

A Conditional Generative Adversarial Network for Weather Radar Beam Blockage Correction

Songjian Tan, *Student Member, IEEE*, and Haonan Chen, *Senior Member, IEEE*

Abstract—Missing or low-quality data regions usually happen to weather radars. One of the most common situations is beam blockage or partial beam blockage. Therefore, correction of weather radar observations that are partially or fully blocked is an indispensable step in radar data quality control and subsequent quantitative applications, especially in complex terrain environments such as the western United States. In this article, we propose a deep learning framework based on generative adversarial networks (GANs) for restoring partial beam blockage regions in polarimetric radar observations using local and global contextual information. Due to the diverse precipitation types, blockage conditions, and ground information in different areas, two radars deployed in two different regions characterized by different precipitation types are used to demonstrate the proposed methodology. Both are S-band operational Weather Surveillance Radar – 1988 Doppler (WSR-88D): KFWS located in Fort Worth, northern Texas, and KDAX located in Davis, northern California. For training the GAN model, this article simulates the partial beam blockage situations by manually cropping observation sectors of both KDAX and KFWS radar data. The trained models were tested using independent precipitation events in Texas and California to demonstrate the model effectiveness in inpainting “missing” data. In addition, this paper cross-tested the data with different precipitation features to examine the generalization capacity of the beam blockage correction models. The beam blockage correction performance is also compared with a traditional linear interpolation approach. The results show that for both domains the continuity of precipitation observations is greatly improved after applying the deep learning-based inpainting approach. For the KFWS test data, some visible discrepancies exist between the results from models trained based on convective and stratiform precipitation events in Texas and California, respectively, yet both models outperform the traditional interpolation method. For the KDAX test data, both the model trained using the KFWS data from convective precipitation events in Texas and the model trained using KDAX data from stratiform precipitation events in California render a similar performance. Although ground truth is not available for the real blocked radar data, the repaired observations demonstrated a great potential for improved quantitative applications.

Index Terms—Radar data inpainting, deep learning, beam blockage, weather radar.

I. INTRODUCTION

WEATHER radar is an indispensable tool for monitoring precipitation patterns, storm movements, and extreme weather events such as tornadoes. The information gathered by weather radar is the most intuitive way to describe various

weather phenomena. Recent years have also seen advances in weather radar technology, allowing for the collection of more detailed and accurate atmospheric conditions. In practice, however, radar signals are often of low quality or missing due to physical occlusion and/or radio frequency interference. To minimize the impact of these problems and other data quality issues, the radar system employs various techniques such as adaptive signal processing, advanced calibration methods, and attenuation correction algorithms.

Nevertheless, a number of radar data problems still exist even after applying advanced radar processing techniques. The most common of these is beam blockage or partial beam blockage (PBB) [1], which is caused by the presence of physical objects such as tall buildings or mountains within the radar beam, thus creating a wedge-shaped blind spot at the back end of the signal beam which can lead to low-quality application products and inaccurate weather forecasts and warnings [2]. To mitigate radar beam blockage, weather radar systems are often placed at elevated locations, such as on top of hills or high buildings, to reduce the beam blockage effect. In addition, radar networks are often designed to ensure that precipitation is detected from multiple angles [3][4][5], thus improving overlapped coverage areas from multiple radars and reducing the chances of missing major weather echoes. Despite these strategies, radar beam blockage can still occur in some cases, such as when a thunderstorm or tornado approaches a radar station (i.e., not observed by other radars in the network environment). In such cases, meteorologists rely on other sources of weather information, such as ground-based observations, satellite imagery, or traditional radar image restoration methods [6]. This article aims to offer a new strategy for beam blockage correction in weather radar data.

On the other hand, image correction has long been a research area in the field of computer vision and image processing. It is a technique used to fill in missing or corrupted parts of an image by synthesizing new pixels based on the surrounding pixels. For radar image restoration, many different approaches have been applied, including traditional interpolation and statistical methods. While interpolation[7] and statistical methods [8] can be effective for image inpainting, they also have some limitations and disadvantages. Since interpolation methods rely on smoothness assumptions and may not be able to capture the texture and structure information in the image, they may produce blurry or unrealistic results, especially when the missing region is large or complex. In addition, interpolation methods may not work well when the missing pixels are located at the edges or corners of the image, as they may introduce artifacts or distortions in the image. Statistical

This research was supported by the U.S. National Science Foundation (NSF) Faculty Early Career Development Program (CAREER). (*Corresponding author: Haonan Chen.*)

Songjian Tan and Haonan Chen are with the Department of Electrical and Computer Engineering, Colorado State University, Fort Collins, CO 80523, USA (e-mail: haonan.chen@colostate.edu).

methods may not work well when the missing pixels are part of complex or high-frequency structures, as they may not be able to capture the details and textures of the image. Also, statistical methods may require prior knowledge or assumptions about the image content, which may not always be available or accurate [9].

In recent years, owing to the latest breakthroughs in deep learning in the computer vision field, there have been different advances in image recognition, image classification, image segmentation, as well as image inpainting [10]. Deep learning-based image inpainting is a rapidly developing field that has shown great promise in restoring damaged or missing image information. Most of the earliest deep learning-based image inpainting methods were based on autoencoders, which could learn the low-dimensional representation of images and then modify the encoded low-dimensional vectors for image restoration. However, these methods were limited in their ability to capture the global structure and texture information of images, and the results were often unsatisfactory. Elad and Aharon developed a convolutional neural network (CNN) to learn the mapping from the masked image to the restored image for image inpainting [11], which achieved promising results in generating high-quality restored images for simple images, but it was not able to handle complex image structures and texture information. Since the introduction of generative adversarial networks (GANs) [12], GANs are widely used in various applications including image inpainting. Essentially, GANs use two deep neural networks: a generator network and a discriminator network [12]. The generator network could generate the restored image, while the discriminator network would evaluate the quality of the restored image and provide feedback, enabling the generator network to generate a more realistic restored image. However, one of the limitations is that the traditional GANs can be unstable and can potentially produce artifacts in the restored images [13].

CGAN offers several advantages over GAN, primarily the ability to generate images based on additional input information, such as the presence and location of radar beam blockage. This allows for more accurate and realistic image generation that takes into account specific input features. In contrast, GAN only generates images based on random noise input. Additionally, CGAN is better equipped to handle imbalanced data distributions that may be present in radar beam blockage correction tasks. GAN may have difficulty generating high-quality samples for the minority class i.e., blocked areas, resulting in a bias towards the majority class i.e., unblocked areas, and inaccurate correction results. The conditional input used in CGAN helps to balance the data distribution, ensuring that both blocked and unblocked areas are equally represented in the generated output. This article aims to build a conditional GAN (CGAN) framework for weather radar beam blockage correction. This CGAN model can more accurately restore the structure and texture information of images while reducing artifacts and blur in the restored images, thus is more suitable for complex weather radar data processing. A specialized model adapted to the variations of the radar operation environment is designed to address the unique characteristics of radar data. The model is trained and tested using weather data collected

from two S-band operational Weather Surveillance Radar – 1988 Doppler (WSR-88D) systems deployed in two different precipitation regimes: KFWS radar in northern Texas which is characterized by convective precipitation, and KDAX radar in northern California, which is typically featured by shallow stratiform precipitation. In addition, this paper performs multifarious blockage simulations to solve beam blockage correction problems with diverse blockage sizes.

The structure of the remaining sections of this article is as follows. Section II describes the research domain, dataset, and data preprocessing before training the deep learning model. Section III details the Unet++ and CGAN-based deep learning architecture for radar beam blockage correction. Section IV analyzes and evaluates the results obtained from different models. In Section V, a thorough discussion of the correction model is provided, and Section VI summarizes the main findings of this study.

II. STUDY DOMAIN AND DATASETS

As mentioned, to improve the accuracy and robustness of the correction model, study domains characterized by different precipitation features, i.e., convective and stratiform precipitation, will be selected so as to enrich the radar data characteristics and enhance the model inpainting capability. This article selects two distinct regions as study domains. The first of these is Northern California, characterized by a Mediterranean climate featuring damp winters and parched summers. The primary source of precipitation in this region is derived from Pacific Ocean storms, which frequently generate stratiform precipitation that is evenly dispersed across a broad area [14]. In contrast, northern Texas experiences a humid subtropical climate characterized by hot summers and mild winters. This arid climate fosters the development of more powerful, localized storms that generate typical convective precipitation, which may result in thunderstorms and heavy downpours [3][4]. Two National Weather Service (NWS) Weather Surveillance Radar—1988 Doppler (WSR-88D) radar systems were chosen within the selected study domains, namely, KDAX radar deployed in Davis near Sacramento in Northern California, and KFWS radar deployed in the Dallas-Fort Worth area in northern Texas. Both KDAX and KFWS are Doppler weather radars that can detect the velocity and direction of weather objects, including the location and intensity of the precipitation. Their maximum detection radius spans approximately 460 km for reflectivity measurements and 230 km for Doppler measurements (<https://nap.nationalacademies.org/read/10394/chapter/11>). Due to beam broadening and elevated beam height as the distance from the radar increases [15], only radar data within the 230 km coverage range are considered in this study. The range and azimuth angle resolution are 0.25 km and 0.5 degrees, respectively, resulting in a data matrix with a size of 920×720 for a full surveillance scan (see Fig. 1a).

It has been determined that the vast majority of deep learning algorithms demand a substantial quantity of data for practical training, coupled with the high-resolution nature of radar imagery that necessitates significant computational resources and time for complete processing, and the challenging

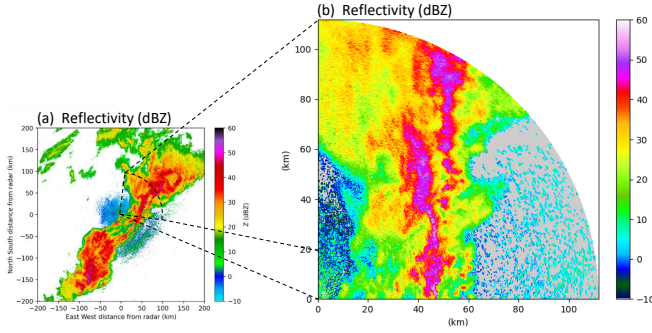


Fig. 1. Example of the KFWS radar observations at 0555UTC, March 1, 2018: (a) original reflectivity measurement; (b) preprocessed reflectivity for a selected subdomain.

task of collecting measurement data free of noise and interference, we have opted to preprocess the raw data. Initially, to incorporate more comprehensive precipitation information, we utilized the KFWS and KDAX radar data from May to September during 2019-2021 as the training set[16]. During this stage, a sequence of steps is employed to prepare the data for further analysis. First, high precipitation information exceeding 25 dBZ, with a continuous length of 8 km or more and within a 15-degree range, would be selected for further analysis. Subsequently, to minimize data uncertainty and blank or low-quality data included, we have chosen to crop the original data size to a new subdomain with the size of 448 (range) $\times$ 192 (azimuth), as depicted in Fig. 1(b), which has been sized for facilitating the convolution operation at a later stage. To enhance the model's robustness, we have rotated and flipped the selected subdomain within the same precipitation event to create pseudo radar observations. Therein, to focus the algorithm more intently on important data features and guarantee the PBB simulated area encompasses uninterrupted precipitation., we assigned that the probability distribution characterizing the location of the selection center corresponds to the magnitude distribution of the reflectivity data contained within the precipitation information during rotation. After preprocessing, the complete dataset will be arranged in descending order according to the cumulative reflectivity information. We extracted the first 10,000 cases of data from each of the two radars from the preprocessed dataset as the final dataset. The first 80% of this data will be allocated for training purposes, and the remaining 20% will be used for testing.

To expedite the convergence of the training data, decrease the possibility of overfitting, and simplify the simulation of PBB masks in the precipitation data, we normalized the preprocessed precipitation data to conform to the target range of -1 to 1 using the following normalization equation [17].

$$x = \frac{1.5 \times (x - \min)}{\max - \min} - 0.5 \quad (1a)$$

$$x[x \leq -0.5] = -1 \quad (1b)$$

$$x[x > 1] = 1 \quad (1c)$$

Note that the reflectivity values in the raw precipitation data range between -10 and 60 dBZ. However, we have established the minimum and maximum limits (i.e., min and max) in equation (1a) as 0 and 60 dBZ, respectively. Consequently, reflectivity data are linearly scaled to fall within the range of [-0.5, 1.0], whereas any reflectivity values below 0 dBZ would be normalized to -1 using (1b), which is because the convolution operation and edge-filling approach tend to smooth out image edges, including cloud edges in radar data, resulting in irregular artifacts in the output feature maps at the edge locations, which can negatively impact the model's training during CNN training [18]. As a result, all precipitation data that falls below -0.5 was treated as clear day data with a value of -1 in Eq. 1b. This 0.5 difference will emphasize the clouds while enabling the model to obtain more effective gradient information at the edge locations. Finally, Eq. 1c applied a truncated normalization process to incorporate the dataset's rare high-reflectivity data outside the normal range.

III. METHODOLOGY

In this section, we will manually simulate the partial beam blockage for the preprocessed data, followed by an explanation of the network layer structure and internal data processing methods. Finally, we will post-process the output data to obtain the results.

A. PBB Simulation

This subsection outlines the manual creation of beam blockage conditions for convective and stratiform precipitation data, wherein specific radial profiles are rendered blank. During the preprocessing stage, the image center locations are determined based on the reflectivity scaling probabilities. To create the PBB mask, a specific region in the center of the image ranging from 4-16 degrees azimuth and starting position in the range 0-248 are designated. This PBB mask region is assigned a value of 1, while the remainder of the data is set to 0. Following this, the normalized data undergo blockage simulation based on the prescribed equation as shown in Eq. 2, allowing the neural network to learn and predict precipitation patterns under these simulated blockage conditions.

$$x_{masked} = x \times (1 - mask) - mask \quad (2)$$

Fig. 2 illustrates the manual process of simulating beam blockage and the variations in the data boundary conditions. The original data is depicted in (a), where reflectivity information is used as an example. A randomly sized mask is shown in (b). As mentioned earlier, the region corrected by the neural network often contains artifacts at the edges. To preserve the details of this portion and prevent unwanted noise, a buffer is added for smoothing with a range value of 1-5 degrees, i.e., the portion between the mask and the blank area in the figure. By fusing Eq. (2), a new image after manual simulation of beam blockage is obtained, as depicted in (c). This image contains

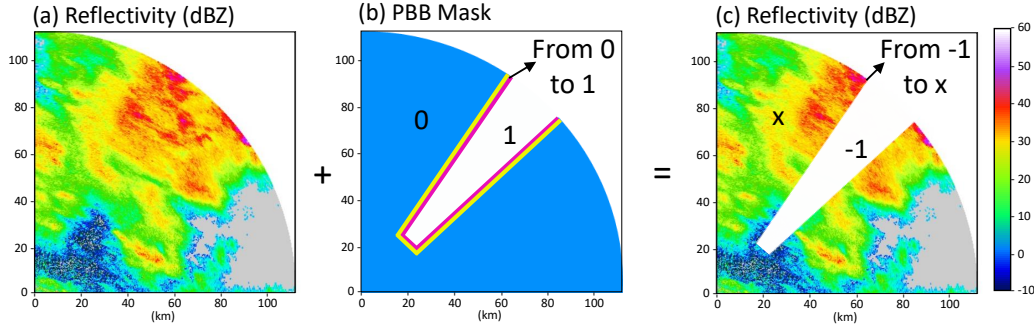


Fig. 2. Illustration of PBB simulation process: (a) reflectivity data, i.e., x in Eq. 2; (b) PBB mask i.e., $mask$ in Eq. 2; (c) simulated data with beam blockage, i.e., x_{masked} in Eq. 2.

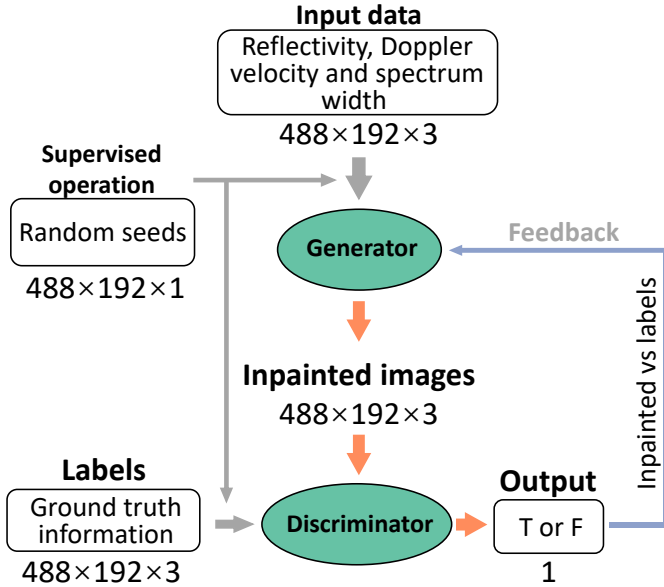


Fig. 3. CGAN-based deep learning framework for weather radar beam blockage correction.

the original image x , ranging from -1 to 1 at the periphery, and the simulated mask as clear air minimum data.

Moreover, in conjunction with the reflectivity data exemplified earlier, we incorporated two additional radar parameters, namely Doppler velocity and spectrum width, commonly employed in radar applications. These parameters underwent similar preprocessing, normalization, and masking procedures as the reflectivity data, and their details are not enumerated here. Currently, the model's input layers entail three data dimensions ranging from -1 to 1 and a mask dimension ranging from 0 to 1, collectively constituting four dimensions.

B. Deep Learning Model Architecture

The workflow of the proposed deep learning model for weather radar beam blockage correction is shown in Fig. 3. The gray arrows represent the input, the orange arrows represent the output, and the blue arrows represent the feedback layer. As previously stated, the reflectivity, Doppler velocity, and spectrum width data procured from KDAX and

KFWS radar, which will be fed into the network as the data layer, undergoes a standardization process, followed by a PBB simulation as shown in Section II and Section III-A. To enhance the model's robustness and generalizability by enabling it to learn from diverse noises and anomalies during training, we introduce a Conditional Generative Adversarial Network (CGAN) as our network architecture [19], necessitating a conditional supervising input layer. In this instance, we utilize Gaussian distributed data with a mean of 0 and a standard deviation of 0.5, which is input into the network at the same size as the radar data, concomitant with the data layer. Subsequently, the simulated input images and random seeds undergo several convolution and deconvolution layers within the generator to extract and generate features from the input. To this end, we utilize the Unet-based network layer as the generator, which exhibits commendable performance in feature extraction and upsampling, which will be detailed in Section III-B1. The images generated by the generator subsequently enter the discriminator along with the ground truth values, which are normalized but not simulated by beam blockage. Within the discriminator, the image and ground truth values undergo a sequence of convolution and pooling operations to extract the final features. These features are then transformed into binary outputs (i.e., true or false) via a final fully connected layer to complete the discrimination process, which is elaborated in Section III-B2. Notably, the discriminator's classification results are fed back to the generator to optimize the loss function via back-propagation, enabling the generator to generate more realistic images. This adversarial process continuously improves the model's performance, as the generator and discriminator confront each other in this manner.

1) *Generator*: As previously stated, the four layers of data simulated by the PBB are initially inputted into the generator. To enhance its nonlinearity and interpretability, a multi-channel convolution process is employed on the input data within the generator, as illustrated in Fig. 4.

In this process, the input data is a tensor with a size of $448 \times 192 \times 4$, which includes three layers of data representing reflectivity, Doppler velocity, and spectrum width, along with one layer of random seed. Initially, a convolution kernel of size $3 \times 3 \times 4$ with a SAME boundary padding and stride of 1 is utilized for convolution in the horizontal direction. The 4-

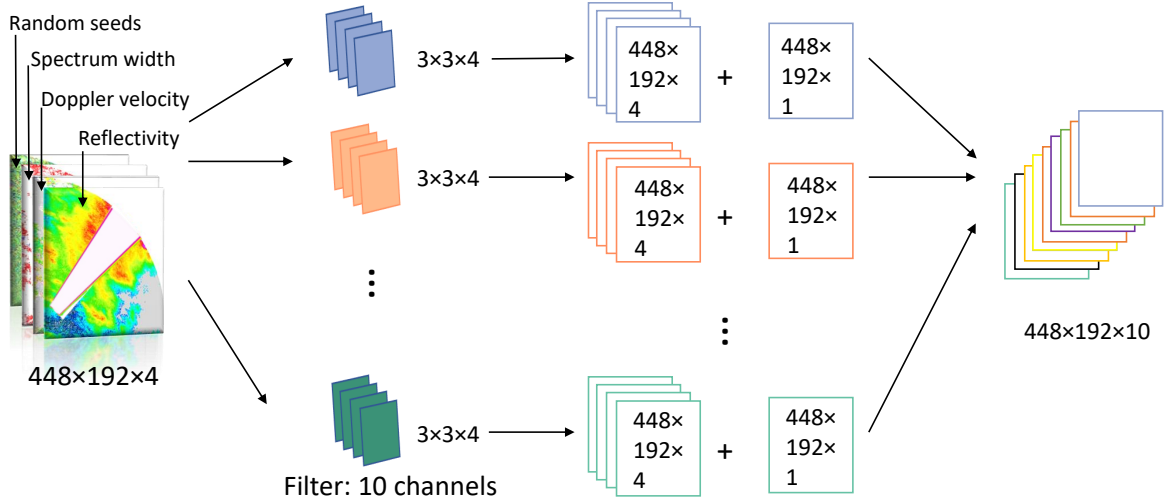


Fig. 4. Multi-channel convolution of input data layers.

dimensional convolution kernel slides across each dimension of the input image, computes the convolution value for each position, and generates a $448 \times 192 \times 4$ feature map. The convolution operation results are summed arithmetically for these four dimensions to obtain a $448 \times 192 \times 1$ feature map. Then, ten different convolution kernels are applied in parallel to extract various features. This process is repeated ten times to produce ten different feature maps. Finally, these ten layers of feature maps are superimposed and input into the Unet-based neural network.

The data characterized by different feature maps are fed into the Unet-based neural network, where the Unet network [20] initially developed for image segmentation, exhibits a typical encoding-decoding architecture. The encoder situated in the first half leverages the pooling layers for gradual down-sampling, augmenting resistance against minor perturbations like image translation and rotation, reducing overfitting risk, minimizing operation numbers, and amplifying the perceptual field's size. The other half is the decoder, which performs up-sampling using deconvolution layers, thus restoring the decoded features to their original size and allowing the spatial and edge information of the input image to be gradually regained. Consequently, the low-resolution feature maps get mapped to the pixel-level segmentation result maps. The U-Net algorithm further compensates for the lost information during down-sampling at the encoding stage using a skip connection between the encoder and decoder, which combines the feature maps at corresponding positions in both processes, equipping the decoder with additional high-resolution information during up-sampling, resulting in a more precise recovery of detailed information in the original image, thus ameliorating the segmentation accuracy.

However, empirical investigations have demonstrated that the appropriate number of layers of the Unet network required for various precipitation models is not consistent [21] [22]. To better capture the global context at different scales for translating unsampled images, we have implemented an enhanced model, Unet++ [23], as a generator. This model retains the

benefits of Unet while allowing for the precise integration of features at varying network levels and a significant reduction in the number of parameters.

Unet++ is an extension of the original Unet that introduces a nested and dense skip pathway, enabling the model to capture multi-scale contextual information by integrating features from different levels of the network, which improves the model's capacity to generate accurate and detailed output images. Additionally, the dense skip connections in Unet++ augment the information flow between the encoder and decoder, thereby facilitating the recovery of more spatial information in the output image. Moreover, Unet++ utilizes fewer parameters and is more efficient than other models. Overall, incorporating Unet++ as the generator for the proposed deep learning model for weather radar beam blockage correction can enhance its performance and robustness, where the operation of the data in the Unet++ based generator network layer is detailed in Fig. 5.

Among these, after the multi-channel convolution operation of the previous stage, ten feature maps with a size of 448×192 will be input. The data will then undergo down-sampling as illustrated in the left half of Fig. 5. Each down-sampling computational unit, i.e., the computing module linked to the down-sampling arrow in the figure, comprises three distinct components. The initial step of the down-sampling calculating unit involves a two-dimensional convolution operation, utilizing a 3×3 convolution kernel, along with SAME boundary fill padding and a stride size of 1. The purpose of This convolution operation aims to train the kernel weights to detect various features across various precipitation events, encompassing cloud edge features for less intense precipitation data like KDAX radar data, and cloudiness texture features for more intense precipitation data such as KFWS radar data [24]. The second component comprises a Leaky Rectified Linear Unit (Leaky ReLU) activation layer featuring a negative partial slope of 0.2 that imparts nonlinearity to the linear weighting while circumventing the issue of vanishing gradients. This results in a superior fit of the precipitation data distribution in more

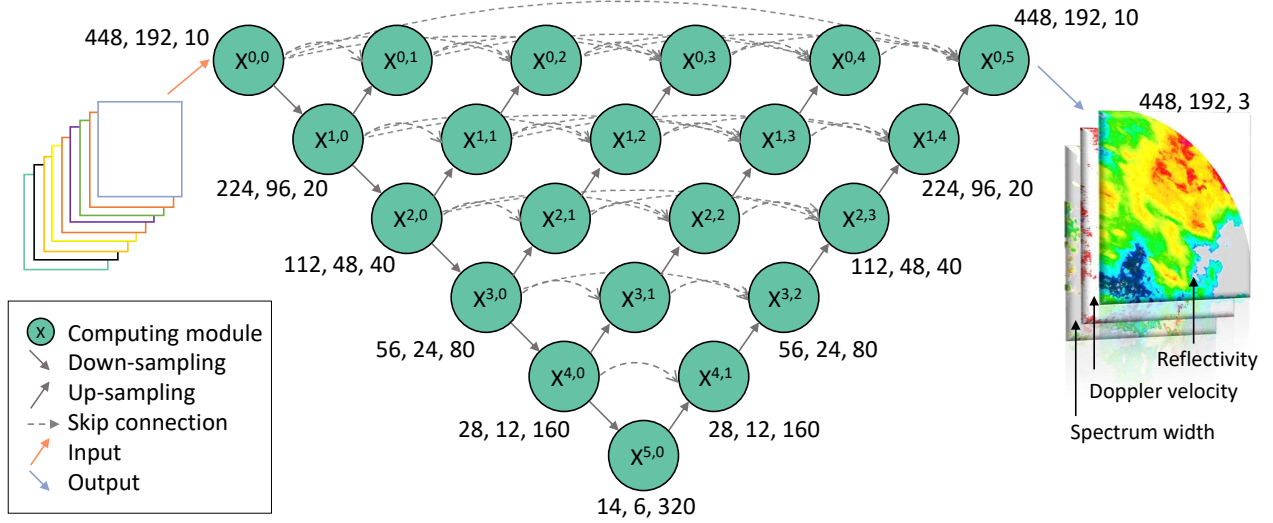


Fig. 5. Unet++ based generator neural network applied in the weather radar beam blockage correction system.

intricate radar data, in line with our model's requirements. The third part comprises a maximum pooling layer with a filter size of 2×2 . The layer operates by dividing the input image into several blocks of 2×2 size and selecting the maximum value in each block as the output. Despite the screening of input data, it still contains a significant amount of noise in the training data. The maximum pooling aids in extracting critical features from the input image, while eliminating noise and superfluous details, resulting in an enhanced restoration effect. Additionally, the process aids in reducing the spatial dimension of the feature mapping for large volumes of radar data in the model, reducing the number of parameters and computational complexity. The aforementioned process will be repeated five times in a sequence of down-sampling, where the size of the radar data is halved while doubling the dimensionality at each step, which results in a final output of high-dimensional data with the most distinctive features at the end of down-sampling, measuring $14 \times 6 \times 320$.

During the decoding stage of upsampling, a skip connection is established between the down-sampled and up-sampled computing modules. The primary function of this connection is to duplicate the data output generated by the down-sampling pooling layer and combine it with the up-sampling data in the channel dimension through concatenation, which results in twice the number of feature maps in the output. The diminution in feature information during the pooling procedure of down-sampling necessitates the utilization of skip connections to integrate the feature maps before down-sampling with those after up-sampling, thereby preserving the coherence of image features. Moreover, since our correction model utilizes precipitation data from targets with distinct features, feature information at varying scales is required to capture image features more effectively. Unet++ skip connections can proficiently extract multiple-scale feature information by incorporating numerous feature maps and uniting them through skip connections, thus leading to superior image reconstruction outcomes. Multiple skip connections can also

diminish the influence of noise in training data and circumvent overfitting predicaments.

The subsequent process is the upsampling stage on the right side of Fig. 5, where each upsampling computing module comprises four components. The first component is the deconvolution operation with a filter size of 2×2 , a stride of 2, and a dilation of 1, which is for increasing the size of the up-sampled input tensor and restoring it to the original input size while retaining the feature information of the original image. In our correction model, as the output of down-sampling is high-dimensional multi-feature data, deconvolution operation can also refine the spatial information of the feature map, thereby enhancing the precision and stability of the model. However, the deconvolution operation is solely intended to recover the image dimension and some of the features, and therefore is not the complete inverse operation of pooling. As a result, the data copied from the skip connection is necessary to fill the feature gap effectively. Therefore, a merging process is needed for these two components in the second part of the upsampling unit, i.e., the concatenation operation. The merged data contains both up-sampled and down-sampled feature information, which can aid the network in better grasping the contextual and semantic information in the image, consequently enhancing the accuracy and effectiveness of image segmentation or restoration. The third component is a convolutional layer with a filter size of 3×3 . As the merged data from the previous stage possesses twice the number of data channels, we require a multi-channel convolutional layer with half the number of channels. Lastly, a similar Leaky ReLU layer is used in down-sampling. This is because the deconvolution operation results in information loss and blurring, necessitating this layer following the deconvolution to increase the nonlinear representation of the network, thereby improving its ability to recover the details and structure of the original image. After five upsampling stages, the data will undergo three-channel convolution with the sigmoid activation function to obtain the corrected data.

During training, the generator's task is to produce output images that closely resemble the real images and successfully fool the discriminator. A loss function is employed to train the generator and update its weight parameters through back-propagation. Equation (3) specifies the exact function used.

$$\mathcal{L}_G = \lambda E_{x,y,z}[|y - G(x,z)|] - E_{x,z}[\log(D(x, G(x,z)))] \quad (3)$$

The generator's loss function is divided into two components. The first component is the $L1$ loss, also known as the mean absolute error (MAE) loss function. In this function, x refers to the input data, which is the radar image generated by PBB simulation, y refers to the corresponding ground truth image, which is the original radar image, z refers to an input random seed, which is the Gaussian distribution random seeds we input, and G is the generator prediction. Thus, this $L1$ loss measures the average absolute difference between the predicted image $G(x,z)$ and the ground truth image y for all input images x , their corresponding ground truth images y , and condition z . The second component is the adversarial loss, which trains the generator to outsmart the discriminator. Here, D denotes the discriminator output, and the natural logarithm is represented by 'log'. The minus sign is used to convert the maximization problem of the discriminator into a minimization problem for the generator. This component computes the average logarithm of the probability that the generator's output image $G(x,z)$ can deceive the discriminator D , considering all input images x and conditions z . In other words, it measures the discriminator's discriminatory outcome for the real image versus the generated image by the generator. In essence, the objective of the two components of the generator is to impart the ability to create true and real images which can surpass the discriminator, while concurrently minimizing the divergence between the inpainted and ground truth images. To prevent issues such as overfitting, a hyperparameter lambda value of 8 is incorporated to regulate the weight distribution between the two components.

2) *Discriminator*: The corrected radar data of size $448 \times 192 \times 3$ output from the generator will be fed into the discriminator, along with the ground truth data of the same dimensions. The specific process is shown in Fig. 6. The primary processing of the discriminator can be broadly categorized into four stages, beginning with the convolutional layer. Similarly to the generator, a filter of 3×3 is used for convolutional processing. However, unlike the generator's convolutional layer, which aims to capture various features to infer missing pixels, the discriminator's convolutional layer focuses on detecting the same class of features to facilitate comparison with the true values. In the case of radar data, the initial convolutional layers focus on extracting low-level features such as edges and textures, representing cloud patches and noise in the data, which allows the subsequent network to learn higher-level features. The middle layers extract mid-level features such as shape and structure, which aid in separating precipitation information from irrelevant details. The latter convolutional layers extract high-level features that are critical in comparing precipitation size and location in the target region. To further improve the generalization ability of the

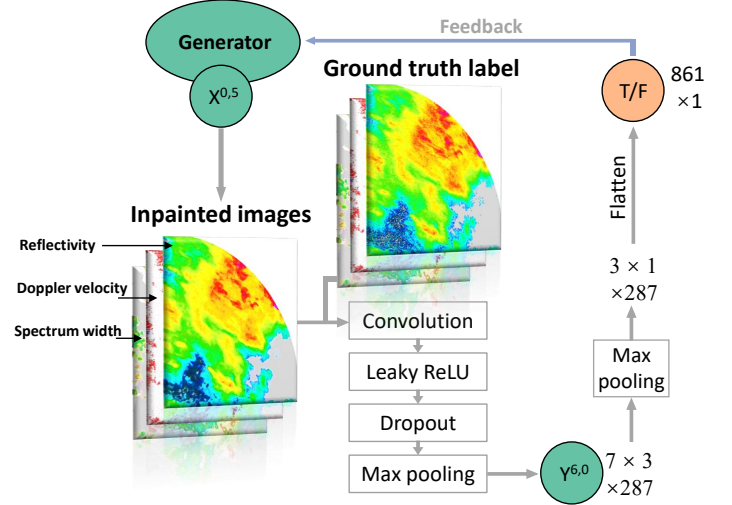


Fig. 6. Discriminator network applied in the CGAN-based weather radar beam blockage correction system.

discriminator and prevent overfitting, the second part of the discriminator also undergoes a Leaky ReLU layer with a negative partial slope of 0.2. Next, a dropout layer is needed to randomly deactivate some neurons to decrease the co-adaptation between neurons in the network, thus enhancing the model's ability to generalize and prevent overfitting. A dropout layer with the rate of 0.1 employed in our correction model is beneficial in reducing the discriminator's reliance on specific local features, which mitigates the risk of the discriminator being excessively sensitive to the repaired image. Consequently, this layer facilitates the discriminator's comprehension and characterization of the entire radar image. The fourth part involves a 2×2 pooling operation that can down-sample the radar image data in the model, thereby extracting the most important features while reducing the spatial size of the data, which helps the discriminator to identify the relevant feature class better. The down-sampling calculating module consisting of the above four parts is integrated within the discriminator, resulting in the processed data size of $7 \times 3 \times 287$ after six layers of this unit. The processed data is then output as $3 \times 1 \times 287$ via the last maximum pooling layer, enabling the transformation of local information in the image into higher-level features for the determination of global features in the final fully connected layer. Subsequently, the feature map is flattened into a one-dimensional vector and mapped to a scalar output value ranging from 0 to 1 via the sigmoid fully connected layer, thus denoting the degree of similarity between the corrected radar image and the ground truth radar image. A value closer to 1 indicates higher similarity, whereas a value closer to 0 indicates higher dissimilarity. The discriminator generates real or fake feedback representing the difference between the restored radar image and the real radar image, which is then fed to the generator to evaluate the restoration

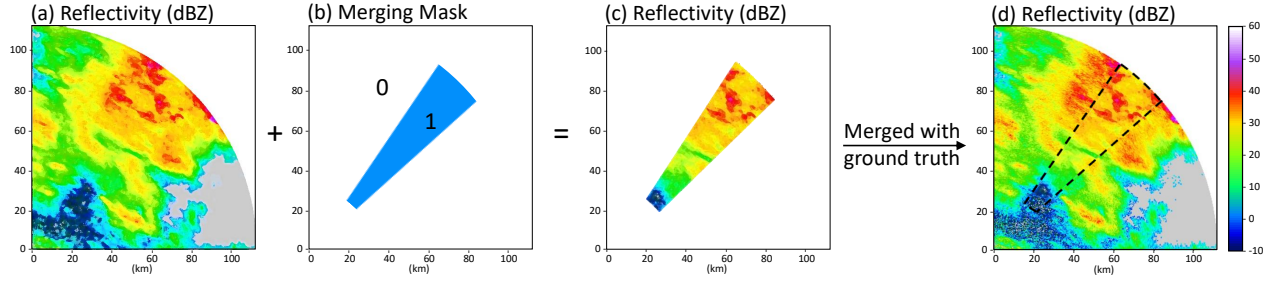


Fig. 7. Output merging process: (a) example of corrected data, i.e., x_{out} in Eq. 5; (b) merging mask, i.e., $mask$ in Eq. 5; (c) inpainted data in PBB region; (d) merged data, i.e., x_{merged} in Eq. 5.

effectiveness.

To provide better guidance for the generator in generating more realistic data, a discriminator loss function is used during training, as expressed in Equation 4.

$$\mathcal{L}_D = -E_{x,y}[\log(D(x,y))] - E_{x,z}[\log(1 - D(x, G(x,z)))] \quad (4)$$

The loss function of the discriminator can be bifurcated into two distinct components: the former pertains to the loss incurred on actual data and their corresponding labels, while the latter entails the loss on the data and labels generated by the generator. In the former component, the discriminator assesses the degree of correspondence between the actual data and labels through its probability output, denoted as $D(x, y)$, where x and y correspond to the input data and label, respectively. As the actual data and labels ought to exhibit high correspondence, a lower loss function value is obtained when the probability of the discriminator output approaches 1, with a minimum value of 0. In the latter component, the loss function evaluates the degree of correspondence between the generated data and labels. Here, x refers to input data, z denotes input random noise, $G(x, z)$ indicates the data and labels generated by the generator, and $D(x, G(x, z))$ represents the probability of the discriminator outputting the generated data and labels. Since the generated data and labels should have low correspondence, a smaller loss function value is obtained when the probability of the discriminator output is closer to 0, with a minimum value of 0. By fusing these two components, the discriminator effectively assesses whether the corrected radar image and ground truth match and outputs the corresponding probability, which is subsequently employed as a component of the generator's loss function to facilitate realistic data and label generation. Notably, our CGAN-based model trains the discriminator and the generator in an alternating manner, where the former is trained first, and its feedback guides the training of the generator in each iteration round.

C. Post-processing: Merge Output

To better simulate the masked condition in practical scenarios and enable a more thorough evaluation of the PBB component, an overlay merging operation will be executed once the correction data has been generated. This operation, as depicted in Fig. 7 with the mask Equation (5), aims to

combine the output of the correction process with the original data to generate a merged image that represents the corrected areas within the context of the original image.

$$x_{merged} = x_{out} \times mask + x \times (1 - mask) \quad (5)$$

Fig. 7 illustrates the merging process of corrected radar data and the variations in the data boundary conditions. The generator produces a corrected radar image as shown in Fig. 7(a), with reflectivity information used as an example. Fig. 7(b) illustrates a merging mask that is of the same size as the PBB mask layer in the data, as depicted in Fig. 2(b). By using the first half of the merging equation, the corrected data that corresponds to the PBB size can be obtained, as shown in Fig. 7(c). The variable x in the second half of the equation denotes the corresponding ground truth data, which is depicted in Fig. 2(a). After the merging operation, a corrected image containing PBB position information can be obtained, as shown in Fig. 7(d).

D. Comparison Model

In order to evaluate the effectiveness of the CGAN-based model, we also included a comparison group that uses a one-dimensional linear interpolation method for correction. This method involves performing linear interpolation from the PBB mask boundary, in dimensions with respect to the azimuth, as specified in Equation (6).

$$y = y_0 + (y_1 - y_0) \frac{x - x_0}{x_1 - x_0} \quad (6)$$

In this equation, x_0 and x_1 represent the known locations of boundaries, y_0 and y_1 represent the precipitation intensity corresponding to those locations, while x and y represent the locations and results of the points to be interpolated, respectively. This formula allows for the estimation of the value of a point between two known points based on their respective values.

IV. RESULTS AND EVALUATION

As previously stated, to enhance the adaptability and generalization of the model across various precipitation patterns and thereby augment the predictive accuracy of the model, a total of three models were employed. Specifically, these comprise the convective precipitation CGAN correction model based

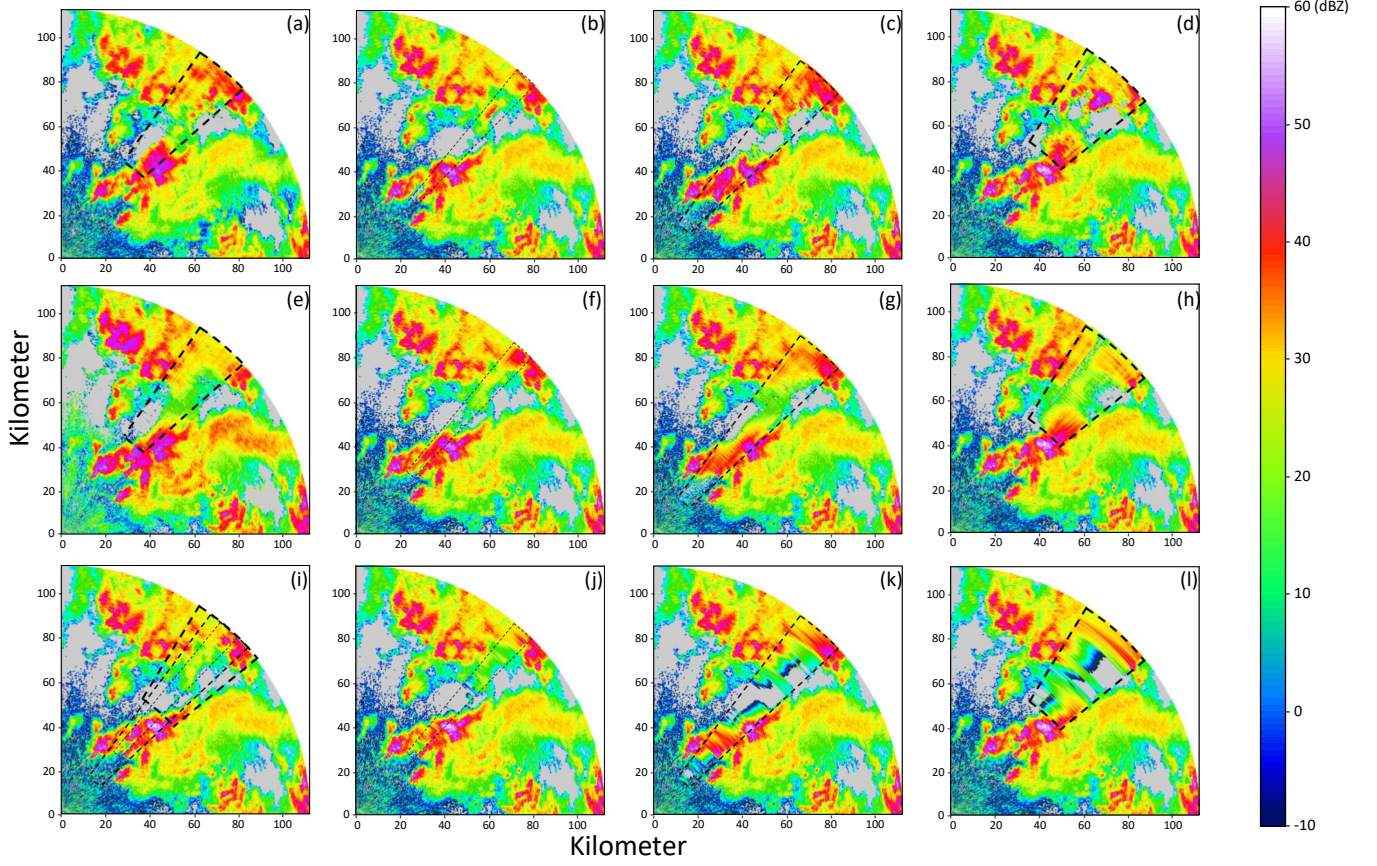


Fig. 8. KFWS test event at 2000UTC, May 16, 2021: (a)-(d) CGAN output with convective (KFWS) training model of random size, 4°, 10° and 16° PBB mask; (e)-(h) CGAN output with stratiform (KDAX) training model of random size, 4°, 10° and 16° PBB mask; (i) ground truth; (j)-(l) linear interpolation output of 4°, 10° and 16° PBB mask.

on KFWS data, the stratiform precipitation CGAN correction model based on KDAX data, and the conventional linear interpolation correction model. In the interest of concision, we designate the machine learning algorithm for KDAX training data as the CA model and for KFWS training data as the TX model. Therein, two CGAN models have been trained using 10,000 precipitation cases each from KDAX radar and KFWS radar, obtained from May to September during 2019-2021 in Northern California and Northern Texas, respectively. As a result of the varying precipitation intensities in the data sets, the learning rates of the network layers for the precipitation features are not uniform, which results in differing convergence rates of the two training models. To address this, the TX model was trained for 20,000 iterations, and the CA model was trained for 30,000 iterations, with learning rate decay applied at 14,000 and 19,000 iterations, respectively.

In order to comprehensively evaluate the performance of the proposed precipitation correction models, we tested a diverse set of precipitation data from May 2017 and May 2021. The test data included random azimuthal PBB masks as well as fixed angle masks of 4, 10, and 16 degrees to enable more rigorous evaluation and comparison of the model's efficacy. Additionally, we performed cross-testing between the two

CGAN models to identify the efficacy of correcting various types of precipitation. The same set of tests was also applied to the linear interpolation comparison group to ensure fair comparison and evaluation.

To assess the effectiveness of the proposed beam blockage correction model, this article employed four statistical hypothesis tests to analyze the true positive (TP), false negative (FN), true negative (TN), and false positive (FP) evaluations. Additionally, we applied four commonly used evaluation metrics, namely probability of detection (POD), false alarm rate (FAR), critical success index (CSI), and Heidke skill score (HSS), to measure the model's performance in detail. The calculation methods for each of these metrics are described below.

$$\text{POD} = \frac{TP}{TP + FN} \quad (7a)$$

$$\text{FAR} = \frac{FP}{TP + FP} \quad (7b)$$

$$\text{CSI} = \frac{TP}{TP + FN + FP} \quad (7c)$$

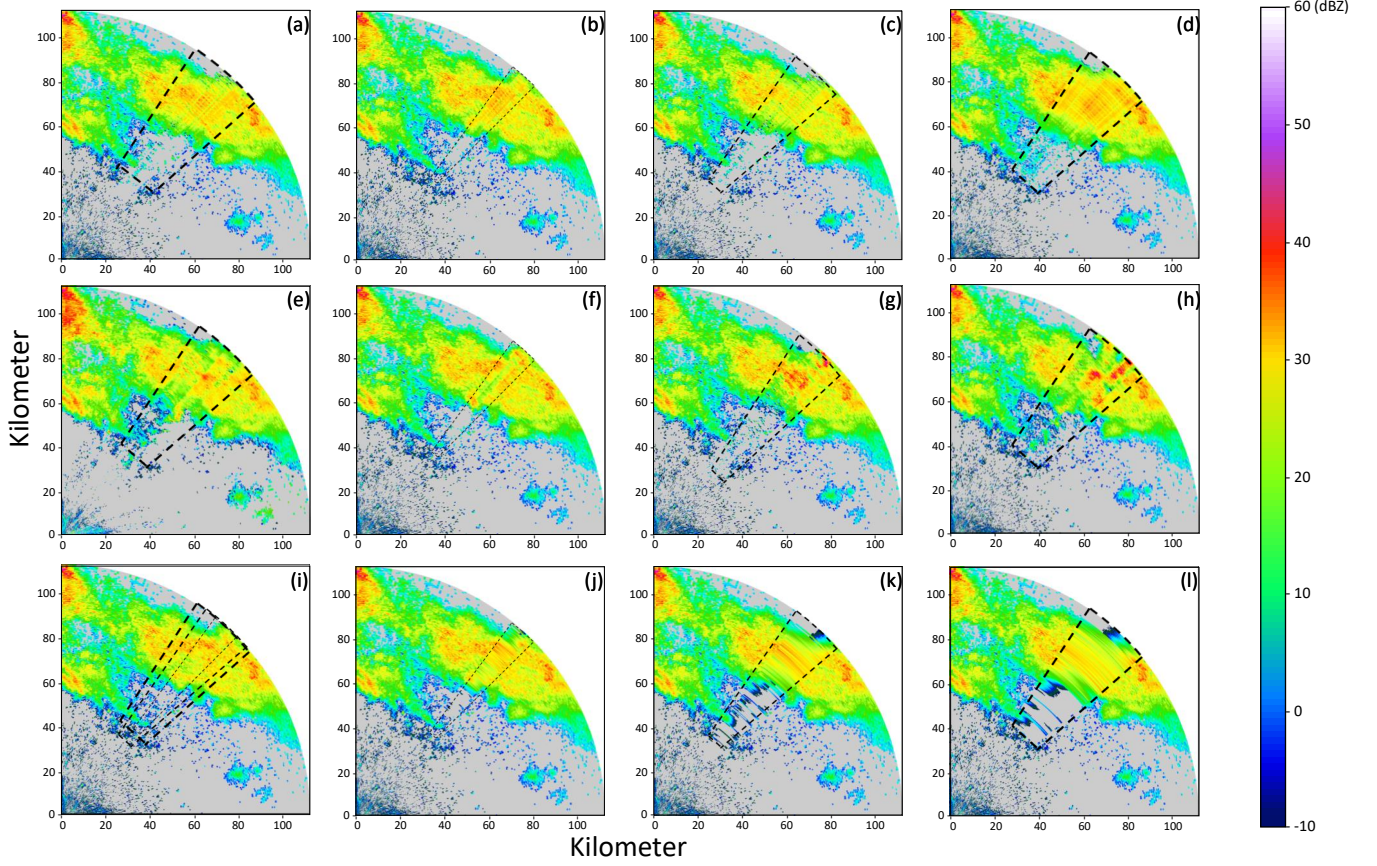


Fig. 9. KDEX test event at 0310UTC, May 07, 2017: (a)-(d) CGAN output with convective (KDEX) training model of random size, 4°, 10° and 16° PBB mask; (e)-(h) CGAN output with stratiform (KDEX) training model of random size, 4°, 10° and 16° PBB mask; (i) ground truth; (j)-(l) linear interpolation output of 4°, 10° and 16° PBB mask.

$$HSS = \frac{2 \times (TP \times TN - FP \times FN)}{(TP + FN) \times (FN + TN) + (TP + FP) \times (FP + TN)} \quad (7d)$$

In this context, the predicted values are generated by the CGAN correction model or linear interpolation, while the actual values are the ground truth data. To determine true positives (TP), both actual and predicted values must be greater than or equal to a given threshold. False negatives (FN) occur when the actual value is greater than or equal to the threshold, but the predicted value is less than the threshold. False positives (FP) occur when the actual value is less than the threshold, but the predicted value is greater than or equal to the threshold. Finally, true negatives (TN) occur when both the actual and predicted values are less than the threshold. To evaluate the performance of the PBB correction model, this study employs four statistical hypothesis tests, namely TP , FN , TN , and FP , along with four commonly used evaluation metrics: the probability of detection (POD), false alarm rate (FAR), critical success index (CSI), and Heidke skill score (HSS). The quality of the model's performance is indicated by higher POD, CSI, and HSS scores and lower FAR scores, all of which are calculated within the range of 0 to 1.

Fig. 8 presents the outcome of the KFWS radar observation with reflectivity data on May 16, 2021, at 20:00 UTC. Overall, the CGAN correction model demonstrates superior accuracy to the linear interpolation approach in both scenarios. Comparing the two CGAN models, it is evident that the TX model provides more realistic and nonlinear correction than the CA model, particularly in regions with high precipitation. However, the CA model is more precise in estimating precipitation intensity in low-precipitation areas. Additionally, the TX model exhibits better cloud edge feature recognition and processing compared to the CA model, which fails to remove precipitation and clear air data accurately.

Regarding correcting the precipitation regional center data, the TX model may overfit, which depends on the magnitude of the ground truth center data. As the beam blockage angle increases in this model, some corrected regional center data becomes inaccurate, although this does not affect the accuracy of cloud shape and precipitation characteristics. The correction difference between the radar center point and the far end appears insignificant as the range of beam blockage becomes larger.

In contrast, as the beam blockage angle increases in the CA model, there is an apparent incoherence of precipitation

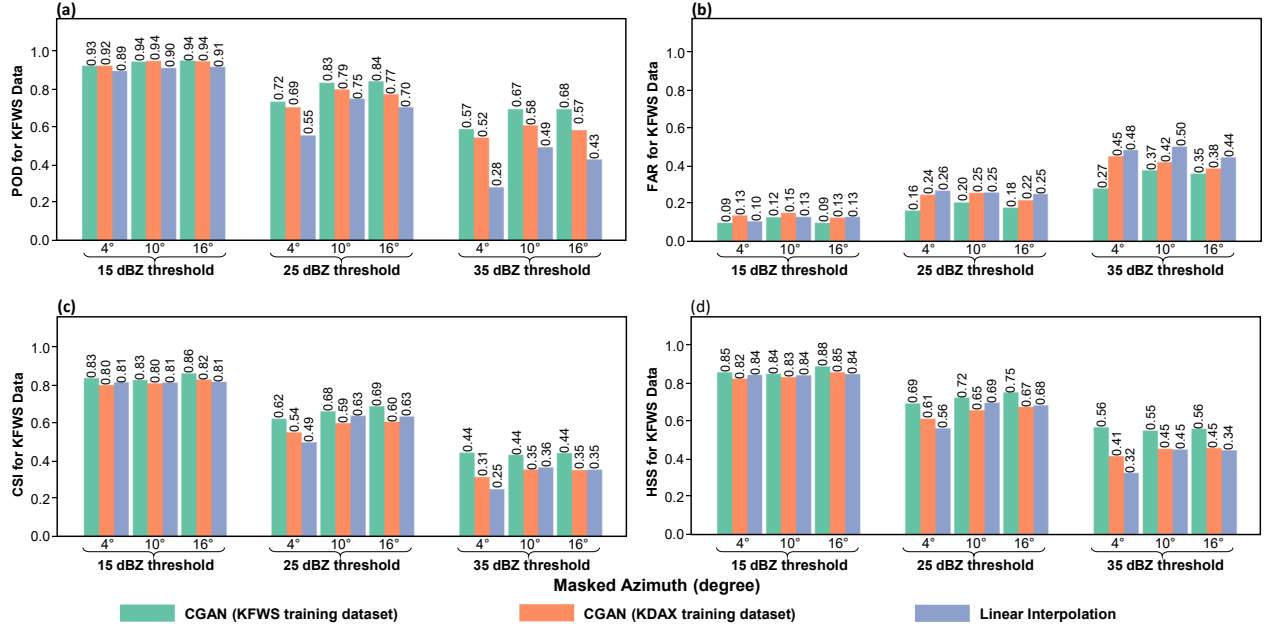


Fig. 10. Evaluation results of the beam blockage correction performance for KFWS test data with different inpainting methods, blockage sizes, and reflectivity thresholds: (a) POD; (b) FAR; (c) CSI; (d) HSS.

information in the correction area, with feature loss in the center of the large angle mask and corrected data similar to stratiform precipitation. The correction difference between the center point and the far end of the radar is noticeable, primarily in cloud boundary information. Furthermore, the unfitting of the precipitation model leads to ripple-like data left by precipitation sampling when a large area of beam blockage appears.

The linear interpolation comparison model reveals that as the beam blockage angle increases, the accuracy of precipitation information reduces, and cloud edge information is lost entirely. Note that this model only guarantees the coherence of precipitation information, and the accuracy of data and features is poor.

Overall, the CGAN correction model outperforms the interpolation model in terms of precipitation information coherence, accuracy, cloud edge accuracy, and feature integrity. Both TX and CA models have pros and cons in terms of data accuracy, with the TX model performing slightly better in feature reconstruction and cloud edge processing for the convective precipitation cases, but both exhibit higher corrected performance than the interpolation model.

Fig. 9 depicts the outcomes obtained on May 7, 2017, at 3:10 UTC, based on the KDAX radar observation with reflectivity data. Upon comparing the two CGAN models, it is evident that the CA model can effectively reconstruct the features of stratified precipitation and reinstate interior and edge information. In contrast, the TX model exhibits slight overfitting in the center, leading to the correction of extraneous detail that is not present in the edge and clear air regions.

In the case of the CA model, simulated precipitation features become more pronounced as the beam blockage angle

increases, albeit not entirely restored. However, the corrected data becomes smoother as the range of beam blockage grows, with no substantial corrected differences between the distance to the radar center point and the far end.

For the TX model, an increase in beam blockage azimuthal angle results in the appearance of convective precipitation features in small magnitudes, which leads to a slight fraction of precipitation above the actual value. Furthermore, the model tends to simulate complex boundary cases at the cloud edge that differ from the ground truth values when there is a large blockage angle.

In contrast to the interpolation model, the CGAN correction model exhibits significantly higher data and cloud edge accuracy when dealing with large-scale stratified precipitation blockage. While the CA model marginally has better performance in terms of feature completeness than the TX model, both models perform well overall in correction and are superior to the interpolation model.

Upon comparing the outcomes of the two experiments, it is evident that the CGAN model demonstrates superior corrective capability over the interpolation model, except for restoring stratified precipitation with low variation values and small blockage angles. In addition, the TX model exhibits higher adaptability and can significantly enhance the accuracy and comprehensiveness of both its own and CA model corrections. However, the CA model is solely effective for rectifying stratiform precipitation.

Figs. 10 and 11 summarize the quantitative assessment results for different beam blockage angles under various correction models. Ten test results were randomly selected from data independent from the training set spanning from 2017 to 2021. The obtained results will undergo evaluation and

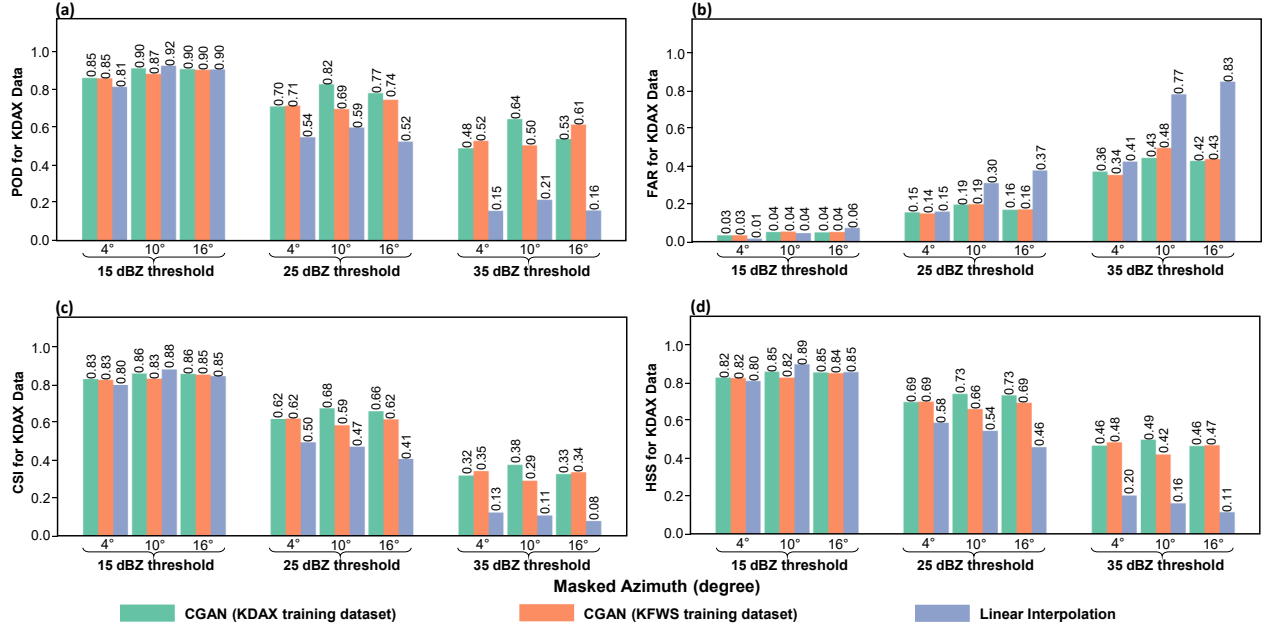


Fig. 11. Evaluation results of the beam blockage correction performance for KDAX test data with different inpainting methods, blockage sizes, and reflectivity thresholds: (a) POD; (b) FAR; (c) CSI; (d) HSS.

subsequent averaging to derive a comprehensive global evaluation. To emphasize differences in correction performance at different rainfall intensities, evaluation scores were computed using three thresholds (15, 25, and 35 dBZ). Fig. 10 reveals that the correction performance decreases as the blockage angle and threshold increase. The performance is observed to decrease progressively in accordance with the same-model test, cross-model test, and interpolation-model gradient. The CA model sometimes exhibits less effectiveness than the interpolation model, which aligns with the conclusion reached in Fig. 8. Furthermore, the correction performance tends to decrease more steeply as the blockage angle increases; i.e., the difference in correction performance between 16-degree and 10-degree angles is much more significant than between 10-degree and 4-degree angles for the same model, which indicates that correction difficulty and performance are not linearly related to blockage size.

Fig. 11 exhibits a similar performance gradient as presented in Figure 10. Notably, for certain small and medium angle corrections, the TX model's performance approaches or surpasses that of the CA model, which aligns with the precipitation intensity inclusiveness highlighted earlier. Additionally, linear interpolation for stratiform precipitation correction leads to entirely erroneous data when evaluated at high thresholds, which explains the gap in correction outcomes between the CGAN model and the interpolation model is more evident in the TX model. It was also found that some of the 25 dBZ threshold performances appeared better than the 15 and 35 dBZ thresholds. This is because the 25 dBZ threshold is an intermediate value between the two extremes of 15 and 35 dBZ. The 15dBZ threshold represents a lower value of reflectivity, and the 35 dBZ threshold represents a higher value

of reflectivity. When using a 15 dBZ threshold, there is a higher likelihood of overestimating the beam blockage and correcting non-blocked regions, leading to false positives. On the other hand, when using a 35dBZ threshold, there is a higher likelihood of underestimating the beam blockage and leaving blocked regions uncorrected, leading to false negatives. The 25 dBZ threshold strikes a balance between these two extremes and allows for more accurate correction of beam blockages.

V. DISCUSSION

A. Impact of Pseudo-Radar Observations on the Beam Blockage Correction Performance

In Section II, it was mentioned that the dataset would incorporate pseudo-radar observations after undergoing cut and filter processes, which can effectively eliminate low-quality observations and decrease the noise and outliers in the training data, thus enhancing the model's robustness and accuracy. However, it is necessary to analyze the differences between the pseudo-radar observations and the original radar data to determine the impact on the model. In convective precipitation, cloud data typically exhibit a significant reflectivity core with irregular echo interior partitions. The screening of pseudo-radar observations primarily focuses on areas with elevated reflectivity, while features of the echo interior partition, which are considerably lower in magnitude than the core peak, are barely selected. This can result in a small range of unsmoothed data in the corrected data. Although the impact can be negligible on the KFWS test data, it renders the KDAX test data less plausible, particularly when significant angle correction is necessary. In the case of the CA model, the precipitation in the original data is in the form of low-intensity and large

areas. Clipping in the image's center can cause the training set to lose some features of the cloud boundaries, leading to a tendency to round off the boundary when handling convective precipitation with complex boundary information, which can limit the model's generalization to new and unseen storm events thus reducing the practical utility. To address this issue, the optimal solution is to increase the amount of training data and add small precipitation events to the training set.

Furthermore, the original plan position indicator (PPI) radar images used in the model are two-dimensional images with centroids. Rotational filtering of the pseudo-radar observations leads to the loss of centroid orientation information. In other words, the distal data of the pseudo-radar observations may point closer to the radar, which can confuse the precipitation characteristics at the far with near radar points during training, making it impossible to accurately evaluate the difference in performance between the two ends of the data in the range direction or continue the experiment to analyze the effect of beam blockage shape on the corrected data. To address this issue, pseudo-radar observations can be modified to employ sectors within concentric circles, ensuring that the range direction always points towards the radar. Alternatively, models with high rotational robustness can be utilized to ensure that the network captures the appropriate features.

B. Potential of Dual Model Systems

In the testing phase, a cross-model testing approach was employed, wherein the convective precipitation model and the stratiform precipitation model were tested for their feasibility in using the correction of the other model. As inferred in Section IV, the convective precipitation training model can accurately simulate beam blockage in stratiform precipitation, encompassing precipitation intensity and cloud edge characteristics, while the stratiform precipitation training model is better equipped for corrections in a limited range of beam blockage. Therefore, the possibility of fusing the two models can be examined. For instance, in northern Texas, convective precipitation is typical in summer, particularly during late afternoons and evenings, while stratiform precipitation is expected in spring and winter. However, simultaneous occurrences of both precipitation types may make using a single model impractical. Therefore, we can merge the dual CGAN models and use the surrounding unblocked data to determine the appropriate precipitation model for the corrected area. In northern California, we can use a dual-model system that combines traditional interpolation methods with CGAN models. Since the edge features of stratiform precipitation clouds are simple, the interpolation method first provides a baseline correction to the radar image, while the CGAN model refines the correction and handles more intricate precipitation patterns.

However, the dual-model system has some potential drawbacks, including increased computational cost and complexity due to the need to train and use multiple models for calibration. Additionally, inconsistent calibration results may arise when using different models, making it difficult to compare and evaluate system performance, thereby necessitating the devel-

opment of new evaluation methods. Despite these concerns, the benefits of a dual-model system outweigh the drawbacks and are essential in the practical application of the model.

C. Impact of Beam Blockage Size on the Correction Model Performance

In Section IV, we discussed a scenario where the linear interpolation model achieved a similar correction performance to the deep learning model in a small-angle blockage test of stratiform precipitation. Since linear interpolation is a straightforward method that estimates missing data points based on the surrounding data, however, this method assumes a linear variation of data between the available data points, which may result in inaccurate estimation when the amount of missing data is large or when the data varies non-linearly. Small areas of stratiform precipitation where the precipitation core is not apparent lead to a linear increase in reflectivity from the cloud edge to the center, corresponding to the linear interpolation assumption. When the radar beam blockage size is significant, the increase in non-linear data results in linear interpolation correction that cannot accurately estimate the missing data. In convective precipitation correction, convective clouds usually comprise several precipitation cores characterized by different peaks, making it challenging to have a uniform linear distribution. Consequently, the interpolation and deep learning test results in the TX model vary considerably.

On the other hand, CGAN can learn to generate realistic data points that match the available data. Nevertheless, the performance of the CGAN model heavily depends on the quantity and quality of the training data. The size and information blocked can impact the quality of the training data. Small radar beam blockages may not affect the quality of the training data, while large radar beam blockages can reduce the quality of the training data, leading to less accurate calibration results. Therefore, the extent and severity of beam blockage and the precipitation distribution within the radar data determine the correction performance. It is crucial to consider these factors when choosing the appropriate correction method for a particular application. The fusion model discussed in the previous subsection can also be beneficial in this regard.

VI. CONCLUSION

The seamless continuation of weather radar imagery is a crucial aspect of data processing and quantitative applications. However, the correction of beam blockages that occur frequently and can result in missing or low-quality data has not been thoroughly investigated. In this study, we have developed and executed a beam blockage correction system based on conditional adversarial neural networks. We trained our dual model using stratiform precipitation data from Northern California (collected by KDAX radar) and convective precipitation observations over the Dallas-Fort Worth area in northern Texas (collected by KFWS radar), and conducted cross-model testing. A conventional linear interpolation approach is also employed during the testing phase for comparison purposes. The major findings of this study are summarized as follows.

- 1) From a visual standpoint, the CGAN-based model for correcting radar beam blockage surpasses conventional interpolation models in terms of its correction efficacy. The correction performance generally declines in the following order: CGAN models trained using similar precipitation events, crossover models trained using distinct precipitation events, and linear interpolation models. In contrast, the TX model is more comprehensive, enabling improved correction of precipitation features and cloud edge prediction in diverse test sets not only in TX but also in CA. Meanwhile, the CA model exhibits suboptimal performance in correcting large-angle beam blockage with notable precipitation differences.
- 2) From the evaluation results, it is concluded that the correction efficacy has a non-linear correlation with the beam blockage size, and the superiority of the CGAN model becomes increasingly prominent as the threshold on precipitation intensity (in terms of radar reflectivity) and blockage size increase.
- 3) The disparity between the performance of the CGAN model and that of the interpolation model is considerably greater during convective precipitation cases than stratiform cases.

In the future, we will incorporate precipitation type analysis and fuse the CGAN model to address the challenge cases when convective and stratiform precipitation coexist. In addition, we can adopt novel pseudo-radar observation filters and measuring metrics to refine the correction efficacy. More importantly, we will focus on real beam blockage or PBB situations. The “simulated” PBB cases are used in this article only for the sake of evaluations. In real situations, it is rather easy to determine the PBB area based on the radar location, radar scan strategy (including radar beam elevation and beamwidth), distance from the radar, and local terrain elevation information. Once the blockage area is identified, we can simply create the PBB mask and perform PBB correction as described in this article. In these cases, it is more challenging to verify the correction performance since no truths are available. A potential solution is to quantify the PBB correction performance through radar quantitative precipitation estimation (QPE). Basically, rainfall estimates from radar data both with PBB and after PBB correction can be derived using standard QPE methodologies (e.g., [1, 14, 25]), and the derived QPE can be compared/evaluated using rain gauge observations in the PBB areas to quantify the improved QPE performance brought by PBB correction.

ACKNOWLEDGEMENT

The authors would like to thank the anonymous reviewers for providing careful reviews and comments on this article. The intermediate features in the deep learning models as well as the model parameters are available upon request.

REFERENCES

- [1] Y. Gou and H. Chen, “Combining radar attenuation and partial beam blockage corrections for improved quantitative application,” *Journal of Hydrometeorology*, vol. 22, no. 1, pp. 139–153, 2021.
- [2] J. Vivekanandan, D. N. Yates, and E. A. Brandes, “The influence of terrain on rainfall estimates from radar reflectivity and specific propagation phase observations,” *Journal of Atmospheric and Oceanic Technology*, vol. 16, no. 7, pp. 837–845, 1999.
- [3] H. Chen and V. Chandrasekar, “The quantitative precipitation estimation system for dallas–fort worth (dfw) urban remote sensing network,” *Journal of Hydrology*, vol. 531, pp. 259–271, 2015.
- [4] V. Chandrasekar, H. Chen, and B. Philips, “Principles of high-resolution radar network for hazard mitigation and disaster management in an urban environment,” *Journal of the Meteorological Society of Japan. Ser. II*, vol. 96A, pp. 119–139, 2018.
- [5] R. Cifelli, V. Chandrasekar, L. Herdman, D. D. Turner, A. B. White, T. I. Alcott, M. Anderson, P. Barnard, S. K. Biswas, M. Boucher, J. Bytheway, H. Chen, H. Cutler, J. M. English, L. Erikson, F. Junyent, D. J. Gottas, J. Jasperse, L. E. Johnson, J. Krebs, J. van de Lindt, J. Kim, M. Leon, Y. Ma, M. Marquis, W. Moninger, G. Pratt, C. Radhakrishnan, M. Shields, J. Spaulding, B. Tehranirad, and R. Webb, “Advanced quantitative precipitation information: Improving monitoring and forecasts of precipitation, streamflow, and coastal flooding in the san francisco bay area,” *Bulletin of the American Meteorological Society*, 2022. [Online]. Available: <https://journals.ametsoc.org/view/journals/bams/aop/BAMS-D-21-0121.1/BAMS-D-21-0121.1.xml>
- [6] I. Crisologo, R. A. Warren, K. Mühlbauer, and M. Heistermann, “Enhancing the consistency of spaceborne and ground-based radar comparisons by using beam blockage fraction as a quality filter,” *Atmospheric Measurement Techniques*, vol. 11, no. 9, pp. 5223–5236, 2018.
- [7] X. Li and M. T. Orchard, “New edge-directed interpolation,” *IEEE transactions on image processing*, vol. 10, no. 10, pp. 1521–1527, 2001.
- [8] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, “Image inpainting,” in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, 2000, pp. 417–424.
- [9] S. Roth and M. J. Black, “Fields of experts: A framework for learning image priors,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, vol. 2. IEEE, 2005, pp. 860–867.
- [10] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, “Generative image inpainting with contextual attention,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5505–5514.
- [11] M. Elad and M. Aharon, “Image denoising via sparse and redundant representations over learned dictionaries,” *IEEE Transactions on Image processing*, vol. 15, no. 12, pp. 3736–3745, 2006.
- [12] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial networks,” *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.
- [13] E. L. Denton, S. Chintala, R. Fergus *et al.*, “Deep

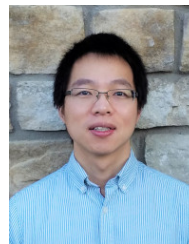
- generative image models using a laplacian pyramid of adversarial networks,” *Advances in neural information processing systems*, vol. 28, 2015.
- [14] H. Chen, R. Cifelli, and A. White, “Improving operational radar rainfall estimates using profiler observations over complex terrain in northern california,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 3, pp. 1821–1832, 2019.
- [15] H. Chen, V. Chandrasekar, and R. Bechini, “An improved dual-polarization radar rainfall algorithm (drops2.0): Application in nasa ifloods field campaign,” *Journal of Hydrometeorology*, vol. 18, no. 4, pp. 917 – 937, 2017.
- [16] L. Han, J. Zhang, H. Chen, W. Zhang, and S. Yao, “Toward the predictability of a radar-based nowcasting system for different precipitation systems,” *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2022.
- [17] Y. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, “Efficient backprop,” in *Neural networks: Tricks of the trade*. Springer, 2002, pp. 9–50.
- [18] C.-C. J. Kuo, “Understanding convolutional neural networks with a mathematical model,” *Journal of Visual Communication and Image Representation*, vol. 41, pp. 406–413, 2016.
- [19] M. Mirza and S. Osindero, “Conditional generative adversarial nets,” *arXiv preprint arXiv:1411.1784*, 2014.
- [20] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III* 18. Springer, 2015, pp. 234–241.
- [21] F. Milletari, N. Navab, and S.-A. Ahmadi, “V-net: Fully convolutional neural networks for volumetric medical image segmentation,” in *2016 fourth international conference on 3D vision (3DV)*. Ieee, 2016, pp. 565–571.
- [22] O. Oktay, J. Schlemper, L. L. Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. McDonagh, N. Y. Hammerla, B. Kainz *et al.*, “Attention u-net: Learning where to look for the pancreas,” *arXiv preprint arXiv:1804.03999*, 2018.
- [23] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh, and J. Liang, “Unet++: A nested u-net architecture for medical image segmentation,” in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings 4*. Springer, 2018, pp. 3–11.
- [24] S. Yao, H. Chen, E. J. Thompson, and R. Cifelli, “An improved deep learning model for high-impact weather nowcasting,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 7400–7413, 2022.
- [25] Y. Gou, H. Chen, H. Zhu, and L. Xue, “Microphysical processes of super typhoon lekima (2019) and their impacts on polarimetric radar remote sensing of

precipitation,” *Atmospheric Chemistry and Physics*, vol. 23, no. 4, pp. 2439–2463, 2023. [Online]. Available: <https://acp.copernicus.org/articles/23/2439/2023/>



environmental monitoring.

Songjian Tan (Student Member, IEEE) received his B.S. degree in Optics from Delaware State University, Dover, DE, USA, in 2020. He is currently working toward a M.S. degree in Computer Engineering at Colorado State University, Fort Collins, CO, USA. His research interests include deep learning and artificial intelligence (AI) for radar remote sensing of precipitation. His work focuses on developing AI models that can improve the accuracy and efficiency of radar data processing in support of weather forecasting, disaster management, and



Sciences Laboratory, Boulder, CO, USA, from 2012 to 2020, first as a Research Student, then a National Research Council Research Associate and a Radar, Satellite, and Precipitation Research Scientist. His research interests span a broad range of remote sensing and multidisciplinary data science, including radar and satellite remote sensing of natural disasters, polarimetric radar systems and networking, clouds and precipitation observations and processes, big data analytics, and deep learning.

Dr. Chen was a recipient of the National Science Foundation (NSF) Faculty Early Career Development Program (CAREER) Award, Ralph E. Powe Junior Faculty Enhancement Award, IEEE Geoscience and Remote Sensing Society (GRSS) Early Career Award, American Meteorological Society (AMS) Editor's Award, and CIRA Research and Service Initiative Award. He serves as an Associate Editor for the *Journal of Atmospheric and Oceanic Technology*, *URSI Radio Science Bulletin*, and the *IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATION AND REMOTE SENSING*.