

Radar Super Resolution Using a Deep Convolutional Neural Network[✉]

ANDREW GEISS^{a,b} AND JOSEPH C. HARDIN^a

^a *Atmospheric Sciences and Global Change Division, Pacific Northwest National Laboratory, Richland, Washington*

^b *Department of Atmospheric Sciences, University of Washington, Seattle, Washington*

(Manuscript received 19 May 2020, in final form 19 August 2020)

ABSTRACT: Super resolution involves synthetically increasing the resolution of gridded data beyond their native resolution. Typically, this is done using interpolation schemes, which estimate sub-grid-scale values from neighboring data, and perform the same operation everywhere regardless of the large-scale context, or by requiring a network of radars with overlapping fields of view. Recently, significant progress has been made in single-image super resolution using convolutional neural networks. Conceptually, a neural network may be able to learn relations between large-scale precipitation features and the associated sub-pixel-scale variability and outperform interpolation schemes. Here, we use a deep convolutional neural network to artificially enhance the resolution of NEXRAD PPI scans. The model is trained on 6 months of reflectivity observations from the Langley Hill, Washington, radar (KLGX), and we find that it substantially outperforms common interpolation schemes for 4× and 8× resolution increases based on several objective error and perceptual quality metrics.

KEYWORDS: Atmosphere; Precipitation; Data processing; Radars/Radar observations; Neural networks; Deep learning

1. Introduction

Image super resolution is a well-established area of research in the field of computer vision and image processing. It involves artificially increasing the resolution of an image beyond the resolution of the sensor used to capture the image. (Nasrollahi and Moeslund 2014) provide a detailed review of classical image super-resolution techniques. Here we focus on single-image super resolution (SISR), where sub-pixel-scale features are inferred based only on the information contained in a single original image, though several techniques exist that take advantage of video (Baker and Kanade 1999; Huang et al. 2018), multiple viewing angles (Tao and Muller 2018; Richard et al. 2020), or in the case of radars, multiple overlapping radars (Bharadwaj 2009). SISR is an ill-posed problem: there are multiple high-resolution scenes that, when downsampled, can produce the same low-resolution image. The simplest of SISR techniques are interpolation schemes, which estimate sub-pixel-scale information from only neighboring pixels in the original image and perform the same operation everywhere in the image regardless of large-scale context. More advanced SISR schemes often operate on image patches, and use additional low-resolution texture information from the area surrounding a pixel along with a dictionary of exemplars to infer likely sub-pixel-scale features (Timofte et al. 2015). These schemes can outperform interpolation in terms of both objective metrics like mean-squared error computed on pixel intensity and in terms of the aesthetics of the result. Recently, deep convolutional neural networks (CNNs) have become a

popular tool for SISR (Wang et al. 2019). Here we apply a state-of-the-art CNN-based SISR scheme to NEXRAD plan position indicator (PPI) scans. Though precipitation is most immediately determined by microphysical processes, the features that appear in PPI scans are ultimately constrained by the synoptic meteorology (cyclone-scale weather features), and similar precipitating features occur across many different PPI scans depending on the regional weather, for instance, the presence of a cold front and corresponding heavy precipitation in an extratropical cyclone. By learning common sub-pixel-scale features in the context of large-scale weather in PPI scans, a neural network can outperform interpolation schemes.

Though introduced in the late 1980s, deep CNNs have become very popular since about 2010 for various image processing tasks, and have consistently and dramatically outperformed existing algorithms for tasks such as image labeling and segmentation (Ronneberger et al. 2015; He et al. 2015; Haug et al. 2017). CNNs are particularly useful for two-dimensional gridded inputs. Instead of learning unique weights for each location in the image they learn weights for many small convolutional kernels (typically 3×3 to 10×10 pixels) that are applied everywhere in the image. This significantly reduces the number of weights in the neural network and makes training large networks feasible. The trend in current research is toward increasingly large and complex CNNs. Recently, several researchers have been successful applying various types of CNNs for SISR (Dong et al. 2014; Ledig et al. 2017; Kim et al. 2016; Lai et al. 2017; Lim et al. 2017; Johnson et al. 2016), and here we extend some of these techniques for use with precipitation radar data.

This is not the first work to look at machine learning-based super resolution in the context of radars. (Gao et al. 2017) showed how a complex valued CNN could be used to improve the resolving capability of mm-wave (~ 220 GHz) imaging radars. This was not a true SISR technique, but rather an

[✉] Supplemental information related to this paper is available at the Journals Online website: <https://doi.org/10.1175/JTECH-D-20-0074.s1>.

Corresponding author: Andrew Geiss, avgeiss@gmail.com

improvement in the image formation process converting from frequency space to an image. (Armanious et al. 2019) used generative adversarial networks to perform SISR on micro-Doppler imagery. Veillette et al. (2018) focused on weather radar, but instead of super resolution focused on generating weather radar data from “radar-like” sources such as lightning flashes and cloud-top height from satellites. As such, we believe ours is the first work to focus on super resolution of weather radar imagery from single images.

There are many applications for weather radar SISR techniques spanning radar operations to research. Operationally, radar SISR allows faster coarser scans to be taken and subsequently increased in resolution, allowing for more efficient scanning. As for research applications, the ability to increase the resolution of the data to a new grid while maintaining as much fidelity and frequency information as possible makes it easier to compare radar observations with both high-resolution models and other sources of instrumentation. For instance, by increasing resolution, precipitation maps could be better incorporated into hydrologic modeling of rainfall in orographic areas. This technique also has the potential to be extended to satellite weather radars to mitigate the nonuniform beam filling problem (Ohsaki and Nakamura 1998). Finally, increasing the resolution in a physically plausible way allows for better 3D visualization of weather radar data without the need for overly smooth isosurfaces.

2. Data

The data used here are NEXRAD composite reflectivity PPI scans from the Langley Hill, Washington, radar (KLGX). The observations used were taken in October, November, and December of 2016, 2017, and 2018. The Langley Hill radar is an S-band Doppler weather radar run by the U.S. National Weather Service. It sits on the coast of Washington State and provides radar coverage of midlatitude cyclones approaching from the Pacific Ocean that are obscured from inland radars by the Olympic mountain range. Because of the annual cycle of the storm tracks and the coastal mountain range this is one of the rainiest extratropical regions in the world in autumn, and most PPI scans at this time of year show precipitation.

We use cases when the radar was operating in volume coverage pattern (VCP) 12, 212, or 215 mode which account for about 63% of the observations. The radar operates in these modes when there is precipitation nearby, so most of the samples used include some precipitating features, though in some only a small fraction of the scan includes precipitation and there are some clear-sky scans typically taken immediately before or after precipitation is present. A PPI scan is a 360° radial (range versus azimuth) scan around the radar. The NEXRAD VCP modes involve taking successive PPI scans at several increasing antenna elevation angles to retrieve three-dimensional observations in a volume around the radar. We convert these volumes to a composite reflectivity by taking the maximum reflectivity with respect to elevation angle for the first six sweeps of each volume scan, resulting in a range by azimuth dataset representing the maximum reflectivity for elevation angles at or below 3.1°. That is, each pixel in the 2D

scans used here represents the maximum reflectivity observed above that point in the corresponding 3D volume scan, excluding the higher scan angles. Composite reflectivity PPI scans are the radar product typically used in TV weather broadcasts and are likely the most familiar weather radar product for the general public.

In the scanning modes used, the radar’s range resolution is 250 m and its azimuthal resolution is 0.5°. Convolutional neural networks are typically built to operate on uniformly gridded data, so the composite reflectivity data are then interpolated on to a 512×512 cartesian grid with a grid spacing of 1.56 km using a nearest neighbor scheme. This reduces the effect of the range dependence of the radar’s spatial sampling: increasing pixel sizes further from the radar. A downside of this approach is that the regridded high-resolution (512×512) scans used here have a lower resolution than the range resolution of the radar. Furthermore, while the cartesian grid is uniform, the weather features in the scans are still subject to the radar’s resolution reduction with respect to range: more distant weather features are not as sharply resolved as precipitation close to the radar. This is an important step for the CNN-based approach, however, because it ensures that the scale (2D area) of weather features does not change depending their position in the scan. Given the success of CNN-based super resolution on Cartesian gridded radar data, developing CNN-based techniques that can be applied to polar gridded data in the radar’s native coordinates is a valuable area for future research.

3. Method

a. Neural network architecture

The neural network consists of two main components. The first is a CNN architecture commonly referred to as a “U-net.” U-nets were first used for segmentation of medical scans (Ronneberger et al. 2015). This network architecture is particularly useful for cases when the output of the network has similar dimensions to the input, and pixels in the output rely on feature information spanning a large range of spatial scales in the input. The U-net downsamples an input image by passing it through several blocks of convolutional layers, and then upsamples it, again through several convolutional layers, to the resolution of the input but with a larger number of channels. Each downsampling/upsampling block in the neural network used is composed of three “densely connected” layers (Haug et al. 2017). Each of the layers in the dense blocks performs batch normalization (Ioffe and Szegedy 2015), followed by a 3×3 convolution, and a rectified linear unit (ReLU) activation function [defined by $f(x) = \max(x, 0)$]. A key feature of this network architecture is that it contains “skip” connections (He et al. 2015) between downsampling and upsampling blocks with the same dimensions and layers within the dense blocks. This means that the shortest path between input and output in this neural network only passes through 1 convolutional layer while the longest passes through 37 (all of the convolutional layers: 9×4 -layer dense blocks and one output layer; see Fig. 1), and the neural network can learn to combine information passing through various numbers of convolutional

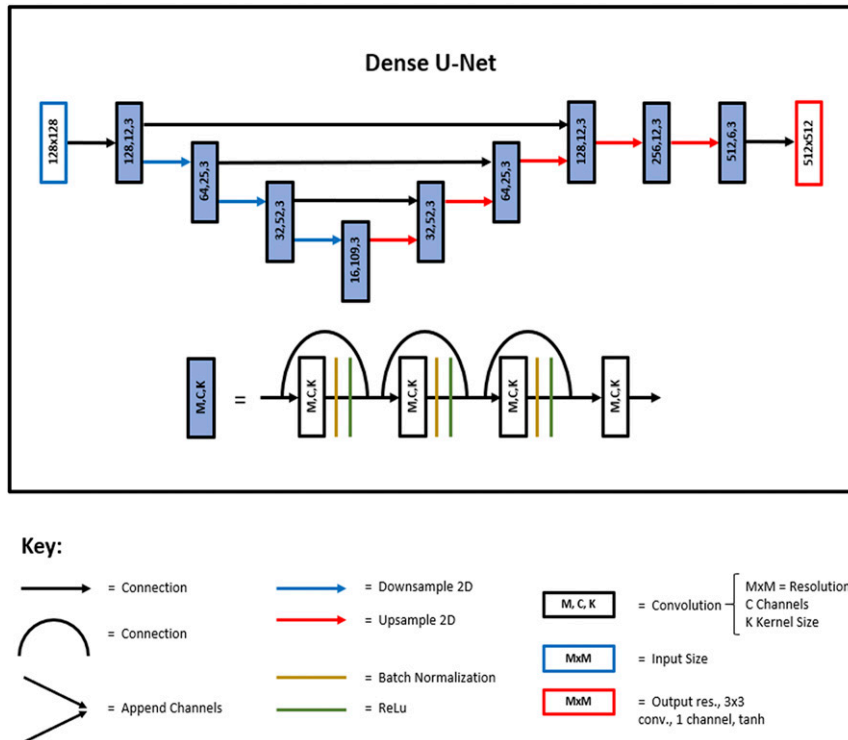


FIG. 1. Diagram of neural network architecture. The left portion of the CNN is a U-net meant to infer meaning from the precipitation features in the low-resolution PPI scan, while the right portion is an upsampling module. Each blue block consists of three densely connected layers where each layer includes a 2D convolution, batch normalization, and rectified linear unit transfer function. The values printed in each blue box represent the output resolution, number of channels, and size of the convolutional kernel.

layers to generate the output. The second half of the neural network is an upsampling module consisting of either two or three upsampling blocks depending on the desired increase in resolution. The motivation for this architecture is that the U-net module can identify useful large-scale features (for instance the presence of a cold front covering a significant portion of the scan) in the low-resolution scan and the upsampling module along with the upsampling blocks in the U-net can map this information to sub-pixel-scale variability. A diagram of the neural network architecture is shown in Fig. 1.

While U-net style neural networks have gained popularity recently, there are several other network architectures that are well suited for this task. We trained several other varieties of CNNs using full PPI scans as described above. We used 90 training epochs in each case and ensured that each neural network had around 10^6 trainable parameters ($\pm 20\%$). Figure 2 shows The mean-squared pixel-wise error (MSE) computed on the validation set for 6 different CNN types: a “residual” network (He et al. 2015), a “dense network” (Haung et al. 2017), a classical U-net (Ronneberger et al. 2015), a U-net with dense blocks (used here), a network similar to SRCNN as proposed by Dong et al. (2014), and a network composed of only convolutional layers inspired by Long et al. (2014). The implementations of each of these CNNs diverge slightly from those in the cited texts; see supplement section 1 for a detailed

description of each. The dense U-net outperforms all of the other architectures, though not by much, so we use it in this study.

b. Training procedure

Several different densely connected U-nets were ultimately trained on the gridded composite reflectivity data for two different SISR tasks. The first involves increasing the resolution of an entire PPI scan. In this case 512×512 data are artificially degraded to a resolution of either 64×64 or 128×128 by taking either 8×8 or 4×4 pixel averages, respectively. During training, the degraded scan is provided as input to the neural network and the original 512×512 scan is used as a target. Because the radar observations look equally realistic regardless of their orientation in the azimuthal plane, we apply data augmentation during training by randomly rotating each scan by 0° , 90° , 180° , or 270° and randomly reflecting in the horizontal and vertical axes, which artificially increases the number of unique observations used during training.

In the second case we train a scale/shift/rotation invariant model. During each training epoch a random region is selected from each scan using a randomly placed window between 192×192 and 512×512 pixels. The chunk of data is then downsampled to a resolution of 192×192 , which is used as a target, and a resolution of 48×48 , which is used as an input to

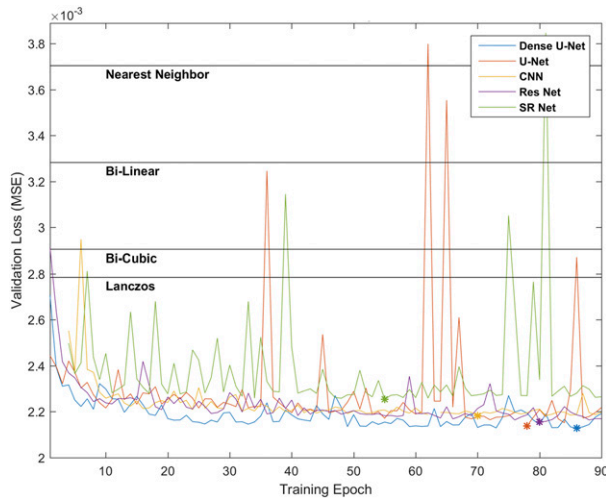


FIG. 2. A comparison of the mean-squared pixel error super-resolution performance of various neural network architectures on the validation set during training. “Dense U-net” (blue line), a U-net composed of several densely connected blocks, is the architecture used. The asterisks denote the epoch at which each model achieved its lowest validation score. Descriptions of the models can be found in Fig. 1 and supplement material section 1.

the neural network. In this case we also use rotation and flipping data augmentation.

Both approaches to SISR are operationally relevant for radar data. Doppler radars trade off range resolution for velocity resolution (Bringi and Chandrasekar 2001). A neural network

trained to enhance the resolution of a full PPI scan can be trained on data collected by a radar operating with high range resolution and low velocity resolution and then applied when the radar is operating with low range resolution and high velocity resolution, to mitigate the effects of this trade-off (Armanious et al. 2019). The second approach, which operates on partial PPI scans and does not rely on a consistent input resolution, can be used to enhance the resolution of a scan beyond the maximum resolution of the radar, though there is no way to validate the result.

The model was designed and trained using Keras with a Tensorflow backend. The CNNs were trained for 250 epochs using an ADAM optimizer (Kingma and Ba 2014) with a learning rate of 0.001 and a batch size of 5. Then the learning rate was reduced by a factor of 10 and the model was trained for an additional 50 epochs from the model state that achieved the lowest validation score during the first 250 epochs. The MSE is used as a loss function and MSE with respect to training epoch is shown in Fig. 3:

$$\text{MSE} = \frac{1}{XY} \sum_{x \in X} \sum_{y \in Y} (t_{x,y} - \hat{t}_{x,y})^2; \quad (1)$$

here, x and y iterate over space, $t_{x,y}$ represents the ground truth reflectivity at a coordinate, and $\hat{t}_{x,y}$ represents the neural network’s predicted reflectivity at that coordinate.

From the fall of 2016 and 2017, 25 478 PPI scans were used for training while 11 381 scans from fall of 2018 were used as a test set. We reserved 25% of the 2016 and 2017 training scans as a validation set. VCP 12 takes the radar about 5 min to complete, so successive scans frequently contain the same

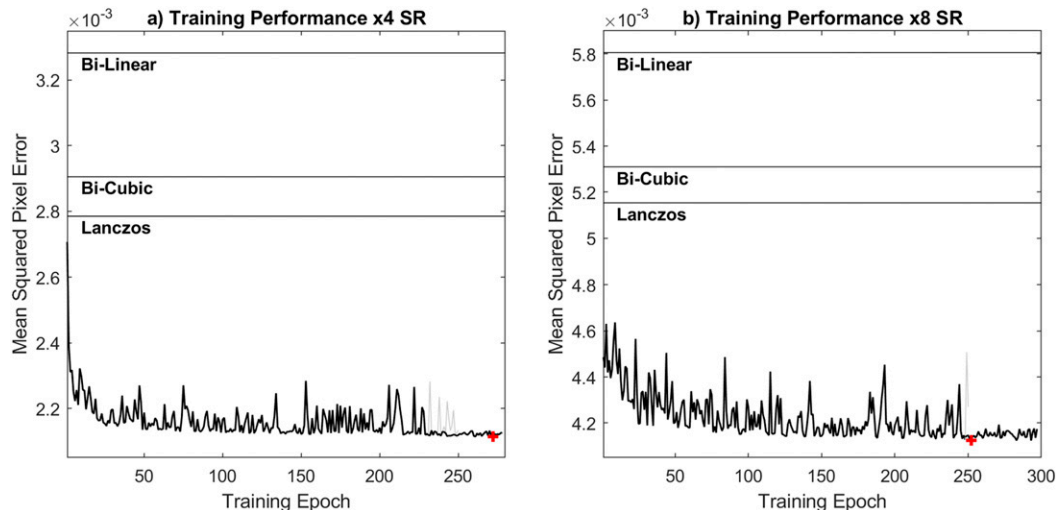


FIG. 3. Mean squared pixel error performance of the CNN during a final training run for 300 epochs. (a) Performance on the validation set for $4\times$ super resolution (128×128 to 512×512), and (b) for $8\times$ super resolution (64×64 to 512×512). In each case, the CNN was trained for 250 epochs, then the model state that achieved the lowest validation score in these first 250 epochs was trained for an additional 50 epochs with a lower learning rate. The gray portion of the curve represents validation loss from the high-learning-rate portion of the initial training that occurred after the model’s best validation score was achieved, the corresponding model states were not used. The lowest overall validation score achieved is marked with a red plus sign. The horizontal lines represent the performance of several common interpolation schemes used as benchmarks.

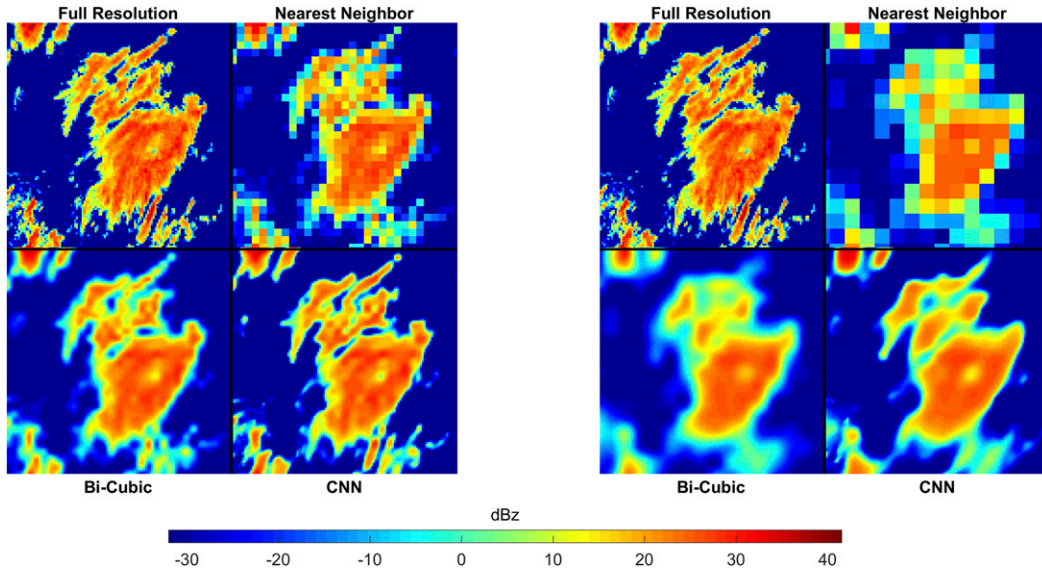


FIG. 4. Examples of (left) 4× and (right) 8× super resolution applied to a KLGX composite reflectivity PPI scan from 23 Dec 2018. (top) The original scan (target) and the degraded scan (input). (bottom) The neural network output to bicubic interpolation. The neural network is much better at preserving sharp edges, isolated small-scale features (middle left of scan), and complicated small-scale variability (upper center and lower right of scan).

precipitation features that have been advected downstream (and modified by the small-scale flow and cloud/rain physics). The frequency and type of precipitation that occurs in this region also undergoes a seasonal cycle, with fewer days with precipitation at the beginning and end of the study period. The validation set is generated by selecting 12 temporally contiguous chunks of data with random start times to prevent the seasonal cycle from influencing the results. The 24 scans (~2 h) from before and after each time-chunk used for validation are excluded to prevent any data leakage between the training and validation sets due to persistence of precipitation features.

c. Validation metrics

We compare the output from the CNN to several common interpolation schemes: nearest neighbor, linear, bicubic, and Lanczos. Nearest neighbor is the simplest and involves simply repeating pixel data, and results in a pixelated looking upsampled image. Linear and bicubic involve fitting the space between known pixels with a linear or cubic (respectively) function, and sampling sub-pixel-scale data from that function. Finally, Lanczos resampling uses a sinc function as a kernel and is useful for preserving sharp edges in upsampled images. Lanczos and bicubic interpolation are frequently used for image upsampling because they tend to produce visually appealing results.

Several different metrics are used to compare the neural network output to these four interpolation schemes. The first is simply the MSE computed between the upsampled scan and the ground truth. Structural similarity index (SSIM) (Wang et al. 2004), is a dimensionless metric between -1 and 1 that is well suited for evaluating the perceptual difference between image pairs. The SSIM was developed to approximate the quality of an image as perceived by the human visual system.

Signal-to-noise ratio (SNR) compares the MSE to the magnitude of the pixel data in the output scan. Here we define SNR as

$$SNR = \frac{\|\hat{I}\|_F^2}{\|I - \hat{I}\|_F^2}, \tag{2}$$

where $\|\cdot\|_F$ is the Frobenius norm, I is the target PPI scan, and \hat{I} is the CNN output. SNR is used here instead of the commonly used peak (P)SNR because a significant portion of many of the scans contain no precipitation, and (2) does not heavily reward reproducing the empty portions of PPI scans while PSNR does. We also report power spectral density (PSD) curves for the CNN output and the various interpolation schemes. The PSD curves show how well each of the techniques preserves small-scale features and can also be used to identify aliasing. Here we define PSD as

$$PSD = 10 \log_{10} \left(\overline{|F_2\{\hat{I}\}|^2} \right), \tag{3}$$

where the overbar represents radial averaging in Fourier space, and $F_2\{\dots\}$ is the 2D Fourier transform.

During training of the neural network, the reflectivity data are normalized to a -1 to 1 scale, and the training loss (MSE) is computed as an average over the whole scan. In the figures that show performance during training (Figs. 2 and 3) we report this dimensionless metric. While this is useful for measuring relative performance between different super-resolution schemes, dimensional values give a more intuitive idea of the practical performance of the neural network. To this end, we also report dimensional mean absolute error (MAE) in dBZ. Computing this value over all pixels in the scans means that the MAE is skewed significantly lower for

TABLE 1. Comparison of mean squared pixel error (MSE), signal-to-noise ratio [SNR; Eq. (2)], structural similarity index (SSIM), and mean absolute pixel error (MAE) computed on the test set for several interpolation schemes and the CNN output. MAE (dBZ) is computed only over regions where the ground truth data shows precipitation.

	Nearest	Bilinear	Bicubic	Lanczos	CNN
4× super resolution					
MSE $\times 10^3$	3.28	2.90	2.57	2.57	1.93
SNR	15.3	16.9	19.3	19.3	25.5
SSIM	0.92	0.92	0.93	0.93	0.95
MAE (dBZ)	6.67	6.68	6.21	6.21	4.99
8× super resolution					
MSE $\times 10^3$	5.66	5.10	4.66	4.66	3.76
SNR	8.6	9.4	10.5	10.6	13.7
SSIM	0.88	0.88	0.89	0.89	0.92
MAE (dBZ)	8.82	8.80	8.34	8.33	7.14
Scale/shift invariant super resolution					
MSE $\times 10^3$	5.33	4.53	3.86	3.64	2.69
MAE (dBZ)	5.26	5.21	4.71	4.56	3.64

scans without many weather features, and does not give a good idea of what the actual difference in dBZ is between schemes for a pixel containing weather, so the MAE is calculated by averaging only over pixels where the ground truth data contain reflectivities greater than a threshold of -32 dBZ and neighboring pixels. This metric is meant to indicate the differences in reflectivity one can expect between schemes when there is weather present.

4. Results

Figure 3 shows the validation set MSE for both the 4×SR and 8×SR cases with respect to training epoch. In both cases the validation MSE remained relatively stable throughout training. The neural networks both begin to outperform the benchmark interpolation schemes (shown as horizontal lines) after only a single epoch but continue to incrementally improve throughout the remainder of the training. As described in section 3b, after reducing the learning rate at 250 epochs the previous state of the model that produced the lowest validation loss is trained for 50 additional epochs at a lower learning rate. The training epochs at the higher learning rate after the model achieved its lowest validation MSE are still shown in light gray. A plus sign indicates the training epoch that produced the best validation set MSE overall.

Sample CNN output for a portion of a PPI scan that contains some small-scale precipitation features is shown in Fig. 4. Figure 4a shows, top to bottom left to right: the original PPI scan, the degraded PPI scan, the result of applying bicubic interpolation to the degraded scan, and the result of applying the CNN-based SR to the degraded scan. The CNN output is subjectively superior to the interpolation scheme. Neither approach is able to recover all of the very finescale precipitation structure that is lost when the original scan is degraded; however, the neural network preserves more of the finescale structure and is notably better at preserving sharp edges associated with the larger features. Figure 4b is the same as Fig. 4a, but in this case for 8×SR. Again, just examining the neural network output is enough to see that it is able to recover more of the small-scale features than interpolation. Notably, in

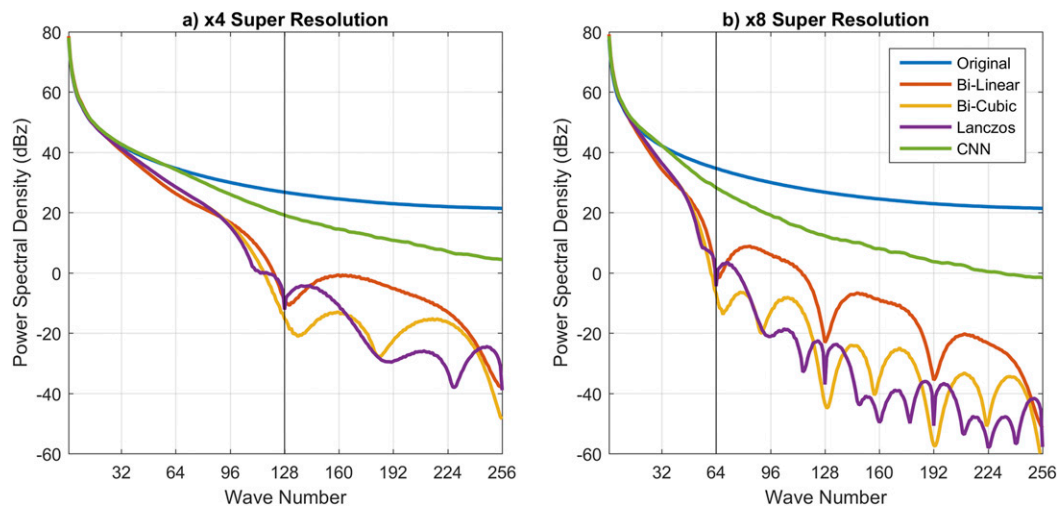


FIG. 5. Radially averaged power spectral density (PSD) of CNN output [Eq. (3)], interpolation schemes, and full-resolution scans. The full-resolution scans are 512×512 pixels and in each panel the vertical black line indicates the resolution of the artificially degraded scan used as input to the neural network. The higher PSD of the CNN output (green line) at higher wavenumbers indicates that the neural network is much better at preserving small-scale variability than common interpolation schemes (purple, yellow, and red lines). It also notably avoids the artifacts that the interpolation schemes introduce at higher wavenumbers, and preserves all of the power at lower wavenumbers up to the Nyquist frequency of the degraded scans (half the distance from the y axis to the black line), which the interpolation schemes do not.

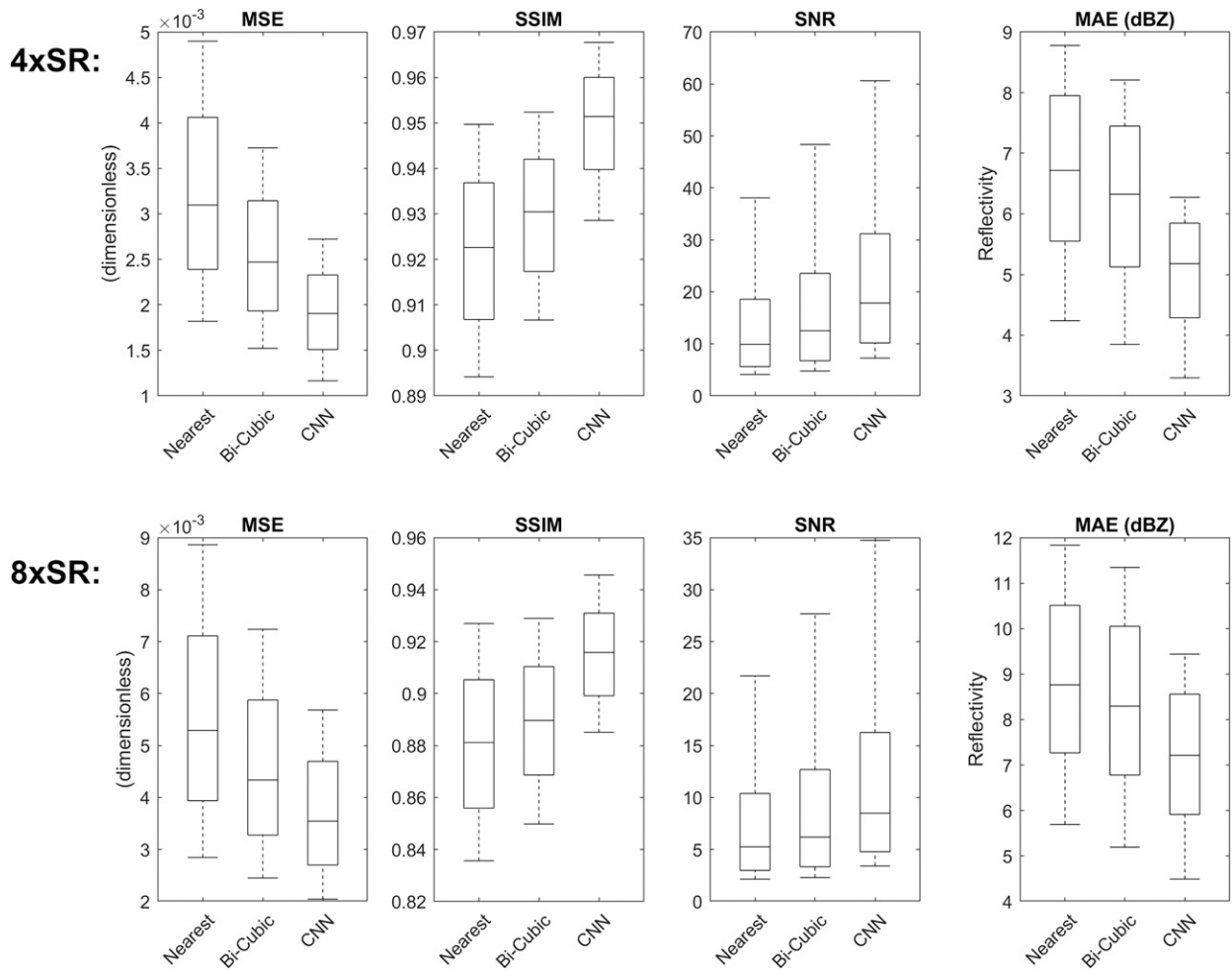


FIG. 6. Comparison of (left to right) mean squared pixel error (MSE), structural similarity index (SSIM), signal-to-noise ratio (SNR), and mean absolute error (MAE) in dBZ computed on the test set for degraded scans (nearest-neighbor interpolation), bicubic interpolation, and the CNN output. For MSE and MAE lower scores are better and for SSIM and SNR higher scores are better. The boxplots indicate the 10th, 25th, 50th, 75th, and 90th percentiles for the 11 381 PPI scans in the test set. The neural network outperforms interpolation by every metric.

both cases, the small parallel linear precipitation features in the upper center and lower right portion of the scan are much better approximated by the neural network. In fact, in the 8×SR case the texture of these features appears to be completely lost when the scan is degraded and the bicubic interpolation scheme combines them into a single blob of precipitation, but the neural network splits the precipitation here into two parallel features, a much better approximation of the input. We speculate the CNN has learned to infer the likely presence of this type of parallel linear feature based on the orientation of the larger nearby precipitation features. The data augmentation process means that the CNN should not have learned to preferentially produce this type of feature with a specific orientation in the case that they are caused by interaction with orography. The CNN also produces subjectively superior results in the region at the top of the sample, where again there are a number of small-scale linear features. It is also much better at localizing small

isolated points of precipitation that tend to be smeared out over a large area by interpolation. There is a good example of this at the left edge of the sample shown in Fig. 4. We provide additional samples from individual scans in section 2 of the supplemental material.

The neural network performance on the test set for the 4×SR, 8×SR, and partial scan cases are listed in Table 1. In all cases the neural networks substantially outperform several common interpolation schemes, and yield error near the value of their final validation error during training, shown in Fig. 3, indicating that the model has not overfit. Other metrics are more useful for evaluating the aesthetic quality of the results in an objective way. Figure 5 shows PSD [Eq. (3)] plots for the full-resolution scan, the CNN, and bilinear, bicubic, and Lanczos interpolation schemes. The neural network preserves substantially more power at high wavenumbers, corresponding to finescale features, than any of the interpolation schemes do. The vertical black line in each panel represents the resolution

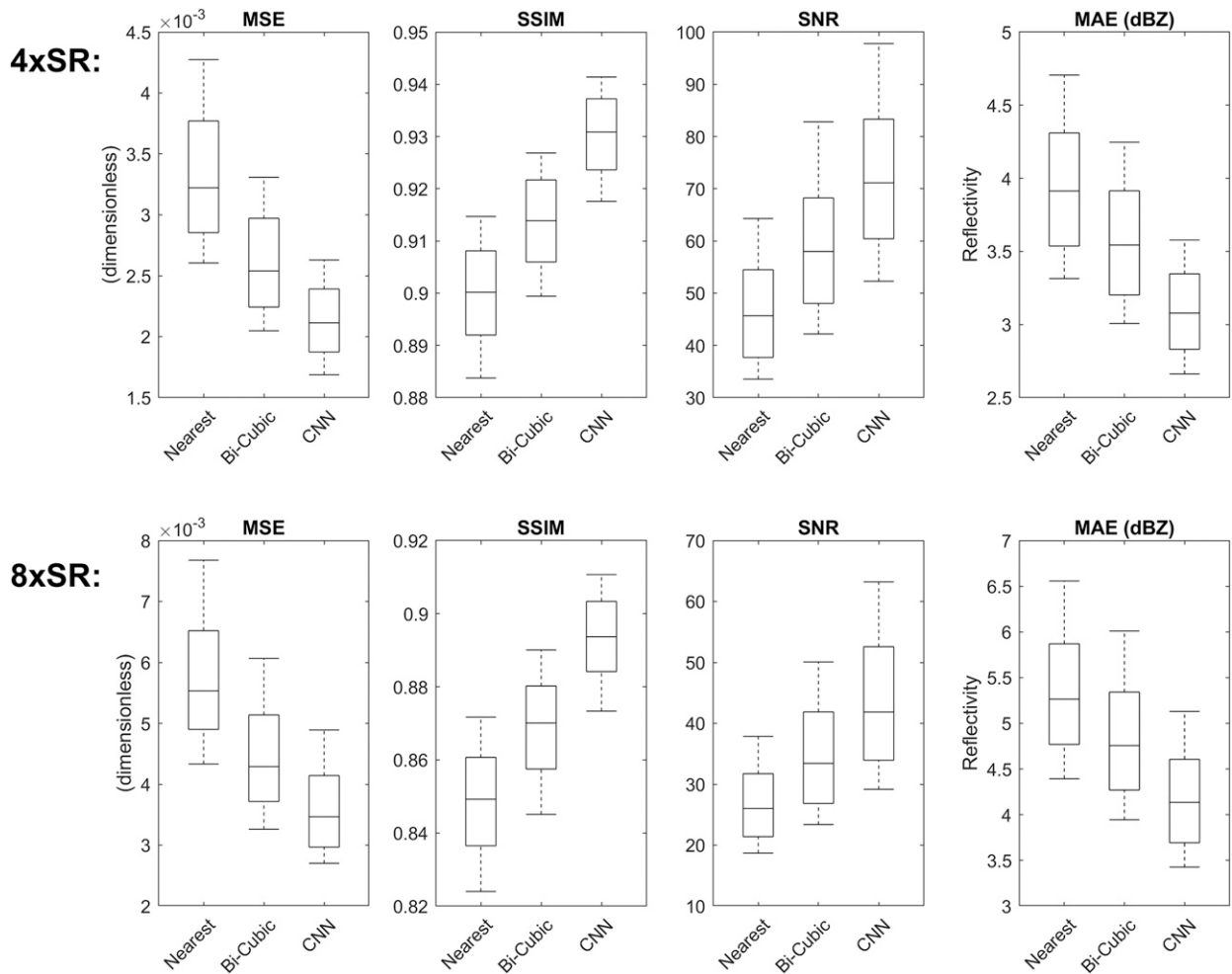


FIG. 7. Error metrics (as in Fig. 6) computed on the top 10% of the test set with the most precipitation. Many of the PPI scans contain large regions with no precipitation and both interpolation and the CNN will achieve high scores for these trivial cases. This is the same as Fig. 6, but only includes scans with many precipitating features. While all scores are slightly worse than in Fig. 6, there is a much larger difference between the performance of the interpolation scheme and the neural network.

of the downgraded scan, and the CNN notably avoids the ripple artifacts that are generated by the interpolation schemes at high wavenumbers to the right of this line. We also note that the PSD line for the neural network overlaps that of the original scan up to the Nyquist frequency (half the resolution) of the degraded scan. The interpolation schemes all lose some of the information even for these low wavenumbers.

Finally, we show the distribution of MSE, MAE, SSIM, and SNR [Eq. (2)] for the test set in Fig. 6, and compare to bicubic and nearest neighbor interpolation as benchmarks. The boxplots in Fig. 6 show the 10th–90th percentiles as whiskers, the 25th–50th percentiles as the box, and the median as the horizontal line in the box computed across all 11 381 samples in the test set. The CNN outperforms interpolation in each metric for both 4 \times SR and 8 \times SR. Note that for MSE and MAE (left and right columns of Fig. 6) lower scores are better while for SSIM and SNR (center columns) higher scores are better. The SSIM metric is defined in such a way that its spatial distribution

within an image can be computed. We show several examples of the spatial distribution of SSIM and the corresponding PPI scan in supplement section 3. The overall result is that portions of the scan that tend to have very high-frequency textured features tend to have lower SSIM scores which is perhaps to be expected. Comparison to SSIM computed for bicubic interpolation shows that the neural network gains a large advantage in this metric by preserving sharper edges for large precipitation features. Figure 7 is the same as Fig. 6, but here only the top 10% of scans in the test set by total (spatially averaged) reflectivity were used, meaning the 10% of scans with the most precipitation features. The CNN and interpolation both get high scores in these metrics for scans with very few precipitation features because performing SR on an empty scan is trivial. Including only the top 10% of scans by total precipitating features removes these trivial cases from the analysis, and in Fig. 7 there is even greater separation between the neural network scores and the interpolation scores, though all of the scores are lower than those in Fig. 6.

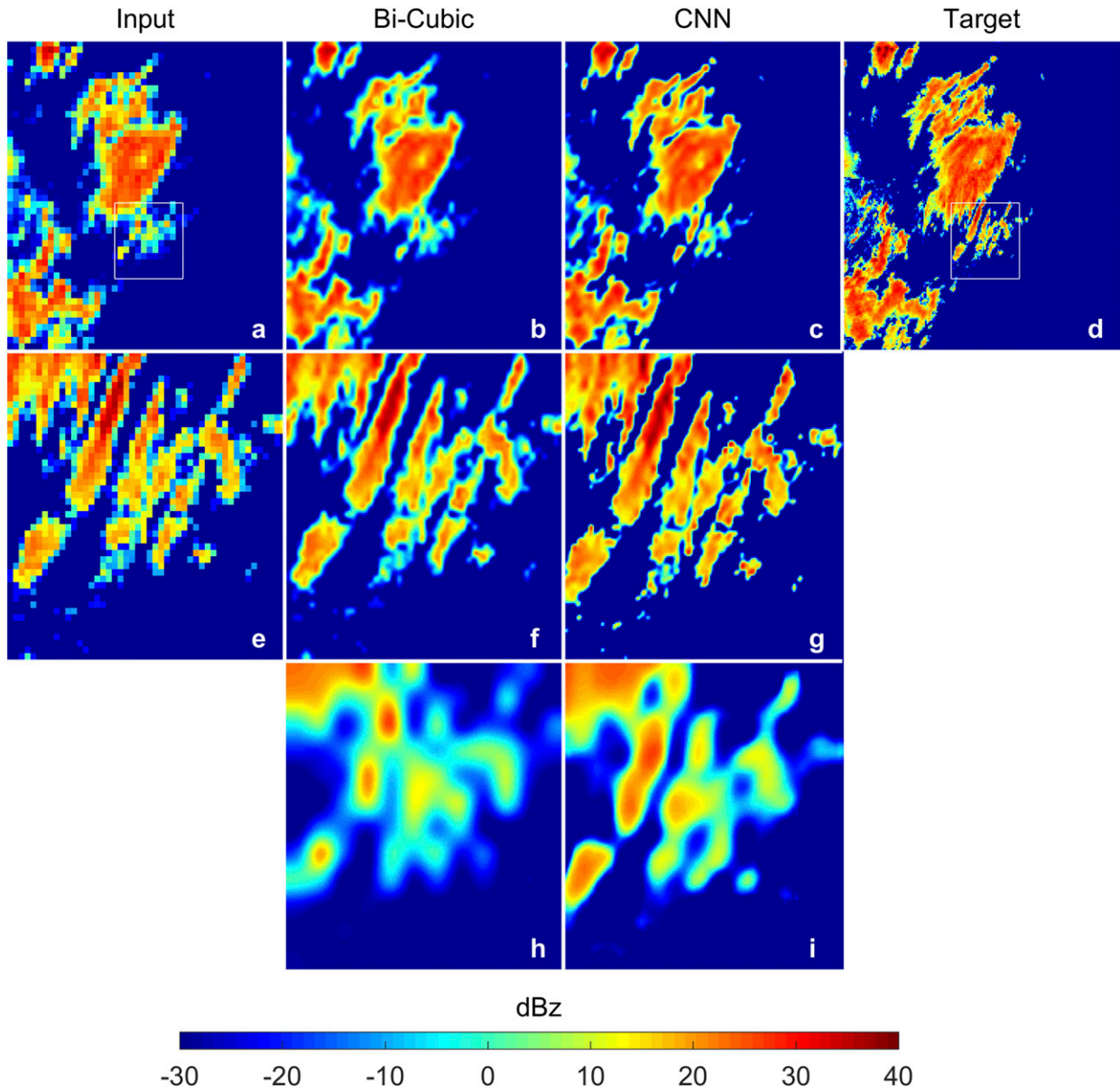


FIG. 8. Application of a neural network trained on randomly scaled and spatially shifted PPI scans to the same example scan shown in Fig. 4. (a)–(d) Application of this CNN to a 192×192 section of the PPI scan [see (d)] that has been degraded to a resolution of 48×48 [see (a)]. In (b) and (c) The results of applying bicubic interpolation and the neural network to (a), respectively, are shown. (e)–(g) Interpolation and the CNN are applied to the boxed region in (d), to increase beyond the resolution of the original scan. In this case there is no ground truth for comparison. (h),(i) Bicubic interpolation and the neural network are each applied twice to the boxed region in (a) to achieve $16\times$ super resolution. While the result is significantly worse than in (f) and (g), the neural network produces a much better-looking result than interpolation.

The CNN trained on randomly shifted and scaled samples of PPI scans also performs very well in terms of the MSE and MAE metrics listed in Table 1, and significantly outperforms interpolation. While we have not performed the same detailed analysis of PSD, SSIM, and SNR in this case, this neural network also produces subjectively superior results (Fig. 8). Figures 8a–d show application of this CNN to a degraded portion of the same PPI scan shown in Fig. 4. There were two

main objectives in training this network. The first is to show that the information learned by these CNNs is shift and scale invariant, which the MSE and MAE demonstrate. The other is that requiring only a portion of a PPI scan as input means that the CNN can easily be applied to sections of a scan that has not been degraded to increase the scan resolution beyond its native resolution. An example is shown in Figs. 8e–g, using the boxed region in Fig. 8a. Finally the bottom row in Fig. 8 (Figs. 8h–i)

show the result of applying the CNN to its own output (using the boxed region in Fig. 8d as input) to achieve $16\times$ SR, and again produce output with higher resolution than the native resolution of the scan. We note here that in doing this we have assumed that the sub-pixel-scale precipitation features that are too small to be resolved by the radar are similar to the larger features that are resolved, which is not necessarily true. While these results are intriguing and we believe the high-resolution output from the CNN is subjectively better than the result of bicubic interpolation, because we have increased beyond the native resolution of the radar there is no ground truth to which to compare our results.

5. Conclusions

Here we have demonstrated that CNN-based super resolution can significantly outperform traditional interpolation schemes when applied to radar data in terms of both pixel-wise error and several different perceptual quality metrics that measure structure, frequency content, and visual fidelity. Increased performance is at least partly a result of the wider perceptive field used in the convolutional neural networks, as well as prior knowledge of the statistics of storms and other weather features based on a large training set. The advantage becomes especially pronounced when applied to larger increases in spatial upsampling. This benefit was demonstrated using a variety of CNN models, including a U-net with dense blocks architecture that showed the best overall performance of the CNNs by a small margin.

Applying super resolution to weather radar data has many applications: increased resolution and frequency content improves the capability for observational comparisons with models operating at fine grid spacing. Physically consistent interpolation will help improve hydrological modeling and have applications to nonuniform beam filling. More qualitatively, increased resolution improves the visual fidelity of products such as isosurfaces and broadcast meteorology products. We believe this could have applications to the nonuniform beam filling problem often encountered in satellite retrievals and future work will focus on testing the bounds of this technique on other measurement platforms. This work lays out the basis for a super-resolution technique that is not limited to weather radar but has potential applications for a wide variety of instrumentation and model data. Very large gridded datasets from precipitation radars, satellite imagers, and climate and weather models for instance, are ubiquitous in the field of atmospheric science. This type of data is very well suited for recently developed convolutional-neural-network-based techniques, like the super-resolution scheme presented here, and further effort in this area has the potential to profoundly change the field moving forward.

Acknowledgments. The authors thank PNNL for hosting the Joint PNNL/UW measurements workshop, from which this original idea developed. AG designed and implemented code, ran experiments and analysis, and wrote the paper. JCH conceived of the original idea and helped with paper writing. The authors declare they have no competing interests. JCH's contributions were supported under the U.S. Department of

Energy Office of Science Biological and Environmental Research as part of the Atmospheric Systems Research Program.

Data availability statement. NEXRAD data are freely available from the National Climatic Data Center; the dataset used here was retrieved from <https://registry.opendata.aws/noaa-nexrad>. Our code is available from https://github.com/avgeiss/nexrad_sr.

REFERENCES

- Armanious, K., S. Abdulatif, F. Aziz, U. Schneider, and B. Yang, 2019: An adversarial super-resolution remedy for radar design trade-offs. arXiv, <https://arxiv.org/abs/1903.01392>.
- Baker, S., and T. Kanade, 1999: Super resolution optical flow. Carnegie Mellon University Robotics Institute Tech. Rep. CMU-RI-TR-99-36, 13 pp.
- Bharadwaj, N., 2009: Networked radar systems: Waveforms, signal processing and retrievals for volume targets. Ph.D. thesis, Colorado State University, 170 pp.
- Bringi, V. N., and V. Chandrasekar, 2001: *Polarimetric Doppler Weather Radar: Principles and Applications*. Cambridge University Press, 636 pp.
- Dong, C., C. Loy, K. He, and X. Tang, 2014: Image super-resolution using deep convolutional networks. arXiv, <https://arxiv.org/abs/1501.0092>.
- Gao, J., B. Deng, Y. Qin, H. Wang, and X. Li, 2017: Enhanced radar imaging using a complex-valued convolutional neural network. arXiv, <https://arxiv.org/abs/1712.10096>.
- Haug, G., Z. Liu, K. Weinberger, and L. Van der Maaten, 2017: Densely connected convolutional networks. arXiv, <https://arxiv.org/abs/1608.06993>.
- He, K., X. Zhang, S. Ren, and J. Sun, 2015: Deep residual learning for image recognition. arXiv, <https://arxiv.org/abs/1512.03385>.
- Huang, Y., W. Wang, and L. Wang, 2018: Video super-resolution via bidirectional recurrent convolutional networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, **40**, 1015–1028, <https://doi.org/10.1109/TPAMI.2017.2701380>.
- Ioffe, S., and C. Szegedy, 2015: Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv, <https://arxiv.org/abs/1502.03167>.
- Johnson, J., A. Alahi, and L. Fei-Fei, 2016: Perceptual losses for real-time style transfer and super-resolution. arXiv, <https://arxiv.org/abs/1603.08155>.
- Kim, J., J. Lee, and K. Lee, 2016: Accurate image super-resolution using very deep convolutional networks. arXiv, <https://arxiv.org/abs/1511.04587>.
- Kingma, D. P., and J. Ba, 2014: Adam: A method for stochastic optimization. arXiv, <https://arxiv.org/abs/1412.6980>.
- Lai, W.-S., J.-B. Huang, N. Ahuja, and M.-H. Yang, 2017: Deep Laplacian pyramid networks for fast and accurate super-resolution. arXiv, <https://arxiv.org/abs/1704.03915>.
- Ledig, C., and Coauthors, 2017: Photo-realistic single image super-resolution using a generative adversarial network. arXiv, <https://arxiv.org/abs/1609.04802>.
- Lim, B., S. Son, H. Kim, S. Nah, and K. Lee, 2017: Enhanced deep residual networks for single image super-resolution. arXiv, <https://arxiv.org/abs/1707.02921>.
- Long, J., E. Shelhamer, and T. Darrell, 2014: Fully convolutional networks for semantic segmentation. arXiv, <https://arxiv.org/abs/1411.4038>.
- Nasrollahi, K., and T. Moeslund, 2014: Super-resolution: A comprehensive survey. *Mach. Vis. Appl.*, **25**, 1423–1468, <https://doi.org/10.1007/s00138-014-0623-4>.

- Ohsaki, Y., and K. Nakamura, 1998: Simulation-based analysis of the error caused by non-uniform beam filling and signal fluctuation in rainfall rate measurement with a spaceborne radar. *J. Meteor. Soc. Japan*, **76**, 205–216, https://doi.org/10.2151/jmsj1965.76.2_205.
- Richard, A., I. Cherabier, M. R. Oswald, V. Tsiminaki, M. Pollefeys, and K. Schindler, 2020: Learned multi-view texture super-resolution. arXiv, <https://arxiv.org/abs/2001.04775>.
- Ronneburger, O., P. Fischer, and T. Brox, 2015: U-Net: Convolutional networks for biomedical image segmentation. arXiv, <https://arxiv.org/abs/1505.04597>.
- Tao, Y., and J.-P. Muller, 2018: Super-resolution restoration of MISR images using the UCL MAGiGAN system. *Remote Sens.*, **11**, 52, <https://doi.org/10.3390/rs11010052>.
- Timofte, R., V. De Smet, and L. Van Gool, 2015: A+: Adjusted anchored neighborhood regression for fast super-resolution. *12th Asian Conf. on Computer Vision*, Singapore, 111–126, https://doi.org/10.1007/978-3-319-16817-3_8.
- Veillette, M., E. Hassey, C. Mattioli, H. Iskenderian, and P. Lamey, 2018: Creating synthetic radar imagery using convolutional neural networks. *J. Atmos. Oceanic Technol.*, **35**, 2323–2338, <https://doi.org/10.1175/JTECH-D-18-0010.1>.
- Wang, Z., A. Bovik, H. Sheikh, and E. Simoncelli, 2004: Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.*, **13**, 600–612, <https://doi.org/10.1109/TIP.2003.819861>.
- , J. Chen, and S. Hoi, 2019: Deep learning for image super-resolution: A survey. arXiv, <https://arxiv.org/abs/1902.06068>.