

Fast Enhanced CT Metal Artifact Reduction Using Data Domain Deep Learning

Muhammad Usman Ghani^{ID} and W. Clem Karl^{ID}, *Fellow, IEEE*

Abstract—Filtered back projection (FBP) is the most widely used method for image reconstruction in X-ray computed tomography (CT) scanners, and can produce excellent images in many cases. However, the presence of dense materials, such as metals, can strongly attenuate or even completely block X-rays, producing severe streaking artifacts in the FBP reconstruction. These metal artifacts can greatly limit subsequent object delineation and information extraction from the images, restricting their diagnostic value. This problem is particularly acute in the security domain, where there is great heterogeneity in the objects that can appear in a scene, highly accurate decisions must be made quickly, and processing time is highly constrained. The standard practical approaches to reducing metal artifacts in CT imagery are either simplistic nonadaptive interpolation-based projection data completion methods or direct image post-processing methods. These standard approaches have had limited success. Motivated primarily by security applications, we present a new deep-learning-based metal artifact reduction approach that tackles the problem in the projection data domain. We treat the projection data corresponding to dense, metal objects as missing data and train an adversarial deep network to complete the missing data directly in the projection domain. The subsequent complete projection data is then used with conventional FBP to reconstruct an image intended to be free of artifacts. This new approach results in an end-to-end metal artifact reduction algorithm that is computationally efficient and therefore practical and fits well into existing CT workflows allowing easy adoption in existing scanners. Training deep networks can be challenging, and another contribution of our work is to demonstrate that training data generated using an accurate X-ray simulation can be used to successfully train the deep network, when combined with transfer learning using limited real data sets. We demonstrate the effectiveness and potential of our algorithm on simulated and real examples.

Index Terms—Computed tomography, metal artifact reduction, sinogram completion, deep learning.

I. INTRODUCTION

METAL artifacts are common problems in conventional computed tomographic (CT) scanners using

Manuscript received April 1, 2019; revised June 26, 2019; accepted July 10, 2019. Date of publication August 27, 2019; date of current version January 13, 2020. This work was supported by the U.S. Department of Homeland Security, Science and Technology Directorate, Office of University Programs, under Grant Award 2013-ST-061-ED0001. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Patrick La Riviere. (*Corresponding author: Muhammad Usman Ghani.*)

The authors are with the Department of Electrical and Computer Engineering, Boston University, Boston, MA 02215 USA (e-mail: mughani@bu.edu; wckarl@bu.edu).

This article has supplementary downloadable material available at <http://ieeexplore.ieee.org>, provided by the authors. The material includes additional experimental results and information. This material is 10.3 MB in size.

Digital Object Identifier 10.1109/TCI.2019.2937221

poly-energetic X-ray sources coupled with energy integrating detectors. When there is a dense metallic object in the field of view, it highly attenuates or completely blocks the X-rays for the corresponding detector locations. When these data are used by the conventional filtered back projection (FBP) method, the resulting image exhibits severe streaking artifacts and inaccurate CT numbers, as illustrated in Figure 1. Such streaking then leads to errors in material or tissue type identification and object segmentations that can severely degrade automated threat recognition in security settings or clinical diagnosis in medical settings. Boas *et al.* [1] reported 21% of medical scans in their study were subject to metal artifacts. The problem is worse in security applications, where metallic objects appear in many stream of commerce items [2], [3]. Thus, metal artifact reduction (MAR) is an important problem in CT imaging.

Motivated by the security application, this paper aims to reduce metal artifacts in CT images by applying deep learning (DL) directly in the projection-domain, prior to image formation. Unlike typical image post-processing approaches, by introducing learning in the projection data domain we aim to avoid the creation of artifacts altogether. Additionally, in our approach, metal-contaminated data is completely removed, therefore the type of metal is not critical. This approach also enables our method to use the conventional and efficient FBP approach for image reconstruction, which is fast and already implemented in most of the CT systems available today. We delete the metal-contaminated projection data and train a conditional generative adversarial network (CGAN) to perform projection data completion. The resulting “enhanced” projection data is subsequently used with the FBP method to reconstruct the CT image. This new deep-learning-based MAR (*Deep-MAR*) approach results in an end-to-end framework that is computationally efficient and tractable, and which fits well into existing CT workflows, allowing easy adoption in existing scanners. In this initial work, we focus on 2-dimensional (planar) problems. A preliminary version of our learning-based sinogram completion approach was presented in [4] for sparse-view tomography.

A. Contributions

1) *Deep-MAR Framework*: The main contribution in this work is the new *Deep-MAR* framework that uses state of the art deep learning (DL) methods based on convolutional neural networks (CNNs) to complete projection data in a tractable and computationally efficient way.

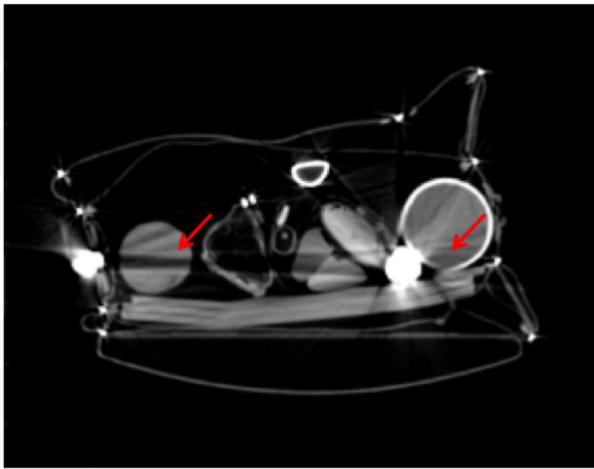


Fig. 1. FBP reconstruction of a slice from a dataset collected to model airport checked-baggage scenarios is presented [3]. Severe metal artifacts have caused object splitting and boundary suppression which will result in inaccurate object segmentation and labeling.

2) *CGAN-Based Sinogram Completion*: The second contribution of this work is the application of state-of-the-art conditional generalized adversarial networks (CGANs) for sinogram completion. CGANs have shown incredible power and generalizability in a variety of image processing problems and yet are relatively new to inverse problems in general and CT in particular. In order to make our work reproducible, simulated test examples, test codes, and trained models have been publicly made available.¹

3) *Network Training With Simulated Data and Transfer Learning*: The availability of adequate training data is a fundamental DL challenge. This challenge is especially acute for security related CT data, where the universe of possible scenes is very large and thus limits what can be obtained through direct physical experiment. The third contribution of this work is demonstrating the use of a large set of simulated data based on accurate X -ray physics coupled with transfer learning exploiting a more limited amount of real experimental data. We have made the simulation setup and simulated data prepared for this work publicly available.² This will motivate and help researchers to develop new machine-learning-based methods to improve CT image reconstruction and analysis.

4) *Pathway to Generating Real Training Datasets*: The fourth contribution of this work is the ability to generate real projection-domain training data pairs matched to our framework. By casting the learning problem as one of sinogram completion rather than direct correction we focus the deep network on the task of missing data generation. This focus means the network does not need to handle all the different possible types of metal that can appear in a sinogram, just the lower-dimensional geometric missing-data configurations. As a consequence, our network only needs matched pairs of sinograms with missing and corresponding complete (but non-metal) data for training.

This structure makes it possible to generate additional real-data physical examples to train the DL-based projection-domain MAR method from existing scene data without metal objects through simple sinogram data deletion, a much easier task.

B. Prior-Work

Existing MAR algorithms can be grouped into three categories: Model-based iterative reconstruction (MBIR-MAR), image-domain MAR (ID-MAR), and projection-domain data completion (PD-MAR). MBIR-MAR methods incorporate a physically derived observation model together with appropriate image-domain priors in an iteratively-solved optimization problem [3], [5]–[10]. MBIR methods can produce excellent results and are a principled way of incorporating prior information. Unfortunately, they are typically computationally expensive, requiring repeated reconstruction and forward projection in an iterative algorithm. This computational cost has unfortunately limited their practical impact to date, especially in security applications where throughput is important.

Image-domain MAR (ID-MAR) methods rely on image-processing techniques as a post-processing strategy to reduce streaks in the reconstructions. ID-MAR methods first reconstruct images which contain unwanted artifacts, and then attempt to correct these artifact-filled images. The problem with this strategy is that once important image structure is lost to artifacts, it can be very difficult to recover it effectively. In the medical context, Soltanian-Zadeh *et al.* [11] estimate streaks by subtracting a low-pass-filtered version of an image from itself and then thresholding the difference image. Streaks are removed by subtracting this estimate from the original image. This technique involves an ad-hoc thresholding step and does not attempt to recover structure lost in the streak areas. In the security context, Karimi *et al.* [12] estimated streaks by computing the difference of penalized least squares (PLS) and penalized weighted least squares (PWLS) reconstructions where metal-contaminated projections are down-weighted and the estimated streaks are later removed from the original reconstruction using subtraction. In the medical context, Gjesteby *et al.* [13] recently proposed coupling conventional normalized MAR (NMAR) with a convolutional neural network. Their method attempts to learn a mapping from the NMAR image to an artifact-free image, with the aim of eliminating the residual artifacts of the NMAR method. Training was done on phantom images with artificially inserted metal objects. The data used appeared limited to 40 slices of a single case, with 5 slices held out for testing. Extending the approach to incorporate more general real data would appear challenging, since obtaining matched image pairs with and without various metal materials would be difficult. Since the nature of the artifacts will change with the nature of the metal in the scene, it would seem that the training set would need to cover the range of metal materials anticipated in addition to the configurational variability, which would be difficult to achieve in the security context.

PD-MAR approaches aim to replace the metal-corrupted projection data with data obtained using neighboring information. These methods basically treat metal-corrupted projection data as

¹[Online]. Available: <https://codeocean.com/capsule/0510438>

²[Online]. Available: <https://github.com/mughanibu/DeepMAR/>

missing data and attempt to complete these projections. Kalender *et al.* [14] proposed a simple approach to replace the metal-corrupted data using the one-dimensional linear interpolation (LI) of neighboring detector channels. This approach is by far the most common MAR scheme and is considered a benchmark [15]. However, its performance deteriorates as the size and number of metal objects increases. In attempting to ameliorate the original metal streaking it can also introduce new streaking artifacts, as can be seen in the results in Section V. In an attempt to improve on the LI method, Mahnken *et al.* [16] proposed an extended, two-dimensional interpolation method that replaces the metal-corrupted projection data with a weighted sum of N -nearest neighbor data points. Rivière *et al.* [17] proposed a statistical objective function-based projection data restoration method that iteratively corrects for data and model mismatch. The method is not focused on data completion or metal artifacts and can be computationally expensive.

NMAR [18] is yet another PD-MAR algorithm that relies on the presence of a prior image similar to the one of interest and combines this prior with a simple interpolation scheme. Zhang *et al.* [19] recently proposed a similar approach, in which a prior image is estimated using a trained CNN. Gjestsby *et al.* [20] trained a CNN to learn mapping from NMAR sinogram to metal-free sinogram. These approaches rely heavily on the prior image and their performance degrades if an appropriately matched prior image is not available. Such prior image-based correction methods are particularly problematic for security scenarios because a good prior image is typically not available due to the high variability of objects and materials present in such scenes.

Claus *et al.* [21] recently proposed a learning-based PD-MAR approach. They trained a fully connected neural network to perform interpolation of missing data in the sinogram domain, and is thus similar in focus to the current work. Their method applies to a highly constrained problem with only a single metallic object of fixed and known size placed at the very center of the field of view. Since the network is trained for this very constrained situation, it is not clear how this method could be extended to realistic scenarios with multiple metal objects placed at arbitrary locations. In addition, its reliance on a fully connected network severely limits scaling to practical problem sizes. Overall, while there has been initial work aimed at applying learning in the CT data domain, these existing works have been limited to small, toy examples and/or highly constrained scenarios not representative of real applications.

Anirudh *et al.* [22] proposed an implicit sinogram completion method for limited angle CT application. In contrast to our approach, their method consists of a network that learns a mapping from incomplete projection data to a final complete-data image. They demonstrated that their approach produces better reconstructions and segmentations using incomplete data as compared to conventional limited angle CT methods.

Learning methods based on CNNs have had impact in a variety of applications. CNNs have demonstrated impressive performance on various image restoration tasks, including image super-resolution [23], image denoising [24], and artifact reduction [25]. However, training CNNs with a mean-square-error loss

function has been shown to result in over-smoothed images [26], [27]. Adversarial training is an elegant strategy to train neural networks [28] and Isola *et al.* [26] suggest that a traditional loss coupled with an adversarial loss is a better overall loss function for *image-to-image translation* tasks. They proposed an approach called a Conditional Generalized Adversarial Network (CGAN) for these tasks. A CGAN consists of two networks: a generator network that learns to perform *image-to-image translation*, and a discriminator network that is trained to discriminate artificial from true images. The discriminator network is used as an adversarial loss for training the generator network, which forces the generator to be better and better at its job in order to fool the discriminator. We use generator and discriminator network architectures that are inspired from Isola *et al.* [26]. Different CNNs have already been successfully applied to perform missing data replacement or completion in the image domain [26], [29]–[31]. However, using CNNs for sinogram completion amounts to learning *sinogram-to-sinogram translation*, which is a different task.

The success of CNNs depends upon successful learning of internal representations, which typically requires large datasets. This need poses a major challenge for the application of CNNs to many areas where access to large datasets is not possible due to privacy or security concerns or the need for highly skilled labor to provide ground truth annotations, e.g., medical imaging, and especially security applications. A recent trend in these areas is to use transfer learning [32]. In this work, we investigate the use of transfer learning in the security context by pre-training the deep network using a large *simulated* dataset and then fine-tuning the network on a modestly sized real dataset.

II. LEARNING-BASED MAR APPROACH

In this paper, we propose a framework we term *Deep-MAR* to reduce metal artifacts in CT images using adversarial deep learning to perform completion of missing projection data in the sinogram domain (i.e. sinogram completion). The *Deep-MAR* framework is motivated by and focused on reducing the effects of metal in checkpoint security imagery.

A. Deep-MAR Algorithm

The major steps of our *Deep-MAR* framework are presented in Figure 2. There are three major steps: i) identification and suppression of metal-contaminated projection data, ii) CGAN based sinogram completion, and iii) efficient FBP image reconstruction and reinsertion of metal objects back into the reconstructed image.

In order to identify the metal-contaminated projection data a conventional FBP reconstruction using the original sinogram is first generated. Thresholding followed by morphological operations are then used to segment the metallic objects in the image, which are forward projected to generate a mask, M , in the sinogram domain corresponding to the metal traces. This mask is made slightly larger than the original thresholding to be sure to remove all the metal. The metal-contaminated projection data is then deleted from the original sinogram. Similar metal segmentation methods have been used in many of the MAR

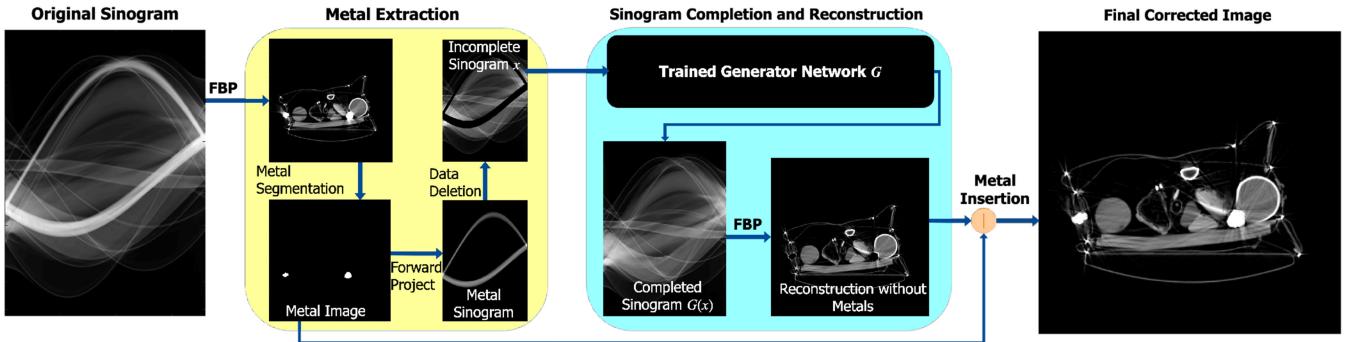


Fig. 2. Work flow and major components of our *Deep-MAR* framework are presented. We start with a metal-corrupted sinogram which is used to create an FBP reconstruction to identify the metal objects in the image and delete the corresponding metal-contaminated data in the sinogram. Incomplete sinogram is input to the generator network to complete the sinogram which is later used for FBP reconstruction.

approaches [12], [16], [18], [33]. One could potentially use more sophisticated metal segmentation approaches, for instance a DL-based approach for object segmentation [34].

B. Learning-Based Sinogram Projection Completion

The heart of the *Deep-MAR* framework is a state of the art CGAN, which is used to perform the sinogram completion task. A CGAN is composed of two distinct networks: a generator network G and a discriminator network D . The two networks are trained by optimizing an objective function based on a mini-max game [26], [28]. The overall mini-max objective function we use is given by:

$$G^* = \arg \min_G \max_D \mathcal{L}_{cGAN}(G, D) + \lambda \mathbb{E}_{x,y}[\|y - G(x)\|_2^2] \quad (1)$$

where, x and y represent input and ground truth pairs of incomplete-data and true complete-data sinograms, respectively; $G(x)$ is the estimated complete-data sinogram produced by the generator; $\mathbb{E}_{x,y}$ defines expectation over data density p_{data} approximated by training data pairs x and y , D denotes the discriminator network, and λ is a hyper-parameter used to control the balance between the two terms in the expression. The inclusion of a discriminator loss was seen to improve the overall result relative to an ℓ_2 only loss. In the Supplementary Material, we provide more detail on the value of the ℓ_2 and discriminator loss terms to the overall formulation.

In the optimization (1) the networks G and D act as adversaries – that is, G attempts to make perfect sinograms while D attempts to detect fakes. The first term in the objective captures this interaction through an adversarial loss between G and D . This loss is given by:

$$\begin{aligned} \mathcal{L}_{cGAN}(G, D) &= \mathbb{E}_{x,y}[\log D(x, y)] \\ &\quad + \mathbb{E}_x[\log(1 - D(x, G(x)))] \end{aligned} \quad (2)$$

were, \mathbb{E}_x defines expectation over incomplete data density, approximated by averaging over input training data samples x . The presence of this loss forces the D network to improve its discrimination ability and thus also forces the generator network G to become better and better at completing sinograms. In other image processing tasks, such an adversarial approach has been shown to produce more robust and higher performance generator

networks G . The second term in the optimization (1) is the traditional loss associated with fit to the training data. In this case we use the ℓ_2 loss because our initial experiments indicated that an ℓ_2 loss performs better for sinogram completion, as compared to choices such as the ℓ_1 loss.

The internal structure of the coupled generator and discriminator network architectures are presented in Figure 3. Convolutional kernels of size 5×5 with a stride of 2 are used at each layer for both the generator and discriminator networks. The generator network, G , has a modified U-Net architecture with a fully convolutional architecture. Instead of using max-pooling layers for down-sampling, two-pixel strided convolutions are used. For up-sampling, transposed convolutions with two-pixel strides are used. Overall 6 down-sampling and 6 up-sampling layers were chosen. Additionally, there are skip connections between each layer i and $n - i$, where n is the total number of layers. The skip connection at each layer merely concatenates outputs at layer i to layer $n - i$. Using stride-2 convolutions result in sub-sampling and significantly larger practical effective receptive field (ERF) as compared to stride-1 convolutions [35]. The theoretical ERF grows with the number of layers – 6 down-sampling and 6 up-sampling layers result in a theoretical ERF of 757×757 pixels [36], so each estimate could potentially use information from very large area of the input sinogram. Kernel size selection is also guided by the theoretical ERF in that using smaller kernels would require more layers to achieve a desired ERF and larger kernels would result in many more learning parameters needing estimation. Empirical testing showed that 5×5 convolutional kernels with a shallower architecture perform better than 3×3 convolutional kernels with a deeper architecture. Dropout is used in the last layer to avoid over-fitting. It is used after all batch-normalization (BN) layers as suggested by Li *et al.* [37] since dropout results in variance shift when applied before BN.

The discriminator, D , is based on the full sinogram image, instead of just patches. Given a ground truth or network output sinogram conditioned on the input incomplete sinogram, our discriminator network classifies the full sinogram as real or fake. The reason behind using a full-sinogram discriminator is that missing data follows sinusoidal structure, which would not be possible to accurately capture using a patch-based discriminator. Additionally, the network is trained to perform non-blind sinogram completion. The missing data are the metal-contaminated

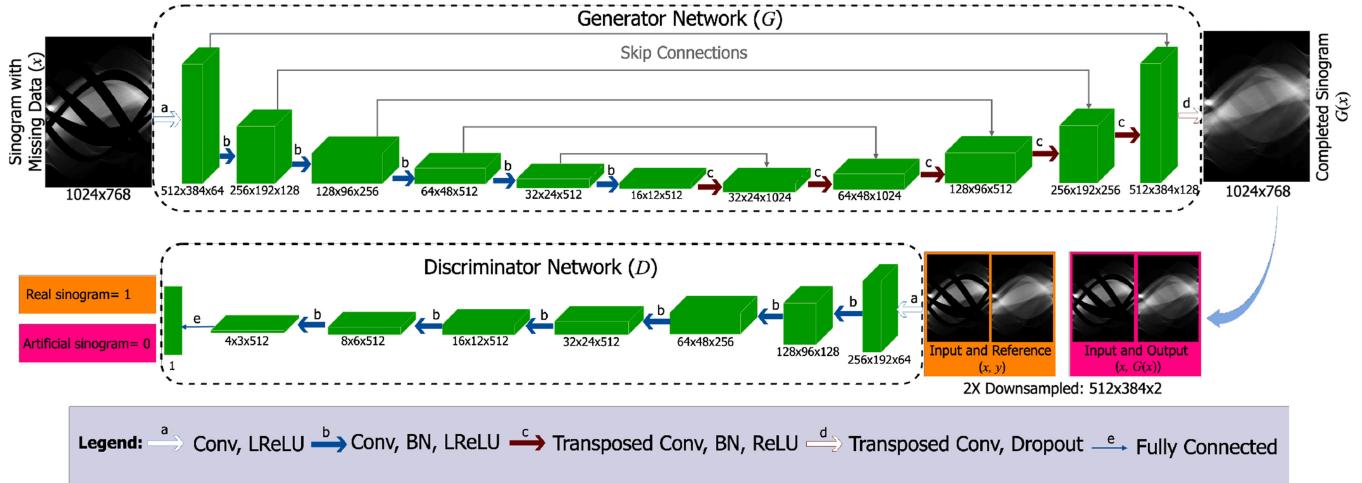


Fig. 3. Overall CGAN framework consisting of a generator and a discriminator network is presented. The generator network follows a *U*-Net-like architecture, where the input is an incomplete sinogram and it learns to complete the sinogram. The discriminator network learns to distinguish between ground truth and generator completed sinograms, and is used as an adversarial loss term to train the generator network so it can produce realistic output. Abbreviated legend at the bottom of the Figure are defined here, Conv: 2-D convolutions, Transposed Conv: transposed convolutions, BN: batch-Normalization, LReLU: leaky rectified linear units, ReLU: rectified linear units.

data points in sinogram, which are masked before the sinogram is input to the generator network. By using mask-specific sinogram completion our loss function is focused on the areas of most interest – where the data is missing. The generator network output is given by: $G(x) = x + M \odot x_{D1}$, where, x_{D1} is the output of the last layer of the generator network, M is the metal mask, and \odot denotes element-wise multiplication. This choice retains non-metal sinogram data as is, and uses the deep network to replace only the metal traces. We use $2X$ downsampled sinograms for discriminator (D) to speedup the training procedure.

III. NETWORK TRAINING

A major challenge in the use of deep network architectures is having sufficient data to allow robust training of the network. In many computer vision and image processing tasks there is access to abundant samples of training data (e.g. ImageNet [38], which contains over 14 million images). In contrast, in the security domain there is a great diversity of possible objects in a scene, yet the amount of physical CT data available for training is severely limited. To address this limitation we generated a training set for sinogram completion using physically accurate X -ray simulation tools. After initial training, our networks are then refined with available physical data through transfer learning. Areas without metal are considered background (air) in this work – that is, the target replacement value for metal regions is background material and, conversely, only locations with background material are considered for virtually embedding metals in real-data. This approach makes the problem more tractable since the network can focus on a smaller space of possibilities.

A. Simulated Training Data Generation

Most conventional CT scanners use a poly-energetic X -ray source coupled with energy integrating detectors. We can

accurately model the data obtained from such scanners as a sum of mono-energetic sinograms weighted by the corresponding relative strength of the source spectrum. The resulting projection output can thus be well modeled as:

$$\begin{aligned} \tilde{I}^{(i)} &= I_0 \cdot \eta(E^{(i)}) \\ I_j^{(i)} &= \tilde{I}^{(i)} \cdot e^{-\int_{L_j} \mu(\vec{x}, E^{(i)}) d\ell} \end{aligned} \quad (3)$$

where $\tilde{I}^{(i)}$ is the relative X -ray source intensity at energy $E^{(i)}$, obtained from the energy-dependent scanner source weighting function $\eta(E^{(i)})$ at energy E_i and scaled by the overall X -ray source intensity or blank scan factor, I_0 . The sinogram contribution at energy $E^{(i)}$ for ray-path j corresponding to the line L_j is denoted by $I_j^{(i)}$ and is obtained from the Beer-Lambert law using the energy-dependent scene attenuation coefficient $\mu(\vec{x}, E^{(i)})$ at location \vec{x} and energy $E^{(i)}$ integrated over the associated ray path. Note that the exponent in the Beer-Lambert expression is a line integral projection, and is approximated by discretizing the integral using standard methods. In this work we use a common ray-based projector model [9], though other models are certainly possible (e.g. distance driven projection, Fourier methods, etc.).

The projection data contributions are degraded by data-dependent and electronic noise. The data-dependent variability follows a Poisson distribution with mean corresponding to $I_j^{(i)}$. The electronic noise is modeled as Gaussian zero-mean and variance σ_e^2 . Since the standard CT data are log-normalized, the final X -ray observation model for ray-path j is defined as:

$$y_j = -\ln \left(\frac{\sum_{i=1}^N \left(g^{(i)} \text{Poisson}(I_j^{(i)}) \right) + \mathcal{N}(0, \sigma_e^2)}{\sum_{i=1}^N g^{(i)} \tilde{I}^{(i)}} \right) \quad (4)$$

where the sum is over the contributions from each of the N source energies used to approximate the continuous spectrum. The term $g^{(i)}$ is the energy-dependent detector response or gain

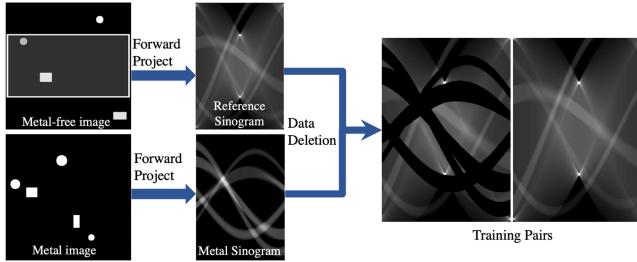


Fig. 4. Setup for simulated training pair generation. A metal-free scene is created using our stochastic bag simulator which is forward projected to compute a reference sinogram using our physically accurate X -ray scanner model. Our stochastic bag simulator then defines geometrical configurations for metallic objects in the scene to generate masks for data deletion in the sinogram domain. These masks are then used to delete data resulting in an incomplete sinogram. A large set of training pairs consisting of incomplete and reference sinograms are created this way to train our CGAN model.

which captures the photon to charge conversion factor, therefore, y_j is measured in units of current. This model is consistent with poly-energetic models used in different published studies [3], [5], [17], [39]. By incorporating a full energy aware ray model, effects such as beam hardening are inherently included. Note that one could consider other sensing models and configurations in a straightforward way by instantiating appropriate simulation models, including, for example, photon-counting detectors, Monte-Carlo-based photon scatter, etc.

Coupled with this physically accurate X -ray scanner model, we also created a stochastic bag simulator. This bag or scene simulator places objects of different material composition and of varying geometric configuration at random locations in the scene. The current instantiation of the simulator uses geometric primitives including circles and rectangles and uses material attenuation coefficients drawn from the NIST XCOM database [40]. The setup for simulated training dataset generation is illustrated in Figure 4. A metal-free scene is created and then multiple metallic object areas are defined with corresponding sinogram data deletions to create multiple training pairs. These training pairs consist of sinograms with data deletions corresponding to metal object locations (incomplete sinograms) and the corresponding sinograms without the metal objects (reference completed sinograms). The entire process can be done quickly allowing the generation of tens of thousands of sinogram pairs, much more than would be possible to do physically. Other, more complex, scene generators could also be used for this purpose.

B. Real Training Data Generation

Acquiring physical matched sinogram pairs with and without metal objects for training is a challenge, especially in security settings. What would be required in general is a real sinogram containing metal objects and the corresponding real sinogram if those objects were absent, but everything else in the scene remained the same. Such consistency is very difficult to achieve in practice.

As a consequence of formulating the PD-MAR problem as one of sinogram completion, our approach only requires metal-trace

masked and corresponding metal-free complete sinograms for CGAN training since we are focusing our learning on missing data completion. We can use sinogram data from scenes which do not contain metal objects, and then create matched sinograms by deleting data in the sinogram that corresponds to geometric configurations of embedded objects. We define these object configurations at locations of the image associated with voids or background. In this way we can manually generate a modest a number of real pairs of incomplete and complete sinograms to train our CGAN. The security scans used for this data generation were obtained from the ALERT Center of Excellence (<http://www.northeastern.edu/alert/transitioning-technology/alert-datasets/>)

C. Real Data Transfer Learning

We combine our two sources of training data (that generated by accurate X -ray physics simulation and that derived from real scans) through transfer learning from the larger simulated data set. Transfer learning involves transfer of knowledge from a learned function, $f_S(\cdot)$, that solves a task \mathcal{T}_S in a source domain \mathcal{D}_S to achieve improved performance in learning a function, $f_T(\cdot)$, to perform a task \mathcal{T}_T in a target domain \mathcal{D}_T [32], [41]. The most common use of transfer learning in DL research involves learning a function, $f_S(\cdot)$, in the form of a deep network from a large set of training data and then copying the first k layers of this network $f_S(\cdot)$ to a target network $f_T(\cdot)$ as an initialization. In general, $\mathcal{D}_S \neq \mathcal{D}_T$ and the target task \mathcal{T}_T may or may not be the same as the source task \mathcal{T}_S [32]. The initial network then undergoes additional refinement or training using a smaller set of training data from the new task. Transfer learning has been shown to provide improved performance, generalization, and robustness to over-fitting as compared to random initializations combined with limited training for various image classification tasks even when \mathcal{D}_S and \mathcal{D}_T are very different from each other [32], [42]–[44].

In this work transfer learning from a larger set of simulated data to a smaller set of real data is used. The learning task (sinogram completion) remains same in our case, i.e., $\mathcal{T}_S = \mathcal{T}_T$, however, the nature of the training data is changed, i.e., $\mathcal{D}_S \neq \mathcal{D}_T$. Additionally, we use the same network architecture for both source and target tasks, and we copy all layers of $f_S(\cdot)$ to $f_T(\cdot)$. We first train the network on a large simulated dataset acquired by the physically accurate simulation setup previously described. This trained network, $f_S(\cdot)$, is later fine-tuned using a smaller amount of real data. This strategy allows us to achieve good performance on real data even with a small real dataset.

IV. EXPERIMENTS

This section describes the details of our experiments, networks, and training. For simplicity, we restrict our experiments to 2-D parallel-beam geometry in this work. Training deep networks is a challenging and time-intensive task, but once trained, inference is very fast. Note that in the examples considered here our proposed Deep-MAR method is very efficient; our generator network G takes approximately 58 milliseconds to perform sinogram completion on an NVIDIA Tesla P100 GPU

at test time, which makes it attractive in real world contexts. We use a CPU implementation of FBP which has not been optimized for efficiency, and on average it takes 1.55 seconds using an Intel Xeon (2.4 GHz) processor on a Linux system with MATLAB R2017b to reconstruct the 2-D images considered in this work.

A. Metal Artifact Dataset

1) *Simulated Dataset:* Our simulated data model is based on the Imatron C300 scanner. Equation (4) was used to simulate poly-energetic CT with a weighting function $\eta(E^{(i)})$ and detector response $g^{(i)} = 2.6 \times 10^{-3} \cdot E^{(i)}$ pA/quanta estimated by Crawford *et al.* [3], source intensity set to $I_0 = 1.7 \times 10^5$ photons per ray, and using $N = 121$ uniformly sampled energy levels between 10 KeV and 130 KeV. The electronic noise variance $\sigma_e^2 = 3.37 \text{ pA}^2$ is modeled as counting statistics of 20 photons detected at 65 KeV following [3]. Photon scatter could certainly be included in our simulation setup in a straight forward way. However, accurate Monte Carlo scatter simulation methods are computationally very expensive ($O(n^6)$) [45]. This additional computational load would limit the amount of simulated training data we could generate. We decided a simpler ray-based simulation was sufficient for sinogram completion training that was required and therefore, in this initial work, we do not include scatter in our simulation setup. Attenuation images of size 475 mm \times 475 mm were generated with attenuation coefficients from the NIST XCOM database [40]. The list of metallic and non-metallic materials used is presented in the Supplementary Material (Section S1). Sinograms at each of the $N = 121$ energy levels were generated using 720 uniformly sampled projection angles between 0° and 180°, and 1024 detector channels per projection angle. In order to match the network input and output size, zero-padding was performed to result in a sinogram of dimensions 1024 \times 768. We generated 10,000 example images using the stochastic bag simulator with non-metallic objects, and placed up to 5 metallic objects in each instance resulting in a training dataset of 50,000 sinograms. An additional 25 suitcase scenes with different configurations of cylinder and sheet objects (similar to the ones in the ALERT Task Order 3 dataset [3]) were manually generated and included in the training dataset. For each of these images, matched pairs of sinograms without and with metallic objects in the scene were generated. We used horizontal flips of this data as a data augmentation strategy. The ASTRA toolbox [46] was used for accelerated forward projection in the training data preparation. We used our stochastic bag simulator to generate 102 reference metal-free scenes and inserted up to 5 metals to result in 510 test dataset sinogram pairs which were used for qualitative and quantitative analysis of our approach on simulated dataset.

2) *Real Training Dataset:* Additional real training data was generated from real sinogram data acquired by an Imatron C300 scanner. The data was part of a collection effort supported by DHS [47]. The scans were performed with a field of view (FOV) of 475 mm \times 475 mm with a peak X-ray source energy level of 130 KeV. Data was re-binned to a parallel geometry with 720 projection angles and 1024 detector channels. Further, to match the network input and output size, zero-padding was done

to result in a sinogram of dimensions 1024 \times 768. In order to create the pairs of sinograms for training of the CGAN, 1,706 different slices which did not contain significant metal objects were manually identified from the scanned bags. The strategy described in Section III-B was used to generate 8,530 pairs of real data sinograms for training with sinogram missing data corresponding to up to 5 virtual metallic objects. Horizontal flips of this data were also used as a data augmentation strategy. This real data was used for transfer learning as previously described.

3) *Real Testing Dataset:* For quantitative analysis of performance on real data, 159 pairs of sinograms were generated by selecting metal-free images from the ALERT Task Order 4 dataset [47] using the strategy described in Section III-B. This strategy was used for quantitative analysis since the reference artifact-free images without any metal objects are available in this settings. For qualitative analysis on real data, slices were selected from the ALERT Task Order 3 dataset [3] which contained metals and caused severe artifacts in the reconstructions. FBP reconstructions were generated for selected slices, and a metal mask identified by applying a threshold of 4000 Modified Hounsfield Units (MHU = $1000 + 1000(\mu - \mu_{\text{water}})/\mu_{\text{water}}$, where $\mu_{\text{water}} = 0.202527 \text{ cm}^{-1}$) [3], [12]. Thresholded mask results were eroded and then dilated with a disk-shaped structuring element of radius 2 and 4 pixels respectively to obtain a final metal mask. Erosion was used to remove very small objects, and a small dilation was used to over-segment the metals. Selected sinograms were input to our *Deep-MAR* framework and final reconstructions were used during testing for qualitative analysis of our proposed method.

B. CGAN Training

To optimize our CGAN sinogram completion network, the original GAN training strategy [28] was followed, where alternations between one gradient descent step on G and one step on D were performed, with the exception that for the first 4 epochs $k = 1, \dots, 4, 6 - k$ iterations on D were done for each gradient descent iteration on G . Mini-batch stochastic gradient descent with batch size 6 was used for both simulated data and real data transfer learning. The standard Adam optimizer [48] was used with learning rate of 0.0002 and momentum parameters $\beta_1 = 0.9$, $\beta = 0.999$. The value of the hyper parameter $\lambda = 10$ was decided empirically. For simulated data, the CGAN was trained for 25 epochs on 50,000 examples using the training scheme just described. Real data was used for transfer learning from the simulated data trained network with just 8,530 pairs of real sinograms. We trained our model on the real dataset for 50 epochs. In order to show effectiveness of transfer learning, we also trained a model without pre-training on the simulated dataset, i.e., training the model from scratch on the real dataset. At test time, the trained generator network G was used to perform sinogram completion. The network was implemented in Tensorflow [49] heavily borrowing code from the Tensorflow implementation of CGAN.³

³[Online]. Available: <http://github.com/yenchenlin/pix2pix-tensorflow>

V. RESULTS AND DISCUSSION

In this section *Deep-MAR* results are shown and its performance is compared to similar computationally light sinogram completion approaches suitable for use in practical settings. Comparison is made to LI-MAR [14], and WNN-MAR [16] on both simulated and real data. For a fair comparison, we set the weight of the original data $\mu_{\text{WNN}} = 0$ for WNN-MAR, and use default values for the rest of the parameters suggested in [16]. The performance of our trained CGAN is also analyzed using attention maps and latent space analysis. The performance of our generator network architecture is compared to a popular CNN architecture VDSR [23] in Supplementary Material. This comparison demonstrates the effectiveness of U-net like architectures as generator network. Furthermore, we present additional experimental results in the Supplementary Material.

A. Sinogram Completion Experiments

Since a key element of our proposed approach is sinogram completion we first focus on the ability of *Deep-MAR* to successfully perform this task. Qualitative sinogram completion results on simulated data are shown in Figure 5. Each row corresponds to a different sinogram completion case. The first row shows the original uncorrected sinograms, the second row shows the results obtained using LI-MAR [14], the third row shows the results obtained using WNN-MAR [16], and the fourth row shows the results obtained using the proposed learning-based completion approach *Deep-MAR*. Column (a) in Figure 5 corresponds to a simulated example, while column (b) and (c) correspond to two different real data examples obtained from the ALERT TO3 dataset [3]. The reference metal-free sinogram for the simulated data example in column (a) is presented in Supplementary Material. The images corresponding to these cases are shown in Figures 6 and 7. LI-MAR and WNN-MAR use limited neighboring data in 1-D and 2-D respectively coupled with spatially invariant weights to produce sinogram completion results. While this limited information is adequate in simple cases with few, small metallic objects, these methods struggle in more realistic security scenarios containing larger areas of missing data and complex configurations. As highlighted by the red arrows, the proposed *Deep-MAR* approach produces sinograms with fewer visual artifacts, such as discontinuities at metal trace edges compared to methods widely used in practice. Additional sinogram completion results presented in Supplementary material show that the proposed *Deep-MAR* based completion results are closer to the ground truth sinograms compared to LI-MAR and WNN-MAR completed sinograms.

To perform a quantitative analysis, we use 510 test examples from the simulated dataset and 159 examples from the real dataset prepared for quantitative analysis. We compare sinogram completion results to reference metal-free sinograms using average mean square error (MSE) and present results in Table I. Our sinogram completion method greatly reduces the average MSE as compared to LI-MAR and WNN-MAR: 89% and 87% average relative MSE reduction in simulated data, and 80% and 75% average relative MSE reduction in real data experiments respectively. We also compare performance of our

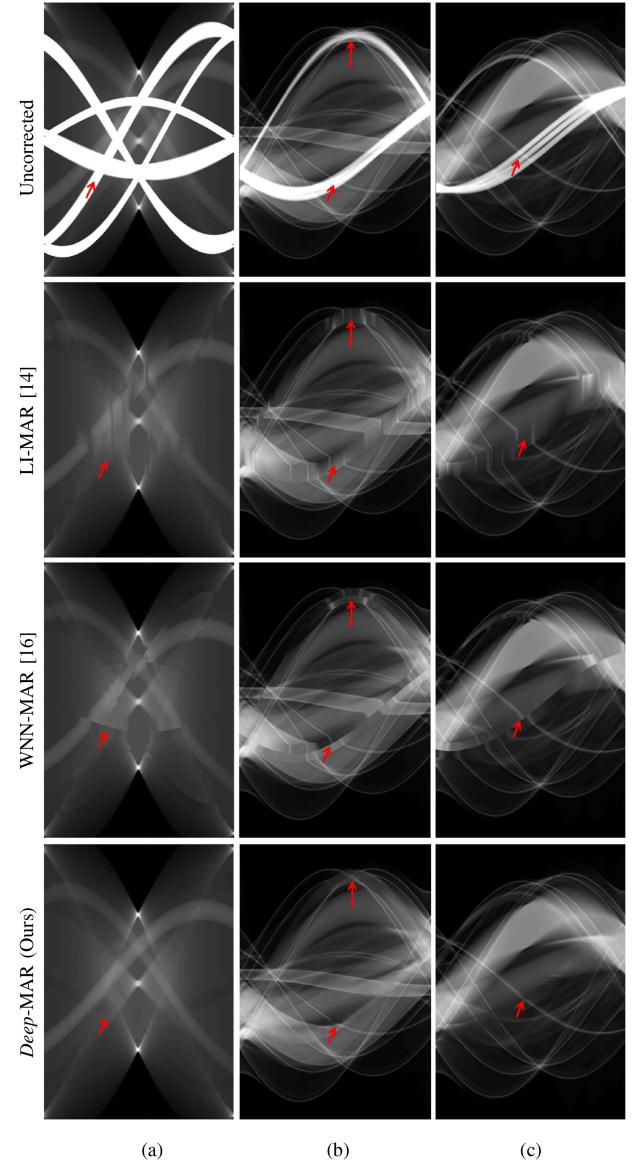


Fig. 5. Sinogram completion results for (a) simulated and (b), (c) real data examples using different methods are presented on display window [0, 5] and [0, 7] respectively. First row shows uncorrected sinograms, second row shows the results of LI-MAR, third row shows the results of WNN-MAR, and fourth row shows the results of the proposed *Deep-MAR* approach, showing its ability to accurately complete the sinograms. Red arrows point at some of the regions in sinograms where other methods visibly struggle.

models trained from scratch and using transfer learning on the real dataset using sinogram completion results. The advantage of transfer learning is evident through a significant improvement of 31% in terms of average relative MSE as compared to the model trained from scratch.

B. Simulated Data Reconstruction Experiments

In this section MAR reconstruction results on simulated data are presented. We use oracle information to segment metals in the simulated data experiments. In Figure 6 a qualitative example is shown corresponding to the sinogram case in Figure 5(a).

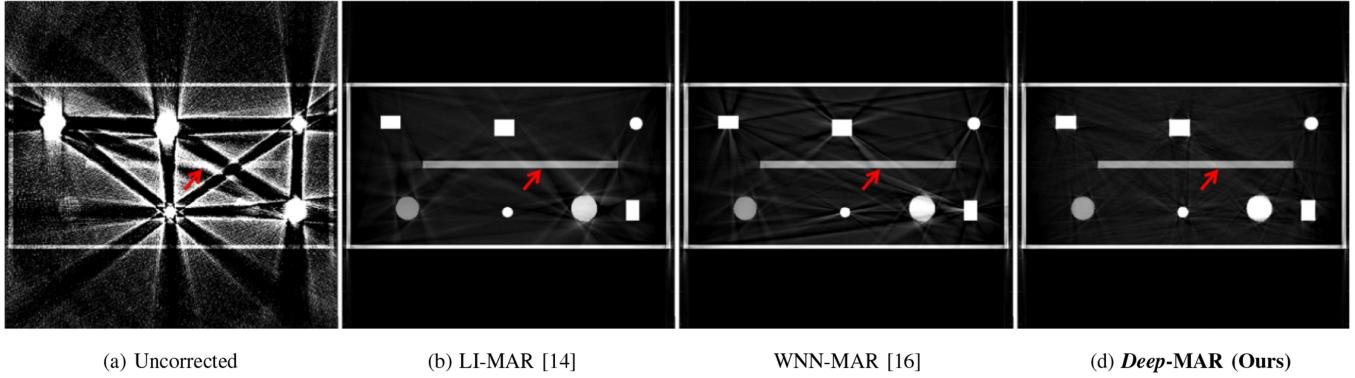


Fig. 6. MAR reconstruction results for the simulated case of Figure 5 presented on display window $[0, 0.15]$ cm^{-1} . Column (a) shows results without MAR, column (b) shows the results of LI-MAR, column (c) shows the results of WNN-MAR, and column (d) shows the results of the proposed *Deep-MAR* approach, showing its ability to reduce artifacts.

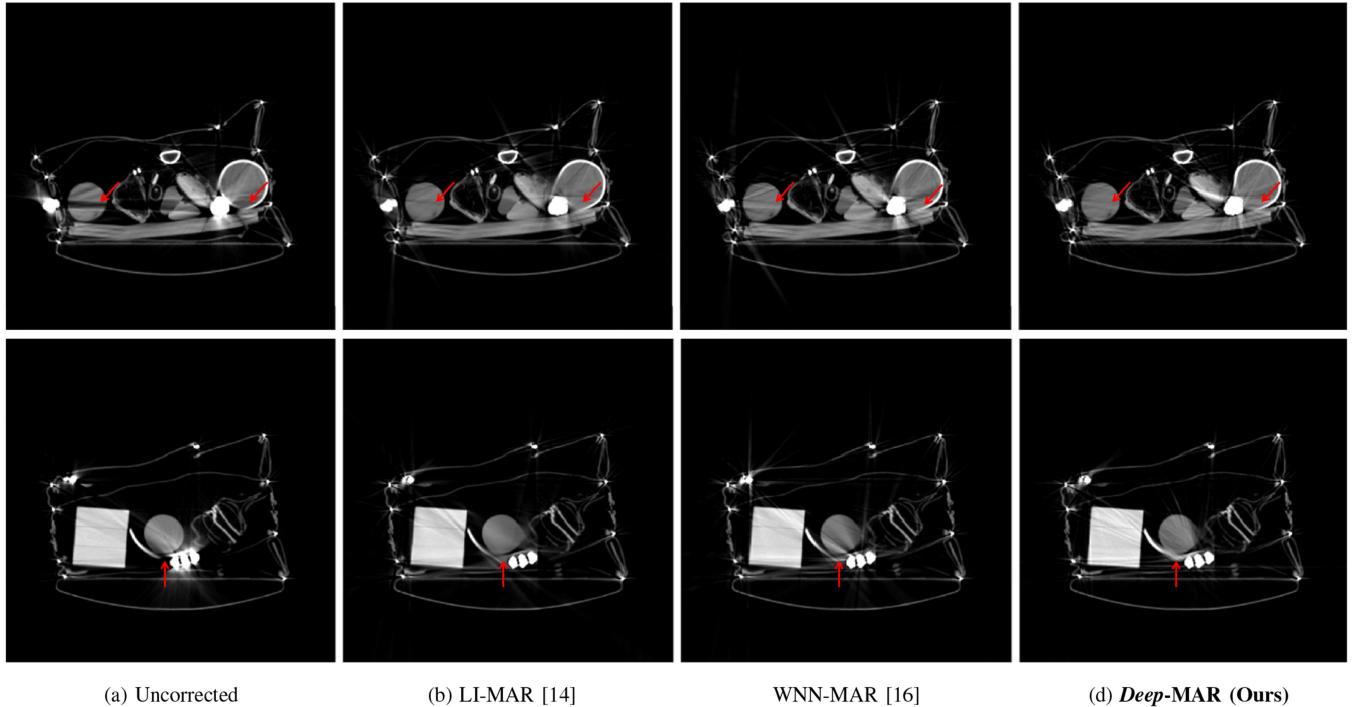


Fig. 7. Reconstruction results for two slices of real data with and without MAR techniques are presented on display window $[0, 0.4]$ cm^{-1} . Example slices represent a variety in number, and type of materials, and varied level of clutter in the field of view. *Deep-MAR* can reliably recover structure and suppress metal artifacts in these challenging scenarios.

TABLE I
SINOGRAM COMPLETION PERFORMANCE COMPARISON IN
TERMS OF AVERAGE MSE

	LI-MAR	WNN-MAR	<i>Deep-MAR</i> (Training from scratch)	<i>Deep-MAR</i> (Transfer Learning)
Simulated	0.1029	0.0872	-	0.0117
Real	0.0216	0.0173	0.0062	0.0043

This example contains the equivalent of so called bulk and sheet objects which are present in security applications. Column (a) presents results with standard FBP without using any MAR

techniques. Other results are produced by applying FBP after performing sinogram completion using the associated sinogram completion method and then inserting the metal objects back into the corresponding reconstructions. Column (b) shows the results of LI-MAR, column (c) shows the results of WNN-MAR, and column (d) shows the results of our new *Deep-MAR* approach. The reference metal-free reconstruction for this example is presented in Supplementary Material. The conventional FBP image naturally exhibits severe streaking artifacts which significantly disrupt both the bulk and sheet objects and which would be undesirable in a security context. While the LI-MAR and WNN-MAR methods have greatly reduced the artifacts, as they are intended

TABLE II
RECONSTRUCTION PERFORMANCE COMPARISON OF DIFFERENT METHODS ON SIMULATED DATA IN TERMS OF AVERAGE MSE, SSIM, AND PSNR IN cm^{-1} UNITS

	Uncorrected	LI-MAR	WNN-MAR	<i>Deep</i> -MAR
MSE	3.9×10^{-2}	4.7×10^{-4}	5.5×10^{-4}	1.2×10^{-4}
SSIM	0.7205	0.9750	0.9713	0.9941
PSNR	33.72	54.67	53.81	60.32

to, these methods still exhibit significant distortions which could affect segmentations of the sheet object and density estimation of the bulk objects. The proposed *Deep*-MAR method successfully suppresses nearly all of the streaking artifacts and successfully preserves the sheet object. Additional reconstruction results for simulated data are presented in the Supplementary Material and show the potential of the *Deep*-MAR method.

In addition to qualitative comparison, quantitative analysis of the reconstruction results using average MSE, average structural similarity (SSIM), and average peak signal to noise ratio (PSNR) of 510 simulated test images is presented in Table II. The proposed *Deep*-MAR method results in significant image quality improvement which is quantified by a 74% and 78% decrease in average MSE as compared to LI-MAR and WNN-MAR methods, respectively. Improvements can be noticed in SSIM and PSNR metrics results as well. Thus the *Deep*-MAR approach not only performs well in qualitatively suppressing metal artifacts, but is also good at correcting the underlying attenuation values, and correct attenuation numbers are important for the successful functioning of security algorithms.

C. Real Data Experiments

In this section end-to-end MAR reconstruction results on real data are presented. First qualitative results for two challenging real data examples are presented in Figure 7. These reconstructions correspond to the sinograms in Figure 5(b) and (c). Column (a) presents uncorrected results with standard FBP without using any MAR techniques. Results using LI-MAR, WNN-MAR, and our *Deep*-MAR are presented in Columns (b), (c), and (d) respectively. Red arrows show regions of particular interest. These include large metal objects in proximity to thin sheets or the boundaries of containers (bulk items), as well as interior regions of objects that should be uniform in intensity. In the conventional FBP image it can be seen that the presence of metal causes streaking through the middle of objects which splits them, shading (intensity variation) of what should be homogeneous regions, and the corruption of container boundaries important for segmentation. While conventional MAR methods (LI-MAR and WNN-MAR) help to some extent with streaking, there still exists significant loss of boundary information. Additionally the existing methods produce new streaking artifacts in some cases. In contrast, the *Deep*-MAR method appears to do a remarkable job on this very difficult task of preserving boundary information and object uniformity. The ability to preserve such object information greatly aids subsequent object segmentation and material recognition tasks. Additional reconstruction results for a variety of real data scenes are presented in the Supplementary Material.

TABLE III
RECONSTRUCTION PERFORMANCE COMPARISON FOR DIFFERENT METHODS ON REAL DATA IN TERMS OF AVERAGE MSE, SSIM, AND PSNR IN cm^{-1} UNITS

	LI-MAR	WNN-MAR	<i>Deep</i> -MAR (Training from scratch)	<i>Deep</i> -MAR (Transfer Learning)
MSE	1.4×10^{-4}	1.8×10^{-4}	7.5×10^{-5}	5.8×10^{-5}
SSIM	0.8158	0.8029	0.8605	0.8840
PSNR	30.49	29.61	32.68	33.91

In addition to qualitative results, quantitative analysis of real data cases was performed on a set of 159 real data examples (as described in Section IV-A3). These results are presented in the Table III. This quantitative analysis reveals that our proposed *Deep*-MAR method results in 59% and 68% reduction of average MSE as compared to LI-MAR and WNN-MAR methods, respectively. Improvements in average SSIM and PSNR metrics are also significant. We also compare performance of our models trained from scratch and using transfer learning on the real dataset using reconstructed images. The advantage of transfer learning is evident through a significant improvement of 23% in terms of average relative MSE as compared to the model trained from scratch.

Artificial generation of objects can be a concern when using machine learning algorithms in general imaging applications. We have not observed any such artificial objects generated by our CGAN approach, though imperfections at different stages of our method can and do lead to artifacts such as streaking, boundary drop out, and intensity shading. These artifacts are similar to what is seen with other completion-based MAR methods, though we believe the enhanced power of the CGAN approach reduces them.

D. Attention Maps

To gain an enhanced understanding of the behavior of our deep network approach we extend the occlusion sensitivity analysis work of Zeiler and Fergus [50] to *image-to-image* translation tasks. The work in [50] focused on image classification tasks and aimed to compute attention maps by occluding different regions of the input image and examining the resulting impact on performance of the deep network on the corresponding image classification task. The idea is that occlusion of more important image areas to the network will correspondingly have greater resulting performance impact. In this work, we apply this idea by repeatedly replacing an 11×11 patch of sinogram values with random data sampled from a uniform distribution $U(0, m)$ (which is thus uninformative) and then examining the resulting sinogram completion performance. Here, m is the maximum value in the sinogram under study. We use overlapping patches with a sliding distance of 6 pixels. We then compute the corresponding MSE for the sinogram regions completed by learned network and generate a 2-D attention map by associating this MSE value with the center of the corresponding noise patch. For better scaling, we display $\log_{10}(\cdot)$ of the MSE. Larger values in this resulting map should correspond to regions of the sinogram with more important influence on the solution generated by the network.

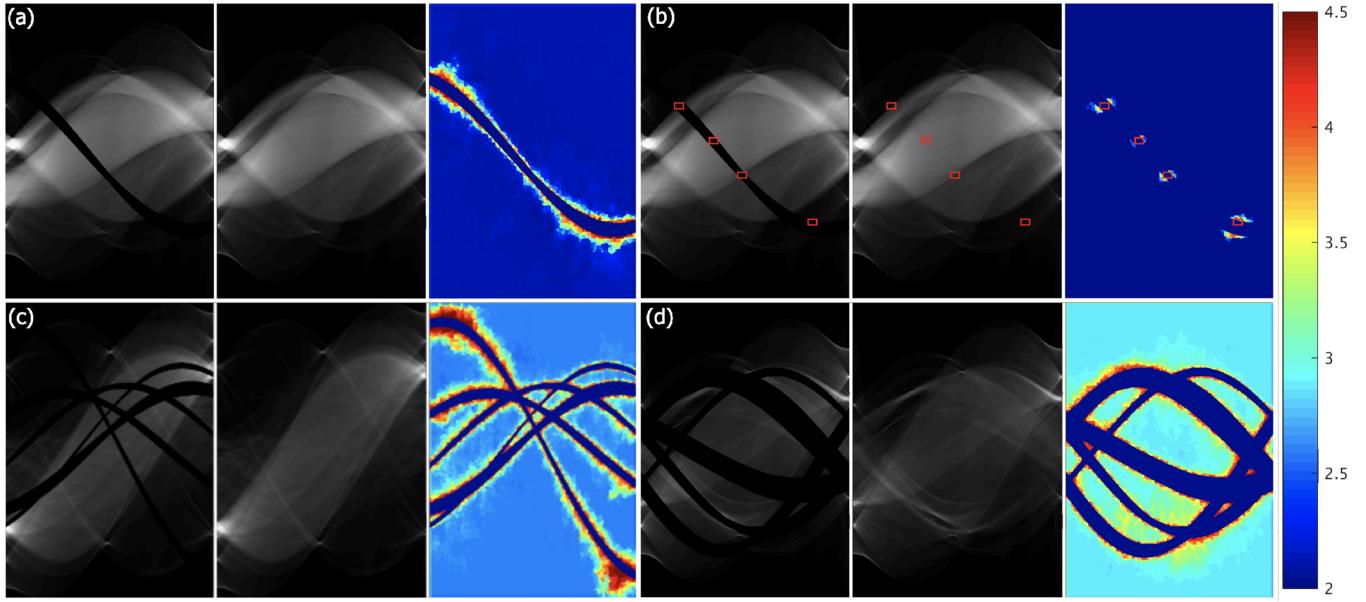


Fig. 8. Attention maps for an easy case, two challenging cases, and four ROIs in the easy case are presented. (a) Attention map of an example with one metal object, (b) attention map focused on four ROIs for the same example shown in red, (c), (d) attention maps for two examples with 5 metal objects. In all cases the missing data sinogram is shown in the left panel, the completed sinogram is show in the middle panel, the attention map is shown in the right panel. It can be seen from the maps that small sinogram gaps seem connected to small decision support regions while challenging scenarios as presented in (d) needs decision support from a larger portion of the sinogram.

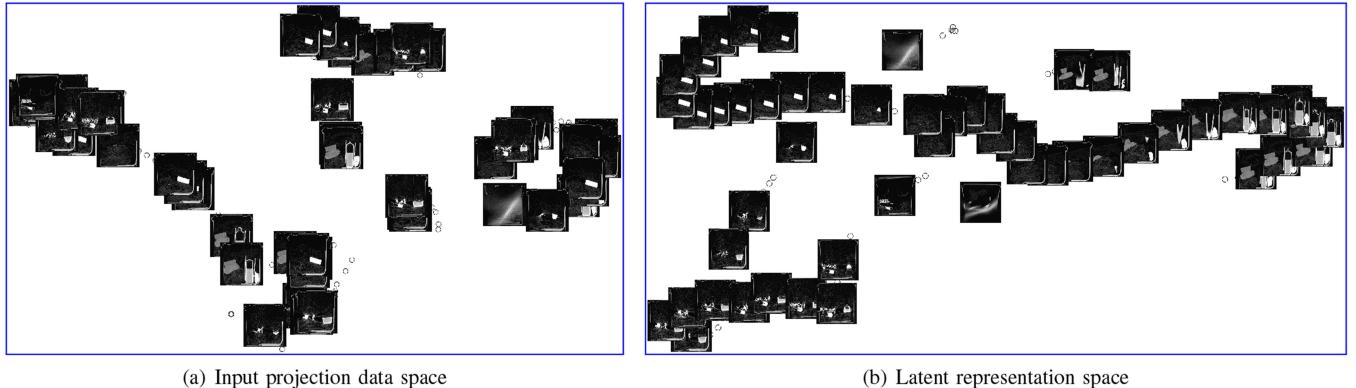


Fig. 9. A 2-D t-SNE visualization of (a) input sinogram data and (b) corresponding latent representation learned by CGAN encoder are presented alongwith the corresponding reconstruction images. The t-SNE visualization of the input sinogram does not exhibit any structure, i.e., similar image slices are not assigned similar t-SNE representations. However, the t-SNE visualization of our CGAN encoder's learned latent representation shows that similar image slices are assigned similar representations. This exhibited structure suggests that our CGAN encoder has successfully learned the semantic information in the sinograms.

Attention maps generated using this strategy are presented in Figure 8. These attention maps provide us a way to understand which areas of the sinogram the generator network considers most important to the result. Figure 8(a) presents the attention map for an example with one metal object. Figures 8(c) and (d) present attention maps for two challenging examples with 5 metal objects. Figure 8(b) presents attention maps for selected regions of interest (ROIs) shown in red. Figures 8(a) and (c) show that the pixels closest to the missing data appear to be the most important in generating an estimate, which seems logical. They also show that the size of the region of major influence also appears to grow with the width of the region of missing data.

Figure 8(d) shows that for challenging problems with significant missing data, more and more of the sinogram is used to generate an estimate. This “adaptivity” is in contrast to existing LI-MAR, and WNN-MAR methods where the region of influence is fixed and unchanged.

E. Latent Space Analysis

The ability to learn latent structure could be argued to be one of the reasons behind the success of deep networks. The generator network that performs sinogram completion in our CGAN framework follows an encoder-decoder architecture, where the

encoder maps high-dimensional input data to a low-dimensional latent representation, which is then used by the decoder to create the final image. One question is whether the resulting learned latent space representation is semantically meaningful – that is, whether it is actually capturing underlying relationships between different scene configurations. To investigate this question we applied the popular t-SNE [51] dimensionality reduction technique to visualize the high-dimensional input and corresponding learned latent space representations of various examples. We used sinogram data from all the 2-D slices of a suitcase and applied the t-SNE approach to the input sinogram data as well as the encoder produced latent sinogram representation to compare the representational information in these two domains. A 2-D presentation was created wherein each point in the input and latent space was tagged with its corresponding reconstructed image. Of course, similar images will have similar corresponding sinograms. The results are presented in Figure 9. The images associated with the input sinogram data do not appear to exhibit significant clustered structure. It does not appear that similar images/slices are grouped together in the original space. In contrast, similar image slices *do* appear to group naturally together in the latent space representation case. This result suggests that our CGAN encoder is able to learn a semantically meaningful latent representation from the sinogram projection data.

VI. CONCLUSION

Metal artifacts pose serious challenges to interpretability of CT reconstructions. While many sinogram completion based methods and image post-processing methods have been proposed, the problem is still very challenging and the subject of active research. In this paper, we present *Deep-MAR*, a new DL-based framework for data domain MAR, that uses a CGAN for data domain sinogram completion. Additionally, we demonstrate the successful use of transfer learning from simulated training data to a small set of real data. Excellent performance is achieved on challenging real security examples and the method is computationally light enough to be used in practical situations. We evaluate the performance of our method on simulated and real datasets and compare the results to two popular MAR methods: LI-MAR, and WNN-MAR. Qualitative and quantitative analysis of reconstruction results show the potential of our approach over conventional methods. In particular, the proposed method robustly suppresses metal artifacts and retains significant structure and reconstructed density numbers. Although, we consider a security scenario in this paper, our technique is generic in that it can be trained and applied to reduce metal artifacts in other applications as well, such as medical imaging.

ACKNOWLEDGMENT

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security.

REFERENCES

- [1] F. E. Boas and D. Fleischmann, "Evaluation of two iterative techniques for reducing metal artifacts in computed tomography," *Radiology*, vol. 259, no. 3, pp. 894–902, 2011.
- [2] L. Martin, A. Tuysuzoglu, W. C. Karl, and P. Ishwar, "Learning-based object identification and segmentation using dual-energy CT images for security," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4069–4081, Nov. 2015.
- [3] C. Crawford, H. Martz, and W. C. Karl, "Research and development of reconstruction advances in CT-based object detection systems—Final report," Dept. Homeland Security Center Excellence, ALERT, Boston, MA, USA, Tech. Rep. HSHQDC-12-J-00056, 2013.
- [4] M. U. Ghani and W. C. Karl, "Deep learning-based sinogram completion for low-dose CT," in *Proc. IEEE 13th Image, Video, Multidimensional Signal Process. Workshop*, 2018, pp. 1–5.
- [5] I. A. Elbakri and J. A. Fessler, "Statistical image reconstruction for polyenergetic X-ray computed tomography," *IEEE Trans. Med. Imag.*, vol. 21, no. 2, pp. 89–99, Feb. 2002.
- [6] G. Wang, D. L. Snyder, J. A. O'Sullivan, and M. W. Vannier, "Iterative deblurring for CT metal artifact reduction," *IEEE Trans. Med. Imag.*, vol. 15, no. 5, pp. 657–664, Oct. 1996.
- [7] B. De Man, J. Nuysts, P. Dupont, G. Marchal, and P. Suetens, "Reduction of metal streak artifacts in x-ray computed tomography using a transmission maximum a posteriori algorithm," *IEEE Trans. Nucl. Sci.*, vol. 47, no. 3, pp. 977–981, Jun. 2000.
- [8] B. Hamelin *et al.*, "Iterative CT reconstruction of real data with metal artifact reduction," in *Proc. 5th IEEE Int. Symp. Biomed. Imag., From Nano Macro*, 2008, pp. 1453–1456.
- [9] P. Jin, C. A. Bouman, and K. D. Sauer, "A model-based image reconstruction algorithm with simultaneous beam hardening correction for X-Ray CT," *IEEE Trans. Comput. Imag.*, vol. 1, no. 3, pp. 200–216, Sep. 2015.
- [10] Z. Chang, D. H. Ye, S. Srivastava, J.-B. Thibault, K. Sauer, and C. Bouman, "Prior-guided metal artifact reduction for iterative X-Ray computed tomography," *IEEE Trans. Med. Imag.*, vol. 38, no. 6, pp. 1532–1542, Jun. 2019.
- [11] H. Soltanian-Zadeh, J. P. Windham, and J. Soltanianzadeh, "CT artifact correction: An image-processing approach," *Proc. SPIE*, vol. 2710, pp. 477–485, 1996.
- [12] S. Karimi, H. Martz, and P. Cosman, "Metal artifact reduction for CT-based luggage screening," *J. X-Ray Sci. Technol.*, vol. 23, no. 4, pp. 435–451, 2015.
- [13] L. Gjesteby *et al.*, "Deep learning methods for CT image-domain metal artifact reduction," *Proc. SPIE*, vol. 10391, 2017, Art. no. 103910W.
- [14] W. A. Kalender, R. Hebel, and J. Ebersberger, "Reduction of CT artifacts caused by metallic implants," *Radiology*, vol. 164, no. 2, pp. 576–577, 1987.
- [15] L. Gjesteby *et al.*, "Metal artifact reduction in CT: Where are we after four decades?" *IEEE Access*, vol. 4, pp. 5826–5849, 2016.
- [16] A. H. Mahnken *et al.*, "A new algorithm for metal artifact reduction in computed tomography: In vitro and in vivo evaluation after total hip replacement," *Investigative Radiol.*, vol. 38, no. 12, pp. 769–775, 2003.
- [17] P. J. La Rivière, J. Bian, and P. A. Vargas, "Penalized-likelihood sinogram restoration for computed tomography," *IEEE Trans. Med. Imag.*, vol. 25, no. 8, pp. 1022–1036, Aug. 2006.
- [18] E. Meyer, R. Raupach, M. Lell, B. Schmidt, and M. Kachelrieß, "Normalized metal artifact reduction (NMAR) in computed tomography," *Med. Phys.*, vol. 37, no. 10, pp. 5482–5493, 2010.
- [19] Y. Zhang and H. Yu, "Convolutional neural network based metal artifact reduction in X-ray computed tomography," *IEEE Trans. Med. Imag.*, vol. 37, no. 6, pp. 1370–1381, 2018.
- [20] L. Gjesteby, Q. Yang, Y. Xi, Y. Zhou, J. Zhang, and G. Wang, "Deep learning methods to guide CT image reconstruction and reduce metal artifacts," *Proc. SPIE*, vol. 10132, 2017, Art. no. 101322W.
- [21] C. Bernhard, J. Yannan, G. Lars, W. Ge, and M. Bruno De, "Metal-artifact reduction using deep-learning based sinogram completion: Initial results," in *Proc. Develop. X-Ray Tomography XI*, 2017, pp. 631–634.
- [22] R. Anirudh, H. Kim, J. J. Thiagarajan, K. A. Mohan, K. Champlie, and T. Bremer, "Lose the views: limited angle CT reconstruction via implicit sinogram completion," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 6343–6352.
- [23] J. Kim, J. K. Lee, and K. L. Mu, "Accurate image super-resolution using very deep convolutional networks," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 1646–1654.

- [24] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.
- [25] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, "Deep convolutional neural network for inverse problems in imaging," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4509–4522, Sep. 2017.
- [26] P. Isola, J. Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2017, pp. 5967–5976.
- [27] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 47–57, Mar. 2017.
- [28] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Advances Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
- [29] R. Köhler, C. Schuler, B. Schölkopf, and S. Harmeling, "Mask-specific inpainting with deep neural networks," in *Proc. German Conf. Pattern Recognit.*, 2014, pp. 523–534.
- [30] S. Chaudhury and H. Roy, "Can fully convolutional networks perform well for general image restoration problems?" in *Proc. 15th IAPR Int. Conf. Mach. Vision Appl.*, 2017, pp. 254–257.
- [31] J. Xie, L. Xu, and E. Chen, "Image denoising and inpainting with deep neural networks," in *Proc. Advances Neural Inf. Process. Syst.*, 2012, pp. 341–349.
- [32] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. Advances Neural Inf. Process. Syst.*, 2014, pp. 3320–3328.
- [33] S. Do and C. Karl, "Sinogram sparsified metal artifact reduction technology (SSMART)," in *Proc. 3rd Int. Conf. Image Formation X-Ray Comput. Tomography*, 2014, pp. 798–802.
- [34] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2015, pp. 3431–3440.
- [35] W. Luo, Y. Li, R. Urtasun, and R. Zemel, "Understanding the effective receptive field in deep convolutional neural networks," in *Proc. Advances Neural Inf. Process. Syst.*, 2016, pp. 4898–4906.
- [36] B. Kayalibay, G. Jensen, and P. Van der Smagt, "CNN-based segmentation of medical imaging data," 2017, *arXiv:1701.03056*.
- [37] X. Li, S. Chen, X. Hu, and J. Yang, "Understanding the disharmony between dropout and batch normalization by variance shift," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 2682–2690.
- [38] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2009, pp. 248–255.
- [39] Z. Chang *et al.*, "Modeling and pre-treatment of photon-starved CT data for iterative reconstruction," *IEEE Trans. Med. Imag.*, vol. 36, no. 1, pp. 277–287, Jan. 2017.
- [40] M. J. Berger and J. H. Hubbell, "XCOM: Photon cross sections on a personal computer," Center for Radiation Research, National Bureau of Standards, Washington, DC, USA, Tech. Rep. NBSIR 87-3597, 1987.
- [41] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
- [42] D. Gutman *et al.*, "Skin lesion analysis toward melanoma detection: A challenge at the international symposium on biomedical imaging (ISBI), hosted by the international skin imaging collaboration (ISIC)," 2016, *arXiv:1605.01397*.
- [43] E. Rodner, M. Simon, G. Brehm, S. Pietsch, J. W. Wägele, and J. Denzler, "Fine-grained recognition datasets for biodiversity analysis," 2015, *arXiv:1507.00913*.
- [44] D. Malmgren-Hansen, A. Kusk, J. Dall, A. A. Nielsen, R. Engholm, and H. Skriver, "Improving SAR automatic target recognition models with transfer learning from simulated data," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 9, pp. 1484–1488, Sep. 2017.
- [45] E.-P. Rührnschopf and K. Klingenbeck, "A general framework and review of scatter correction methods in cone beam CT. Part 2: Scatter estimation approaches," *Med. Phys.*, vol. 38, no. 9, pp. 5186–5199, 2011.
- [46] W. Van Aarle *et al.*, "Fast and flexible X-ray tomography using the ASTRA toolbox," *Opt. Express*, vol. 24, no. 22, pp. 25129–25147, 2016.
- [47] C. Crawford, "Advances in automatic target recognition (ATR) for CT based object detection system—Final report," Dept. Homeland Secur. Center Excellence, Boston, MA, USA, Task order HSHQDC-12-J-00429, 2014.
- [48] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
- [49] M. Abadi *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," 2016, *arXiv:1603.04467*.
- [50] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vision*, 2014, pp. 818–833.
- [51] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, 2008.