Compressive Sensing Radar Imaging With Convolutional Neural Networks

QIAO CHENG\textsuperscript{\textregistered}, (Member, IEEE), ACHINTHA AVIN IHALAGE\textsuperscript{\textcopyright}, YUJIE LIU\textsuperscript{\textregistered}, AND YANG HAO\textsuperscript{\textregistered}, (Fellow, IEEE)

School of Electronic Engineering and Computer Science, Queen Mary, University of London, London E1 4NS, U.K.

Corresponding author: Yang Hao (y.hao@qmul.ac.uk)

This work was supported by the EPSRC and NIHR under “Medical Devices and Vulnerable Skin: Intelligent sensing to promote self-management” under Grant EP/N02723X/1.

ABSTRACT In the area of radar imaging at any frequency band from microwave to optics, the technique of compressive sensing (CS) enables high resolution with reduced number of antenna elements and measurements. However, CS methods suffer from high computational complexity and require parameter tuning to ensure good image reconstruction under different noise, sparsity and undersampling levels. To alleviate such issues, we present a machine learning approach that combines CS and convolutional neural network (CNN) for radar imaging. This CS based CNN (CS-CNN) method maintains good characteristics of CS methods, such as sparse sampling and high resolving power but is free from time-consuming computer optimization and demanding spaces for data storage. In the meantime, it is also robust to environment changes like noise, target sparsity and sampling rate. We have conducted extensive computer simulations for both qualitative and quantitative evaluations. Finally, we experimentally validate the technique with a demonstration of stable high resolution imaging using a sparse multiple-input multiple-output (MIMO) array where traditional imaging methods suffer from serious grating lobes. This approach is generic and can be easily extended to other applications of electromagnetic imaging and sensing.

INDEX TERMS Antenna arrays, compressive sensing, convolutional neural networks (CNN), machine learning, radar imaging.

I. INTRODUCTION

Microwave and millimeter-wave (mmW) imaging systems have received considerable attention owing to their capability of penetrating many optically opaque materials such as plastic and cloth while being strongly reflected by metallic materials. Radiation at these frequencies is non-ionizing unlike X-ray and therefore relieves the concerns about adverse effect on the human body. These characteristics make microwave and mmW imaging suitable for a wide variety of commercial and scientific applications like nondestructive testing (NDT), material characterisation, security scanning, and medical imaging. While microwave imaging is often used in NDT and medical imaging for its better penetration capability, mmW imaging with a frequency region from 30 to 300 GHz is more attractive for personnel surveillance imaging. This is because the relatively small wavelengths and wide bandwidths of mmW signals enable high resolving power in both the range and cross-range dimensions. Due to enormous advances made in semiconductor technology over the past few years, monolithic microwave integrated circuit (MMIC) with moderate costs are achievable in the mmW range and beyond [1]. Nonetheless, the fabrication of a full dense array imaging system with a high number of channels is still prohibitive, especially at higher frequencies in the THz region.

In recent years, compressive sensing (CS) has been successfully applied to many array imaging systems like switched array [2]–[5], phased array [6] and multiple-input multiple-output (MIMO) array [7]–[9] as well as reconfigurable antennas [10] and dynamic metasurfaces [11], [12]. This technique provides an alternative way for sparse array design which greatly reduces the required number of antennas and hence brings down the overall system cost. Data acquisition time can also be greatly reduced using sparse array and undersampling. Moreover, the CS recovery algorithms usually offer higher resolution reconstructions than traditional methods, thanks to the sparsity-driven optimization. Despite having so many advantages, CS approaches suffer
from high computational demands in image reconstruction and data storage. This is because CS algorithms are based on optimizations which are iterative in nature and thus require intensive computing power to converge. When the target scene becomes relatively large, the resulting large dimension sensing matrix can be problematic for storage and processing. This weakness hinders CS in certain applications where fast image reconstruction is necessary. In addition, the parameter configuration in CS algorithms is associated with many factors like array size and topology, sensing and sampling techniques, target scene sparsity, location and discretization, noise condition and so on. Any change of these factors can result in a different optimization problem that requires fine tuning of the parameters to ensure good reconstruction. This drawback seriously limits CS performance in many imaging applications where noise condition and target scene sparsity are changing rapidly.

More recently, several deep neural networks have been proposed for CS image reconstruction [13]–[15]. In [13], a sparse recovering framework based on stacked denoising autoencoder (SDA) was proposed to capture statistical dependencies between the different elements of certain signals. Similarly, Kulkarni et al. [14] presented a CNN-based non-iterative algorithm (ReconNet) to reconstruct images from compressively sensed random measurements. Then, Yao et al. [15] introduced a deep residual reconstruction network (DR2-Net) that consists of a linear mapping network for a high quality preliminary image first to be reconstructed and then further improved by the residual network. This combination was demonstrated to outperform both the SDA and ReconNet in terms of speed and quality. These machine learning approaches effectively avoid expensive iterative computations and seem to be a good alternative for many CS imaging applications. However, one important prerequisite for these methods to work is the availability of a large training data set with ground truth images. While this is not an issue for computer vision problems, ground truth images are often not available in many radar imaging applications. Moreover, the sensing matrix is usually more ill-conditioned and cannot be freely designed to have similar properties like a random matrix. Such restrictions make neural network implementations more challenging in radar applications. Nevertheless, several machine learning-based methods were proposed for SAR and ISAR image reconstruction [16]–[19]. All these methods assume that the ground truth images can be modelled by the convolution of the target scattering coefficients and an ideal point spread function (PSF). These synthetic ground truth images are then trained with synthetic raw data calculated using a simplified forward imaging model. While good performance has been reported, this approach neglected the fact that many factors affecting the imaging system cannot be easily modelled in simulations. Network models based on such synthetic training data only work well for measured data in a controlled environment.

In this paper, we propose a CS based CNN (CS-CNN) approach for fast and high-resolution radar image reconstruction. The network is trained with low resolution images generated by conventional methods and selected high resolution images by CS methods. Without need for parameter tuning and time-consuming computer optimization, the proposed approach achieves good image quality that is comparable to traditional CS methods but outperforms them in the case of low SNR and sparsity levels. Simulation and experiments have been carried out to verify the effectiveness of the proposed method. The rest of the paper is organized as follows. The conventional CS imaging model and the proposed CS-CNN framework are introduced in II. Section III gives the details of simulations and experiments, followed with a short discussion of real world applications. Finally, Section IV presents the conclusion.

II. CS-BASED CNN RADAR IMAGING

A. IMAGING MODEL

A typical array imaging system is shown in Fig. 1, with a 2-D planar array located in the x-y plane being responsible for transmitting and receiving electromagnetic waves. Assuming the Born approximation and ignores the multi-scattering, the scattered field at the receiving antenna \( \vec{r}_R \) due to electromagnetic wave from the transmitting antenna \( \vec{r}_T \) can be expressed as

\[
\mathbf{s}(\vec{r}_T, \vec{r}_R, k) = \iint_V g(\vec{r}) e^{-j k (\vec{r}_T + \vec{r}_R)} \frac{1}{R_T R_R} \, dx \, dy \, dz, \tag{1}
\]

where \( k \) is the wavenumber, \( g(\vec{r}) \) is the reflectivity coefficients of the target scene, \( R_T = |\vec{r} - \vec{r}_T| \) and \( R_R = |\vec{r} - \vec{r}_R| \) are the distances from the transmitter and receiver to the observing point \( \vec{r} \), respectively. This equation is for the general MIMO case, where the transmit and receive arrays can be designed in arbitrary shapes. Simplifications can be made for the switched array case by setting \( R_R \) the same as \( R_T \). There are many recovery algorithms available to retrieve the target scene from the measured data. Conventionally, the radar image at a location \( \vec{r} \) is given by the back propagation (BP) method

\[
I(\vec{r}) = \sum_{N_T} \sum_{N_R} \sum_{N_g} \mathbf{s}(\vec{r}_T, \vec{r}_R, k) e^{j k (R_T + R_R)}, \tag{2}
\]
where \( N_f, N_T, N_R \) are the number of frequencies, transmit and receive antennas, respectively. This formula can also be efficiently calculated using fast Fourier transform under the condition of uniform sampling in the \( x \)-\( y \) plane and further approximations [20].

On the other hand, CS approaches try to reformulate (1) into a matrix multiplication form with additional undersampling functionality as

\[
y = AHg + n,
\]

where \( y \) is a vector of the undersampled data, \( A \) is the undersampling operator that offers frequency and spatial undersampling, \( g \) is the vector of the reflectivity coefficients, \( H \) is the system response matrix, and \( n \) is the noise vector. The dimension of \( H \) determined by the number of Tx/Rx pairs and voxels (pixels for 2D case). Then each element of \( H \) is calculated using (1) for a specific Tx/Rx pair and voxel position. This underdetermined system is usually solved by rewriting it as the following optimization problem

\[
\min_g \|g\|_1 \quad s.t. \quad \|AHg - y\|_2 \leq \varepsilon,
\]

where \( \varepsilon \) is a non-negative real parameter that defines the noise level. The \( \ell_1 \) norm is suitable for most sparse target scenes, other sparsifying transformations can also be used depending on the type of the target scene. CS recovery algorithms is based on minimizing the residual while also imposing sparsity constraints on the solution. Consequently, the computational complexity and data storage become problematic for 3D and real-time imaging applications. To make it worse, the parameter configuration in such algorithms can be easily affected by the system and environment changes. Such practical issues will be addressed in the following subsection.

**B. CS-CNN IMAGING FRAMEWORK**

In radar imaging the received data can be considered as a mapping of the scene reflectivity function. This mapping is nonlinear due to the complex electromagnetic wave scattering. In this machine learning imaging framework, we try to use CNN to iteratively learn the non-linearities of the underlying physical model. The framework of the proposed CS-CNN is illustrated in Fig. 2. Training samples for the network are obtained directly from the raw radar measurements. More specifically, the input images can be either raw measurements or low resolution images calculated using conventional imaging algorithms. On the other hand, the output images are high resolution images calculated using CS algorithms. In comparison to conventional methods, CS methods rely less on approximations and thus offer more accurate representation of the true physical model when used as output images. It is worth noting that although ideal PSF can also be used to generate high resolution output images, it is difficult and impractical to do so in real experiments. More importantly, network models based on pure synthetic data only consider limited system factors. Their performance can be degraded in practical scenarios. In contrast, CS reconstructions can be easily obtained in both simulations and experiments and thus are more practical to be used as output images. Given a sufficient amount of training data, we expect the CNN to reconstruct images with comparable performance like CS methods. In addition, undersampling with suppressed grating lobes can also be achieved just like CS methods. More importantly, the image reconstruction process will be much faster than CS methods as it only involves few simple matrix multiplications.

As aforementioned, CS algorithms require parameter tuning for different environment conditions to ensure good reconstruction. To overcome this issue, we can feed the network with good reconstruction samples from a few limited conditions and let the neural network learn how to cope with different environments. The trained network will be able to infer good reconstructions under a more broad environment conditions. In such a way, we avoid generating samples for every possible condition. More specifically, suppose we are imaging in a noise varying environment. We first consider few SNR cases across a wide range of SNR conditions, with fine tuned parameters for each SNR case. Then a large quantity of CS images are generated for each SNR case with randomly distributed targets each time. These images at a limited number of SNR cases are fed into the network to learn how to imaging with more different noise levels. It is worth noting that even there exists an optimum parameter for each SNR case, bad reconstructions still occur due to the randomness nature of noise. Such bad reconstructions are excluded from the training samples.

Fig. 3 illustrates the neural network architecture for the experimental data obtained using a sparse MIMO array. Opposed to treating the magnitude and phase images as two separate channels of the same image, we enable better feature extraction by processing each data stream independently with stacked convolutional layers. This implies that each branch is only specialized in learning vital features from the corresponding input (either magnitude or phase) rather than trying to extract features from both datasets simultaneously. Likewise, our strategy stands out from the existing approaches that employ magnitude-only input images [18] or complex-valued CNNs [16], [21] that are usually coupled
FIGURE 3. Full convolutional neural network based on experimental data. The magnitude and phase of the input data are processed separately with two independent 2D-convolutional layer blocks. These two branches are fused with a concatenation layer, followed by three convolutional-transpose (deconvolutional) layers for image upsampling. A Lambda layer is introduced at the end to account for size mismatch between the predicted image and the target image.

with high implementation and computational complexity. Input and output dimensions of the network are 60 × 60 and 100 × 100, respectively. A 3 × 3 kernel is used in all convolutional/deconvolutional layers. The stride parameter is set to (2, 2) in all layers except ConvM3 and ConvP3, where the stride is (1, 1). L2 regularization parameter is set to 1e−4 in ConvM1 and ConvP1 layers to mitigate overfitting phenomenon. Rectified linear unit (ReLU) is used as the activation function in all hidden layers. Note that the Lambda layer at the end is introduced to account for size mismatch between the predicted image and the target image. This is particularly useful as the input image dimension varies a lot with different imaging systems and configurations. The database consists of 5760 samples, each containing a BP image and a CS image. The database is split into 90% training and 10% test sets. Mean squared error is used as the loss function which is optimized using Adam optimizer [22]. Learning rate is calibrated to 0.001 and the exponential decay parameters, β1 and β2 are fixed at 0.9 and 0.999. The model is trained for around 20 epochs and the performance is monitored on a validation set generated by randomly sampling 20% of training data. Final predictions are made on the holdout test set. The model is implemented in python programming language environment using Keras deep learning framework [23] with TensorFlow [24] backend.

III. SIMULATION AND EXPERIMENTAL RESULTS

In this section, we evaluate the effectiveness of the proposed CS-CNN framework with both simulated and measured data. For the CS recovery algorithm, we adopt the split augmented lagrangian shrinkage algorithm (SALSA) [25]. SALSA is based on a variable splitting to obtain an equivalent constrained optimization formulation, which is then addressed using the alternating direction method of multipliers (ADMM). SALSA is particularly suitable for image reconstruction problems and allows for both wavelet-based
FIGURE 4. Image reconstruction results. From left to right are ground truth, CS, CS-CNN and BP reconstructions, respectively. From top to bottom are three different target scenes.

A. SIMULATION
In the simulation, we assume a 20 cm × 20 cm uniform switched array of 400 transceivers is operating at 30 GHz with point scatterers randomly distributed in a plane 40 cm away from the aperture. The raw data is obtained using the forward imaging model (1). Input images are calculated using the BP algorithm (2). CS reconstructions are obtained with the SALSA solver and used as the output of the network. The CS reconstructions are set to have 100 × 100 pixels. In order to train the network to distinguish different sparsity levels, the number of scatterers is randomly varied in each simulation. A total of 9078 samples are generated including three SNR levels at 10 dB, 20 dB and 30 dB. Note that the noise was added to the raw echoed data before reconstruction. Two parameter configurations are used for the SALSA solver: \( \tau = 1e^{-3}, \mu = 1e^{-2}, tol = 1e^{-6} \) and \( \tau = 1e^{-4}, \mu = 1e^{-2}, tol = 1e^{-6} \). The former is more suitable for the case of 10 dB and 20 dB SNR while the latter is more suitable for the case of 30 dB SNR.

Fig. 4 gives three image reconstruction examples at different environment configurations. From left to right, each column represents ground truth, images reconstructed by CS, CS-CNN and BP, respectively. All images are plotted in dB scale with a dynamic range of 40 dB. The first case shown in the first row has 5 point scatterers with noise level at SNR = 5 dB. It can be observed that both CS and CS-CNN methods correctly resolved all 5 scatterers. However, the CS reconstruction also comes with a lot of background noise which can result in false detection of targets. This is a common example when CS parameters become unsuitable.
for the reconstruction. This often happens in case of noisy environments, data undersampling and complex target scene. Although fine tuning of CS parameters can help to some extent, it is troublesome and not practical to consider all different factors that affect the CS image reconstruction. The BP method, on the other hand, failed to resolve the two closely spaced scatterers on the top due to resolution limitation. Strong side lobes are also accompanied with each scatterer. The second case shown in the second row has 15 point scatterers with noise level at SNR = 15 dB. Due to the improved noise condition, CS reconstructions are relatively more stable than in the first case. It can be clearly observed both CS and CS-CNN reconstructions achieve good performance with all scatterers correctly reconstructed. The BP reconstruction still has low resolution with serious sidelobe interference as expected. The last case shown in the third row has 45 point scatterers with a noise level at SNR = 25 dB. Due to the improved noise condition, CS reconstructions are relatively more stable than in the first case. It can be clearly observed both CS and CS-CNN reconstructions achieve good performance with all scatterers correctly reconstructed. The BP reconstruction still has low resolution with serious sidelobe interference as expected. The last case shown in the third row has 45 point scatterers with a noise level at SNR = 25 dB. In this case, some scatterers are more closely spaced than previous examples which makes it quite challenging to resolve them. As can be seen from Fig. 4(j), the CS reconstruction shows few unwanted artifacts in some regions. Similarly, the CS-CNN method shows a slightly worse reconstruction with image ghosting in areas where scatterers are closely packed. This is expected as the CNN trained model is only an approximation of the true physical model. In other words, the resolving power of the trained model can only approach the physical model in certain degrees. This approximation will more accurate with increased number of training samples. Therefore, to limited training samples in practice, the resolving power of the CS-CNN method will always be weaker than the CS method. Nevertheless, the CS-CNN reconstruction still outperforms the BP one.

To investigate how the CS-CNN performs in more practical situations, we tested another target as shown in Fig. 5. This target is a pixelated version of the Queen Mary University of London (QMUL) logo. There are 443 point scatterers in total, which has much more scatterers than the network trained for. Three SNR values at 0 dB, 10 dB and 20 dB are tested. The reconstructed images are demonstrated in the rest of Fig. 6. Clearly, the CS-CNN images outperform the CS images in low SNR conditions as the target contours are easily discernible and backgrounds are cleaner. When compared to the ground truth, both methods failed to reconstruct the details of all scatterers in the high SNR case shown in Fig. 6(c) and Fig. 6(f). However, the two methods differ from each other in terms of imaging outcome. The CS method offers better resolving power with a sparse reconstruction but missed many scatterers. On the other hand, the CS-CNN method comes with a lower resolving power but gives a more coherent image.

Previous results only show a qualitative comparison of images, to quantitatively evaluate image quality, we adopt the Structural Similarity Index (SSIM) for comparison. The SSIM was reported to outperform more often used quality metrics like MSE and peak signal-to-noise ratio (PSNR) [29]. The SSIM varies from 0 to 1 with larger values represent higher similarities. The test data is generated with varied SNR ranging from 0 dB to 30 dB with 5dB spacing. For each SNR case, the target scene sparsity level is varied by changing the number of point scatterers from 5 to 55 with a spacing of 10. For each SNR and sparsity pair, we run 100 trials with all scatterers randomly distributed in each trial. Therefore, a total of 4200 samples are generated and tested. Then, the averaged SSIM for each SNR and sparsity pair are calculated and plotted in Fig. 7(a). The two CS methods are both based on the SALSA solver with the same two parameter configurations showed earlier in this section. As can be seen from the surf plot, the CS-CNN method outperforms both CS methods in the case of low SNR and low sparsity levels (large number of scatterers). Their performances become similar in the case of high SNR and high sparsity levels (small number of scatterers). However, the CS-CNN method becomes slightly worse than CS methods in the case of high SNR and low sparsity levels due to relatively weaker resolving power. Fig. 7(b) gives a better visualization of the SSIM comparison with a fixed sparsity level of 55 point scatterers. The two CS methods behave differently with one suitable for the lower SNR cases and the other suitable for the higher SNR cases. This performance difference indicates the need for parameter tuning to ensure good CS reconstruction. It can also be noticed that the CS-CNN method achieves a relatively more flat SSIM curve as compared to both CS methods. This proves the CS-CNN method is more robust to noise variation. Similarly, we can also observe increased robustness against sparsity variation in the low SNR cases in Fig. 7(a).

As emphasized earlier that the CS-CNN method can be much more efficient than CS methods, here we verify this claim by calculating the time and space complexities for all four methods showed in Fig. 7. More specifically, the average time needed to reconstruct an image is used as the time complexity. Then, the space complexity represents the space needed to store necessary data for reconstruction, e.g., sensing matrix for CS methods or network architecture and the weights for the CS-CNN method. The computer used for test comes with an Intel i9-9900k CPU and a Nvidia Titan V GPU. As can be seen from Table 1, the CS-CNN method
is overwhelmingly faster than both CS methods, thanks to its iteration-free calculation. It is worth noting that CS-CNN method is also considerably faster than the BP method. Therefore, if we consider the BP images are used as input for the network, the overall processing time of the CS-CNN approach is at the same level as the BP method. This speed-up is particularly attractive for 3D and real-time imaging applications. Another significant aspect of the CS-CNN is its low space complexity. While the CS methods in this case took 33.8 MB space to store a 400 \times 100000 complex-valued sensing matrix in double precision, the saved CS-CNN network only occupies 2.1 MB space. This complexity reduction becomes more pronounced as the dimension of the sensing matrix increases. In fact, the number of neural network parameters are independent of the input/output image sizes. Therefore, while the space complexity of the CS method has approximately quadratical growth with increasing dimension, the space required to store the CS-CNN model would be almost constant.

### TABLE 1. Time and space complexities of different methods.

| Methods | BP | CS1 | CS2 | CS-CNN |
|---------|----|-----|-----|--------|
| Time    | 0.1 s | 16.5 s | 11.4 s | 0.002 s |
| Space   | 0 MB | 33.8 MB | 33.8 MB | 2.1 MB |

#### B. EXPERIMENT

In the numerical examples we used a planar switched array for imaging. However, it would be too costly to build such
an array with 400 elements in practice. Fortunately, it is well known that MIMO array systems can achieve similar performances like traditional dense arrays but with far fewer elements [20]. Therefore, we built a 2D sparse MIMO array imaging system [30] to evaluate the proposed imaging algorithm. As shown in Fig. 8, the MIMO array consists of 12 transmit and 12 receive elements and are connected to a 24-port Rohde & Schwarz vector network analyzer (VNA) for signal transmission and reception. The array aperture is 42.5 cm × 42.5 cm with Vivaldi antennas operating from 3.5 GHz to 8.5 GHz. Here we only consider 2D image reconstruction with signals obtained at 6 GHz. The element spacing of the equivalent virtual aperture is around 3.9 cm, which translates to 0.8 wavelength at 6 GHz.

As described previously, random point scatterers were adopted to generate training samples in the numerical simulations and demonstrated good performance in final image prediction. Intuitively, it would be great to take the same approach in the experiment. However, it is quite challenging to design such a target scene with randomly distributed scatterers whose total number of scatterers is also changing all the time, let alone measuring them for several thousand times. Therefore, to simplify the experiment, we fabricated two small disks and wrapped them with aluminum foil to work as point targets. As shown in Fig. 8(a), both targets are mounted on a NSI-2000 near-field scanner and raster scanned to acquire a large number of samples in a very short time. A laptop was used to control both the scanner and the VNA for real-time data acquisition. We varied the edge to edge spacing of the two targets \( d \) to get more training samples. It should be noted that many imaging applications like magnetic resonance imaging (MRI) have access to a variety of online datasets. Training samples can be easily generated from such database.

The network was trained with a total number of 5760 selected samples from \( d = 1, 3 \) and 5 cm. Fig. 9 demonstrates the image reconstructions of four target scenes with \( d = 1 \) cm. From left to right, each column represents images reconstructed by CS, CS-CNN and BP methods, respectively. As can be seen from the third column, the BP algorithm failed to resolve the two closely spaced targets. In addition, we can also observe very strong grating lobes surrounding the true targets. This is because the target scene is undersampled with the equivalent element spacing larger than half a wavelength. By contrast, judging from the first target scene, both CS and CS-CNN methods successfully reconstructed the two targets without any grating lobes or side lobes. However, In the rest three target scenes, the CS method failed to give faithful reconstructions like the CS-CNN method. The three CS reconstructions with either too sparse or not sparse enough results are very common examples due to lack of CS parameter tuning. Moreover, due to undersampling with only 144 channels of data, the CS algorithm becomes very sensitive to the parameter configurations, which resulted in a large number of failed reconstructions.

As the ground truth is often not available in such real world experiments, it is difficult to calculate traditional metrics which are often used in simulations. Therefore, we present a quantitative evaluation metric by counting the successful reconstruction rates of each algorithm. A successful reconstruction is only considered when two targets are easily distinguishable and no false targets and grating lobes appear. A total of 3041 test samples were obtained from a separate measurement campaign with target spacing \( d = 2 \) cm. The successful reconstruction rates of CS and CS-CNN methods are summarized in Table 2. Parameter configuration of \( \tau, \mu, tol \) of the three CS methods are \((1e-3, 1e-1, 1e-5), (1e-4, 1e-1, 1e-4)\) and \((1e-3, 1e-1, 1e-4)\), respectively. It can be noticed that even a small parameter variation can result in a distinct reconstruction performance. Moreover, the best CS configuration only achieves 56% successful rate which makes it impractical for accurate imaging in such undersampling conditions. On the other hand, the CS-CNN method gives an extraordinary high successful rates without need for parameter tuning.

| Methods | CS1 | CS2 | CS3 | CS-CNN |
|---------|-----|-----|-----|---------|
| Successful rate | 25.7% | 2.3% | 56.2% | 97.0% |

**TABLE 2. Successful reconstruction rates of different methods.**

**C. DISCUSSION**

To this end, we can safely conclude that the proposed CS-CNN approach offers much better efficiency and robustness and still achieves good imaging quality. As compared to conventional CS imaging, the implementation of this method will be much easier for practical applications due to low computational complexity and small data storage. The CS-CNN approach is particularly useful for short-range personnel surveillance imaging to improve image resolution and speed up image reconstruction. Real-time high resolution imaging is achievable with electronically scanned array systems. Moreover, the adoption of sparse array design helps to reduce the system cost with much fewer antennas. The undersampling capability is also attractive for MRI systems where scanning is extremely slow and requires patients to hold still during the whole process. The CS-CNN approach can greatly speed up the data acquisition and makes MRI more efficient for both doctors and patients. More specifically, we can train
the network with reconstruction pairs from undersampled and fully sampled data. The trained model will be able to reconstruct good images with undersampling patterns. This approach is also beneficial for computational imaging and inverse scattering problems where image formation usually involves computationally intensive optimizations. We expect to see more CS-CNN applications in future work.

**IV. CONCLUSION**

In this paper, we proposed a machine learning imaging method based on CNN. The network architecture consists of 9 hidden layers with both real and imaginary input data which are fused mid way through the network, enabling vigorous feature extraction. The trained network inherits the advantages of CS methods like high resolving power but requires significantly less computing power. Moreover, its reconstructions remain robust against changing environment conditions without need for tedious parameter tuning like CS methods. We demonstrate the effectiveness of the proposed scheme with simulations on a switched array and experiments on a sparse MIMO array.

**ACKNOWLEDGMENT**

The authors would also like to thank Dr A. Andy and G. Simpson for their help in setting up the MIMO array imaging system.
REFERENCES

[1] S. Ahmed, A. Schiessl, F. Gumbmann, M. Tiebout, S. Methfessel, and L.-P. Schmidt, “Advanced microwave imaging,” IEEE Microw. Mag., vol. 13, no. 6, pp. 26–43, Sep. 2012.

[2] Q. Cheng, A. Alomainy, and Y. Hao, “On the performance of compressed sensing-based methods for millimeter-wave holographic imaging,” J. Opt., vol. 55, no. 4, pp. 728–738, 2016.

[3] S. Li, G. Zhao, H. Sun, and M. Amin, “Compressive sensing imaging of 3-D object by a holographic algorithm,” IEEE Trans. Antennas Propag., vol. 66, no. 12, pp. 7295–7304, Dec. 2018.

[4] H. Zaman and M. Fakharzadeh, “1.5-D sparse array for millimeter-wave imaging based on compressive sensing techniques,” IEEE Trans. Antennas Propag., vol. 66, no. 4, pp. 2008–2015, Apr. 2018.

[5] S. Hu, C. Shu, Y. Alfadhli, and X. Chen, “THz sparse periodic array imaging system using compressed sensing,” IET Microw. Antennas Propag., vol. 14, no. 11, pp. 1157–1161, Sep. 2020.

[6] Q. Cheng, A. Alomainy, and Y. Hao, “Near-field millimeter-wave phased array imaging with compressive sensing,” IEEE Access, vol. 5, pp. 18975–18986, 2017.

[7] X. Hu, N. Tong, Y. Zhang, G. Hu, and X. He, “Multiple-input–multiple-output radar super-resolution three-dimensional imaging based on a dimension-reduction compressive sensing,” IET Radar, Sonar Navig., vol. 10, no. 4, pp. 757–764, Apr. 2016.

[8] Q. Cheng and Y. Hao, “Study on sparse MIMO array for compressive sensing imaging,” in Proc. IEEE Conf. Antenna Meas. Appl. (CAMA), Sep. 2018, pp. 1–3.

[9] S. Hu, C. Shu, Y. Alfadhli, and X. Chen, “W band imaging system using linear sparse periodic antenna array and compressive sensing for personnel screening,” IEEE Access, vol. 7, pp. 173603–173611, 2019.

[10] Q. Cheng, A. Alomainy, and Y. Hao, “Luneburg lens imaging with compressive sensing,” in Proc. 6th Asia–Pacific Conf. Antennas Propag. (APCPAC), Oct. 2017, pp. 1–3.

[11] J. N. Gollub, O. Yurduseven, K. P. Trofatter, D. Arnitz, M. F. Imani, T. Sleasman, M. Boyarsky, A. Rose, A. Pedross-Engel, H. Odabasi, T. Zvolensky, G. Lipworth, D. Brady, D. L. Marks, M. S. Reynolds, and D. R. Smith, “Large metasurface aperture for millimeter wave computational imaging at the human-scale,” Sci. Rep., vol. 7, no. 1, May 2017, Art. no. 42650.

[12] L. Li, Y. Shuang, Q. Ma, H. Li, H. Zhao, M. Wei, C. Liu, C. Hao, C.-W. Qiu, and T. J. Cui, “Intelligent metasurface imager and recognizer,” Light, Sci. Appl., vol. 8, no. 1, pp. 1–9, Dec. 2019.

[13] A. Mousavi, A. B. Patel, and R. G. Baraniuk, “A deep learning approach to structured signal recovery,” in Proc. 53rd Asilomar Conf. Commun., Control, Comput., Sep. 2015, pp. 1336–1343.

[14] K. Kulkarni, S. Loith, P. Turaga, R. Kerviche, and A. Ashok, “ReconNet: Non-iterative reconstruction of images from compressively sensed measurements,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 449–458.

[15] H. Yao, F. Dai, S. Zhang, Y. Zhang, Q. Tian, and C. Xu, “DR2-net: Deep residual reconstruction network for image compressive sensing,” Neurocomputing, vol. 359, pp. 483–493, Sep. 2019.

[16] J. Gao, B. Deng, Y. Qin, H. Wang, and X. Li, “Enhanced radar imaging using a complex-valued convolutional neural network,” IEEE Geosci. Remote Sens. Lett., vol. 16, no. 1, pp. 35–39, Jan. 2019.

[17] X. Gao, D. Qin, and J. Gao, “Resolution enhancement for inverse synthetic aperture radar images using a deep residual network,” Microw. Opt. Technol. Lett., vol. 62, no. 4, pp. 1588–1593, Apr. 2020.

[18] C. Hu, L. Wang, Z. Li, and D. Zhu, “Inverse synthetic aperture radar imaging using a fully convolutional neural network,” IEEE Geosci. Remote Sens. Lett., vol. 17, no. 7, pp. 1203–1207, Jul. 2020.

[19] H. Mu, Y. Zhang, C. Ding, Y. Jiang, M. H. Er, and A. C. Kot, “Deep-Imaging: A ground moving target imaging based on CNN for SAR-GMTI system,” IEEE Geosci. Remote Sens. Lett., early access, Jan. 30, 2020, doi: 10.1109/LGRS.2020.2967456.

[20] X. Zhuge and A. G. Yarovoy, “Study on two-dimensional sparse mimo uwb arrays for high resolution near-field imaging,” IEEE Trans. Antennas Propag., vol. 60, no. 9, pp. 4173–4182, Dec. 2012.

[21] Z. Zhang, H. Wang, F. Xu, and Y.-Q. Jin, “Complex-valued convolutional neural network and its application in polarimetric SAR image classification,” IEEE Trans. Geosci. Remote Sens., vol. 55, no. 12, pp. 7177–7188, Dec. 2017.

[22] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2017, arXiv:1412.6980. [Online]. Available: https://arxiv.org/abs/1412.6980

[23] F. Chollet. (2015). Keras. https://github.com/fchollet/keras

[24] M. Abadi. (2015). TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. http://tensorflow.org/

[25] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, “Fast image recovery using variable splitting and constrained optimization,” IEEE Trans. Image Process., vol. 19, no. 9, pp. 2345–2356, Sep. 2010.

[26] A. Beck and M. Teboulle, “A fast iterative shrinkage-thresholding algorithm for linear inverse problems,” SIAM J. Imag. Sci., vol. 2, no. 1, pp. 183–202, Jan. 2009.

[27] J. M. Bioucas-Dias and M. A. T. Figueiredo, “A new TwIST: Two-step iterative Shrinkage/Thresholding algorithms for image restoration,” IEEE Trans. Image Process., vol. 16, no. 12, pp. 2992–3004, Dec. 2007.

[28] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, “Sparse reconstruction by separable approximation,” IEEE Trans. Signal Process., vol. 57, no. 7, pp. 2479–2493, Jul. 2009.

[29] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” IEEE Trans. Image Process., vol. 13, no. 4, pp. 600–612, Apr. 2004.

[30] Q. Cheng, Y. Liu, H. Zhang, and Y. Hao, “A generic spiral MIMO array design method for short-range UWB imaging,” IEEE Antennas Wireless Propag. Lett., vol. 19, no. 5, pp. 851–855, May 2020.