is

TABLE 1.1: Usual requirements for video delivery.

Throughput	∼100 Mbps
Range	15 meters with up to 3 walls
Audio	2 Audio MP3 stereo streams (128kbps)
Video	2 HD-Video
Remote Gaming	HD-Video stream replaced by 1 Remote Gaming (30 Mbps)
Video/Voice calls	2 VoIP calls (95 Kbps) 1 Video IP Phone (384 Kbps)
IP Data	1 Mbps
Interference	Some co-channel/adjacent channel interference

the implementation of multiple antennas at the transmitter and receiver, referred to as MIMO, which exploits multipath channel conditions to optimize channel usage. Moreover, beamforming can be utilized when a channel with a clear line of sight is accessible. The 802.11ac standard facilitates 256-QAM modulation, while the forthcoming 802.11ax standard enables 1024-QAM modulation. These advanced modulation techniques are anticipated to enhance performance for video transmission. Thus, the 802.11 standard has the potential to facilitate exceedingly high data rates, enabling the transmission of 4K resolution videos. The 5GHz spectrum, which is less congested than the 2.4GHz spectrum, is utilized by both the 802.11n and 802.11ac standards. Furthermore, the 802.11e amendment strives to introduce Quality of Service (QoS) to Wi-Fi, prioritizing packets that are sensitive to latency, like those for video streaming. An extensive analysis of QoS issue for 802.11 is presented in this article [2]. For video streaming over Wi-Fi, the 802.11aa amendment is significant as it outlines several multiple access channel (MAC) enhancements for robust audio and video streaming, and its explanation and evaluation are available in [3]. Wi-Fi's advantage is that all data transmitted can be encrypted, making it less susceptible to interception.

The challenges and solutions for cellular networks and Wi-Fi are similar in many ways, but there are also some differences that affect the complexity of solutions. Unlike Wi-Fi, cellular networks use licensed spectrum, which allows for more efficient use of the spectrum. However, the use of different frequency bands for cellular networks in different regions makes the design of the analog front-end at the receiver more complicated. Although cellular networks are widely available in developed countries and are the most popular communication technology, they are still prone to issues like fading and blind spots. The designers of cellular networks are aware of the demand for video delivery and are continuously working on improving the network's performance.

The latest and widely used technology, commonly known as 5G, will use several intelligent techniques to handle the increasing demand for video delivery. One such technique is adaptive streaming [4], where videos are divided into short segments, and each segment is available at different quality levels, based on factors such as video signal-to-noise ratio (SNR), spatial resolution, and frame rate. Another technique called scalable video coding [5] involves transmitting multiple layers of data based on the user's channel conditions.

Both cellular and WiFi technologies also employ similar physical layer technologies, such as OFDMA, MIMO, and beamforming. Additionally, a key common feature is that channel coding is integrated into the standard. The Wi-Fi standard, 802.11ac, requires the use of convolutional coding, while LDPC is an optional addition that provides a 1-2dB gain in performance, especially ultra-low latency. LDPC codes are more suitable for high-end receivers due to their higher complexity. In contrast, for cellular networks like 3G and 4G, turbo codes and LDPC codes are more commonly used. Recently, polar codes have gained popularity in 5G due to their explicit code construction and simple decoding schemes with modest complexity and memory requirements.

1.1.1.2 Efficient Compression Algorithms

Additionally, newer video codecs are available that support 4K and 8K videos, offer higher dynamic range, 10-bit color accuracy, and can achieve compression rates of up to 50%.

1.1.2 Joint Source-Channel Coding (JSCC)

Unequal error protection (UEP) [6–8] is a technique used in communication systems to provide different levels of error protection to different parts of a transmitted signal. In UEP, more important parts of a signal, such as headers or critical information, are given higher levels of protection, while less important parts are given lower levels of protection. This is done by encoding the signal with different error correction codes (ECCs) that have varying levels of redundancy, resulting in different levels of error resilience. The purpose of UEP is to maximize the overall quality of the received signal by allocating resources to protect the most important parts of the signal more effectively. UEP is commonly used in applications where the transmitted data consists of different types of information with varying degrees of importance, such as audio, video, or image transmission.

Product Codebook Codebook Vector Quantization(PCCOVQ) [9] is another joint source-channel coding scheme that is used for the compression and transmission of correlated

sources over noisy channels. In PCCOVQ, the source data is first partitioned into subsets, and each subset is then quantized using a separate codebook. The resulting codewords are then reassembled into a single sequence and quantized again using a global codebook. The resulting codewords are then transmitted over the noisy channel. At the receiver, the codewords are decoded by first decoding the global codewords and then using them to decode the partitioned codewords. PCCOVQ has been shown to provide better performance than other joint source-channel coding schemes, particularly for highly correlated sources. However, both schemes do not satisfy graceful degradation.

1.2 Uncoded Image/Video Transmission

In digital image transmission, various communication designs are employed based on the prevailing channel conditions. However, these designs may suffer from two critical limitations that affect the reliability and performance of the transmission process. The first limitation is known as the *cliff effect*, and the second is referred to as the *threshold effect*. The cliff effect in digital image transmission occurs when the quality of the received signal drops below a critical threshold due to adverse channel conditions. Traditional transmission designs are often optimized for specific channel characteristics, such as signal-to-noise ratio (SNR) or channel capacity. If the actual channel conditions deteriorate beyond what was expected during the design phase, the system's error rate increases significantly, leading to a sudden and complete failure of the transmission process. As a consequence, the received image quality experiences a severe and abrupt degradation. The threshold effect, on the other hand, manifests when the transmission performance fails to improve even if the channel conditions turn out to be better than initially assumed. Conventional designs may operate close to the limit for the expected conditions, but they lack the adaptability to exploit the potential improvements in transmission quality when the channel conditions unexpectedly surpass the original estimates. Consequently, the image transmission performance remains stagnant, and the system fails to capitalize on the available channel gains.

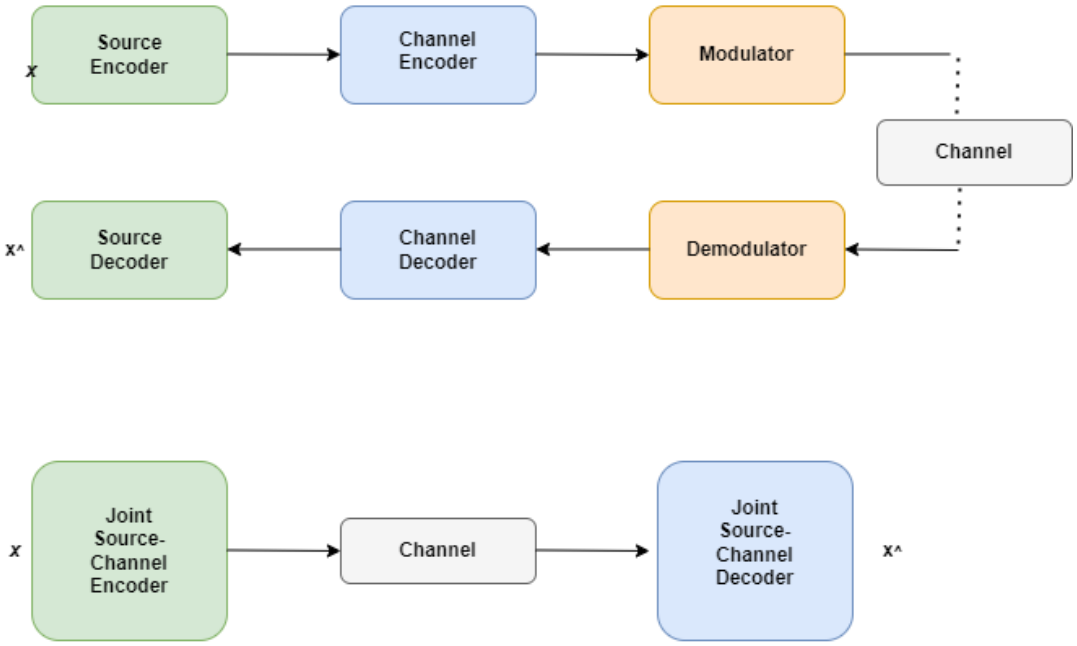


FIGURE 1.1: Wireless image transmission

To overcome the challenges posed by the cliff and threshold effects, JSCC emerges as a promising solution for digital image transmission. JSCC involves the integration of source coding and channel coding in a unified framework, allowing for joint optimization of the transmission process. By jointly considering the characteristics of the image source and the wireless channel, JSCC can dynamically adjust its coding strategies during transmission. This adaptability ensures that, even under adverse channel conditions leading to the cliff effect, the system can still recover the image to some extent by leveraging the redundancy inherent in the source. Thus, JSCC significantly reduces the likelihood of complete transmission failure and provides a certain level of image quality in challenging channel scenarios. Moreover, in instances where the channel conditions turn out to be better than expected, JSCC capitalizes on the improved spectral conditions. By efficiently exploiting the available channel gains, JSCC achieves higher coding efficiency, approaching the Shannon limit more closely and enhancing the overall image transmission performance. This adaptive nature of JSCC empowers the system to perform optimally across a wide range of channel conditions, providing a more robust and efficient solution compared to traditional transmission designs. Although many papers have considered JSCC, they still have separate compression and channel coding components, and consider bits as a common currency between the two. In that sense, they do not provide a direct mapping from the source domain to the channel input space. On the other hand, analog modulation schemes, such as AM/FM directly modulate input signal onto channel input. More interestingly, such methods are known to achieve the optimal performance when

transmitting Gaussian source samples over an AWGN channel [10]. This motivated the question whether similar benefits can be obtained in more practical systems.

Motivated by the theoretical gains of uncoded transmission, an uncoded image/video delivery method was developed in [11]. Rather than using channel codes, the proposed method maps pixel values directly onto constellation points. The quality of video received by each user is determined by the level of corruption caused by the channel. Although SoftCast cannot meet the performance achieved by state-of-the-art digital schemes in a static/known channel condition, it has an inherent robustness against channel variations, which makes it attractive for practical applications. While traditional digital transmission selects the coding rate based on the channel condition and may require retransmission in case of sudden negative changes, resulting in a pause in video playback, SoftCast provides graceful degradation. This means that the user experiences a continuous degradation in video quality instead of a complete failure of decoding, making it a more reliable option, particularly in fast-varying channel conditions. The promising results in [11] has led to flurry of activity in uncoded image/video delivery in the following decade, where many improvements have been proposed. One such improvement will also be presented in this thesis. However, all these methods can be considered as heuristic model-based designs, where various linear and non-linear operations are combined and optimized to improve the end-to-end quality. A major breakthrough took place in this area with the adoption of data-driven deep learning techniques.

In conventional wireless communication systems, source encoding and channel encoding are two distinct blocks that operate independently. The former is for compression and gets rid of the redundant information within a prescribed distortion target, and the latter is for reliable transmission, and introduces structured redundancy to mitigate the effects of noise and interference in the channel. According to Shannon's separation theorem, there is no loss of optimality in this two-step process (under the usual Shannon theoretic assumptions of infinite blocklength and unbounded complexity), and it is widely used in practical implementations.

Assuming that the transmitter and receiver have accurate channel state information, they can determine the optimum modulation scheme and coding rate that can transmit information at a rate close to the capacity of the channel. Similarly, the compression rate is chosen such that the bits produced by the source encoder can be conveyed reliably by the channel encoder. However, this procedure has several drawbacks. When the actual channel condition is worse than the estimated quality, the receiver cannot decode the channel codeword, and hence, the whole process fails, leading to maximum distortion at the decoder. Also, if the channel condition happens to be better than the estimated value, reconstruction quality would not improve. This is known as *cliff effect* and causes

practical system to operate well below the capacity, and hence, far from the optimal theoretically achievable (OPTA) to avoid catastrophic failures.

JSCC has the potential to overcome these limitations. While JSCC has been extensively studied in the literature [11], most of the works consider separate source and channel coding algorithms, whose parameters are optimized jointly. A fully joint architecture would require mapping the source samples directly to channel inputs, without going through any compression step, similarly to amplitude or frequency modulation (AM/FM) schemes. Such an uncoded approach, particularly targeting wireless image and video transmission, has been originally proposed in [12], and recently popularized by [11] as SoftCast.

1.2.1 Distinctive Characteristics of Image Processing

Processing of images holds a unique and paramount position, setting itself apart from traditional data processing methodologies. Images, as a form of visual information, present a multitude of complexities and challenges that demand specialized approaches for meaningful analysis and interpretation. One of the key differences that distinguish image data from other types of data lies in its spatial structure. Images are represented as two-dimensional arrays of pixels, capturing rich visual content, intricate details, and spatial relationships that are often imperceptible to other data processing techniques.

Unlike conventional numerical or textual data, images do not follow a linear sequence of values. Instead, they convey visual patterns and colors, making them more suitable for tasks like object recognition, scene understanding, and content-based retrieval. The dimensionality of image data, as well as the visual content it encapsulates, necessitates the use of domain-specific techniques tailored to extract meaningful features and patterns effectively. Another notable difference is the subjectivity involved in interpreting images. Human perception and cognitive biases play a significant role in how images are understood and analyzed. As a result, image processing tasks often require human intervention, such as in image annotation and validation, making it distinct from other types of data processing that rely primarily on objective algorithms and rules. Moreover, image data can be considerably larger in size and volume compared to some other types of data. High-resolution images or image sequences from video streams can occupy a significant amount of storage space and require substantial computational resources for processing. Handling such large volumes of data demands efficient compression and processing techniques tailored to the unique characteristics of visual information. Let us give some comparisons to highlight the difference the image processing with other data type such as audio. One fundamental difference is the dimensionality of the data. Image

data is typically represented in two dimensions (2D) with pixels arranged in rows and columns, while audio data is one-dimensional (1D) and represented as a sequence of samples along the time axis. Images convey visual information, while audio conveys auditory information. The nature of the information they represent is fundamentally different, which leads to variations in how they are processed and analyzed. Image data relies on spatial features such as edges, textures, and colors. On the other hand, audio data relies on temporal features like frequency patterns, harmonics, and sound duration. While both image and audio data can exhibit some sparsity in certain domains, the nature of sparsity may differ. For images, sparsity is often observed in transform domains like the DCT or wavelet domains, where certain coefficients are close to zero. In audio, sparsity can be observed in the frequency domain as well, but it may manifest differently due to the distinct nature of audio signals. Therefore, image processing stands as a distinct and specialized field within the broader landscape of data processing. Its unique characteristics and challenges distinguish it from traditional data processing methodologies, making it a crucial and complex domain in the digital realm. The spatial structure of images, represented by two-dimensional arrays of pixels, enables the capture of rich visual content and intricate details, setting them apart from conventional numerical or textual data. This spatial arrangement necessitates domain-specific techniques for extracting meaningful features and patterns effectively. By understanding the intricacies of image transmission, we can pave the way for enhanced multimedia applications, improved remote sensing capabilities, and more immersive visual experiences in the ever-evolving landscape of wireless communication technologies.

1.2.2 Particular Constraints Imposed by Image Transmission

Image transmission, particularly this thesis' focus over various communication channels, can present several constraints and challenges due to the nature of images. Some of the particular constraints imposed by image transmission include:

Bandwidth limitations: Images often have large file sizes, especially when they are of high resolution or contain complex visual content. Transmitting such large files over networks with limited bandwidth can lead to slow transmission speeds or even timeouts.

Data loss: During image transmission, data packets may get lost or corrupted due to network issues. Loss of even a small percentage of data can significantly degrade image quality and affect the overall visual perception.

Latency: Image transmission can suffer from latency issues, where the time delay between sending and receiving images can be noticeable. This is especially critical in real-time applications like video conferencing or remote monitoring.

Compression artifacts: To reduce file size and improve transmission efficiency, images are often compressed. However, lossy compression methods can introduce artifacts and degrade image quality, especially in regions with fine details or gradients.

Noise and interference: Image transmission can be susceptible to noise and interference, particularly in wireless communication channels. These unwanted signals can corrupt image data and result in visual distortions.

Transmission errors: In addition to data loss, transmission errors can occur due to signal attenuation, electromagnetic interference, or other factors, leading to inaccuracies in the transmitted image.

Format compatibility: Different devices and applications may use various image formats. Ensuring compatibility between the sender's and receiver's devices is essential to avoid issues during transmission.

Security and privacy concerns: Images can contain sensitive information, and their transmission must be secure to prevent unauthorized access, tampering, or interception.

Real-time requirements: Some applications, such as video streaming or video conferencing, demand real-time image transmission with minimal delay to maintain a smooth and responsive experience.

To overcome these constraints, various techniques are employed, including error correction codes, data compression methods, adaptive transmission protocols, and encryption for security. The choice of image format and transmission protocol depends on the specific application and available resources. Image transmission constraints significantly impact wireless communication systems due to the specific challenges posed by wireless channels. Wireless communication inherently suffers from limited bandwidth, increased susceptibility to noise and interference, and higher rates of data loss compared to wired connections. These constraints are exacerbated when transmitting image data, as images often have larger file sizes and require more resources for processing and transmission. The bandwidth limitations can lead to slower image transfer rates, causing latency issues in real-time applications. Additionally, the presence of noise and interference in wireless channels can introduce errors and distortions in the transmitted images, resulting in a degradation of image quality. Moreover, the data loss that can occur during transmission poses a serious challenge in maintaining image integrity. To address these issues and enhance image transmission over wireless communication systems, innovative approaches are required, including advanced error correction techniques, efficient image compression algorithms, adaptive transmission protocols, and robust encryption methods to ensure secure and reliable data transfer. And in this thesis, we will be providing novel methods

to address some of these constraints and enhance the efficiency and reliability of image transmission over communication channels.

1.2.3 Problem Definition

We consider the transmission of images over a time-varying channel with limited bandwidth, and subject to a constraint on transmitter power. Let H, W and C, denote the height, width and color dimensions, respectively, of the input image. Then the input image $\mathbf{x} \in \mathbb{R}^n$, where $n = H * W * C$, is mapped to a set of continuous valued channel input symbols $\mathbf{y} \in \mathbb{C}^k$ via an encoder $C : \mathbb{R}^n \mapsto \mathbb{C}^k$, which satisfies the average power constraint $\bar{P}$, such that

$$\frac{1}{k} E[||\mathbf{y}||_2^2] \leq \bar{P}. \tag{1.1}$$

We will define the *bandwidth ratio* $R = k/n$, which is the average number of channel symbols for each source symbol. In conventional digital transmission, the introduction of channel coding is a necessary step. However, this thesis takes a different approach by applying JSCC, effectively merging both processes into a single unified step. The wireless channel introduces noise and interference during image transmission. Let H be the channel gain vector, $\mathbf{H} \sim CN(0, \sigma_1^2\mathbf{I})$, that represents the effect of the wireless channel on the transmitted image, accounting for attenuation and let $\mathbf{w} \sim CN(0, \sigma_2^2\mathbf{I})$ be an independent and identically distributed (i.i.d.) complex Gaussian noise vector. The received channel output, denoted by Z, can then be expressed as Z = H * Y + W, where $\mathbf{z} \in \mathbb{C}^k$. The received channel output is decoded the reconstructed image, denoted by X', using the decoding function $X' = D(Z)$, where $\mathbf{X'} \in \mathbb{R}^n$. The overall performance can be evaluated using metrics such as peak signal-to-noise ratio (PSNR) or structural similarity index (SSIM). Therefore, the overall aim of this thesis is developing a encoder and decoder pairs that can efficiently utilize the available bandwidth and power resources while maximizing the end-to-end reconstruction quality, measured by

$$\text{PSNR} = 10 \log_{10} \left(\frac{A^2}{\text{MSE}(X, X')} \right) dB, \tag{1.2}$$

where $\text{MSE}(X, X') = ||X - X'||_2^2$ and A is the maximum possible value of the image pixels (255 in our case).

1.2.4 Deep Joint Source-Channel Coding for Wireless Image Transmission

Important milestone in solving the above communication problem was achieved by the recently proposed deep learning based schemes [13]. A new wireless image transmission technique called deep joint source and channel coding (DeepJSCC) is proposed in [13], which similarly to [11], directly maps image pixel values to complex-valued channel input symbols without relying on explicit codes for compression or error correction. However, unlike in [11], the encoder and decoder functions are parameterized by two convolutional neural networks (CNNs) that are jointly trained in a data-driven manner, forming an autoencoder with a non-trainable layer in the middle representing the noisy communication channel. The results show that the proposed DeepJSCC scheme outperforms digital transmission using JPEG or JPEG2000 compression concatenated with a capacity-achieving channel code at low signal-to-noise ratio (SNR) and channel bandwidth values in the presence of additive white Gaussian noise (AWGN). DeepJSCC also exhibits a graceful performance degradation as the channel SNR varies with respect to the SNR value assumed during training, avoiding the 'cliff effect'. In the case of a slow Rayleigh fading channel, DeepJSCC learns noise-resilient coded representations and significantly outperforms separation-based digital communication at all SNR and channel bandwidth values.

1.3 Objectives

This dissertation focuses on the wireless image transmission problem, and proposes novel techniques to improve the end-to-end performance for both the model and data-driven approaches. We also show the benefits of combining the two approaches.

1.3.1 Contributions

This dissertation proposes novel wireless image/video transmission schemes for different scenarios for available resources such as memory, bandwidth, power and point-to-point or distributed channels (multiple access channels) which are quite important. We propose a novel approach, named ZZCast, for conventional wireless image transmission that integrates conventional techniques with deep neural network architectures, resulting in a simple but highly effective method. To validate its effectiveness, we conducted extensive experiments, which yielded promising results.

Our approach, named ZZCast, presents a robust solution to conventional wireless image transmission by combining the strengths of traditional techniques with state-of-the-art deep learning methods. Through extensive experimentation, we have demonstrated the effectiveness of ZZCast and its potential to revolutionize the field. Furthermore, we have proposed an extension, called DeepJSCC-MMSE, which reduces memory requirements and incorporates advanced techniques like MAC and adaptive bandwidth to improve bandwidth efficiency. Our exploration of advanced methods such as variational autoencoder and diffusion models further highlights the potential of our approach. We are enthusiastic about continuing to develop and refine our methodology and exploring its applications in various fields.

1.4 Outline

In this dissertation, we first present preliminary materials in Chapter 2 for understanding the fundamental concepts that will be used in the following chapters. This chapter will give a simple intuition and basic knowledge around this thesis.

In Chapter 3, we present a simple alternative to SoftCast, called ZZCast. We will show that ZZCast provides reduced computational complexity. It is a very good fit for simple devices such as sensors with limited computational capabilities. Extensive comparisons with digital alternatives will be provided, and the main benefits of ZZCast will be highlighted. We will also present a deep learning aided decoder architecture to illustrate the potential gains when the receiver can be a more capable device.

In Chapter 4, we present DeepJSCC-MMSE, a joint model and data driven approach, where the powerful DeepJSCC architecture is combined with a simple channel estimation module so that the benefits of DeepJSCC can be extended to all channel signal-to-noise ratio (CSNR) values while training a single encoder/decoder pair. We compare the performance of DeepJSCC-MMSE with the widely used alternative attention-decoder. We extend its application to OFDM channels and show its superiority over attention-decoder particularly when the number of parallel channels is large. We also show the robustness of the scheme against channel estimation errors.

In Chapter 5, we explore the dynamic-rate allocation for DeepJSCC and propose a novel architecture that utilizes a control mechanism with ACK/NACK feedback to determine the required bandwidth for image transmission. Our approach aims to achieve the target performance metric and optimize bandwidth efficiency. We compare the performance of our proposed algorithm with a baseline approach and demonstrate its effectiveness in terms of target performance and bandwidth allocation.

In Chapter 6, we introduce a MAC version of DeepJSCC-MMSE, where we employ multi-task learning, deep metric learning, and curriculum learning to enable the efficient transmission of multiple images by the decoder. We compare the performance of our algorithm with the point-to-point single image transmission, which employs the same neural network architecture. Additionally, we show the fairness of the results obtained by multiple transmitters.

In Chapter 7, we take into account novel generative models in wireless image/video transmission, which will be used to improve the perceptual quality of the reconstructed image of the receiver. DeepJSCC scheme employing generative models will be presented, its pros and cons will be discussed and the applications on wireless image transmission will be exemplified.

In Chapter 8 of the thesis, we will present novel DeepJSCC architectures based on the transformer architectures, which has recently gained popularity for their state-of-the-art performance in a large variety of downstream tasks in vision applications, such as classification and segmentation. Pros and cons will be discussed and potential improvements on the wireless transmission will be demonstrated via considering two distinct scenarios of task-oriented communication and wireless image transmission.

Finally, in Chapter 9, we conclude the thesis on novel data and model-driven approaches to wireless image transmission, and discuss potential future directions, as well as open questions and challenges that need to be addressed.

Chapter 2

Preliminaries

This chapter serves as an introduction to the fundamental concepts and methods of wireless image transmission. It lays out the basic pipeline used for this process. Our primary aim is to provide the reader with essential knowledge and establish a strong foundation for understanding the later parts of the thesis.

Additionally, we conduct a literature survey in this chapter to highlight the significance of our research in both the image processing and wireless communication communities. By doing so, we emphasize the importance of our work in advancing the fields of image processing and wireless communication.

In summary, this chapter acts as a starting point, equipping the reader with essential background information and demonstrating the relevance and impact of our research in the broader context of image processing and wireless communication.

2.1 Wireless Image Transmission

Image transmission involves sending digital images over a communication channel from one location to another. The process encompasses several key steps: Image Acquisition: The initial step involves capturing the image using a camera, scanner, or other image capture devices. The image is then stored in a digital format, such as JPEG or PNG.

Image compression plays a crucial role in dealing with bandwidth constraints during wireless transmission. Its main objective is to reduce the file size of a digital image while preserving its visual quality. Since bandwidth limitations are widespread in wireless communication, employing image compression helps alleviate the burden on data transmission. By reducing the size of the image, the demand for network resources is

minimized, making the transmission process more efficient and practical. There are two primary methods used for image compression: lossless and lossy compression.

Lossless Compression: This approach reduces the file size without sacrificing any image information. Lossless compression algorithms, such as run-length encoding and Huffman coding, are utilized. These methods are ideal for images requiring preservation in their original form, such as medical images and satellite imagery.

Lossy Compression: In this method, the file size is reduced by discarding non-critical image information. Lossy compression algorithms, like transform coding, transform image data into a different domain (e.g., the frequency domain) and then discard high-frequency information, which is often less perceptually significant. While lossy compression can significantly reduce file size, it may result in a loss of visual quality.

Various algorithms are available for image compression, catering to different needs and image types, whether the goal is to achieve lossless or lossy compression. Some popular algorithms used in this context include JPEG, JPEG 2000, GIF, PNG, BMP, and more.

When it comes to lossless compression, algorithms like PNG and BMP are commonly utilized. These methods preserve all image data without any loss, ensuring an exact representation of the original image after decompression.

On the other hand, for situations where a certain degree of image quality reduction can be tolerated, lossy compression algorithms like JPEG and JPEG 2000 are often preferred. These algorithms achieve higher compression ratios by selectively removing non-essential image data. While this results in some loss of visual fidelity, the trade-off is a significantly reduced file size, making them suitable for scenarios with bandwidth limitations.

By having access to a range of compression algorithms, we can tailor the compression process according to specific requirements, image content, and available bandwidth, ensuring efficient and optimized image transmission.

Image Encoding: The compressed image is then encoded into a digital signal suitable for transmission over the communication channel. Encoding methods, such as ASCII or Unicode, are utilized.

Transmission: The encoded image is sent over the chosen communication channel, which can be either a wired or wireless network. This thesis primarily focuses on wireless transmission.

Image Decoding: At the receiving end, the encoded image is decoded back into its original format, reversing the encoding process to retrieve the original image.

Image Decompression: The received, compressed image is decompressed to recover the original image.

Image Display: Finally, the image is displayed on a suitable device, such as a monitor or a smartphone screen. Metrics are used to evaluate the effectiveness and quality of the transmission process. These metrics can include objective measurements such as Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), Mean Squared Error (MSE), and others. These metrics provide numerical values that help assess how well an image has been recovered and how close to the original in terms of visual fidelity.

Metrics are essential in determining the efficiency and success of image transmission algorithms, as they allow researchers and practitioners to objectively measure the impact of compression on image quality and the extent of data reduction achieved. By using appropriate metrics, researchers can make informed decisions about selecting the best compression algorithm for specific applications and ensure that the desired balance between file size and image quality is achieved.

2.2 Compressive Sensing (CS)

CS is a technique used to acquire and reconstruct signals that are sparse or compressible in a certain domain. The basic idea behind CS is that if a signal has a sparse representation in a certain domain, it can be reconstructed using a smaller number of measurements than the traditional Nyquist-Shannon sampling rate would require. CS allows for the acquisition of high-dimensional signals using a smaller number of measurements.

Compressive Sensing, also known as Compressed Sensing or CS, is a revolutionary technique used to acquire and reconstruct signals that exhibit sparsity or compressibility in a certain domain. The fundamental concept behind CS is that if a signal can be sparsely represented in a specific domain, it is possible to accurately reconstruct the signal using a considerably smaller number of measurements compared to the traditional Nyquist-Shannon sampling rate.

In traditional signal acquisition, the Nyquist-Shannon sampling theorem states that a signal must be sampled at a rate of at least twice its highest frequency to be accurately reconstructed. However, in many cases, the signals of interest have a sparse representation in some domain and therefore can be accurately reconstructed using a smaller number of samples.

The main components of CS are the measurement process and the reconstruction algorithm. In the measurement process, a linear projection of the signal is taken and is

compressed into a smaller number of measurements. In the reconstruction algorithm, the original signal is estimated from these measurements by finding the sparsest possible representation of the signal in a certain domain.

Mathematically, let's consider a signal $\mathbf{x}$ of length N, which can be expressed in some domain (e.g., time or frequency) as a linear combination of a small number K of non-zero coefficients ($K \ll N$). In other words, the majority of elements in $\mathbf{x}$ are zero or close to zero. We can represent this sparse signal as:

$$\mathbf{x} = \mathbf{\Psi}\boldsymbol{\alpha}$$

where $\mathbf{\Psi}$ is an $N \times N$ transformation matrix that defines the domain in which $\mathbf{x}$ is sparse, and $\boldsymbol{\alpha}$ is a $K \times 1$ vector containing the non-zero coefficients of $\mathbf{x}$. In the context of CS, K is the sparsity level of the signal.

The goal of Compressive Sensing is to efficiently acquire a set of M measurements ($M < N$) of the signal $\mathbf{x}$ through the process of linear projection:

$$\mathbf{y} = \mathbf{\Phi}\mathbf{x}$$

where $\mathbf{\Phi}$ is an $M \times N$ measurement matrix. The compressed measurements $\mathbf{y}$ can then be used to reconstruct the original signal $\mathbf{x}$.

To reconstruct the original signal $\mathbf{x}$ from the compressed measurements $\mathbf{y}$, we need to solve the underdetermined system of equations:

$$\mathbf{y} = \mathbf{\Phi}\mathbf{\Psi}\boldsymbol{\alpha}$$

Due to the sparsity assumption, the solution to this system can be found by solving the following l1-norm minimization problem:

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_1 \quad \text{subject to} \quad \mathbf{y} = \mathbf{\Phi}\mathbf{\Psi}\boldsymbol{\alpha}$$

where $\|\boldsymbol{\alpha}\|_1$ represents the l1-norm of vector $\boldsymbol{\alpha}$. This optimization problem aims to find the sparsest representation of the signal $\mathbf{x}$ in the domain defined by $\mathbf{\Psi}$ while satisfying the compressed measurements $\mathbf{y}$.

CS has been applied in a wide range of fields including image and video compression, medical imaging, wireless communication, and many more. The technique has been proven

to be successful in many applications where traditional signal processing techniques are not efficient.

In the context of our thesis, Compressive Sensing (CS) emerges as a highly relevant and promising technique. As we propose the transmission of compressed versions of images, we encounter a scenario that can be effectively addressed by CS. CS was initially developed for underdetermined signals, precisely where the number of measurements is significantly smaller than the dimension of the signal being transmitted. In our wireless setup, bandwidth mismatch is a ubiquitous challenge that often results in limited resources for data transmission. By employing CS, we can transmit sparse vectors of the image, effectively reducing the number of measurements required for data transmission. At the receiver end, CS estimation techniques can then be applied to accurately recover the original vectors, overcoming the bandwidth constraints. This integration not only enhances the efficiency of wireless transmission but also enables the reconstruction of high-quality images, thereby contributing to the advancement of image communication in bandwidth-constrained wireless environments. In one of our proposed communication schemes, we transmit both sparse and non-sparse image vectors from a sender to a receiver through a noisy channel. Before transmission, the sender encodes each image vector based on its sparsity. If an image vector is determined to be sparse, we exploit CS techniques to reduce the bandwidth requirements. The CS algorithm reduces the dimension of the sparse vector before transmission to improve efficiency. If an image vector is non-sparse, no dimensionality reduction is applied, and the original vector is transmitted as it is.

At the receiver, we receive the measurements from the transmission channel. For sparse image vectors, we employ the Approximate Message Passing (AMP) algorithm to estimate the original sparse vectors accurately. The AMP algorithm helps recover the sparse vectors even in the presence of noise introduced during transmission.

The overall goal of this communication system is to achieve efficient transmission by utilizing CS techniques for sparse vectors while ensuring accurate recovery of both sparse and non-sparse image vectors at the receiver using the AMP algorithm.

This system design allows for optimized bandwidth utilization for sparse image vectors while maintaining reliable signal recovery for both sparse and non-sparse cases, making it suitable for various applications.

2.3 Deep Learning

Deep learning, a subset of machine learning, is a cutting-edge approach that harnesses the power of artificial neural networks (ANNs) with multiple layers. These interconnected layers enable the model to automatically learn intricate patterns and features present in the data. The success of deep learning has been particularly evident in various complex tasks, including image and speech recognition, natural language processing, and computer vision.

In deep learning, models like convolutional neural networks (CNNs) and recurrent neural networks (RNNs) have demonstrated remarkable capabilities. CNNs excel in image and visual data analysis, leveraging specialized layers like convolutional and pooling layers to extract relevant features from images. RNNs, on the other hand, are well-suited for sequential data, such as time series and natural language, owing to their ability to capture temporal dependencies.

The power of deep learning lies in its ability to tackle problems that were once considered insurmountable. One of its groundbreaking achievements was in image recognition, where deep learning models outperformed traditional computer vision methods by a significant margin. Similarly, in natural language processing, deep learning-based language models, like Transformers, revolutionized tasks like machine translation and text generation.

However, the success of deep learning comes at a cost. Training deep learning models demands vast amounts of data and significant computational resources. For complex tasks, such as training deep neural networks with millions of parameters, specialized hardware, like GPUs and TPUs, is often required to accelerate the computations.

Furthermore, deep learning models can be susceptible to overfitting, a phenomenon where the model memorizes the training data and fails to generalize well on unseen examples. Regularization techniques, early stopping, and data augmentation are employed to mitigate overfitting and improve generalization.

Backpropagation is a fundamental algorithm at the core of training artificial neural networks. As a supervised learning technique, backpropagation is employed to optimize the model's weights and biases based on the error between predicted outputs and the true target values.

The process of backpropagation begins by making forward passes through the neural network, where the input data propagates through successive layers, generating predictions. The error is then computed by comparing these predictions to the actual target values. The goal is to minimize this error during the training process.

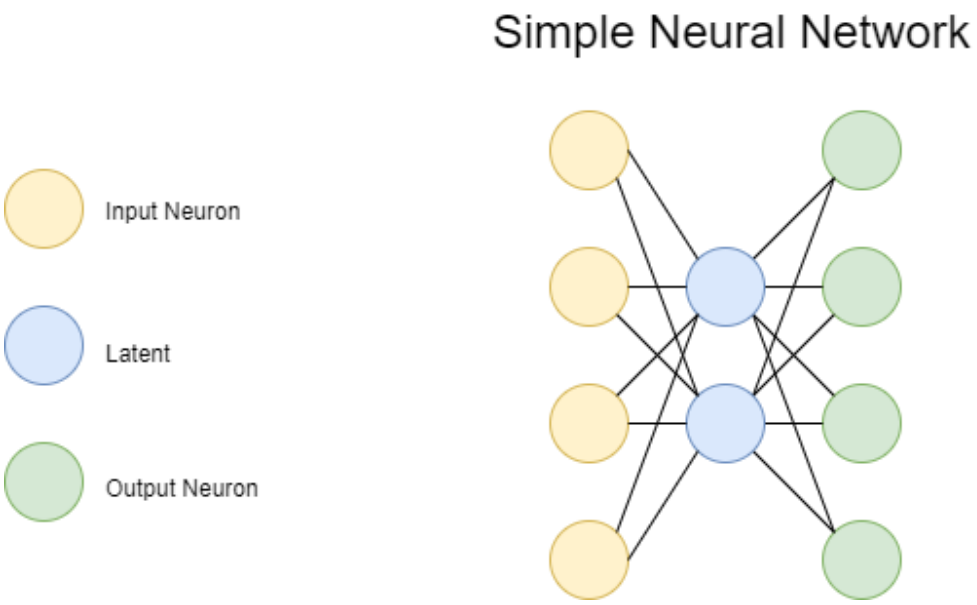


FIGURE 2.1: Simple Neural Network

By leveraging the chain rule from calculus, the algorithm computes the gradient of the error with respect to each model parameter during the backward pass. These gradients indicate how much each parameter should be adjusted to reduce the error. Gradient descent, or its variants like stochastic gradient descent (SGD) and Adam, is used to update the model's weights iteratively. This optimization process continues until convergence, achieving a set of optimized weights that best fit the training data.

Backpropagation's versatility has made it a fundamental tool for training a wide range of neural network architectures, facilitating advancements across various domains. To enhance the understanding of the concepts presented in our thesis, a diagram illustrating 2.1 has been given. The superiority of DL in image-related tasks has been widely demonstrated, showcasing its potential for transformative applications. One key component of DL that holds immense significance for our work is the Autoencoder. Autoencoders have proven to be invaluable in image compression, where they can efficiently learn a compact representation of an image. The Autoencoder is a neural network architecture consisting of an encoder and a decoder. The encoder compresses the input image into a lower-dimensional latent representation, capturing its essential features. The decoder then reconstructs the compressed representation back into the original image. As demonstrated from Chapter 4 to the end of our thesis, we extensively leverage Autoencoders to model the encoder and decoder architectures with parameterized deep neural networks. By harnessing the powerful backpropagation algorithm, we enable these neural networks to learn the optimal encoder-decoder pair tailored to our specific task. The utilization of DNNs throughout our thesis aims to revolutionize image compression, leading to improved transmission efficiency and data storage while preserving essential image details.

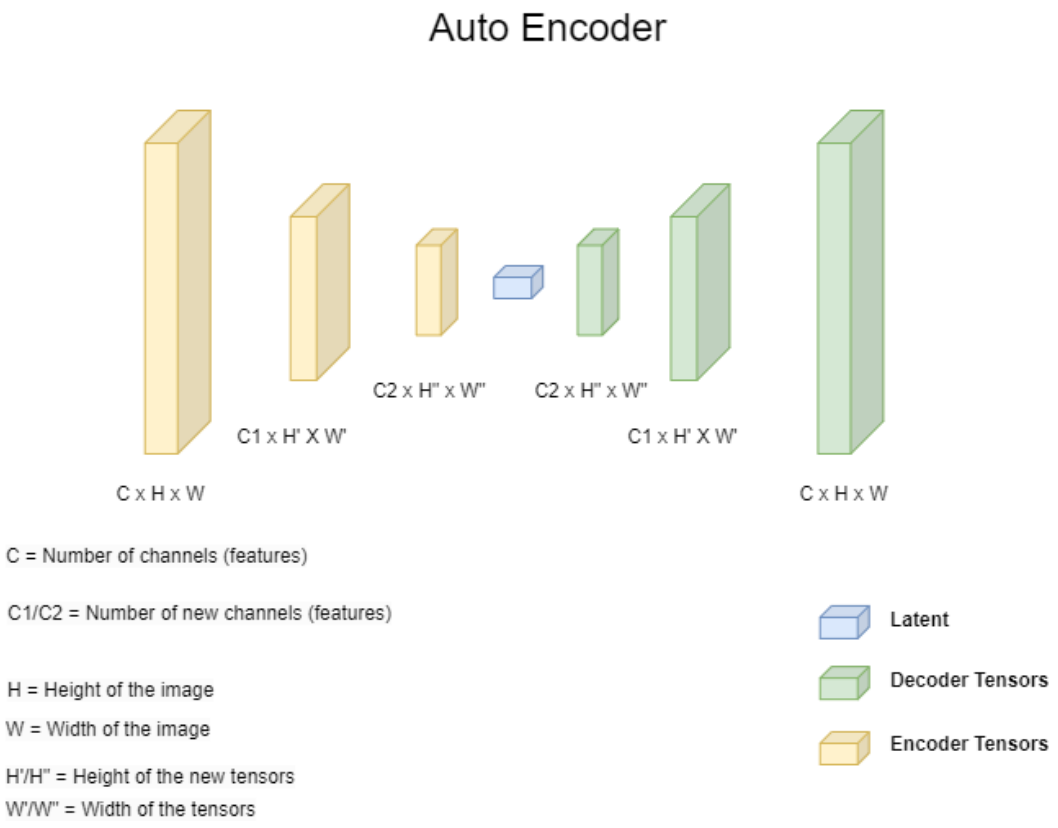


FIGURE 2.2: Autoencoder

To aid in comprehending the architecture and functionality of Autoencoders in our research, an illustrative diagram depicting the structure and operations has been shown 2.2

In our research, we will be adopting Autoencoder-style encoder-decoder deep neural networks to facilitate efficient image compression. Additionally, to simulate the transmission process, we will incorporate a differentiable channel that allows us to analyze the network's performance under various communication conditions.

2.4 JSCC Problem

Consider the JSCC problem of transmitting an independent and identically distributed (i.i.d.) memoryless Gaussian source, $\mathbf{u} \in \mathbb{R}^k$ in $n = \beta k$-uses of a power-constrained additive white Gaussian noise (AWGN) channel. Here, β represents the bandwidth ratio, which denotes the channel usage per source. Although Shannon proved that separation is optimal in asymptotic regions, JSCC outperforms separation-based coding schemes in small blocklength regimes. When $\beta = 1$, Goblick [10] proved that uncoded analogue transmission (simply scaling the source) is optimal in terms of mean square error(MSE) distortion. However, in practical applications, bandwidth mismatch is prevalent. In

general, the received quality of the transmitted image depends on the rate β as well as the channel condition. In digital transmission, in the asymptotic limit, the capacity increases as $\mathrm{R} = \log(1 + SNR)$ with the channel quality.

Hekland *et al.* [14] showed an analytical way to exploit Archimedes' spirals to increase the 'bandwidth' in analogue transmission, which is based on Shannon-Kotel'nikov mappings(SK-mapping). In [15], this idea is applied to practical image transmission. Although it is sub-optimal in terms of bandwidth allocation and power optimization, they showed that the performance improves in high-SNR condition. The price to be paid is the performance loss in low-SNR conditions. Another study [16] uses digital transmission to transmit important values of image/video; however, they tend to use analogue delivery to transmit details. To transmit as much as possible data, they adapt SK-mapping by deriving optimum allocation analogue symbols plus power optimization. The study [17] proposed space-filling curves which are suitable for Gaussian and Laplacian sources, and maximum likelihood (ML) and minimum-mean-squared-error (MMSE) decoding algorithms are derived. In [18] a neural network is constructed with the hyperspherical coordinates of the input space based on deep learning. Proof of variational upper bound for the MSE is given in [19], which models the system as a variational autoencoder(VAE), and the derivation depends on rate-distortion theory. Same authors [20] improved the former paper by exploiting the manifold nature of JSCC, revising hyperspherical coordinates and using a more flexible regularizer. Since the projection onto a complex manifold is a highly discontinuous function, they split the encoder into two arms and a selector. Again, the same authors propose two selection methodologies in [21] one based on the intuitive rule, and the other has a mixture of experts setup. Authors [22] apply fine-tuning techniques to improvement, where recurrent neural network (RNN) based channel AE is proposed. Long-short-term memory (LSTM) cells are used, which are effective for short block lengths. Neural JSCC [23] proposes a method for bit allocation for source and channel coding. Discrete VAE adds noise into the latent codes, and it learns to both compress and error-correcting, given a fixed bit-length and computational budget.

2.5 Deep-JSCC Improvements

Shannon showed that feedback does not increase the channel capacity. However, feedback could be utilized in analogue transmission to get better performance and [24] is the first practical example that exploits channel output feedback to improve the performance, and it outperforms conventional coding designs. Same authors [25] suggest layered representation, which adapts according to bandwidth constraint. Further study [26] indicates that the decoder can estimate the SNR and exploits it to decode the transmitted image

adaptively. The decoder uses the pilot signal sent by the transmitter to estimate the SNR, which will assist in the decoding process. Moreover, benefiting from the estimated SNR, the denoising module (DM) learns SNR-adaptive denoising. Channel degradation is a problem in wireless communication. Since it is subject to change, ad-hoc design can not be employed. To handle this issue, in [27] the authors proposed VAEs instead of autoencoders. And, since the objective is image transmission, instead of PSNR metric, they adapt SSIM metric to set a loss function. Existing work on DL based JSCC trains the designed network to operate under a specific SNR regime. The article [28] proposes attention modules to resource assignment strategy, which dynamically adjusts the compression resources to more critical tasks to overcome this bottleneck. Instead of manually designed resource allocation, dynamic source and channel coding are employed according to the range of SNR. Another lack of end-to-end JSCC overlooks realistic and challenging multipath fading channels. This study [29] combines trainable CNN layers with non-trainable but differentiable layers for channel model by domain expert knowledge such as channel estimation and equalization into decoder with OFDM blocks and improves the overall performance.

2.6 Deep Learning based Compressive Sensing

There are iterative based models to recover vectors from under-determined equations. Deep unfolding proposes neural networks instead of iterative algorithms. This study [30] is the first example to recover sparse vectors from underdetermined equations by using neural networks. Authors [31] extended the study, and suggest a universal denoising function within the algorithm independent from the source distribution. Although there are some constraints to recover vectors, another study [32] is making use of prior knowledge on the structure of vectors in the relevant domain. Therefore, fewer measurements are needed to same accuracy exploiting GANs. Another example [33] in the literature was proposed sparse denoising GANs can recover details. Slepian and Wolf at 1973 [34] showed that an encoder that has no access to the correlated side information could asymptotically achieve the same compression rate as when side information is available at both the encoder and the decoder. Authors [35] proposes rather than relying on hand-crafted source modelling, powerful conditional deep generative model to learn the distributed encoder and decoder. Also, the study [36] showed normalizing flow prior, estimating the distribution of the underlying signal conditioned on the observations. According to the study [37] linear programming, inversion corresponds to finding a latent code vector that explains observed measurements as much as possible.

2.7 Deep Learning for Communications

The article [38] presents and discusses several novel applications of deep learning for the physical layer by assuming a communications system as an autoencoder, end-to-end reconstruction task that jointly optimizes transmitter and receiver parts simultaneously. However, it requires a differentiable channel model. This study [39] proposes a model which alleviates the problem of a differentiable channel model by developing *alternating* algorithm. The receiver is trained according to the true gradient of the loss; however, the transmitter is trained on an approximation of the loss function gradient. According to the study [40], end-to-end training is limited by channel gradient and GANs can learn to mimic the actual channel model by the data-driven learning algorithm, therefore to improve training stability, WGAN is proposed. OFDM extension is showed in [41] end-to-end learning of communications through NN-based autoencoders. In [42], the algorithm iterates between supervised training of the receiver and reinforcement learning-based training of the transmitter. The article [43] compares BLER performance of the learned system with that of a practical baseline. Reinforcement learning is also employed in NN-based AE [44].

Radio transformer networks proposed in [45] for modulation recognition by leveraging spatial transformer networks and introducing new radio domain appropriate transformations. In [46], the authors proposed unsupervised machine learning problem focusing on optimizing reconstruction loss through artificial impairment layers in an autoencoder and introduced several new regularizing layers which emulate common wireless channel impairments.

As opposed to conventional OFDM, which first estimates the channel state information (CSI) explicitly, [47] is proposed a deep learning-based approach to estimate CSI implicitly and recovers the transmitted symbols directly. They showed that it is more robust when fewer training pilots are used, the cyclic prefix is omitted, and nonlinear clipping noise exists. It has already been mentioned that the channel transfer function is hard to model. It subjects to vary with time and location. So, without any prior information about the channel [48] is presented. Conditional generative adversarial networks are used to represent channel effects. In [49], the authors introduced as a communication system without pilots. Instead of information from pilots, the transmitter DNN learns to transform the input data robustly to various channel conditions. The receiver also has two DNN modules for channel information extraction and data recovery.

The authors [50] investigate MIMO detection, which is specially designed by unfolding an iterative algorithm and adding some trainable parameters.

Channel estimation is also another topic that is handled by deep learning tools. There are two examples that the authors [51] and [52] are proposed to channel estimation and pilot design exploiting deep neural networks.

2.8 Image Compression / Denoising

The study [53] presents a set of full-resolution lossy image compression methods based on neural networks. An RNN-based encoder and decoder are demonstrated as binarized, and a neural network for entropy coding. Then, it gives better compression ability. Deep Image Prior [54] is the first example of untrained networks, and building on [55] is an untrained deep neural network that can generate natural images from very few weight parameters. This compresses images into a concise set of network weights, which is similar to wavelet-based thresholding.

U-Net is introduced in [56], which is one of the state of the art denoising model. The architecture of U-Net is designed to handle both the encoding and decoding of the input image in a single network. This is achieved by using a contracting path that captures the context of the input image through a series of convolutional and pooling layers. The context is then used to create a decoding path that generates a segmented image output. The key feature of U-Net is its 'U' shape architecture, which combines the contracting path with an expansive path. This allows the network to preserve the spatial information of the input image while also maintaining a high-level understanding of the overall context. U-Net has become particularly popular in medical image analysis, where it has been used for tasks such as segmentation and cell detection. It has also been applied in other domains such as satellite image analysis and natural disaster detection. And, in this thesis we will use U-Net as a denoiser for images. Overall, U-Net is a powerful and versatile neural network architecture that has become a widely used tool in image processing tasks.

2.9 Nonlinear Transform for Image Transmission

A CNN-based DCT-like transform is proposed in [57] as preprocessing and postprocessing, for image compression. They use convolutional layer, nonlinear mapping and linear transform to get a nonlinear transform. There is a fully connected layer on top of the convolutional layers, and since it is a linear transformation, it could be interpreted as replacing the DCT layer, and the initial part is considered as non-linear preprocessing. The authors in [58] analyse the energy distribution among signal elements. They propose a decorrelation transform which leads to significant gain by diverting the energy

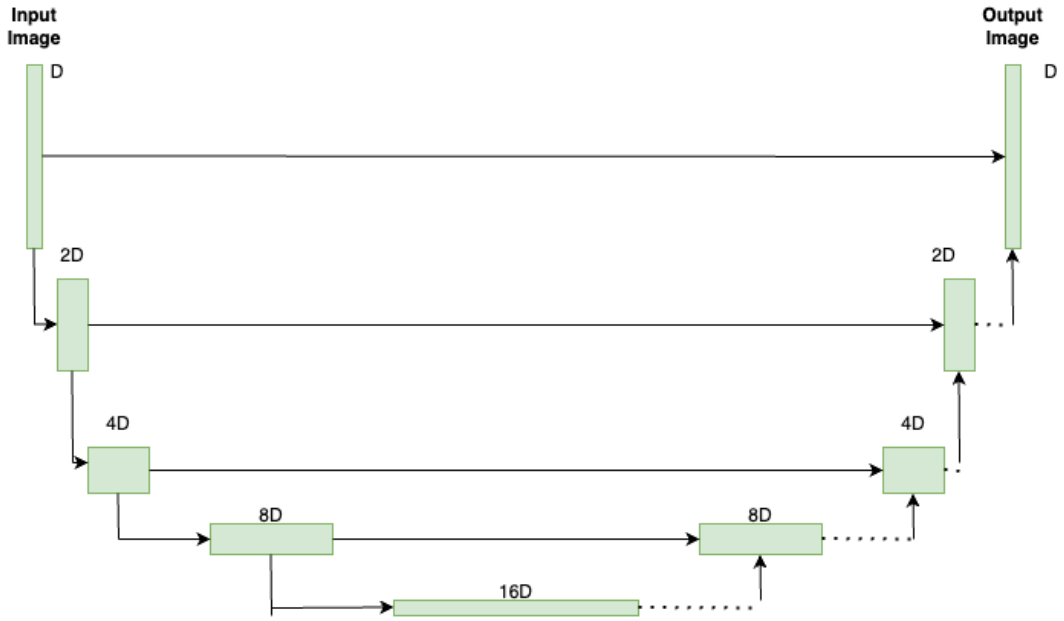


FIGURE 2.3: U-Net

diversity in signal representation. Same authors introduce practical energy modelling schemes in [59] for uncoded-linear transform transmission, again highlighting the signal energy modelling.

i-Wave [60] is another version of a wavelet transform that exploits the power of neural networks. It augments the lifting procedure used in JPEG2000 compression. CNN-based wavelet-like transform is proposed to benefit from neural networks in the *lifting procedure* in wavelet decomposition.

2.10 Others

Transformers are introduced in [61] dispensing with recurrence and convolutions entirely. The study [62] demonstrates that deep generative networks are used to channel estimation. Semantic communication is another topic that attracts more attention nowadays. DeepJSCC work is the very first example of semantic communication - is not interested in transmitting whole data - instead extracts and delivers the minimal amount of information required for the desired task.

2.11 Conclusion

This chapter has provided the necessary preliminaries that will serve as a foundation for the subsequent chapters in this thesis. The literature review in the fields of image

processing and wireless transmission has been presented to emphasize the significance and relevance of this research, which sits at the intersection of these two vibrant communities.

The inclusion of literature from the image processing and wireless transmission community, particularly the ones utilizing deep learning tools, showcases the current state-of-the-art and indicates that these areas are actively evolving and gaining momentum. This underlines the importance of exploring novel approaches to address the challenges and opportunities that arise when combining these domains.

As we progress through this thesis, we will leverage both model-driven and data-driven techniques. Model-driven approaches will allow us to exploit prior knowledge and domain expertise, providing a solid theoretical framework for our research. On the other hand, data-driven techniques, such as deep learning tools, will empower us to harness the vast amounts of available data to gain insights and create powerful predictive models.

By integrating these diverse methodologies, we aim to contribute to both the image processing and wireless transmission communities, bridging the gap between the two and advancing the state-of-the-art in this exciting and interdisciplinary field.

Throughout the following chapters, we will put our focus on experimental evaluations, empirical studies, and critical analyses to demonstrate the effectiveness of our proposed approaches. The ultimate goal is to offer valuable contributions that will benefit not only the academic community but also industry and practical applications in various domains.

Chapter 3

Uncoded Image/Video Transmission

In this chapter, we present an uncoded/analog image/video transmission scheme called ZZ-Cast, and compare it with other analog counterparts proposed in the literature as well as the conventional digital transmission scheme. ZZ-Cast outperforms the well-known SoftCast benchmark, and the digital transmission scheme in low bandwidth and low channel signal to noise ratio (SNR) conditions. The metadata-free version of ZZ-Cast is also presented. Contribution of this chapter is a fair comparison between digital and analog transmission schemes, where we take the metadata transmission cost into account in our evaluation of the uncoded/analog alternatives.

3.1 Introduction

As we have outlined in Chapter 1, JSCC can potentially improve the end-to-end performance of wireless image/video transmission in the practical finite blocklength regime. Theoretical results show that simple symbol-by-symbol uncoded transmission can achieve the optimal Shannon theoretic limit when transmitting Gaussian source samples over an AWGN channel. Potential benefits of such a simple approach for a practical system has been demonstrated in [11], which proposed a simple analog transmission approach to image transmision, combining a linear transform code with a non-linear power allocation scheme. In SoftCast, 2-dimensional discrete cosine transform (2D-DCT) is first applied to the image, which de-correlates the signal. Similar frequency components are grouped in the same chunk and compression is applied by discarding the chunks whose energy is below a threshold. Chunk refers to a spatially localized and disjointed segment or part of an image that has been divided into smaller, manageable components. This threshold is determined according to available channel bandwidth. The remaining chunks, typically corresponding to low-frequency components, are transmitted directly over the channel

by linear scaling. The scaling factors, which correspond to power allocations across the chunks are determined according to empirical variance of the chunks. The power allocation follows the one derived in [63] assuming zero-mean Gaussian distributed data. However, SoftCast requires the transmission of additional metadata in digital form. The receiver must be informed not only the indexes of the transmitted chunks, but also the scaling factor applied for the decoder to be able to estimate the elements of these chunks. Each chunk is arranged to obtain raw OFDM symbols. At the decoder side, the linear least square error (LLSE) estimator is used to decode the DCT coefficients of the transmitted chunks. Since the operations are linear, the reconstruction quality of the image of the decoder, measured in peak signal to noise ratio (PSNR), changes linearly with respect to the channel signal to noise ratio (CSNR). Thanks to this linear behavior the *cliff effect* is not experienced in SoftCast as opposed to conventional digital image transmission.

Over the last years, uncoded transmission has been attracting more attention. This is mainly due to the emergence of new applications ranging from virtual/augmented reality (VR/AR) and online gaming to video conferencing, whose stringent latency constraints are difficult to meet with digital communications, where efficient compression codecs introduces significant delays. Several improved version of SoftCast has been introduced over the recent years [14,64]. A theoretical analysis in [59] highlighting the importance of the decorrelation transform and the energy modeling of the underlying signals. Wavelet transform is employed [64] and Karhunen-Loeve transform (KLT) is applied in [65] as the decorrelation transform instead of 2D-DCT which was proposed in SoftCast. CG-Cast [66] and G-Cast [67] propose improving the visual quality of the transmitted images based on gradient values. They proposed to improve image quality by transmitting gradient values to the receiver. SparseCast [68] is proposed to reduce the bandwidth requirements via exploiting compressive sensing (CS). After applying 2D-DCT on image blocks and thresholding, a unique grouping of the coefficients is employed. Then, each vector is multiplied by a pseudo-random measurement matrix whose size depends on the sparsity level of the corresponding vector. The receiver employs a minimum mean squared error (MMSE) estimation or approximate message passing (AMP) algorithm according to the sparsity level of the vector. ECast [69] focuses on joint sub-carrier matching and power allocation to optimal end-to-end performance of transmission.

To reduce the metadata overhead, an adaptive, L-shaped chunk division method is proposed in [59]. It based on non-uniform chunk division and tried to group similar components in the same chunk that their dimension is not the same. Although the simulations show significant reduction in metadata, signal energy distribution modeling is required, which brings overhead.

Zhang et al. [70] proposes a blind data detection algorithm, which does not need to transmit any metadata about the power scaling factors. The squared amplitude of received signals is used at the receiver to scale them back. At the beginning of the transmission, only the normalization factor must be sent to the receiver directly, the resulting scheme that is in [70] is almost metadata-free. However, this results in a biased estimate of the transmitted signal. Zong et al. [71] proposed an approximately unbiased estimator. In order to reduce the estimation error of their scheme, they consider chunk-based estimation. This results in a trade-off between the signal energy distribution, which becomes less accurate with increasing chunk size, and the estimation error. They numerically analyze the effect of various chunk sizes on the performance.

When we summarize the literature, the main ideas to introduce ZZCast are efficient grouping, reduce metadata burden, and decrease the bandwidth requirements.

3.1.1 A new uncoded transmission scheme: ZZCast

In this section, we propose a novel analog image/video transmission scheme, and show that it outperforms previously known analog schemes in all the scenarios considered. Below we explain the novel aspects of the proposed ZZCast in detail.

1) Zigzag Encoding: The best performance in SoftCast is achieved when a distinct scaling factor is applied to each DCT coefficient. However, this would result in a huge amount of metadata, making the scheme impractical. Therefore, SoftCast groups the coefficients into chunks. Since only one single scaling coefficient is used per chunk, it is important to group DCT coefficients that are more likely to have similar energy values into the same chunk. An adaptive L-shaped chunk-optimization scheme is proposed in [59]. However, this scheme has significant computational complexity. An alternative curve-fitting approach is also proposed in [59], which still requires the estimation of the curve parameters.

Here, we consider a simple zigzag encoding scheme, which orders the DCT coefficients of the whole frame in the zigzag order starting from the lowest frequency, see 3.1. Zigzag ordering of the DCT coefficients sorts them in the increasing order in frequency. Natural images are more likely to have more significant values for low-frequency components. Accordingly, we convert the 2D DCT matrix into a vector following the zigzag order, which is likely to sort the DCT coefficients in descending order in terms of their magnitudes. We have also observed that the first coefficient of the DCT transform, which corresponds to the DC component, typically has a significantly larger magnitude. Therefore, in order to remove this coefficient, we subtract the mean value of the pixel intensities from all the pixels prior to the DCT transform. This results in a zero-mean image matrix. The

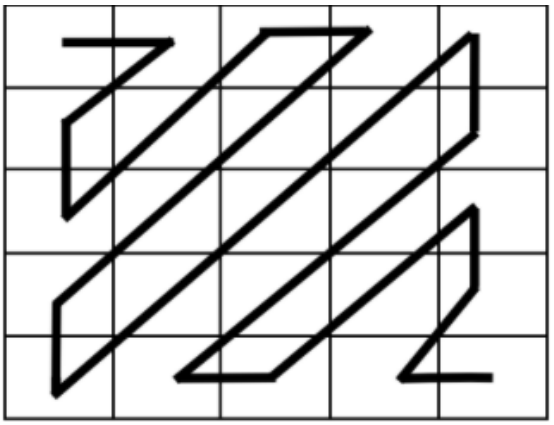


FIGURE 3.1: ZigZag encoding

mean value is sent to the decoder separately as metadata. This can also be considered as treating the first DCT coefficient as a separate chunk on its own.

2) Co-Utilization of Two Different Bandwidth Reduction Techniques: One of the challenges of uncoded transmission is the lack of flexibility in matching the source dimension to the available channel bandwidth. One simple way to reduce the channel bandwidth that has been commonly used in the literature is to use thresholding to remove chunks of coefficients that have low energy [11]. In [68], thresholding is applied to individual DCT coefficients, and the remaining sparse vectors are transmitted using a sparsity-aware encoding scheme, where the allocated channel bandwidth to each chunk depends on its sparsity. In ZZCast, in addition to thresholding of the individual DCT coefficients, we will also remove complete chunks that are sparse; that is, chunks with energy values less than a certain threshold, which we call the *zero factor*. This not only allows the encoder to satisfy the bandwidth constraint, but also decreases the amount of metadata it needs to transmit. The zero factor is one of the parameters of the proposed ZZCast scheme that must be specified before transmission.

3) Measurement ratio The remaining chunks are vectors with different levels of sparsity due to thresholding. Following [68], we will use linear projection onto a lower dimension using a predetermined measurement matrix. Therefore, each channel input symbol corresponds to a different linear combination of the non-zero values in a chunk. CS theory demonstrates that a system of underdetermined equations can be solved with high probability if the solution is sparse. The question of determining the optimal number of measurements needed to recover a sparse signal arises. It is well-known that a higher number of measurements generally leads to better signal recovery quality [72]. However, in our specific setting, where the measurements are transmitted over a wireless channel, increasing the number of samples necessitates additional channel bandwidth. To strike a balance between signal recovery quality and bandwidth utilization, we carefully selected

a measurement ratio of 2. The measurement ratio indicates the number of symbols transmitted per non-zero element in the vector. This choice was informed by empirical experiments conducted to evaluate the trade-off between reconstruction performance and bandwidth requirements. While having more measurements can indeed enhance the quality of the reconstructed image, it also escalates the demand on available bandwidth. We observed through our experiments that beyond a certain point, increasing the number of measurements no longer yields significant improvements in the overall performance of the system. Thus, our decision to employ a measurement ratio of 2 was based on practical considerations, optimizing both signal recovery and bandwidth utilization in our specific image processing and wireless transmission context. By conducting numerous experiments, we were able to identify an appropriate balance that ensures efficient utilization of resources while achieving satisfactory signal reconstruction results.

We note here that run-length coding could be employed in our sparsity level metadata. They show a high correlation and run-length coding decreases the total metadata. Secondly, variance values can be further compressed via Huffman coding, which can significantly reduce the amount of metadata. However, we have set metadata of the sparsity level for 2-bits and variance value 32-bits per chunks.

PSNR and the structural similarity index (SSIM) are the most common metrics in image processing. The former depends on the minimum square error (MSE) between the original and the reconstructed signal. The latter is an index for the perceived quality. Chunk size is another important parameter that directly relates to the output SNR and the amount of metadata. When we decrease the chunk size, it gives better results in terms of end-to-end quality; however, it increases the amount of metadata we need to deliver reliably to the receiver. So, there is a trade-off between the chunk size and the amount of metadata.

3.2 System Model

We are expressing the core optimization problem in an alternative form below:

$$\begin{aligned} \text{Maximize} \quad & \text{PSNR}(X, X') \\ \text{subject to} \quad & E[||Y||^2] \leq \bar{P} \\ & \text{k} \leq \text{available bandwidth} \end{aligned} \tag{3.1}$$

In this formulation, we seek to maximize the Peak Signal-to-Noise Ratio (PSNR) between the original signal $\mathbf{X}$ and its reconstructed version $\mathbf{X}'$. The variables $\mathbf{X}$ and $\mathbf{X}'$ represent the original and reconstructed signals, respectively.

The optimization is subject to two constraints. The first constraint ensures that the expected power of the encoded signal $\mathbf{Y}$ is bounded by $\bar{P}$. This constraint helps control the power consumption or transmission capabilities of the encoding process.

The second constraint limits the length of the encoded signal $\mathbf{Y}$ to be no greater than BW. This constraint is usually associated with communication or storage constraints, where BW represents the available bandwidth for transmitting or storing the encoded signal.

By solving this optimization problem, we aim to find the optimal encoding/decoding scheme that maximizes the quality of the reconstructed signal while satisfying the given power and bandwidth constraints.

3.2.1 ZZCast Algorithm

Here we present the proposed ZZCast algorithm in detail. We will consider the transmission of gray-scale images and videos; however, the proposed techniques can be used for color images by applying the algorithm separately on the R, G, and B planes.

In the ZZCast algorithm, we normalize the images before applying the 2D-DCT algorithm. This is achieved by subtracting the average value of the pixel intensities from all the pixels. This average value in general corresponds to the average brightness of the image, and hence, it is image-dependent. Therefore, it has to be provided to the receiver as metadata so that it can be added to the reconstruction at the very end of the decoding process. The purpose of this normalization is to make the first DCT coefficient, which represents the DC-component, zero. We have observed that this provides a significant improvement in the end-to-end quality, as normally the DC-component is much larger than other DCT coefficients, which results in inefficient power allocation.

We can also simply subtract a constant value, independent of the image. This will reduce the metadata requirement, with only a slight increase in the average distortion according to our simulation results.

After the normalization, we apply 2D-DCT to the whole image. As in other analog transmission schemes in the literature, we will then group the DCT coefficients into *chunks*. These chunks form the basis of compression (by removal of chunks or application of CS) as well as power allocation. Several different techniques have been used in the literature to form the chunks. We use a zigzag scheme to first convert the two-dimensional DCT matrix into a vector. The result is a coefficient vector with entries ordered in vertical and horizontal spatial frequencies. This resulting vector is divided into N chunks of equal size, and each of which consists of M values. We then set the values that are smaller than

the *sparsity threshold*, τ_k, to zero. This results in sparse vectors. We discard the chunks which have less number of nonzero values than another parameter called *zero-factor*. The chunks size, sparsity threshold, and zero-factor parameters that must be selected according to the available bandwidth. These two steps serve as compression, and reduces the required channel bandwidth. The same frequency components of all the DCT blocks are stacked into the same blocks. Most natural images have DCT energy focused at low-frequency components, therefore high-frequency components will likely be set to zero and have minimal effect on the image quality. As we have discussed above, metadata size is another design issue, as the variance value for each block must be transmitted correctly to the receiver. If we decrease the number of transmitted blocks, this would reduce the amount of meta-data as a result. Figure 3.2 shows the effect of chunk size in case no compression and example image is selected Lenna. Although the transmission is uncoded, it is not devoid of processing. As we have previously discussed, when the source and channel bandwidths are perfectly matched, uncoded transmission becomes optimal with a simple scaling [10], which can be considered a form of 'processing'. However, in practical scenarios, bandwidth mismatches are common in wireless image transmission. In this case, we opt for a suboptimal solution to assess its performance.

To determine the scaling factors required for the suboptimal solution, we explicitly formulate the objective function and derive the scaling coefficients. This approach allows us to carefully analyze the trade-offs between compression, distortion, and transmission efficiency. By explicitly stating the objective function, we can gain insights into the underlying processes that influence the transmission and make informed decisions regarding the scaling coefficients.

Lemma 3.1. *Given L data vectors $x_1, \ldots, x_L$ each consisting of $m_i = |x_i|$ samples from a zero-mean Gaussian distribution with variance λ_i for $i = 1, ..., L$, the linear encoder that minimizes the linear least squares estimation (LLSE) in the presence of AWGN scales the i-th vector by*

$$g_i = \lambda_i^{-1/4} \left(\sqrt{\frac{P}{M_i \sum_j \lambda_j}} \right), \quad i = 1, \ldots, N. \tag{3.2}$$

Proof : We start by transmitting values in chunk i, denoted as x_i. Each value is scaled by a factor g_i, and the receiver receives $y_i = g_i * x_i$, where g_i is the scaling factor for chunk i. Additionally, a zero-mean random noise with variance σ^2 (same for all chunks) is added to the received signal, resulting in $\hat{y_i} = y_i + n$. The receiver then decodes $\hat{x_i} = \hat{y_i}/g_i = x_i + n/g_i$.

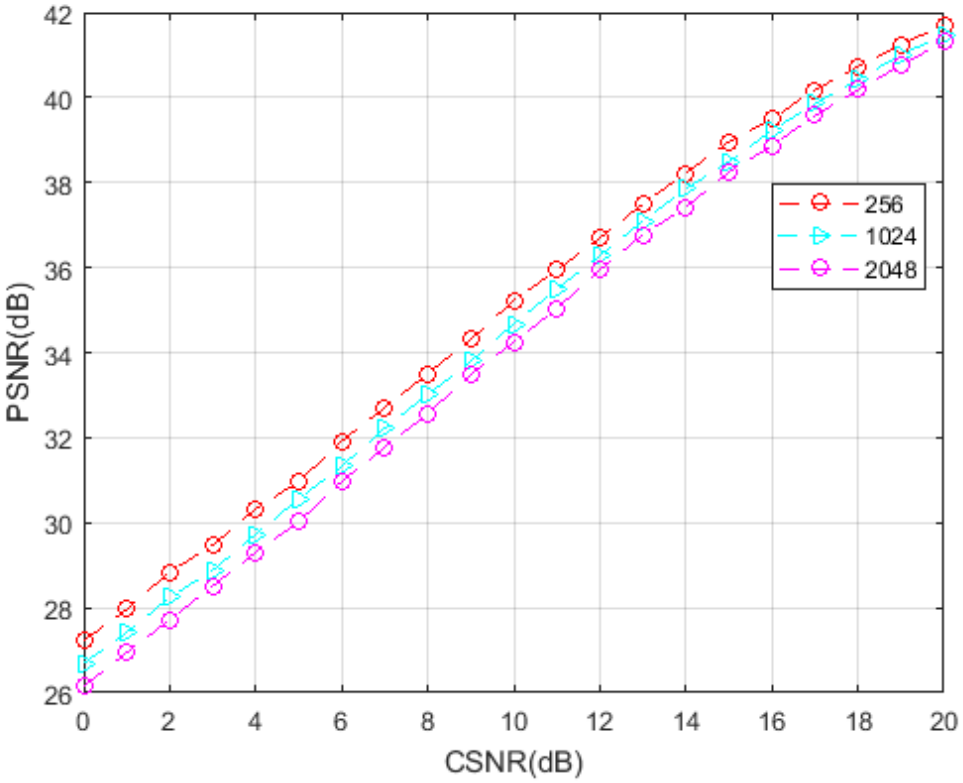


FIGURE 3.2: Effect of Chunk Size

To quantify the error, we define the expected mean square error as

$$\text{err} = E[\sum_i (\hat{x_i} - x_i)^2] = \sum_i E[n^2]/g_i^2 = \sum_i \sigma^2/g_i^2. \tag{3.3}$$

Let $\lambda_i = M_i * E[x_i^2]$ represent the weighted power of chunk x_i, and $\mu_i = E[y_i^2]$ denote its power after applying the gain g_i. We have a total power budget P.

$$\begin{aligned} \min \quad & \sigma^2 M_i \sum_i \lambda_i/\mu_i \\ \text{subject to} \quad & \sum_i \mu_i \le P \\ & \mu_i \ge 0. \end{aligned} \tag{3.4}$$

To solve this constrained optimization problem, we introduce the Lagrangian

$$L = \sigma^2 M_i \sum_i \lambda_i/\mu_i + \gamma \left(\sum_i \mu_i - P \right). \tag{3.5}$$

By differentiating L with respect to μ_i and γ and setting the derivatives to zero, we find that

$$\sqrt{\gamma} = M_i \sum_j \sqrt{\lambda_j \sigma^2}/P \tag{3.6}$$

and

$$\mu_i = \sqrt{\lambda_i \sigma^2/\gamma} = P\sqrt{\lambda_i}/M_i \sum_j \sqrt{\lambda_j}. \tag{3.7}$$

Consequently, the scaling factor g_i is given by

$$g_i = \sqrt{\mu_i/\lambda_i} = \sqrt{P/(\sqrt{\lambda_i} M_i \sum_j \sqrt{\lambda_j})}. \tag{3.8}$$

The sparsity level of vector x_j refers to the number of its non-zero entries, and is denoted by k_j, i.e. $\|x\|_0 \triangleq k_j$. To perform power allocation according to [63], the empirical mean of vector x_j is first subtracted from all its entries to obtain a zero-mean vector. This mean value must be sent to the receiver as metadata. However, in our algorithm, zigzag composition creates almost zero-mean vectors that reduces the amount of metadata approximately by half.

We set $k_j \ll \mu_j \ll N$, where μ_j is the number of measurements sent for vector x_j, which provides a trade-off between the accuracy and channel bandwidth. Note that, to distinguish the received signals for each of the vectors, the decoder needs to know μ_j's, which increases the meta-data size. To reduce the meta-data, we choose μ_j's from a set of S predefined measurement levels. For each block, sparsity level is chosen as the nearest number in the set that is greater than k_j. Size of metadata for each block is $\log_2 S$ bits. If the total number of nonzero values is equal to the block size, we do not need to sparsify the blocks, we only transmit them.

Channel symbols are formed by pairing consecutive elements as the in-phase and quadrature components of a complex symbol. This keeps linearity between input signals and output signals.

The receiver employs MMSE/AMP estimation according to the sparsity level, which is transmitted as metadata. If it is a fully non-zero vector, it employs MMSE estimation according to the formula derived in [63] exploiting the variance values. Otherwise, it employs AMP recovery using the sparsity level information. If the lowest sparsity level information is sent, the receiver reconstructs the vector as a zero vector because there is no transmitted values at all. When the estimation is completed, it applies inverse of the zigzag transform, followed by inverse 2D-DCT, and adds the mean value that was subtracted before at the beginning, respectively.

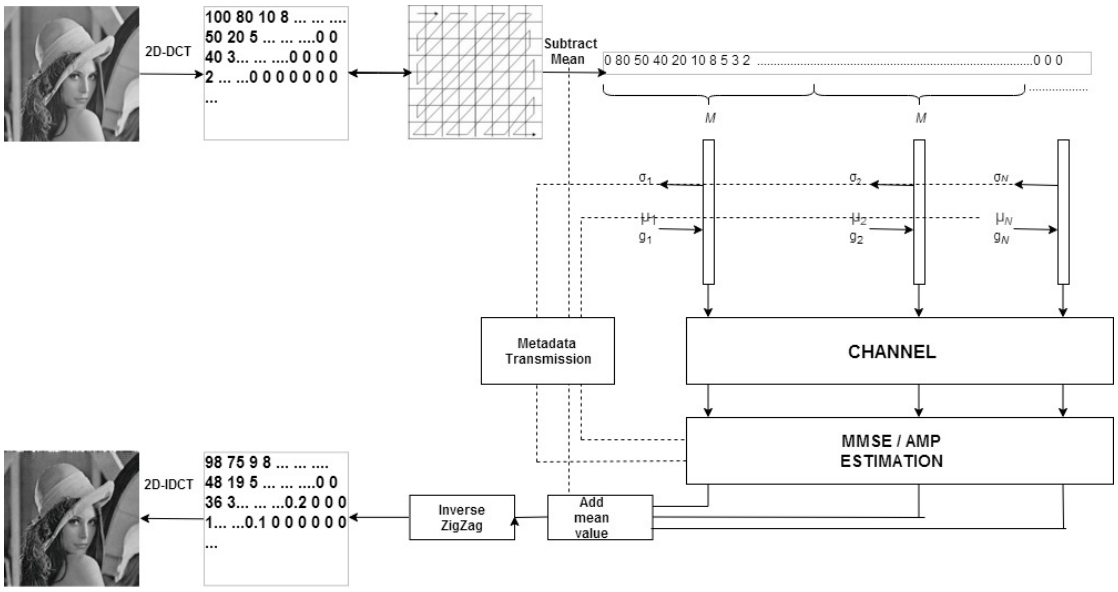


FIGURE 3.3: Block diagram of the proposed ZZCast transmission scheme

Algorithm 1 ZZCast Algorithm

1: Apply 2D-DCT
2: Apply Zigzag
3: Subtract mean
4: Divide the input data into chunks, each containing M elements
5: Determine the type of vector
6: Calculate power coefficients
7: Apply power allocation
8: Transmit through channel
9: Metadata is transmitted through reliable channel
10: Choose estimator according to metadata
11: Apply Estimation
12: Add mean value
13: Apply inverse Zigzag
14: Apply inverse 2D-DCT

3.3 Numerical Results

In this section, ZZCast is compared with SoftCast [11], SparseCast [68], and digital compression with digital transmission. We have chosen JPEG as the digital compression scheme, followed LDPC channel code. The type of modulation scheme has an impact on the result of digital transmission. Therefore, different modulation schemes with different code rate combinations are compared. As we mentioned in the introduction, the digital scheme suffers from the so-called cliff effect. An error in decoding results in either decoding a wrong or invalid channel codeword, which is then mapped to the wrong source codeword, destroying the reconstruction completely. Therefore, the transmitter must ensure that the data rate of the code used is sufficiently below the capacity of the

channel to guarantee correct decoding with high probability. We have completed code pair simulations for LDPC + M-QAM modulation schemes using rate 1/2 as (4096,8192) and 2/3 as (4096,6144) and observed the frame error rate for different CSNR values. We find the CSNR value to achieve a frame error rate that is close to zero, which is then called the 'threshold' CSNR. We assume that if the CSNR is greater than this threshold, no error is experienced during channel transmission. Otherwise, we assume that digital transmission is not reliable. So, we calculate the available bit-budget according to the rate, modulation scheme and available bandwidth. Then, we employ the JPEG algorithm to compress the input image within the pre-calculated bit-budget. We assume that we are able to employ this operation only if the CSNR is above the threshold. Here, we would like to define another metric the Structural Similarity Index (SSIM) is a widely used metric for measuring the similarity between two images. It takes into account luminance, contrast, and structure to provide a more comprehensive evaluation of image similarity compared to pixel-wise metrics.

$$\mathrm{SSIM}(X,\hat{X}) = \frac{(2\mu_X\mu_{\hat{X}} + C_1)(2\sigma_{X\hat{X}} + C_2)}{(\mu_X^2 + \mu_{\hat{X}}^2 + C_1)(\sigma_X^2 + \sigma_{\hat{X}}^2 + C_2)} \tag{3.9}$$

where X is the input image and $\hat{X}$ is the output (or reconstructed) image. μ_X and $\mu_{\hat{X}}$ are the mean intensities of images X and $\hat{X}$ respectively. σ_X and $\sigma_{\hat{X}}$ are the standard deviations of intensities of images X and $\hat{X}$ respectively. $\sigma_{X\hat{X}}$ is the covariance between images X and $\hat{X}$. C_1 and C_2 are constants used for numerical stability.

Kodak-24 Images We apply the considered schemes on the Kodak image set, which consists of 24 high-resolution natural images of size 768×512. We have simulated them under four different bandwidth constraints: 150K, 100K, 75K, and 50K. Metadata is not counted for the bandwidth and it is sent separately. Table 3.1 shows how much metadata is sent along with symbols. These are the exact number of bits ZZ-Cast uses. Notice that mean and variance values are sent using 32-bit floating point representation. We will add the metadata in bits to obtain total bit budget to be used in digital transmission for a fair comparison with analog transmission. We note here that, this comparison is new in the literature and it considers metadata as part of the bandwidth that has been overlooked so far. JPEG compresses the image according to this total bit budget and we calculate the PSNR and SSIM values according to the output of the JPEG compression. For example, if we use 150K bandwidth : QPSK (2 bit/symbol) with 1/2 rate LDPC results in 2 bit/symbol $\times$ 150000 symbols $\times$ 0.5 = 150000 bits, plus 46185 bits that would be used by ZZCast for metadata. JPEG must compress the image to 196185 bits.

Particular Example: Lenna As we discussed earlier, there is a trade-off between the chunk size and the amount of metadata. Chunk sizes of different schemes are tuned to get equal amount of metadata. ZZCast's chunk size is selected as 256 whereas SoftCast

Channel bandwidth (in symbols) :	150K	100K	75K	50K
Metadata (in bits) :	46185	36283	28702	21459

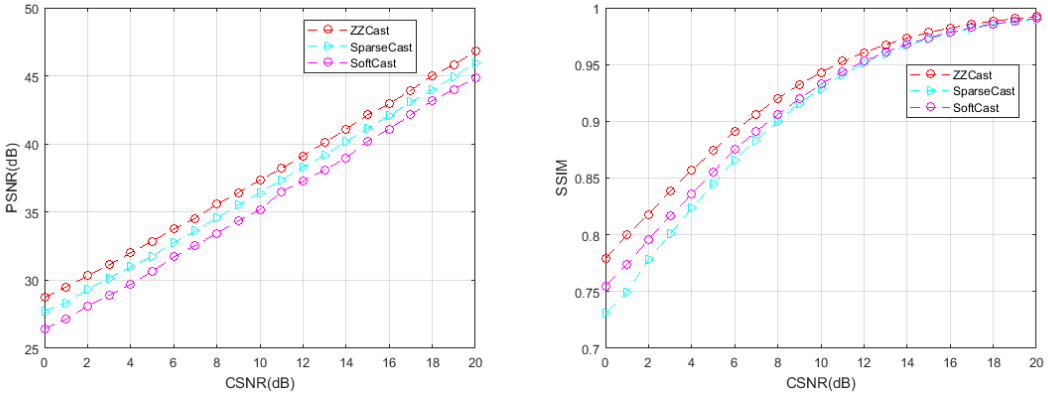
TABLE 3.1: Amount of Metadata that must be transmitted by ZZCast for different bandwidth constraints.

and Sparsecast's chunk sizes are selected as 512. Experiments are done through four different bandwidths i.e. 131K, 75K, 50K, and 15K. In this section, metadata is also included to total bit budget. JPEG compresses according to this total bit budget the image and the output value is again PSNR and SSIM. For example, 75K Lenna : BPSK (1 bit/symbol) with 1/2 rate LDPC 1 bit/symbol$\times$75000 symbols$\times$0.5 = 37500 bits, plus 25824 bits from metadata. JPEG must compress the image under 63324 bits.

In the 'No Compression' case, the bandwidth equals the number of pixels, i.e. 131K. Results show that ZZ-Cast outperforms other schemes, as it employs the best grouping strategy.

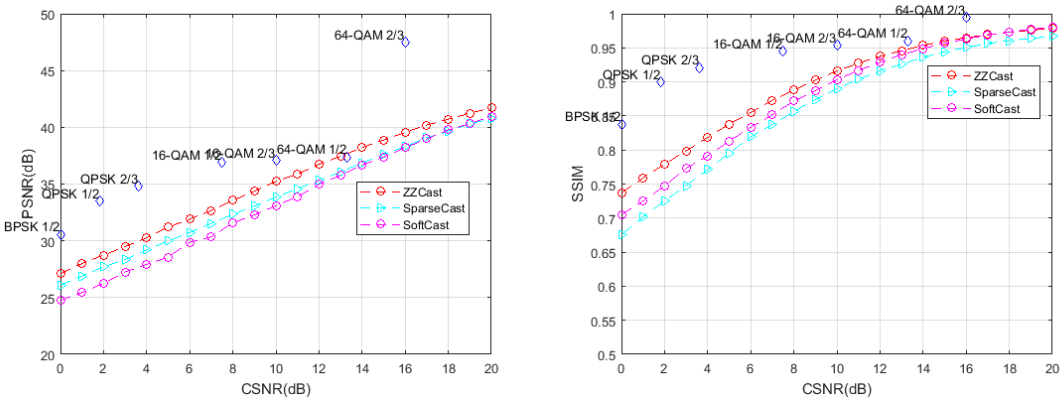
Experiment	SoftCast	SparseCast	ZZ-Cast
75K Lenna	19456	16896	25248
50K Lenna	12928	16896	18784
15K Lenna	4416	NW	6368

TABLE 3.2: The amount of metadata used in bits.



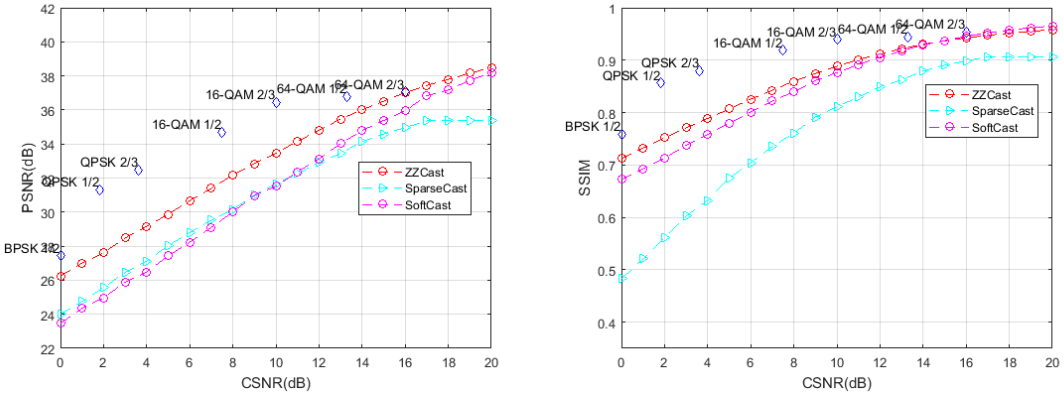
PSNR SSIM

FIGURE 3.4: Lenna - 131K Symbol Length



PSNR SSIM

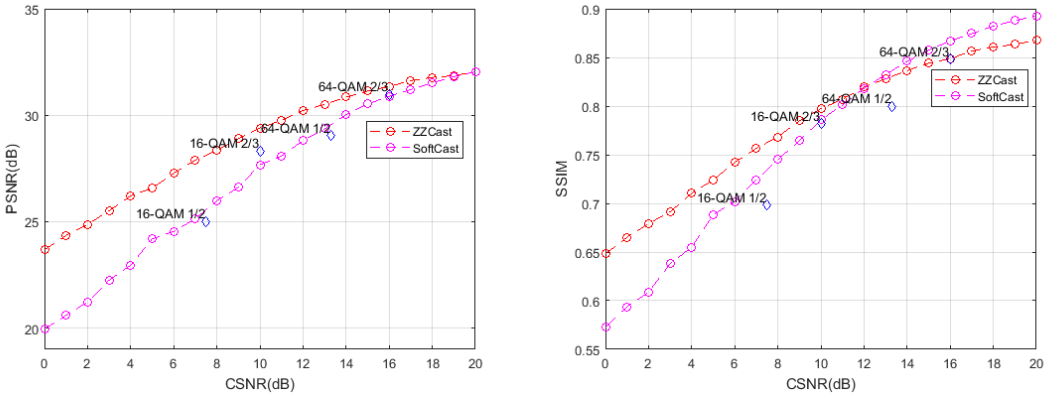
FIGURE 3.5: Lenna - 75K Symbol Length



PSNR SSIM

FIGURE 3.6: Lenna - 50K Symbol Length

Next figures are extreme cases. SoftCast and ZZCast results are presented since digital scheme and SparseCast are not able to operate under this bandwidth constraint. Symbol Length is 15K and metadata is 6368-bits. Digital scheme with 16-QAM is showed only since the other type of modulation schemes with JPEG are not working. We would like to emphasize that when the image wanted to be compressed below 5 percent of its original dimension, digital schemes are not working. That is the power of ZZCast.



PSNR SSIM

FIGURE 3.7: Lenna - 15K Symbol Length

According to results, ZZCast outperforms other uncoded modulation transmission schemes. Although digital scheme beats it in the high bandwidth and high SNR conditions, ZZ-Cast surpasses digital scheme in the low bandwidth (i.e. 15K) and low SNR cases.

We present herein a performance evaluation for the fading channel, demonstrating the superiority of ZZCast over SoftCast across two distinct bandwidth scenarios. This compelling outcome underscores ZZCast's potential as an image/video codec system for wireless image transmission. Its inherent simplicity, ease of implementation, graceful degradation, and remarkable performance collectively endorse ZZCast as an advantageous choice.

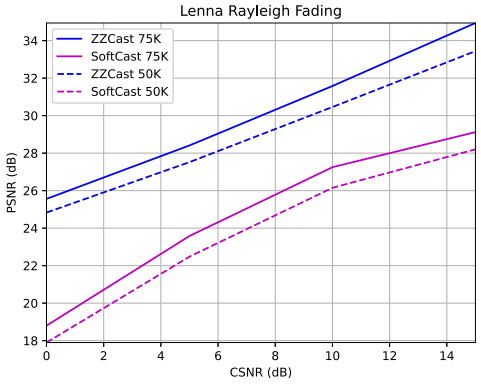


FIGURE 3.8: Fading Channel

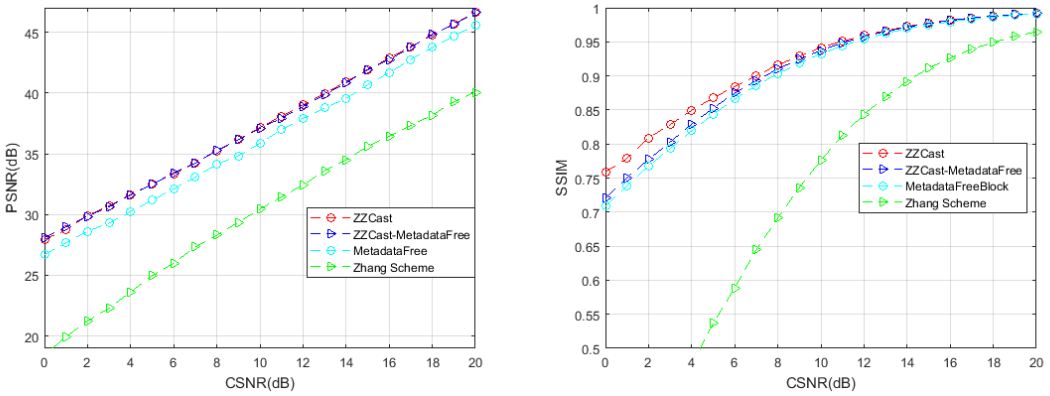
3.4 Reducing Metadata

Section 3 shows that metadata puts a huge burden on the end-to-end system. It decreases the actual bandwidth capacity since we have to dedicate part of the bandwidth we use to send the signal to the metadata. Secondly, it is sent to the receiver by using digital transmission, which means that there is no complete elimination of the cliff effect. Therefore, two studies [70, 71] are proposed to mitigate the burden of metadata. The first one [70] proposes blind estimation method to estimate received signal, and the other [71] extends this idea to chunk-based blind estimation method. They don't employ any compression. In other words, number of channel use equals number of channel source. So, in order to compare them with ZZCast, we used bandwidth constrained as 131072 symbols (i.e. image size). ZZCast allocates these symbols into actual transmitted symbols and and the amount of metadata transmitted. Note that, ZZCast's metadata burden to bandwidth constraint inversely proportional with CSNR because of the fact that when CSNR increases achievable modulation transmission schemes enables more bits. For example, when CSNR is 0dB, we employ BPSK 1/2 which needs a 10K symbols for transmitting 5K bits; however, the same amount of bits could be transmitted if we employ QPSK 2/3 around 4K symbols in case CSNR is 6dB. Therefore, we keep 131072 symbols for experiments but the amount of transmitted pixels are adapted based on CSNR.

Metadata Free ZZCast: In this part, we would like to introduce another version of ZZCast which is completely metadata-free. There are three sources of metadata in ZZCast: mean values of chunks, variance values of chunks and sparsity levels. They must be sent to the receiver correctly. However, in the nature of the zigzag scheme, there is no need to transmit any mean value since it makes chunks more or less zero mean. The only mean value, which is subtracted before the zigzag operation must be sent. In order to get rid of this value, we could set it as 128. In [71], the authors suggest that power allocation could be done according to norm values of chunks instead of variance values of chunks. So, if the receiver is able to estimate the chunks accordingly, there is no need to transmit any metadata related to variance values. Lastly, sparsity level information must be sent to the receiver. However, if we set a procedure beforehand, we don't need to transmit it anymore. Instead of adaptive vectors in the original ZZCast, we fix the number of vectors that will be transmitted according to bandwidth constraint. Bandwidth is divided by three. Firstly, vectors are sent fully non-zero vectors until one third of bandwidth usage. Secondly, vectors are sent through compression applied by first sparsity level until two third of bandwidth usage. Thirdly, vectors are sent through compression applied by second sparsity level. Lastly, the remaining pixels are eliminated

so as to satisfy bandwidth constraint. And, we employ the proposed estimation [71] at the receiver, which is metadata-free ZZCast.

We present the results of metadata-free schemes and original ZZCast which metadata has already been considered into bandwidth constraint.



PSNR SSIM

131 K Lenna

FIGURE 3.9: Comparison between metadata-free schemes

These results suggest that metadata increases the performance when we look into the original ZZCast and metadata-free ZZCast. However, there is no significant difference between them. Moreover, the metadata-free version enables us to use our algorithm below the 0-dB in which we do not rely on digital transmission.

3.5 Video Transmission

In this section, we present a modification of our novel zigzag algorithm for video transmission.

3.5.1 Depth ZigZag Algorithm

Group of pictures (GoP) is another parameter that arises in video coding/transmission. It specifies how many frames are coded and transmitted jointly. To exploit temporal correlations between frames, some joint non-linear transform, e.g., 3D-DCT, can be applied to a GoP. The correlation between frames is taken into account through motion compensation and motion estimation in the digital compression schemes. In order to take the similar components in the video, we are grouping elements according to their

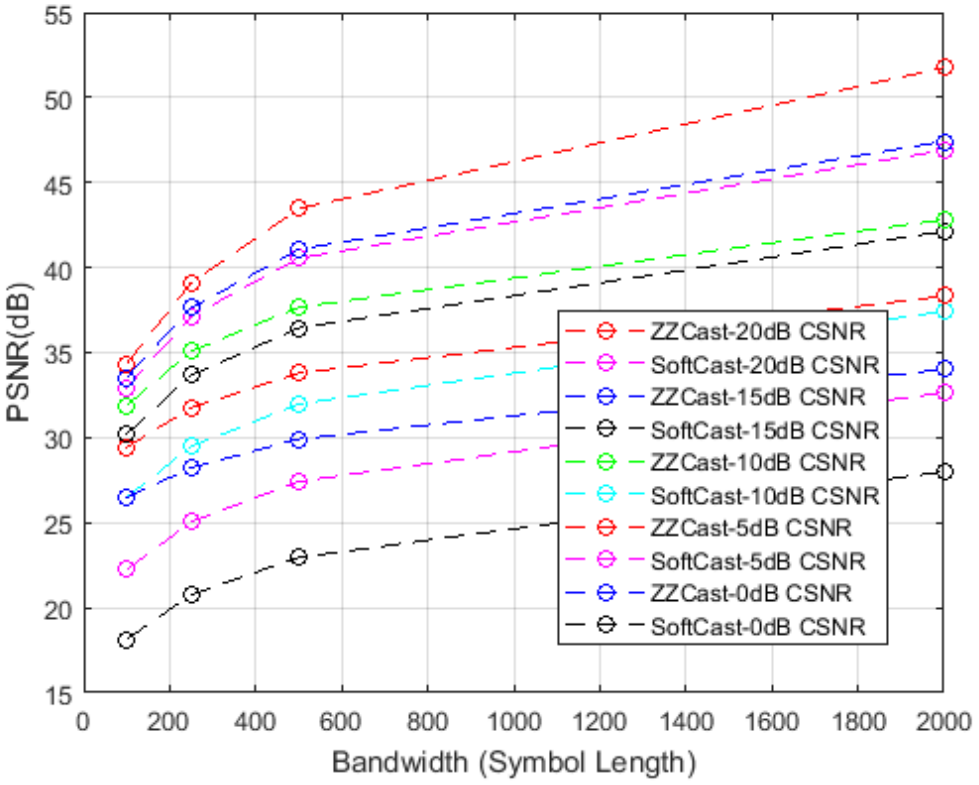


FIGURE 3.10: Akiyo - ZZCast vs SoftCast

GoP as well. For example, we choose GoP as 4 and we start chunking 1st, 5th and 9th frame consecutively since they tend to have more similar values.

We present the results of ZZCast and SoftCast with different bandwidth and CSNR values in 3.10. It can be seen that ZZCast outperforms SoftCast by more than 5dB. As in the image transmission case, we compare our results with up to date digital video codec HEVC software for video encoder and decoder, where the *encoder lowdelay main.cfg* file is used. Figure 3.11 presents the results. Although the digital scheme has better results than ZZCast, graceful degradation is not observed. As we have presented before, metadata-free ZZCast enables us to apply our algorithm when CSNR is lower than 0 dB. We present a 40K bandwidth constraint in Figure 13, which HEVC still can operate the video at -10 dB assuming that Shannon capacity is used instead channel coding and modulation scheme experiments. HEVC still outperforms metadata-free ZZCast; however, please note that Shannon capacity is assumed. If the channel is worser than -10 dB, even if Shannon capacity is assumed, HEVC is not able to operate anymore, but metadata-free ZZCast does. That is the advantage of ZZCast over HEVC.

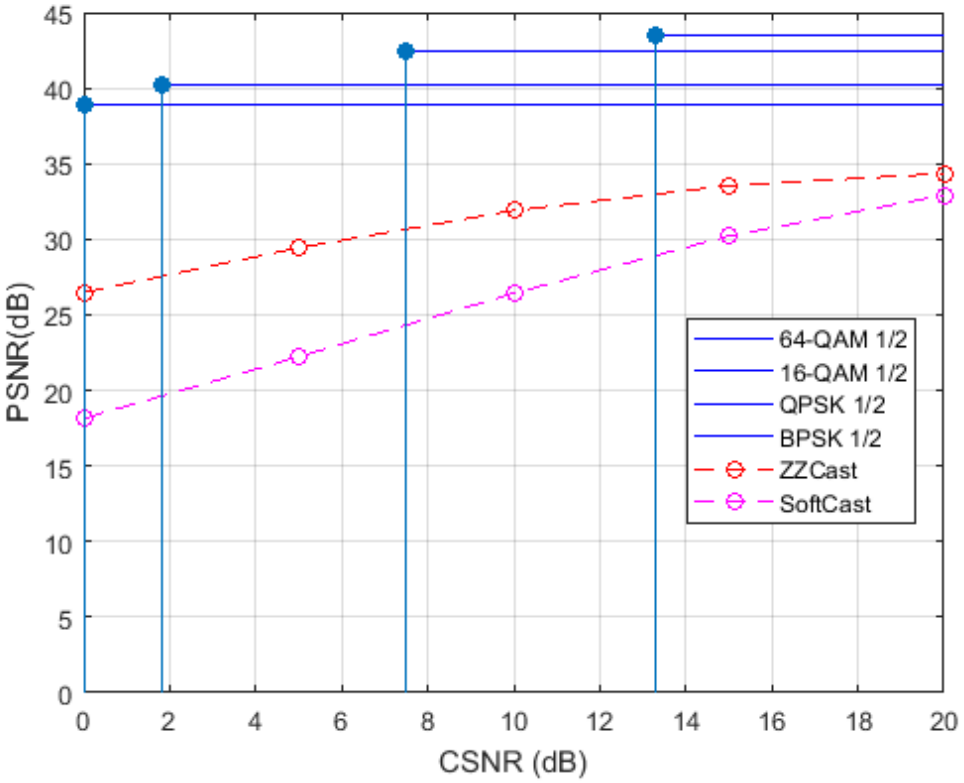


FIGURE 3.11: Akiyo - 100K

3.6 ZZCast Extensions

We change the source coding mechanisms for ZZCast. Wavelet transmission is another alternative for 2D-DCT and it outperforms it due to the better representation capacity. We extend this mechanism to better utilize bandwidth exploiting spherical spirals. We are able to modify AMP algorithm which sparsity level has priorly transmitted to the receiver.

3.6.1 Discrete Wavelet Transform(DWT)

We start this section by comparing 2D-DCT and multi-level discrete 2-D wavelet transform with the Cohen-Daubechies-Feauveau (CDF) 9/7 wavelet. We set the same number of non-zero values and plot results accordingly. As the result suggests, CDF 9/7 wavelet transform can better compress images than 2D-DCT. We compared DCT based ZZCast and DWT based ZZCast in figures.

Metric :	PSNR	SSIM
2D-DCT :	31.86	0.85
Wavelet Tranmission :	32.09	0.89

TABLE 3.3: Kodak-24 Dataset.

Metric :	PSNR	SSIM
2D-DCT :	35.81	0.91
Wavelet Tranmission :	36.68	0.94

TABLE 3.4: Lenna.



FIGURE 3.12: Effect of Transmission

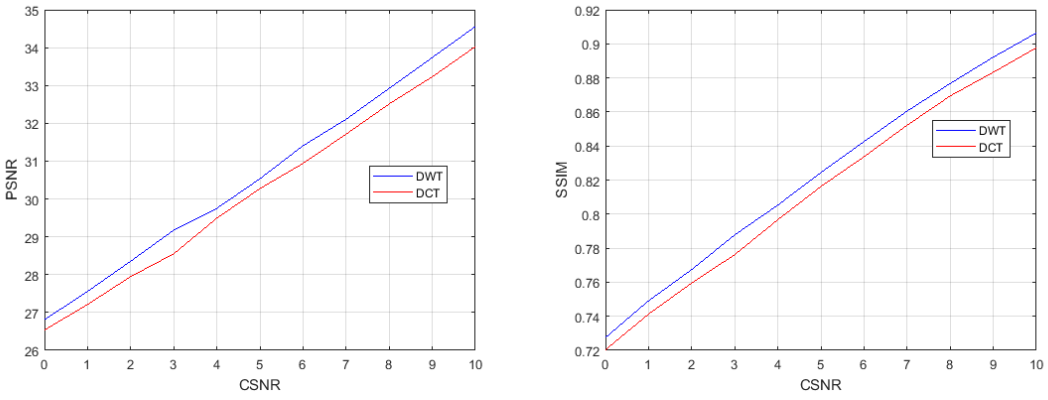


FIGURE 3.13: Comparison between DCT and DWT (50K Lenna ZZCast)

3.6.2 AMP Sparsity Level Information

AMP algorithm does not exploit the sparsity level information. We have already transmitted this information to the receiver and the estimation algorithm could use this side information to recover the signal better than the general AMP algorithm. In order to

exploit this idea, we propose a weighted-AMP algorithm. Firstly, we present the algorithm and compare it with AMP. Then, we use the weighted-AMP algorithm in a place for the AMP algorithm to recover sparse signals. Let us first introduce the problem and present the algorithm, and subsequently, provide the modified version.
The problem can be represented as follows:

Given measurements $\mathbf{y}$ and a sensing matrix $\mathbf{A}$, we want to recover the sparse signal $\mathbf{x}$ from the underdetermined system of linear equations:

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{z}$$

where:

- $\mathbf{y}$ is the M-dimensional vector of measurements.
- $\mathbf{A}$ is an $M \times N$ sensing matrix (usually a random matrix).
- $\mathbf{x}$ is the N-dimensional vector representing the sparse signal we want to recover.
- $\mathbf{z}$ is the M-dimensional noise vector.

The AMP algorithm can be represented by the following update equations:

Update the residual:
$\mathbf{v}_t = \mathbf{y} - \mathbf{A}\hat{\mathbf{x}}_t + b_t\mathbf{v}_{t-1}$

Update the next estimate of the sparse signal:
$\hat{\mathbf{x}}_{t+1} = \eta_{st}(\hat{\mathbf{x}}_t + \mathbf{A}^T\mathbf{v}_t; \lambda_t)$

Update the threshold parameter and sparsity estimate:
$b_t = \frac{1}{M}\|\hat{\mathbf{x}}_t\|_0$
$\lambda_t = \frac{\alpha}{\sqrt{M}}\|\hat{\mathbf{v}}_t\|_2$

number of non-zeros	Weighted	AMP
32≤x<64	0.0198	0.1145
16≤x<32	0.0202	0.1147

TABLE 3.5: MSE performance when perfect channel case

The Weighted-AMP algorithm can be modified :

Update the residual:

$\mathbf{v}_t = \mathbf{y} - \mathbf{A}\hat{\mathbf{x}}_t + b_t\mathbf{v}_{t-1}$

Update the next estimate of the sparse signal with auxiliary information $\boldsymbol{c_t}$

$\hat{\mathbf{x}}_{t+1} = \eta_{st}(\hat{\mathbf{x}}_t + \mathbf{A}^T\mathbf{v}_t; \lambda_t; \boldsymbol{c_t})$

Update the threshold parameter and sparsity estimate:

$b_t = \frac{1}{M}\|\hat{\mathbf{x}}_t\|_0$

$\lambda_t = \frac{\alpha}{\sqrt{M}}\|\hat{\mathbf{v}}_t\|_2$

Cost function sets 0 for non-zero values. We present a comparison between AMP and Weighted-AMP. With a total vector length of 256, when the number of non-zero elements is as in the table, and recovery is employed in a noise-free environment. Weighted-AMP outperforms AMP. Figure 3.14 shows the probability of exact recovery for measurement rate. Weighted-AMP outperforms AMP in this experiment since it converges rapidly to the exact recovery and its recovery performance is better even there is a fewer amount of measurements. Lastly, we employ both algorithms in ZZCast to just observe recovery performance. Although there is no huge amelioration between AMP and Weighted-AMP, we remind you that due to bandwidth constraints we limit the number of vectors that are recovered via AMP at the receiver.

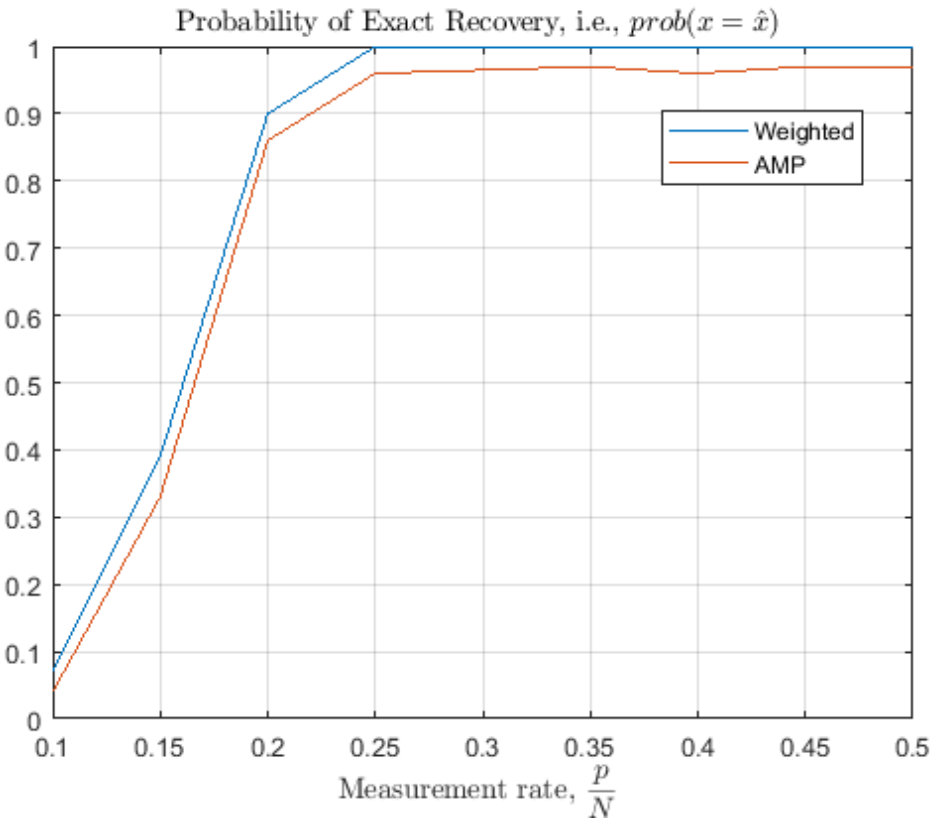


FIGURE 3.14: Exact recovery performance of AMP algorithms for synthetic data

3.7 DL-aided Uncoded Image/Video Tranmission

Deep Learning utilization in image transmission was already introduced [13]. However, data-driven approaches bring more calculation and training complexity. In order to relax drawbacks, model-driven approaches are widely recommended [73]. Like the model-driven approach, we employ encoding as conventional method i.e. ZZCast encoding but employ deep neural based alternatives as a decoder. For widely applicable scenario is for that the transmitter is a mobile device which hardware capability is limited. However, receiver may apply complex decoding algorithms because it is central based system such as disaster coordination center. According to decoder design, we are proposing two different schemes.

3.7.1 ZZCast with DNN-Decoders

Next, we propose a slightly modified version of ZZCast encoder here. After applying the transform operation, such as 2D-DCT, employing the zigzag operation outputs a vector. Without creating chunks from this vector as before, we simply just 'cut' its tale out according to the bandwidth constraint. Zigzag algorithm basically puts the frequency components in an increasing order, and we keep the low-frequency components which tend to have a greater impact on the images perceptual quality.

In [55], Heckel and Hand proposed the idea of deep decoder and they showed that compression of images can be achieved exploiting neural networks. Untrained neural network that generates natural images with very few parameters. The intuition is that thresholding values in wavelet compression can be employed via neural networks, and they demonstrate that neural networks perform better than wavelet thresholding. Authors show that ImageNet images can be recovered from compressed values via linear layers without training. By the same manner, deep decoder will be used to recover the image as an 'inverse-DCT' operation.

According to our experiments, if zigzag operation is employed at the transmitter, its inverse must be applied at the decoder. We have observed that the convolutional layers cannot learn the inverse zigzag operation and the training completely fails. In order to accelerate training we added inverse 2D-DCT step at the end [57], although neural networks are capable of finding inverse operation. Our baseline pipeline is given in Fig.3.15

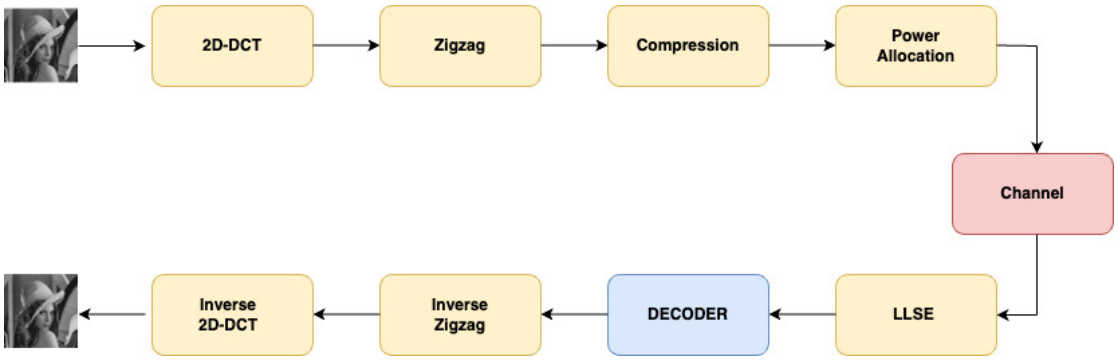


FIGURE 3.15: Baseline pipeline of DL-aided ZZCast decoder (No improvement)

3.7.2 ZZCast with Convolutional Decoder

CNNs are known to perform well at image processing tasks. Inherent inductive bias of CNNs leads to better algorithms for computer vision. Accordingly, we change the decoder algorithm in deep decoder to a CNN.

Secondly, although CNNs have better performance in image processing compared to linear layers, employing DNN-based decoder(blue box) in Fig. 3.15 after the LLSE does not improve the performance. According to this finding, we employ U-Net architecture at the end Fig. 3.16.

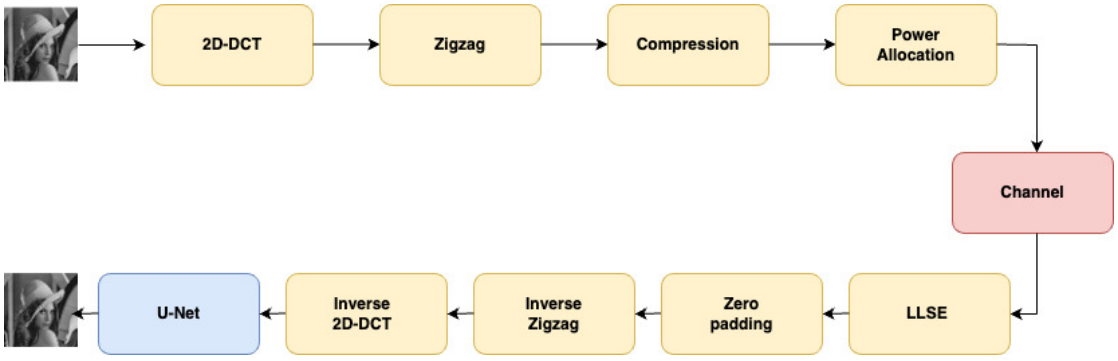


FIGURE 3.16: Pipeline of U-Net aided ZZCast decoder

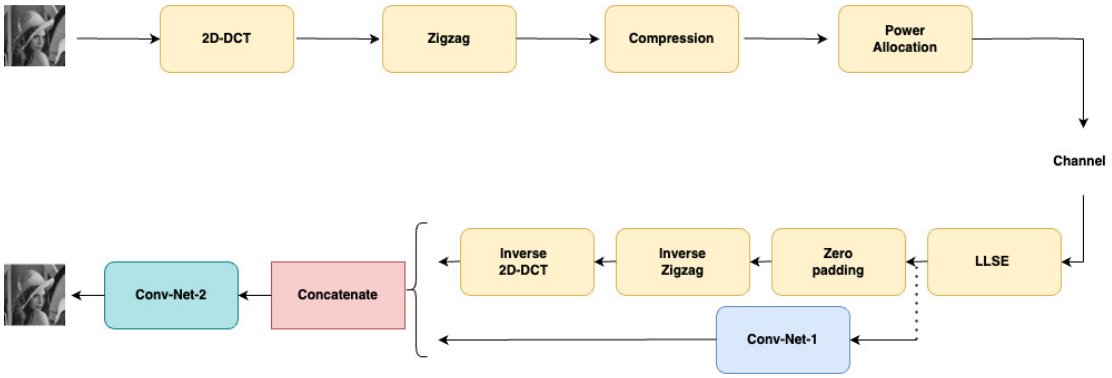


FIGURE 3.17: Pipeline of Conv aided ZZCast decoder

In our performance evaluation presented in Table 3.6, we acknowledge that we did not explicitly consider the influence of the channel on the end-to-end performance of compression and decompression capabilities. Our main focus was on assessing the performance in terms of the Mean Square Error (MSE) metric.

While the channel's impact on the overall system performance is essential, our study was primarily concerned with evaluating the accuracy of the compression and decompression processes themselves, rather than the effects of the transmission channel. The MSE metric allowed us to quantify the reconstruction quality and compare the performance of different algorithms in terms of signal fidelity.

Among the various configurations we examined, we found that incorporating a U-Net style denoiser at the decoder yielded the best performance. By leveraging this denoiser architecture, we were able to achieve superior compression and decompression capabilities, resulting in lower MSE values.

Furthermore, we introduced an adversarial loss, specifically the cross-entropy loss, to enhance the performance of both the Conv-aided and WGAN-GP aided decoders. This addition aligns with existing literature, which suggests that incorporating adversarial learning can lead to improvements in the overall performance. The combination of the MSE metric and the adversarial loss further enhances the quality of the reconstructed images.

While our evaluation did not consider the channel conditions, our focus was primarily on analyzing the performance of the compression and decompression processes themselves. By selecting the best-performing configuration and leveraging both the U-Net style denoiser and adversarial loss, we were able to achieve remarkable results in terms of image quality and fidelity. In Figure 3.18, we present the outcomes of our study conducted on the Additive White Gaussian Noise (AWGN) channel, where we introduce a novel approach by combining traditional uncoded image transmission techniques with Deep

Metric :	PSNR
U-Net :	25.11
Adversarial Loss :	24.53
Convolutional:	24.42
Baseline :	24.02

TABLE 3.6: Performance of methods without transmission

Neural Networks (DNNs). To the best of our knowledge, this is the first attempt to integrate these conventional methods with the capabilities of DNNs. By leveraging this integration, our aim is to enhance the efficiency and performance of image transmission in the presence of AWGN. We train our decoders using the Pytorch library [74] on the CIFAR-10 dataset [75] which consists of 60000 32×32 images. We split the dataset into 45000 images for training, 5000 images for validation, and 10000 images for testing. We use a batch size of 32 and train for 500 epochs.

We use the Adam optimizer [76] with learning rate 10^{-3}. The learning rate is scheduled according to the validation set, where the learning rate is decreased by a factor of 0.8 if the validation loss does not improve for four epochs consecutively. We use an average power $\bar{P} = 1$ and change the channel noise power σ^2 accordingly to satisfy CSNR.

Our findings make a significant contribution to the advancement of this field by exploring the synergy between uncoded transmission methods and DNNs. This unique fusion of techniques opens up new possibilities for improving image transmission systems, presenting opportunities for further research and development in the area of deep learning-enhanced communication systems.

The use of DNNs in image transmission is a promising avenue to mitigate noise effects, improve error resilience, and achieve higher fidelity reconstructions. We believe that the successful combination of traditional transmission methods with DNNs can lead to transformative advancements in image communication, making our study an important step towards harnessing the potential of deep learning for enhanced image transmission in challenging channel conditions.

3.8 Conclusion

In this section, we proposed the simple but effective uncoded wireless image transmission alternative, and showed its superiority over other alternatives, including separation-based digital transmission. We also extended the proposed algorithm to the case of bandwidth mismatch and modified the classical recovery algorithm accordingly. Then, we considered a deep learning based decoder that can be trained in a data-driven manner. Although

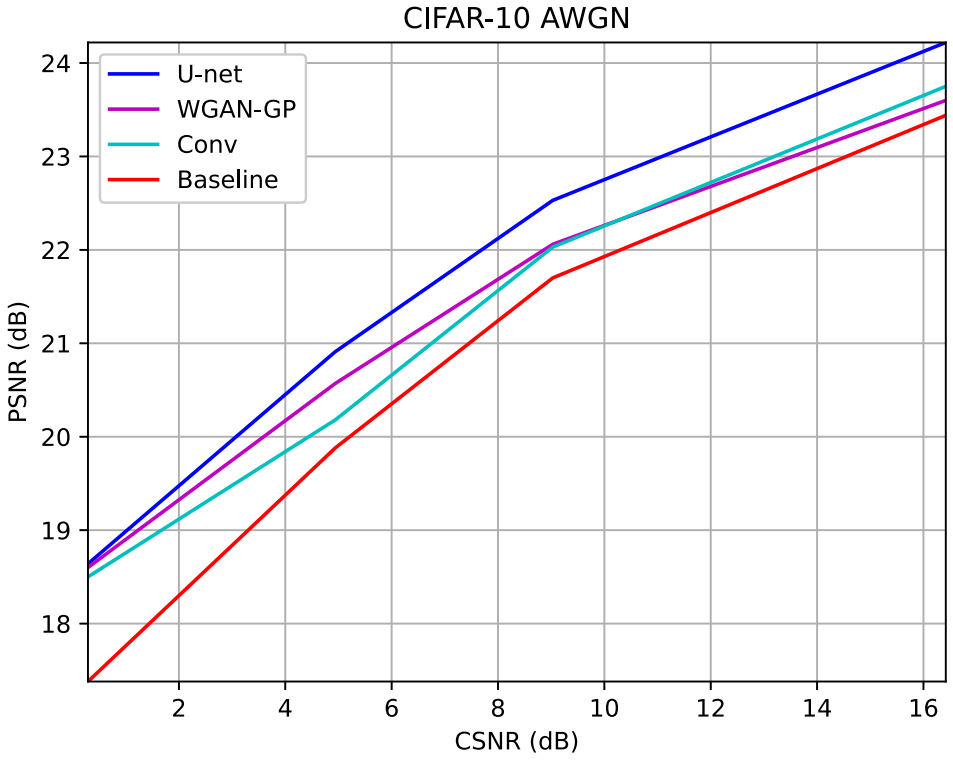


FIGURE 3.18: AWGN BW=128, simple-DCT, Conv, WGAN-GP and U-Net comparison.

the encoding algorithm is simple, decoder's performance compensates for it and achieves acceptable results. We demonstrated the power of the proposed algorithm for wireless image transmission through extensive numerical experiments by considering different contraints on the available memory, bandwidth and power.

Chapter 4

Model-Driven Deep Joint Source-Channel Coding over Time-Varying Channels

In this chapter, we study SNR-independent training of DeepJSCC. Machine learning techniques have recently offered powerful tools for designing joint source-channel coding (JSCC) schemes. Recent works have shown that deep learning (DL) based JSCC schemes (DeepJSCC) for wireless image transmission can outperform state-of-the-art separate source and channel coding designs. However, existing works on DL based JSCC schemes either require many models trained for specific channel conditions, increasing the storage requirements, or need to introduce additional parameters in the model to adapt to new channel conditions, increasing the complexity of the scheme. In this chapter, we propose DeepJSCC-MMSE, an end-to-end optimized JSCC scheme for wireless image transmission over wireless fading channels, in which a single DeepJSCC encoder/decoder pair is trained in concatenation with a channel equalization module at the receiver which can adapt to any channel condition without additional model parameters. We show that DeepJSCC-MMSE can outperform parameterized equalization techniques, without requiring the encoder and decoder to be retrained for new channel conditions, paving the way for generalized DL based JSCC schemes.

4.1 Introduction

In this chapter, we study conventional wireless transmission systems consist of two distinct blocks: a source encoder, which compresses the source information to reduce redundancy, followed by a channel encoder, which reintroduces redundancy to ensure reliable

transmission by mitigating the effects of channel noise. According to Shannon's separation theorem [1], there is no loss of optimality in this two-step process if the channel blocklength goes to infinity, and it is widely used in practical implementations. For example, in wireless image transmission, the image is first compressed by an image compression algorithm, such as JPEG2000 [77] or BPG [78] before a channel code, such as convolutional or LDPC codes, is used to transmit the compressed bits over the noisy channel. Although separation is known to be sub-optimal in the practical finite blocklength regime, it is used in almost all digital communication systems today partially due to the modularity of its design, but also due to the lack of alternatives that can offer significant performance gains with reasonable complexity.

Information theoretically speaking, joint source-channel coding (JSCC) can outperform separation in the finite blocklength regime [79]. In JSCC, the encoder directly maps the source to channel symbols by combining source and channel coding into a single step. Despite the complexity in designing effective JSCC schemes, recently, it was shown that deep neural networks (DNNs) can be used to parameterize JSCC schemes and optimized in a data driven manner [13]. However, a key limitation of the results presented in [13, 24, 80] was that a separate DNN needs to be trained for each channel condition, which leads to large storage requirements, limiting the practicality of these codes. This limitation was tackled in [28, 81, 82], where a single network is trained to adapt to new channel conditions by accepting the channel state information (CSI) as an input to both the encoder and decoder, through the *attention feature (AF) module*. In digital communications, depending on the CSI, the encoder and decoder may employ different source and channel codes. For example, as the channel signal-to-noise ratio (CSNR) decreases, the encoder would choose a lower channel code rate and modulation order in order to improve the reliability of the transmission. This, in turn, would entail a lower source code rate to achieve the same delay. The AF module is meant to provide a similar functionality, albeit in a parameterized way. The module allows DeepJSCC to adjust the dimensionality reduction applied to the source image and the redundancy it introduces for error correction. The authors in [28] showed that the AF module allows a single model to perform at least as well as multiple models trained across different channel conditions. In [82], it is shown that this approach can also be extended to OFDM channels, where the encoder/decoder pair adapts to the variations in multiple subchannels. However, while this solves the storage problem associated with DeepJSCC, the AF module introduces additional complexity to the architecture and increases the training requirements as it increases the number of parameters.

In this section, we propose a simple alternative to the AF module, which does not employ any additional parameters to the DeepJSCC scheme proposed in [13], while enabling the model to adapt to a wide range of channel conditions. This is achieved

by introducing a simple equalizer, based on the minimum mean-squared error (MMSE) estimator. We treat the transmitted signal as a sequence of independent and identically distributed (i.i.d) Gaussian symbols, and apply MMSE estimation at the receiver. When the channel noise is Gaussian, MMSE estimation is a simple scaling of the received signal, which depends on the CSNR. The contributions of this paper are as follows:

- We propose DeepJSCC-MMSE, a DL based JSCC scheme that operates optimally at each channel condition without introducing additional parameters.
- We consider CSI only at the receiver, which compute the MMSE estimate of the transmitted symbols for equalization, and show that it is sufficient to learn an encoder/decoder pair that generalizes across a wide range of CSNRs.
- We show the efficacy of DeepJSCC-MMSE also over multiple carriers, i.e., orthogonal frequency division multiplexing (OFDM).
- Our results show that DeepJSCC-MMSE achieves similar or better performance under AWGN and Rayleigh fading than an equivalent architecture employing the AF module for CSI adaptation, with lower complexity.
- DeepJSCC-MMSE also shows better immunity against channel estimation errors than an equivalent architecture employing the AF module.

4.2 Problem Definition

We consider the problem of wireless image transmission over slow fading channels (i.e., Rayleigh) additive white Gaussian noise (AWGN) channels. We are following the same notation as in [13]. The source $\mathbf{x} \in \mathbb{R}^n$ is mapped to a set of continuous valued channel input symbols $\{\mathbf{z}_i\}_{i=1}^M, \mathbf{z}_i \in \mathbb{C}^{k/M}$ via an encoder $f : \mathbb{R}^n \mapsto \mathbb{C}^k$, which satisfies the average power constraint $\bar{P}$, such that

$$\frac{1}{k}\sum_{i=1}^{M} E[||\mathbf{z}_i||_2^2] \leq \bar{P}. \tag{4.1}$$

We will define the *bandwidth ratio* $R = k/n$. Each channel input symbol vector $\mathbf{z}_i$ is then transmitted using an orthogonal frequency division multiplexing (OFDM) physical layer with M sub-channels. The channel transfer function of each sub-channel is defined as

$$\Gamma_i(\mathbf{z}_i) = h_i\mathbf{z}_i + \mathbf{n}_i, \tag{4.2}$$

where $\mathbf{z}_i \in \mathbb{C}^{k/M}$ are the symbols in the ith sub-channel, h_i is the sub-channel gain, and $\mathbf{n}_i \sim CN(0, \sigma^2\mathbf{I})$ is an independent and identically distributed (i.i.d.) complex Gaussian noise vector.

We assume that the channel gain h_i and the noise power σ^2 can be estimated at the receiver as $\hat{h}_i$ and $\hat{\sigma}^2$, respectively. This allows the receiver to perform channel equalization as

$$\hat{\mathbf{y}}_i = \frac{\hat{h}_i^*}{|\hat{h}_i|^2}\mathbf{y}_i, \tag{4.3}$$

where $\mathbf{y}_i = \Gamma_i(\mathbf{z}_i)$ and $\hat{h}_i^*$ is the complex conjugate of $\hat{h}_i$. Consequently, the average channel signal-to-noise ratio (CSNR) over all sub-channels, is defined as

$$\text{CSNR} = 10\log_{10}\left(\frac{1}{M}\sum_{i=1}^{M}\frac{|h_i|^2\bar{P}}{\sigma^2}\right) dB. \tag{4.4}$$

From this channel model, we will consider two special cases:

1. $h_i = 1,\ \forall i$, which corresponds to the additive white Gaussian noise (AWGN) channel,
2. $h_i \sim CN(0, H_c)$ and remains constant for the duration of a single image transmission, which corresponds to a slow fading channel.

The equalized symbols are then passed to a decoder $g : \mathbb{C}^k \mapsto \mathbb{R}^n$ to obtain a reconstruction $\hat{\mathbf{x}} = g(\{\mathbf{y}_i\}_{i=1}^M)$.

We measure the quality of the reconstruction using the peak signal-to-noise ratio (PSNR) metric, defined as

$$\text{PSNR} = 10\log_{10}\left(\frac{A^2}{\text{MSE}(\mathbf{x},\hat{\mathbf{x}})}\right) dB, \tag{4.5}$$

where $\text{MSE}(\mathbf{x},\hat{\mathbf{x}}) = ||\mathbf{x}-\hat{\mathbf{x}}||_2^2$ and A is the maximum possible value of the image pixels (255 in our case). The end-to-end optimization objective can therefore be defined as

$$(f^\star, g^\star) = \underset{f,g}{\text{argmin}}\, \mathbb{E}_{\mathbf{x}}[\text{MSE}(\mathbf{x},\hat{\mathbf{x}})]. \tag{4.6}$$

4.3 Proposed Solution

To solve the optimization problem in Eqn. (4.6), we parameterize the encoder and decoder functions by $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$, respectively, using DNNs. The architectures for $f_{\boldsymbol{\theta}}$ and $g_{\boldsymbol{\phi}}$ are shown in Fig. 4.1. The architecture is fully convolutional, where C refers to the number of channels in the output tensor of the convolution operation. C_{out} refers to the number of channels in the final output tensor of the encoder $f_{\boldsymbol{\theta}}$, which decides the number of channel uses k per image. It controls the compression. We use various proposed modules from other works that have demonstrated strong efficacy when used

for image processing problems. Specifically, we use the 'Pixel shuffle' module [83], which reshapes the input tensor by increasing the height and width of the input such that the channel dimensions are reduced while the height and width dimensions are increased. We also use the *generalised divisive normalization (GDN)* layer, initially proposed in [84], which is effective in density modeling and compression of images. Finally, we used the simplified attention module proposed in [85], which reduces the computation cost of the attention module originally proposed in [86]. Let us introduce the basic blocks in here. A 3x3 Convolution Block is a common building block in convolutional neural networks (CNNs). It consists of a 3x3 convolutional layer followed by an activation function. The convolutional layer applies a set of learnable filters to the input data, producing feature maps that capture different patterns in the data. The activation function introduces non-linearity to the output of the convolutional layer, allowing the network to model complex relationships in the data.

$$\text{output} = \text{Activation}(\text{Conv2D}(\text{input}, \text{kernel_size} = 3, \text{stride} = 1, \text{padding} = 1)) \tag{4.7}$$

The Leaky ReLU activation function is a variation of the popular Rectified Linear Unit (ReLU) function. It introduces a small slope for negative input values, preventing the "dying ReLU" problem and allowing gradients to flow during backpropagation.

$$\text{Leaky ReLU}(x) = \begin{cases} x & \text{if } x \geq 0 \\ \alpha \cdot x & \text{if } x < 0 \end{cases} \tag{4.8}$$

GDN is a normalization technique used in some neural networks to normalize the input data in a divisive manner. It aims to normalize the input signal to have zero-mean and unit-variance by applying a set of learnable gain and bias parameters.

$$\text{GDN Block} \quad \text{output} = \frac{\text{input}}{\sqrt{\text{Local Contrast} + \epsilon}} \times \text{Gain} + \text{Bias} \tag{4.9}$$

Pixel Shuffle is a technique used in some neural networks, especially in super-resolution tasks. It reshapes the input tensor by rearranging its elements to increase the spatial resolution.

$$\text{Pixel Shuffle Block} \quad \text{output} = \text{PixelShuffle}(\text{input}, \text{upscale_factor}) \tag{4.10}$$

To ensure that the power constraint is met, we normalize the encoder output $\tilde{\mathbf{z}} = f_{\boldsymbol{\theta}}(\mathbf{x})$ as

$$\mathbf{z} = \sqrt{k\bar{P}}\frac{\tilde{z}}{\sqrt{\tilde{z}^*\tilde{z}}}. \tag{4.11}$$

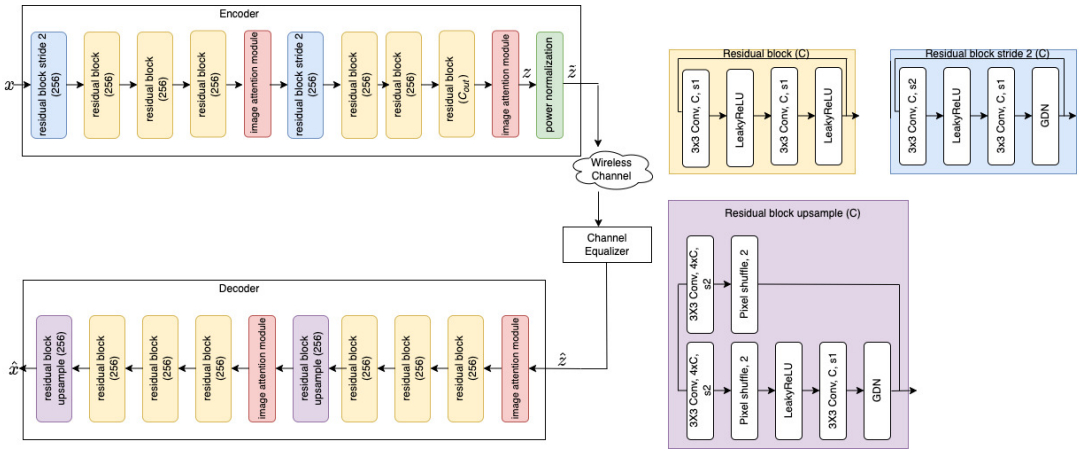


FIGURE 4.1: Architecture of the proposed method

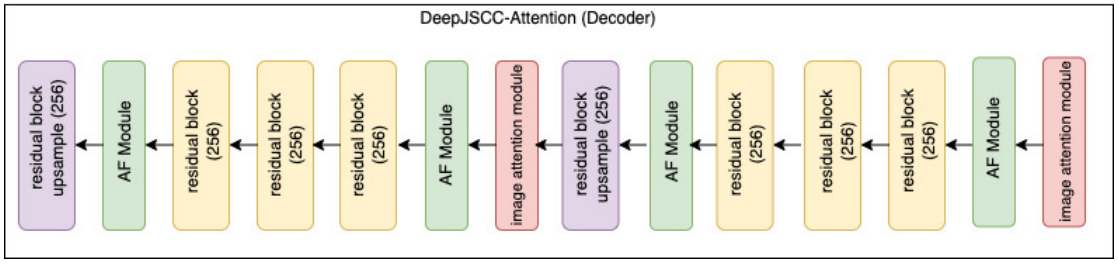


FIGURE 4.2: DeepJSCC-Attention Decoder Architecture

The vector $\mathbf{z}$ are then divided into equal chunks of size k/M to form the sub-channel input symbol set $\{\mathbf{z}_i\}_{i=1}^{M}$. As such, the optimization problem can be redefined as

$$(\boldsymbol{\theta}^{\star}, \boldsymbol{\phi}^{\star}) = \underset{\boldsymbol{\theta},\boldsymbol{\phi}}{\operatorname{argmin}} \mathbb{E}_{\mathbf{x}}[\mathrm{MSE}(\mathbf{x}, \hat{\mathbf{x}})]. \tag{4.12}$$

This allows us to search for the optimal encoding and decoding functions using gradient descent.

In order to adapt the decoder to different CSNRs, we utilize the minimum mean-squared error (MMSE) estimator of $\mathbf{z}_i$ using the equalized received symbols $\hat{\mathbf{y}}_i$. The MMSE estimator of $\mathbf{z}_i$ is defined as $\hat{\mathbf{z}}_i = E[\mathbf{z}_i|\hat{\mathbf{y}}_i]$. It has a closed form solution if both $\mathbf{z}_i$ and $\hat{\mathbf{y}}_i$ are jointly Gaussian

$$\hat{\mathbf{z}}_i = \mathbf{K}_{\mathbf{z}_i\hat{\mathbf{y}}_i}\mathbf{K}_{\hat{\mathbf{y}}_i}^{-1}(\hat{\mathbf{y}}_i - E[\hat{\mathbf{y}}_i]) + E[\mathbf{z}], \tag{4.13}$$

where $\mathbf{K}_{\mathbf{z}_i\hat{\mathbf{y}}_i}$ and $\mathbf{K}_{\hat{\mathbf{y}}_i}$ are the covariance matrices of $(\mathbf{z}_i, \hat{\mathbf{y}}_i)$ and $\hat{\mathbf{y}}_i$, respectively. If we assume that the values in $\mathbf{z}_i$ are i.i.d. zero mean and Gaussian, then the estimator can be simplified to

$$\hat{\mathbf{z}}_i = \frac{|\hat{h}_i|^2\bar{P}}{|\hat{h}_i|^2\bar{P} + \hat{\sigma}^2}\hat{\mathbf{y}}_i. \tag{4.14}$$

This is exactly the assumption we make and we subtract the empirical means of $\mathbf{z}_i$ before transmission to remove the overhead of having to transmit the mean to the receiver. The MMSE estimates $\{\hat{\mathbf{z}}_i\}_{i=1}^{M}$ are then concatenated to form a single vector $\hat{\mathbf{z}} \in \mathbb{C}^k$ before being fed to the decoder to obtain the reconstruction $\hat{\mathbf{x}} = g_{\boldsymbol{\phi}}(\hat{\mathbf{z}})$.

4.4 Performance Evaluation

4.4.1 Experimental Setup

We compare DeepJSCC-MMSE with DeepJSCC-Attention, which uses the AF module proposed in [28]. To ensure fairness, we use the same encoder architecture $f_{\boldsymbol{\theta}}$ for both schemes, while the decoder architecture for DeepJSCC-Attention uses AF modules rather than the MMSE method we propose. The decoder architecture for the DeepJSCC-Attention scheme can be found in Fig. 4.2. The AF module can be described by the following function:

$$\hat{\mathbf{V}} = \mathbf{V} \otimes \hat{\mathbf{c}}, \tag{4.15}$$

where

$$\mathbf{V} = \left[\mathbf{V}_1 \dots \mathbf{V}_c\right], \tag{4.16}$$

and $\mathbf{V}_i \in \mathbb{R}^{h \times w}$. The matrix $\mathbf{V} \in \mathbb{R}^{c \times h \times w}$ is the output of a convolutional layer and reshaped into a row matrix where each element is a channel slice from the output tensor. It is multiplied by a weight matrix $\hat{\mathbf{c}} \in \mathbb{R}^c$ via the Kronecker product $\otimes$, which is obtained via the AF module as follows:

$$\hat{\mathbf{c}} = \text{Sigmoid}(\text{Linear}(\text{LeakyReLU}(\text{Linear}(\mathbf{c}, \text{CSI})))), \tag{4.17}$$

where $\mathbf{c} \in \mathbb{R}^c$ is a context vector formed by averaging each sub-matrix in $\mathbf{V}$, and CSI are the estimated channel state coefficients $(\hat{h}_1, ..., \hat{h}_M, \sigma^2)$. The input to the first linear layer is the concatenation of the context vector $\mathbf{c}$ and the coefficients $(\hat{h}_1, ..., \hat{h}_M, \sigma^2)$.

We train our models using the Pytorch library [74] on the CIFAR-10 dataset [75] which consists of 60000 32×32 images. We split the dataset into 45000 images for training, 5000 images for validation, and 10000 images for testing. We use a batch size of 128 and train for 500 epochs. We train and test the models with CSNR range $[0, 15]$ dB. We use the Adam optimizer [76] with learning rate 10^{-4}. The learning rate is scheduled according to the validation set, where the learning rate is decreased by a factor of 0.8 if the validation loss does not improve for four epochs consecutively. We use an average power $\bar{P} = 1$ and change the channel noise power σ^2 accordingly to satisfy CSNR. During

TABLE 4.1: Number of parameters for the compared methods

DeepJSCC-MMSE	22.588×10^6
DeepJSCC-Attention	22.614×10^6

training, we sample a CSNR value uniformly from the set $[0-15]$dB per batch of training data in order to expose the networks to a wide range of channel conditions.

The number of parameters for each architecture are presented in Table 4.1. DeepJSCC-MMSE uses less parameters than Attention-Decoder. It brings into saving memory requirements and fast training.

4.4.2 Baseline

As baseline, we consider the approach employed in [28, 81, 82], where the channel gain information available at the receiver, $\mathbf{h}$, and the noise variance σ^2 are fed into the decoder network, so that the decoding algorithm can learn to adapt to the channel condition, we call this baseline architecture DeepJSCC-Attention. This is achieved by concatenating this information to each layer output and employing a fully connected layer to reduce the dimension back to the its original size. This way, the CSI is used at each layer of the decoder network to extract as much information from the received signal as possible.

4.4.3 Numerical results

We begin by considering a special case of our problem where there is a single carrier ($M = 1$). We will assume here that the receiver is able to accurately estimate the CSI coefficients ($\hat{h}_i = h_i$, $\hat{\sigma}^2 = \sigma^2$). Fig. 4.3 shows the result under the AWGN channel ($h_i = 1$). We see that our proposed DeepJSCC-MMSE scheme performs better than the DeepJSCC-Attention scheme, although the improvement is somewhat limited. However, when we consider a Rayleigh fading channel ($H_c = 1$), the improvement is much more substantial, as seen in Fig. 4.4. We emphasize that DeepJSCC-MMSE achieves this gain is obtained with fewer parameters than DeepJSCC-Attention. The parameter count of both schemes is shown in Table 4.1. This shows that the contribution of the AF module is actually limited and our more simplistic approach can achieve similar or better results.

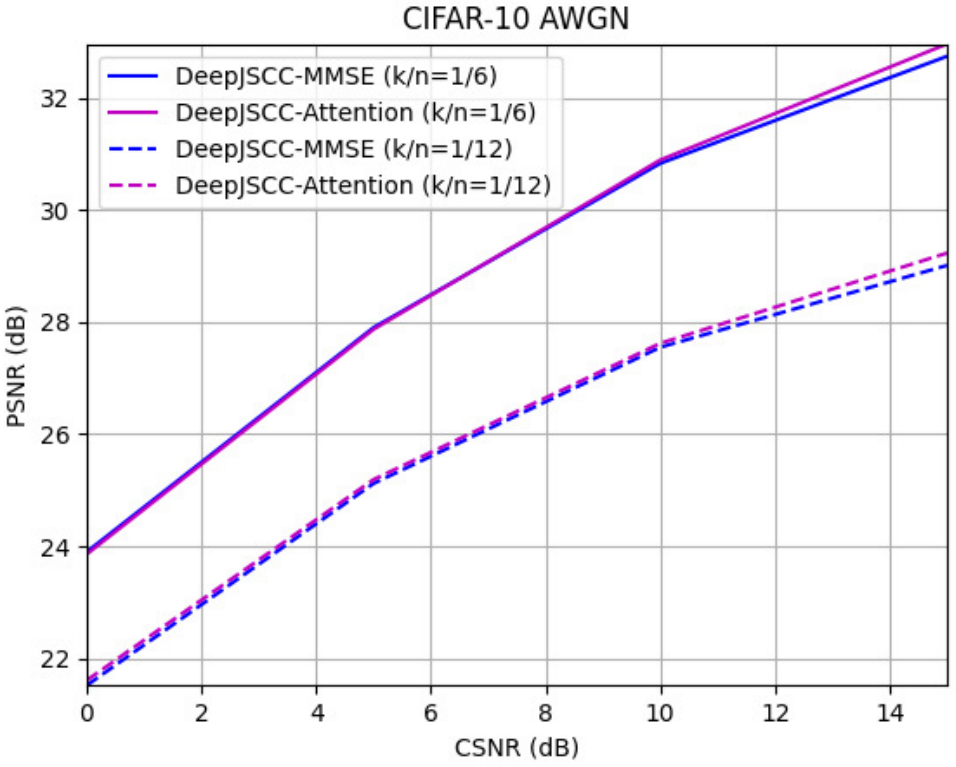


FIGURE 4.3: Performance of DeepJSCC-MMSE and DeepJSCC-Attention for R = 1/6 and R = 1/12 AWGN channel. Uniform distribution of SNR from 0dB to 15dB.

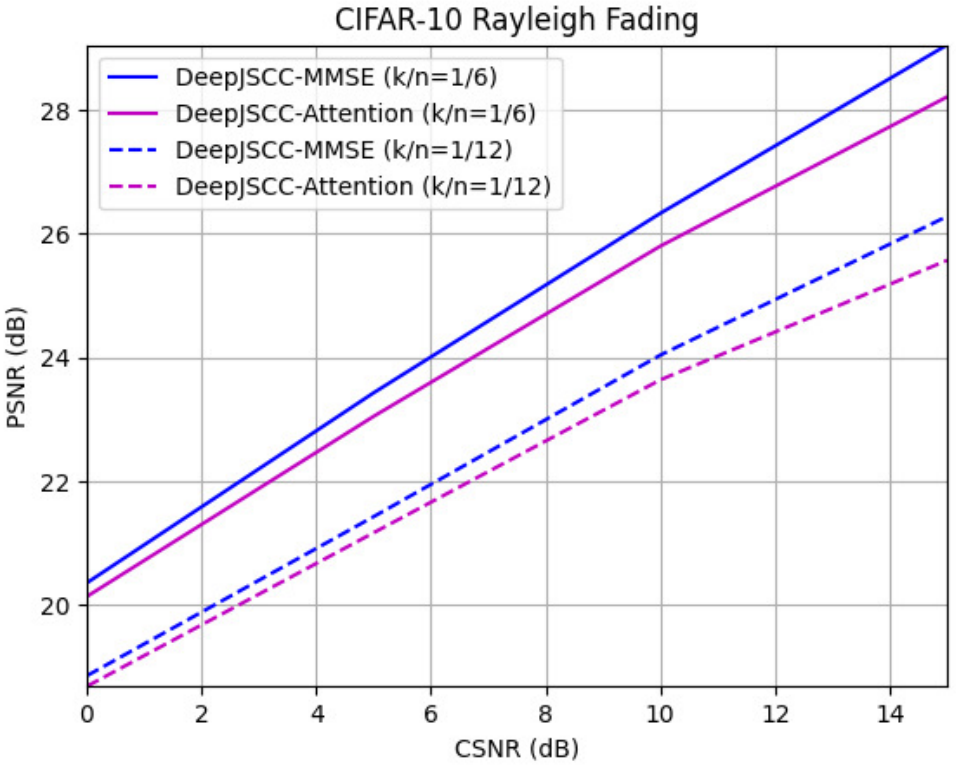


FIGURE 4.4: Performance of DeepJSCC-MMSE and DeepJSCC-Attention for R = 1/6 and R = 1/12 Rayleigh channel. Uniform distribution of SNR from 0dB to 15dB.

Next we consider the OFDM case with different number of sub-channels $M \in \{2, 4, 8\}$ and accurate CSI estimation, as in the single carrier case. Here, each sub-channel gain $h_i \sim CN(0, 1)$ is sampled i.i.d and the results are shown in Fig. 4.5. We see that the performance gain of DeepJSCC-MMSE over the DeepJSCC-Attention is even greater than what we observe in the single carrier case. Moreover, whereas DeepJSCC-MMSE improves with increasing number of sub-channels, DeepJSCC-Attention does not. This shows that our approach scales very well to multiple parallel channels, which is important for practical systems that use OFDM waveforms.

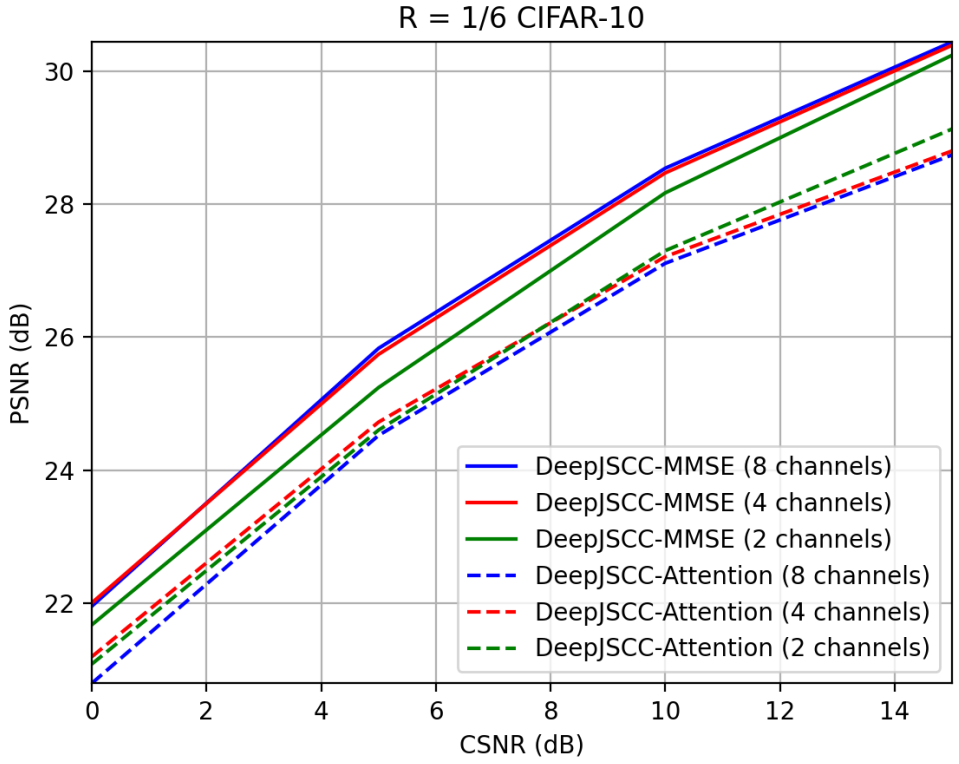


FIGURE 4.5: Performance of DeepJSCC-MMSE and DeepJSCC-Attention for R = 1/6 in multiple channel uses.

Finally, we study the robustness of each scheme under CSI estimation error for $M = 1$ in the Rayleigh fading scenario. That is, we try to analyze how sensitive the networks are to the mismatch between the CSI coefficients used to adapt the decoder and the true CSI coefficients of the channel. To do so, we assume that there is noise in the estimation of the noise power σ^2 in the form of an AWGN noise, such that $\hat{\sigma}^2 = \sigma^2 + n_e$, where $n_e \sim N(0, \sigma_e^2)$. We continue to assume that the channel gain is accurately estimated ($\hat{h}_i = h_i$). Fig. 4.6 shows the result of testing the model with noisy CSI estimation with different estimation noise power values. We see a similar result to the Rayleigh fading case as in Fig. 4.4, demonstrating the superiority of DeepJSCC-MMSE in terms

of its robustness against CSI estimation errors. We also test the robustness of both schemes by fixing the estimated channel noise power to specific values, while testing over a range of CSNRs. The results are shown in Fig. 4.7. We see that although both models gracefully degrade as the CSNR falls below the estimated CSNR value, and improves when the CSNR is above the estimated CSNR, DeepJSCC-MMSE performs better than DeepJSCC-Attention in almost all cases, except for when the estimated CSNR is 13 dB and the true CSNR is less than 8 dB. This is perhaps due to the overly optimistic estimation of MMSE, and shows a potential weakness in this scheme.

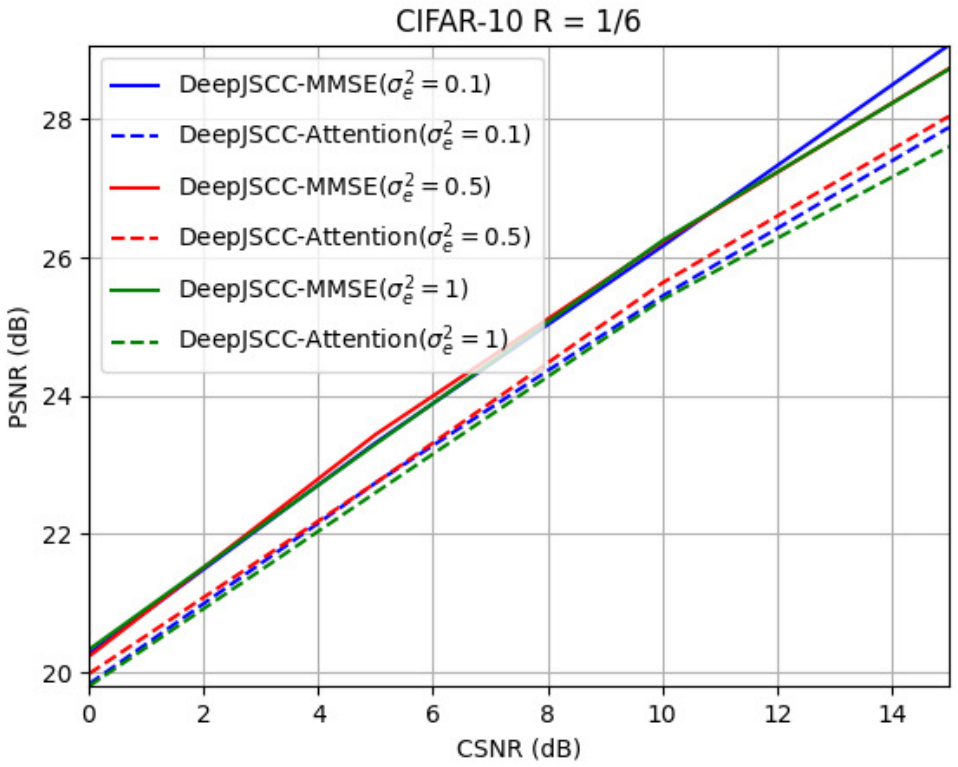


FIGURE 4.6: Channel estimation mismatch performace of DeepJSCC-MMSE and DeepJSCC-Attention in case noise in the noise power

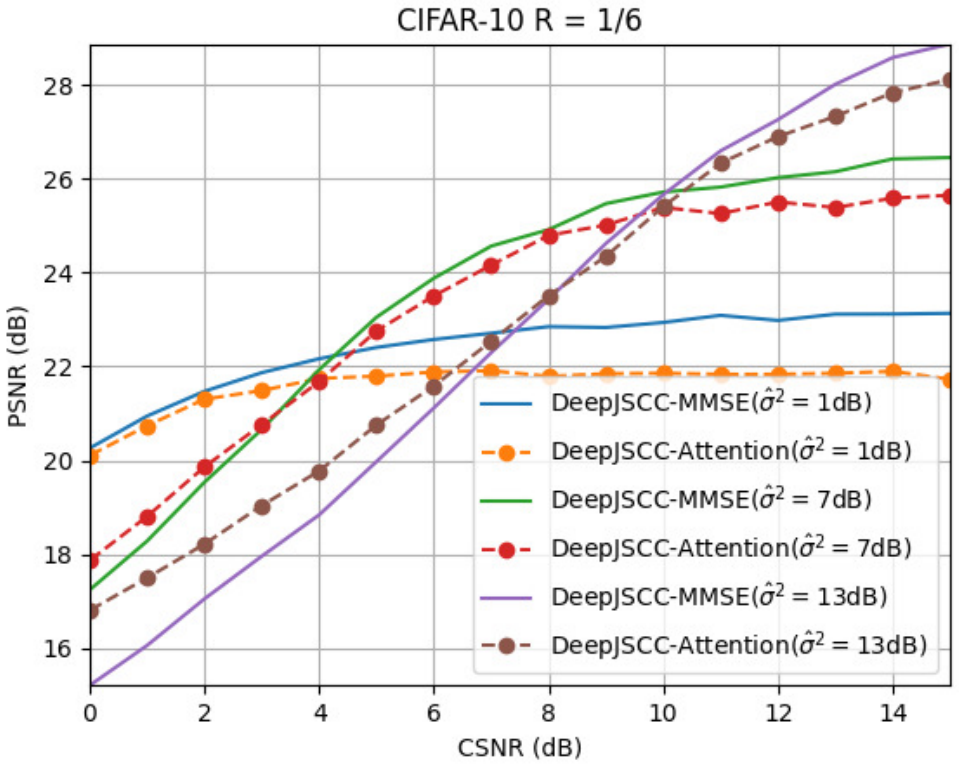


FIGURE 4.7: Fixed channel estimation performance of DeepJSCC-MMSE and DeepJSCC-Attention

4.5 Extensions

In this section, we present three proposed extensions to DeepJSCC-MMSE, which are based on our prior experiences and aimed at improving the results. Specifically, we investigate two questions: firstly, whether transmitting information about the class of the image can lead to an improvement in image reconstruction, and secondly, whether there is any additional prior information that can assist in the reconstruction process even if it is not transmitted to the receiver. These ideas stem from the concept of transmitting metadata along with the image data Chapter 3.

4.5.1 Metadata-Aided Image Recovery

In this section, we present an innovative idea inspired by the architecture of conditional generative adversarial networks (CGAN) [87]. Our experimental setup involves exploring whether transmitting label information to the receiver without any errors can improve the decoding algorithm's performance. The results of our experiments affirm that transmitting label information as a watermark improves the overall performance. In the original

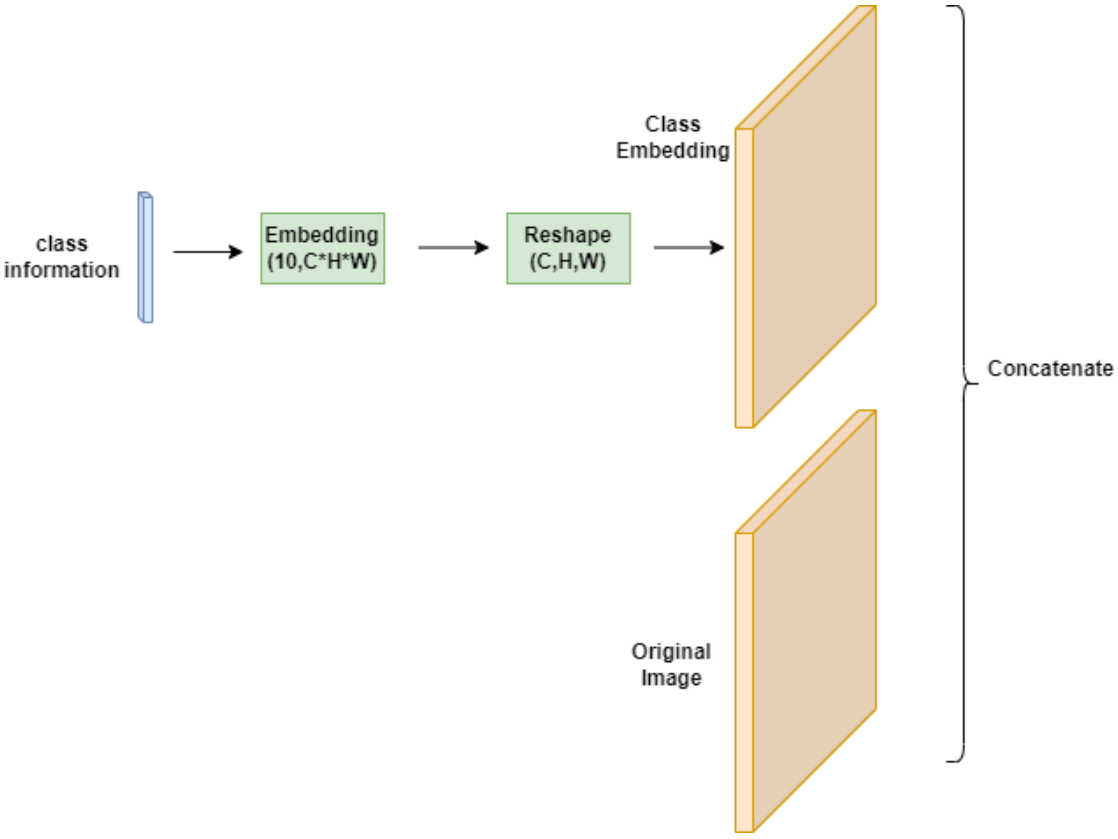


FIGURE 4.8: Proposed Model

DeepJSCC-MMSE encoder, the latent dimension primarily focused on the class label and lost some high-frequency image components. However, by transmitting the label information as a watermark, the latent dimension can instead focus on image details, leading to a slight improvement in performance. Specifically, the transmitter encodes the image using the DeepJSCC-MMSE encoder with concatenates the label embedding at the receiver. We apply a multilayer perceptron (MLP) to obtain the original latent dimension and employ the DeepJSCC-MMSE decoder. We are presenting our results in Fig. 4.9 and Fig. 4.10; these results underscore the effectiveness of our methodology, leading to more accurate outcomes. Through extensive experimentation and analysis, our study provides compelling evidence of the substantial improvement achieved by our proposed approach in the realm of MMSE estimation. We are providing the diagram to illustrate it in Fig. 4.8. Just to note, transmitting the class information through a reliable channel requires consideration. Since there are 10 classes, 4 bits should be sufficient for source coding. However, if you opt for BPSK 1/2 LDPC encoding, you will need 8 bits. Nevertheless, this is just a small burden on the end-to-end system, as the image is transmitted via 512 symbols, and the metadata needs only 4 symbols.

Algorithm 2 Enhanced Image Encoding and Reconstruction with Class Information

1: Initialize class information embedding matrix using nn.Embedding$(10, 1024)$
2: Reshape class embedding $(1, 32, 32)$
3: Concatenate class embedding with original image $(4, 32, 32)$
4: Pass concatenated input through Encoder neural network
5: Transmit encoded features through transmission channel
6: Perform MMSE recovery to estimate encoded features : estimated_data $(16, 8, 8)$
7: Embed class information for decoder using nn.Embedding$(10, 1024)$
8: Reshape class embedding $(16, 8, 8)$
9: Concatenate estimated data with class embedding $(32, 8, 8)$
10: Pass concatenated decoder input through Decoder neural network
11: Reconstruct the image
12: Calculate the metric

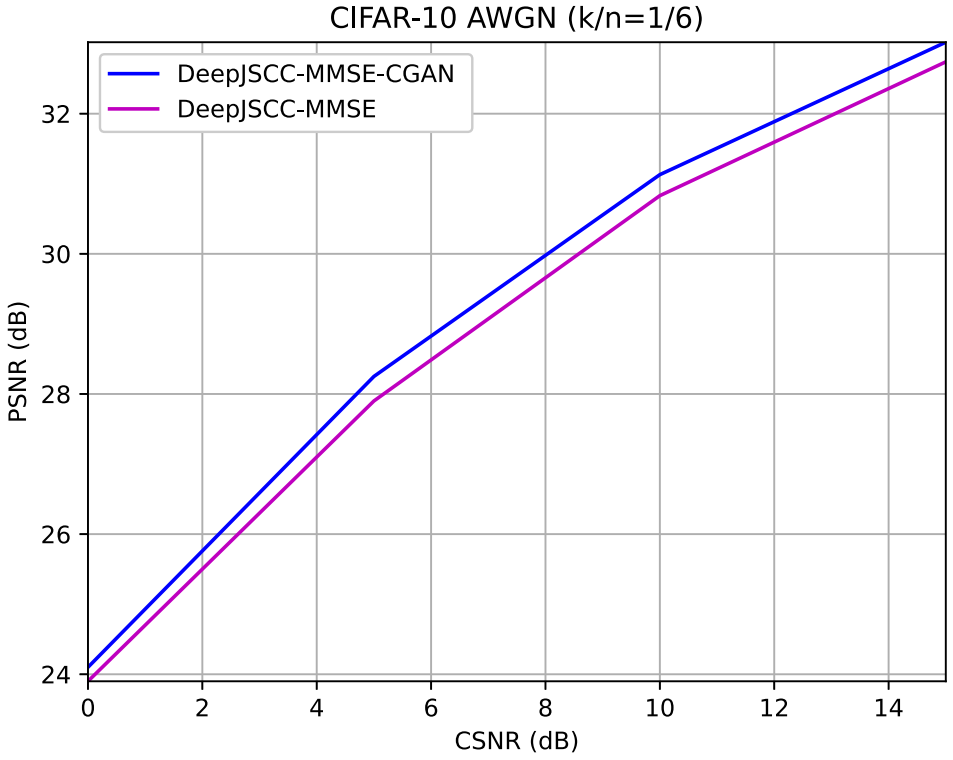


FIGURE 4.9: CGAN-inspired DeepJSCC-MMSE and DeepJSCC-MMSE tcn=16

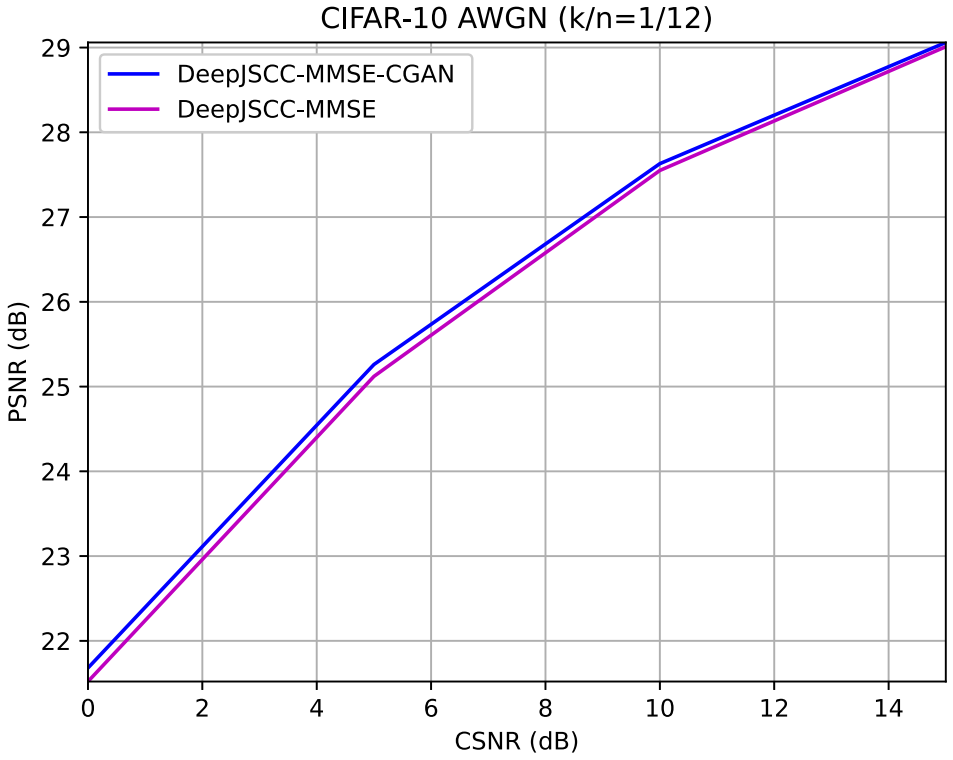


FIGURE 4.10: CGAN-inspired DeepJSCC-MMSE and DeepJSCC-MMSE tcn=8

4.5.2 Unsupervised Image Enhancement Recovery

The second improvement in our study draws inspiration from the subsequent work of CGAN, specifically InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets [88]. This work demonstrated that Generative Adversarial Networks (GANs) have the capability to learn not only the visual features but also the semantic features of the training dataset. Building upon these insights, our proposal involves replacing the original data distribution of each class with auxiliary information, which serves as conditions in the Conditional GAN (CGAN) setup.

By incorporating this auxiliary information, we anticipate an improvement in the overall performance of the model. Notably, this approach eliminates the requirement for explicit label information while introducing an additional classifier, referred to as $\mathbf{Q}$, which evaluates the quality of the learned auxiliary features. $\mathbf{Q}$ approximates the mutual information between the auxiliary vector and the generated sample, providing a measure of how effectively the auxiliary features are learned.

Through this novel approach, we aim to enhance the interpretability and representation learning capabilities of our model, allowing for a deeper understanding of the latent features and their influence on the generated samples. By leveraging the insights from

InfoGAN and incorporating auxiliary information, our proposal holds the potential to achieve superior performance and provide valuable insights into the learned representations within the GAN framework. In our study, we tackle a straightforward task using unsupervised learning to extract semantic features from the training dataset. To accomplish this, we employ three distinct loss functions tailored to different aspects of fidelity.

Firstly, the cross-entropy loss is utilized to measure class fidelity, ensuring that the generated samples accurately represent their respective classes. This loss encourages the model to capture the essential characteristics and class-specific features of the training data.

Secondly, the quality fidelity is addressed through an ordinary wireless image transmission model, maximizing the PSNR. This loss function pushes the decoder to produce high-quality samples that closely resemble the real data distribution, thus enhancing the overall fidelity of the recovered samples.

Lastly, the style fidelity is considered using a mean squared error (MSE) loss. We assume that the style vectors conform to a normal distribution with a mean of 0 and a standard deviation of 1. By incorporating the MSE loss, we enforce the generated samples to align with the desired random distribution, specifically an expectation maximization scheme. The logarithm of the probability density function (pdf) of the normal distribution is deduced to be proportional to x^2, which makes mean squared error a suitable choice for this purpose.

It's noteworthy that within the framework of this proposed method, the transmission of metadata for image reconstruction is entirely omitted, setting it apart from the initial approach. Instead, the power of the neural network comes to the forefront, adeptly absorbing information through an unsupervised learning approach.

The network's learning trajectory is guided by a meticulously designed loss function, a linchpin in propelling the model towards peak performance. This function not only fine-tunes the model but also enriches its ability to comprehensively grasp and extrapolate from the nuances inherent in the training data.

By incorporating these three loss functions, we achieve a comprehensive and robust approach that improves the overall performance beyond the DeepJSCC-MMSE model. Our novel design enhances the fidelity of the generated samples, capturing both class-specific features and desired style distributions. This advancement contributes to the overall effectiveness and quality of the generated samples, surpassing the performance of existing approaches in the field. We are offering the diagram to serve as a visual representation of the idea 4.11.

Algorithm 3 Enhanced Image Encoding and Reconstruction Unsupervised

1: Convert class information into one hot encoded vector
2: Apply linear neural network to labels nn.Linear$(10, 1024)$
3: Reshape class vector into $(1, 32, 32)$
4: Randomly generate two numbers from normal distribution
5: Apply linear neural network to numbers nn.Linear$(2, 1024)$
6: Reshape style vector into $(1, 32, 32)$
7: Concatenate class and style embeddings with original image $(5, 32, 32)$
8: Pass concatenated input through Encoder neural network
9: Transmit encoded features through transmission channel
10: Perform MMSE recovery to estimate encoded features : estimated_data $(16, 8, 8)$
11: Split decoded data in $(3, 32, 32)$, $(1, 32, 32)$ $(1, 32, 32)$
12: Apply sigmoid activation function to first vector to recover the image
13: Flatten the style matrix
14: Apply linear neural network to numbers nn.Linear$(1024, 2)$
15: Flatten the class matrix
16: Apply linear neural network to numbers nn.Linear$(1024, 10)$
17: Modify loss function: MSE(image reconstruction) + α * MSE(style output, style input) + β * CE(class output, class input)

4.5.3 Multi-Image Transmission

In this section, we introduce a novel idea inspired by the Gaussian distribution. While conventional approaches involve transmitting a single image to the receiver, we propose a paradigm shift by simultaneously transmitting multiple images. According to the Gaussian theorem, the distribution of these transmitted images is more likely to conform to a Gaussian distribution compared to the distribution of a single image.

To evaluate the effectiveness of this approach, we present results for the scenario where multiple images (e.g., 8 images) are transmitted concurrently. Our findings indicate that our approach learns to allocate power more judiciously compared to the baseline DeepJSCC-MMSE approach, which is based on single image transmission. This intelligent power allocation strategy leads to superior performance, surpassing the results obtained in the baseline method. Our proposed approach demonstrates the potential of leveraging multiple image transmissions to enhance the overall performance of image transmission systems, providing a promising direction for future research and development in this field. We are presenting the diagram to offer a graphical depiction 4.14.

Based on the aforementioned observation, we propose the inclusion of a simple transformer network just before the power normalization step. The aim of this addition is to rearrange the values within the set, anticipating an improvement in performance compared to the initial trial. However, our results indicate that this modification does not

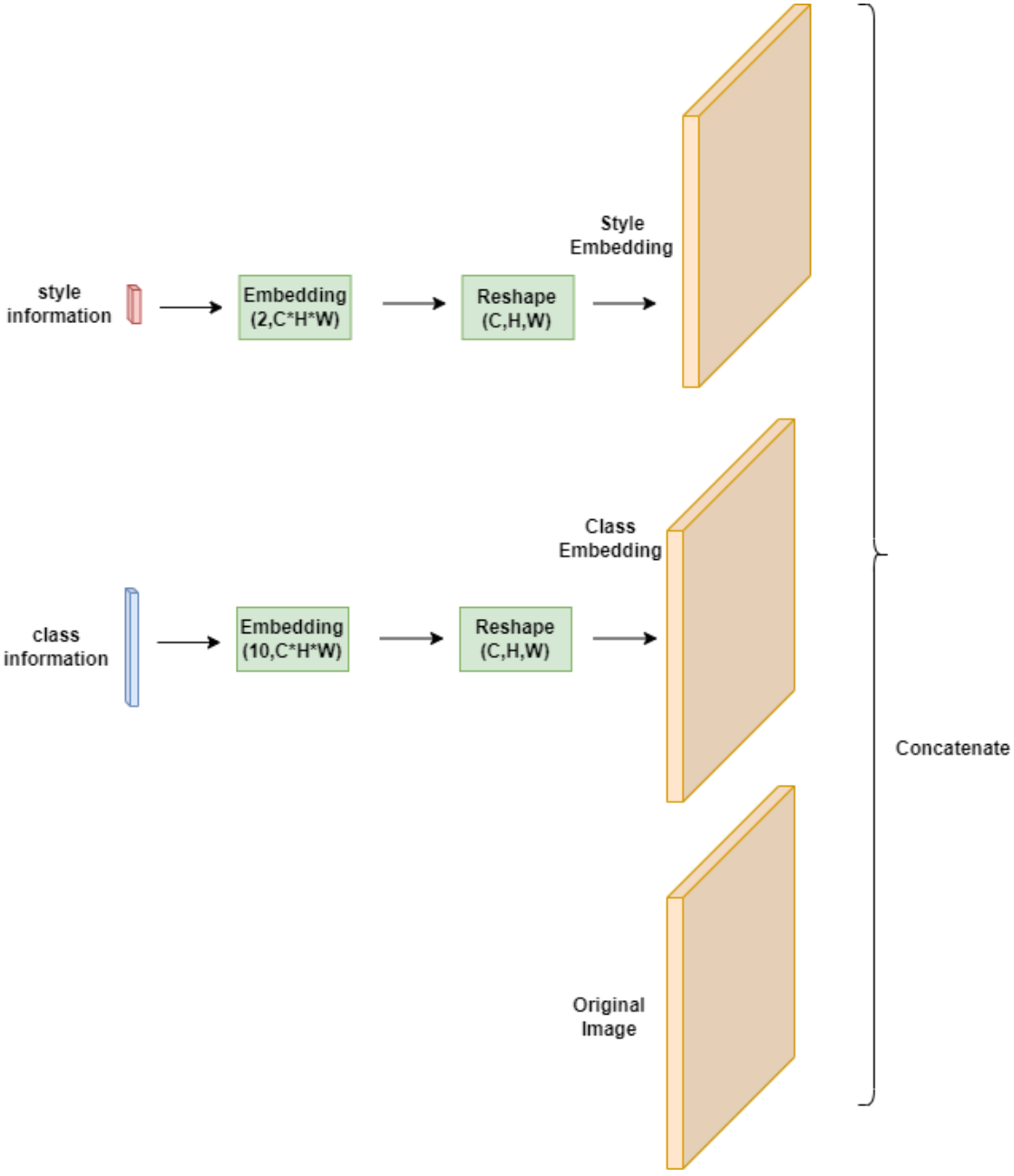


FIGURE 4.11: Proposed Model

Algorithm 4 Multiple Image Transmission

1: Apply encoder network to the batch of images
2: Divide the batch into $(NN, N_{cross-image})$, where $NN \times N_{cross-image}$ is batch size
3: Normalize the input for $N_{cross-image}$ images
4: Transmit encoded features through transmission channel
5: Apply decoder network to the batch of images
6: Calculate the metric

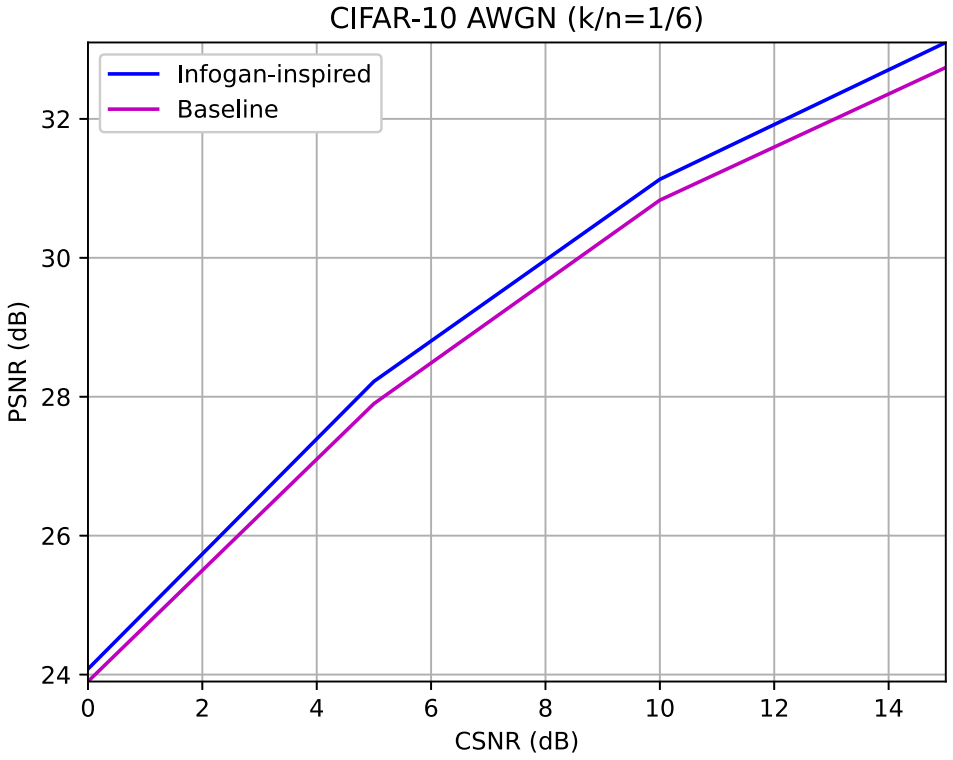


FIGURE 4.12: InfoGAN-inspired DeepJSCC-MMSE and CGAN-inspired DeepJSCC-MMSE tcn=16

lead to significant enhancements. Further investigation is necessary to explore the potential benefits of incorporating transformer architecture to optimize the transmission of multiple images.

Additionally, we experimented with simultaneous compression operations, such as 3D convolution, but the results were comparable to those obtained without these operations. We suspect that the existing techniques, such as batch normalization or generative division normalization (GDN), already account for similar transformations in the single image case. Thus, alternative strategies or architectural modifications may be required to leverage the benefits of compression in the context of multiple image transmission.

4.6 Conclusion

In this chapter, we have presented DeepJSCC-MMSE, a novel end-to-end optimized Joint Source-Channel Coding (JSCC) scheme designed for wireless image transmission. DeepJSCC-MMSE leverages the Minimum Mean Square Error (MMSE) estimator to adapt the decoder to diverse channel conditions, enabling robust performance across a wide range of transmission scenarios.

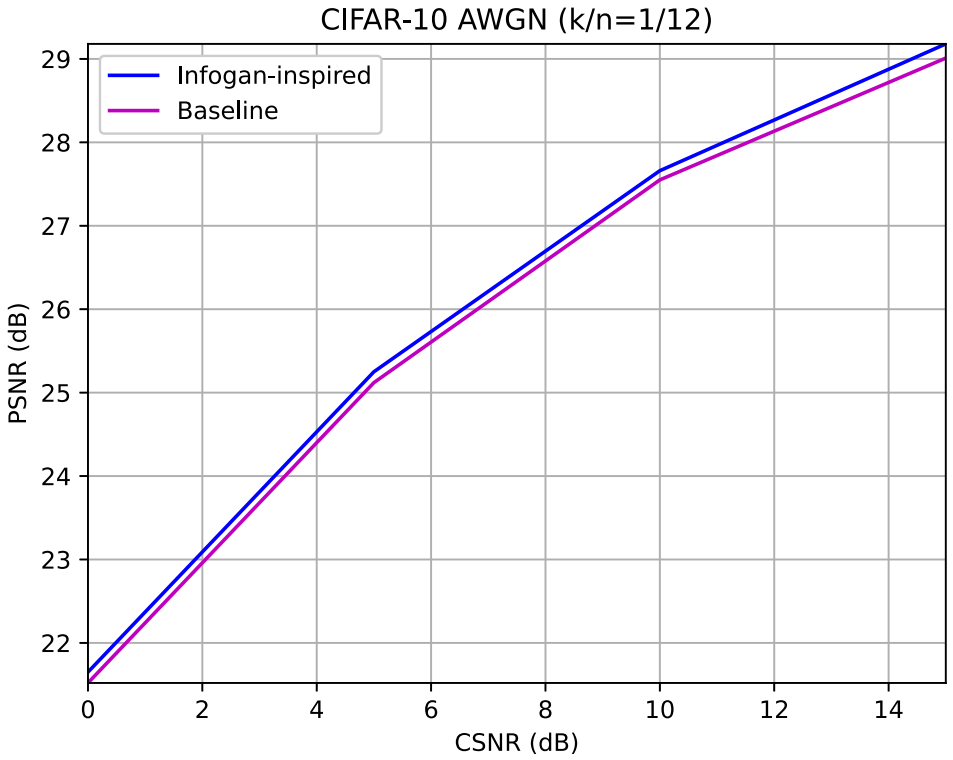


FIGURE 4.13: InfoGAN-inspired DeepJSCC-MMSE and CGAN-inspired DeepJSCC-MMSE tcn=8

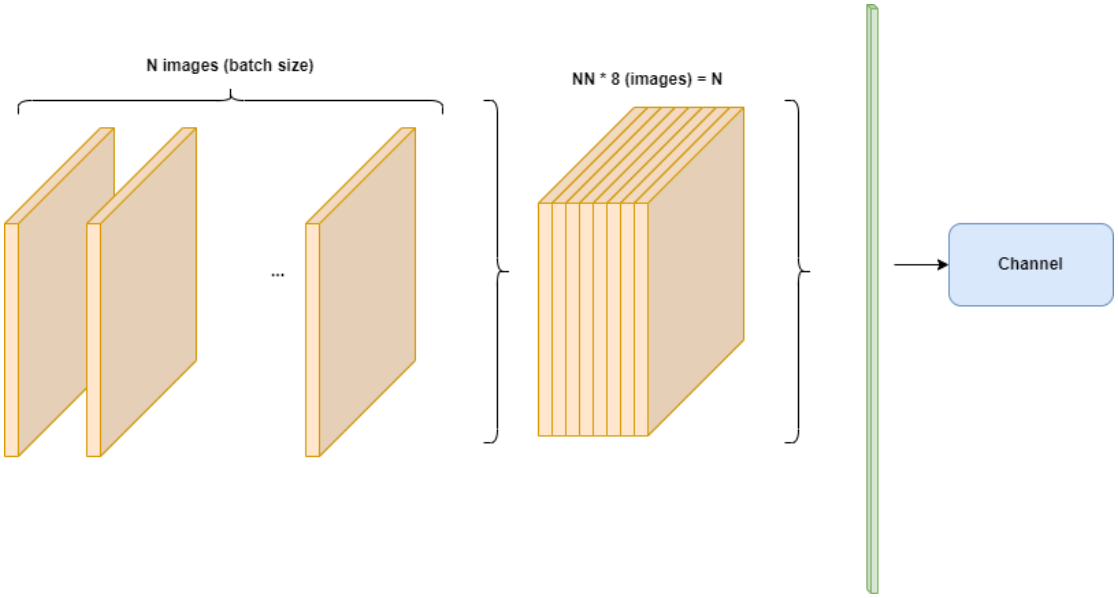


FIGURE 4.14: Proposed Model

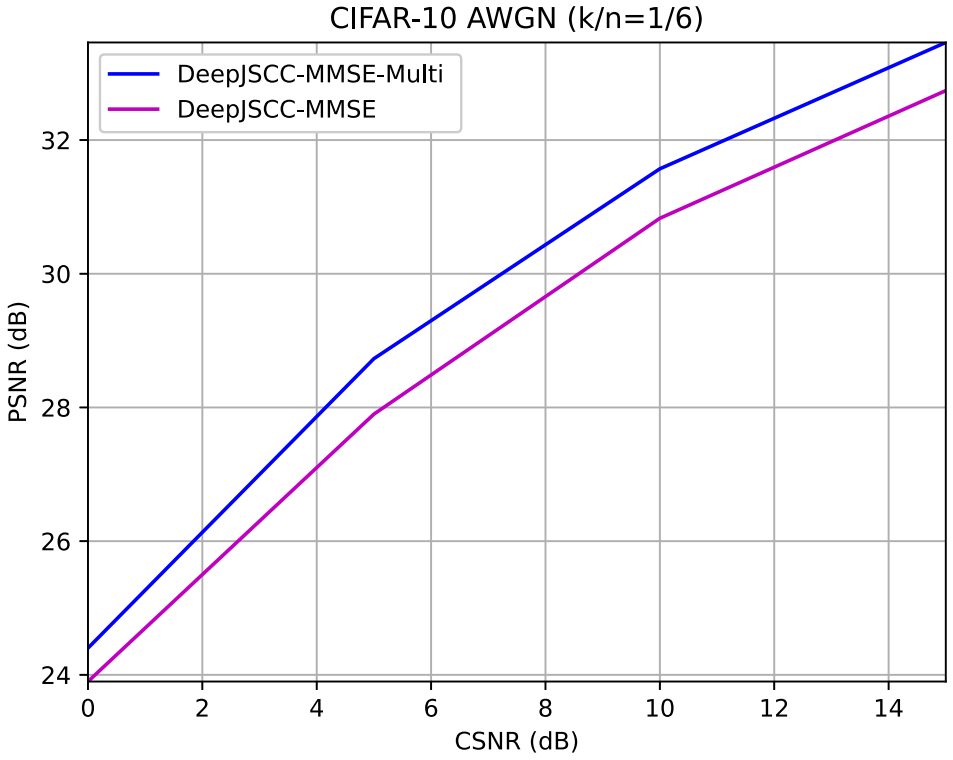


FIGURE 4.15: Multiple image transmission DeepJSCC-MMSE and DeepJSCC-MMSE tcn=16

One notable advantage of DeepJSCC-MMSE is its ability to achieve channel adaptivity without introducing additional parameters or complex algorithms. The scheme is inherently designed to dynamically adjust to varying channel conditions, ensuring optimal performance in different wireless environments.

Furthermore, DeepJSCC-MMSE demonstrates excellent scalability in the context of Orthogonal Frequency Division Multiplexing (OFDM) physical layer transmission. As the number of sub-channels increases, the scheme retains its effectiveness and efficiency, making it well-suited for high-capacity wireless communication systems.

By introducing DeepJSCC-MMSE, we provide a valuable contribution to the field of wireless image transmission, offering a practical and efficient solution for achieving reliable and adaptive communication in challenging wireless environments.

Compared to a deep learning based approach, proposed in [28], DeepJSCC-MMSE achieves similar or better performance under both AWGN and Rayleigh fading channels, with lower complexity. DeepJSCC-MMSE also shows greater robustness towards channel estimation errors than the deep learning based approach in [28]. This chapter is published in WSA-2023 conference.

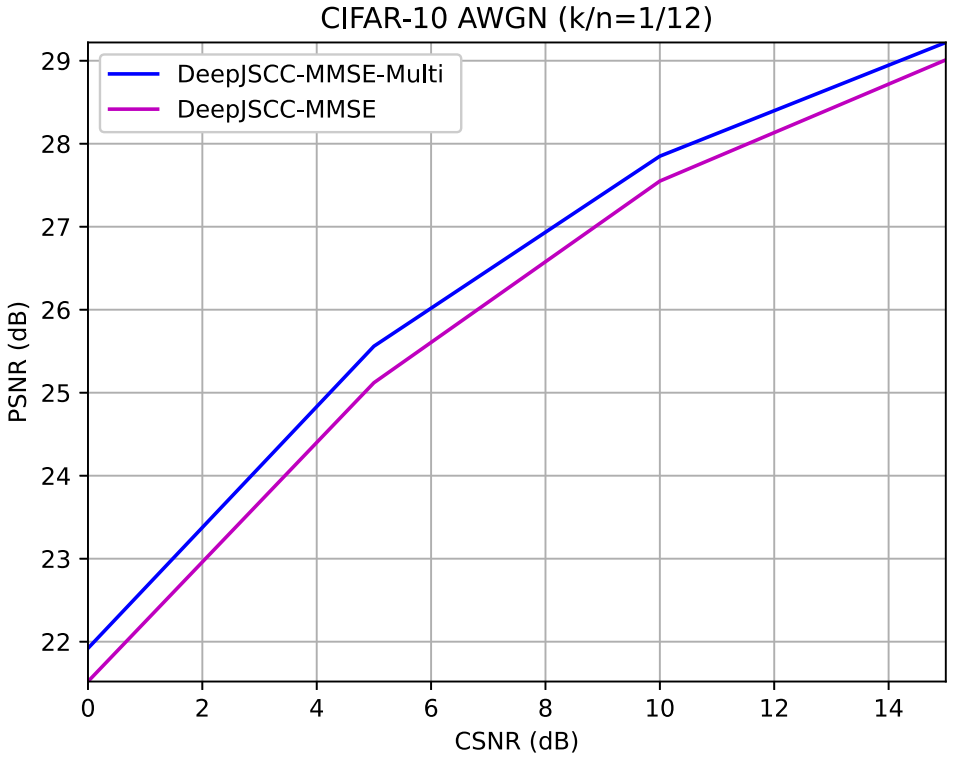


FIGURE 4.16: Multiple image transmission DeepJSCC-MMSE and DeepJSCC-MMSE tcn=8

In addition to our primary research on DeepJSCC-MMSE for wireless image transmission, we further expanded our investigation to explore the potential of using metadata as a watermark in the transmission process. Building upon the insights from InfoGAN, we extended the application of unsupervised learning techniques to this domain.

To evaluate the effectiveness of multiple transmission, we conducted a series of experiments and carefully analyzed the results. Our findings reveal that the incorporation of metadata as a watermark can significantly enhance the overall performance of image transmission, contributing to improved integrity and authenticity of the transmitted images. The utilization of InfoGAN-inspired methods in unsupervised learning further strengthens the ability to learn meaningful representations from the transmitted images.

Furthermore, our research sheds light on the benefits of employing multiple transmission under specific conditions. By transmitting multiple images simultaneously, we observed potential gains in efficiency and throughput, paving the way for enhanced image transmission capabilities in scenarios where higher data rates or improved image quality are desired.

Chapter 5

Variable Bandwidth ratio Deep Joint Source Channel Coding

This chapter presents an innovative method called DeepJSCC with dynamic bandwidth allocation, which addresses the limitations of previous solutions that were designed for a fixed channel bandwidth. In this method, we use an adaptive joint source-channel coding (JSCC) technique for wireless image transmission that operates at multiple bandwidth ratios using a single deep neural network (DNN) model after training. Our approach enables the model to dynamically control the rate based on both the channel condition and image content, while satisfying the target PSNR. To achieve this, we use a control unit that explores the trade-off space between the bandwidth ratio and target metric. We train the control unit using a 'discriminator' network that determines whether the target PSNR is met or not, allowing us to perform end-to-end training of the entire JSCC scheme. Our method offers another solution for deep JSCC schemes that can automatically adjust its bandwidth ratio with a single network model, without compromising on performance or requiring additional resources.

5.1 Introduction

In certain applications, like IoT, where communication resources are limited, it is important to have a JSCC scheme that can dynamically adjust its bandwidth ratio to save channel bandwidth while still transmitting enough source information. However, current deep JSCC methods are usually trained with a fixed bandwidth ratio and require multiple trained models to achieve multiple bandwidth ratio JSCC. It has been studied, but the models are not adaptable to the channel condition as they are only trained with a single SNR and lack a built-in mechanism or policy for automatic rate control with the

multiple bandwidth ratio model. Therefore, in this work, we suggest an adaptive deep JSCC scheme that learns to dynamically control the bandwidth ratio based on the channel condition and image contents using a single deep neural network model. DeepJSCC and other competitors, such as DeepJSCC-MMSE, set the bandwidth ratio before transmission; that is, the same channel resources are used independent of the input image. Here, we consider a slightly different problem in which the receiver wants to reconstruct images at a certain target PSNR value, and we allow variable-length transmission. We adopt DeepJSCC-MMSE in this work to refrain from multiple trainings for each CSNR. Our scheme is created to reduce the use of channel bandwidth in high SNR situations or when the source image has limited information. At the same time, it gives on par with performance in terms of the image quality matches that of a specialized model trained for that bandwidth ratio.

5.2 Related Work

The paper referenced as [80] presents an innovative approach to transmitting images over wireless channels through adaptive-bandwidth technology. The images are divided into layers that are transmitted over time or frequency and can be aggregated to enhance the image quality upon receipt. The authors explore two scenarios: one where layers are sent sequentially to refine reconstruction and another where layers are independent and can be retrieved in any order. This is the first practical multiple-description Joint Source-Channel Coding (JSCC) scheme developed and tested for practical information sources and channels. The authors demonstrate that their proposed method, named DeepJSCC-l, can progressively transmit the source with minimal losses in end-to-end performance when compared to a single transmission. Additionally, DeepJSCC-l performs comparably to state-of-the-art digital progressive transmission schemes in low signal-to-noise ratio and small bandwidth regimes, with the added benefit of graceful degradation with channel SNR. However, this model lacks the ability to adapt to changes in channel conditions as they are trained with a fixed SNR. Furthermore, they do not incorporate a built-in mechanism that facilitates automatic bandwidth ratio control using the multiple bandwidth ratio model.

The paper entitled [89] introduces a new approach to joint source-channel coding (JSCC) is a groundbreaking attempt to address the issue at hand. This innovative technique involves adjusting the transmission rate based on both the channel signal-to-noise ratio (SNR) and the contents of the image being transmitted. The authors employ a single neural network model that includes SNR-adaptive modules [28] to adjust intermediate

features based on SNR. Additionally, the paper introduces a policy network that dynamically determines the number of active features for transmission. The decision process is made differentiable using the Gumbel-Softmax trick. The experimental results of this deep learning model demonstrate that it can be trained to assign different rates to different SNRs and image contents. As a result, bandwidth usage is reduced when SNR is high or the image contains less information, while image quality is maintained at the same level as that of a specialized model. They applied Gumbel-Softmax trick since the training of the policy network is challenging due to the sampling process is discrete. It makes difficult to optimize the network using back-propagation as it is non-differentiable. While the REINFORCE algorithm is commonly used to overcome this issue, it often results in slow convergence and high variance. As a better alternative, they use the Gumbel-Softmax trick, which involves sampling from the Gumbel-Softmax distribution to resolve non-differentiability. This work employs sophisticated architectures, such as attention modules for CSNR-free training and a policy network, which introduces more complexity. It is worth mentioning that CSI is utilized at both the transmitter and receiver in this study.

The objective of the study [90] is to develop a communication system that can complete specific tasks. To achieve this, the researchers used the Information Bottleneck (IB) framework, which balances the rate and distortion. Their method aims to maximize the mutual information between the inference result and the encoded feature, while minimizing the mutual information between the encoded feature and input data. This approach is intended to enhance the accuracy of the inference while reducing the communication overhead. It should be noted that this is the first time the IB framework has been utilized in creating wireless edge inference systems. To simplify the IB formulation for deep neural networks with high-dimensional features, researchers use a variational approximation called the Variational Information Bottleneck (VIB) to create an upper bound. They also employ a sparsity-inducing distribution as the variational prior in the VIB framework to eliminate unnecessary dimensions in the encoded feature vector, reducing communication overhead. This approach is referred to as Variational Feature Encoding (VFE). In addition, they have developed a communication approach that can adapt to changing environments by allowing for adjustments to signal length during transmission. This approach, known as Variable-length variational feature encoding (VL-VFE), is based on dynamic neural networks and can adjust the active dimensions based on the specific channel conditions. They conducted simulations on their proposed task-oriented communication methods for image classification tasks in both static and dynamic channel conditions. Extensive simulations showed that VFE and VL-VFE performed better than traditional communication designs and existing learning-based joint source-channel coding methods for data-oriented communication.

5.3 System Model

Our study specifically examines the scenario of point-to-point communication in an uplink setting, with a single receiver, while addressing the challenge of distributed wireless image transmission over an Additive White Gaussian Noise (AWGN) channel. In our research, we carefully consider the average power constraint as a critical factor in the transmission process. And, the receiver

$$\mathbf{y} = \mathbf{x} + \mathbf{n} \tag{5.1}$$

where the noise is complex Gaussian noise. It is worth mentioning that in our initial proposal, both the transmitter and the receiver have access to Channel State Information (CSI), whereas in our control unit proposal, only the receiver possesses this crucial information. We note here that CSI is available transmitter and receiver at our initial proposal and receiver only in our control unit proposal. We formulate the optimization function as follows:

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{N} R_i \\ \text{subject to} \quad & \text{PSNR}(\mathbf{x}, \mathbf{x}') \geq PSNR_{target} \end{aligned} \tag{5.2}$$

where R_i is the bandwidth ratio of transmitted i-th image. By explicitly addressing the availability and utilization of CSI in our proposals, we aim to explore the potential enhancements in the performance of wireless communication systems under different scenarios.

5.4 Initial Proposal

In our study, we have employed the VL-VFE (Variable Length-Variance Feature Extraction) method to address the challenge of adaptive bandwidth allocation. This method involves sorting the values in descending order of importance and training the network to identify the optimal cutoff point that satisfies the target metric. Although this approach led to a marginal reduction in bandwidth usage, we observed a potential bottleneck in its effectiveness. We hypothesize that the bottleneck arises from the process of ordering the values, which may not align with the appropriate power allocation required for achieving optimal performance. Further investigation is warranted to explore alternative techniques that can overcome this limitation and enhance the efficiency of bandwidth allocation in adaptive systems.

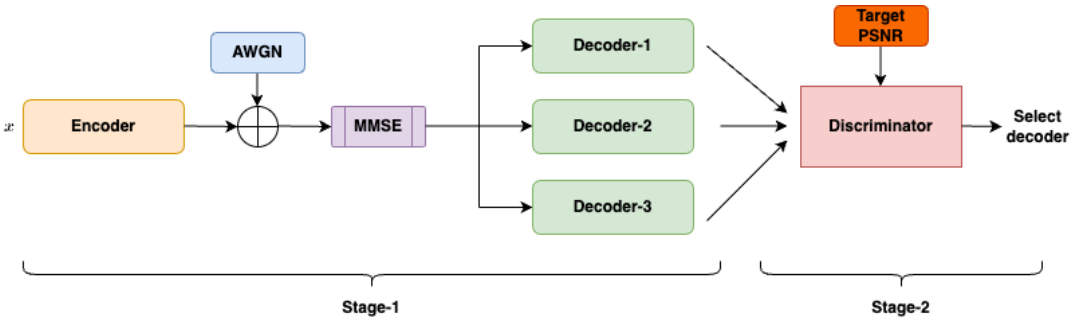


FIGURE 5.1: Proposed solution for adaptive bandwidth

5.5 Proposed Solution

Researchers [80] have studied multiple bandwidth ratio algorithms that involve training a single model. However, these models have a drawback as they lack adaptability to channel conditions since they are trained with only one signal-to-noise ratio (SNR). Furthermore, they do not have a mechanism or policy for automatic bandwidth-ratio control based on the model. In contrast, in [89] the authors introduced a policy network that takes advantage of the trade-off between the rate and the quality of the signal. They utilized the Gumbel-Softmax technique to train the policy network, making it differentiable and enabling end-to-end training of the entire JSCC system. The use of attention modules [91] has been adapted to avoid multiple training caused by varying SNRs. While the authors of a previous study have addressed this issue, we propose a simpler method that requires fewer network parameters than [89] and has lower computational complexity. Our method does not require additional processes such as the Gumbel-Softmax trick to ensure differentiability. At the end, our method outperforms the competitor in terms of PSNR. The pipeline we have established for our study is outline in Fig.5.1. We trained multiple DeepJSCC-MMSE networks for various bandwidth ratios, including 1024, 768, and 512 employing the same architecture in Chapter 4. To ensure that the target performance metric, such as PSNR, is met, we trained a control unit neural network inspired by the discriminator architecture. This control unit determines whether the target metric is satisfied or not. We used the binary cross-entropy loss function, which is commonly used in classification tasks, since our task is to determine whether the target metric has been met or not. The equation is provided as:

$$-(y \log(p) + (1 - y) \log(1 - p)) \tag{5.3}$$

where p confronts for probability and serves as a quantitative measure, providing insights into the likelihood or confidence level associated with a particular outcome, whereas y represents the assigned labels or categories, enabling us to distinguish between different

classes. Therefore, y is the true binary label (0 or 1) $\hat{y}$ is the predicted probability of the positive class (between 0 and 1) The loss function could be written as

$$-\sum_{c=1}^{M} y_{o,c} \log(p_{o,c}) \tag{5.4}$$

which is the total sum of mislabeled outcomes. The binary cross-entropy loss measures the dissimilarity between the true label and the predicted probability. Lower values of the loss indicate better alignment between the predictions and the ground truth labels. The goal in training a binary classifier is to minimize this loss function using optimization techniques like gradient descent. The metric serves as a quantitative measure of the performance. It is important to highlight that the problem addressed in this study holds significant relevance across various applications, including IoT and radar image transmission, where achieving high image clarity is crucial. In these domains, the proposed solution plays a pivotal role in ensuring that specific performance metrics, such as Peak Signal-to-Noise Ratio (PSNR), are met. By setting and surpassing a predefined threshold for image recovery quality, our proposed solution guarantees superior image clarity and fidelity. This capability is instrumental in enhancing the overall effectiveness and reliability of image transmission systems in critical applications where image quality is of paramount importance. This novelty underscores the potential impact of our solution in advancing the field and opening up new possibilities for improving image transmission in critical domains. After making predictions, if the predicted reliability surpasses a specific threshold, the control unit will send an ACK signal to stop the transmission of the current message block and proceed to the next one. However, if the reliability criterion is not met at using first decoder, a NACK signal is sent, and the transmission round will continue. After the last decoder is employed, the transmission will end regardless of the reliability level, and the image reconstructed accordingly. The proposed 'discriminator' design can be summarized in algorithm 5.

5.6 Numerical Results

In this study, we present the benchmark results based on the work of reference [89]. It is important to acknowledge that in our scenario, Channel State Information (CSI) is only available at the receiver. Consequently, it is important to note that the results reported in this study may not reach the same level of performance as studies where CSI is available at both the transmitter and receiver. Nevertheless, we have meticulously trained the neural network as recommended in the original paper, ensuring that our methodology aligns with established practices and guidelines. By presenting these benchmark results,

Algorithm 5 Discriminator Training

Initialize $D(\cdot)$
while condition **do**
  Create real label and fake labels for n decoders
  Recover the transmitted image using n decoders
  if output1 $\geq$ targetPSNR **then**
    label1 = real
  end if
  if output2 $\geq$ targetPSNR **then**
    label2 = real
  end if
  if output3 $\geq$ targetPSNR **then**
    label3 = real
  end if
  Concatenate real labels from n outputs
  Calculate loss function accordingly
  Calculate gradients for backward pass
end while

we contribute to the existing body of knowledge and provide valuable insights into the performance of our approach under the given constraints.

We train our DeepJSCC-MMSE decoders using the Pytorch library [74] on the CIFAR-10 dataset [75] which consists of 60000 32×32 images. We split the dataset into 45000 images for training, 5000 images for validation, and 10000 images for testing. We use a batch size of 128 and train for 500 epochs. We train and test the models with CSNR range $[0, 15]$ dB. We use the Adam optimizer [76] with learning rate 10^{-4}. The learning rate is scheduled according to the validation set, where the learning rate is decreased by a factor of 0.8 if the validation loss does not improve for four epochs consecutively. We use an average power $\bar{P} = 1$ and change the channel noise power σ^2 accordingly to satisfy CSNR. During the training process, we employ a uniform sampling strategy to select Channel Signal-to-Noise Ratio (CSNR) values from the range of 0 dB to 15 dB for each batch of training data. This approach ensures that the neural networks are exposed to a diverse range of channel conditions, allowing them to learn robust features and adapt to various scenarios. Similarly, we follow the same procedure for training the control unit.

It is important to note that if the target Peak Signal-to-Noise Ratio (PSNR) is not satisfied during training, we apply a unique approach. Instead of directly labeling it as 0, we assign a random number generated uniformly from the range (0, 0.1). This

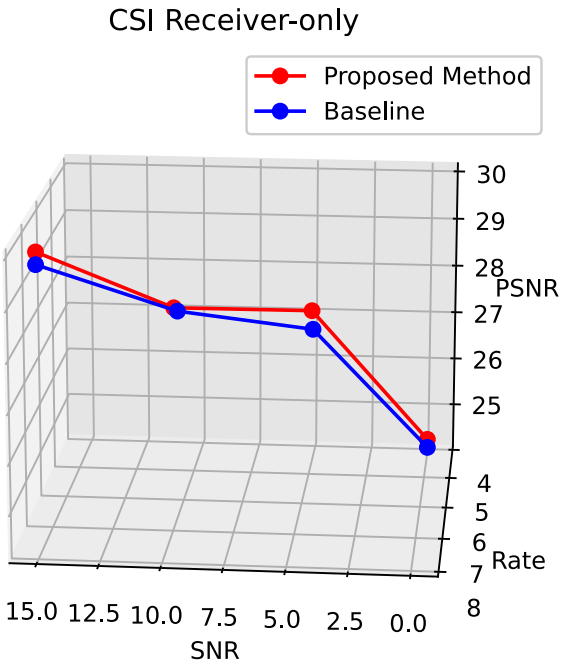


FIGURE 5.2: Proposed solution for adaptive bandwidth

modification, based on our experiments, has shown to accelerate the learning process by relaxing the backpropagation algorithm. By incorporating this strategy, we enhance the network's ability to capture subtle nuances in image reconstruction, leading to improved performance. In Fig. 5.2, we present our noteworthy results, showcasing the effectiveness of our methodology. These results serve as evidence of the successful application of our proposed approach and demonstrate its potential for enhancing image transmission and reconstruction in challenging channel conditions.

5.7 Conclusion

When compressing images for transmission based on bandwidth constraints using DeepJSCC or similar methods, a fixed bandwidth ratio is typically employed. However, this approach can pose challenges as the receiver may only require images of a specific quality, while all images are transmitted at the same compression ratio, even those with low complexity. Consequently, valuable resources are wasted. To address this issue and mitigate gradient-related problems, we have introduced a novel approach that allows for multiple bandwidth ratios. Our method is straightforward, adaptive, and does not rely

on complex algorithms. Importantly, our experimental results have demonstrated its superiority over the baseline method, which utilizes MMSE estimation at the receiver while employing the same architecture. Our proposed approach provides an effective solution to optimize resource utilization and enhance image compression for transmission within bandwidth constraints.

Chapter 6

Model-Driven Distributed Deep Joint Source Channel Coding over a Multiple Access Channel

In this chapter, we are studying how to send different images over a multiple access channel (MAC). The separation theorem is applicable to MACs with independent sources in the infinite blocklength regime; and the capacity region of a MAC can be characterized practical joint source-channel coding (JSCC) with finite block lengths. While Non-Orthogonal Multiple Access (NOMA) is known to achieve the MAC capacity region, a non-orthogonal Joint Source-Channel Coding (JSCC) scheme for practical systems has not been previously studied. Our experiments show that our approach improves the quality of reconstructed images compared to current DeepJSCC approaches, especially for low bandwidth ratios.

6.1 Introduction

Consider two users trying to send two independent images to a destination node over the same channel. One simple method is to use time division multiple access (TDMA) where each user transmits using half of the channel bandwidth using the same trained DeepJSCC network presented in the previous chapters. However, the aim of our research is to create a DeepJSCC framework that can take advantage of non-orthogonal communications.

In this chapter, we will propose two different methods. First is the initial attempt which employs iterative decoding techniques with multiple decoders. Second is the method

which employs the findings in [92] merging the Chapter 4's model-based techniques. Both models outperform the TDMA based DeepJSCC multiple user scenario. The former outperforms the results around 0.2dB in exchange for decoder complexity. The latter outperforms the TDMA based competitor significantly with acceptable increase in computational complexity.

In order to achieve the capacity region of a MAC, NOMA is necessary, while orthogonal transmission methods like TDMA are not as effective. Despite the fact that signals from different users can cause interference, it is possible to achieve higher rates and reach capacity using either joint decoding [93] or successive interference cancellation (SIC) with message splitting [94].

In the mean time, attempts have been made to use DNNs for implementing SIC in NOMA, as evidenced by [95–97]. The use of continuously varying amplitude signals in transmission can facilitate graceful degradation with changes in channel quality. However, this also means that the decoder functions as an estimator and will inevitably introduce some noise in the reconstructed signal. In contrast to digital communication, the decoder cannot completely recover the transmitted codeword, resulting in the inability to achieve perfect interference cancellation. Distributed compression [98] typically utilizes distinct encoding and decoding functions for each source signal, leading to increased complexity and a lack of scalability.

6.2 System Model

Our study focuses on the scenario where two transmitters communicate with a single receiver in an uplink setting, considering the problem of distributed wireless image transmission over a MAC. We set the average power constraint at both transmitters. And, the receiver receives:

$$\mathbf{y} = \mathbf{x}_1 + \mathbf{x}_2 + \mathbf{n} \tag{6.1}$$

where the noise is complex Gaussian. We note here that CSI is available receiver only and the loss function is written:

$$\mathcal{L} = \sum_{x_1, x_2} \text{MSE}(x_1, \hat{x_1}) + \text{MSE}(x_2, \hat{x_2}) \tag{6.2}$$

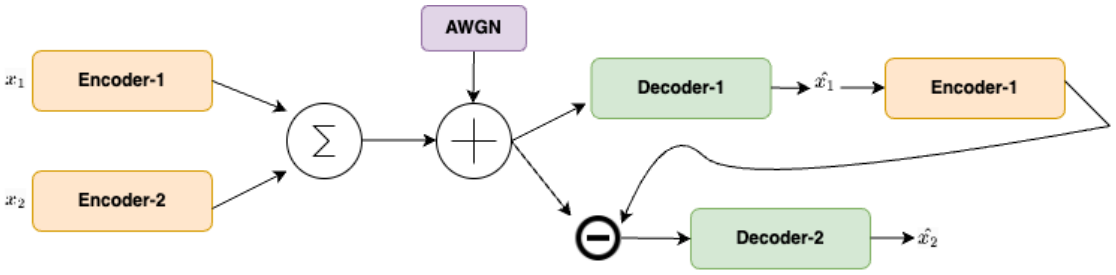


FIGURE 6.1: Initial Proposal

6.3 Complex Attempt

In this section, we will discuss our first attempt to solve the problem. We used n decoders at the receiver and an encoder to estimate the encoded signal at the first transmitter. Although it improved the results by approximately 0.2dB, it was not feasible due to the computational complexity. It is important to note that even though we trained multiple iterations, the complexity increased and the results only slightly improved. Therefore, we explored different approach which does not bring into computation complexity.

6.4 Novelties

Multi-Task Learning (MTL) is a powerful technique that involves jointly training multiple tasks to enhance overall performance [99,100]. In the context of the MAC architecture, it combines multiple signals into a joint representation, allowing the decoder to reconstruct multiple images simultaneously, inherently transforming it into a MTL problem.

MTL methods often lead to improved performance for both tasks by either adjusting the feature space or sharing parameters between tasks. Drawing inspiration from these MTL research techniques, we propose a novel input augmentation embedding approach to enhance separability in our MAC model. Additionally, to further reduce the number of parameters, we leverage a Siamese neural network-based architecture [101].

By integrating MTL principles, our MAC model benefits from the synergistic relationship between tasks, leading to better representations and more efficient parameter utilization. The proposed input augmentation and Siamese architecture contribute to improved overall performance, effectively addressing the problem at hand.

Like Deep Metric Learning (DML), our approach involves projecting multiple instances into a lower-dimensional latent space using the same network. In DML, only a subset of pairs are typically used for training, and the choice of training pairs has a significant impact on performance [102]. As a result, we have developed a systematic and

efficient training methodology instead of using all potential training pairs, which would be impractical.

Curriculum Learning (CL) is a training method that involves gradually training a network on simpler tasks before adapting it to the main task of interest [103]. CL has already been used in the literature. For example, in [104] authors start training process at higher SNRs and gradually adjusts to lower SNRs. Distributed compression methods, such as the one described in reference [105], use different settings for the encoding process on each device. However, this approach requires a lot of resources for training because it involves a significant number of parameters. To address this issue, we propose using a Siamese neural network architecture [101] that allows the two encoders in the transmitter devices to share their parameters. We present a technique for enhancing the distinctness of signals in the aggregate representation domain, without making substantial modifications to the successful DeepJSCC architecture or substantially increasing the number of parameters. This is achieved by implementing an input augmentation method.

Rather than using individual decoders for each image at the receiver, we have implemented a method that utilizes image-specific parameters only in the final layer of the decoder. This approach reduces the number of parameters and simplifies the training process while also improving the separation of transmitted signals. The decoder produces an image tensor with dimensions $2C_{\text{in}} \times W \times H$, which is then reshaped into two separate images with a shape of $2 \times C_{\text{in}} \times W \times H$. It is worth noting that while device embeddings introduce new trainable parameters, they only depend on the image size and do not increase with the number of devices. Figure 6.3 presents the block diagram. We are including the diagram to provide a visual representation that clarifies this concept 6.2. We are presenting an algorithm to enhance our current task 6.

Algorithm 6 DeepJSCC-NOMA-MMSE Algorithm

1: Prepare datasets
2: Employ device embeddings
3: Encode
4: Employ power normalization
5: Transmit over the sanme shannel
6: Perform MMSE estimation
7: Decode
8: Calculate the metric

6.5 Numerical Results

The CIFAR-10 dataset is utilized for both training and testing in our study, comprising 50,000 instances of training data and 10,000 instances of test data. To conduct training,

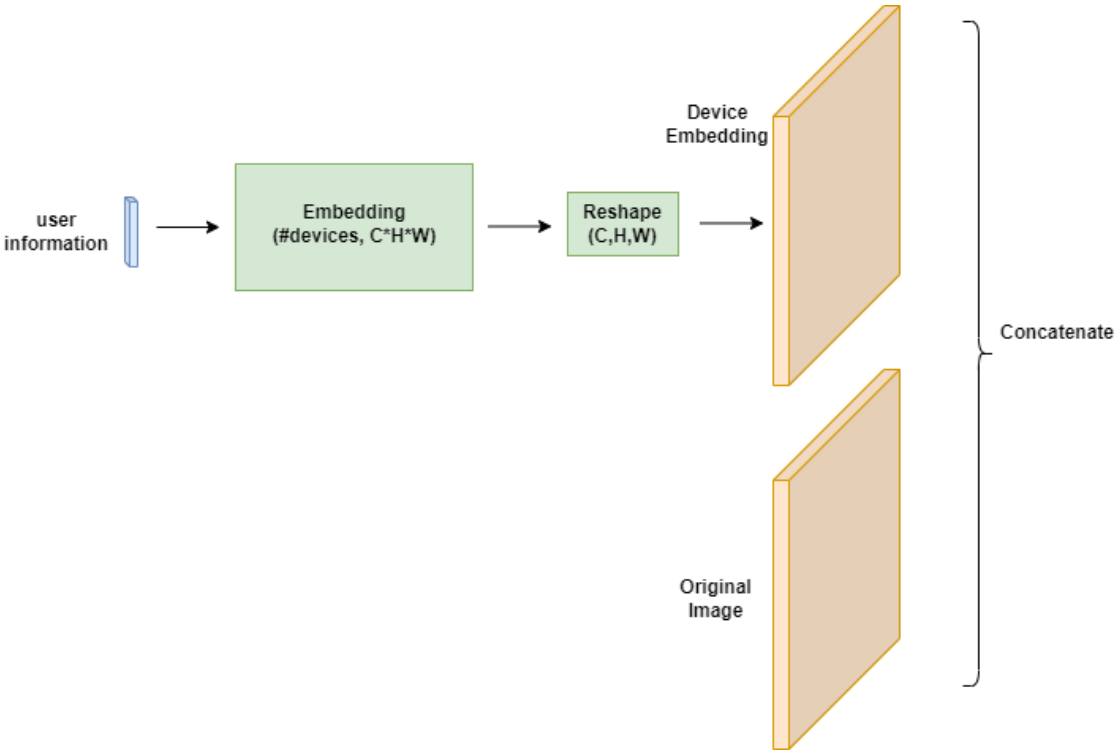


FIGURE 6.2: Proposed Model

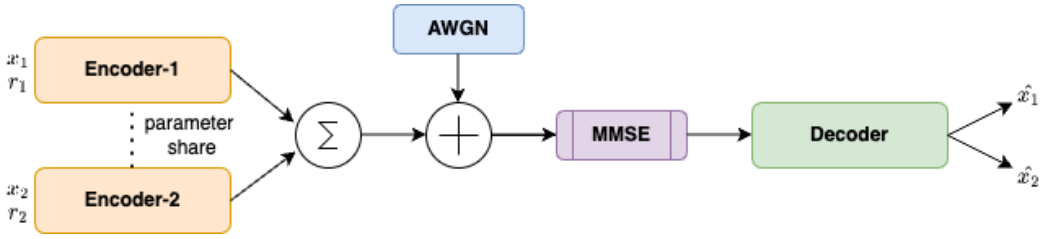


FIGURE 6.3: Proposed Method

we divided the dataset into two subsets, with 45,000 instances allocated for training and 5,000 instances used for validation.

We continue the training process until the model shows no improvement for 10 consecutive epochs. Throughout the training and validation stages, we apply different signal-to-noise ratio (SNR) values to each instance, randomly selected from a range of [0, 15] dB. We report the testing outcomes for every SNR value with the identical model used during training. Instead of using 45,000 x 45,000 pairs as described in Section IV-C, we randomize the training instances or pairs before each epoch. We train our models using the Pytorch library [74] on the CIFAR-10 dataset [75] which consists of 60000 32×32 images. The proposed method utilizes a set of training data consisting of 200,000 distinct pairs as [92] suggested. We use a batch size of 64 and train until no improvement is observed for consecutive 10 epochs. We train and test the models with CSNR range $[0, 15]$ dB. We use the Adam optimizer [76] with learning rate 10^{-4}. We use an average

power $\bar{P} = 1$ and change the channel noise power σ^2 accordingly to satisfy CSNR. During training, we sample a CSNR value uniformly from the set $[0-15]$dB per batch of training data in order to expose the networks to a wide range of channel conditions. In this part, we will present the results accordingly. We compare our proposed method with TDMA version of DeepJSCC-MMSE. The results show that the proposed architecture exploits bandwidth better than TDMA. Just to note that DeepJSCC-MMSE-NOMA performs slighlty better than Attention-Module version that we mentioned in 4.

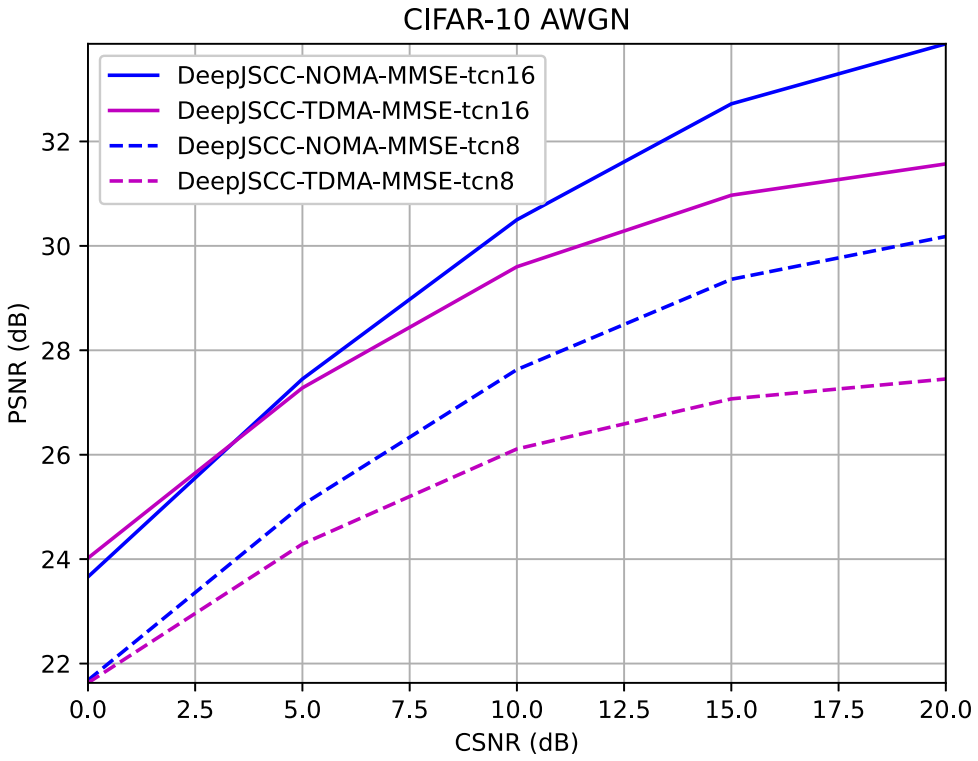


FIGURE 6.4: Performance of DeepJSCC-NOMA-MMSE and DeepJSCC-TDMA for AWGN.

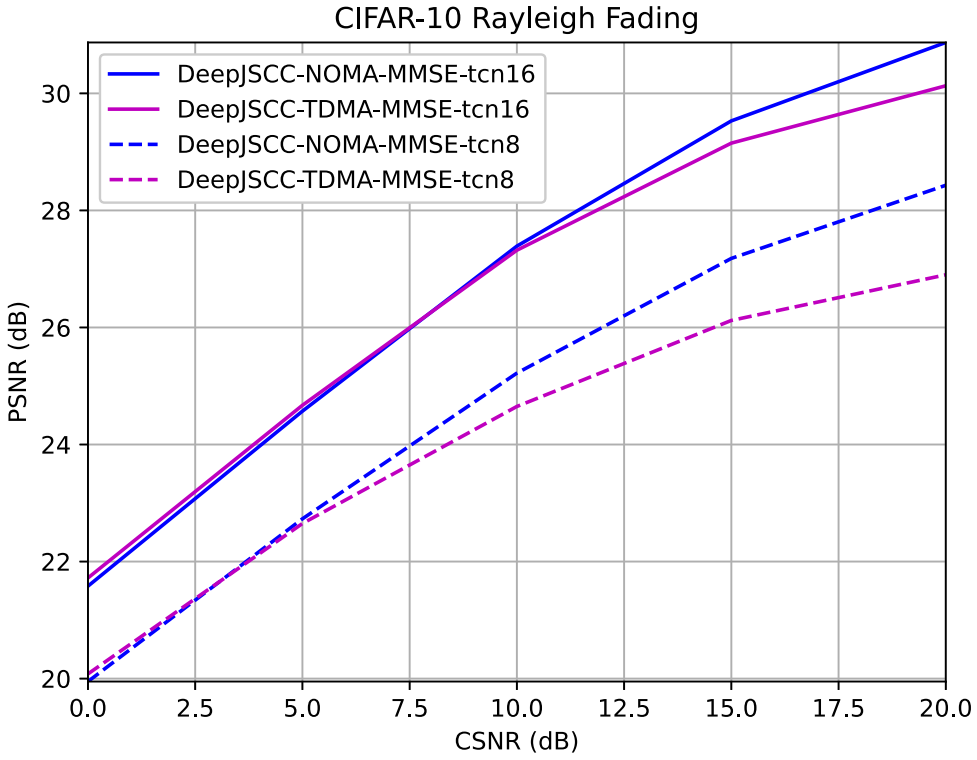


FIGURE 6.5: Performance of DeepJSCC-NOMA-MMSE and DeepJSCC-TDMA for Rayleigh Fading.

6.6 Conclusion

We have introduced an innovative method for compressing and transmitting images in a multi-user uplink scenario. Our approach leverages the power of Non-Orthogonal Multiple Access (NOMA), where all transmitters utilize the same Deep Neural Network (DNN) architecture. To overcome the challenges posed by the analog transmission process in DeepJSCC, we have devised a unique input augmentation technique. This enables the receiver to successfully restore both transmitted images and accurately attribute each image to its respective transmitter. Our experimental results demonstrate that our proposed technique, DeepJSCC-NOMA, surpasses its time-division counterpart. Furthermore, by incorporating Minimum Mean Square Error (MMSE), we have successfully trained the neural network without being limited by the Channel Signal-to-Noise Ratio (CSNR). These findings highlight the effectiveness and superiority of DeepJSCC-NOMA-MMSE in image compression and transmission for multi-user scenarios.

Chapter 7

Deep Generative Models in Deep Joint Source Channel Coding Wireless Image Transmission

In this chapter, we mainly focus on exploiting deep generative models for DeepJSCC, such as variational autoencoders, normalizing flows, generative adversarial networks (GANs) and diffusion models. Although these models are widely used in computer vision tasks such as object detection, image compression, segmentation and classification, their usage in the transmission is overlooked so far. Their main usage in the CV domain is 'generating' new images after getting the training dataset's distribution. Their main difference is the learning strategy to learn dataset distributions. They have shown great success in generating high-quality samples, but each has some limitations. The basic one, VAE generates an image that 'interpolates' the data distribution; however it uses surrogate loss (KL divergence loss). GANs are effective tools; however, they are notorious for their adversarial training nature. Flow-based methods construct complex distributions using simple distributions in multiple steps. But, they have to use specialized architectures to create reversible transforms. Lastly, diffusion models are proposed and have already attracted attention nowadays. Since Markov chain is defined for diffusion steps to slowly add random noise to data and then learn to reverse the diffusion process to construct desired data samples from the noise, unlike the other methods, diffusion models are learned with a fixed procedure, and latent variables have high dimensionality. GAN-level sample quality without adversarial training, flexible model architectures, exact log-likelihood computation and inverse problem solving without re-training models are pros of this method on the other hand computational complexity and sampling time are cons. Finally, diffusion models connect to normalizing flow models, allowing exact likelihood computation and representation learning.

7.1 Introduction

Deep Generative Models (DGMs) are a class of machine learning models that are designed to generate new data samples that are similar to those in a given training dataset. They are typically based on neural networks, and are trained using a variant of unsupervised or semi-supervised learning.

The most popular types of DGMs are Variational Autoencoders (VAEs) and Generative Adversarial Networks (GANs). VAEs are based on the idea of learning a probabilistic latent representation of the data, and then using this representation to generate new samples. GANs, on the other hand, consist of two neural networks: a generator network that generates new samples, and a discriminator network that tries to distinguish the generated samples from the real ones. The generator and discriminator networks are trained together, with the generator trying to produce samples that can fool the discriminator, and the discriminator trying to correctly identify the generated samples.

DGMs have been applied to a wide range of tasks such as image generation, text-to-speech synthesis, and music generation. They have also been used to generate synthetic data for tasks such as data augmentation, domain adaptation, and anomaly detection. In this chapter, we will be proposing them for wireless image transmission.

7.2 VAE

A Variational Autoencoder (VAE) is a generative model that learns the probability distribution of input data. It comprises two main components: an encoder network and a decoder network. The encoder network maps input data to a lower-dimensional representation, known as the latent space. The decoder network then maps the latent space back to the original input space, reconstructing the initial data.

Compared to traditional autoencoders, VAEs learn a probabilistic mapping between the input and latent spaces. The encoder network approximates the actual probability distribution of the data, while the decoder network generates new samples from this distribution.

The process of training VAE involves maximizing the likelihood of data under a learned probabilistic model while keeping the latent space as simple as possible. To accomplish this, a regularization term called the 'KL-divergence' is added to the loss function, which measures the difference between the learned latent space distribution and a prior distribution.

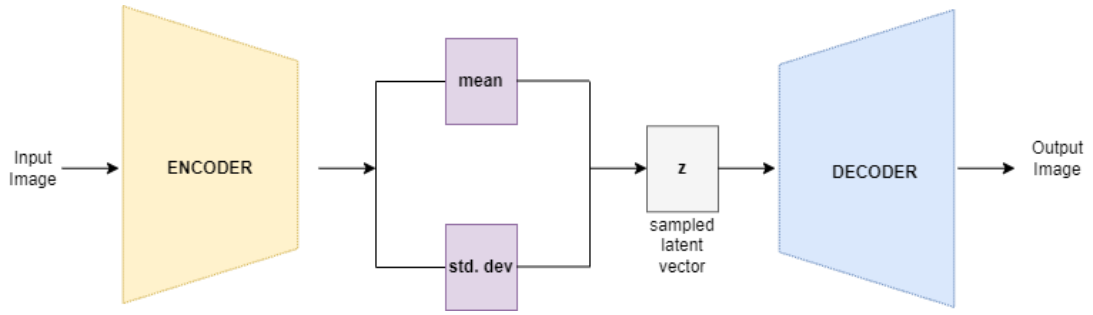


FIGURE 7.1: VAE

VAEs have been used for various tasks, including image generation, video prediction, and anomaly detection. They can also be combined with other models like GANs to enhance the quality of generated samples. The block diagram in Figure 7.1 illustrates the proposed architecture. Research [27] has shown that Variational Autoencoders (VAEs) are more effective than Autoencoders (AEs) in scenarios where channel mismatch occurs during wireless image transmission. VAEs have a superior performance when the actual channel performance is poorer than the trained Channel Signal-to-Noise Ratio (CSNR) because they learn the distribution instead of the actual latent dimension. This results in the generation of better quality images in comparison to AE's reconstruction, especially when there is a high level of noise present. However, the paper suggests that when the train and test Signal-to-Noise Ratios (SNRs) match, AE outperforms VAE due to the manifold hypothesis. This hypothesis states that real-world high dimensional data is made up of low dimensional data embedded in a high dimensional space. The structure of the data can be described utilizing only a few dimensions. The AE's latent space is disjointed and non-continuous, meaning that inputs are mapped deterministically to a latent vector. In contrast, VAE maps inputs to a probability distribution, which can result in gaps in the latent space and decoder issues. To create a latent vector, we map the input to a mean vector $\mu(x)$ and a vector of standard deviations $\sigma(x)$. These parameters define a diagonal Gaussian distribution, from which we sample a latent vector. To accomplish this, we replace the last layer of a traditional autoencoder with two layers that output the mean and standard deviation. Additionally, we introduce a KL divergence term to the loss function. This term serves two purposes: first, it allows us to train the encoder network by gradient descent, and second, it encourages the latent vectors to be more centralized and uniformly distributed. This forces the encoder to find latent vectors that follow a standard Gaussian distribution that the decoder can effectively decode. This is similar to the concept in wireless communication theory, where a Gaussian input to the AWGN channel leads to optimality. If this condition is met, the system operates more efficiently.

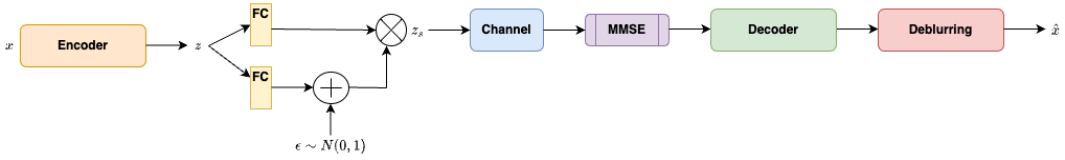


FIGURE 7.2: Proposed Method

Our approach to wireless image transmission is based on utilizing Variational Autoencoder (VAE) and optimal power allocation, independent of Channel Signal to Noise Ratio (CSNR) values. Our understanding of the mathematical background presented in Chapter 3 has led us to use the VAE's latent vector, which generates a Gaussian distribution, making it suitable for optimal power allocation, even though it uses a surrogate loss. Our research has shown that using the SIREN activation function, with ω_0 set to 0.3 as empirically chosen, produces better image results. This has been confirmed in another study [106]. While our results are not currently better than DeepJSCC-MMSE, they are promising, and VAE may become an excellent tool for wireless image transmission. However, we must note that using Autoencoders (AEs) and VAEs with the Mean Squared Error (MSE) metric can cause images to appear blurry due to the 'moderating' effect at the decoder. Therefore, it is crucial to consider this effect and optimize the parameters accordingly to improve the overall quality of the transmitted images.

Based on the chosen metric, which is the mean squared error (MSE), autoencoders (AEs) and variational autoencoders (VAEs) have a tendency to produce 'moderate' values at the decoder, resulting in blurry images. This is a common issue in VAE architectures due to their inherent mathematical function. To address this issue, we propose a post-processing step to recover the blurry images. Instead of using the blurry output from the VAE, we apply a de-blurring algorithm at the end. Our experimental results show positive improvements. It is important to note that the training procedure involves two stages: first, the VAE's encoder and decoder are optimized according to the chosen metric, and then the post-processing block is optimized based on the same metric. We use a simple post-processing block consisting of multiple ResNet blocks (e.g., 8 blocks). However, there are other neural-network-based filters that can be used for de-blurring images, and further investigation is necessary to determine their effectiveness. We train our models using the Pytorch library [74] on the CIFAR-10 dataset [75] which consists of 60000 32×32 images. We split the dataset into 45000 images for training, 5000 images for validation, and 10000 images for testing. We use a batch size of 128 and train for 500 epochs. We train and test the models with CSNR range $[0, 20]$ dB. We use the Adam optimizer [76] with learning rate 10^{-4}. The learning rate is scheduled according to the validation set, where the learning rate is decreased by a factor of 0.8 if the validation loss does not improve for four epochs consecutively. We use an average power $\bar{P} = 1$

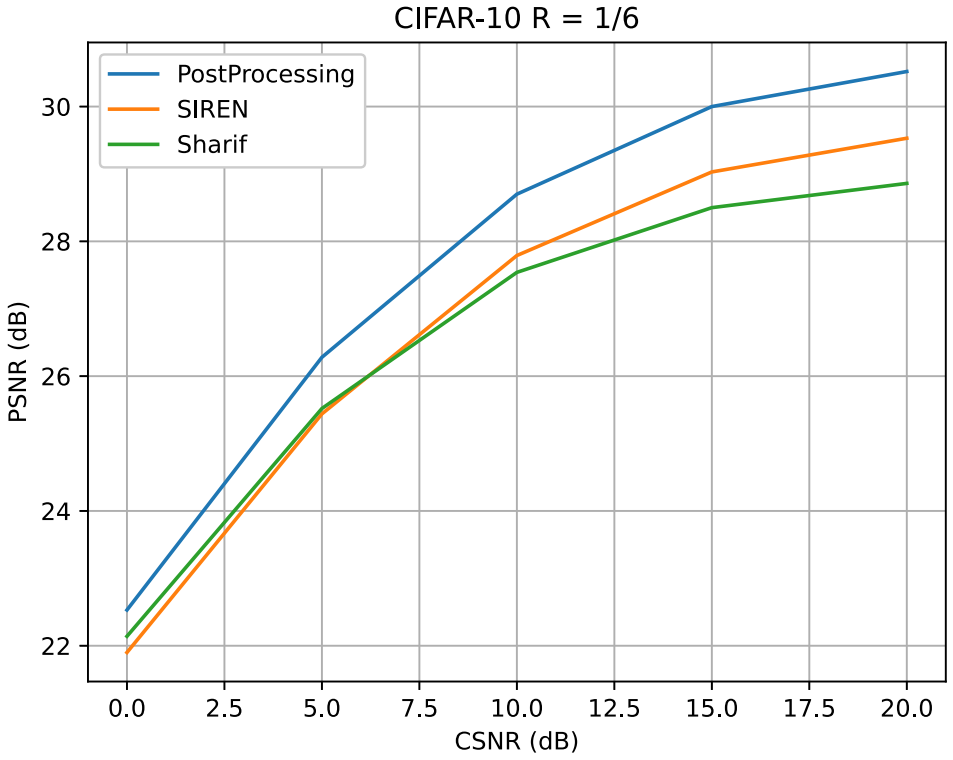


FIGURE 7.3: Comparison

and change the channel noise power σ^2 accordingly to satisfy CSNR. During training, we sample a CSNR value uniformly from the set $[0 - 20]$dB per batch of training data in order to expose the networks to a wide range of channel conditions.

Based on our experimental results, we have demonstrated that the use of the SIREN activation function in our proposed approach outperforms the baseline method [27]. It is worth noting that we used the same architecture and MMSE estimation in both methods. Additionally, our findings suggest that employing postprocessing techniques can significantly improve overall performance. We conclude that while VAEs are known to outperform AEs in many applications, powerful postprocessing algorithms can potentially outperform the results achieved with AEs in wireless image transmission.

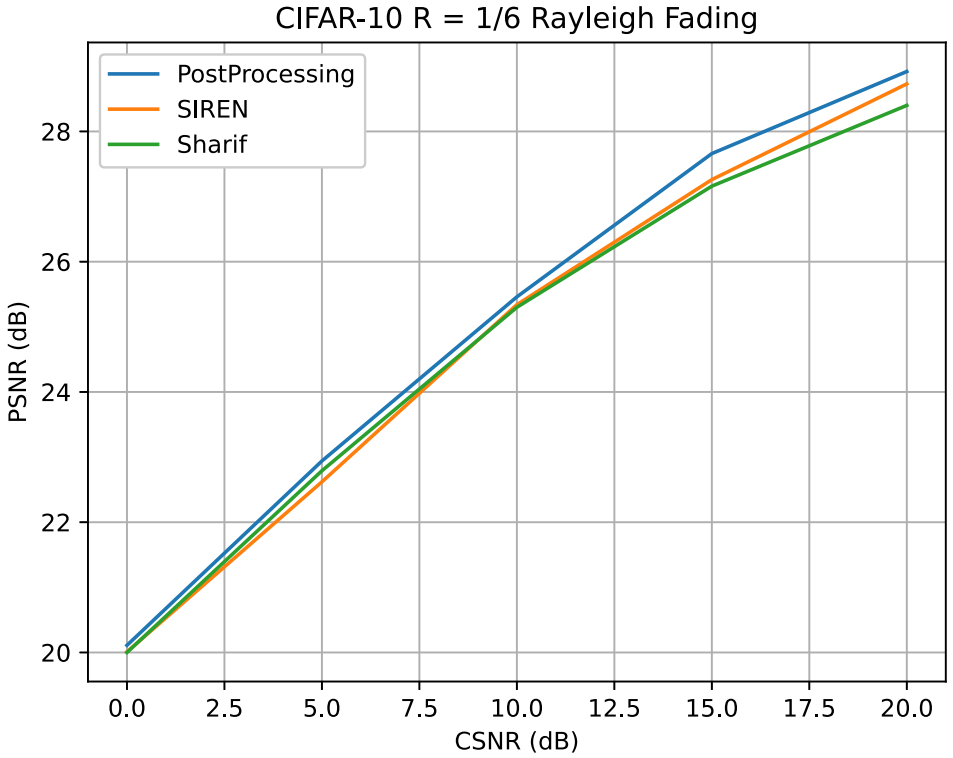


FIGURE 7.4: Comparison

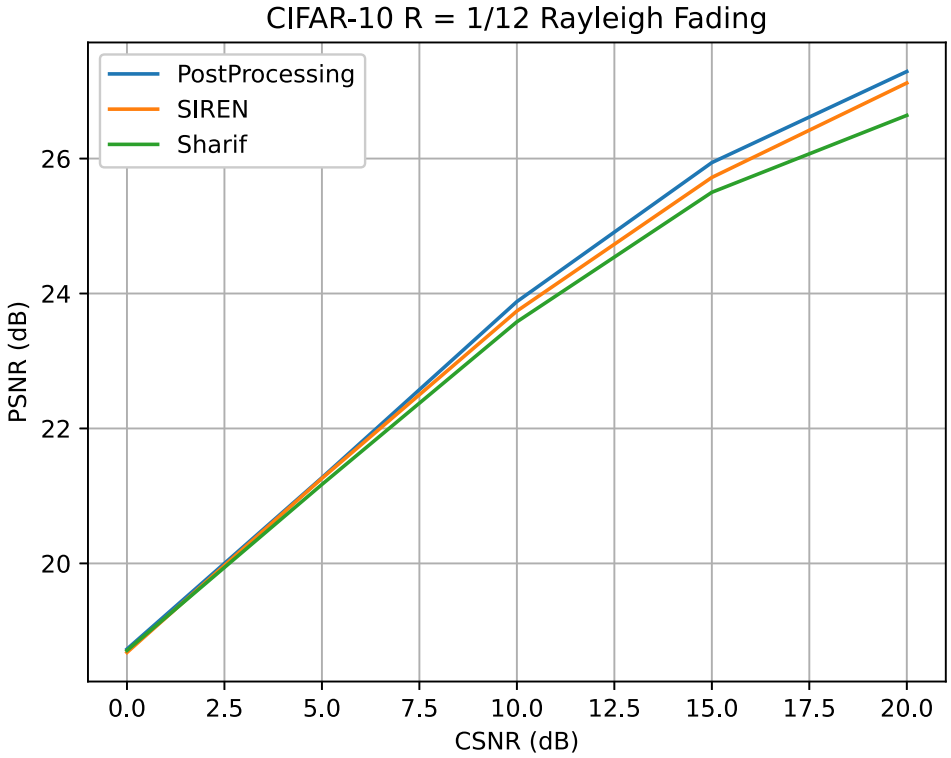


FIGURE 7.5: Comparison

7.3 Normalizing Flows

Normalizing flows are a machine learning model class used for density estimation and generative modeling. They transform a simple probability distribution into a more complex one, which approximates the true data distribution, by using a sequence of invertible, differentiable transformations. These transformations are chosen from a set of parameterized functions, such as affine or radial basis functions, and their parameters are learned during training to minimize the difference between the approximated and true data distribution. Normalizing flows can model highly structured, high-dimensional data distributions, such as images or audio, and can handle large amounts of data. Planar flows, radial flows, and coupling flows are popular types of normalizing flows.

Real-NVP (Real Non-Volume Preserving) [107] is a type of normalizing flow model that has gained attention in the machine learning community due to its ability to generate high-quality samples that closely resemble real-world data. Real-NVP works by using a series of invertible transformations to map a simple random noise distribution to the target data distribution. The key to Real-NVP's success lies in its use of a specific architecture called a 'coupling layer', which employs a neural network to learn the necessary

transformations. This architecture is also supplemented by a technique called 'masking', which ensures that the generated samples are highly structured and maintain the same underlying patterns as the real data.

Real-NVP has been used effectively in a variety of applications, such as image generation, density estimation, and variational inference. It has shown promising results in modeling complex data distributions, including high-dimensional data such as images and audio. Additionally, Real-NVP's ability to model complex distributions makes it a useful tool in a wide range of applications, from generating realistic images to simulating complex systems.

In our approach, we aim to transform the VAE's output latent space distribution into a more optimal one for wireless transmission without sacrificing too much information about the data. To achieve this, we utilize normalizing flows to approximate the optimal distribution, gradually increasing the encoder capacity until the desired transformation is achieved. The use of normalizing flows has been shown to be effective in changing the data distribution in a beneficial way. We insert the normalizing flow module between the VAE output and the power normalization input, as demonstrated in the paper by Rezende et al. [108]. Our approach builds upon the work of Han et al. [109], who have already demonstrated the effectiveness of using normalizing flows for wireless image transmission.

7.4 GAN

GANs are trained in an adversarial way where the generator tries to fool the discriminator into thinking that its generated samples are real, while the discriminator tries to correctly identify the real samples from the generated ones. During training, the generator is updated to produce samples that are increasingly difficult for the discriminator to differentiate from real samples. GANs have shown impressive results in generating high-quality images, audio, and text. They have been used in a variety of applications, such as image synthesis, image-to-image translation, and data augmentation. One of the challenges of GANs is their instability during training, which requires careful hyperparameter tuning and training strategies to ensure convergence.

To be more precise, the GAN training process involves optimizing both the generator and discriminator networks simultaneously in a minimax game. The generator takes as input a random noise vector and produces a new sample, while the discriminator takes as input a sample from either the training data or the generator, and outputs a probability score indicating whether the sample is real or fake. The generator aims to produce samples

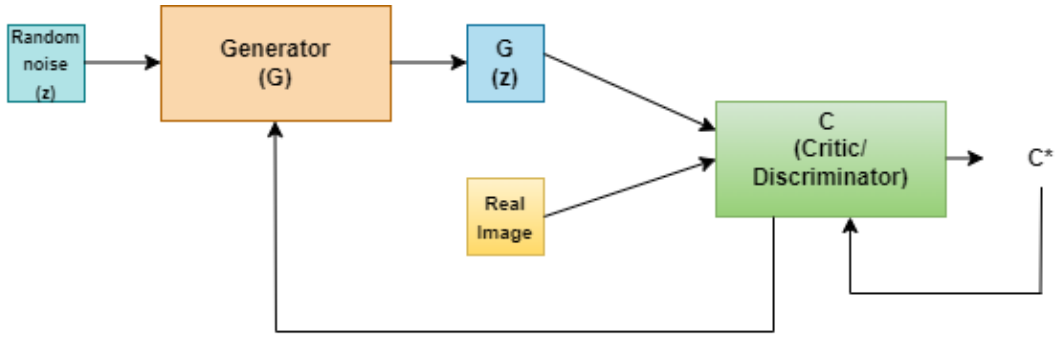


FIGURE 7.6: Block diagram of WGAN.

that have a high probability of being classified as real by the discriminator, while the discriminator aims to correctly classify real samples with high probability and generated samples with low probability. Through this adversarial training process, the generator gradually learns to produce samples that are more realistic, while the discriminator becomes better at distinguishing between real and generated samples. Once training is complete, the generator can be used to generate new samples that closely resemble the real training data.

In recent years, GANs have undergone significant improvements to address issues such as mode collapse, instability during training, and lack of diversity in generated samples. Various techniques have been proposed to improve GANs, such as Wasserstein distance, gradient penalty, and self-attention mechanism. These techniques have been shown to improve the stability of GAN training and enhance the quality of generated samples.

Moreover, new architectures such as Progressive GANs [110] have been introduced to generate high-quality, high-resolution images in a progressive manner by gradually increasing the resolution of generated images. Similarly, BigGANs [111] have been proposed to generate high-fidelity images at large scales using a parallel architecture and hierarchical latent spaces. These architectures have pushed the boundaries of image generation and have enabled the generation of highly realistic and diverse images.

To add to this, one common issue with GANs is mode collapse, where the generator produces limited varieties of outputs or even the same output regardless of the input. This is especially prevalent when training with high-dimensional data or when the generator is not powerful enough to match the complexity of the data distribution. Another issue is that GANs do not provide a direct way to calculate the likelihood of a sample, which can make them less suitable for some applications such as density estimation. Despite these challenges, GANs continue to be an active area of research and have shown promising results in many applications.

A major challenge with traditional GANs is that the optimization process can be unstable, resulting in mode collapse or convergence to a suboptimal solution. WGAN [112] address this issue by using a different loss function that provides more meaningful gradients and encourages the generator to produce more diverse samples. The pipeline is provided in 7.6. Specifically, instead of using the binary cross-entropy loss as in a traditional GAN, WGAN uses the Wasserstein distance as the objective function. The Wasserstein distance is a measure of the distance between two probability distributions, which has desirable properties for stable GAN training. Algorithm is provided in 7.

Algorithm 7 Wasserstein Generative Adversarial Network (WGAN)

Input: Training dataset X, number of epochs N_{epochs}, critic iterations per generator iteration N_{critic}, learning rate α
Output: Trained generator G and critic D
Initialize generator G and critic D with random weights
for *epoch* $\leftarrow 1$ N_{epochs} **do**
  for *critic_iter* $\leftarrow 1$ N_{critic} **do**
    Sample a mini-batch of real data samples $\{x_1, x_2, \ldots, x_m\}$ from X
    Sample a mini-batch of noise vectors $\{z_1, z_2, \ldots, z_m\}$ from a noise distribution $p_z(z)$
    Compute the Wasserstein distance $\hat{W}_D$:
      $\hat{W}_D \leftarrow \frac{1}{m}\sum_{i=1}^{m} D(x_i) - \frac{1}{m}\sum_{i=1}^{m} D(G(z_i))$
    Update the critic parameters using gradient ascent:
      $\nabla_{\theta_D} \leftarrow \frac{1}{m}\sum_{i=1}^{m} \nabla_{\theta_D} D(x_i) - \nabla_{\theta_D} D(G(z_i))$
      $\theta_D \leftarrow \theta_D + \alpha \cdot \text{clip}(\nabla_{\theta_D}, -c, c)$ (clip weights to enforce 1-Lipschitz constraint)
  end for
  Sample a mini-batch of noise vectors $\{z_1, z_2, \ldots, z_m\}$ from $p_z(z)$
  Update the generator parameters using gradient descent:
    $\nabla_{\theta_G} \leftarrow \frac{1}{m}\sum_{i=1}^{m} \nabla_{\theta_G} D(G(z_i))$
    $\theta_G \leftarrow \theta_G - \alpha \cdot \nabla_{\theta_G}$
end for

To compute the Wasserstein distance, the discriminator in a WGAN is not constrained to output a probability, but rather a real-valued score. This score represents how far the discriminator believes the sample is from the true data distribution. The generator is then trained to minimize the negative discriminator score on the generated samples. By doing so, the generator learns to produce samples that are closer to the true distribution, resulting in more diverse and realistic samples.

To further improve the training stability, WGAN introduces the concept of weight clipping, where the discriminator's weights are clipped to a fixed range to enforce a Lipschitz constraint on the discriminator function. However, this approach can lead to suboptimal performance and convergence issues. To overcome these problems, WGAN-GP [113] introduces a gradient penalty term to enforce the Lipschitz constraint instead of weight clipping, which leads to improved performance and stability.

A Wasserstein Generative Adversarial Network with Gradient Penalty (WGAN-GP) is a variant of the original Generative Adversarial Network (GAN) architecture that utilizes the Wasserstein distance as the loss function for training the generator and the discriminator networks. The main idea behind the WGAN-GP is to improve the stability and quality of the generated samples by introducing a gradient penalty term in the original WGAN loss function.

The WGAN-GP uses the Wasserstein distance, also known as the Earth Mover's distance, to measure the distance between the true data distribution and the generated data distribution. Unlike the original GAN, which uses the Jensen-Shannon divergence, the Wasserstein distance is known to be more stable and less sensitive to the mode collapse problem, which is when the generator only produces samples from a small subset of the target distribution.

The gradient penalty term is used to enforce the Lipschitz constraint on the discriminator function, which ensures that the discriminator remains differentiable and helps to improve the stability of the training process. The gradient penalty term is calculated as the norm of the gradient of the discriminator with respect to the generated samples, and it is added to the original Wasserstein loss to penalize the discriminator for deviating from the Lipschitz constraint.

In practice, WGAN-GP has been shown to generate high-quality samples and improve the stability of the training process compared to the original GAN. However, it is still an active area of research and there are many ongoing efforts to improve the performance of WGAN-GP and other variants of GANs. Just to remind here that in this thesis' Chapter 3, we employed this version in our implementation.

We have used GANs as a denoiser in wireless image transmission. Our architecture was not as complex as [114] which uses current SOTA StyleGAN [115] architecture. Furthermore, the authors have proposed an architecture which utilize GAN for end-to-end wireless image transmission. However, the authors have already demonstrated the power of GANs in wireless image transmission.

7.5 Diffusion Models

Denosing Diffusion Probabilistic Models (DDPM) is a type of generative model that uses a diffusion process to gradually transform noise into data samples. The model has two main stages: forward diffusion and reverse denoising diffusion. In the forward diffusion stage, a predetermined diffusion process is applied to an image, which gradually introduces Gaussian noise until the image becomes pure noise. In the reverse denoising

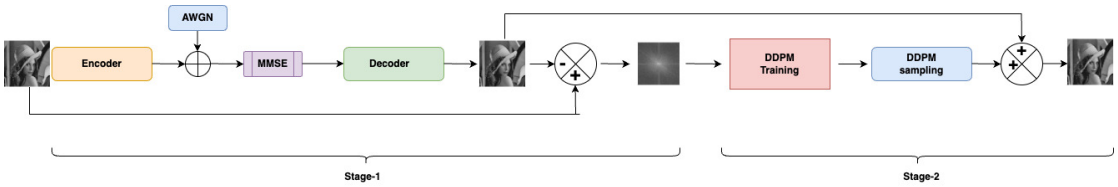


FIGURE 7.7: Block diagram of the proposed method.

diffusion stage, a neural network is trained to perform the opposite process, gradually removing the noise from the image starting with pure noise, resulting in a real image. Both processes are stochastic.

The model is trained using maximum likelihood estimation, where the goal is to maximize the log-likelihood of the data under the model. During training, the model learns the parameters of the diffusion process and the neural network used for the reverse denoising diffusion. The model can then be used to generate new samples by sampling noise from the basic distribution and passing it through the reverse denoising diffusion process. DDPM has been shown to be effective at generating high-quality images and has been used in a variety of applications such as image synthesis and denoising. Score-based methods are a type of generative models that use the likelihood gradient of data to generate novel samples. Instead of learning the probability density functions of data, score-based methods estimate the derivative of the logarithmic likelihood of data by adjusting model parameters, known as the 'score'. This score is then used to perform gradient ascent on the likelihood function to generate new samples. These methods learn the gradient of the distribution, which can be used to produce diverse samples through stochastic sampling in reverse. One popular score-based method is Score Matching, which estimates the density of the data using the score function. Another is Noise Contrastive Estimation (NCE), which uses the score function to learn a generative model. Other examples include Likelihood Ratio GAN (LR-GAN) and Energy-based GAN (EB-GAN). Score-based methods have been shown to be effective in a wide range of applications such as density estimation, image generation, and language modeling.

In this section, we propose a post-processing technique for improving the end-to-end performance of DeepJSCC-MMSE, which involves using a DDPM model as a refiner. The DDPM model can be formulated as a powerful denoiser, allowing it to refine the distorted signals generated during the JSCC process. The diffusion process introduces noise that prevents some values from being transmitted, and the refiner can help recover these lost values to improve the overall quality of the transmitted signal also it refines the output image due to noise introduced during wireless transmission. We trained DeepJSCC-MMSE for various bandwidths and calculated the difference between the output and the original image, which we refer to as the 'compensation' term originating from compression

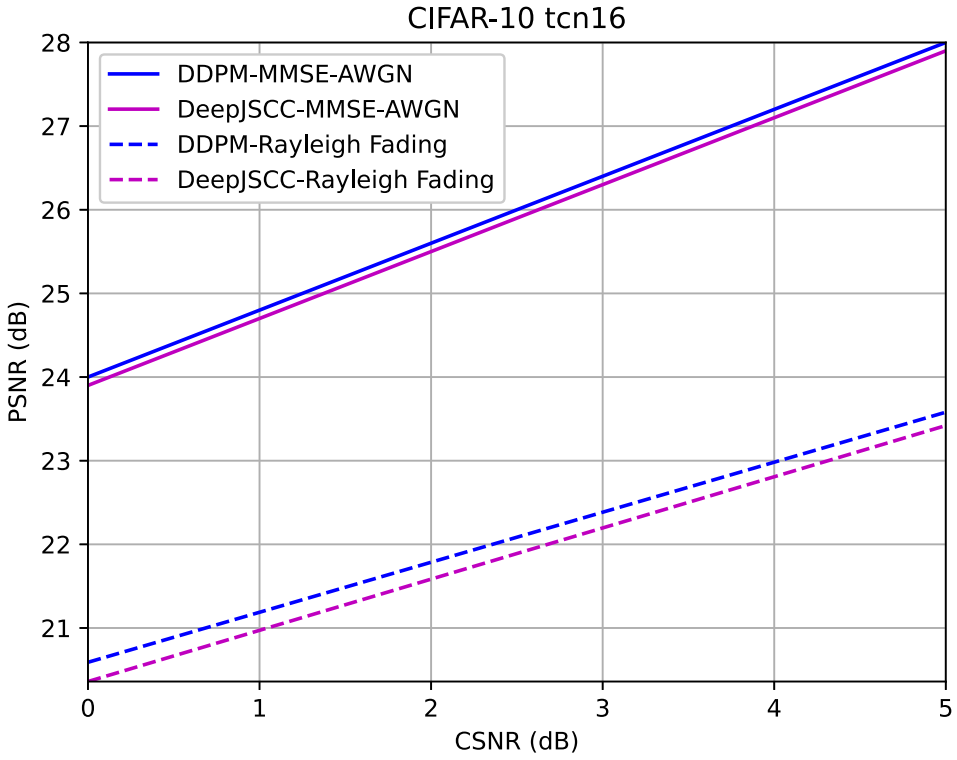


FIGURE 7.8: Results

and transmission. Next, we train a DDPM network to learn the distribution of the compensation image. We then generated a compensation term by sampling from the trained DDPM, which has learned the distribution of this term. Finally, we added this compensation term to the output of DeepJSCC-MMSE, expecting it to compensate for the loss due to bandwidth constraints and channel effects. The results are presented in Fig. 7.8. This result gives us that DDPM based refinement has great potential in wireless image transmission.

7.6 Conclusion

We proposed deep generative models to improve the performance of DeepJSCC. In our simulation results, we showed that deep generative models are one of the good candidates in wireless image transmission. They can be applied as a post-processing block to DeepJSCC-MMSE algorithm. End-to-end architectures and complex filter designs have been proposed; however, they only operates in limited regime and brings into computational complexity. We presented compelling results that demonstrate the superior performance of our proposed method compared to prior approaches. Through our meticulous experimentation and analysis, we have shown that our method not only outperforms

the baseline results but also exhibits significant advancements in terms of accuracy, efficiency, and adaptability. The successful outcome of our study validates the efficacy of our approach and underscores its potential to make a substantial impact in the field. The achieved improvements open up new avenues for more accurate and reliable predictions, enabling enhanced decision-making and problem-solving in various domains. With these remarkable results as a foundation, future research can build upon our findings and explore further advancements, ultimately pushing the boundaries of what is achievable in this area of study.

Chapter 8

Transformers in Deep Joint Source Channel Coding Wireless Image Transmission

The field of image processing architecture has witnessed a significant shift towards the adoption of transformer architectures. While convolutional neural networks (CNNs) have long been the standard and have delivered impressive results, recent research has demonstrated that transformer architectures, particularly Vision Transformers (ViTs), can surpass CNN-backbone architectures even when their model sizes are comparable. Transformers exhibit a remarkable capability to process sequential data effectively, making them well-suited for scenarios where time-varying channels affect the transmission. Moreover, transformers can be trained to extract meaningful information from incomplete and noisy data, which is a common occurrence in wireless transmissions.

In the context of wireless image transmission, transformers have shown promising potential for joint source-channel coding (JSCC). This chapter aims to investigate the performance of ViT-based autoencoder architectures within the domain of JSCC for wireless image transmission. By exploring the potential of ViT architectures in the joint optimization of source coding and channel coding, we aim to shed light on their effectiveness in preserving image quality while efficiently utilizing wireless channels.

This chapter holds significance as it bridges the gap between state-of-the-art transformer architectures and JSCC, exploring their potential synergy. By examining ViT-based autoencoder architectures and their suitability for JSCC, we contribute to the understanding of how transformer models can be leveraged in practical implications for wireless image transmission. Our findings offer valuable insights into the broader applicability

of ViTs and open avenues for further advancements in the field of image processing, specifically in the context of wireless communication.

Overall, this chapter contributes to the ongoing exploration of transformers in image processing and provides a unique perspective on their viability in JSCC. It highlights the potential of ViT-based architectures for efficient and high-quality wireless image transmission.

8.1 Introduction

Transformers, a deep learning architecture, have found applications in wireless transmission to boost the effectiveness of communication systems. Their efficacy has been proven in diverse tasks such as channel coding, recognizing modulation schemes, and detecting signals.

Transformers offer a notable edge in wireless transmission due to their ability to effectively process sequential data, particularly in scenarios where the transmission is affected by time-varying channels. Furthermore, transformers can be trained to extract meaningful information from incomplete and noisy data, a common occurrence in wireless transmissions.

In the field of channel coding, transformers have been integrated with autoencoder architectures and have showcased exceptional performance surpassing conventional coding techniques such as Turbo codes and Low-Density Parity-Check (LDPC) codes. Transformers have also been successfully employed in modulation recognition tasks, enabling the identification of the modulation type employed in a signal, which holds significant advantages for dynamic spectrum management. Furthermore, architectures based on transformers have been utilized in signal detection applications, enhancing the capability to detect signals even in the presence of noise.

In summary, transformers have been extensively validated as a powerful asset in the realm of wireless transmission, showcasing their capacity to significantly improve the efficiency of a wide range of communication systems. Ongoing research is actively exploring the potential of transformers in key areas including wireless channel coding, modulation recognition, signal detection, and channel estimation.

In the literature, vision transformers have been extensively investigated for specific vision tasks such as image classification and segmentation. However, there are only a few studies that examine their ability to generate [116–118] or reconstruct images and their utilization in bi-directional tasks like compression [117, 118] or joint source channel

coding [119]. Additionally, previous works have only examined a specific type of vision transformer that uses a shifted-window approach. Therefore, the optimal design of a vision transformer backbone for transmission tasks, particularly for joint source channel coding, remains an unresolved issue.

Our goal is to address this research gap in the literature by examining the performance of different state-of-the-art vision transformer backbone architectures, [120–125] from the perspective of joint source channel coding. Our aim is also to identify the primary challenges associated with using and modifying ViT backbones. Our ultimate goal is to offer valuable insights into the utilization of transformer-based autoencoder architectures for vision tasks, specifically focusing on coded image transmission and classification in wireless channel. Through our research, we aim to identify and showcase the most effective architecture for these tasks, providing a solid foundation for further advancements in the field.

In this chapter, we want to highlight that we are specifically interested in examining how the choice of the backbone ViT architecture affects the performance of DJSCC. Therefore, we will not be exploring alternative approaches such as entropy coding, and instead will be focusing solely on the conventional DJSCC scenario. It should also be noted that while our main focus is on DJSCC, our findings may also be applicable to other tasks such as compression. One key difference between transformers and CNNs is that transformers lack the inherent biases towards translation invariance and feature hierarchies. However, these aspects can still be learned through adequate training data or pre-training on other large-scale tasks.

8.2 Preliminaries

The Transformer architecture, initially introduced for natural language processing tasks, has been successfully adapted and applied to various computer vision tasks, including image processing [61, 126]. The transformer model consists of two key components: the encoder and the decoder.

The encoder component of the transformer architecture is responsible for capturing the spatial relationships and dependencies among different regions of an image. It consists of a stack of identical layers, each comprising two sub-layers: the multi-head self-attention (MSA) mechanism and the feed-forward neural network (FFN).

The MSA mechanism is a crucial element in the transformer architecture that enables the model to capture global interactions and contextual information. It operates on a set of input embeddings, typically referred to as queries, keys, and values. These embeddings

are linear projections of the input data and allow the model to compute attention weights to capture the importance of different regions within the image.

In the case of image processing, the input image is divided into smaller spatial regions, and each region is represented as an embedding. These embeddings are typically generated from the input image through a set of linear projections. The MSA mechanism takes three types of embeddings:

Query ($\mathbf{Q}$) represents the queries, which are used to find the relevance of different regions in the image. Key ($\mathbf{K}$) represents the keys, which are used to encode the spatial information of the different regions. Value ($\mathbf{V}$) represents the values, which are used to provide the actual content or features of the different regions.

Let's denote the number of regions (or spatial locations) in the image as N, and the dimensionality of the embeddings (query, key, and value) as d_{model}. Then, the dimensions of the matrices $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ are as follows:

$\mathbf{Q} \in \mathbb{R}^{N \times d_{\text{model}}}$: Each row of $\mathbf{Q}$ corresponds to the query embedding of a particular region in the image. It contains d_{model} dimensions, representing the learned information for each region. $\mathbf{K} \in \mathbb{R}^{N \times d_{\text{model}}}$: Each row of $\mathbf{K}$ corresponds to the key embedding of a particular region in the image. It also has d_{model} dimensions, encoding spatial information for each region. $\mathbf{V} \in \mathbb{R}^{N \times d_{\text{model}}}$: Each row of $\mathbf{V}$ corresponds to the value embedding of a particular region in the image. It contains d_{model} dimensions, representing the actual features or content of each region.

The attention mechanism calculates the importance or relevance of each region with respect to the other regions in the image. This is achieved by computing the attention weights, which indicate how much attention should be given to each value embedding ($\mathbf{V}$) based on the compatibility between the query embeddings ($\mathbf{Q}$) and the key embeddings ($\mathbf{K}$).

Mathematically, the attention mechanism is defined as:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_{\text{model}}}}\right)\mathbf{V}, \tag{8.1}$$

where the operation $\mathbf{Q}\mathbf{K}^T$ calculates the pairwise dot products between the query and key embeddings, and the softmax function normalizes these dot products across the spatial dimensions of the image. The $\sqrt{d_{\text{model}}}$ term is a scaling factor to stabilize the gradients during training.

To capture diverse spatial relationships and improve representation learning, the Transformer uses multiple sets of learned query, key, and value projections. These multiple

sets are called 'heads'. Each head learns different patterns and interactions in the input, allowing the model to attend to different parts of the image simultaneously. The outputs of all attention heads are concatenated and linearly transformed to produce the final output of the attention mechanism.

The FFN sub-layer in each Transformer layer provides non-linear transformations to the output of the MSA mechanism. It consists of two linear layers separated by a non-linear activation function, such as the Rectified Linear Unit (ReLU). The FFN sub-layer aims to model complex interactions and learn high-level representations by incorporating non-linear transformations.

The Feed-Forward Neural Network (FFN) sub-layer in each Transformer layer provides non-linear transformations to the output of the Multi-Head Self-Attention (MSA) mechanism. It consists of two linear layers separated by a non-linear activation function, such as the Rectified Linear Unit (ReLU). The FFN sub-layer aims to model complex interactions and learn high-level representations by incorporating non-linear transformations.

Mathematically, the FFN sub-layer can be represented as:

$$\mathrm{FFN}(\mathbf{x}) = \phi(\mathbf{x}\mathbf{W} + \mathbf{b})\tilde{\mathbf{W}} + \tilde{\mathbf{b}}, \tag{8.2}$$

where:

$\mathbf{x}$ is the input to the FFN layer. $\mathbf{W}$ and $\tilde{\mathbf{W}}$ are weight matrices for the first and second linear layers, respectively. $\mathbf{b}$ and $\tilde{\mathbf{b}}$ are bias vectors for the first and second linear layers, respectively. ϕ represents the activation function, such as the Rectified Linear Unit (ReLU). In this equation, the input $\mathbf{x}$ is first linearly transformed by the weight matrix $\mathbf{W}$ and bias vector $\mathbf{b}$, and then passed through the activation function ϕ. The resulting output is then linearly transformed again by the weight matrix $\tilde{\mathbf{W}}$ and bias vector $\tilde{\mathbf{b}}$.

The FFN sub-layer provides the flexibility for the model to capture complex patterns and relationships in the data through the activation function and the two sets of weight matrices and bias vectors. The block diagram of transformer architecture is given in the Fig. 8.1

8.3 Classification

Wireless communication involves a classification task where the goal is to differentiate between signal types transmitted through wireless systems such as Wi-Fi or cellular networks. The task requires training a classifier to analyze distinct features of signals,

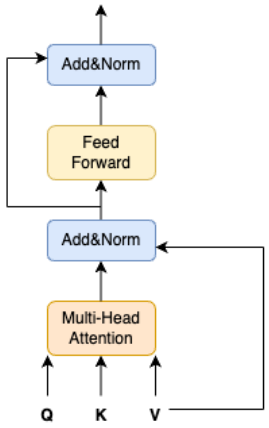


FIGURE 8.1: Block diagram

like voice, video, or data, and assign them to their respective categories using a labeled dataset. The classifier applies the learned features to new signals during the classification process to identify their type and categorize them accordingly.

A case in point can be observed in a cellular network where the classifier is taught to differentiate between different modulation schemes, like CDMA, GSM, and LTE, by analyzing their distinct characteristics such as power spectral density and cyclic prefix.

In a wireless environment, the precision of the classifier can be affected by various factors, such as noise, fading, interference, and the dynamic wireless surroundings. To tackle these issues, there are several approaches that can be utilized, such as selecting important features and reducing dimensions, using ensemble methods, and employing deep learning algorithms.

In wireless sensor networks, a classifier can be utilized to identify different events or activities by analyzing sensor data. For example, it can be used to identify human activity or objects in a surveillance setting. The classifier is trained on a dataset of labeled examples, where each example corresponds to a particular event or activity. Once trained, the classifier can process new sensor data and classify it into the appropriate event or activity category.

Transformers are known to be data hungry models, requiring large amounts of labeled data to achieve optimal performance. To address this, we are employing AugMix [127], a technique that allows for the generation of diverse augmented images, effectively increasing the size and diversity of the training dataset and potentially improving the performance of transformer models.

AugMix is a deep learning technique that enhances model robustness and generalization by leveraging a combination of diverse augmentations in a randomized manner. By selecting three augmentation types at random from a predefined set of operations such as flipping, rotating, shearing, and blurring, AugMix introduces variation into the input image. These augmentations are repeatedly applied to the image, each time with a distinct set of randomly chosen augmentations. To increase the diversity further, AugMix introduces a parameter known as 'strength', which determines the magnitude of the applied augmentations. The strength value is randomly sampled for each augmentation, ensuring that the augmented images differ significantly from the original input. Deep learning models trained using the augmented images generated by AugMix have demonstrated exceptional performance across various benchmark datasets.

Although we have experimented with several transformer architectures for feature extraction, we have chosen to employ CoAtNet [120] as our primary model. This hybrid model that utilizes CNNs for local feature extraction and transformers for global context modeling. This fusion allows CoAtNet to effectively capture both local and global information in images. The paper presents extensive experiments on various benchmark datasets, demonstrating that CoAtNet achieves state-of-the-art performance in image classification, object detection, and semantic segmentation tasks. The findings suggest that the integration of convolution and transformer modules can lead to improved visual representation capabilities. To further enhance the capabilities of transformers and improve their performance, we are leveraging the power of the AugMix data augmentation technique. By combining CoAtNet with AugMix, we aim to unlock the full potential of transformers in our experiments and achieve superior results in classification task.

For the transmission part, the proposed approach in [90] is to use an information bottleneck method to transmit class information to the receiver, by employing an encoder as a CNN-based feature extraction network and a CNN-based channel coding network. To sort the feature values according to importance, the approach applies a γ function to the module. The process starts with feature extraction to reduce source components, sort the values according to importance, or force the network output with importance. In this work, we modify the encoder part with transformer architectures, unlike [90] which uses a CNN-based backbone. Additionally, we utilize data augmentation with AugMix [127] since transformers benefit from larger datasets. Our results show significant improvement over [90], which uses CNN-based architecture for feature extraction, achieving similar performance with fewer symbols (see Table 8.1). This improvement is even more noteworthy when compared to the digital counterpart, which only transmits class information in bits (e.g. 4 bits for 10 classes), using BPSK LDPC 1/2, and requiring a total of 8 symbols. Our approach achieves this level of performance with 7.5 symbols due

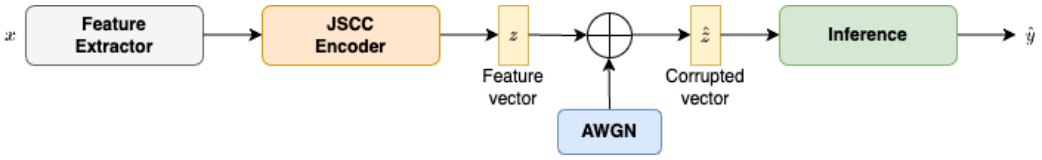


FIGURE 8.2: Pipeline of the Model

CSNR (in dB) :	10dB	15dB	20dB
CoAtNet-based (activated dim|accuracy) :	17|90.21	16|90.87	15|90.94
CNN-based (activated dim|accuracy) :	23|90.62	23|91.37	21|91.44

TABLE 8.1: CoAtNet-based encoder and CNN-based encoder(intermediate dim = 64)

to I/Q transmission. More importantly, we demonstrate the practical application of the findings in [90] by modifying the feature extraction block with a transformer backbone.

8.4 Image Transmission

In this section, we reiterate our objective, which is outlined in 4.6, and present our proposed architecture that we consider to be state-of-the-art. This architecture, called the focal transformer, is depicted in Figure 8.3. It has been shown to achieve better results than the Swin Transformer in various image processing tasks, including classification, object detection, and segmentation.

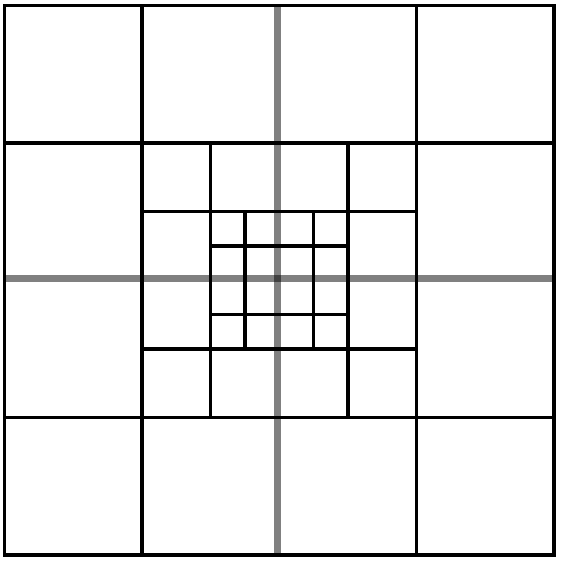


FIGURE 8.3: Focal Attention

We have opted for the focal transformer as our core architecture because it is outperforming the current state-of-the-art transmission architecture, namely Swin Transformer [122]. The key idea behind the Swin Transformer is to handle the spatial resolution of images

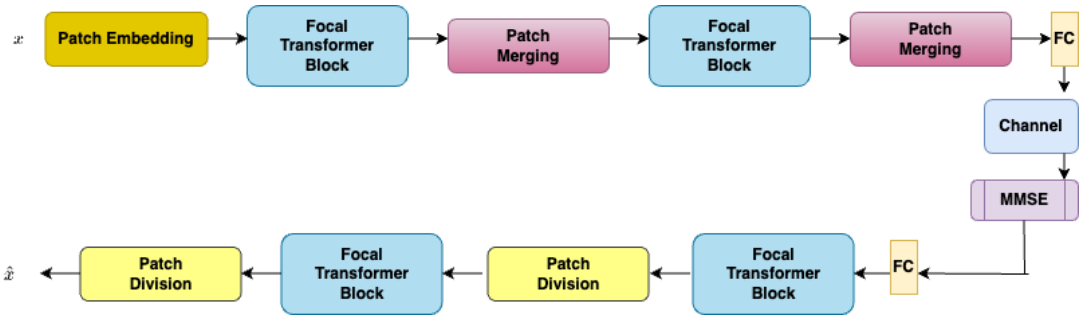


FIGURE 8.4: Proposed Pipeline (CIFAR-10)

in a hierarchical manner, allowing for more efficient computation and memory usage. Instead of processing the entire image at once, the Swin Transformer divides the image into smaller non-overlapping patches, similar to other Vision Transformer (ViT) models. However, unlike ViT, which uses self-attention across all patches, Swin Transformer uses a combination of local and global self-attention mechanisms. The Focal Transformer introduces a novel self-attention mechanism known as 'focal self-attention' which operates in a unique manner. When a query token in the feature map is examined, it pays fine-grained attention to its immediate surroundings, essentially attending to itself. However, as the query token explores more distant regions, it transitions to summarizing tokens, capturing broader visual dependencies in a coarser manner. By employing this approach, the Focal Transformer efficiently covers the entire high-resolution feature map while significantly reducing the number of tokens involved in self-attention computation compared to traditional full self-attention. Consequently, it excels in capturing both short- and long-range visual dependencies while being computationally efficient.

The 'focal self-attention' mechanism enables each token to attend to others in a focal manner, leading to the development of a series of Focal Transformer models. These models leverage a multi-scale architecture, striking a balance between computational cost and performance for high-resolution images. Additionally, the feature map is partitioned into multiple windows, with tokens sharing the same surroundings. This further enhances the efficiency of the focal self-attention mechanism, contributing to the Focal Transformer's impressive performance in various computer vision tasks. Our methodology closely aligns with the framework described in the paper [128]. However, instead of adopting the swin transformer block, we have incorporated the focal transformer block, as illustrated in Figure 8.4. This choice enables us to leverage the focal self-attention mechanism, which effectively combines localized fine-grained interactions with broader global interactions, enhancing the overall performance of our approach. In [129], a modified version is proposed that replaces self-attention with focal modulation, resulting in the replacement of focal nets with focal modulation. Focal modulation differs from the self-attention mechanism in that it aggregates the input context in the first step, whereas with self-attention,

aggregation occurs after the query interacts with the input context. While the superiority of focal modulation is demonstrated, for our wireless image transmission problem, we only consider the focal transformer. However, if high-resolution images need to be transmitted, the proposed architecture would require modification by incorporating four blocks instead of two.

It is worth mentioning that we made modifications to the original scheme proposed in [128] by extracting the modulation block and incorporating MMSE estimation, which was also utilized in Chapter 4. We train our models using the Pytorch library [74] on the CIFAR-10 dataset [75] which consists of 60000 32×32 images. We split the dataset into 45000 images for training, 5000 images for validation, and 10000 images for testing. We use a batch size of 128 and train for 500 epochs. We train and test the models with CSNR range $[0, 15]$ dB. We use the Adam optimizer [76] with learning rate 10^{-4}. The learning rate is scheduled according to the validation set, where the learning rate is decreased by a factor of 0.8 if the validation loss does not improve for four epochs consecutively. We use an average power $\bar{P} = 1$ and change the channel noise power σ^2 accordingly to satisfy CSNR. During training, we sample a CSNR value uniformly from the set $[0 - 15]$dB per batch of training data in order to expose the networks to a wide range of channel conditions. We are presenting the results.

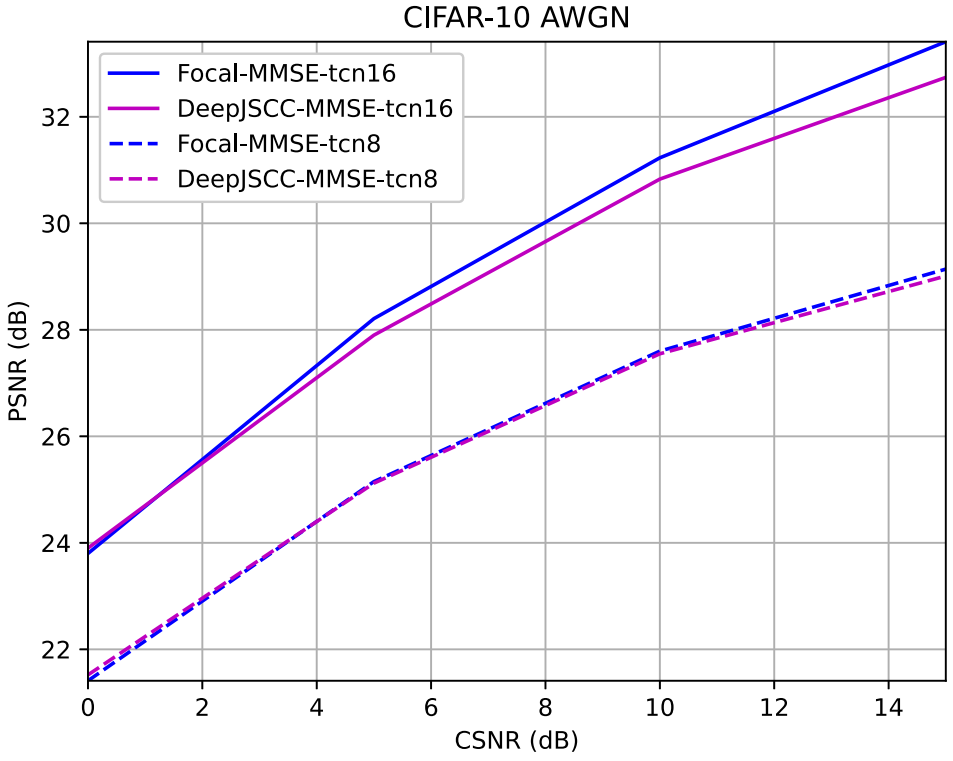


FIGURE 8.5: Comparison

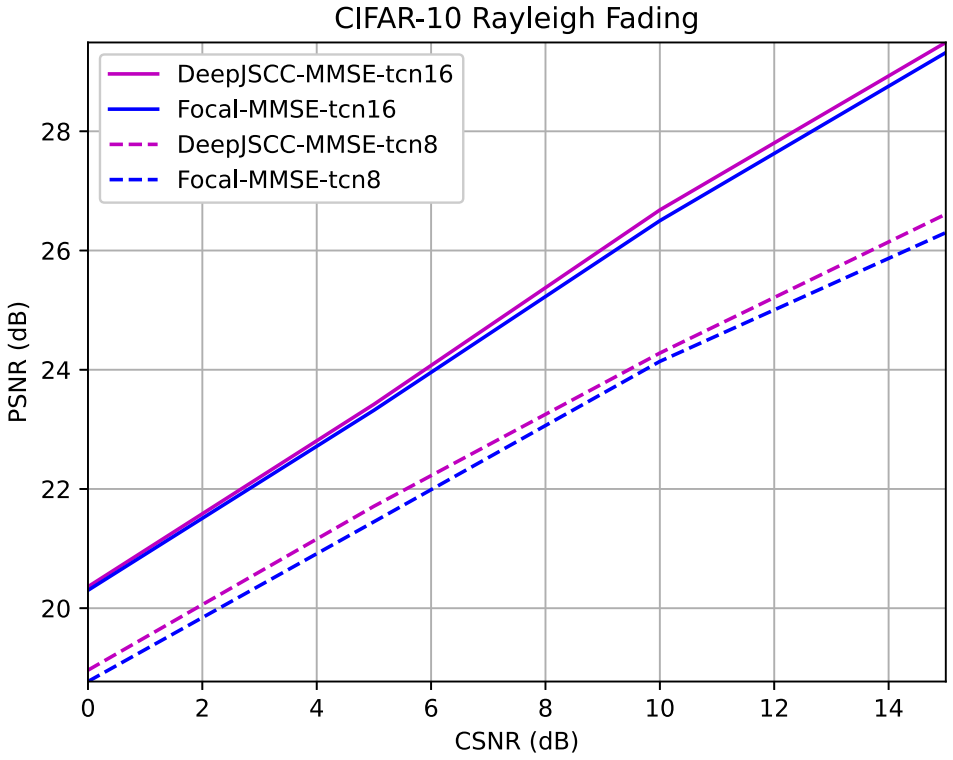


FIGURE 8.6: Comparison

In general, the performance of the focal transformer and swin transformer based architectures is quite similar, particularly in low-CSNR conditions. However, as the CSNR improves, the focal transformer based architecture outperforms the swin transformer based architecture. We hypothesize that in low-CSNR scenarios, the transformer architecture allocates more symbols to channel coding, which limits the effectiveness of the focal transformer. However, in high-CSNR conditions, the focal transformer operates more effectively and results in improved PSNR. We included the CNN-based architecture, DeepJSCC-MMSE to demonstrate the effectiveness of transformer architectures. If we were to extend the study to high-resolution images, we believe that the improvement achieved by the focal transformer architecture would be even more significant. We conclude here that this scheme is the state-of-the-art.

8.5 Conclusion

In this chapter, we have explored the potential of Transformer-based architectures in the context of wireless communication, with a specific focus on image transmission and

classification tasks. Transformers have emerged as a powerful deep learning architecture, showcasing remarkable capabilities in various domains, including natural language processing, computer vision, and now wireless communication.

We began by providing an overview of Transformers and their advantages in processing sequential data, which makes them particularly suitable for wireless communication scenarios affected by time-varying channels and noisy environments. Transformers have demonstrated superior performance in tasks such as channel coding, modulation recognition, and signal detection, surpassing traditional techniques and offering new possibilities for improving the efficiency and reliability of communication systems.

Moreover, we highlighted the need for investigating Transformer-based architectures in the domain of image transmission. While Transformers have been extensively studied in vision tasks such as image classification and segmentation, their potential in generating and reconstructing images for transmission purposes remains largely unexplored. We recognized the research gap and outlined our goal to examine different state-of-the-art vision Transformer backbone architectures from the perspective of joint source channel coding, aiming to identify the most effective architecture and address the challenges associated with modifying ViT backbones.

Through our exploration, we have demonstrated the feasibility and benefits of incorporating Transformers in image transmission. Transformers can effectively process sequential image data, capture long-range dependencies, and adapt to different wireless channel conditions, leading to improved reconstruction quality, robustness against channel impairments, and enhanced transmission efficiency.

Furthermore, we introduced the concept of Transformer-based image classification in wireless communication, where the goal is to differentiate between different signal types transmitted over wireless systems. We discussed the role of classifiers in analyzing distinct features of signals and assigning them to their respective categories. By leveraging Transformers in the classification process, we can benefit from their ability to learn meaningful representations from image data and improve the accuracy of signal classification.

In conclusion, Transformers offer a promising avenue for advancing wireless communication systems, particularly in image transmission and classification tasks. The research conducted in this area has provided valuable insights into the capabilities of Transformer-based architectures and their potential for enhancing the efficiency, quality, and robustness of wireless communication. Further research and development are needed to explore new architectural designs, optimize model performance, and adapt Transformers to specific wireless communication scenarios.

We anticipate that future advancements in Transformer-based architectures, coupled with advancements in wireless communication technologies, will lead to transformative applications and improvements in areas such as image transmission, joint source channel coding, and beyond. By leveraging the power of Transformers, we can unlock new possibilities and open doors to more efficient and reliable wireless communication systems.

Chapter 9

Conclusion

In this concluding chapter, we summarize the key findings of our thesis and emphasize the practicality and effectiveness of the proposed algorithms. Furthermore, we provide insights and suggestions for future research and potential applications in related fields.

What we have done? Chapter 3 of this thesis introduced a novel analog transmission method called ZZ-Cast for transmitting images and videos. The chapter provides a comprehensive evaluation of ZZ-Cast's performance against other analog and digital transmission methods under low bandwidth and low channel signal to noise ratio (SNR) conditions. Our results demonstrate that ZZ-Cast outperforms the well-known Soft-Cast benchmark and the digital method in these scenarios. Additionally, we presented a metadata-free version of ZZ-Cast, which takes into account the costs of transmitting metadata in evaluating the uncoded/analog transmission options. This provides a fair comparison between digital and analog transmission methods, which is crucial for selecting the most efficient transmission method for a given scenario. The findings presented in this chapter highlight the potential of analog transmission methods and the importance of considering metadata transmission costs when evaluating transmission options. This research opens up new possibilities for efficient image and video transmission in low-resource scenarios. In Chapter 4, we explore the potential of training DeepJSCC independent of the signal-to-noise ratio (SNR). With the emergence of machine learning, joint source-channel coding (JSCC) schemes can be effectively created. DeepJSCC, a JSCC scheme based on deep learning, has been shown to outperform existing designs that separate source and channel coding for wireless image transmission. However, current methods for DeepJSCC either require multiple models to be trained for specific channel conditions, leading to storage issues, or they necessitate including additional parameters in the model to adapt to new channel conditions, increasing the system's complexity. In this study, we present DeepJSCC-MMSE, an optimized JSCC scheme

for wireless image transmission over fading channels. It comprises a single DeepJSCC encoder/decoder pair that is trained with a channel equalization module at the receiver, which can adapt to any channel condition without requiring additional model parameters. We demonstrate that DeepJSCC-MMSE can outperform parameterized equalization techniques without retraining the encoder and decoder for new channel conditions, laying the groundwork for generalized DeepJSCC schemes based on deep learning. In Chapter 5, we proposed variable bandwidth ratio model for DeepJSCC. In the context of image compression for transmission, conventional methods such as DeepJSCC often employ a fixed compression ratio. Nevertheless, this approach presents challenges since the receiver may only require images of a specific quality. Consequently, all images are transmitted using the same compression ratio, regardless of their complexity. This inefficiently utilizes resources. To address this concern and mitigate gradient-related issues, we propose a novel approach that facilitates the use of multiple compression rates. Our method is straightforward and adaptive, eliminating the need for intricate algorithms. Through our experiments, we have demonstrated that our approach surpasses the performance of the baseline method, which utilizes MMSE estimation at the receiver while employing the same architecture. In the Chapter 6, we conducted research on the application of Deep Joint Source-Channel Coding (DeepJSCC) for distributed image transmission over a noisy multiple access channel (MAC). In the context of finite block length regimes, the conventional approach of separating source and channel coding is not optimal. To overcome this limitation, we proposed a novel joint image compression and transmission scheme that employs Non-Orthogonal Multiple Access (NOMA) to send compressed image representations non-orthogonally. While NOMA has the potential to achieve the capacity region, the development of a non-orthogonal Joint Source-Channel Coding (JSCC) scheme for practical systems has not been extensively explored in previous research. Our experimental results demonstrate that the proposed scheme enhances the reconstructed image quality compared to existing DeepJSCC methods, particularly under low bandwidth ratios. In Chapter 7, we explored the potential of utilizing deep generative models, including variational autoencoders, normalizing flows, generative adversarial networks (GANs), and diffusion models, for Deep Joint Source-Channel Coding (DeepJSCC). While these models have demonstrated exceptional performance in various computer vision tasks, such as object detection, image compression, segmentation, and classification, their potential in transmission is yet to be fully explored. Despite their different learning strategies for modeling dataset distributions, each model has its own limitations. For example, variational autoencoders generate images that interpolate the data distribution but use a surrogate loss function. On the other hand, GANs are effective but have an adversarial training nature. Flow-based methods require specialized architectures to create reversible transforms. Diffusion models are learned using a fixed

procedure and have high dimensional latent variables, but offer exact log-likelihood computation and inverse problem solving without re-training models. Lastly, score-based models are connected to normalizing flow models and allow for exact likelihood computation and representation learning. Our exploration in this chapter aimed to identify the most suitable deep generative model for DeepJSCC in noisy transmission environments. We demonstrated potential gains in wireless image transmission via deep generative models. In Chapter 9, we conducted an investigation into the transformer architecture, which has gained popularity for its superior performance in various image tasks, surpassing the state-of-the-art convolutional neural networks. Among these transformer architectures, the vision transformer (ViT) has demonstrated its ability to capture local features effectively, leading to a significant improvement in classification and segmentation tasks. This raises a crucial question: Can ViT's success in utilizing local features be extended to wireless transmission, especially in the context of joint source-channel coding (JSCC) of images over wireless channels? Our research explored this question and presented state-of-the-art transformer architectures have great potential in wireless image transmission with JSCC.

Research Challenges In this part, we will delve into the implementation details of our proposed architectures. While all the results presented in this study are 'in silico', it is essential to consider hardware adaptability for practical applications. In this regard, the authors have previously proposed [130] to convert continuous outputs into digitized form, which can be readily implemented in conventional digital transmission hardware. Furthermore, the use of GPUs in training neural networks has been proven to be highly beneficial, and advancements in analog circuit designs are rapidly progressing. Authors have already demonstrated that [131] DeepJSCC has the potential to be applied in security operations. For instance, the 'learning with errors' algorithm has been proposed as a network security algorithm and there have been numerous papers on the application of security algorithms in neural networks [132]. Another important consideration for neural networks is memory occupation. With a large number of parameters, memory requirements can be a significant issue. To address this, sparse neural networks [133] and lottery ticket algorithms [134] have been proposed to decrease memory usage. These techniques are important for making DeepJSCC more practical for real-world applications where memory and processing constraints can be a limiting factor.

Future Directions The significance of this thesis can also be observed through the literature. Various studies have been conducted to merge traditional methods such as Discrete Cosine Transform, Wavelet Transform, and frequency-domain learning [135]. For instance, Wave-ViT [136] combines invertible down-sampling using both wavelet transforms and self-attention learning. Moreover, a recent paper [137] models the inverse-DCT matrix as a transformer and uses zigzag-based hard encoding to compress the image,

which was proposed in Chapter 3. These examples highlight the relevance of this thesis in the field and its potential impact on future research.

Bibliography

[1] C. E. Shannon, "A mathematical theory of communication," *The Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948.

[2] H. Zhu, M. Li, I. Chlamtac, and B. Prabhakaran, "A survey of quality of service in ieee 802.11 networks," *IEEE wireless communications*, vol. 11, no. 4, pp. 6–14, 2004.

[3] H. I. Zawia, R. Hassan, and D. P. Dahnil, "A survey of medium access mechanisms for providing robust audio video streaming in ieee 802.11 aa standard," *IEEE Access*, vol. 6, pp. 27 690–27 705, 2018.

[4] M. Graf, C. Timmerer, and C. Mueller, "Towards bandwidth efficient adaptive streaming of omnidirectional video over http: Design, implementation, and evaluation," in *Proceedings of the 8th ACM on Multimedia Systems Conference*, 2017, pp. 261–271.

[5] D. Wu, Y. T. Hou, and Y.-Q. Zhang, "Scalable video coding and transport over broadband wireless networks," *Proceedings of the IEEE*, vol. 89, no. 1, pp. 6–20, 2001.

[6] H. Ha and C. Yim, "Layer-weighted unequal error protection for scalable video coding extension of h. 264/avc," *IEEE Transactions on Consumer Electronics*, vol. 54, no. 2, pp. 736–744, 2008.

[7] J. Nightingale, Q. Wang, and C. Grecos, "Optimised transmission of h. 264 scalable video streams over multiple paths in mobile networks," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 4, pp. 2161–2169, 2010.

[8] C. W. Yap, K. N. Ngan, and R. Liyanapathirana, "A combined source-channel video coding scheme for mobile channels," *Signal Processing: Image Communication*, vol. 14, no. 6-8, pp. 559–574, 1999.

[9] P. A. Floor, T. A. Ramstad, and N. Wernersson, "Power constrained channel optimized vector quantizers used for bandwidth expansion," in *2007 4th International Symposium on Wireless Communication Systems*. IEEE, 2007, pp. 667–671.

[10] T. Goblick, "Theoretical limitations on the transmission of data from analog sources," *IEEE Transactions on Information Theory*, vol. 11, no. 4, pp. 558–567, 1965.

[11] S. Jakubczak and D. Katabi, "Softcast: one-size-fits-all wireless video," in *ACM SIGCOMM Computer Communication Review*, vol. 40, no. 4. ACM, 2010, pp. 449–450.

[12] H. Coward, "Joint source-channel coding: Development of methods and utilization in image communications," Ph.D. dissertation, Norwegian University of Science and Technology, 2001.

[13] E. Bourtsoulatze, 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.

[14] F. Hekland, P. A. Floor, and T. A. Ramstad, "Shannon-kotel-nikov mappings in joint source-channel coding," *IEEE Transactions on Communications*, vol. 57, no. 1, pp. 94–105, 2009.

[15] M. Cagnazzo and M. Kieffer, "Shannon-kotelnikov mappings for softcast-based joint source-channel video coding," in *2015 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2015, pp. 1085–1089.

[16] F. Liang, C. Luo, R. Xiong, W. Zeng, and F. Wu, "Hybrid digital–analog video delivery with shannon–kotel'nikov mapping," *IEEE Transactions on Multimedia*, vol. 20, no. 8, pp. 2138–2152, 2017.

[17] Y. Hu, J. Garcia-Frias, and M. Lamarca, "Analog joint source-channel coding using non-linear curves and mmse decoding," *IEEE Transactions on Communications*, vol. 59, no. 11, pp. 3016–3026, 2011.

[18] Y. M. Saidutta, A. Abdi, and F. Fekri, "M to 1 joint source-channel coding of gaussian sources via dichotomy of the input space based on deep learning," in *2019 Data Compression Conference (DCC)*. IEEE, 2019, pp. 488–497.

[19] ——, "Joint source-channel coding for gaussian sources over awgn channels using variational autoencoders," in *2019 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2019, pp. 1327–1331.

[20] ——, "Joint source-channel coding of gaussian sources over awgn channels via manifold variational autoencoders," in *2019 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2019, pp. 514–520.

[21] ——, "Joint source-channel coding over additive noise analog channels using mixture of variational autoencoders," *IEEE Journal on Selected Areas in Communications*, 2021.

[22] Z. Xuan and K. Narayanan, "Analog joint source-channel coding for gaussian sources over awgn channels with deep learning," in *2020 International Conference on Signal Processing and Communications (SPCOM)*. IEEE, 2020, pp. 1–5.

[23] K. Choi, K. Tatwawadi, A. Grover, T. Weissman, and S. Ermon, "Neural joint source-channel coding," in *International Conference on Machine Learning*. PMLR, 2019, pp. 1182–1192.

[24] D. B. Kurka and D. Gündüz, "Deepjscc-f: Deep joint source-channel coding of images with feedback," *IEEE Journal on Selected Areas in Information Theory*, vol. 1, no. 1, pp. 178–193, 2020.

[25] ——, "Successive refinement of images with deep joint source-channel coding," in *2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. IEEE, 2019, pp. 1–5.

[26] M. Ding, J. Li, M. Ma, and X. Fan, "Snr-adaptive deep joint source-channel coding for wireless image transmission," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 1555–1559.

[27] A. H. Estiri, M. R. Sabramooz, A. Banaei, A. H. Dehghan, B. Jamialahmadi, and M. J. Siavoshani, "A variational auto-encoder approach for image transmission in noisy channel," in *2020 10th International Symposium onTelecommunications (IST)*. IEEE, 2020, pp. 227–233.

[28] J. Xu, B. Ai, W. Chen, A. Yang, P. Sun, and M. Rodrigues, "Wireless image transmission using deep source channel coding with attention modules," *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[29] M. Yang, C. Bian, and H.-S. Kim, "Deep joint source channel coding for wirelessimage transmission with ofdm," *arXiv preprint arXiv:2101.03909*, 2021.

[30] M. Borgerding, P. Schniter, and S. Rangan, "Amp-inspired deep networks for sparse linear inverse problems," *IEEE Transactions on Signal Processing*, vol. 65, no. 16, pp. 4293–4308, 2017.

[31] O. Musa, P. Jung, and G. Caire, "Plug-and-play learned gaussian-mixture approximate message passing," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 4855–4859.

[32] A. Bora, A. Jalal, E. Price, and A. G. Dimakis, "Compressed sensing using generative models," in *International Conference on Machine Learning*. PMLR, 2017, pp. 537–546.

[33] K. Wu and C. Zhang, "Deep generative adversarial networks for the sparse signal denoising," in *2018 24th International Conference on Pattern Recognition (ICPR)*. IEEE, 2018, pp. 1127–1132.

[34] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Transactions on information Theory*, vol. 19, no. 4, pp. 471–480, 1973.

[35] J. Whang, A. Acharya, H. Kim, and A. G. Dimakis, "Neural distributed source coding," *arXiv preprint arXiv:2106.02797*, 2021.

[36] J. Whang, E. Lindgren, and A. Dimakis, "Composing normalizing flows for inverse problems," in *International Conference on Machine Learning*. PMLR, 2021, pp. 11 158–11 169.

[37] Q. Lei, A. Jalal, I. S. Dhillon, and A. G. Dimakis, "Inverting deep generative models, one layer at a time," *arXiv preprint arXiv:1906.07437*, 2019.

[38] T. O'shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 563–575, 2017.

[39] F. A. Aoudia and J. Hoydis, "Model-free training of end-to-end communication systems," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 11, pp. 2503–2516, 2019.

[40] S. Dörner, M. Henninger, S. Cammerer, and S. ten Brink, "Wgan-based autoencoder training over-the-air," in *2020 IEEE 21st International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. IEEE, 2020, pp. 1–5.

[41] A. Felix, S. Cammerer, S. Dörner, J. Hoydis, and S. Ten Brink, "Ofdm-autoencoder for end-to-end learning of communications systems," in *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. IEEE, 2018, pp. 1–5.

[42] F. A. Aoudia and J. Hoydis, "End-to-end learning of communications systems without a channel model," in *2018 52nd Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2018, pp. 298–303.

[43] S. Dörner, S. Cammerer, J. Hoydis, and S. Ten Brink, "Deep learning based communication over the air," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 132–143, 2017.

[44] M. Goutay, F. A. Aoudia, and J. Hoydis, "Deep reinforcement learning autoencoder with noisy feedback," in *2019 International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOPT)*. IEEE, 2019, pp. 1–6.

[45] T. J. O'Shea, L. Pemula, D. Batra, and T. C. Clancy, "Radio transformer networks: Attention models for learning to synchronize in wireless systems," in *2016 50th Asilomar Conference on Signals, Systems and Computers*. IEEE, 2016, pp. 662–666.

[46] T. J. O'Shea, K. Karra, and T. C. Clancy, "Learning to communicate: Channel auto-encoders, domain specific regularizers, and attention," in *2016 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*. IEEE, 2016, pp. 223–228.

[47] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in ofdm systems," *IEEE Wireless Communications Letters*, vol. 7, no. 1, pp. 114–117, 2017.

[48] H. Ye, G. Y. Li, B.-H. F. Juang, and K. Sivanesan, "Channel agnostic end-to-end learning based communication systems with conditional gan," in *2018 IEEE Globecom Workshops (GC Wkshps)*. IEEE, 2018, pp. 1–5.

[49] H. Ye, G. Y. Li, and B.-H. F. Juang, "Deep learning based end-to-end wireless communication systems without pilots," *IEEE Transactions on Cognitive Communications and Networking*, 2021.

[50] H. He, C.-K. Wen, S. Jin, and G. Y. Li, "Model-driven deep learning for mimo detection," *IEEE Transactions on Signal Processing*, vol. 68, pp. 1702–1715, 2020.

[51] M. Soltani, V. Pourahmadi, A. Mirzaei, and H. Sheikhzadeh, "Deep learning-based channel estimation," *IEEE Communications Letters*, vol. 23, no. 4, pp. 652–655, 2019.

[52] M. Soltani, V. Pourahmadi, and H. Sheikhzadeh, "Pilot pattern design for deep learning-based channel estimation in ofdm systems," *IEEE Wireless Communications Letters*, vol. 9, no. 12, pp. 2173–2176, 2020.

[53] G. Toderici, D. Vincent, N. Johnston, S. Jin Hwang, D. Minnen, J. Shor, and M. Covell, "Full resolution image compression with recurrent neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5306–5314.

[54] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9446–9454.

[55] R. Heckel and P. Hand, "Deep decoder: Concise image representations from untrained non-convolutional networks," *arXiv preprint arXiv:1810.03982*, 2018.

[56] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[57] D. Liu, H. Ma, Z. Xiong, and F. Wu, "Cnn-based dct-like transform for image compression," in *International Conference on Multimedia Modeling*. Springer, 2018, pp. 61–72.

[58] R. Xiong, F. Wu, J. Xu, X. Fan, C. Luo, and W. Gao, "Analysis of decorrelation transform gain for uncoded wireless image and video communication," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1820–1833, 2016.

[59] R. Xiong, J. Zhang, F. Wu, J. Xu, and W. Gao, "Power distortion optimization for uncoded linear transformed transmission of images and videos," *IEEE Transactions on Image Processing*, vol. 26, no. 1, pp. 222–236, 2017.

[60] H. Ma, D. Liu, R. Xiong, and F. Wu, "iwave: Cnn-based wavelet-like transform for image compression," *IEEE Transactions on Multimedia*, vol. 22, no. 7, pp. 1667–1679, 2019.

[61] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[62] E. Balevi, A. Doshi, A. Jalal, A. Dimakis, and J. G. Andrews, "High dimensional channel estimation using deep generative networks," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 1, pp. 18–30, 2020.

[63] K.-H. Lee and D. Petersen, "Optimal linear coding for vector channels," *IEEE Transactions on Communications*, vol. 24, no. 12, pp. 1283–1290, 1976.

[64] X. Fan, R. Xiong, F. Wu, and D. Zhao, "Wavecast: Wavelet based wireless video broadcast using lossy transmission," in *2012 Visual Communications and Image Processing*. IEEE, 2012, pp. 1–6.

[65] A. Hagag, X. Fan, and F. E. A. El-Samie, "Hypercast: hyperspectral satellite image broadcasting with band ordering optimization," *Journal of Visual Communication and Image Representation*, vol. 42, pp. 14–27, 2017.

[66] H. Liu, R. Xiong, X. Fan, D. Zhao, Y. Zhang, and W. Gao, "Cg-cast: scalable wireless image softcast using compressive gradient," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 6, pp. 1832–1843, 2018.

[67] R. Xiong, H. Liu, S. Ma, X. Fan, F. Wu, and W. Gao, "G-cast: Gradient based image softcast for perception-friendly wireless visual communication," in *2014 Data Compression Conference*. IEEE, 2014, pp. 133–142.

[68] T.-Y. Tung and D. Gündüz, "Sparsecast: Hybrid digital-analog wireless image transmission exploiting frequency-domain sparsity," *IEEE Communications Letters*, vol. 22, no. 12, pp. 2451–2454, 2018.

[69] Z. Zhang, D. Liu, X. Ma, and X. Wang, "Ecast: An enhanced video transmission design for wireless multicast systems over fading channels," *IEEE Systems Journal*, vol. 11, no. 4, pp. 2566–2577, 2015.

[70] T. Zhang and S. Mao, "Metadata reduction for soft video delivery," *IEEE Networking Letters*, vol. 1, no. 2, pp. 84–88, 2019.

[71] S. Zong, S. Gao, G. Tu, C. Zhang, and D. Chen, "A metadata-free pure soft broadcast scheme for image and video transmission," *Signal Processing: Image Communication*, vol. 78, pp. 359–367, 2019.

[72] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proceedings of the National Academy of Sciences*, vol. 106, no. 45, pp. 18 914–18 919, 2009.

[73] N. Shlezinger, J. Whang, Y. C. Eldar, and A. G. Dimakis, "Model-based deep learning," *arXiv preprint arXiv:2012.08405*, 2020.

[74] A. Paszke *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[75] A. Krizhevsky, G. Hinton *et al.*, "Learning multiple layers of features from tiny images," 2009.

[76] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[77] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: an overview," *IEEE Transactions on Consumer Electronics*, vol. 46, no. 4, pp. 1103–1127, Nov. 2000.

[78] F. Bellard, *Better Portable Graphics*, 2014 (accessed March 13, 2020), https://bellard.org/bpg/.

[79] V. Kostina and S. Verdú, "Lossy joint source-channel coding in the finite blocklength regime," *IEEE Transactions on Information Theory*, vol. 59, no. 5, pp. 2545–2575, May 2013.

[80] D. B. Kurka and D. Gündüz, "Bandwidth-agile image transmission with deep joint source-channel coding," *IEEE Transactions on Wireless Communications*, vol. 20, no. 12, pp. 8081–8095, 2021.

[81] T.-Y. Tung, D. B. Kurka, M. Jankowski, and D. Gündüz, "Deepjscc-q: Channel input constrained deep joint source-channel coding," in *IEEE International Conference on Communications*, 2022, pp. 3880–3885.

[82] H. Wu, Y. Shao, K. Mikolajczyk, and D. Gündüz, "Channel-adaptive wireless image transmission with OFDM," *IEEE Wireless Communications Letters*, 2022.

[83] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 1874–1883.

[84] J. Ballé, V. Laparra, and E. P. Simoncelli, "Density modeling of images using a generalized normalization transformation," *arXiv preprint arXiv:1511.06281*, 2015.

[85] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Learned image compression with discretized gaussian mixture likelihoods and attention modules," in *2020 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020, pp. 7936–7945.

[86] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-Local Neural Networks," in *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2018, pp. 7794–7803.

[87] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[88] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "Infogan: Interpretable representation learning by information maximizing generative adversarial nets," *Advances in neural information processing systems*, vol. 29, 2016.

[89] M. Yang and H.-S. Kim, "Deep joint source-channel coding for wireless image transmission with adaptive rate control," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 5193–5197.

[90] J. Shao, Y. Mao, and J. Zhang, "Learning task-oriented communication for edge inference: An information bottleneck approach," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 197–211, 2021.

[91] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *NIPS*, 2017.

[92] S. F. Yilmaz, C. Karamanli, and D. Gunduz, "Distributed deep joint source-channel coding over a multiple access channel," *arXiv preprint arXiv:2211.09920*, 2022.

[93] T. M. Cover, *Elements of information theory*. John Wiley & Sons, 1999.

[94] A. J. Grant, B. Rimoldi, R. L. Urbanke, and P. A. Whiting, "Rate-splitting multiple access for discrete memoryless channels," *IEEE Transactions on Information Theory*, vol. 47, no. 3, pp. 873–890, 2001.

[95] N. Shlezinger, R. Fu, and Y. C. Eldar, "Deepsic: Deep soft interference cancellation for multiuser MIMO detection," *IEEE Transactions on Wireless Communications*, vol. 20, no. 2, pp. 1349–1362, 2020.

[96] I. Sim, Y. G. Sun, D. Lee, S. H. Kim, J. Lee, J.-H. Kim, Y. Shin, and J. Y. Kim, "Deep learning based successive interference cancellation scheme in nonorthogonal multiple access downlink network," *Energies*, vol. 13, no. 23, p. 6237, 2020.

[97] T. Van Luong, N. Shlezinger, C. Xu, T. M. Hoang, Y. C. Eldar, and L. Hanzo, "Deep learning based successive interference cancellation for the non-orthogonal downlink," *IEEE Transactions on Vehicular Technology*, 2022.

[98] N. Mital, E. Özyılkan, A. Garjani, and D. Gündüz, "Neural distributed image compression using common information," in *2022 Data Compression Conference (DCC)*. IEEE, 2022, pp. 182–191.

[99] R. Caruana, "Multitask learning," *Machine Learning*, vol. 28, no. 1, pp. 41–75, 1997.

[100] Y. Zhang and Q. Yang, "A survey on multi-task learning," *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[101] G. Koch, R. Zemel, R. Salakhutdinov *et al.*, "Siamese neural networks for one-shot image recognition," in *ICML Deep Learning Workshop*, vol. 2. Lille, 2015, p. 0.

[102] E. Hoffer and N. Ailon, "Deep metric learning using triplet network," in *International Workshop on Similarity-Based Pattern Recognition.* Springer, 2015, pp. 84–92.

[103] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proceedings of the 26th annual international conference on machine learning*, 2009, pp. 41–48.

[104] E. Ozfatura, Y. Shao, A. Perotti, B. Popovic, and D. Gunduz, "All you need is feedback: Communication with block attention feedback codes," *arXiv preprint arXiv:2206.09457*, 2022.

[105] W. F. Lo, N. Mital, H. Wu, and D. Gündüz, "Collaborative semantic communication at the edge," *arXiv preprint arXiv:2301.03996*, 2023.

[106] Z. Xuan and K. Narayanan, "Low-delay analog distributed joint source-channel coding using sirens," in *2021 29th European Signal Processing Conference (EUSIPCO).* IEEE, 2021, pp. 1601–1605.

[107] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real nvp," *arXiv preprint arXiv:1605.08803*, 2016.

[108] D. Rezende and S. Mohamed, "Variational inference with normalizing flows," in *International conference on machine learning.* PMLR, 2015, pp. 1530–1538.

[109] T. Han, J. Tang, Q. Yang, Y. Duan, Z. Zhang, and Z. Shi, "Generative model based highly efficient semantic communication approach for image transmission," *arXiv preprint arXiv:2211.10287*, 2022.

[110] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.

[111] A. Brock, J. Donahue, and K. Simonyan, "Large scale gan training for high fidelity natural image synthesis," *arXiv preprint arXiv:1809.11096*, 2018.

[112] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International conference on machine learning.* PMLR, 2017, pp. 214–223.

[113] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," *Advances in neural information processing systems*, vol. 30, 2017.

[114] E. Erdemir, T.-Y. Tung, P. L. Dragotti, and D. Gunduz, "Generative joint source-channel coding for semantic image transmission," *arXiv preprint arXiv:2211.13772*, 2022.

[115] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4401–4410.

[116] Y. Jiang, S. Chang, and Z. Wang, "Transgan: Two pure transformers can make one strong gan, and that can scale up," in *NeurIPS*, 2021.

[117] R. Zou, C. Song, and Z. Zhang, "The devil is in the details: Window-based attention for image compression," in *CVPR*, 2022.

[118] Y. Qian, M. Lin, X. Sun, Z. Tan, and R. Jin, "Entroformer: A transformer-based entropy model for learned image compression," in *ICLR*, 2022.

[119] J. Dai, S. Wang, K. Tan, Z. Si, X. Qin, K. Niu, and P. Zhang, "Nonlinear transform source-channel coding for semantic communications," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 8, pp. 2300–2316, 2022.

[120] Z. Dai, H. Liu, Q. V. Le, and M. Tan, "Coatnet: Marrying convolution and attention for all data sizes," in *NeurIPS*, 2021.

[121] J. Guo, K. Han, H. Wu, Y. Tang, X. Chen, Y. Wang, and C. Xu, "Cmt: Convolutional neural networks meet vision transformers," in *CVPR*, 2022.

[122] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *CVPR*, 2021.

[123] X. Chu, Z. Tian, Y. Wang, B. Zhang, H. Ren, X. Wei, H. Xia, and C. Shen, "Twins: Revisiting the design of spatial attention in vision transformers," in *NeurIPS*, 2021.

[124] Z. Zhang, H. Zhang, L. Zhao, T. Chen, S. Arik, and T. Pfister, "Nested hierarchical transformer: Towards accurate, data-efficient and interpretable visual understanding," *AAAI*, 2022.

[125] T. Xiao, M. Singh, E. Mintun, T. Darrell, P. Dollar, and R. Girshick, "Early convolutions help transformers see better," in *NeurIPS*, 2021.

[126] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *ICLR*, 2020.

[127] D. Hendrycks, N. Mu, E. D. Cubuk, B. Zoph, J. Gilmer, and B. Lakshminarayanan, "Augmix: A simple data processing method to improve robustness and uncertainty," *arXiv preprint arXiv:1912.02781*, 2019.

[128] K. Yang, S. Wang, J. Dai, K. Tan, K. Niu, and P. Zhang, "Witt: A wireless image transmission transformer for semantic communications," 2022. [Online]. Available: https://arxiv.org/abs/2211.00937

[129] J. Yang, C. Li, and J. Gao, "Focal modulation networks," *arXiv preprint arXiv:2203.11926*, 2022.

[130] T.-Y. Tung, D. B. Kurka, M. Jankowski, and D. Gündüz, "Deepjscc-q: Constellation constrained deep joint source-channel coding," *IEEE Journal on Selected Areas in Information Theory*, 2022.

[131] T.-Y. Tung and D. Gunduz, "Deep joint source-channel and encryption coding: Secure semantic communications," *arXiv preprint arXiv:2208.09245*, 2022.

[132] W. Sirichotedumrong, T. Maekawa, Y. Kinoshita, and H. Kiya, "Privacy-preserving deep neural networks with pixel-based image encryption considering data augmentation in the encrypted domain," in *2019 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2019, pp. 674–678.

[133] C. Louizos, M. Welling, and D. P. Kingma, "Learning sparse neural networks through l_0 regularization," *arXiv preprint arXiv:1712.01312*, 2017.

[134] J. Frankle and M. Carbin, "The lottery ticket hypothesis: Finding sparse, trainable neural networks," in *International Conference on Learning Representations*.

[135] K. Xu, M. Qin, F. Sun, Y. Wang, Y.-K. Chen, and F. Ren, "Learning in the frequency domain," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1740–1749.

[136] T. Yao, Y. Pan, Y. Li, C.-W. Ngo, and T. Mei, "Wave-vit: Unifying wavelet and transformers for visual representation learning," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXV*. Springer, 2022, pp. 328–345.

[137] X. Li, Y. Zhang, J. Yuan, H. Lu, and Y. Zhu, "Discrete cosin transformer: Image modeling from frequency domain," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 5468–5478.