Low-Dimensional Manifold-Constrained Disentanglement Network for Metal Artifact Reduction

Chuang Niu, Wenxiang Cong, Feng-Lei Fan, Mengzhou Li, and Ge Wang

Abstract—Deep neural network-based methods have achieved promising results for computed tomography (CT) metal artifact reduction (MAR), most of which use many synthesized paired images for supervised learning. As synthesized metal artifacts in CT images may not accurately reflect the clinical counterparts, an artifact disentanglement network (ADN) was proposed with unpaired clinical images directly, producing promising results on clinical datasets. However, as the discriminator can only judge if large regions semantically look artifact-free or artifact-affected, it is difficult for ADN to recover small structural details of artifact-affected CT images based on adversarial losses only without sufficient constraints. To overcome the illposedness of this problem, here we propose a low-dimensional manifold (LDM)-constrained disentanglement network (DN), leveraging the image characteristics that the patch manifold of CT images is generally low dimensional. Specifically, we design an LDM-DN learning algorithm to empower the DN through optimizing the synergistic loss functions used in ADN while constraining the recovered images to be on a low-dimensional patch manifold. Moreover, learning from both paired and unpaired data, an efficient hybrid optimization scheme is proposed to further improve the MAR performance on clinical datasets. Extensive experiments demonstrate that the proposed LDM-DN approach can consistently improve the MAR performance in paired and/or unpaired learning settings, outperforming competing methods on synthesized and clinical datasets.

Index Terms—Disentanglement network (DN), low-dimensional manifold (LDM), metal artifact reduction (MAR).

I. INTRODUCTION

METAL objects in a patient, such as dental fillings, artificial hips, spine implants, and surgical clips, will significantly degrade the quality of computed tomography (CT) images. The main reason for such metal artifacts is that the metal objects in the field of view strongly attenuate x-rays or even completely block them so that reconstructed images from the compromised/incomplete data are corrupted in various ways, which usually show as bright or dark streaks. As a result, the metal artifacts significantly affect medical image analysis and subsequent clinical treatment. Particularly, the metal artifacts degrade the counters of the tumor and organs at risk, raising great challenges in optimizing a radio-therapeutic plan [1], [2].

Over the past decades, extensive research efforts [3] have been devoted to CT metal artifact reduction (MAR), leading to various MAR methods. Traditionally, MAR methods mainly focus on the correction of projection data with psychical [4] and other prior knowledge [5], [6]. Particularly, the popular linear [5] and normalized [6] interpolation methods consider a metal trace in the sinogram as a data-missing region and fill in this region with interpolated data. Then, the artifact-reduced image can be reconstructed from the refined projection data using a reconstruction algorithm, such as filtered backprojection (FBP). However, the projection domain methods tend to produce secondary artifacts as it is difficult for the estimated projection values to perfectly match the ground truth. In practice, original projection data and the corresponding reconstruction algorithm are not publicly accessible. To apply the projection-based methods in the absence of original sinogram data, researchers such as Bal and Spies [7] proposed a post-processing scheme that generates the sinogram through forward projection of a CT image first and then applies the projection-based method on the reprojected
sinogram. However, the reprojection and reconstruction may introduce extra errors.

To overcome the limitations of the projection domain methods, researchers are dedicated to reduce metal artifacts in the image domain [8]–[12] or dual domain [13]–[15]. Particularly, with deep learning techniques [16], data-driven methods have recently made great progress in the area of MAR. However, most existing deep-learning-based methods are fully supervised, requiring a large number of paired training images, i.e., each artifact-affected image is associated with the co-registered artifact-free image. In clinical scenarios, it is infeasible to acquire a large number of such paired images. Therefore, the prerequisite for training these methods is to simulate artifact-affected images by inserting metal objects into artifact-free images so that paired images are obtained. However, simulated images cannot reflect all real conditions due to the complex physical mechanism of metal artifacts and many technical factors of the imaging system, degrading the performance of the fully supervised models. To avoid synthesized data, the recently proposed artifact disentanglement network (ADN) [17] only uses clinical unpaired artifact-affected and artifact-free CT images to train a disentanglement network (DN) for minimized adversarial losses, giving promising results on clinical datasets, outperforming the fully supervised methods trained with the synthesized data. However, without strong supervision, the proposed ADN method is not perfect and often fails to preserve structural details due to the insufficient constraints imposed by the discriminators.

In this study, we improve the MAR performance on clinical datasets from two aspects. First, we formulate the MAR problem as the artifact disentanglement while at the same time leveraging the low-dimensional manifold (LDM) of image patches to help recover structural details. Specifically, we train a DN with ADN losses and simultaneously constrain a reconstructed artifact-free image to have a low-dimensional patch manifold. The idea is inspired by the LDM model (LDMM) [18] for image processing and CT image reconstruction [19]. However, how to apply the iterative LDMM algorithm to train the DN is not trivial. To this end, we carefully design an LDM-DN algorithm for simultaneously optimizing objective functions of the DN and the LDM constraint. Second, we improve the MAR performance of the LDM-DN on the real dataset by integrating both unpaired and paired supervision. Particularly, the unpaired supervision is the same as that used in ADN where unpaired images come from artifact-free and artifact-affected groups, and the paired supervision relies on synthesized paired images to train the model in a pixel-to-pixel manner. Although the synthesized data cannot perfectly simulate the clinical scenarios, they still provide helpful information for recovering artifact-free images from artifact-affected ones. Finally, we design a hybrid training scheme to combine both the unpaired and paired supervision for further improving MAR performance on clinical datasets.

The remainder of this article is organized as follows. In the next section, we review the related work. In Section III, we describe the proposed method, including the problem formulation, the construction of a patch set, the computation of a manifold dimensionality, the corresponding optimization algorithm, and the hybrid training scheme. In Section IV, we evaluate the proposed LDM-DN algorithm for MAR on synthesized and clinical datasets. Extensive experiments show that our proposed method consistently improves the performance of both the state-of-the-art ADN methods and the paired learning methods. Finally, we conclude this article in Section V.

II. RELATED WORK

A. Metal Artifact Reduction

CT MAR methods can be classified into three categories, including projection domain methods, image domain methods, and dual-domain methods.

Projection-based methods correct projections for MAR. Some methods of this type [20]–[22] directly correct corrupted data by modeling the underlying physical process, such as beam hardening and scattering. However, the results with these methods are not satisfactory when high atomic number metals are presented. Thus, a more sophisticated way is to treat metal-affected data within the metal traces as unreliable and replace them with the surrogates estimated by reliable ones. Linear interpolation (LI) [5] is a basic and simple method for estimating metal corrupted data. The LI method is prone to generate new artifacts and distorts structures due to mismatched values between the linearly interpolated data and the unaffected data. To address this problem, the prior information is employed to generate a more accurate surrogate in various ways [6], [7], [23]. Among these methods, the state-of-the-art normalized MAR (NMAR) method [6] is widely used due to its simplicity and accuracy. NMAR introduces a prior image of tissue classification for normalizing the projection data before the LI is operated. With the normalized projection data, the data mismatch caused by LI can be effectively reduced for better results than that of the generic LI method. However, the performance of NMAR largely depends on accurate tissue classification. In practice, the tissue classification is not always accurate so that NMAR also tends to produce secondary artifacts. Recently, deep neural networks [24]–[26] were applied for projection correction and achieved promising results. However, such learning-based methods require a large number of paired projection data.

The image domain methods directly reduce metal artifacts by post-processing CT images. Conventionally, some image processing techniques were leveraged to estimate and remove streak artifacts in artifact-affected images. For example, Soltanian-Zadeh et al. [8] identified the streaking artifacts by thresholding the difference image between the original image and its specific low-passed version. However, these handcrafted methods are limited in applications and performance due to the associated assumptions on metal artifacts. As the data-driven approach has a learning ability to identify various metal artifacts from complex background images, deep learning models for MAR are more advantageous than the traditional methods. The first work demonstrating the effectiveness of the deep learning method for MAR was proposed by Gjesteby et al. [27]. The authors designed an encoder–decoder network to remove remaining metal artifacts.
in images corrected using NMAR [6], showing a potential of
deep learning for MAR. Also, RL-ARCNN [10] was designed
for MAR, which uses a convolutional neural network via resid-
ual learning, achieving better results than the plain CNN [28].
cGANMAR [11] regards MAR as an image-to-image trans-
formation and adapts the Pix2pix [29] model to improve
the MAR performance. DestreakNet [12] takes an NMR-
corrected image and a map of image details as the inputs to a
dual-stream network, effectively improving the MAR results.

To benefit from both projection and image domains, various
dual-domain methods were also proposed. In the CNN-MAR
method [13], the CNN first takes the original image and cor-
rected images by BHC [4] and LI [5] as the inputs and then
produces a CNN output image, which is used to generate
a prior image. Then, the projection data of the prior
image is used to correct the original projection data for the
final reconstruction with FBP. DuoDoNet [14] and its vari-
ant [15] introduce an end-to-end dual-domain network to
simultaneously correct a sinogram and the corresponding CT
images.

All the above deep learning methods for MAR require a
large number of synthesized paired project datasets and/or CT
images for training. A recent study [17] has shown that the models [11], [13] trained on the synthesized data cannot gen-
eralize well on the clinical datasets. Then, ADN was designed
and tested on clinical unpaired data, achieving promising
results. However, without a strong supervision, ADN can hardly recover structural details in challenging cases.

In this study, we introduce a novel image prior, i.e., LDM,
and different levels of supervision to train an enhanced dis-
entangle network for an improved MAR performance on
clinical images. Our proposed LDM-constrained disentangle
ment framework with synergistic supervision has a potential
to empower image domain and dual-domain-based methods.

B. Low-Dimensional Manifold

The patch set of natural images has been proved coming from
an LDM [30]–[33]. Based on this low dimensionality of the patch manifold, LDMM first computes the dimen-
sion of the patch manifold based on the differential geometry
and then uses the dimension to regularize an image recov-
ery problem, including image inpainting, super resolution, and
denoising. Based on LDMM, LDMMNet [34] proposes to regu-
larize the combination of input data and output features within
an LDM in the context of the classification task, showing a
competitive performance over popular regularizers, such as
low-rank and Dropout. Recently, Cong et al. [19] proposed
to use LDMM in regularizing the CT image reconstruction,
demonstrating that the LDMM has a strong ability to recover
detailed structures in CT images. Abdullah et al. [35] lever-
age LDMM to reconstruct magnetic resonance images from
sparse k-space measurements by constraining the dimension of
the patch manifold and obtained an improved performance.
Zeng et al. [36] extended the LDMM from image patches to
surface patches in the point cloud for 3-D point cloud
denoising. Zhu et al. [37] leveraged LDMM for hyperspec-
tral image reconstruction, based on the observation that the
spatial–spectral blocks of hyperspectral images are close to
a collection of LDMs. Inspired by the recent results with
LDMM, here we propose an LDM-constrained DN with both
paired and unpaired supervision for improving the MAR
performance on clinical datasets.

III. METHOD

A. Problem Formulation

Let us first introduce the generic neural network-based
MAR method in the image domain. In the supervised learning
mode, we assume that the paired data \( \{x^a_i, x^g_i\}_{i=1}^{N} \) are available, where each artifact-affected image \( x^a_i \) has a corresponding artifact-free image \( x^g_i \) as the ground truth, and \( N \) is the num-
ber of paired images. Then, the deep neural network for MAR
can be trained on this dataset by minimizing the following loss function:

\[
\mathcal{L}_{sup}(\theta) = \frac{1}{N} \sum_{i=1}^{N} \ell\left(g(x^a_i; \theta), x^g_i\right) \tag{1}
\]

where \( \ell \) denotes a loss function, such as the L1-distance func-
tion, and \( g(x^a_i; \theta) \) represents the predicted artifact-free image
of the artifact-affected image \( x^a_i \) by the neural network func-
tion \( g \) with a parameter vector \( \theta \) to be optimized. In practice,
a large number of paired data are synthesized for training the
model, as the clinical datasets only contain unpaired images.

To improve the MAR performance on clinical datasets,
ADN adapts the generative adversarial learning-based DN
for MAR, only requiring the unpaired dataset, \( \{x^a_i, y_j\}, i = 1, \ldots, N_1, j = 1, \ldots, N_2 \), where \( y_j \) represents an artifact-free
image that is not paired with \( x^a_i \), and \( N_1 \) and \( N_2 \) denote the
number of artifact-affected and artifact-free images, respec-
tively. The ADN [17] model consists of several encoders
and decoders, which are trained with several loss functions,
including two adversarial losses, a reconstruction loss, a cycle-
consistent loss, and an artifact-consistent loss. For simplicity,
we denote the ADN loss functions as

\[
\mathcal{L}_{adn}(\theta) = \frac{1}{N_1 N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \ell_{adn}\left(f\left([x^a_i, y_j], \theta\right), [y_j, x^a_i]\right) \tag{2}
\]

where \( \ell_{adn} \) is the combination of all losses of ADN, \( f(\cdot) \) represents a general function of all ADN modules during
training and has multiple inputs and outputs, and here \( \theta \)
denotes the parameters of all modules in ADN. Specifically,
the losses of ADN include two adversarial losses that, respec-
tively, remove and add metal artifacts, a reconstruction loss to
preserve the original contents and avoid fake regions/tissues,
the artifact consistency loss to enforce the removed and syn-
thesized metal artifacts to be consistent, and the self-reduction
loss to constrain that the clean images can be recovered
from the synthesized artifact-affected images. During training,
all loss functions are optimized simultaneously. Without the
paired supervision, the ADN can hardly recover the detailed
structures accurately.

In this work, we introduce a general image property known
as LDM to improve the MAR performance.
As analyzed in the previous studies [18], [19], we reasonably assume that a patch set of artifact-free images samples an LDM. Then, we formulate the MAR problem as follows:

$$\min_{\theta, \mathcal{M}} \mathcal{L}(\theta) + \lambda \dim(\mathcal{M}(P(\theta)))$$

(3)

where $P(\theta)$ denotes the patch set of artifact-free and/or artifact-corrected images and is determined by the network parameters $\theta$, $\mathcal{M}$ is a smooth manifold isometrically embedded in the patch space, $\mathcal{L}(\theta)$ can be any network loss functions, such as $\mathcal{L}_{\text{sup}}$ for paired learning or $\mathcal{L}_{\text{adn}}$ for unpaired learning, and $\lambda$ is a balance hyperparameter. Instead of optimizing the pixel values of an individual image in the traditional image processing method [18], here we aim to optimize network parameters by constraining the predicted patch set $P(\theta)$ to have an LDM for all training images.

To solve the above optimization problem, we need to specify the construction of a patch set, the computation of a manifold dimensionality, and the learning algorithm for simultaneously optimizing the network loss functions and the dimensionality of a patch manifold. In the following sections, we will describe each of these components.

B. Construction and Manifold Dimensionality of Patch Set

In this work, we adopt the state-of-the-art disentangle network in our proposed LDM-based optimization framework under different levels of supervision. For such a DN, we leverage its two branches to construct a patch set. As shown in Fig. 1, one is the artifact-correction branch that maps an artifact-affected image $x^a$ to an artifact-corrected image $\hat{x}$, and the other is an artifact-free branch that maps an artifact-free image $y$ to itself $\hat{y}$. Considering the spatial correspondence between the input/output image and its convolutional feature maps, we concatenate each image patch and its feature vectors to represent the patch. Specifically, some papers [38], [39] have demonstrated that each feature vector in the learned convolutional feature maps corresponds to a fine-grained image patch. Therefore, the high-level feature vectors of the encoder can be used to enhance the representation ability of pixel values, which has been empirically demonstrated effective in Section IV-C2. For the artifact-correction branch, we take patches from the artifact-corrected images, denoted as $\{P_i(\hat{x}, z^i_x)\}$, where $z^i_x$ denotes the transformed version of the original latent code in ADN using a convolutional layer. This transformation aims to compress the feature channels so the dimension of the feature vector is equal to that of the patch vector, please see Section IV-B for the specific values. Similarly, for the artifact-free branch, we take patches from the original images, denoted as $\{P_i(y, z^i_y)\}$, where $z^i_y$ denotes the transformed latent code of $z_y$. As we assume that the patch set of the images without artifacts samples an LDM, the final patch set is the concatenation of these two patch sets, i.e., $P(\hat{x}, z^i_x) \cup P(y, z^i_y)$. Since the patches are determined by the network parameters $\theta$ and input images, hereafter we simply denote the patch set constructed from all possible unpaired images as $P(\theta) = \{P(\hat{x}, z^i_x) \cup P(y, z^i_y)\}$.

In our implementation, the input image size is $H \times W$ and the step size is $s$ for downsampling the encoder features, then the patch size is $s \times s$, the dimension of $z^i_x$ or $z^i_y$ is $s^2 \times H/s \times W/s$, and each patch vector $P_j(\theta) \in \mathbb{R}^d$, $d = 2s^2$. The effect of this design for patch set construction is shown in Section IV-C2.

We adopt the method of LDM [18] to compute the dimensionality $\dim(\mathcal{M})$ of a patch manifold $\mathcal{M}$ as expressed in the following theorem.

Theorem 1: Let $\mathcal{M}$ be a smooth submanifold isometrically embedded in $\mathbb{R}^d$. For any patch $P_j(\theta) \in \mathcal{M}$

$$\dim(\mathcal{M}) = \sum_{i=1}^{d} ||\nabla_{\mathcal{M}} \alpha_i(P_j(\theta))||^2$$

(4)

where $\alpha_i(\cdot)$ is the coordinate function, i.e., $\alpha_i(P_j(\theta)) = P'_j(\theta)$, $P'_j(\theta)$ is the $i$th element in the patch vector $P_j(\theta)$, and $\nabla_{\mathcal{M}} \alpha_i(P_j(\theta))$ denotes the gradient of the function $\alpha_i$ on $\mathcal{M}$. More details on the definition of $\alpha_i$ on $\mathcal{M}$ can be found in [18]. In our implementation, the patch is parameterized by the network parameter vector $\theta$.

C. Optimization

According to the construction of a patch set and the definition of a patch manifold dimensionality, we can reformulate (3) as

$$\min_{\theta, \mathcal{M}} \mathcal{L}(\theta) + \sum_{i=1}^{d} ||\nabla_{\mathcal{M}} \alpha_i||^2_{L^2(\mathcal{M})},$$

s.t. $P(\theta) \subset \mathcal{M}$

(5)

where

$$||\nabla_{\mathcal{M}} \alpha_i||^2_{L^2(\mathcal{M})} = \left( \int_{\mathcal{M}} ||\nabla_{\mathcal{M}} \alpha_i(p)||^2 dp \right)^{1/2}$$

(6)

where (6) is the continuous version of (4), and $p \in \mathcal{M}$ is a patch vector and equivalent to $P_j(\theta)$. To solve this problem, we
design an iterative algorithm, named LDM-DN, for optimizing the LDM-constrained DN. The LDM-DN is based on the algorithm for image processing introduced in [18]. However, the original LDMM [18] is applied to process a single image while LDM-DN is applied to optimize the parameters of neural networks using all training images.

Specifically, given \((\theta^k, \mathcal{M}^k)\) at step \(k\) satisfying \(P_{\theta^k} \subset \mathcal{M}^k\), where \(\mathcal{M}^k\) denotes the parameter manifold, an iterative algorithm, named LDM-DN, for optimizing the parameters of neural networks is designed as follows:

1. Update \(\theta^{k+1}\) and \(\alpha^{k+1}\) as the minimizers of the following objective with the fixed manifold \(\mathcal{M}^k\):

   \[
   \min_{\theta, \alpha} \mathcal{L}(\theta) + \sum_{i=1}^d \|\nabla_{\mathcal{M}} \alpha_i\|_{L^2(\mathcal{M})}^2,
   \]

   s.t. \(\alpha(P(\theta^k)) = P_j(\theta).
   \]

   (7)

2. Update \(\mathcal{M}^{k+1}\):

   \[
   \mathcal{M}^{k+1} = \left\{(\alpha_1^{k+1}(p), \ldots, \alpha_d^{k+1}(p)) : p \in \mathcal{M}^k\right\}.
   \]

   (8)

3. Repeat the above two substeps until convergence.

It is noted that if the iteration converges to a fixed point, \(\alpha^{k+1}\) will be very close to the coordinate functions, and \(\mathcal{M}^{k+1}\) and \(\mathcal{M}^k\) will be very close to each other.

Equation (7) is a constrained linear optimization problem, which can be solved using the alternating direction method of multipliers. Thus, we reduce the above optimization algorithm to the following iterative procedure.

1. Update \(\alpha_i^{k+1}, i = 1, \ldots, d\), with a fixed \(P(\theta^k)\):

   \[
   \alpha_i^{k+1} = \arg \min_{\alpha_i} \sum_{j=1}^d \|\nabla_{\mathcal{M}} \alpha_i\|_{L^2(\mathcal{M})}^2 + \mu \|\alpha(P(\theta^k)) - P(\theta^k) + d^k\|_F^2.
   \]

   (9)

2. Update \(\theta^{k+1}\):

   \[
   \theta^{k+1} = \arg \min_{\theta} \mathcal{L}(\theta) + \mu \|\alpha(r^{k+1}(\theta)) - P(\theta) + d^k\|_F^2.
   \]

   (10)

3. Update \(d^{k+1}\):

   \[
   d^{k+1} = d^k + \alpha^{k+1}(P(\theta^k)) - P(\theta^k + d^k).
   \]

   (11)

where \(d^k\) is the dual variable.

Using the standard variational approach, the solutions to the objective function (9) can be obtained by solving the following PDE:

\[
-\Delta_{\mathcal{M}} u(p) + \mu \sum_{q} \delta(p - q)(u(q) - v(q)) = 0, \quad p \in \mathcal{M}
\]

\[
\frac{\partial u}{\partial n}(p) = 0, \quad p \in \partial \mathcal{M}
\]

(12)

where \(\partial \mathcal{M}\) is the boundary of \(\mathcal{M}\), and \(n\) is the outer normal of \(\partial \mathcal{M}\). Note that here variables \(p\) and \(q\) mean the patch vectors that are determined by the network parameter vector \(\theta\), which is not explicitly denoted for simplicity.

Equation (12) can be solved with the point integral method. For solving the Laplace–Beltrami equation, the key is the following integral approximation:

\[
\int_{\mathcal{M}} \Delta_{\mathcal{M}} u(p) \tilde{R}_i(p, q) dq \approx -\frac{1}{t} \int_{\mathcal{M}} (u(p) - u(q)) \tilde{R}_i(p, q) dq + 2 \int_{\partial \mathcal{M}} \frac{\partial u}{\partial n} \tilde{R}_i(p, q) dt_q
\]

(13)

where \(t > 0\) is a hyperparameter and

\[
R_i(p, q) = C_t \tilde{R}\left(\frac{|p - q|^2}{4t}\right).
\]

(14)

\(R : R^+ \rightarrow R^+\) is a positive \(C^2\) function which is integrable over \([0, +\infty)\), and \(C_t\) is a normalizing factor.

We usually set \(R(r) = e^{-r}\), then \(\tilde{R}_i(p, q) = R_i(p, q) = C_t \exp(|p - q|^2/4t)\) is Gaussian.

Based on the above integral approximation, we approximate the original Laplace–Beltrami equation as

\[
\int_{\mathcal{M}} (u(p) - u(q)) \tilde{R}_i(p, q) dq + \mu \sum_{q \in \mathcal{M}} \tilde{R}_i(p, q)(u(q) - v(q)) = 0.
\]

(16)

This integral equation is easy to discretize over a point cloud.

To simplify the notation, we denote the patch set \(P(\theta^k) = \{p_j\}_{j=1}^m\), where \(m\) is the number of patches. We assume that the patch set samples the submanifold \(\mathcal{M}\) and is uniformly distributed. Then, the integral equation can be written as

\[
\frac{1}{m} \sum_{j=1}^m \tilde{R}(p_i, p_j)(u_i - u_j) + \mu t \sum_{j=1}^m \tilde{R}(p_i, p_j)(v_i - v_j) = 0
\]

(17)

where \(v_j = v(p_j)\), and \(|\mathcal{M}|\) is the volume of the manifold \(\mathcal{M}\).

We rewrite (17) in the matrix form

\[
(L + \tilde{\mu} W)u = \tilde{\mu} Wv
\]

(18)

where \(v = (v_1, \ldots, v_m)\), \(\tilde{\mu} = \mu tm/|\mathcal{M}|\), and \(L\) is an \(m \times m\) matrix,

\[
L = D - W.
\]

(19)

\(W = (w_{ij}), i, j = 1, \ldots, m\) is the weight matrix, \(D = \text{diag}(d_i)\) with \(d_i = \sum_{j=1}^m w_{ij}\), and

\[
w_{ij} = R_i(p_i, p_j), \quad p_i, p_j \in P(\theta^k), \quad i, j = 1, \ldots, m.
\]

(20)

Then, the solutions to the objective function (9) can be obtained by solving \(u\) in (18).

More specifically, the final LDM-DN learning algorithm is described in Algorithm 1, where we assume that the patch set of all images samples an LDM. However, it is impractical to optimize the LDM problem when the number of patches is very large. To this end, we randomly select a batch of images to construct the patch set and then estimate the coordinate functions \(U\), update the network parameters \(\theta\) and dual variables \(d\) in each iteration. Thus, in our implementation, the
number of iterations in training the network is the same as that in the LDM optimization. While practically updating the dual variables \( d \) in the original LDMM algorithm [18], the values of \( d \) usually increase as the number of iterations increases. As the number of iterations is usually very large, the loss value of the LDM term in step 6 of Algorithm 1 will become increasingly large, leading to an instability. To overcome this problem, the dual variables are normalized in step 7 of Algorithm 1.

Algorithm 1: LDM-DN Learning Algorithm

Input: DataSet including unpaired training data \( \{x_i^a, y_i\}_{i=1}^{N_a} \) and/or paired training data \( \{x_i^a, y_i^b\}_{i=1}^{N_b} \), initial network parameters \( \theta^0 \), initial dual variables \( d^0 \), hyperparameters \( \lambda \) and \( \mu \), the number of training epochs \( E \), and the batch size \( bs \).

Output: Network parameters \( \theta^* \).

1. for \( e \in \{1, \cdots, E\} \) do
2. for \( B \in \text{DataSet} \) do
3. Compute the outputs of the disentanglement network given a batch of data \( B = \{x_i^a, y_i\}_{i=1}^{bs} \) or 
   \( B = \{x_i^a, y_i^b\}_{i=1}^{bs} \cup \{x_i^a, y_i^b\}_{i=1}^{bs} \), and construct the patch set, \( P(\theta^k) \), as described in Sub-section III-B.
4. Compute the weight matrix \( W = (w_{ij}) \) and \( L \) with \( P(\theta^k) \), as in Eqs. (20) and (19).
5. Solve the following linear systems to obtain \( U \) \((L + \mu W)U = \mu WV \), where \( V = P(\theta^k) - d^k \).
6. Update \( \theta^k+1 \) using Adam with the following loss function:
   \( J(\theta) = \mathcal{L}(\theta) + \lambda ||U - P(\theta^k) + d^k||^2 \).
7. Construct the patch set \( P(\theta^k+1) \) with \( \theta^k+1 \) and update \( d^k+1 \) as follows:
   \( d^k+1 = d^k + U - P(\theta^k+1) \),
   \( d^k+1 = (d^k - \min(d^k))/\max(d^k) - \min(d^k)) \).
8. \( k \leftarrow k + 1 \).
9. end for
10. end for
11. \( \theta^* \leftarrow \theta^k \).

D. Combination of Paired and Unpaired Learning

ADN only requires unpaired clinical images for training so that the performance degradation of a supervised learning model can be avoided when it is first trained on the synthesized dataset and then transferred to a clinical application. However, the GAN loss-based weak supervision is not enough for recovering full image details. On the other hand, although the synthesized data may not perfectly simulate real scenarios, it does provide helpful information via strong supervision. To benefit from both the strongly and weakly supervised learning, here we design a hybrid training scheme. During training, both unpaired clinical images and paired synthetic images are selected to construct a mini-batch, where the number of unpaired images is the same as that of paired images. Specifically, the unpaired images are used to train all modules and the paired images are naturally used to train the artifact-correction branch, where artifact-free and artifact-corrected images are constrained by the LDM, as shown in Fig. 2. Therefore, the loss function of the combination learning strategy is

\[
\min_{\theta, \mathcal{M}} \mathcal{L}_{\text{adv}}(\theta) + \mathcal{L}_{\text{sup}}(\theta) + \dim(\mathcal{M}(P(\theta)))
\]

where we simply set each loss term having the same contribution to the total loss, and all terms are simultaneously used to optimize the network parameters.

IV. EXPERIMENTAL DESIGN AND RESULTS

A. Datasets

In our experiments, we evaluated the proposed method on one synthesized dataset from DeepLesion [40] and one clinical dataset from Spineweb,\(^1\) which are the same as those used in ADN [17]. It is worth mentioning that the proposed and all competing methods in this article only focus on 2-D slices.

For the synthesized dataset, 4118 artifact-free CT images were randomly selected from DeepLesion. Then, the paired images with and without metal artifacts were synthesized using the method introduced in CNNMAR [13]. Finally, 3918 pairs of images were used for training and 200 pairs for testing. For a fair comparison, the images used for training and testing, and all preprocessing processes are the same as those used in this article of ADN [17].

For the clinical dataset, 6170 images with metal artifacts and 21 190 images without metal artifacts are selected for training, and additional 100 images with metal artifacts were selected for evaluation. The criteria for selecting these images are the same as that in the ADN study. Specifically, if an image contains pixels with Hounsfield unit (HU) values greater than 2500 HU and the number of these pixels is larger than 400, then it is grouped into the artifact-affected class. The images with the largest HU values less than 2000 are grouped into the artifact-free class. Furthermore, to study the effectiveness of combining both paired and unpaired supervision, we randomly selected 6170 images from the artifact-free group. Then, we extracted 6170 metal objects from the images in the artifact-affected group and used CatSim [41] to simulate the paired images by inserting each extracted metal shape into a selected artifact-free image. Finally, 6170 synthesized paired images were obtained.

B. Implementation Details

We implemented different network architecture variants for different learning paradigms, as shown in Fig. 3. For unpaired

\(^1\)spineweb.digitalimaginggroup.ca
learning, we use the same architecture as ADN [17] as shown in Fig. 3(a), and the architectures of all other learning paradigms are the variants of ADN. In Fig. 3(b), to construct the patch set for the LDM constraint, we add two convolutional layers on the top of the encoders in the artifact-corrected and artifact-free branches, respectively, as described in Section III-B. For paired learning only, we simply use the encoder–decoder in the artifact-correction branch as shown in Fig. 3(c). In combination with paired learning and the LDM constraint, we keep two encoder–decoder branches as shown in Fig. 3(d).

In Figs. 1 and 3(b) and (d), the extra convolutional layers are used to compress the channels of the latent code. Specifically, the input image size is 1 × 256 × 256, the downsampling rate is 8, the matrix of $Z_i$ is of 512 × 64 × 64, the matrix of $Z_i^a$ is of 64 × 64 × 64, the patch size is 8 × 8, and the dimension of the point in the patch set is 128. Note that these values are automatically computed according to the descriptions in Section III-B.

We implemented the proposed method in PyTorch. To be fair, we keep all hyperparameters the same as those in ADN [17]. In Algorithm 1, we set the batch size $bs = 1$ due to the limited GPU memory and we empirically set $\lambda = 1$ to balance the LDM and ADN loss terms.

C. Results on Synthesized Dataset

To quantitatively evaluate different methods, we used the popular peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) metrics in our experiments.

1) Reimplementation of ADN: To simulate the unpaired learning, the synthesized paired images were divided into two groups, and then the artifact-affected images were selected from one group and the artifact-free images from the other group. In [17], the ratio of numbers of images in these two groups was simply set to 1:1. However, in clinical scenarios, the number of artifact-affected images is much smaller than the number of artifact-free images. Therefore, we evaluated the effect of the ratio on the MAR performance in the unpaired learning setting. In Table I, ADN-0.85, ADN-0.50, and ADN-0.15 signify various ratios of artifact-affected images to all images. Table I shows that there is little difference among the models trained with different ratios between artifact-affected and artifact-free images. As a representative ratio between the artifact-affected and artifact-free images in the clinical conditions, the ADN-0.15 serves as the baseline in all following experiments, which is denoted as ADN* for simplicity. We found that our reproduced results of ADN* are slightly better than the reported results in the original paper as shown in Table III. To be fair, we use the better reproduced results for comparison.

2) Ablation Study on the Patch Set Construction: To study the effectiveness of the approach for constructing the patch set introduced in Section III-B, here we evaluated some alternative variants. For the full setting, we use the pixel values concatenated with its convolutional features to represent this patch, and then the patch set is constructed with the patches from both artifact-corrected and artifact-free branches as shown in Fig. 1. Based on this full setting, we evaluated the other three variants. First, we only use the pixel values to represent the patch, and the patch set only consists of the patches from the artifact-correction branch, denoted as LDM-basic. Second, we use the pixel values with its convolutional features to represent this patch, and the patch set only consists of the patches from both artifact-corrected and artifact-free branches, denoted as LDM-fea. Third, we only use the pixel values to represent the patch, and the patch set consists of the patches from both artifact-corrected and artifact-free branches, denoted as LDM-high. Table II presents the results of different variants. Specifically, the basic variant LDM-basic can improve the performance of the baseline method ADN*. Then, LDM-fea can further improve the performance, demonstrating the effectiveness of extra latent features for enhancing the patch representation. In contrast, LDM-high that incorporates the artifact-free image patches without enhanced patch representation even harmed the performance compared with that of LDM-basic. However, the full setting that incorporates the
patches from an artifact-free branch with enhanced patch representation can further improve the performance of LDM-fea. We attribute these results to that the effectiveness of LDM depends on the measurement of patch similarity according to (20), which is further determined by the representing features of patches. When using the pixel values with limited ability to represent patches, it is hard to accurately measure the similarity especially for the patches from different unpaired images. In this case, the errors may be introduced in estimating the dimension of the patch manifold, leading to degraded results. On the contrary, incorporating the artifact-free patches represented by augmented features can help estimate the dimension of the manifold, leading to the improved results.

3) Comparative Results: On the synthesized dataset, we evaluated the quantitative and qualitative performance of our proposed method as well as the compared methods. Table III gives the comparison results of the proposed method with the competing methods in the paired and unpaired learning settings. In Table III, ADN* is our improved ADN, see Section IV-C1 for details. The results show that the proposed LDM-DN method outperformed ADN in terms of both PSNR and SSIM metrics in the unpaired learning setting. For the paired learning part in Table III, Sup corresponds to the network architecture in Fig. 3(c), which was trained with the paired data. It is noted that this encoder–decoder architecture contains the skip connections between the encoder and the decoder, which is the same as ADN. LDM-Sup adds the LDM constraints to the Sup during the paired training, corresponding to the architecture in Fig. 3(d). Although the paired images have accurate pixel-to-pixel supervision, the LDM-based learning algorithm can further improve the performance. These results demonstrate that our proposed LDM-DN algorithm explicitly incorporates the LDM prior and consistently improves the existing models in the paired and unpaired learning settings. Clearly, other ways exist to enforce the LDM constraint and could be used or synergistically incorporated into this framework for even better MAR results.

We also visually compared the results as shown in Fig 4. The visual impressions are consistent with the numerical results. In the unpaired learning setting, although ADN can remove a majority of metal artifacts, the local details were not well preserved. By comparing the results of ADN* and LDM-DN, an evident improvement was made on these details. Compared with the unpaired learning (Sup versus ADN*), the ideal paired learning gave better results on the synthesized test dataset. In this case, our proposed LDM-DN learning algorithm obtained further improvements, where the boundaries of structures are sharper visually (LDM-Sup versus Sup).

In addition, we also inspected the difference of CT values between the predicted and the ground-truth images for each method, showing that the predicted pixels values of the proposed methods LDM-Sup and LDM-DN are closer to the ground truth. Fig. 5 shows that LDM-DN consistently works better than the state-of-the-art unpaired learning method ADN for processing different levels of metal artifacts. On the other hand, the proposed LDM constraint is only required in the offline training process. When the trained networks are used for inference, LDM-ADN and LDM-AND-Sup take the same amount of time as ADN and AND-Sup, because all of them use the same encoder and decoder, i.e., $E_{\mu} \rightarrow G_I$ in Fig. 3, to process testing images without extra computation burden. All the above results demonstrate the effectiveness of the proposed LDM-DN algorithm.

**TABLE III**

|       | Paired learning | Unpaired learning |
|-------|-----------------|-------------------|
|       | CNNMAR [13]     | UNet [42]         | eGANMAR [11]    | Sup | LDM-Sup | CycleGAN [43] | DIP [44] | MUNIT [45] | DRIT [46] | ADN [17] | ADN* | LDM-DN |
| PSNR  | 32.5            | 34.8              | 34.1             | 37.6 | 38.0     | 30.8           | 26.4      | 14.9       | 25.6      | 33.6      | 34.0 | 35.0   |
| SSIM  | 91.4            | 93.1              | 93.4             | 96.1 | 96.3     | 72.9           | 75.9      | 7.5        | 79.7      | 92.4      | 92.9 | 94.2   |
D. Results on Clinical Dataset

In this section, we evaluated the proposed networks on the clinical dataset. As there are no ground-truth images for evaluating the performance of the models, here we can only show the visual results of three examples in Fig. 6. The qualitative results were compared in two aspects. First, the artifact-affected regions should be recovered as much as possible. Second, the structures that are not affected by metal artifacts should be preserved as much as possible. In Fig. 6, red and blue boxes are used to emphasize the artifact-affected regions, and the green boxes are to emphasize the artifact-free structures. We see the superiority of the proposed LDM-DN algorithm in preserving artifact-free contents (the green boxes for \textit{LDM-DN} versus \textit{ADN*} especially in the first and third cases) and recovering artifact-affected regions (the blue and red boxes for \textit{LDM-DN} versus \textit{ADN*} especially in the second and third cases). We also evaluated the performance of the model trained with synthesized images on the same dataset in a supervised learning manner. As analyzed in Section IV-C, the results of paired learning are better than those unpaired learning counterparts on the synthesized dataset. However, the results in Fig. 6 indicate that the performance of \textit{Sup} is definitely worse than that of the unpaired learning model for the clinical dataset, as the synthesized data does not reflect the real conditions. This is consistent with the observation in [17]. However, although the performance of \textit{Sup} is degraded, it still shows some merits over the unpaired learning methods, such as some structures are sharper (green boxes of \textit{Sup} versus \textit{ADN*} and \textit{LDM-DN}) and some regions are better (the blue boxes for \textit{Sup} versus \textit{ADN*} and \textit{LDM-DN} in the second case). Combining \textit{ADN} and \textit{Sup} leads to an overcorrection, where some structures are overly brighter (the green boxes for \textit{ADN-Sup} versus others) and some regions are overly dark (the blue boxes for \textit{ADN-Sup} versus others). We attribute these results to that \textit{Sup} tends to keep and sharpen the edges and simultaneously \textit{ADN} is enforcing the structures to look real by enlarging the “black holes” especially when the artifact-affected regions are relatively large. Here, we propose to combine all merits of unpaired learning, paired learning, and LDM through a hybrid learning scheme. As the results of \textit{LDM-DN-Sup} shown in Fig. 6, it can inherit all good points as analyzed above. Particularly, compared with \textit{ADN-Sup}, LDM in \textit{LDM-DN-Sup} constrains that the structurally similar patches, especially the adjacent patches, to be coherent without dramatic changes to become too dark or too bright, so that the black holes generated by \textit{ADN-Sup} can be avoided. The above results on the clinical dataset demonstrate the effectiveness of LDM and the superiority of the hybrid training scheme.

V. DISCUSSION AND CONCLUSION

In this study, we used the CT images under a large HU window for training and testing, which is consistent with the settings in [17]. However, in a recent study [47], we found that if the MAR network is trained with images
under a large window, it is difficult to produce good results when images are analyzed in a small window. Furthermore, we proposed a multiwindow network in [47] to solve this problem. This multiwindow network can be incorporated into the LDM-DN framework to improve the MAR performance in multiwindow settings. On the other hand, most MAR methods were evaluated with not only image quality metrics but also visual inspection. These quantitative metrics are useful to evaluate and compare the effectiveness among different MAR methods, such as for radiation therapy and proton therapy planning [48], in which HU accuracy and structural fidelity are clinically important. Nevertheless, not all textual information can be fully reflected, and PSNR and SSIM are not strongly correlated to diagnostic performance. Therefore, the effectiveness of various MAR methods should be studied further in an application-specific fashion.

We have proposed an LDM-constrained DN for MAR. Specifically, we have designed an LDM-DN learning algorithm to simultaneously optimize the ADN losses of the deep neural networks and constrain the recovered images to have a low-dimensional patch manifold representation. The LDM-DN algorithm can effectively help preserve and recover structural details in CT images. Finally, we have designed a hybrid optimization scheme to combine strongly and weakly supervised learning via our LDM-DN optimization algorithm for integrating paired synthetic data and unpaired clinical data. The experimental results on synthesized and clinical datasets have clearly demonstrated the superiority of the proposed method over the competing methods. At this stage, neither the ADN method nor our proposed method LDM-DN could be perfect but we need to keep improving CT MAR results step by step. LDM-DN serves as such a step has a clear potential to boost the performance of various CT MAR methods.

REFERENCES

[1] D. Giantsoudi et al., “Metal artifacts in computed tomography for radiation therapy planning: Dosimetric effects and impact of metal artifact reduction,” Phys. Med. Biol., vol. 62, no. 8, pp. R49–R80, Apr. 2017.
[2] M. Maerz, O. Koelbl, and B. Dobler, “Influence of metallic dental implants and metal artefacts on dose calculation accuracy,” Strahlenther Onkol., vol. 191, no. 3, pp. 234–241, Mar. 2015.
[3] L. Gjesteby et al., “Metal artifact reduction in CT: Where are we after four decades?,” IEEE Access, vol. 4, pp. 5826–5849, 2016.
[4] J. M. Verburg and J. Seco, “CT metal artifact reduction method correcting for beam hardening and missing projections,” Med. Phys. Biol., vol. 57, no. 9, pp. 2803–2818, 2012.
[5] 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.
[6] E. Meyer, R. Raupach, M. Lell, B. Schmidt, and M. Kachelrieß, “Empirical scatter correction (ESC): A new CT scatter correction method and its application to metal artifact reduction,” in Proc. IEEE. Nucl. Sci. Symp. Med. Imag. Conf., 2010, pp. 2036–2041.
[7] J. Wang, S. Wang, Y. Chen, J. Wu, J. Coatrieux, and L. Luo, “Metal artifact reduction in CT using fusion based prior image,” Med. Phys., vol. 43, no. 1, pp. 1–12, 2016.
[8] B. E. H. Claus, Y. Jin, L. A. Gjesteby, G. Wang, and B. D. Man, “Metal-artifact reduction using deep-learning based Sinogram completion: Initial results,” in Proc. Fully3D, 2017, pp. 613–635.
[9] M. U. Ghani and W. C. Karl, “Fast enhanced CT metal artifact reduction using data domain deep learning,” IEEE Trans. Imag. Comput., vol. 6, pp. 181–193, 2020.
[10] L. Gjesteby et al., “Deep learning methods for CT image-domain metal artifact reduction,” in Proc. Opt. Eng. Appl., 2017, pp. 1–9.
[11] A. P. Broxton, T. Ishikawa, V. de Silva, and A. Zomorodian, “On the local behavior of spaces of natural images,” Int. J. Comput. Vis., vol. 76, no. 1, pp. 1–12, 2008.
[12] G. E. Carlsson, T. Ishkawa, and V. de Silva, “Local behavior of spaces of natural images,” Int. J. Comput. Vis., vol. 54, nos. 1–3, pp. 83–103, 2004.
[13] G. E. Carlsson, T. Ishkawa, and V. de Silva, “Local behavior of spaces of natural images,” Int. J. Comput. Vis., vol. 54, nos. 1–3, pp. 83–103, 2004.
[14] W. Zhu, Q. Qiu, J. Huang, G. Sapiro, and I. Daubechies, “LDMNET: Low dimensional manifold regularized neural networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2018, pp. 2743–2751.
[35] S. Abdullah, O. Arif, M. B. Arif, and T. Mahmood, “MRI reconstruction from sparse $k$-space data using low dimensional manifold model,” IEEE Access, vol. 7, pp. 88072–88081, 2019.
[36] J. Zeng, G. Cheung, M. Ng, J. Pang, and C. Yang, “3D point cloud denoising using graph laplacian regularization of a low dimensional manifold model,” IEEE Trans. Image Process., vol. 29, pp. 3474–3489, 2020.
[37] W. Zhu, Z. Shi, and S. Osher, Low Dimensional Manifold Model in Hyperspectral Image Reconstruction. Cham, Switzerland: Springer Int., 2020, pp. 295–317.
[38] J. Long, N. Zhang, and T. Darrell, “Do ConvNets learn correspondence?” in Proc. 27th Int. Conf. Neural Inf. Process. Syst. (NIPS), vol. 1, 2014, pp. 1601–1609.
[39] C. Niu, J. Zhang, G. Wang, and J. Liang, “GATCluster: Self-supervised Gaussian-attention network for image clustering,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2020, pp. 735–751.
[40] K. Yan, X. Wang, L. Lu, and R. M. Summers, “DeepLesion: Automated mining of large-scale lesion annotations and universal lesion detection with deep learning,” J. Med. Imag., vol. 5, no. 3, pp. 1–11, 2018.
[41] B. D. Man et al., “CatSim: A new computer assisted tomography simulation environment,” in Proc. Med. Imag. Phys. Med. Imag., vol. 6510, 2007, pp. 856–863.
[42] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” in Proc. Med. Image Comput. Comput. Assist. Intervent. (MICCAI), 2015, pp. 234–241.
[43] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 2242–2251.
[44] D. Ulyanov, A. Vedaldi, and V. Lempitsky, “Deep image prior,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2018, p. 6.
[45] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz, “Multimodal unsupervised image-to-image translation,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Sep. 2018, pp. 179–196.
[46] H.-Y. Lee, H.-Y. Tseng, J.-B. Huang, M. Singh, and M.-H. Yang, “Diverse image-to-image translation via disentangled representations,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Sep. 2018, pp. 36–52.
[47] C. Niu, M. Li, and G. Wang, “Multi-window learning for metal artifact reduction,” in Proc. Develop. X-Ray Tomography XIII, 2021, Art. no. 11840.
[48] Y. Jin et al., “Metal artifact reduction for radiation therapy: A simulation study,” in Proc. Med. Imag. Phys. Med. Imag., 2018, pp. 200–205.