

Research Article ResNet-Enabled cGAN Model for Channel Estimation in Massive MIMO System

Jyoti Deshwal Yadav (), Vivek K. Dwivedi (), and Saurabh Chaturvedi ()

Department of Electronics and Communication Engineering, Jaypee Institute of Information Technology, Noida 201309, India

Correspondence should be addressed to Saurabh Chaturvedi; saurabh.chaturvedi@jiit.ac.in

Received 22 April 2022; Revised 8 July 2022; Accepted 5 August 2022; Published 29 August 2022

Academic Editor: Abdul Basit

Copyright © 2022 Jyoti Deshwal Yadav et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Massive multiple-input multiple-output (MIMO), or large-scale MIMO, is one of the key technologies for future wireless networks to exhibit a large accessible spectrum and throughput. The performance of a massive MIMO system is strongly reliant on the nature of various channels and interference during multipath transmission. Therefore, it is important to compute accurate channel estimation. This paper considers a massive MIMO system with one-bit analog-to-digital converters (ADCs) on each receiver antenna of the base station. Deep learning (DL)-based channel estimation framework has been developed to reduce signal processing complexity. This DL framework uses conditional generative adversarial networks (cGANs) and various convolutional neural networks, namely reverse residual network (reverse ResNet), squeeze-and-excitation ResNet (SE ResNet), ResUNet++, and reverse SE ResNet, as the generator model of cGAN for extracting the features from the quantized received signals. The simulation results of this paper show that the trained residual block-based generator model of cGAN has better channel generation performance than the standard generator model in terms of mean square error.

1. Introduction

The frequency of wireless data traffic has increased dramatically worldwide during the last few decades [1]. This scenario puts considerable pressure on the current wireless communication system. The instant increase in the demand for several laptops and smart devices, the prevalence of online gaming and social networking, and the highdemand services like interactive media led to an outburst in cellular network data traffic, with demands projected to rise continuously at a rapid rate [2, 3]. As per Cisco's visual networking index forecast, the global mobile data traffic increased approximately ten-fold from 2013 to 2018 [4]. By the end of 2022, most of the traffic will be derived from mobile phones. The previous wireless generation systems were not able to manage this enormous amount of data. This scenario has motivated the consideration of a wireless technology known as multiple-input multiple-output (MIMO), both in theory and practice [5]. MIMO technology uses multiple antennas for a considerable improvement in spectral efficiency. This technology is divided into three parts:

point-to-point MIMO, multiuser MIMO, and massive MIMO [6]. The point-to-point MIMO is the simplest form of MIMO in which a base station (BS) equipped with an antenna array serves a terminal equipped with an antenna array [6]. In multiuser MIMO, the array of terminals using the same time-frequency resources is served by a single base station [7]. This scenario comes from a point-to-point MIMO setup by splitting the K-antenna terminal into multiple separate terminals. The massive MIMO is a scalable version of multiuser MIMO. It is also called the extension of MIMO technology, which includes hundreds and even thousands of antennas at the BS to enhance the throughput and energy efficiency [8]. The underlying idea that allows for a considerable gain in energy efficiency is that energy can be absorbed with high intensity into smaller areas when a large number of antennas are used [9]. By utilizing the multiplexing gains and spatial diversity, massive MIMO improves the robustness and spectral efficiency of wireless communication systems under limited channel fading and bandwidth [10]. The issues associated with practical massive MIMO systems are the high power consumption and expensive hardware. The utilization of analog-to-digital converters (ADCs) is one of the promising solutions for this issue. The power transmitted by these complex ADCs is inversely proportional to the number of antennas [11]. The power consumed by each ADC increases exponentially with the number of quantization bits and linearly with the sampling rate [12, 13]. Therefore, low-resolution one-bit ADCs are considered for a massive MIMO system.

In a one-bit massive MIMO system, deep learning (DL) techniques are being applied to empower its full potential [14]. The influence of DL techniques grew rapidly in early 2000s. DL is a subset of machine learning (ML) that makes use of neural networks and utilizes supervised, unsupervised, and reinforcement learning [15]. DL is used in a number of domains, including automatic speech recognition, object detection, and image classification. It is also used in many wireless communication applications, such as resource management, spectral sensing, beamforming, signal detection, and channel estimation [16-18]. Using the DL techniques, a task can be analyzed easily and with reliable results. In [19], DL techniques have been successfully used for signal detection with nonlinear distortion and joint channel estimation. In [20], a deep convolutional neural network (CNN) has been used to explore channel correlation and improve the channel estimation accuracy. DL approaches can also be used to solve the beam selection problem in massive MIMO systems [21].

In [22], a DL technique was used to estimate the channel for one-bit massive MIMO system. By using this DL technique, it is difficult to generate a more realistic channel matrix because of information loss with subsequent layers in neural networks [22]. This paper employs the same DL technique used in [22] with different CNN architectures to generate more realistic and accurate channels for one-bit massive MIMO system.

The performance of the proposed approach is measured in terms of mean square error (MSE) and the number of used BS antennas. The conditional generative adversarial network (cGAN) is a form of GAN that is used to determine an adaptive loss function, called GAN loss, for variety datasets and applications. The learning curves for the generator and discriminator models of all the cGAN versions are also tracked and estimated. The following are the main contributions of this paper: The reverse residual network (reverse ResNet), squeeze-and-excitation ResNet (SE ResNet), ResU-Net++, and reverse SE ResNet versions of the ResNet architecture are successfully implemented as a generator model of cGAN, and their performance is compared in terms of MSE and the number of BS antennas used. The learning curves of the generator and discriminator models are also examined. The learning curves show that both the generator and discriminator are following their adversarial property, indicating that the training scheme is working properly. However, after upgrading the ResNet to its variants, the discriminator may need to be upgraded as well to keep up with the more powerful generator. The results exhibit that for channel estimation, the cGAN with reverse SE ResNet as a generator model outperforms the other techniques. The rest of this paper is structured as follows: The architecture of a one-bit

massive MIMO system is described in Section 2. The characteristics of DL models and channel estimation using cGAN are discussed in Section 3. The simulation results are presented and compared in Section 4. Finally, Section 5 concludes the paper.

2. Architecture of One-Bit Massive MIMO System

In a basic massive MIMO setup, each BS is equipped with a large number of antennas and serves a cell with a large number of users. Each user is considered as a single antenna. All users simultaneously occupy the full time-frequency resources both in uplink and downlink transmissions. On the BS and user sides, an uplink massive MIMO system with a uniform linear array (ULA) is considered. As shown in Figure 1, there are M BS antennas and K single-antenna mobile users (MUs). Each antenna at BS is equipped with two one-bit ADCs for real and imaginary components of the signal received from each MU antenna. By following the channel model of [22], the channel matrix for the K MUs can be expressed by

$$h_{k} = \sum_{l=1}^{N_{\rho}} \omega_{l} a \left(\beta_{\text{azi}}^{k}, \beta_{\text{aod}}^{k} \right), \tag{1}$$

where *l* is the channel path, N_{ρ} is the number of channel paths, the complex gain of each path *l* is ω_l , and *a*(.) denote the array response vector of the BS. β_{azi}^k and β_{aod}^k denote the azimuth angle of departure and elevation angle of departure, respectively. Finally, the full channel matrix *H* for *K* MUs can be expressed by (2), where the dimension of *H* is $M \times K$:

$$H = [h_1, h_2, \dots, \dots, h_k, \dots, \dots, h_K].$$
(2)

2.1. Applying One-Bit ADCs for Channel Estimation. The channel is estimated at the BS by using a pilot sequence of length τ from the user side. As shown in Figure 1, K MUs concurrently transmit the pilot sequence of length τ to BS. The received signal Y at the BS before one-bit quantization is given by [22]:

$$Y = H\phi + N, \tag{3}$$

where the dimension of Y is $M \times \tau$, N is a noise matrix whose dimension is $M \times \tau$, and ϕ is the randomly assigned pilot sequence from K users. During the quantization process, the real and imaginary components of a signal from BS antennas are separately quantized using one-bit ADC. The function used for quantization is signum function sgn(.). After one-bit quantization, the received signal Y can be expressed by [22]

$$Y = \operatorname{sgn} (H\phi + N), \tag{4}$$



FIGURE 1: Architecture of one-bit massive MIMO system.

where sgn(.) is

$$sgn = \begin{cases} 1 \text{ for } x \ge 0, \\ -1, \text{ otherwise.} \end{cases}$$
(5)

The quantized signal *Y* takes the values from the set $\{1+j, 1-j, -1+j, -1-j\}$. The goal of this research is to use an adversarial DL model to extract the channel matrix \hat{H} from the quantized received signal *Y*, as well as to evaluate their performance in terms of MSE and the number of BS antennas employed.

3. Channel Estimation Using cGAN

In this section, CNN-based cGAN has been used to perform the channel estimation tasks. In this work, the channel estimation is considered as an image-to-image translation problem by considering channel matrix H and received signal Yas two channel images with parameters of $M \times N \times 2$ and $M \times \tau \times 2$, respectively. Here two channels signify the real and imaginary part of the complex matrix.

3.1. cGAN Architecture. The standard GAN is composed of two neural networks, namely generator G and discriminator D. As shown in Figure 2, G learns the data distribution from the original dataset and generates new images [23]. The G is trained to improve the quality of generated data to fool the discriminator, so that generated samples produced by Gare considered real ones, and the D is trained to differentiate between generated and real samples. If D can successfully differentiate between the generated and real samples, then G will receive feedback based on D's success, so that G can learn to generate samples similar to the real ones. Both networks work against each other to achieve the best results [22]. To achieve this goal, the cGAN loss is composed of two parts: the adversarial loss and the L1 loss [24]. The adversarial loss can be expressed by

$$L_{cGAN}(G, D, Y, H) = \min_{G} \max_{D} \mathbb{E}[\log D(H)] + \mathbb{E}[\log (1 - D(G(Y)))].$$
(6)

The expression of L1 loss is given by

$$L_1 = \left(\mathbb{E} \left[\left\| H - G(Y) \right\|^2 \right] \right). \tag{7}$$

Here L_1 distance is added to the generator loss to encourage the low-frequency accuracy of the generated image. The L_1 distance is preferred over L_2 distance because it produces images with less blurring. Finally, the complete expression for the cGAN loss is given by [22]

$$\begin{split} L_{\text{cGAN}}(G, D, Y, H) &= \min_{G} \max_{D} \mathbb{E}[\log D(H)] \\ &+ \mathbb{E}[\log \left(1 - D(G(Y))\right)] + \lambda L_1(G). \end{split}$$

The architecture of G and D neural networks must be carefully chosen in order to make training easier [25].

3.2. Proposed CNN Architectures for G Model of cGAN. In this paper, a cGAN architecture is used to create artificial channel samples in a one-bit massive MIMO system illustrated in Figure 3. To make the training process easier, the architecture for the G model must be chosen carefully [26]. After conducting a thorough model exploration for the G model and taking into account prior findings [22, 27], a few ResNet variant architectures have been applied to the G model in this work, and the learning curves of the G and D models have also been analyzed to verify the proper functionality of training scheme. In the cGAN model, both



FIGURE 3: Proposed cGAN architecture for channel estimation in one-bit massive MIMO system.

G and D using CNN architectures are composed of many convolutional layers, batch normalization (BN) layers, activation functions, and dropout layers. The first CNN architecture implemented for the G model of cGAN in this paper is reverse ResNet [26] for channel estimation. As depicted in Figure 4, the order of BN layer, activation function, and convolution layer in the residual block of ResNet described in [28] was changed from Conv-BN-ReLU to BN-ReLU-Conv in this CNN architecture. This modified architecture of ResNet, i.e., reverse ResNet, trains faster and achieves better results than ResNet [28]. The second CNN architecture implemented for the cGAN G model in this paper is SE ResNet, in which an SE block is added to ResNet to perform feature recalibration.

In the SE ResNet architecture, SE block is used after each residual block in order to improve the network's representation [29]. As demonstrated in Figure 5, the global average pooling layer is used for the squeeze phase to minimize the overfitting by reducing the total number of parameters in the model, and two fully connected layers are used for the excitation phase to fine-tune the obtained features for precise channel estimation in the SE block following a channel-wise scaling operation. The third ResNet variant implemented

for the cGAN G model in this paper is ResUNet++ [30]. A few modifications have been done in this CNN architecture to make it less complex. The output of each SE ResNet block is concatenated with its corresponding layer in the decoder. ResUNet++ is based on the ResUNet architecture that was presented in [27] for channel estimation. The ResUNet++ architecture takes advantage of both the residual and SE blocks in the U-Net CNN architecture [31]. A residual block spreads information across layers and reduces the vanishing gradient problem, while an SE block performs the feature recalibration [32]. Figure 6 depicts the block diagram of modified ResUNet++ architecture. In the modified ResUNet++ architecture, the encoder uses the SE block with residual block, whereas the decoder uses a sequence of deconvolution, BN, dropout, and concatenation with the SE ResNet through skip connections [32].

Reverse SE ResNet is the fourth CNN architecture used in this paper for the G model of cGAN. As shown in Figure 7, this CNN architecture combines reverse ResNet and SE ResNet. Because of the reverse ResNet and the SE ResNet, this combination allows for faster training and better feature calibration.



FIGURE 4: Reverse ResNet architecture for G model of cGAN.



FIGURE 5: SE ResNet architecture for G model of cGAN.



FIGURE 6: ResUNet++ architecture for G model of cGAN.



FIGURE 7: Reverse SE ResNet architecture for G model of cGAN.

3.3. Discriminator Model Architecture. The PatchGAN architecture [24] has been used for the D model in this paper. As reported in [24], the PatchGAN discriminator takes individual patches of an image rather than the entire image and classifies them separately as real and fake, which encourages sharp and high-frequency details of an image. The cGAN D model has two-dimensional convolutional layers with a kernal size of 4, BN layers, and ReLU activation function. The advantage of adopting PatchGAN is that the image size is unrestricted, as well as the image resolution and texture structures are unaffected [33].

4. Simulation Results

4.1. Datasets and Model Training. In this paper, an indoor massive MIMO scenario "I1_2p4" of [34] has been considered for channel estimation using cGAN in one-bit massive MIMO system. In this scenario, there is a $10m \times 10m$ dimensional room with two tables.

Users are arranged on x-y grids and antennas are placed on up part of the ceiling. A dataset is generated by taking this scenario, called the DeepMIMO dataset. The parameters for channel simulation are listed in Table 1. Moreover, four

TABLE 1: Parameters for simulation.

Antenna array type	ULA
Bandwidth (BW)	0.01 GHz
Antenna spacing	BW/2
Total paths	10



FIGURE 8: Model comparison in terms of MSE.

channel datasets are generated with different number of BS antennas; *M* varies from 64 to 256, and the number of MUs is fixed to K = 32. For the training and testing purposes, the generated DeepMIMO dataset components were shuffled and distributed in the ratio of 70% and 30%, respectively [35]. To train the G model of cGAN, the Adam optimization algorithm [36] with a learning rate of 2×10^{-4} has been used. The RMSProp algorithm [37] has been adopted with a learning rate of 2×10^{-5} for stable training of the D model.

4.2. *Matrices Estimation.* This paper makes the use of MSE to compute the variation between the actual channel matrix and the estimated channel matrix. The MSE is expressed by [22]

MSE =
$$10 \log_{10} \{ \mathbb{E} [H - \hat{H}] 2 \},$$
 (9)

where H and \hat{H} are the vectors to represent the actual channel matrix and estimated channel matrix over all antennas. $\mathbb{E}[.]$ calculates values of expectation.

4.3. Performance Comparison. In this paper, the cGAN G model has been implemented using four variants of ResNet architecture, namely reverse ResNet, SE ResNet, ResUNet++, and reverse SE ResNet for channel estimation in one-bit massive MIMO system, and their performance is compared

TABLE 2: Performance comparison of cGAN models.

CNN architecture for G model of cGAN	Minimum MSE (dB)
ResNet	-37.2479
Reverse ResNet	-38.0479
ResUNet++	-35.0092
SE ResNet	-37.9267
Reverse SE ResNet	-40.1313



FIGURE 9: Model performance comparison by varying the number of BS antennas.

with the cGAN G model implemented with ResNet architecture [28] and U-Net architecture [29]. Figure 8 and Table 2 exhibit the performance comparison in terms of MSE. As shown, the cGAN G model with reverse SE ResNet architecture outperforms the remaining cGAN models. The cGAN G model based on reverse SE ResNet is capable of producing better channel estimation results than that of other models because an SE block performs feature recalibration through which a network can learn to use global data to selectively highlight the essential features and suppress the nonessential ones.

Moreover, by reversing the residual block, the model trains faster and achieves better results than the results of original residual block. As depicted in Figure 9, the MSE performance is compared by varying the number of BS antennas and setting the pilot sequence length to 8 and the number of MUs to 32. It can be observed that the MSE value of all cGAN models increases a bit when the number of BS antennas is increased from 64 to 256. The MSE value of cGAN G model with reverse SE ResNet architecture stays in the range from -40 dB to -39 dB. There is no noticeable improvement in MSE for all the G models by increasing the number of BS antennas from 64 to 256. Figures 10 and



FIGURE 10: Comparison of the G training curves of all cGAN models.



FIGURE 11: Comparison of the D training curves of all cGAN models.

11 demonstrate the training curves of the cGAN G and D models. It appears that balancing the convergence of both G and D is extremely difficult. When one of them is over trained, the system becomes unstable. It is observed that G performs poorly in beginning epochs and improves with subsequent epochs, but D performs well in initial epochs because D can quickly distinguish between a real and a fake one. As shown in Figure 12, the training curves of cGAN G model using ResNet and reverse SE ResNet CNN architectures are compared with the training curves of cGAN D model. With subsequent epochs, the training curves of both the G and D models converge. Figure 12 depicts the adver-



FIGURE 12: Comparison of the G and D losses.



FIGURE 13: Comparing the training and testing losses of generator and discriminator of cGAN.

sarial property of cGAN. As can be seen, finding a reasonable gradient to follow during training for the G model in the early epochs is challenging; therefore, the G loss is fairly random. The G eventually improves, but it still does not converge properly as shown in Figure 12.

The user's traffic and mode collapse in cGAN are two possible explanations of this convergence failure. Both the G and D models are trying to minimize their own loss function. It is also clear from Figure 12 that the same cGAN model performs differently with different G models, and the strength of D varies in proportion to the strength of G. Lastly, the training and testing losses of cGAN G and D models are compared in Figure 13. As shown in Figure 13, there is a large gap between the G's training and testing losses, but only a small gap between the D's training and testing losses. It can be concluded that cGAN performance with testing data is better and satisfactory while performing channel estimation for massive MIMO systems.

5. Conclusion

In this paper, the cGAN generator models have been implemented using different CNN architectures, including reverse ResNet, SE ResNet, SE ResNet, ResUNet++, and reverse SE ResNet, and their performances have been compared in terms of MSE for channel estimation. As shown in the simulation results, a cGAN model with a reverse SE ResNet generator and a PatchGAN discriminator can be used to estimate channels precisely. The cGAN discriminator model can be developed with reverse SE ResNet CNN architecture and compared in future work to gain better results for channel estimation. Furthermore, other GAN models, such as deep convolutional GAN and cycle GAN, can also be proposed and used instead of cGAN for channel estimation in a one-bit massive MIMO system.

Data Availability

The underlying data supporting the results may be made available on request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- R. Chataut and R. Akl, "Massive MIMO systems for 5G and beyond networks-overview, recent trends, challenges, and future research direction," *Sensor*, vol. 20, no. 10, p. 2753, 2020.
- [2] P. Sabeti, A. Farhang, I. Macaluso, N. Marchetti, and L. Doyle, "Blind channel estimation for massive MIMO: a deep learning assisted approach," in *IEEE International Conference on Communications*, pp. 1–6, Dublin, Ireland, 2020.
- [3] A. Khansefid and H. Minn, "On channel estimation for massive MIMO with pilot contamination," *IEEE Communications Letters*, vol. 19, no. 9, pp. 1160–1163, 2015.
- [4] H. Papadopoulos, C. Wang, O. Bursalioglu, X. Hou, and Y. Kishiyama, "Massive MIMO technologies and challenges towards 5G," *IEICE Transactions on Communications*, vol. -E99.B, no. 3, pp. 602–621, 2016.
- [5] L. Fan, S. Jin, C.-K. Wen, and H. Zhang, "Uplink achievable rate for massive MIMO systems with low-resolution ADC," *IEEE Communication Letter*, vol. 19, no. 12, pp. 2186–2189, 2015.
- [6] T. Marzetta, E. Larsson, H. Yang, and H. Ngo, *Introduction in Fundamentals of Massive MIMO*, Cambridge University Press, 2016.
- [7] E.-G. Larsson, O. Edfors, F. Tufvesson, and T.-L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 186–195, 2014.
- [8] H. Halbauer, A. Weber, D. Wiegner, and T. Wild, "Energy efficient massive MIMO array configurations," in 2018 IEEE Globecom Workshops (GC Wkshps), pp. 1–6, Abu Dhabi, United Arab Emirates, 2018.
- [9] D. Muirhead, M.-A. Imran, and K. Arshad, "A survey of the challenges, opportunities and use of multiple antennas in cur-

rent and future 5G small cell base stations," *IEEE Access*, vol. 4, pp. 2952–2964, 2016.

- [10] D. Borges, P. Montezuma, R. Dinis, and M. Beko, "Massive MIMO techniques for 5G and beyond-opportunities and challenges," *Electronics*, vol. 10, no. 14, p. 1667, 2021.
- [11] C. Kong, A. Mezghani, C. Zhong, A.-L. Swindlehurst, and Z. Zhang, "Multipair massive MIMO relaying systems with one-bit ADCs and DACs," *IEEE Transactions on Signal Processing*, vol. 66, no. 11, pp. 2984–2997, 2018.
- [12] Z. Xiao, J. Zhao, T. Liu, L. Geng, F. Zhang, and J. Tong, "On the energy efficiency of massive MIMO systems with lowresolution ADCs and lattice reduction aided detectors," *Symmetry*, vol. 12, no. 3, p. 406, 2020.
- [13] S. Jacobsson, G. Durisi, M. Coldrey, U. Gustavsson, and C. Studer, "Throughput analysis of massive MIMO uplink with low-resolution ADCs," *IEEE Transactions on Wireless Communications*, vol. 16, no. 6, pp. 4038–4051, 2017.
- [14] M. B. Mashhadi and D. Gündüz, "Deep learning for massive MIMO channel state acquisition and feedback," *Journal Indian Institute of Science*, vol. 100, no. 2, pp. 369–382, 2020.
- [15] A. Ly and Y.-D. Yao, "A review of deep learning in 5G research: channel coding, massive MIMO, multiple access, resource allocation, and network security," *IEEE Open Journal* of the Communications Society, vol. 2, pp. 396–408, 2021.
- [16] J. Gao, M. Hu, C. Zhong, G.-Y. Li, and Z. Zhang, "An attention-aided deep learning framework for massive MIMO channel estimation," *IEEE Transactions on Wireless Communications*, vol. 21, no. 3, pp. 1823–1835, 2022.
- [17] M. Meenalakshmi, S. Chaturvedi, and V. K. Dwivedi, "Performance analysis of polar codes in 5G new radio," in *International Conference on Signal Processing and Communication*, pp. 96–99, Noida, India, 2021.
- [18] M. Meenalakshmi, S. Chaturvedi, and V. K. Dwivedi, "Deep learning techniques for OFDM systems," *IETE Journal of Research*, pp. 1–15, 2021.
- [19] Y. Zhang, M. Alrabeiah, and A. Alkhateeb, "Deep learning for massive MIMO with 1-bit ADCs: when more antennas need fewer pilots," *IEEE Wireless Communication Letter*, vol. 9, no. 8, pp. 1273–1277, 2020.
- [20] E. Balevi and J. G. Andrews, "Two-stage learning for uplink channel estimation in one-bit massive MIMO," in *Asilomar Conference on Signals, Systems, and Computers*, pp. 1764– 1768, Pacific Grove, CA, USA, 2020.
- [21] M. Kim, W. Lee, and D.-H. Cho, "Deep scanning-beam selection based on deep reinforcement learning in massive MIMO wireless communication system," *Electronics*, vol. 9, 2020.
- [22] Y. Dong, H. Wang, and Y.-D. Yao, "Channel estimation for one-bit multiuser massive MIMO using conditional GAN," *IEEE Communications Letters*, vol. 25, no. 3, pp. 854–858, 2021.
- [23] A. A. Al-Shargabi, J. F. Alshobaili, A. Alabdulatif, and N. Alrobah, "COVID-CGAN: efficient deep learning approach for COVID-19 detection based on CXR images using conditional GANs," *Applied Sciences*, vol. 11, no. 16, p. 7174, 2021.
- [24] K. Lata, M. Dave, and K.-N. Nishanth, "Image-to-image translation using generative adversarial network," in *International conference on Electronics, Communication and Aerospace Technology*, pp. 186–189, Coimbatore, India, 2019.
- [25] E. Balevi, A. Doshi, and J.-G. Andrews, "Massive MIMO channel estimation with an untrained deep neural network," *IEEE*

Transactions on Wireless Communications, vol. 19, no. 3, pp. 2079–2090, 2020.

- [26] E. Balevi and J.-G. Andrews, "Wideband channel estimation with a generative adversarial network," *IEEE Transactions on Wireless Communications*, vol. 20, no. 5, pp. 3049–3060, 2021.
- [27] J. D. Yadav, V. K. Dwivedi, and S. Chaturvedi, "Performance comparison of cGAN models for channel estimation in onebit massive MIMO system," in *IEEE Microwave Theory and Techniques in Wireless Communications*, pp. 296–300, Riga, Latvia, 2021.
- [28] S. Zagoruyko and N. Komodakis, "Wide residual networks," 2016, https://arxiv.org/abs/1605.07146.
- [29] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Conference on Computer Vision and Pattern Recognition*, pp. 7132–7141, Salt Lake City, UT, USA, 2018.
- [30] D. Jha, D. Johansen, T.-D. Lange, P. Halvorsen, and M.-A. Riegler, "A comprehensive study on colorectal polyp segmentation with ResUNet++, conditional random field and test-time augmentation," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 6, pp. 2029–2040, 2021.
- [31] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer Assisted Intervention*, pp. 234–241, Munich, Germany, 2015.
- [32] F.-I. Diakogiannis, F. Waldner, P. Caccetta, and C. Wu, "ResUNet-a: a deep learning framework for semantic segmentation of remotely sensed data," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 162, pp. 94–114, 2020.
- [33] Y. Zhou, K. Yu, M. Wang et al., "Speckle noise reduction for OCT images based on image style transfer and conditional GAN," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 1, pp. 139–150, 2022.
- [34] A. Alkhateeb, "DeepMIMO: a generic deep learning dataset for millimeter wave and massive MIMO applications," in *Information Theory and Applications Workshop*, pp. 1–8, San Diego, USA, 2019.
- [35] E. Balevi, A. Doshi, A. Jalal, A. Dimakis, and J.-G. Andrews, "High dimensional channel estimation using deep generative networks," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 1, pp. 18–30, 2021.
- [36] D.-P. Kingma and J.-L. Ba, Adam: A Method for Stochastic Optimization, International Conference on Learning Representations, 2015.
- [37] S. Ruder, "An overview of gradient descent optimization algorithms," 2016, https://arxiv.org/abs/1609.04747.