

Article

# Robust Deep Learning Models for OFDM-Based Image Communication Systems in Intelligent Transportation Systems (ITS) for Smart Cities

Nazmul Islam  and Seokjoo Shin \* 

Department of Computer Engineering, Chosun University, Gwangju 61452, Republic of Korea; nazmul@chosun.kr  
\* Correspondence: sjshin@chosun.ac.kr

**Abstract:** Internet of Things (IoT) ecosystem in smart cities demands fast, reliable, and efficient image data transmission to enable real-time Computer Vision (CV) applications. To fulfill these demands, an Orthogonal Frequency Division Multiplexing (OFDM)-based communication system has been widely utilized due to its higher spectral efficiency and data rate. When adapting such a system to achieve fast and reliable image transmission over fading channels, noise is introduced in the signal which heavily distorts the recovered image. This noise independently corrupts pixel values, however, certain intrinsic properties of the image, such as spatial information, may remain intact, which can be extracted as multidimensional features (in the convolution layers) and interpreted (in the top layers) by a Deep Learning (DL) model. Therefore, the current study analyzes the robustness of such DL models utilizing various OFDM-based image communication systems for CV applications in an Intelligent Transportation Systems (ITS) environment. Our analysis has shown that the EfficientNetV2-based model achieved a range of 70–90% accuracy across different OFDM-based image communication systems over the Rayleigh Fading channel. In addition, leveraging different data augmentation techniques further improves accuracy up to 18%.

**Keywords:** artificial intelligence; image processing; intelligent transportation systems; IoT technologies; machine learning; OFDM; wireless communication



**Citation:** Islam, N.; Shin, S. Robust Deep Learning Models for OFDM-Based Image Communication Systems in Intelligent Transportation Systems (ITS) for Smart Cities.

*Electronics* **2023**, *12*, 2425.  
<https://doi.org/10.3390/electronics12112425>

Academic Editors: Binod Vaidya and Byung Rae Cha

Received: 20 April 2023  
Revised: 19 May 2023  
Accepted: 25 May 2023  
Published: 26 May 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The rapid advancement of technology in recent years has led to the development of vast network of interconnected devices and systems known as the Internet of Things (IoT) ecosystem in smart cities, which has transformed the way we interact with the physical world. From smart homes to connected vehicles, IoT technology has enabled smart devices to communicate with each other, share information, and make decisions based on the shared data. These smart devices, sensors, and actuators have enabled us to collect vast amounts of data from our surroundings to develop intelligent systems, such as Intelligent Transportation Systems (ITS), that can transform the way we commute. ITS leverages IoT technologies and mostly relies on image communication systems to use data from sensors and cameras to analyze traffic flow, predict congestion, and optimize traffic signals in real-time to create a smarter and more efficient transportation system [1–4]. The success of ITS relies heavily on the efficient and reliable transmission of data between devices, which requires high data rates and link capacities to handle the large amounts of data generated by the IoT devices in real-time. To achieve this, Orthogonal Frequency Division Multiplexing (OFDM) is widely used as an efficient communication system for high data rate and reliable wireless communication [5,6]. The OFDM communication system uses both frequency and time domain multiplexing to transmit data over multiple sub-carriers. This division of the data stream into several subcarriers allows for parallel transmission, leading to more efficient use of the available bandwidth, and enabling high-speed data transmission with minimal interference and noise. Additionally, OFDM-based communication systems can

benefit from using different modulation techniques to further improve data rate and link capacity. Among them, M-ary Quadrature Amplitude Modulation (M-QAM) is the most efficient technique, which can utilize higher order M-QAM to achieve greater bandwidth efficiency and link capacity with a greater number of bits per symbol [5,7].

Visual data are crucial in ITS environment, and the ability to transmit and process a large number of images efficiently is essential for computer vision (CV) applications in such an environment. While high data rate OFDM-based image communication systems can facilitate the fast transmission of images, there are significant challenges in processing the vast amount of image data with added noise introduced by the utilization of different OFDM systems and the environment itself. To address these challenges, researchers are exploring the use of artificial intelligence (AI) and deep learning (DL) to improve the performance of CV applications in IoT ecosystem. AI and DL have emerged as powerful tools for processing and analyzing large amounts of data in real-time for various CV applications [8]. In the context of ITS, DL algorithms can be used for CV applications to extract meaningful information from image data to enable better decision-making for traffic management and safe driving. For instance, autonomous vehicles rely on AI algorithms to navigate traffic safely and efficiently, and AI algorithms can analyze data from vehicle sensors to identify potential safety hazards and alert drivers or emergency services, contributing to road safety. Additionally, traffic cameras equipped with DL algorithms can analyze traffic patterns and adjust traffic signals in real-time to reduce congestion and enhance road safety.

Although both of these areas of OFDM-based wireless communication systems and DL-based CV applications (such as classification tasks) have been extensively studied in their respective fields, their combination has not been thoroughly explored as per literature review, mainly due to their differences. Therefore, our main objective in this paper is to combine the two and conduct a multi-tier analysis of OFDM-based image communication systems for DL applications while availing third-party computational resources. The contributions of this paper can be further categorized into two major sections:

- Firstly, we present a comprehensive analysis of various OFDM-based image communication systems for CV applications. The analysis includes the use of high-order M-QAM systems to increase data rates, different channel models to simulate real-world environments, and different channel estimation techniques to evaluate the tradeoff between image quality and system complexity. Performance evaluation of the communication systems and the quality of recovered images transmitted over these systems are analyzed using various performance matrices.
- Secondly, we conduct robustness analysis of DL-based CV applications in smart cities, specifically, traffic sign recognition (classification task) in ITS environments. Here, the respective performances of two distinct DL models are analyzed, on recovered images from various OFDM-based image communication systems implemented in the first contribution. Additionally, we have utilized different augmentation techniques to improve the performance of the DL models for the CV application.

The rest of the paper is organized as follows. In Section 2, we present related works on OFDM-based image communication system and robustness of DL models on image perturbation. Section 3 introduces the methodology of the study followed by simulation results and discussion in Section 4. Finally, we conclude the paper with limitations of the study and future research direction.

## 2. Related Work

### 2.1. OFDM-Based Image Communication System

Orthogonal Frequency Division Multiplexing (OFDM) is a standard communication system that has been adapted across various technologies and standards. In this section, we will discuss a few of the related works that utilize OFDM communication system for image transmission. The authors in [9] investigated the transmission of gray scale images using an OFDM system with various even-ordered M-QAM modulations (from 4-QAM to 256-QAM) and channels with different fading and shadowing parameters. The results demonstrate

that lower order M-QAM modulation provides a higher quality of recovered image due to a lower Bit Error Rate (BER). Authors in [10] evaluated the performance of Quadrature Phase Shift Keying (QPSK), 4-QAM, 16-PSK, 16-QAM modulation for an OFDM-based image transmission system over an Additive White Gaussian Noise (AWGN) channel. Their results showed that 16-QAM provided the best performance in terms of image quality and BER. Authors in [11] investigate the integration of Discrete Wavelet Transform (DWT) and Fast Fourier Transform (FFT)-based OFDM with QPSK modulation for the transmission of grayscale images in AWGN and Rayleigh channels. They propose the integration of two adaptive filtering techniques, least mean squares (LMS) and recursive least squares (RLS), to reduce BER and concludes LMS outperforms RLS in terms of noise pattern detection and efficient recovery of the modulating signal. Authors in [12] proposed a new technique for transmitting images over underwater time-dispersive fading channels, called Progressive Zero-Padding (PZP)-OFDM using QPSK modulation, which improves the BER in underwater environments. Authors in [13] analyzed the BER performance of image transmission in non-OFDM and OFDM communication systems under various channel conditions (AWGN, Rician and Rayleigh Fading channels) and modulation techniques (Binary-PSK, QPSK, 16-QAM). Their results show that OFDM has better performance compared to non-OFDM, and overall BPSK in AWGN had the lowest BER. Authors in [14] have implemented a real-time practical OFDM system for image transmission using Raspberry Pi (RPI) and Pluto software-defined radios (PlutoSDR). The performance of the system was evaluated by comparing the BER for different modulation schemes, including BPSK, QPSK, 16-PSK, and 256-PSK, where BPSK had the best performance. Authors in [15] investigated the impact of channel estimation for OFDM based image transmission through AWGN channel with various modulation schemes (BPSK, QPSK, 8-PSK, and 16-QAM), and concluded that Least Square (LS) method significantly improved the quality of the restored images. Authors in [16] employed two types of OFDM systems, FFT-OFDM and DWT-OFDM, to transmit compressed images using Discrete Cosine Transform (DCT), Wavelet (WAV) transform, and compressive sensing methods using different modulation techniques (BPSK, QPSK, 16-QAM, and 64-QAM) over AWGN channel. Their results showed that 16-QAM DWT-OFDM outperformed FFT-OFDM with lower Mean Square Error (MSE) while avoiding the use of a cyclic prefix. Authors in [17] compared different modulation techniques (QPSK, 16-QAM, and 64-QAM) for transmitting grayscale and Red Green Blue (RGB) images over OFDM image communication system under varying channel conditions (AWGN and Rayleigh Fading), and filters (Wiener, Median, and No Filter). Their results show QPSK in AWGN channel had better recovered images using a median filterer for a lower Signal-to-Noise Ratio (SNR) value. Authors in [18] investigated the effect of improved tone reservation (ITR) on the peak-to-average power ratio (PAPR) of a 16-QAM-OFDM system for transmitting images over AWGN channel. Their results show that ITR can effectively reduce PAPR, leading to improved image quality and fewer transmission errors. The authors in [19,20] performed a comprehensive analysis of image communication under various conditions in Single User Multiple Input Multiple Output (SU-MIMO) and Massive MIMO OFDM systems, respectively. They evaluated the effect of various channel conditions (AWGN, Rayleigh), modulation techniques (BPSK, QPSK, 8-PSK, 16-PSK, 32-PSK, 64-PSK), antenna configurations, number of users, and transformation techniques (FFT, Fractional Fourier Transform, DWT, DCT) on image communication in Fifth Generation (5G) networks. The results from the former study indicated that using the DCT transformation technique improved the BER compared to FFT and increasing the number of received antennas led to higher quality recovered images. The later study indicated that DWT outperformed FRFT and FFT in terms of BER; however, as the number of users continued to increase in the Massive MIMO system, the quality of the recovered image decreased irrespective of the transformation technique used. Additionally, both studies showed that lower-order PSK modulation produced better results overall.

There have been several studies of image transmission across OFDM communication system with DL based channel coding and decoding techniques for CV applications. For

instance, [21] first proposed deep joint source-channel coding (JSCC) for wireless image transmission by combining the source and channel coding into a single auto-encoder structure. It optimized source compression and error correction coding through back-propagation, which outperformed conventional schemes (separate source and channel coding) under AWGN and Rayleigh flat fading channels. This scheme was further extended to larger neural networks [22], feedback network [23], progressive transmission [24], and additional attention modules [25,26]. Motivated by model-based deep learning, DL-based JSCC were also extended to OFDM image communication system with fading channels in [27,28]. The former study fed the neural network-encoded image as frequency domain OFDM baseband symbols and used adversarial measures to further improve the quality of reconstructed images. The latter utilized double attention mechanism to better map image features to subchannels and ensure important features were transmitted over the high-quality subchannels. Both [27,28] outperformed conventional separate coding schemes.

The aforementioned research works focused on OFDM-based image communication systems; however, they have not considered, multi-tier analysis of the blocks in the OFDM physical layer (PHY) that encompasses higher-order modulation schemes, various channel models and channel estimation. Additionally, it is not yet practical to adapt PHY as a deep learning architecture. Furthermore, DL application of the received images for a specific task has not been considered, only system performance and image quality has been studied. The authors in [29–31] attempted to mitigate this gap and considered DL-based applications, and analyzed the performance of DL models on received images from various OFDM communication systems. Authors in [29] determined that images from FFT-based OFDM were better for image transmission for cloud-based DL applications, as opposed to DCT-based systems. Authors [30] investigated the impact of channel correction on DL models in the OFDM-based image communication system and showed that channel correction had a major impact on the quality of recovered images, although deep learning models still produced acceptable results without it (with a lesser degree of improvement). Authors in [31] evaluated the impact of higher order M-QAM on the performance of the DL model. Their results showed that the DL model accuracy was lower on images from higher order M-QAM compared to lower order M-QAM, but the overall accuracy was still relatively high and suitable for applications that are unaffected by small fluctuations in DL accuracy. The studies have investigated the DL-based applications for image communication systems; however, they have not considered multi-tier analysis of the OFDM communication system, including fading channels, modulation techniques, and channel estimation methods. Additionally, they have not evaluated different DL models nor have they exploited any model training techniques, such as data augmentation, to improve the DL model accuracy. Therefore, for this study we consider multi-tier analysis of various OFDM-based image communication systems and robustness analysis of multiple DL models on images recovered from these systems to enable CV applications in ITS. The interoperability and seamless integrability of OFDM systems with various modulation techniques makes it the ideal choice for hybrid systems, such as massive MIMO [32]. Therefore, the focus of this paper is on the physical layer of OFDM communication systems, and the MIMO systems are left for future work.

## 2.2. Robustness of DL Models on Image Perturbation

Robustness analysis of DL models on noisy images from different communication systems for CV applications are also considered in this study. In this section, we will discuss the related work based on robustness of DL models on images with different types of noise perturbation. Despite the success of Deep Neural Network (DNN) with complex and high-dimensional image data, they are still not robust against image perturbations. Authors in [33] were the first to discover the vulnerability of DNNs to certain input perturbations, which result in significant discontinuities in their outputs. These perturbations were named *adversarial examples* and were found to cause a wide range of deep learning models to misclassify, regardless of the different model architecture or training data. The

authors also examined several interesting properties of neural networks, including their capacity for learning intricate functions, the existence of adversarial examples that can confuse the network, and their ability to generalize and extrapolate. Authors in [34] carry out further studies on adversarial models and present defense systems, such as adversarial training, input transformations, and robust optimization to minimize the effect of adversarial examples.

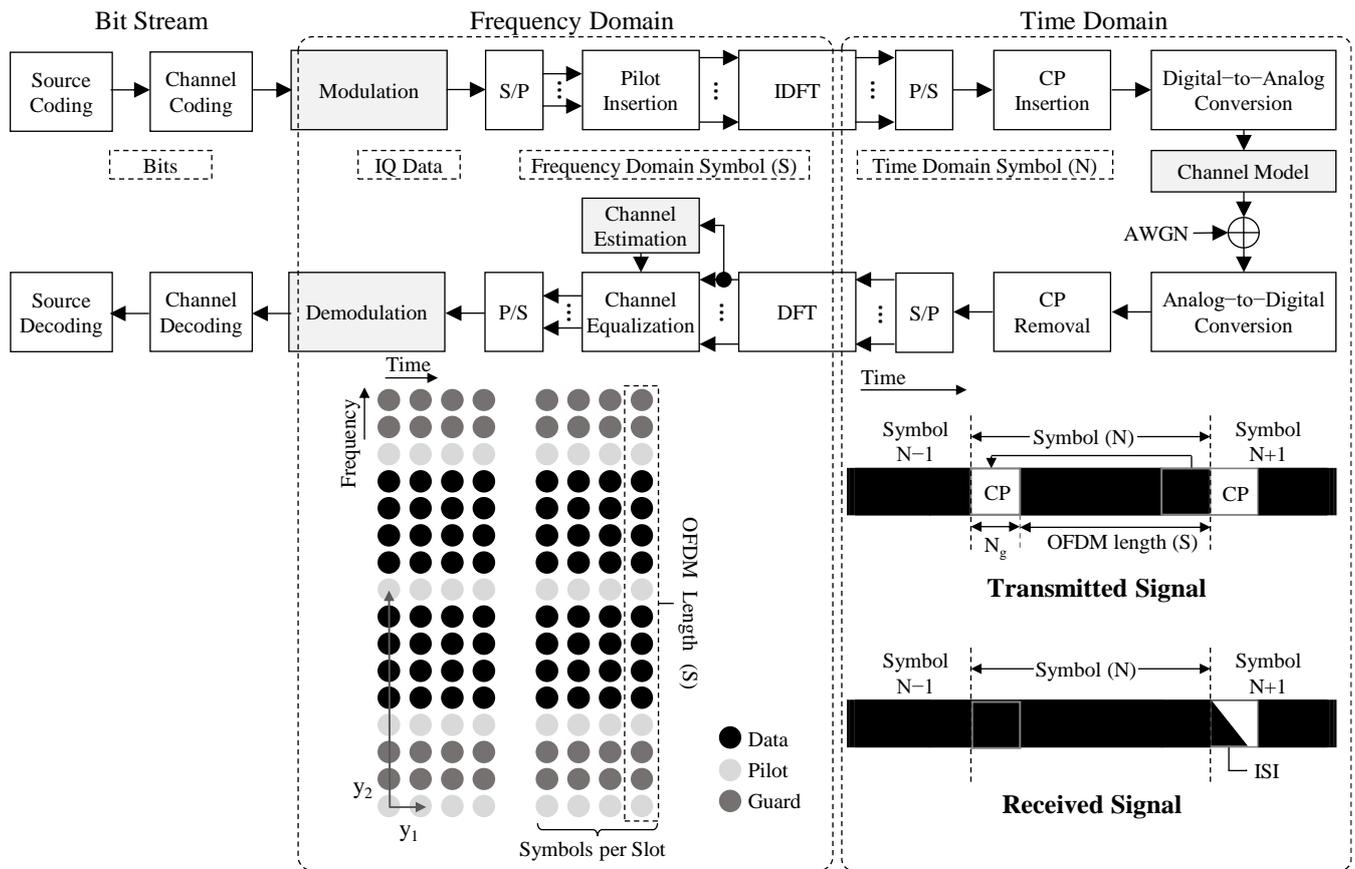
Among various image perturbation, visible noise on digital images is a type of perturbation resulting from perceptible alterations made to the image pixels, and it can have an impact on DL-based image classification tasks. The introduction of visible noise in digital images during the processes of its acquisition, encoding, transmission, and processing is an inherent consequence of the utilization of electronic components, particularly sensors and actuators [35,36]. A comprehensive overview of digital image noise models can be found in [36], which highlights various types of noise that can impact digital images, such as additive Gaussian noise, quantization noise, color quantization with dither, salt–pepper noise, Rayleigh noise, gamma noise, uniform noise (white noise), and Poisson noise. The paper also discusses methods for modeling and analyzing these noises using probability density functions and statistical models, and common techniques for removing them, such as spatial filtering, frequency filtering, and wavelet denoising.

Deep learning has made remarkable advancements in various domains, particularly in image classification. However, the accuracy of these models degrades when images are subjected to such distortions and noise [37]. Authors in [37] evaluated various factors that contribute to image quality, including resolution, noise, and blur, and evaluated their individual impacts on DNNs. The outcomes indicated that noise and blur have a stronger impact on DNNs than other factors. The authors also presented strategies, such as pre-processing and data augmentation, for improving DNN performance on low-quality images. Authors in [38] evaluated the robustness of Convolutional Neural Networks (CNN) to various types of image degradations, such as Gaussian noise, salt and pepper noise, Joint Photographic Experts Group (JPEG) compression, and motion-blur. The results showed that CNNs are generally more robust to image degradation compared to traditional methods, but they still have limitations. To address these limitations, the authors proposed a CNN architecture called a capsule network, which was shown to be more robust to degradation. Authors in [39] investigate the effect of color information on the robustness of CNNs for image classification tasks. They have compared the performance of CNNs trained on color images and grayscale images when subjected to degradation, such as Gaussian noise and JPEG compression. The results show that CNNs trained on color images are more robust to degradation compared to those trained on grayscale images, due to the additional cues provided by the color information. In CV tasks, CNNs have shown outstanding performance when applied to identically and independently distributed data. However, they are vulnerable to changes in data distribution [33] and color corruption [40] as they only use local features [41,42]. Authors in [43] proposed Vision Transformers (ViT) to mitigate this challenge by considering global image context and reduced bias towards local textures, resulting in robustness towards occlusions. Although the aforementioned studies have shown promising results in improving the accuracy and robustness of CNN models against various noise perturbation on the images, they have not considered various communications systems and the performance of DL models on images with heavy noise perturbations introduced by these communication systems. Therefore, in current study we focus on the performance of CNN models against noise from various OFDM-based image communication systems for CV applications in ITS.

### 3. Methods

#### 3.1. Physical Layer of OFDM Communication System

OFDM is a parallel transmission system that allows spectra of subchannel to overlap with subcarrier tones and coherent detection maximizing the available bandwidth. The physical layer (PHY) of the OFDM communication system is shown in Figure 1.



**Figure 1.** Basic block diagram of OFDM physical layer with frequency–domain and time domain symbol representation.

In the OFDM system, serial high data-rate sub-streams are divided into low data-rate sub-streams and encoded onto orthogonal subcarriers. The spectral shape of the subcarriers is such that the discrete subchannels at the orthogonal subcarrier frequencies have a spectrum of 0 and there are no Inter Carrier Interference (ICI) among the subchannels. As seen in Figure 1, the encoded bit stream is mapped onto a complex-valued in-phase and quadrature (IQ) constellation plane and converted to IQ data according to the chosen modulation techniques in the modulation block. Deployed modulation techniques will be further discussed in Section 3.1.3. The serial data stream is then converted into parallel data stream and a frequency domain OFDM symbol,  $X(k)$ , is generated by inserting training data and guard bands into the IQ data. The training data are the pilot symbols (known data-sequence) that are inserted in the data stream to carry out channel estimation [44]. For pilot insertion, there are multiple techniques based on the location and order of pilots in the transmission. We have used comb-type pilot insertion with linear interpolation as it provides improved results [44]. In comb-type pilot insertion, pilots are inserted into specific OFDM subcarriers that are continuously transmitted throughout the communication.

The frequency domain OFDM symbol,  $X(k)$ , is then transformed into the time-domain OFDM symbol,  $x_t(n)$ , using an N-point inverse discrete Fourier transform (IDFT) followed by parallel-to-serial (P/S) transmission. IDFT transformation reduces the system complexity of parallel symbols and removes any pulse shift that occurred in the modulation process [7]. The IDFT equation is given below,

$$x_t(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)e^{\frac{j2\pi kn}{N}}, \tag{1}$$

where  $j = \sqrt{-1}$ ,  $N$  is the DFT length,  $X(k)$  is the frequency-domain OFDM symbol for  $k = 0, 1, \dots, N - 1$ , and  $x_t(n)$  is the resultant time-domain OFDM symbol. In the next step, a guard interval known as a cyclic prefix (CP), is sliced from the end of  $x_t(n)$  symbol and prepended to  $x_t(n)$  to form the full time-domain OFDM symbol,  $x_{cp}(n)$ , which extends the portion of the OFDM symbol in a cyclic manner. The CP mitigates the Inter-Symbol Interference (ISI) and ICI from multipath radio channels and neighboring OFDM symbols, and it is selected to be larger than the anticipated delay spread. OFDM symbol with CP length  $N_g$  is given below:

$$x_{cp}(n) = \begin{cases} x_t(N + n), & -N_g \leq n \leq \dots, N - 1 \\ x_t(n), & \text{otherwise} \end{cases}. \quad (2)$$

The signal,  $x_{cp}(n)$ , is then elevated to the radio frequency (RF) by the RF front end and transmitted through the air via wireless channel. During the transmission over the air, the signal experiences the effect of multiple paths and fading effect that creates duplicate signals with different time lags, phases, and amplitudes at the receiver. Applied wireless channel models will be further discussed in Section 3.1.1. The radio signal is then captured and re-converted to IQ samples by the receiver front end. The carrier synchronizer recovers the time-domain OFDM symbols at the receiver. The CP is then removed from  $y_{cp}(n)$  to yield the remaining IQ samples,  $y(n)$ , which is given by:

$$y(n) = \begin{cases} y_{cp}(n), & -N_g \leq n \leq \dots, N - 1 \\ y_{cp}(n + N_g), & \text{otherwise} \end{cases}. \quad (3)$$

The symbols are then converted to parallel stream and is transformed into the frequency-domain OFDM symbol,  $Y(k)$ , using discrete Fourier transform (DFT). The DFT transformation is given by:

$$Y(k) = \frac{1}{N} \sum_{n=0}^{N-1} y(n) e^{-\frac{j2\pi kn}{N}}. \quad (4)$$

The symbol  $Y(k)$  undergoes channel estimation and channel equalization to produce an estimate of the transmit frequency-domain IQ data,  $\hat{X}(k)$ . The channel estimation compares the received data sequence and pilot symbols with the previously known ones to obtain an estimate of the Channel Impulse Response (CIR) and perform channel correction. Channel estimation techniques used in the experiment will be further discussed in Section 3.1.2. This estimate is then demodulated into soft bits (log-likelihood),  $\tilde{b}$ , which are further converted into binary output bits,  $\hat{b}$ , by the channel decoder before passing it to the next communication layer.

For source coding and decoding, variable length coding (VLC) is used instead of full-length coding (FLC) as it reduces the code length and does not concern errors in padded bits as opposed to FLC [45]. It is simpler than probabilistic coding techniques and avoids synchronization problems at the receiver, improving the visual perception quality of the recovered images. However, its limitation is the additional overhead and inefficiency due to sending side information (code length) for each transmission (each image in our case). The extent of this paper is focused on the lower PHY while the source and channel coding are interesting directions for the future.

### 3.1.1. Wireless Channel

A simplified wireless channel is well specified in [44,46]. The time-domain received signal over a wireless channel can be defined as

$$y_{cp}(n) = x_{cp}(n) \otimes h(n) + w(n), \quad (5)$$

where  $y_{cp}(n)$ ,  $x_{cp}(n)$ ,  $h(n)$ ,  $w(n)$ , and  $\otimes$  are the received signal, transmitted signal, channel impulse response, AWGN, and convolution operator, respectively, in the time domain. The frequency-domain received signal is obtained by DFT transformation of (5) as

$$Y(k) = X(k) \cdot H(k) + W(k), \tag{6}$$

where  $Y(k)$ ,  $X(k)$ ,  $H(k)$ ,  $W(k)$ , and  $\cdot$  are the received signal, transmitted signal, channel impulse response, AWGN, and element-wise product, respectively, in the frequency domain. AWGN with zero mean Gaussian distribution and uniform spectral density is often used to model the additive noise (for example, electric noise, thermal noise, interference) in the OFDM system. Therefore, the AWGN,  $w(n)$ , from Gaussian distribution with zero mean and standard deviation, is given by

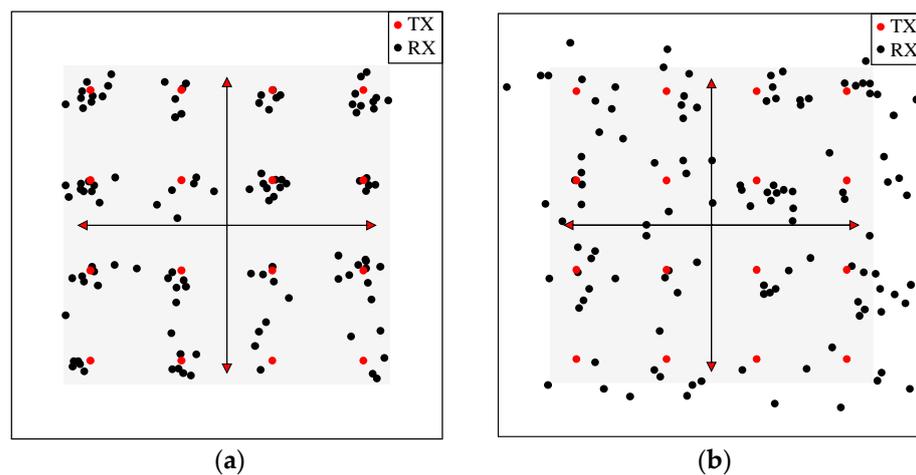
$$w(n) = \begin{cases} \sigma \cdot \mathcal{N}_N(0, \sigma), & \text{For real} \\ \sigma \cdot [\mathcal{N}_N(0, 1) + j\mathcal{N}_N(0, \sigma)], & \text{For complex} \end{cases} \tag{7}$$

where  $\mathcal{N}_N(0, \sigma)$  is the Gaussian noise vector with 0 mean and standard deviation  $\sigma = \frac{N_0}{2}$ , and  $N_0$  is the power spectral density.

Signal transmitted over wireless channel experience reflection, diffraction, and scattering, leading to multiple versions of the same signal arriving at the receiver with different amplitude, phase, and delay. This phenomenon is known as the multipath fading effect and can be modelled as linear finite impulse-response (FIR) filter give as [47]

$$y_t = \sum_{l=0}^{L-1} h_l \cdot x_{(t-l)}, \text{ where, } h_l = \sum_{k=1}^K \sqrt{\Omega_k} z_k \text{sinc}\left(\frac{\tau_k}{T_s} - l\right), \tag{8}$$

vector  $\Omega$  and  $\tau$  are the power-delay profile (PDP) of fading process, and  $z_k$  and  $T_s$  are the complex-valued variable and sampling period of the discrete signal, respectively.  $L$  is the filter length chosen, such that  $|h_l|$  is small for  $l \geq L$  or  $l < 0$ . The effect of different channels on frequency-domain symbols of the proposed 16-QAM OFDM system is shown in Figure 2.



**Figure 2.** Effect of Various Channel Models on 16-QAM Modulation Using LS Channel Estimation at  $E_b/N_0$  15 dB under (a) the AWGN Channel and (b) the Rayleigh Fading Channel with AWGN.

For the multipath Rayleigh Fading channel, the real and imaginary part of  $z_k$  are independent and identically distributed (IID) Gaussian random variables, therefore  $|z_k|^2$  follows the Rayleigh Fading distribution.  $K$  is the number of paths in multipath fading channel and if  $K = 1$  it is said to be flat fading with only on path, whereas, if  $K > 1$  it is frequency selective fading with multipath interference. For the experiment, we have

considered simple AWGN channel and Rayleigh Fading channel for the OFDM-based image communication system.

### 3.1.2. Channel Estimation

Channel estimation is crucial to obtain information about the Channel Impulse Response (CIR). Pilot-aided channel estimation is often used at the receiver by inserting known data sequences (pilots) in the form of block, comb, or scatter patterns to sample the channel distortions [48,49]. These pilots are recognizable at the receiver and are either constant or low auto-correlation sequences, such as Zadoff–Chu sequences. For channel correction, the received bit sequence is compared with the known ones at the receiver to estimate the CIR, which is then equalized to mitigate the channel distortions [48,50].

The most common pilot-aided channel estimation method is the least square (LS) technique. The LS method increases the sampling rate of the channel frequency responses at the pilot subcarriers, and then uses interpolation, such as linear, spline, or cubic, to estimate the Channel Impulse Response (CIR) at the data subcarriers [44,46,49]. The LS estimator of the channel is given by:

$$\hat{X}(k) = \frac{Y(k)}{\hat{H}_p^{LS}}, \quad (9)$$

$$\text{where, } \hat{H}_p^{LS} = \mathcal{F}\left(\frac{Y_p}{X_p}\right). \quad (10)$$

$\hat{X}(k)$  is the estimated channel,  $\hat{H}_p^{LS}$  contains the LS estimate,  $Y_p \in \mathbb{C}^{N_p \times 1}$  is the received pilots,  $X_p \in \mathbb{C}^{N_p \times N_p}$  is a matrix with known transmitted pilot symbols on its diagonals, and  $\mathcal{F}(\cdot)$  is the interpolation operation, such as linear, cubic, spline, etc. [48]. The LS algorithm is simple because it disregards noise and ICI, therefore, it is calculated with minimal complexity without any channel statistics knowledge; however, it results in high means square error (MSE).

Minimum Mean Square Error (MMSE) channel estimation technique utilizes the second-order statistics of the channel conditions (prior channel knowledge) and performs channel estimation [51,52]. It uses channel auto covariance to reduce the MSE, as shown below

$$\hat{H}_p^{MMSE} = R_H [R_H + (X_p X_p^H)^{-1}]^{-1} \hat{H}_p^{LS}, \quad (11)$$

where  $R_H$  is the frequency-domain correlation matrix at the pilot-symbols with  $R_H = E[H_p H_p^H]$  and the channel coefficient matrix can be obtained via interpolation [51]. In our experiment we have considered both LS and MMSE channel estimation with linear interpolation. Linear interpolation utilizes the frequency response of two neighboring pilots to estimate the frequency response of the data subcarriers in between them, therefore, it has simple computational complexity [51]. Overall, MMSE channel estimation outperforms LS estimator, especially in lower  $E_b/N_0$  regions. However, as the number of subcarriers in an OFDM system increases, the computational complexity of the MMSE estimator also increases due to the computations required for matrix inversions [53].

### 3.1.3. M-QAM Modulation

Most practical OFDM communication systems exhibit passband frequency responses, and to achieve high throughput with bandwidth and power efficiency, digital passband modulation is frequently employed in these systems. Digital passband modulation schemes can be categorized according to the variation in amplitude, phase, or frequency of the transmitted signal with respect to the message signal, such as ASK, PSK, FSK, and QAM. Among them, M-QAM is an advanced modulation technique that varies both amplitude and phase of the transmitted signal simultaneously, resulting in greater bandwidth and power efficiency which can be extended to advanced systems, such as MIMO [5,6,54]. Additionally, high data rate can be achieved when utilizing higher order M-QAM; however,

using higher order M-QAM for higher data rates comes with a cost of higher BER, cell-to-cell interference, smaller coverage area, and hardware complexity [5,6,54]. Despite these challenges, M-QAM is extensively used in various IEEE standards and wireless communication systems, such as 5G digital video broadcast communications, satellite communications, Wi-Fi, WiMAX, VDSL, and more. A few applications with different order of M-QAM for modern communication standards are given in Table 1 [55].

**Table 1.** Applications utilizing M-QAM Modulation for modern communication standards.

Communication Systems	M-QAM Utilized
IEEE 802.11n/g/ad/ay	16, 64
IEEE 802.16m (WiMAX 2)	16, 64
IEEE 802.11ac/af/ah	16, 64,256
DVB-T2 (Digital Video Broadcasting-Terrestrial)	16, 64,256
IEEE 802.22b	16, 64,256
TS 36.331 (Release 14-LTE Advanced Pro)	16, 64,256
TS 36.331 (Release 15-5G support)	16, 64,256, 1024
IEEE 802.11ax (Wi-fi 6)	16, 64,256, 1024

The M-QAM signal waveforms are composed of two independently amplitude-modulated carriers that are in quadrature to each other as shown below [56]

$$s(t) = I(t) \cos(2\pi f_c t) - Q(t) \sin(2\pi f_c t), \quad 0 \leq t \leq T, \quad (12)$$

where  $T$  time duration of symbol,  $f_c$  is the carrier frequency, and  $I(t)$  and  $Q(t)$  are the respective IQ components of the signal.

Wireless communication systems aim to achieve high data-rate transmission with efficient utilization of limited bandwidth for the best quality of service (QoS), and high spectral and power efficiency. This can be achieved by reducing the average transmit power of the constellation design in the modulation schemes [5,6,54]. Most of the M-QAM constellation designs proposed decades ago (from early 1960s) are still being used in commercial communication systems. M-QAM has the unique capability to encode information in both amplitude and phase of the transmitted signal, allowing higher spectral efficiency. This enables more bits to be encoded per symbol for a given average energy in the constellation design [5,6]. Rectangular M-QAM (RQAM) constellation design is widely used due to its improved error performance and use of a simple maximum likelihood detection method, which reduces system complexity [55,57]. Square M-QAM (SQAM) is a type of RQAM constellation that is optimized for even-length symbols (16-QAM, 64-QAM, 256-QAM, and 1024-QAM) [58]. The distribution of constellation points (symbols) in SQAM forms a perfect square lattice with equal rows and columns and maximizes the minimum Euclidean distance between points for a given average symbol power. In SQAM, each constellation points shares boundaries with a maximum of four neighboring points and differs by only one bit, therefore it has perfect gray coding, with a Gray Code penalty (GP) value 1 [57]. The GP was introduced in [57] as the average difference in bits between adjacent symbols in a constellation

$$G_P = \frac{1}{M} \sum_{i=1}^M G_P^{S_i}, \quad (13)$$

where  $M = 2^k$ ,  $k$  is the length of symbol and  $G_P^{S_i}$  is the GP for the  $i^{th}$  data symbol,  $S_i$ . In SQAM, the  $k$  bits of the serial data stream are represented on a two-dimensional constellation design using gray coding. The  $I(t)$  and  $Q(t)$  distributed over the set  $\{\pm d, \pm 3d, \dots, \pm(M-1)d\}$

independently, where  $2d$  is the adjacent Euclidean distance between the two constellation points given by [56]

$$d = \sqrt{\frac{3 \log_2 M \cdot E_b}{(M^2 - 1)}}, \quad (14)$$

where  $E_b$  is the bit energy and  $M$  is the order of QAM. Bit Error Rate (BER) is a vital factor in evaluating the effectiveness of constellation designs in modulation schemes. Later sections will cover the analysis of the experimental results, but the theoretical bit error calculation of SQAM is given by [56]

$$B_e = \frac{1}{\log_2 \sqrt{M}} \sum_{i=1}^{\log_2 \sqrt{M}} B_e(i), \quad (15)$$

where  $B_e(i)$  is the bit error probability of the  $i^{\text{th}}$  bit.

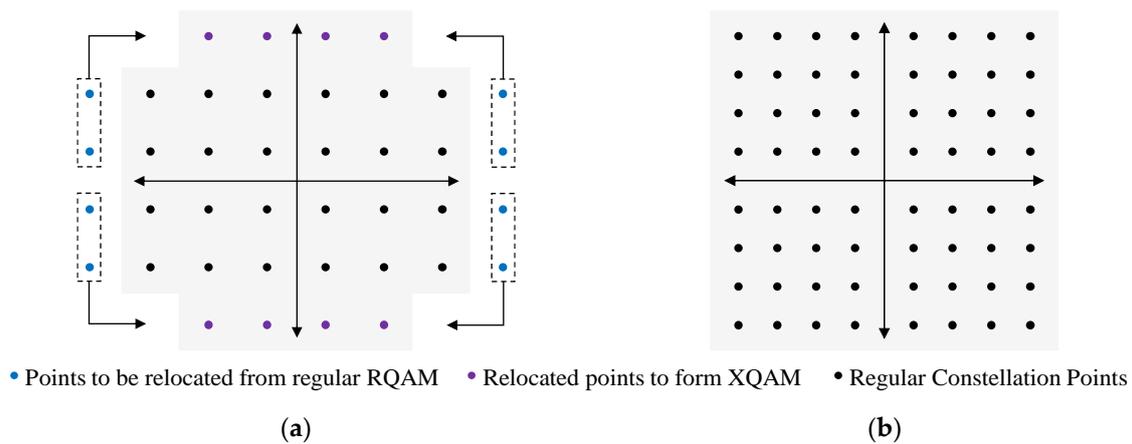
When designing constellation points with odd length of symbols (32-QAM, 128-QAM, 512-QAM) in RQAM, the constellation points are usually distributed to form a perfect rectangle, either with a horizontal shape parallel to the in-phase axis or with a vertical shape parallel to the imaginary axis, both having the same average energy; however, this is not desirable as it increases the peak and average powers. To overcome this, [57] proposes SQAM with the outer corner constellation points relocated for odd length of symbols forming a cross shape constellation design referred to as cross-constellation M-QAM (XQAM). XQAM reduces the peak and average energy and provides at least 1 dB gain over perfect rectangular shape constellation design [57,59]. The generalized GP for higher order XQAM constellation design can be given as  $\left(1 + \frac{1}{\sqrt{2M}} + \frac{1}{3M}\right)$  [59]. For the BER calculations, authors in [57] proposed an approximate expression which was reproduced by authors in [59] for XQAM as

$$B_e(M) \cong G_p \frac{A_n}{\log_2 M} \frac{1}{2} \operatorname{erfc} \left( \frac{d}{\sqrt{N_0}} \right), \quad (16)$$

where  $G_p$  is the GP,  $A_n$  is the average number of nearest neighbors of a constellation point [given by  $\left(4 - \left(\frac{6}{\sqrt{2M}}\right)\right)$  for XQAM of  $M \geq 32$ ],  $N_0$  is the noise density,  $2d$  is the adjacent Euclidean distance between the two constellation points and  $\operatorname{erfc}(\cdot)$  is the error function defined as:

$$\operatorname{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^{\infty} e^{-t^2} dt. \quad (17)$$

The decision region for SQAM is simple because the IQ component has distinct vertical and horizontal decision regions. However, for XQAM constellations, only horizontal and vertical decision regions are not sufficient as their corner constellation points are not available. In XQAM, the end columns symbols are relocated to new cross-type positions in the constellation, forming three types of symbols, edge symbols, corner symbols, and interior symbols [57,59], as shown in Figure 3. The inner symbols form a closed square decision region, while the edge symbols form semi-infinite rectangles. The decision region for the corner symbols forms a 45-degree angle along the horizontal and vertical axes. Due to this irregular pattern, some of the noisy symbols are not transformed into the mapped decision region, resulting in higher bits in error for XQAM constellation design [5,55].



**Figure 3.** Constellation design for M-QAM modulation. (a) XQAM for odd length of symbol (32-QAM). (b) SQAM for even length of symbol (64-QAM).

Higher order M-QAM utilizes a greater number of bits per symbol and can significantly increase the data rate. The potential capacity of a system utilizing higher order M-QAM can be calculated by using Shannon’s channel capacity formula [60] as

$$C = B \log_2 \left( 1 + \frac{S}{N} \right) \text{ bits/s,} \tag{18}$$

where  $C$  is the channel capacity in bits per second,  $B$  is the bandwidth of the channel in Hertz,  $S$  is the signal power, and  $N$  is the noise power. Using Equation (18) to express Shannon bound as bandwidth efficiency,  $\eta = \frac{C}{B}$ , it can be expressed as:

$$\eta = \log_2 \left( 1 + \frac{S}{N} \right) \text{ bits/s/Hz.} \tag{19}$$

In IEEE 802.11ax (Wi-Fi 6) standard, 160 MHz is the maximum allocated channel width [61]. Considering that, we can calculate the bandwidth efficiency using Equation (19) for higher order M-QAM, as given in Table 2. It shows that with higher order M-QAM, the data rate increases significantly. Additionally, the table also shows PAPR and GP values for higher order M-QAM calculated in [55]. From the table, we can observe that XQAM exhibits higher GP compared to SQAM due to its irregular pattern; however, SQAM has higher PAPR compared to XQAM, due to its full lattice structure [55]. For the experiment we have considered higher order M-QAM (16-QAM to 1024-QAM) modulation with both SQAM and XQAM constellation design for the OFDM-based image communication system.

**Table 2.** Parameters and bandwidth efficiency of various XQAM and SQAM constellation designs.

M-QAM	Bits per Symbol (k)	PAPR	GP	Bandwidth Efficiency <sup>†</sup>
16 <sup>+</sup>	4	1.80	1	640 Mbps
32 <sup>*</sup>	5	1.70	1.166	800 Mbps
64 <sup>+</sup>	6	2.333	1	960 Mbps
128 <sup>*</sup>	7	2.073	1.065	1.12 Gbps
256 <sup>+</sup>	8	2.647	1	1.28 Gbps
512 <sup>*</sup>	9	2.28	1.039	1.44 Gbps
1024 <sup>+</sup>	10	2.81	1	1.6 Gbps

<sup>\*</sup> XQAM constellation; <sup>+</sup> SQAM constellation; <sup>†</sup> Per 160 MHz Channel.

### 3.2. Deep Learning for Image Classification Task

The Artificial Neural Network (ANN) can learn from examples and use that knowledge to classify new, unseen data. This is referred to as a *classification* task and ANNs are

particularly well-suited to this task [62,63]. The capability to classify new unseen data (known as *generalization*) make ANNs highly effective in solving classification problems where the desired outcome cannot be pre-determined [63].

ANNs consist of input and output layers, with one or more hidden layers made up of interconnected nodes called neurons. The most common type of layer is the fully connected layer, where every neuron is connected to every other neuron in the previous and next layer. The *activation function* determines the transmission of information between these neurons in the layers. To effectively train an ANN, a loss is calculated between the predicted outputs and the actual outputs. The training algorithm tracks and updates the weights of the neurons to minimize the loss over time, using *optimization* techniques. Moreover, *backpropagation* algorithm is used to propagate error information from the last layer to the first layer for weight modification, and train the model in iteration or epoch. During training, the model assesses its performance on validation data (which is distinct from the training data) to fine-tune the hyperparameters and improve the model architecture. If the model is not trained enough, it is referred to as *underfitting*, meaning that it did not learn the training set well enough. On the other hand, *overfitting* occurs when the network has learned the training set too well, causing it to struggle with new data [64]. Different regularization techniques are adapted to mitigate the underfitting and overfitting problem. The final evaluation of the model's generalization is performed on unseen data, known as testing or inferencing.

A Deep Neural Network (DNN) is a type of ANN with multiple hidden layers. The term "deep" refers to the number of sequential layers within the network, indicating the number of times the input data passes through the transfer functions of sequential layers. This results in a more complex model capable of capturing and representing high-level abstractions and patterns in the data. DNNs are designed to learn and make predictions based on large amounts of complex data, by processing and transforming the data iteratively through multiple hidden layers [8]. Convolutional neural network (CNN) is a type of DNN that was introduced in [65] to provide an efficient learning method for images. The CNN architecture mostly consists of a convolutional layer, pooling layer, activation function, fully connected layer, and output layer. The convolutional layer is used for extracting features from input data. It utilizes a filter (represented by a small weight matrix) that slides over the input data, performing element-wise multiplication and addition to produce a set of *feature maps*. These filters have adjustable weights, learned through backpropagation, and are used to recognize local patterns in the input data. The convolutional layer is effective in learning features that are invariant to shifts in the input, reducing model complexity and improving model generalization. The size, stride, and padding of the filters can be adjusted to control the spatial dimensions of the output feature maps. By using multiple convolutional layers, the model can learn hierarchical representations of the input data, making it effective for tasks such as image classification and object detection.

A convolution layer consists of a set of  $F$  filters known as *depth*. The weights of the filters are  $W^f \in \mathbb{R}^{a \times b}$ ,  $f = 1, \dots, F$ . These weights generate feature map,  $Y^f \in \mathbb{R}^{n' \times m'}$  from an input matrix  $X \in \mathbb{R}^{n \times m}$  as per the given convolution

$$Y_{i,j}^f = \sum_{k=0}^{a-1} \sum_{l=0}^{b-1} W_{a-k,b-l}^f \cdot X_{1+s(i-1)-k, 1+s(j-1)-l}, \quad (20)$$

where  $s$  is called *stride* with value  $\geq 1$ ,  $n' = 1 + (n + a - 2)/s$  and  $m' = 1 + (m + b - 2)/s$ . It is assumed  $X$  is zero-padded which denotes,  $X_{i,j} = 0$  for all  $i \notin [1, n]$  and  $j \notin [1, m]$ . Furthermore, in image processing,  $X$  is usually a three-dimensional tensor; therefore, the weights are three-dimensional as well and applies on all input channels simultaneously. After obtaining the feature maps from a convolution layer, a *pooling layer* performs down-sampling on the feature maps obtained from the previous convolution layer. The purpose of pooling is to reduce the spatial dimensions of the feature maps, making the network more

computationally efficient, and invariant to translations in the input image. Therefore, the pooling layers partitions  $Y$  into  $p \times p$  regions with values representing the most important information from the set and computes a single output value using pooling techniques such as max-pooling and average-pooling, which, respectively, take the maximum or average value from the features in the pooling window as the output before passing it to the next layer.

The *activation function* (non-linearity layer) in CNN introduces non-linearity into the network, which benefits the stacking of multiple layers in a network. The activation function is typically applied to each element of the input vector and is used to determine the output of the network, often in a binary format, such as 0 or 1. The resulting non-linearity enables deep neural networks to model complex, non-linear systems and provides a powerful tool for pattern recognition tasks. Previously, *tanh* and *sigmoid* activation functions were popular; however, with the understanding that most data are centered around zero, newer techniques, such as rectified linear units (ReLU) and exponential linear units (ELU), have become prevalent because they offer non-linear behavior near zero [66,67].

A *fully connected layer* (also known as a dense layer) in CNN is connected to all neurons in the previous layer and next layer. It takes the output from the convolutional layers and applies a matrix multiplication operation, followed by a bias offset, to produce a final prediction for the image classification. The final layer in CNN is usually a fully connected layer, which produces a vector of values, each representing a probability for a given class (category that data sample belongs to). More details on CNN can be referred to in [8].

Augmentation Techniques: Improving the generalization of the model and mitigating the overfitting problem is a major challenge in the deep learning field, including in CNN models. Data augmentation is a method that addresses this challenge by improving the sufficiency and diversity of the training data. It is used to artificially expand the size of the training dataset by generating modified versions of images in the dataset, creating a more diverse and comprehensive representation of the real-world data. The goal is to make the model more robust to variations in the data and reduce overfitting by increasing the diversity of the training set. Common techniques include random rotation, scaling, flipping, cropping, and adding noise to the images. Additionally, data augmentation helps to reduce bias in the model by creating new data that helps the model better capture the nuances and variations in the real-world data. LeNet-5 [68] was one of the first CNN applications that utilized image data augmentations for handwritten digit classification. The authors in [69] presents various existing methods and promising developments of data augmentation in deep learning.

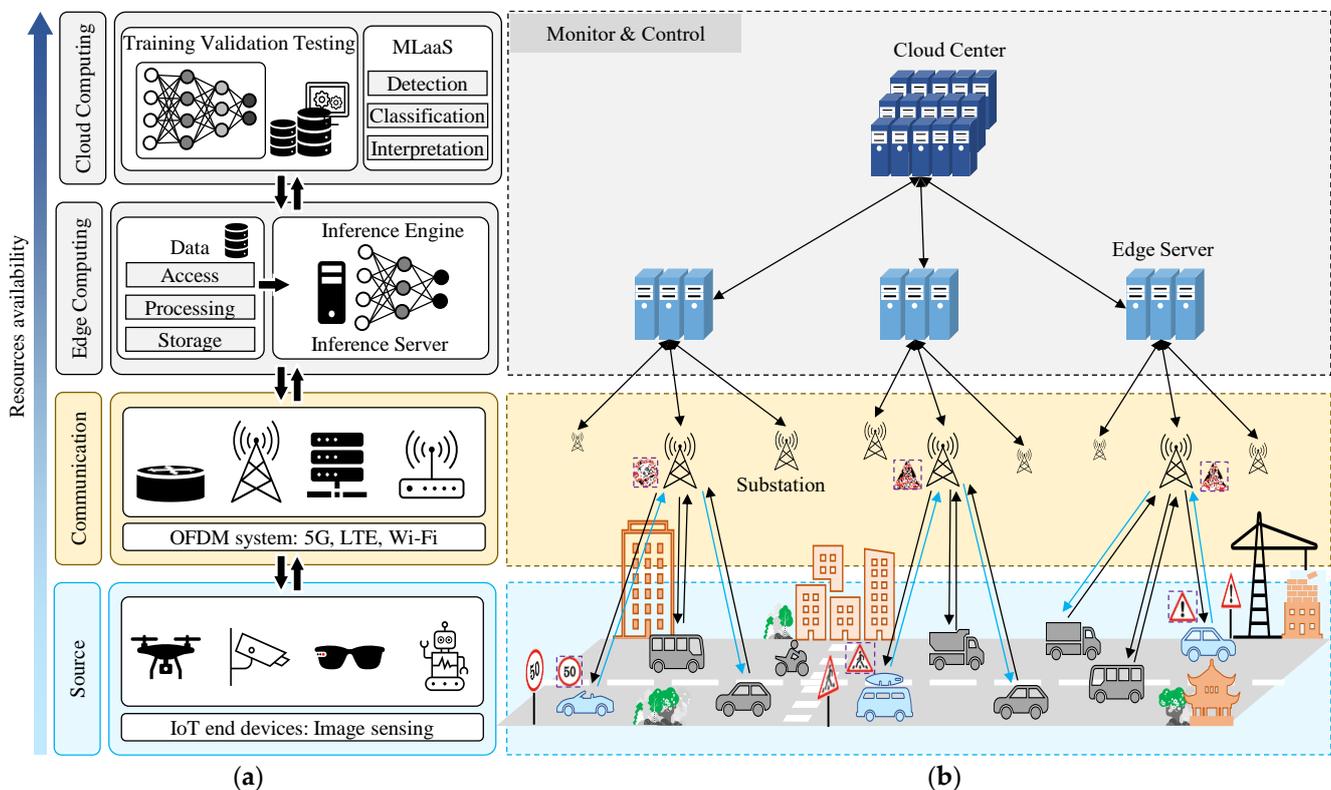
## 4. Results and Discussion

### 4.1. Case Study

The integration of advanced information and communication technologies along with Internet of Things (IoT) devices has brought a significant transformation to the Intelligent Transportation System (ITS) in smart cities. The incorporation of these technologies has resulted in synchronized transport networks, improved driving experience, optimized traffic management, and facilitated intelligent vehicular applications [1,2]. The image communication system lies at the core of the ITS, enabling applications that directly connect vehicles with traffic infrastructures and management systems [3]. One such application of traffic sign recognition in ITS was considered in this experiment. Traffic sign recognition is an essential CV application of an ITS environment and the collection and recognition of these signs are critical to ensure smooth traffic flow and prevention of accidents [3]. Effective traffic sign recognition in ITS requires both the collection of real-time images using high data rate communication systems, and the accurate recognition of these images using AI-enabled systems, respectively. In this experiment, the performance analysis of both these aspects was considered.

Furthermore, to effectively deploy AI-enabled system in an ITS environment for CV applications, a robust computing infrastructure is required for handling large amounts of

image data, processing it in real-time, and performing complex computations [70–72]. Edge servers and cloud servers are two potential solutions that can meet these requirements [72]. Edge servers are located close to the data source, and can process data locally, leading to faster processing, reduced latency, and improved reliability. Cloud servers, on the other hand, provide vast computing power and storage capacity, making them ideal for data-intensive tasks such as DL model training. By combining the computing power of edge and cloud servers, it is possible to process and analyze data in real-time while minimizing latency and reducing communication costs forming and edge–cloud collaboration [72]. Figure 4 shows such an infrastructure for the traffic sign recognition application in ITS, as considered in this experiment. Images are first transmitted to the edge server through various OFDM-based image communication systems, where they are introduced to noise and distortions from the system. The edge server then carries out inferencing on the received distorted images using a trained model, which were trained in the cloud server using a public dataset. The cloud center has several models that are task specific to enable Machine Learning as a Service (MLaaS) [72,73]. In this scenario, the edge server requested a model for classification task to carry out traffic sign recognition in the ITS environment.



**Figure 4.** An overview of used case scenario utilizing Edge–Cloud collaboration for deep learning-based CV applications in (a) generic IoT ecosystem and (b) ITS environment within a Smart City.

#### 4.2. Performance Analysis of Image Communication System

For the experiments we have considered OFDM communication system for image transmission. Within the system, we have considered higher order M-QAM modulation to attain a high data rate, at the cost of added noise in the images. Additionally, we have considered various channel models and channel estimation techniques to evaluate the noise and image quality under different system conditions. Multipath AWGN channel and Rayleigh Fading channel with AWGN were used as channel models to simulate real-world environmental conditions. Simple LS and more complex MMSE channel estimation were used to evaluate the trade-off between image quality and system complexity for

DL applications. Table 3 shows the simulation parameters of the OFDM-based image communication system.

**Table 3.** Simulation parameters of the image communication system.

Parameters	Values
Source Coding	Variable Length Coding
Modulation Technique	M-QAM
Modulation Order	$M \in \{16, 32, 64, 128, 256, 512, 1024\}$
Length of Symbol (k)	$\log_2(M)$
Subcarriers (S)	$n \times 2^n$
Cyclic Prefix (CP)	$s/4$
Pilots (P)	CP/4
Pilot Insertion	Comb-type
Channel Model	AWGN
	Rayleigh Fading with AWGN
Channel Estimation	LS
	MMSE
Channel Equalization	Zero-Forcing

#### 4.2.1. Performance Matrices

**BER:** The Bit Error Rate (BER) is a standard performance metric in digital wireless communication systems to measure the quality of the reconstructed signal at the receiver. It calculates the ratio of incorrect bits to the total number of bits transmitted, typically expressed as a ratio. The BER is a crucial measure in determining the maximum achievable data rate for a given system design, considering the presence of noise and other sources of interference in the channel.

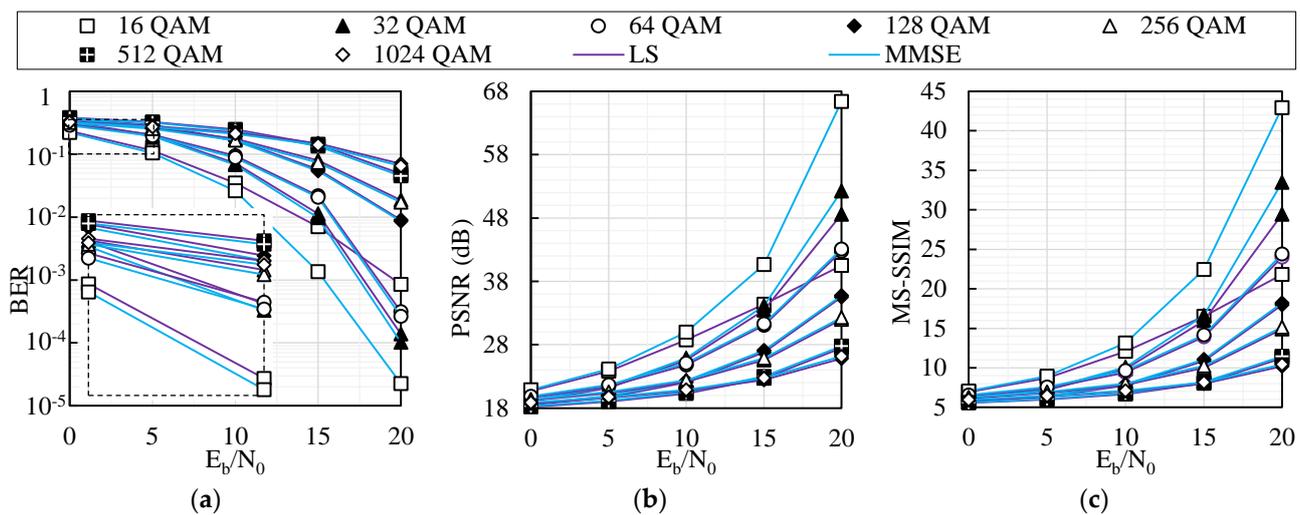
**PSNR:** Peak Signal-to-Noise Ratio (PSNR) is an important metric to evaluate the quality of recovered images in image communication systems. Unlike BER, which indicates the performance of the communication system in bit level, PSNR provides a measure of the perceptual quality of the images by comparing the received image with the original transmitted image. PSNR is a full-reference image quality measure that uses an objective approach based on explicit numerical criteria, such as comparisons with ground truth or prior knowledge expressed in terms of statistical parameters and tests. The PSNR value is measured in Decibels (dB) and approaches infinity as the mean square error (MSE) approaches zero, implying that a higher PSNR value indicates a higher-quality image. A low PSNR value, on the other hand, signifies significant numerical differences between the original and recovered images [74].

**MS-SSIM:** The MS-SSIM (multi-scale structural similarity index) is another full-reference image metric developed to compare the quality of an input image to that of a reference image with no distortion. Developed by Wang et al. [74], the MS-SSIM is correlated with the quality perception of the human visual system (HVS). It works by aggregating the inner similarity indexes obtained from multiple spatial scales (resolutions) to estimate the overall similarity between the input and reference images. MS-SSIM is an improved metric of single-scale SSIM (SS-SSIM), resolving the limitations of being only suitable for limited visual contexts and unable to account for the vast visual diversity [75]. MS-SSIM is also measured in dB and the values are usually represented in log scale to observe high quality results,  $-10\log_{10}(1 - M)$ , where  $M$  is the MS-SSIM score.

#### 4.2.2. Evaluation of Communication Systems over AWGN Channel

Figure 5a illustrates the BER in logarithmic scale versus  $E_b/N_0$  for the proposed OFDM-based image communication system over AWGN channel, using higher order M-QAM and different channel estimation techniques. The graph indicates that lower order M-QAM experiences a sharp drop in BER with higher  $E_b/N_0$ , characterized by the waterfall trend, whereas the drop in the higher order M-QAM is insignificant across all  $E_b/N_0$  values due to the increased number of error bits. At  $E_b/N_0$  20 dB, 16-QAM and

1024-QAM have reached their lowest mean BER of  $2.25 \times 10^{-5}$  and 0.05 using MMSE channel estimation, respectively. For channel estimation analysis, MMSE outperforms LS channel estimation across all M-QAM, and for 16-QAM at  $E_b/N_0$  20 dB, the use of MMSE resulted in the highest improvement (approximately 40 times lower BER) compared to LS channel estimation. Overall, the 16-QAM system using MMSE channel estimation has the lowest average BER and 1024-QAM system using LS estimation has the highest average BER.

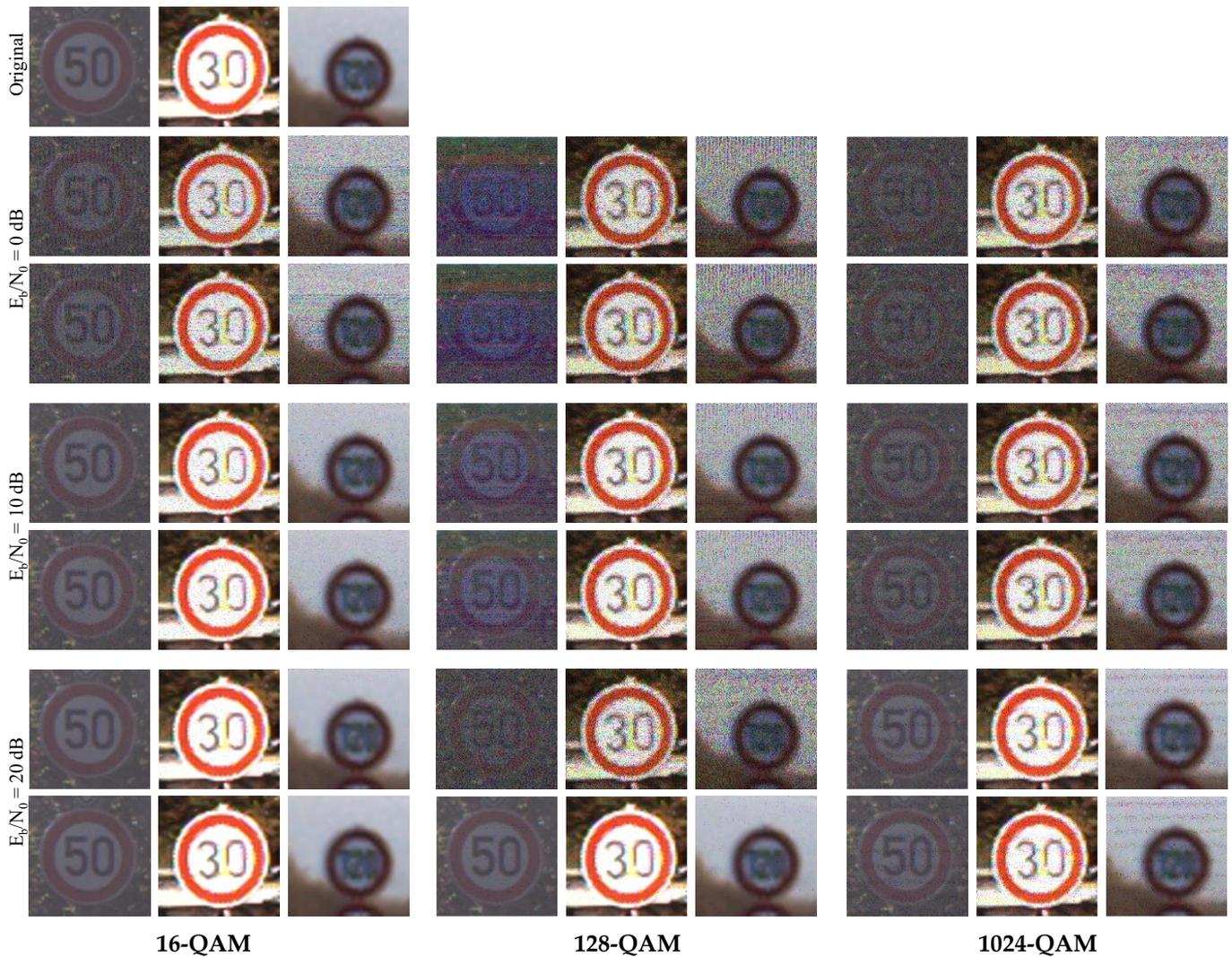


**Figure 5.** Performance analysis of various OFDM systems in terms of (a) BER, (b) PSNR, and (c) MS-SSIM with respect to various  $E_b/N_0$  under AWGN channel.

The image– quality measure in terms of PSNR of the recovered images for the proposed systems is shown in Figure 5b. The PSNR values follow a similar pattern to the BER and improve with higher  $E_b/N_0$ ; however, when using higher order M-QAM, the PSNR values decrease indicating the decline in image quality across all  $E_b/N_0$ . Additionally in high  $E_b/N_0$  regions, the quality of the image improves significantly for lower order M-QAM with a steep upward incline compared to higher order M-QAM, which has a more flattened incline. At  $E_b/N_0$  20 dB, 16-QAM and 1024-QAM have obtained their highest quality images with mean PSNR value of 66 dB and 26 dB using MMSE channel estimation, respectively. With respect to channel estimation, MMSE outperformed LS across all M-QAM, and 16-QAM at  $E_b/N_0$  20 dB achieved highest improvement of 26 dB when utilizing MMSE channel estimation; however with the higher order M-QAM, the improvement due to different channel estimation is less significant. MS-SSIM evaluation metrics (in logarithmic scale) have almost identical trends with PSNR, as illustrated in Figure 5c. For higher  $E_b/N_0$  the MS-SSIM improves significantly for lower order M-QAM, however, the improvement is slow for the higher order M-QAM. Lower order M-QAM has a higher MS-SSIM score and demonstrates significant improvement with increased  $E_b/N_0$ , while higher order M-QAM does not exhibit a significant improvement in the image quality across  $E_b/N_0$ . At  $E_b/N_0$  20 dB, 16-QAM and 1024-QAM have obtained their highest quality images with mean MS-SSIM values 43 dB and 10 dB using MMSE channel estimation, respectively. In terms of channel estimation, MMSE performs better than LS for all M-QAM (16-QAM at  $E_b/N_0$  20 dB achieved and improvement of 20 dB using MMSE); however, for higher order M-QAM, the difference in performance between the two techniques is not substantial. Overall, the 16-QAM system with MMSE channel estimation had the best image quality retention and 1024-QAM with LS had the worst image quality retention throughout the experiments in terms BER, PSNR, and MS-SSIM evaluation matrix.

For visual analysis, Figure 6 shows three example images from the dataset and the distortions from different communication systems when transmitted over AWGN channel.

From the visual inspection, the noise increases across higher order M-QAM; however, with increase in  $E_b/N_0$ , the image quality still improves significantly for lower order M-QAM (16-QAM) compared to higher order M-QAM (128-QAM and 1024-QAM). Although the images may be corrupted at lower  $E_b/N_0$ , the information in the region of interest (ROI) remains visible even with additional noise in the surrounding regions due to the way the traffic sign are designed with different color contrast [76,77]. In terms of different channel estimation techniques, there is no visible distinction between the LS and MMSE as the images are almost visually identical in comparison to the two channel estimation techniques.

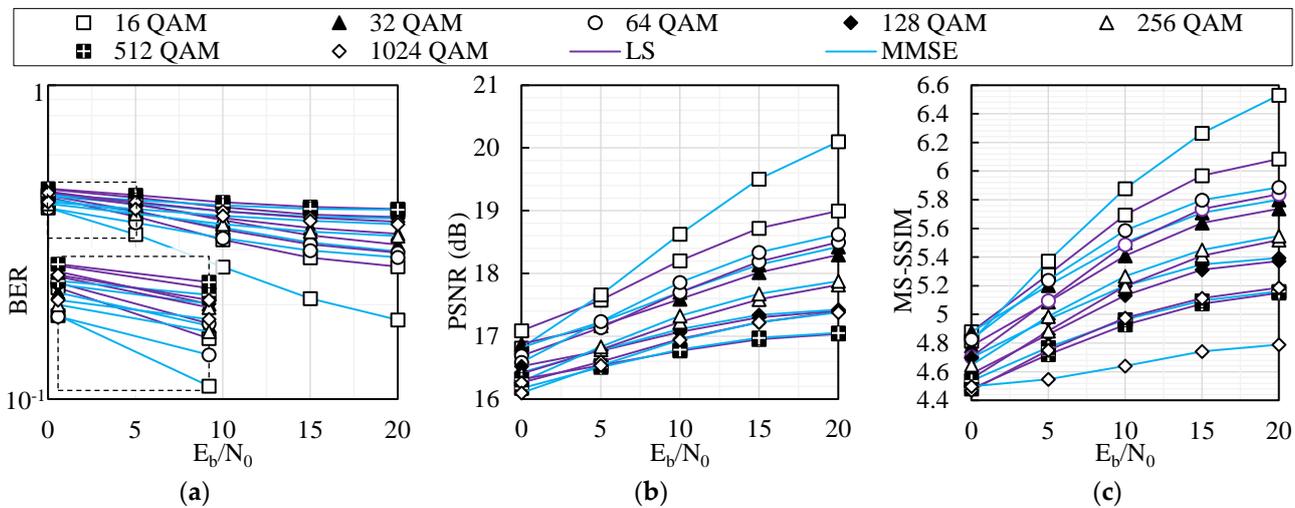


**Figure 6.** Sample recovered images from OFDM system over AWGN channel. The top row of each  $E_b/N_0$  represents received images from system using LS channel estimation, whereas the bottom row images are from systems using MMSE channel estimation.

#### 4.2.3. Evaluation of Communication Systems over Rayleigh Fading Channel

The BER performance of the OFDM-based image communication system with higher order M-QAM modulation and various channel estimation techniques over the Rayleigh Fading channel is presented in Figure 7a. The BER is presented on a logarithmic scale and is plotted against the  $E_b/N_0$  for the proposed systems. From the graph we can see that there is no significant improvement in the BER across  $E_b/N_0$  compared to the systems over simple AWGN channel as discussed in Section 4.2.2, this is due to heavy noise and distortions introduce by the fading effect of the Rayleigh Fading channel. The lowest BER was recorded with 16-QAM system utilizing MMSE channel estimation, achieving values

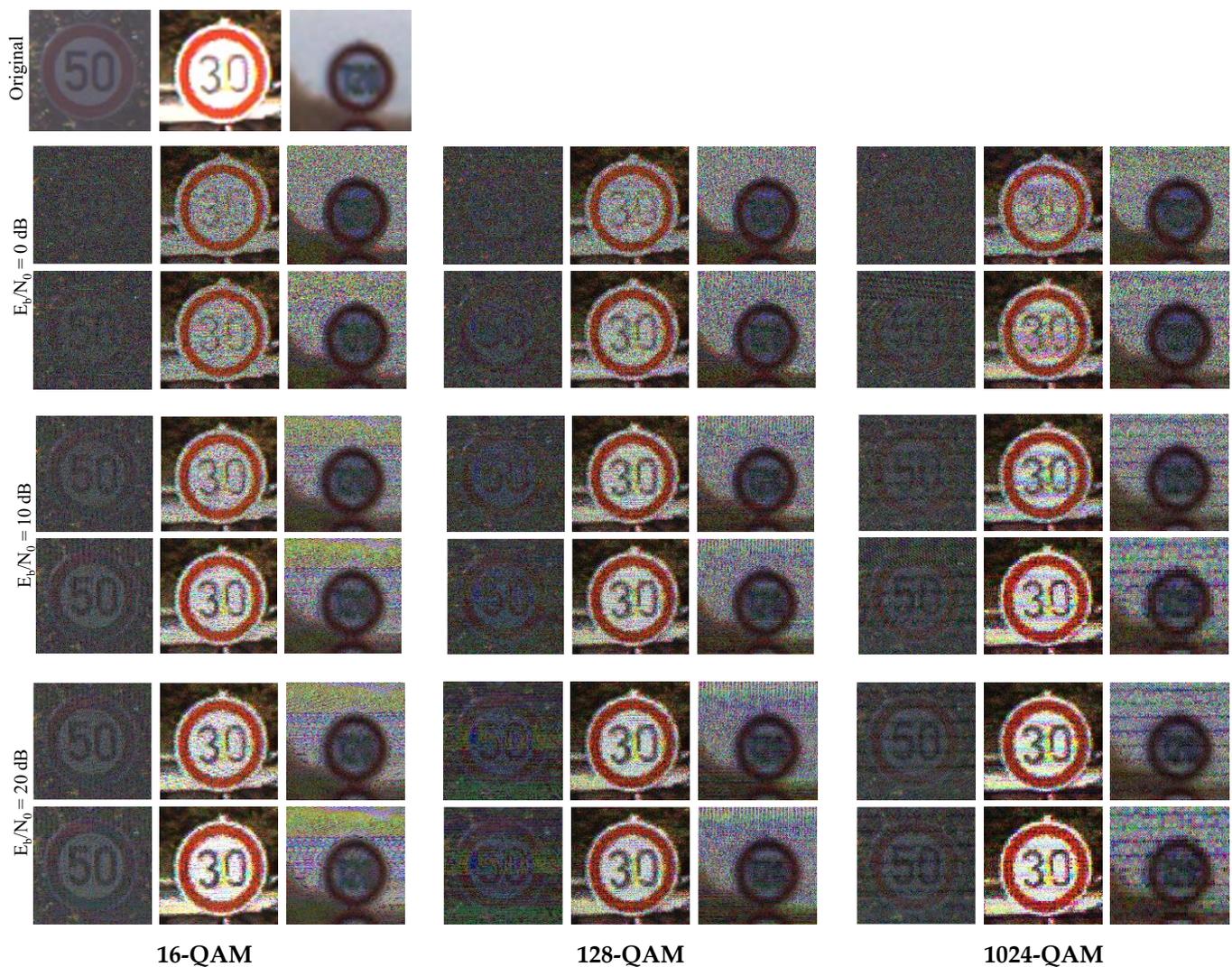
of 0.41 to 0.18 across  $E_b/N_0$  0 dB to 20 dB, respectively. The highest BER was obtained by 512-QAM with only a small drop from 0.47 to 0.40 across the  $E_b/N_0$ . In terms of channel estimation, MMSE performed better than LS for all M-QAM throughout the experiment and the difference in performance was significant at the lower  $E_b/N_0$  region due to its prior channel knowledge. Additionally, 16-QAM leveraged the greatest improvement in BER (difference of 0.1 at  $E_b/N_0$  20 dB) when using MMSE channel estimation, however, the difference in the performance was insignificant for rest of the M-QAM with respect to different channel estimation techniques. Overall, the BER increases with a higher order of M-QAM and MMSE channel estimation performed better than LS channel estimation throughout the experiment. 16-QAM has the lowest average BER using MMSE channel estimation and 512-QAM has the highest average BER using LS channel estimation across  $E_b/N_0$ .



**Figure 7.** Performance analysis of various OFDM systems in terms of (a) BER, (b) PSNR, and (c) MS-SSIM with respect to various  $E_b/N_0$  under the Rayleigh Fading channel.

The PSNR evaluation on the received images is shown in Figure 7b and it shows a similar pattern to BER. The PSNR value improves with higher  $E_b/N_0$ , however it decreases with higher order M-QAM across all  $E_b/N_0$  due to increased error bits that subsequently resulted in lower image quality. The 16-QAM and 1024-QAM systems demonstrated their best image quality using MMSE channel estimation at  $E_b/N_0$  20 dB, with average PSNR values of 20 dB and 17 dB, respectively. In terms of channel estimation, in contrast to the BER analysis, the PSNR value of images from systems using LS was better than MMSE from  $E_b/N_0$  0 dB to 5 dB; however, after  $E_b/N_0$  5 dB, all the M-QAM systems performed better with MMSE channel estimation, and the highest quality improvement of the image due to different channel estimation was observed by the 16-QAM system (improvement of 1 dB). Throughout the experiments, the 16-QAM system with MMSE channel estimation maintained the highest level of image quality retention (except at  $E_b/N_0$  0 dB), while the 512-QAM system with LS estimation exhibited the lowest image quality. Like PSNR, the MS-SSIM values exhibit some improving trends with higher  $E_b/N_0$  but show a decline in image quality for higher order M-QAM as shown in Figure 7c. The 16-QAM and 1024-QAM systems had the highest MS-SSIM score at  $E_b/N_0$  20 dB, with scores of 6.5 and 5.18 while using MMSE and LS channel estimation, respectively. In terms of channel estimation, all the M-QAM systems performed better with MMSE channel estimation across  $E_b/N_0$  5 dB to 20 dB, except for 1024-QAM, where MMSE channel estimation showed significant drop compared to LS. Overall, the 16-QAM system with MMSE channel estimation maintained the highest MS-SSIM score, while the 1024-QAM system with MMSE estimation had the lowest throughout the experiment.

Figure 8 shows recovered images from communication system over Rayleigh Fading channel. Compared to the visual inspection of images from AWGN channel discussed in Section 4.2.2, the images from the Rayleigh Fading channel are heavily distorted and most of the ROI information at the lower  $E_b/N_0$  are not clearly visible. The amount of noise on the images decreases slightly with higher  $E_b/N_0$ , and that is sufficient to make the information on the ROI visible due to the robust design of the traffic signs with different color contrast [76,77]. Therefore, for higher  $E_b/N_0$ , the images may be heavily distorted but the information in the ROI (speed limit) is still slightly visible, and the DL models can exploit that and extract the features from the ROI to predict the signs correctly, which will be further discussed in Section 4.4. Another trend that is observed is that the images from the M-QAM systems with odd length of symbol (128-QAM) have reduced the brightness compared to the even length of the symbol, which retained brightness the same as the original image. In terms of channel estimation, the images at  $E_b/N_0$  0 dB from system utilizing MMSE channel estimation (bottom row) is clearer compared to LS channel estimation (top row) across the M-QAM, where the information in the ROI is not visible at all for the LS channel estimation. Therefore, it indicates that MMSE retains better quality images for lower  $E_b/N_0$  compared to LS in terms of human visual perspective.



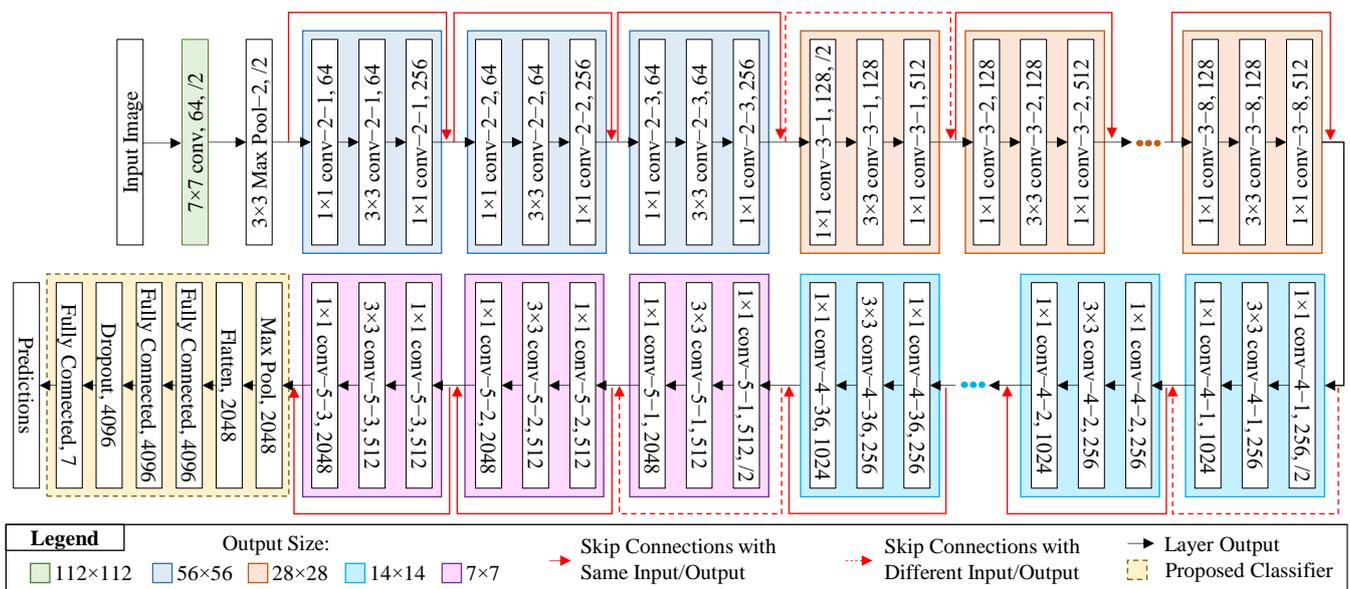
**Figure 8.** Sample recovered images from OFDM system over Rayleigh Fading channel. The top row of each  $E_b/N_0$  represents received images from system using LS channel estimation, whereas the bottom row images are from systems using MMSE channel estimation.

### 4.3. Robustness of Deep-Learning Models under Various Communication Systems

In this section we discuss the robustness of DL models on images recovered from different communication systems presented in Section 4.2. We briefly describe the CNN models utilized in the experiment and the motivation of the choice, followed by the robustness analysis of the individual models with and without augmentation techniques under the influence of the different communication systems.

#### 4.3.1. CNN-based Models Used in the Experiment

ResNet152V2 is a CNN model belonging to the ResNet family, commonly used for image classification tasks in computer vision applications [78,79]. It is a more advanced and intricate version of the ResNet model with a total of 152 layers and utilizes skip connections (referred to as residual connections) which make it easier for the model to learn from lower layers and improve overall performance. ResNet152V2 employs a bottleneck design, which reduces the number of parameters required in each layer while maintaining the same level of representation power, making it a more efficient model compared to the earlier models in the series. Additionally, ResNet152V2 uses better weight initialization to prevent the vanishing and exploding gradient problems, and normalization approaches, by combining batch normalization and weight normalization, to improve the stability and speed of training [78,79]. ResNet152V2 also utilizes different training techniques, such as augmentation and stochastic depth to improve the model’s generalizability, making it more potent and reliable for image classification tasks compared to other models in the family [78,79]. Overall, ResNet152V2 performs better than its predecessor and other CNN models in terms of accuracy while requiring fewer parameters. However, it is a very deep and complex model with many layers which demands significant computational resources. These demands can be fulfilled by edge-cloud collaboration as discussed in Section 1. Therefore, for our experiment we have considered ResNet152V2 architecture for feature extraction with a proposed classifier for the classification task, as shown in Figure 9, and parameters of the model are mentioned in Table 4.



**Figure 9.** Illustration of ResNet152-based proposed deep learning model for traffic sign recognition in ITS. For a layer ‘1 × 1 conv–2–1, 64’, 1 × 1 is the filter size, 2–1 is module number, and 64 is the number of filters. The ellipses show that the blocks are repeated.

**Table 4.** Parameters of the DL models used in the experiments.

	ResNet152V2	EfficientNetV2-B0
Model Input Size:	224 × 224	
Top Layers:	Max Pool, 2048	Average Pool, 1280
	Flatten, 2048	Flatten, 1280
	Fully Connected, 4096 *	Fully Connected, 256 *
	Fully Connected, 4096 *	Fully Connected, 7 +
	Dropout, 40%	
	Fully Connected, 7 +	
Total parameter:	83,534,343	6,249,047
Trainable parameter:	83,390,599	6,188,439
Non-trainable parameter:	143,744	60,608

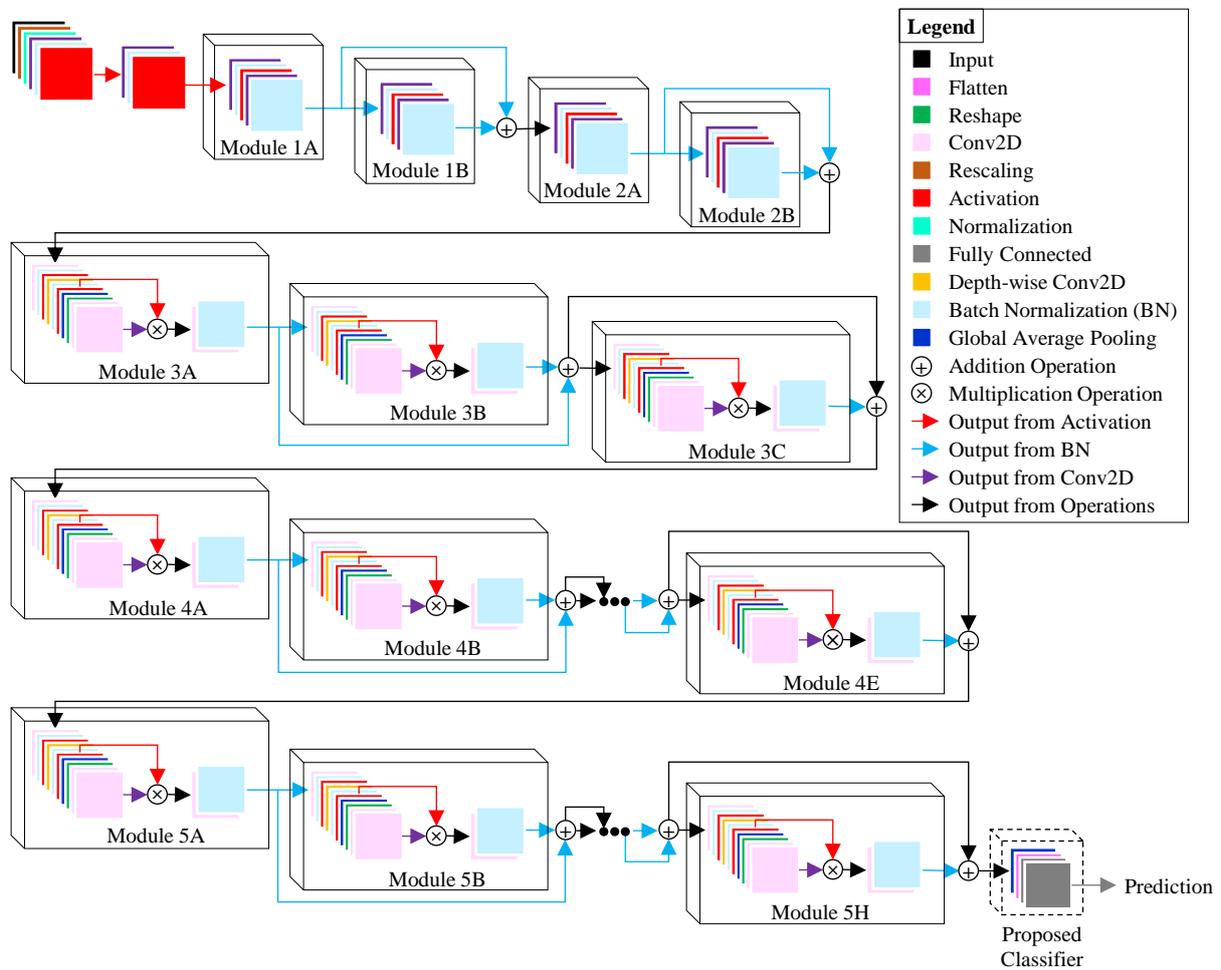
\* ReLU Activation + Softmax Activation.

EfficientNet [80] is a family of CNN models that are designed to achieve state-of-the-art accuracy with highly efficient use of computational resources. Since its inception, it has become popular for computer vision applications due to its ability to achieve high accuracy on image classification tasks while using fewer parameters and FLOPs (floating-point operations) than other popular CNN models. EfficientNet utilizes neural architecture search (NAS) to design the baseline model, EfficientNet-B0, which has a better trade-off between parameters and accuracy. The model is then uniformly scaled up in terms of depth, width, and resolution to obtain a family of models, ranging from EfficientNet-B0 (the smallest model) to EfficientNet-B7 (the largest model). Despite using depth-wise convolution to achieve superiority in terms of the number of parameters and FLOPs, EfficientNetV1 had limitations in fully utilizing accelerators, which limited its training and inference speed. EfficientNetV2 [81] mitigates these limitations while ensuring parameter efficiency. It proposes three solutions, first, to adjust the size and regularization progressively during training; second, a non-uniform scaling strategy to add more layers in later stages; and third, the proposed Fused-MBConv in the early stage to improve training speed (introducing a small overhead on parameters and FLOPs). Therefore, in our experiment we have used EfficientNetV2-B0 to achieve state-of-the-art accuracy while ensuring parameter efficiency using the smallest model (B0) in the family and leveraging faster training and inference time (V2). The EfficientNetV2-B0 architecture (adapted from [81,82]) with our proposed classifier is shown in Figure 10 and the parameters are mentioned in Table 4.

For this study, we have considered the GTSRB dataset with speed limit signs of seven different classes, 30, 50, 60, 70, 80, 100, and 120. The dataset was divided into 80% for training, 10% for validation, and 10% for testing. For training, Stochastic Gradient Descent (SGD) optimizer and cross entropy-loss was used for both models, with learning rates ranging from 0.01 to 0.000001 and 0.1 to 0.0001 for ResNet152V2 and EfficientNetV2-B0, respectively, using the reduce learning rate (ReduceLR) technique. Furthermore, various augmentation techniques, such as 20-degree rotation, horizontal and vertical shift with factor 0.1, nearest fill mode, and 0.25 to 1.25 factor zoom and brightness, were applied to achieve high validation accuracy during training (as discussed in Section 3.2).

The accuracy of a DL model is commonly evaluated using three metrics, training accuracy, validation accuracy, and test accuracy. The training accuracy reflects the model's ability to accurately predict data that were used for training. The validation accuracy assesses the model's ability to generalize to new data during training, while the test accuracy measures the model's performance on unseen data after training. In the experiment, ResNet152V2 achieved 98% training accuracy, and EfficientNetV2-B0 achieved 100% training accuracy. The training accuracy is comparable to related classification study one GTSRB dataset using ResNet152 [83] and EfficientNetV2 [84] family which achieved 96% and 98%, respectively. The validation accuracy of the models showed significant difference with and without applying augmentation techniques in the experiment. ResNet152V2 achieved 84% validation accuracy without augmentation, indicating overfitting and poor generalization; however, when the augmentation techniques were applied, the validation

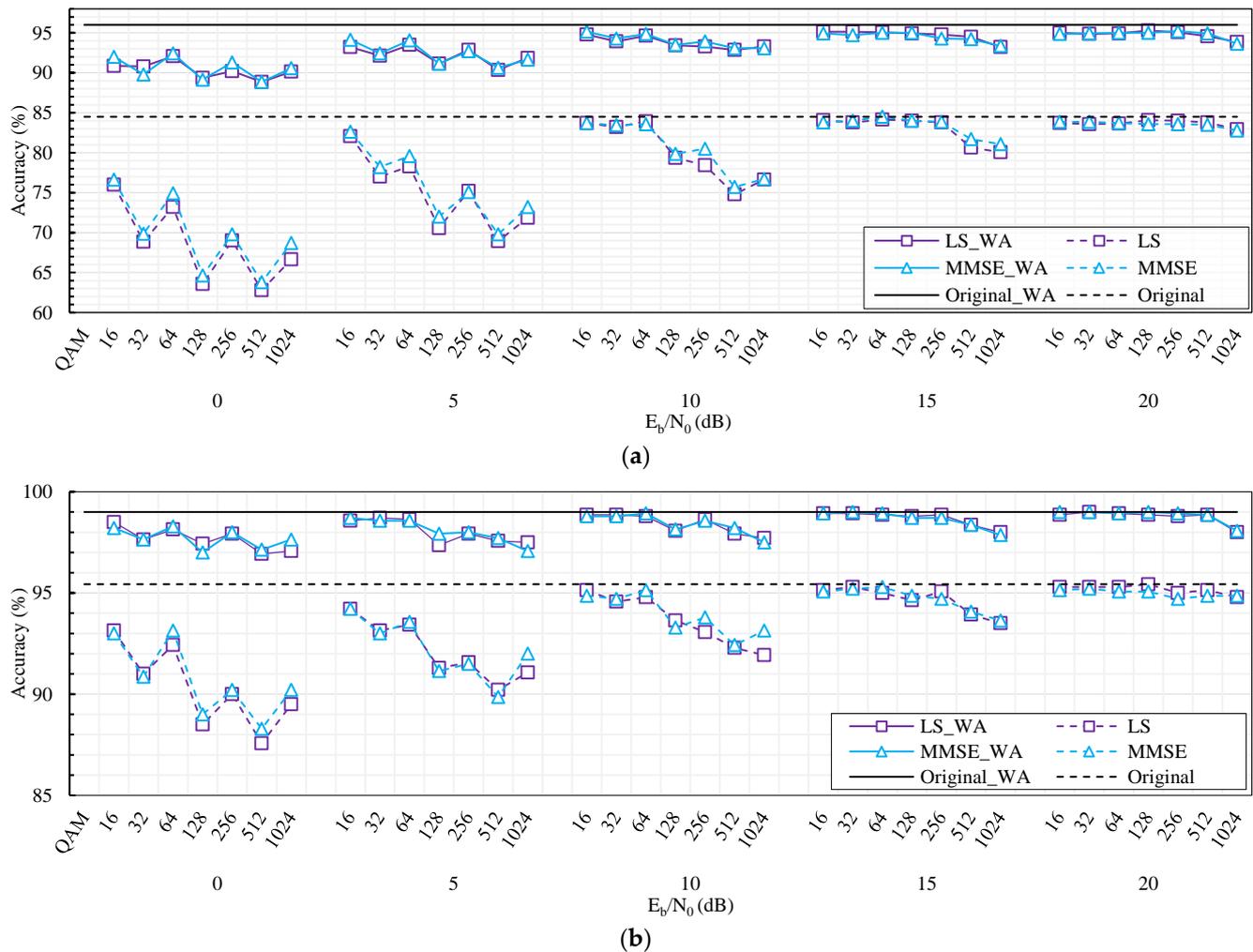
accuracy improved to 96%. Similarly, EfficientNetV2-B0 achieved 95% validation accuracy without augmentation, which improved to 99% with augmentation. Both the training and validation of the two models were completed on pristine (original) images. The inference (test) accuracy of the models on noisy images from different communication systems is further discussed.



**Figure 10.** Illustration of EfficientNetV2-B0-based proposed deep learning model for traffic sign recognition in ITS.

#### 4.3.2. Performance Analysis of DL Models over AWGN Channel

The DL model performance in terms of classification accuracy on the reconstructed images from various communication systems over AWGN channel is shown in Figure 11. The dotted lines illustrate the model accuracy achieved without applying any augmentation techniques during training, whereas the solid lines represent the accuracy achieved with augmentation techniques (WA) applied during training. The accuracy of the DL model corresponds to the mean value across all inferring images for the specific M-QAM and  $E_b/N_0$ .



**Figure 11.** DL model performance of (a) ResNet152V2 and (b) EfficientNetV2-B0 on recovered images with respect to  $E_b/N_0$  for various OFDM systems under AWGN channel. For order of M-QAM,  $M \in \{16, 32, \dots, 1024\}$  across each  $E_b/N_0$ .

Performance analysis of ResNet152V2 over AWGN channel: When augmentation techniques are not applied during the training of ResNet152V2 model, the accuracy achieved on the clean (original) images is 84%, and the overall accuracy of the model is low, as shown in Figure 11a. This is due to the model not being able to learn and generalize well without augmentation techniques. From the graph, we can observe that the accuracy of the model decreases significantly with higher order M-QAM in the lower  $E_b/N_0$  regions; however, for the higher  $E_b/N_0$  regions the model is able to generalize well and achieves almost same accuracy as it performed on the original images across all M-QAM (except 512-QAM and 1024-QAM). We can see a clear distinction between the accuracy of the model on images from M-QAM systems with odd length of symbol and even length of symbol with a zig-zag pattern in the lower  $E_b/N_0$  regions. There is a sharp drop in the accuracy of the model on images from M-QAM systems with odd length of symbol (32-QAM, 128-QAM, and 512-QAM) which is due to the increased error bits, as discussed in Section 3.1.3. The highest difference in accuracy on images across different M-QAM was observed at  $E_b/N_0$  0 dB, with 14% difference between highest accuracy (16-QAM) and lowest accuracy (512-QAM), this difference is less than 2% for  $E_b/N_0$  20 dB. For channel estimation, accuracy was better on systems using MMSE channel estimation; however, the improvement is insignificant. Overall, the lowest accuracy observed without augmentation is 63% on images from 512-QAM system using LS channel estimation at  $E_b/N_0$  0 dB. On the other hand, images from

16-QAM to 256-QAM achieved a highest accuracy of 84% (same as on the original images) for  $E_b/N_0$  15 dB and 20 dB.

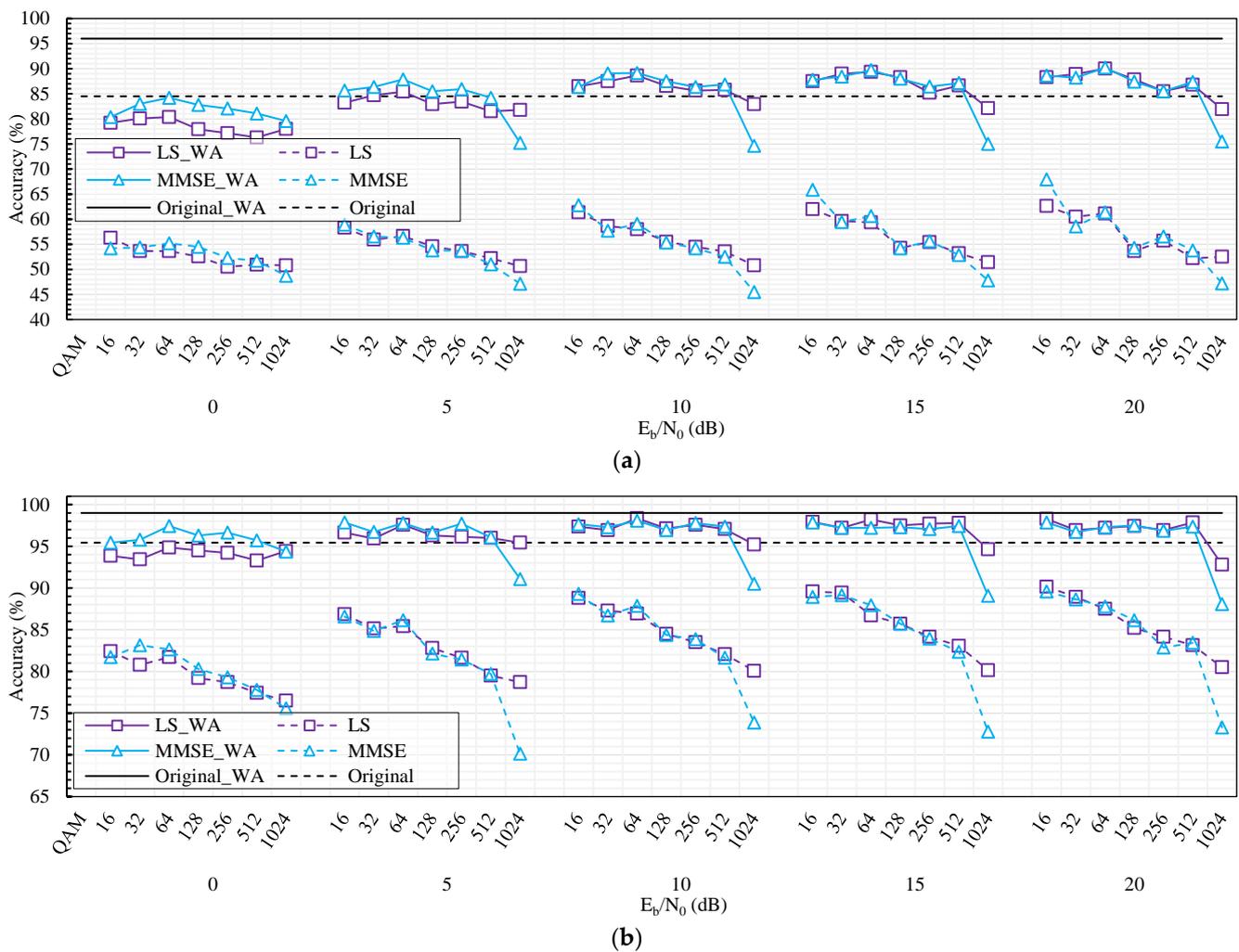
When augmentation techniques were applied, the ResNet152V2 was able to learn better and the accuracy of the model on clean (original) images improved to 96%. The impact of M-QAM with odd and even length of symbol is still observed, however the overall accuracy of the model improved significantly across all M-QAM. There is no drastic drop in accuracy across higher order M-QAM and the difference in the accuracy is less than 4% for all  $E_b/N_0$ . This is due to the model being able to generalize well across all M-QAM as it was trained with different augmentation techniques. Accuracy with respect to channel estimation techniques has overlapping patterns, indicating that there is no significant impact when using more complex channel estimation (such as MMSE) on the accuracy of the DL model. With augmentation, the lowest accuracy obtained was 89% for 512-QAM system at  $E_b/N_0$  0 dB; however, beyond  $E_b/N_0$  10 dB the accuracy of the model was greater than 93% across all M-QAM and different channel estimation techniques.

Performance analysis of EfficientNetV2-B0 over AWGN channel: In contrast to ResNet152V2, EfficientNetV2-B0 achieved 95% accuracy on the clean (original) images even without applying any augmentation techniques during training as shown in Figure 11b, this is due to its better learnability, as discussed in Section 4.3.1. However, there is a drop in the accuracy of models on images across higher order M-QAM. At  $E_b/N_0$  0 dB, the highest drop in accuracy of 6% was observed on images from 16-QAM (highest accuracy) to 512-QAM (lowest accuracy); however, the difference in accuracy across M-QAM systems becomes smaller with increase in  $E_b/N_0$ . For  $E_b/N_0$  20 dB the accuracy curve almost flattens (indicating no difference in accuracy across higher order M-QAM), and the accuracy was almost same as it was on the original images. This indicates that the model can generalize the noise on the images across all systems and provide better accuracy even for higher order M-QAM systems.

Furthermore, applying augmentation techniques during training drastically improved the EfficientNetV2-B0 model and it was able to predict almost all the clean images correctly, achieving 99% accuracy. The difference in accuracy on images from higher order M-QAM is less than 2% with only little distinctions between M-QAMs with odd and even length of symbol. When applying augmentation, the lowest accuracy obtained was 97% on images from 512-QAM at 0 dB. However, beyond  $E_b/N_0$  10 dB, the model accuracy on images from most of the systems was same as it performed on the original images (except 128-QAM at 10 dB, 512-QAM and 1024-QAM at 15 dB, and 1024-QAM at 20 dB). Throughout the experiment, the accuracy from system with different channel estimation had overlapping patterns and there were no clear indications that complex channel estimation (such as MMSE) improves the model accuracy. In comparison to the two DL models, EfficientNetV2-B0 outperformed ResNet152V2 across all image communication systems, regardless of whether augmentation was applied or not, this is due to its robustness as discussed in Section 4.3.1.

#### 4.3.3. Performance Analysis of DL Models over Rayleigh Fading Channel

In the evaluation of communication systems and image quality discussed in Section 4.2, we have observed severe degradation in performance when the Rayleigh Fading channel is utilized, as compared to employing only a simple AWGN channel model. We can see a similar trend in the accuracy of the DL models; however, when the DL models utilized augmentation techniques during training, they have achieved favorable results even on the Rayleigh Fading channel, as shown in Figure 12.



**Figure 12.** DL model performance of (a) ResNet152V2 and (b) EfficientNetV2-B0 on recovered images with respect to  $E_b/N_0$  for various OFDM systems under the Rayleigh Fading channel. For order of M-QAM,  $M \in \{16, 32, \dots, 1024\}$  across each  $E_b/N_0$ .

Performance analysis of ResNet152V2 over Rayleigh Fading Channel: When using ResNet152V2 without applying augmentation techniques during training, the accuracy on the images from the Rayleigh Fading channel achieved very low accuracy, as shown in Figure 12a. The accuracy of the DL model dropped on images from higher order M-QAM systems and there was no significant improvement with higher  $E_b/N_0$ , in fact the difference in accuracy on the images from lower order M-QAM and higher order M-QAM was more significant in the higher  $E_b/N_0$  region. The difference in accuracy on images from 16-QAM (highest accuracy) to 1024-QAM (lowest accuracy) was 8% at  $E_b/N_0$  0 dB; however, the difference was 21% at  $E_b/N_0$  20 dB. This is because the image quality in lower order M-QAM improved more significantly compared to those in higher order M-QAM with the increase in  $E_b/N_0$ . Additionally, it also indicates that the model was not able to generalize well on the images from different M-QAM systems. The highest accuracy obtained without augmentation was 68% at  $E_b/N_0$  20 dB for 16-QAM system with MMSE channel estimation and the lowest accuracy of 46% was obtained at  $E_b/N_0$  10 dB on images from the 1024-QAM system with MMSE channel estimation. For analysis on different channel estimations, there was no significant difference in the improvement in accuracy (except for 16-QAM and 1024-QAM) due to different channel estimation, as it had overlapping trends throughout the experiment. Images from 16-QAM system using MMSE channel estimation achieved almost 5% better accuracy compared to LS at  $E_b/N_0$  15 dB and 20 dB; however, images from

1024-QAM achieved almost same improvement in accuracy using LS channel estimation compared to MMSE for  $E_b/N_0$  higher than 5 dB.

The ResNet152V2 model trained using augmentation techniques performed much better even on the multipath Rayleigh Fading channel. Overall, the model accuracy improved, and after  $E_b/N_0$  10 dB the model accuracy was greater than 86% across all systems (except 1024-QAM). The highest accuracy of 90% was obtained by 64-QAM at  $E_b/N_0$  15 dB and 20 dB and the lowest accuracy of 75% was obtained at  $E_b/N_0$  10 dB on images from 1024-QAM system, both using MMSE channel estimation. Overall, the difference in accuracy on images across higher order M-QAM was less than 4% (except 1024-QAM) which indicates the model was able to generalize well on images even from higher order M-QAM system. In terms of the effect of channel estimation techniques, there were distinct differences in the accuracy of the model on images from systems utilizing different channel estimation techniques in the lower  $E_b/N_0$  regions. From  $E_b/N_0$  0–10 dB the accuracy on images from systems using MMSE channel estimation was greater compared to LS channel estimation, and at  $E_b/N_0$  0 dB this difference was the highest with almost 5% improvement in accuracy for 64-QAM, 128-QAM, 256-QAM and 512-QAM. The gap gradually decreased for  $E_b/N_0$  higher than 10 dB, and an overlapping pattern (except 1024-QAM) of accuracy on images from systems utilizing both channel estimation techniques was observed.

Performance analysis of EfficientNetV2-B0 over Rayleigh Fading Channel: Like ResNet152V2, the accuracy of EfficientNetV2-B0 also decreases on images from systems using the Rayleigh Fading channel compared to simple AWGN channel, as shown in Figure 12b. Without applying augmentation techniques during training, the model was not able to generalize well, and the accuracy of the models decreases significantly with higher order M-QAM. There is a slight improvement in accuracy of the model on images from lower order M-QAM with the increase in  $E_b/N_0$ ; however, accuracy on images from higher order M-QAM shows very little improvement. Without augmentation, the highest accuracy of 90% was obtained on images from the 16-QAM system at  $E_b/N_0$  20 dB and the lowest accuracy of 70% was obtained on images from 1024-QAM at  $E_b/N_0$  5 dB, using MMSE channel estimation. There was no significant difference in improvement by using different channel estimation techniques, except on images from the 1024-QAM system, where the model on images from system utilizing LS obtained approximately 8% greater accuracy compared to MMSE at  $E_b/N_0$  greater than 5 dB. The model performed slightly better on systems utilizing MMSE for 32-QAM, 64-QAM, 128-QAM, and 512-QAM at  $E_b/N_0$  0 dB, however this improvement is insignificant considering the overall performance throughout the experiment.

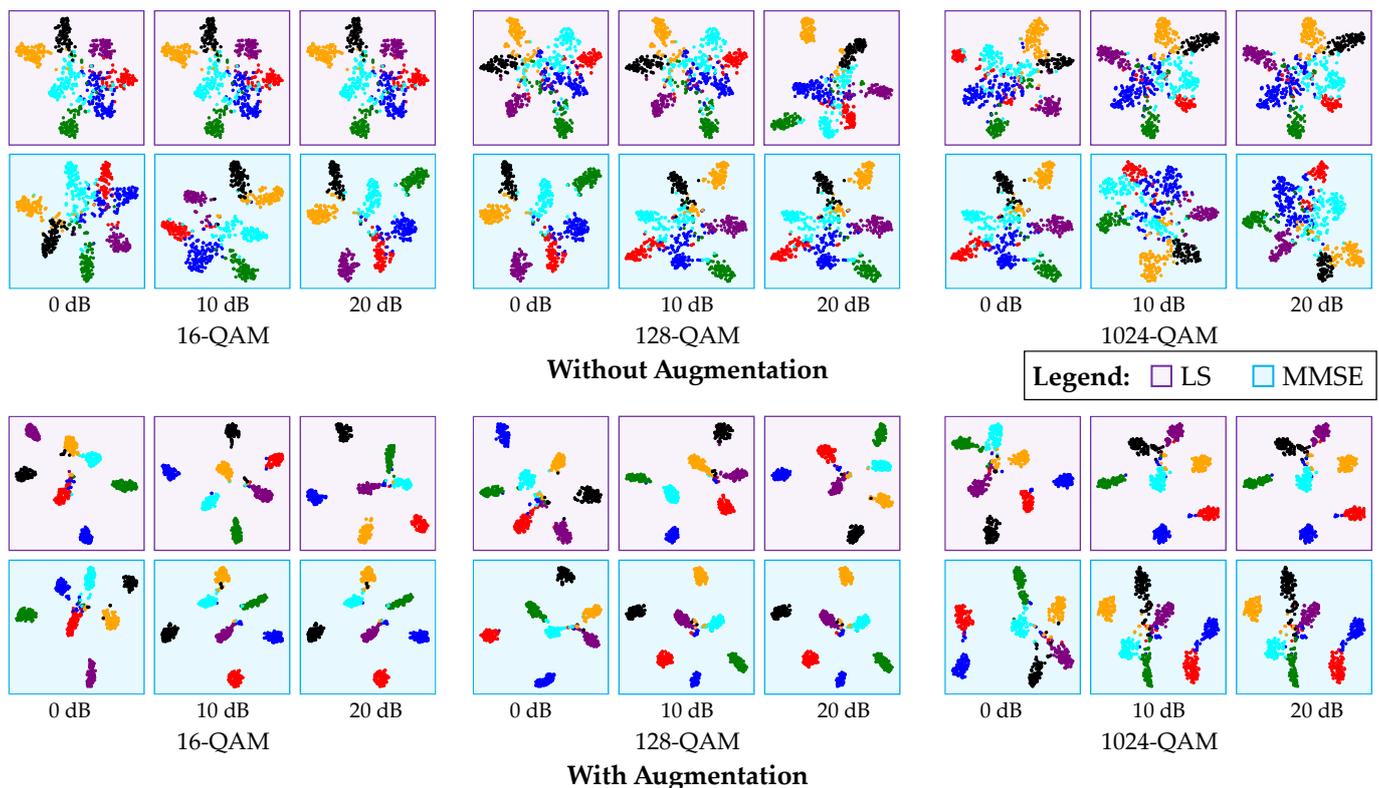
The EfficientNetV2-B0 model trained with augmentation techniques had drastic improvement across all systems. The model could generalize well since the difference in accuracy across M-QAM systems is less than 2% (except 1024-QAM) at all  $E_b/N_0$ . Throughout the experiment, the accuracy of the model was greater than 95% across all systems (except 1024-QAM). The highest accuracy of 99% was obtained on images from 16-QAM system at  $E_b/N_0$  20 dB, whereas the lowest accuracy of 88% was obtained on images from 1024-QAM system at  $E_b/N_0$  20 dB, both using MMSE channel estimation techniques. The improvement in accuracy on images from systems using different channel estimation was observed only at  $E_b/N_0$  0 dB, where accuracy was approximately 4% better on images from systems using MMSE compared to LS across all systems (except 16-QAM and 1024-QAM). However, at higher  $E_b/N_0$  regions, different channel estimations did not have a significant impact on the accuracy, except for 1024-QAM, where accuracy on images from a system using LS channel estimation had approximately 5% greater accuracy compared to MMSE. Which indicates that the effect of using complex channel estimation (such as MMSE) improves the model accuracy at lower  $E_b/N_0$  regions; however, it can have contrasting effect at higher  $E_b/N_0$  regions and higher order M-QAM.

Comparing the performance of the two models, EfficientNetV2-B0 had greater accuracy on images from systems over the Rayleigh Fading channel, compared to ResNet152V2.

The accuracy of the EfficientNetV2-B0 model trained using augmentation was from 88 to 99%, whereas for ResNet152V2 it was between 75% and 90% across all systems.

4.4. Discussion

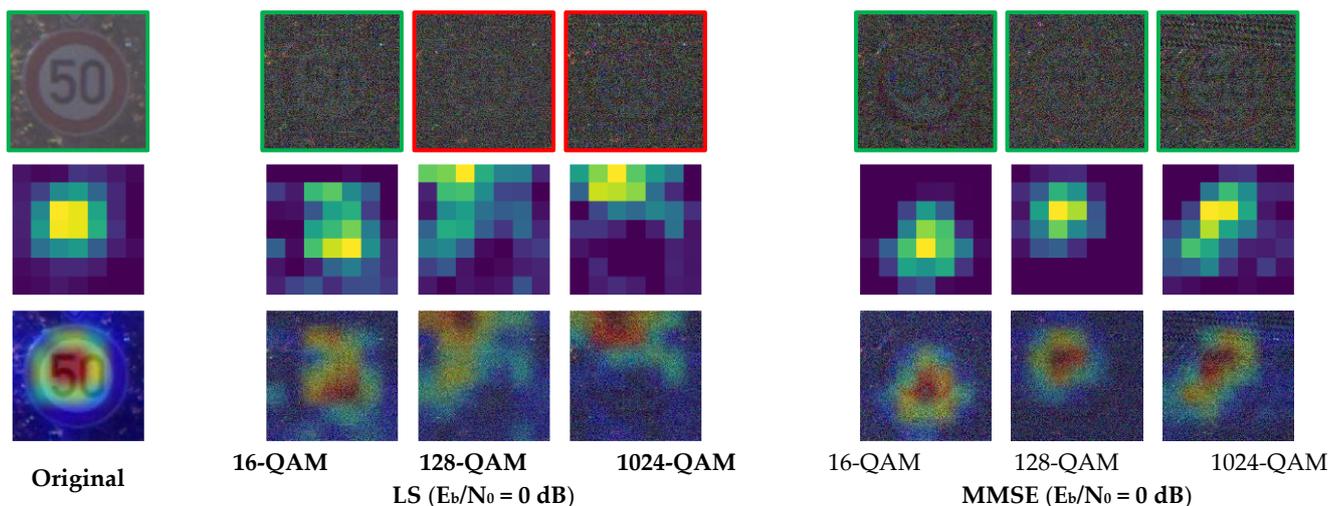
The generalizability of the deep learning model across the M-QAM system using different channel estimation can be further observed using the t-distributed stochastic neighbor embedding (t-SNE) [85] visualization algorithm, as shown in Figure 13. It is a probabilistic dimensionality reduction algorithm which maps high-dimensional feature points into low-dimensional feature space (typically 2D or 3D) while preserving the similarities. Features close to each other in the high-dimensional space are mapped closer to each other in the lower-dimensional space with high probability. Points that are close together in the t-SNE plot are likely to have similar features or attributes (in terms of color, texture, or shape), whereas points that are far apart are likely to be dissimilar. From Figure 13, it can be observed that the same class points from the DL model trained with augmentation are close together and do not have many overlapping points of different class (color) across  $E_b/N_0$  0 dB, 10 dB, and 20 dB for the M-QAM systems. Whereas points from the DL model trained without augmentation are more scattered and overlap with other classes. There are no significant differences in class separability for different channel estimation used across the M-QAM systems. Overall, DL model trained with augmentation provides better separability between the seven classes and their features, resulting in better classification performance across all communication systems compared to DL model trained without augmentation.



**Figure 13.** Two-dimensional t-SNE plot showing feature space analysis of EfficientNetV2-B0 model on recovered images across different OFDM-based image communication systems under the Rayleigh Fading channel. Each color of the dots represents one of the seven classes chosen from the GTSRB dataset.

Gradient-weighted Class Activation Mapping (Grad-CAM) heat map is another visualization algorithm that is used to visualize the attention region for a CNN while making

predictions [86,87]. It uses the gradient of the final convolutional layer with respect to the predicted class to weigh the layer activations, and then averages these weighted values over the spatial dimensions to obtain a class activation map. The map shows the crucial region of the image that is used to make the prediction by the CNN model. Figure 14 shows the Grad-CAM visualization of EfficientNetV2-B0 model prediction on images recovered from different communication systems. The images that were correctly predicted has Grad-CAM heat map concentrated in the ROI region of the traffic sign, whereas the two images that were wrongly predicted has the Grad-CAM heat map outside the ROI, therefore it could not extract the important features from the ROI to make the accurate prediction for the traffic sign.



**Figure 14.** Grad-CAM visualizations of EfficientNetV2-B0 model (with Augmentation) on sample image recovered from different OFDM-based image communication systems under the Rayleigh Fading channel. The recovered image from different communication system (**top**), Grad-CAM heat map of the DL model’s attention region (**middle**) and superimposed heat map over the image (**bottom**) is shown. Images with green border are correctly predicted and images with red border are incorrectly predicted.

For the t-SNE and Grad-CAM analysis, we visualize the performance of EfficientnetV2-B0 on noisy images recovered from OFDM communication system using the Rayleigh Fading channel only. The reason for this is that we have observed from Section 4.3 that EfficientnetV2-B0 is superior and more robust compared to ResNetV152, and the Rayleigh Fading channel replicates the real-world environment inducing more noise on the images.

For the experiment, the simulation of the various image communication systems and the BER analysis was conducted using Python 3.9.7, and the image quality analysis (PSNR and MS-SSIM) was performed using MATLAB R2021b. The performance analysis of the DL models was carried out using Python 3.9.7 and TensorFlow 2.2.0. The experiments were carried out jointly on Intel Core i5-4590 CPU and NVIDIA Tesla V100 (32 GB Memory) GPU. The GPU resource provider is mentioned in the acknowledgement.

## 5. Conclusions and Future Work

In this study, we have analyzed the performance of two DL models on images transmitted over various OFDM wireless communication systems for CV applications. Specifically, we have considered higher order M-QAM systems with different channel models and channel estimation techniques. The main objective was to achieve a higher data rate to enable real time CV applications while maintaining the overall communication system complexity. In general, the utilization of a higher order M-QAM in the fading channel environment leads to heavily distorted signals, regardless of the channel estimation techniques. However, our results have shown that the feature extractor of a DL model can be robust

against these distortions with suitable data augmentation techniques, thereby improving the model generalizability across the higher order M-QAM. In other words, this trained feature extractor can extract meaningful features even from very noisy images, which can be utilized for downstream tasks, such as traffic sign recognition in ITS.

The present study implemented simple variable source coding technique and legacy OFDM PHY for communication system. As future work, we are interested in implementing PHY as a DL architecture and consider DL-based Joint Source-Channel coding for communication system and extending the application domain. Additionally, DL applications using advanced communication systems, such as MIMO and other hybrid systems, can be studied as an extension of this study.

**Author Contributions:** Conceptualization, N.I.; methodology, N.I.; software, N.I.; validation, S.S.; formal analysis, N.I.; investigation, N.I.; resources, N.I. and S.S.; data curation, N.I.; writing—original draft preparation, N.I.; writing—review and editing, S.S.; visualization, N.I.; supervision, S.S.; project administration, S.S.; funding acquisition, S.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research is supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2018R1D1A1B07048338).

**Data Availability Statement:** All the datasets used in this study are publicly available. The GTSRB dataset used for Traffic Sign Recognition is accessible at: <https://sid.erda.dk/public/archives/daaeac0d7ce1152aea9b61d9f1e19370/published-archive.html> (accessed on 9 February 2022).

**Acknowledgments:** The experiments in this paper were performed with GPU resources provided by the Korea NIPA (National IT Industry Promotion Agency).

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Cai, L.; Pan, J.; Zhao, L.; Shen, X. Networked Electric Vehicles for Green Intelligent Transportation. *IEEE Commun. Stand. Mag.* **2017**, *1*, 77–83. [[CrossRef](#)]
2. Shah, S.A.A.; Ahmed, E.; Imran, M.; Zeadally, S. 5G for Vehicular Communications. *IEEE Commun. Mag.* **2018**, *56*, 111–117. [[CrossRef](#)]
3. Zhang, K.; Leng, S.; Peng, X.; Pan, L.; Maharjan, S.; Zhang, Y. Artificial Intelligence Inspired Transmission Scheduling in Cognitive Vehicular Communications and Networks. *IEEE Internet Things J.* **2019**, *6*, 1987–1997. [[CrossRef](#)]
4. Gallego-Madrid, J.; Sanchez-Iborra, R.; Ortiz, J.; Santa, J. The Role of Vehicular Applications in the Design of Future 6G Infrastructures. *ICT Express* **2023**, *9*. [[CrossRef](#)]
5. Barry, J.R.; Lee, E.A.; Messerschmitt, D.G. *Digital Communication*; Springer Science & Business Media: Berlin, Germany, 2012; ISBN 1-4615-0227-6.
6. Simon, M.K.; Alouini, M.-S. *Digital Communication over Fading Channels: A Unified Approach to Performance Analysis*; Wiley series in telecommunications and signal processing; John Wiley & Sons: New York, NY, USA, 2000; ISBN 978-0-471-31779-1.
7. Chiueh, T.-D.; Tsai, P.-Y. *OFDM Baseband Receiver Design for Wireless Communications*; John Wiley & Sons: New York, NY, USA, 2008; ISBN 978-0-470-82248-7.
8. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; Adaptive computation and machine learning; The MIT Press: Cambridge, MA, USA, 2016; ISBN 978-0-262-03561-3.
9. Tuli, A.; Kumar, N.; Kanchan Sharma, K.; Sharma, S.T. Image Transmission Using M-QAM OFDM System over Composite Fading Channel. *IOSR J. Electron. Commun. Eng. IOSR-JECE* **2014**, *9*, 69–77. [[CrossRef](#)]
10. Krishna, D.; Anuradha, M.S. Image Transmission through OFDM System under the Influence of AWGN Channel. *IOP Conf. Ser. Mater. Sci. Eng.* **2017**, *225*, 012217. [[CrossRef](#)]
11. Mannan, A.; Habib, A. Adaptive Processing of Image Using DWT and FFT OFDM in AWGN and Rayleigh Channel. In Proceedings of the 2017 International Conference on Communication, Computing and Digital Systems (C-CODE), Islamabad, Pakistan, 8–9 March 2017; pp. 346–350.
12. Esmail, H.; Jiang, D. Progressive ZP-OFDM for Image Transmission Over Underwater Time-Dispersive Fading Channels. In Proceedings of the 2018 International Conference on Computing, Electronics & Communications Engineering (iCCECE), Southend, UK, 16–17 August 2018; pp. 226–229.
13. Agarwal, A.; Kumar, B.S.; Agarwal, K. BER Performance Analysis of Image Transmission Using OFDM Technique in Different Channel Conditions Using Various Modulation Techniques. In *Computational Intelligence in Data Mining: Advances in Intelligent Systems and Computing*; Behera, H.S., Nayak, J., Naik, B., Abraham, A., Eds.; Springer: Singapore, 2019, Volume 711, pp. 1–8; ISBN 978-981-10-8054-8.

14. Patel, J.; Seto, M. Live RF Image Transmission Using OFDM with RPi and PlutoSDR. In Proceedings of the 2020 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), London, ON, Canada, 30 August 2020; pp. 1–5.
15. Rajesh, V.; Abdul Rajak, A.R. Channel Estimation for Image Restoration Using OFDM with Various Digital Modulation Schemes. *J. Phys. Conf. Ser.* **2020**, *1706*, 012076. [[CrossRef](#)]
16. Al-Shably, Z.H.; Hussain, Z.M. Performance of FFT-OFDM versus DWT-OFDM under Compressive Sensing. *J. Phys. Conf. Ser.* **2021**, *1804*, 012087. [[CrossRef](#)]
17. Mohsin, M.J.; Saad, W.K.; Hamza, B.J.; Jabbar, W.A. Performance Analysis of Image Transmission with Various Channel Conditions/Modulation Techniques. *TELKOMNIKA (Telecommun. Comput. Electron. Control)* **2020**, *18*, 1158–1168. [[CrossRef](#)]
18. Ghanim, Z.N.; Omran, B.M. OFDM PAPR Reduction for Image Transmission Using Improved Tone Reservation. *Int. J. Electr. Comput. Eng.* **2021**, *11*, 416–423. [[CrossRef](#)]
19. Kansal, L.; Gaba, G.S.; Chilamkurti, N.; Kim, B.-G. Efficient and Robust Image Communication Techniques for 5G Applications in Smart Cities. *Energies* **2021**, *14*, 3986. [[CrossRef](#)]
20. Kansal, L.; Berra, S.; Mounir, M.; Miglani, R.; Dinis, R.; Rabie, K. Performance Analysis of Massive MIMO-OFDM System Incorporated with Various Transforms for Image Communication in 5G Systems. *Electronics* **2022**, *11*, 621. [[CrossRef](#)]
21. Bourtsoulatze, E.; Kurka, D.B.; Gündüz, D. Deep Joint Source-Channel Coding for Wireless Image Transmission. *IEEE Trans. Cogn. Commun. Netw.* **2019**, *5*, 567–579. [[CrossRef](#)]
22. Burth Kurka, D.; Gündüz, D. Joint Source-Channel Coding of Images with (Not Very) Deep Learning. In Proceedings of the International Zurich Seminar on Information and Communication (IZS 2020), Zurich, Switzerland, 26 February 2020; pp. 90–94.
23. Kurka, D.B.; Gunduz, D. DeepJSCC-f: Deep Joint Source-Channel Coding of Images With Feedback. *IEEE J. Sel. Areas Inf. Theory* **2020**, *1*, 178–193. [[CrossRef](#)]
24. Kurka, D.B.; Gunduz, D. Successive Refinement of Images with Deep Joint Source-Channel Coding. In Proceedings of the 2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Cannes, France, 2–5 July 2019; pp. 1–5.
25. Xu, J.; Ai, B.; Chen, W.; Yang, A.; Sun, P.; Rodrigues, M. Wireless Image Transmission Using Deep Source Channel Coding with Attention Modules. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 2315–2328. [[CrossRef](#)]
26. Ding, M.; Li, J.; Ma, M.; Fan, X. SNR-Adaptive Deep Joint Source-Channel Coding for Wireless Image Transmission. In Proceedings of the ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6 June 2021; pp. 1555–1559.
27. Yang, M.; Bian, C.; Kim, H.-S. OFDM-Guided Deep Joint Source Channel Coding for Wireless Multipath Fading Channels. *IEEE Trans. Cogn. Commun. Netw.* **2022**, *8*, 584–599. [[CrossRef](#)]
28. Wu, H.; Shao, Y.; Mikolajczyk, K.; Gunduz, D. Channel-Adaptive Wireless Image Transmission with OFDM. *IEEE Wirel. Commun. Lett.* **2022**, *11*, 2400–2404. [[CrossRef](#)]
29. Ahmad, I.; Islam, N.; Kim, E.; Shin, S. Performance Analysis of Cloud Based Deep Learning Models in OFDM Based Image Communication System. In Proceedings of the Korean Institute of Communication Sciences Conference, KICS, Jeju Island, Republic of Korea, 22–24 June 2022; pp. 500–501.
30. Ahmad, I.; Islam, N.; Shin, S. Performance Analysis of Cloud-Based Deep Learning Models on Images Recovered without Channel Correction in OFDM System. In Proceedings of the 2022 27th Asia Pacific Conference on Communications (APCC), Jeju Island, Republic of Korea, 19–21 October 2022; pp. 225–259. [[CrossRef](#)]
31. Islam, N.; Ahmad, I.; Shin, S. Robustness of Deep Learning Enabled IoT Applications Utilizing Higher Order QAM in OFDM Image Communication System. In Proceedings of the 2023 International Conference on Artificial Intelligence in Information and Communication (ICAIC), Bali, Indonesia, 20 February 2023; pp. 630–635. [[CrossRef](#)]
32. Banelli, P.; Buzzi, S.; Colavolpe, G.; Modenini, A.; Rusek, F.; Ugolini, A. Modulation Formats and Waveforms for 5G Networks: Who Will Be the Heir of OFDM?: An Overview of Alternative Modulation Schemes for Improved Spectral Efficiency. *IEEE Signal Process. Mag.* **2014**, *31*, 80–93. [[CrossRef](#)]
33. Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; Fergus, R. Intriguing Properties of Neural Networks. *arXiv* **2013**, arXiv:1312.6199.
34. Goodfellow, I.J.; Shlens, J.; Szegedy, C. Explaining and Harnessing Adversarial Examples. *arXiv* **2014**, arXiv:1412.6572.
35. Kim, K.-S.; Lee, H.-Y.; Lee, H.-K. Spatial Error Concealment Technique for Losslessly Compressed Images Using Data Hiding in Error-Prone Channels. *J. Commun. Netw.* **2010**, *12*, 168–173. [[CrossRef](#)]
36. Boyat, A.K.; Joshi, B.K. A Review Paper: Noise Models in Digital Image Processing. *arXiv* **2015**, arXiv:1505.03489. [[CrossRef](#)]
37. Dodge, S.; Karam, L. Understanding How Image Quality Affects Deep Neural Networks. In Proceedings of the 2016 Eighth International Conference on Quality of Multimedia Experience (QoMEX), Lisbon, Portugal, 6–8 June 2016; pp. 1–6.
38. Roy, P.; Ghosh, S.; Bhattacharya, S.; Pal, U. Effects of Degradations on Deep Neural Network Architectures. *arXiv* **2023**, arXiv:1807.10108.
39. De, K.; Pedersen, M. Impact of Colour on Robustness of Deep Neural Networks. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), Montreal, BC, Canada, 11–17 October 2021; pp. 21–30.
40. Hendrycks, D.; Dietterich, T. Benchmarking Neural Network Robustness to Common Corruptions and Perturbations. *arXiv* **2019**, arXiv:1903.12261.

41. Geirhos, R.; Rubisch, P.; Michaelis, C.; Bethge, M.; Wichmann, F.A.; Brendel, W. ImageNet-Trained CNNs Are Biased towards Texture; Increasing Shape Bias Improves Accuracy and Robustness. *arXiv* **2018**, arXiv:1811.12231.
42. Brendel, W.; Bethge, M. Approximating CNNs with Bag-of-Local-Features Models Works Surprisingly Well on ImageNet. *arXiv* **2019**, arXiv:1904.00760.
43. Naseer, M.; Ranasinghe, K.; Khan, S.; Hayat, M.; Khan, F.S.; Yang, M.-H. Intriguing Properties of Vision Transformers. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 23296–23308.
44. Coleri, S.; Ergen, M.; Puri, A.; Bahai, A. Channel Estimation Techniques Based on Pilot Arrangement in OFDM Systems. *IEEE Trans. Broadcast.* **2002**, *48*, 223–229. [[CrossRef](#)]
45. Ahmad, I.; Shin, S. A Novel Hybrid Image Encryption–Compression Scheme by Combining Chaos Theory and Number Theory. *Signal Process. Image Commun.* **2021**, *98*, 116418. [[CrossRef](#)]
46. Shen, Y.; Martinez, E. Channel Estimation in OFDM Systems. *Freescale Semiconductor*. **2006**, AN3059.
47. Jeruchim, M.C.; Balaban, P.; Shanmugan, K.S.; Jeruchim, M.C. (Eds.) *Simulation of Communication Systems: Modeling, Methodology, and Techniques*, 2nd ed.; Information technology—Transmission, processing, and storage; Kluwer Academic/Plenum Publishers: New York, NY, USA, 2000; ISBN 978-0-306-46267-2.
48. van de Beek, J.-J. Synchronization and Channel Estimation in OFDM Systems. Doctoral Thesis, Luleå University of Technology, Luleå, Sweden, 1998.
49. Choi, J.-W.; Lee, Y.-H. Optimum Pilot Pattern for Channel Estimation in OFDM Systems. *IEEE Trans. Wirel. Commun.* **2005**, *4*, 2083–2088. [[CrossRef](#)]
50. Kim, W.; Ahn, Y.; Kim, J.; Shim, B. Towards Deep Learning-Aided Wireless Channel Estimation and Channel State Information Feedback for 6G. *J. Commun. Netw.* **2023**, *25*, 61–75. [[CrossRef](#)]
51. Yi, X.; Zhong, C. Deep Learning for Joint Channel Estimation and Signal Detection in OFDM Systems. *IEEE Commun. Lett.* **2020**, *24*, 2780–2784. [[CrossRef](#)]
52. Cortes, J.A.; Canete, F.J.; Diez, L. Channel Estimation for OFDM-Based Indoor Broadband Power Line Communication Systems. *J. Commun. Netw.* **2023**, *25*, 151–166. [[CrossRef](#)]
53. Wang, F. Pilot-Based Channel Estimation in OFDM System. Master’s Thesis, University of Toledo, Toledo, OH, USA, 2011.
54. Ergen, M. *Mobile Broadband: Including WiMAX and LTE*; Springer: New York, NY, USA, 2009; ISBN 978-0-387-68189-4.
55. Singya, P.K.; Shaik, P.; Kumar, N.; Bhatia, V.; Alouini, M.-S. A Survey on Higher-Order QAM Constellations: Technical Challenges, Recent Advances, and Future Trends. *IEEE Open J. Commun. Soc.* **2021**, *2*, 617–655. [[CrossRef](#)]
56. Cho, K.; Yoon, D. On the General BER Expression of One- and Two-Dimensional Amplitude Modulations. *IEEE Trans. Commun.* **2002**, *50*, 1074–1080. [[CrossRef](#)]
57. Smith, J. Odd-Bit Quadrature Amplitude-Shift Keying. *IEEE Trans. Commun.* **1975**, *23*, 385–389. [[CrossRef](#)]
58. Campopiano, C.; Glazer, B. A Coherent Digital Amplitude and Phase Modulation Scheme. *IEEE Trans. Commun.* **1962**, *10*, 90–95. [[CrossRef](#)]
59. Vitthaladevuni, P.K.; Alouini, M.-S.; Kieffer, J.C. Exact BER Computation for Cross QAM Constellations. *IEEE Trans. Wirel. Commun.* **2005**, *4*, 3039–3050. [[CrossRef](#)]
60. Shannon, C.E. A Mathematical Theory of Communication. *SIGMOBILE Mob. Comput. Commun. Rev.* **2001**, *5*, 3–55. [[CrossRef](#)]
61. Bellalta, B. IEEE 802.11 Ax: High-Efficiency WLANs. *IEEE Wirel. Commun.* **2016**, *23*, 38–46. [[CrossRef](#)]
62. Yegnanarayana, B. *Artificial Neural Networks*; 2011 print; Prentice Hall of India: New Delhi, India, 2005; ISBN 978-81-203-1253-1.
63. Lawrence, S.; Giles, C.; Tsoi, A. *What Size Neural Network Gives Optimal Generalization? Convergence Properties of Backpropagation*; University of Maryland: College Park, MD, USA, 1996; pp. 1–37.
64. Chauvin, Y. Generalization Performance of Overtrained Back-Propagation Networks. In *Neural Networks*; Almeida, L.B., Wellekens, C.J., Eds.; Lecture Notes in Computer Science; Springer: Berlin, Germany, 1990; Volume 412, pp. 45–55, ISBN 978-3-540-52255-3.
65. LeCun, Y. Generalization and Network Design Strategies. In *Connectionism in Perspective*; Elsevier: Zurich, Switzerland, 1989.
66. Clevert, D.-A.; Unterthiner, T.; Hochreiter, S. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). *arXiv* **2015**, arXiv:1511.07289.
67. Nair, V.; Hinton, G.E. Rectified Linear Units Improve Restricted Boltzmann Machines. In Proceedings of the 27th International Conference on International Conference on Machine Learning, Haifa, Israel, 21–24 June 2010; pp. 807–814.
68. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-Based Learning Applied to Document Recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [[CrossRef](#)]
69. Shorten, C.; Khoshgoftaar, T.M. A Survey on Image Data Augmentation for Deep Learning. *J. Big Data* **2019**, *6*, 60. [[CrossRef](#)]
70. Kim, D.; Park, S.; Kim, J.; Bang, J.Y.; Jung, S. Stabilized Adaptive Sampling Control for Reliable Real-Time Learning-Based Surveillance Systems. *J. Commun. Netw.* **2021**, *23*, 129–137. [[CrossRef](#)]
71. Kiran, N.; Pan, C.; Wang, S.; Yin, C. Joint Resource Allocation and Computation Offloading in Mobile Edge Computing for SDN Based Wireless Networks. *J. Commun. Netw.* **2020**, *22*, 1–11. [[CrossRef](#)]
72. Ahmad, I.; Shin, S. Perceptual Encryption-Based Privacy-Preserving Deep Learning in Internet of Things Applications. In Proceedings of the 2022 13th International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Republic of Korea, 19 October 2022; pp. 1817–1822. [[CrossRef](#)]
73. Ribeiro, M.; Grolinger, K.; Capretz, M.A.M. MLaaS: Machine Learning as a Service. In Proceedings of the 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA), Miami, FL, USA, 9–11 December 2015; pp. 896–902.

74. Hore, A.; Ziou, D. Image Quality Metrics: PSNR vs. SSIM. In Proceedings of the 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 23–26 August 2010; pp. 2366–2369.
75. Wang, Z.; Simoncelli, E.P.; Bovik, A.C. Multiscale Structural Similarity for Image Quality Assessment. In Proceedings of the THE Thirty-Seventh Asilomar Conference on Signals, Systems & Computers, Pacific Grove, CA, USA, 9–12 November 2003; pp. 1398–1402.
76. De La Escalera, A.; Moreno, L.E.; Salichs, M.A.; Armingol, J.M. Road Traffic Sign Detection and Classification. *IEEE Trans. Ind. Electron.* **1997**, *44*, 848–859. [[CrossRef](#)]
77. Bahlmann, C.; Zhu, Y.; Ramesh, V.; Pellkofer, M.; Koehler, T. A System for Traffic Sign Detection, Tracking, and Recognition Using Color, Shape, and Motion Information. In Proceedings of the IEEE Proceedings. Intelligent Vehicles Symposium, Las Vegas, NV, USA, 6–8 June 2005; pp. 255–260.
78. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.
79. He, K.; Zhang, X.; Ren, S.; Sun, J. Identity Mappings in Deep Residual Networks. In *Computer Vision—ECCV 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer: Cham, Switzerland, 2016.
80. Tan, M.; Le, Q. Efficientnet: Rethinking Model Scaling for Convolutional Neural Networks. In Proceedings of the 36th International Conference on Machine Learning, PMLR, Long Beach, CA, USA, 9–15 June 2019; pp. 6105–6114.
81. Tan, M.; Le, Q. Efficientnetv2: Smaller Models and Faster Training. In Proceedings of the International Conference on Machine Learning, PMLR, Virtual, 18–24 July 2021; pp. 10096–10106.
82. Ahmad, I.; Shin, S. A Perceptual Encryption-Based Image Communication System for Deep Learning-Based Tuberculosis Diagnosis Using Healthcare Cloud Services. *Electronics* **2022**, *11*, 2514. [[CrossRef](#)]
83. Li, Y.; Li, M.; Luo, B.; Tian, Y.; Xu, Q. DeepDyve: Dynamic Verification for Deep Neural Networks. In Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event USA, 9–13 November 2020; pp. 101–112.
84. Zhang, Z.; Ye, M.; Xie, Y.; Liu, Y. PG-Prnet: A Lightweight Parallel Gated Feature Extractor Based on An Adaptive Progressive Regularization Algorithm. In Proceedings of the 2022 3rd International Conference on Big Data & Artificial Intelligence & Software Engineering, Guangzhou, China, 21–23 October 2022.
85. Van der Maaten, L.; Hinton, G. Visualizing Data Using T-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.
86. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; Torralba, A. Learning Deep Features for Discriminative Localization. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2921–2929.
87. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.