Medical Image Tampering Detection: a New Dataset and Baseline

Benjamin Reichman[1], Longlong Jing[1], Oguz Akin[2], and Yingli Tian[1,*]

[1] The City University of New York, New York, NY 10031
[2] Memorial Sloan-Kettering Cancer Center, New York, NY, 10065
*Corresponding author: ytian@ccny.cuny.edu

Abstract. The recent advances in algorithmic photo-editing and the vulnerability of hospitals to cyberattacks raises the concern about the tampering of medical images. This paper introduces a new large scale dataset of tampered Computed Tomography (CT) scans generated by different methods, LuNoTim-CT dataset, which can serve as the most comprehensive testbed for comparative studies of data security in healthcare. We further propose a deep learning-based framework, ConnectionNet, to automatically detect if a medical image is tampered. The proposed ConnectionNet is able to handle small tampered regions and achieves promising results and can be used as the baseline for studies of medical image tampering detection.

Keywords: Tamper Detection · Healthcare Data Security · Medical Imaging · CT Scans · Deep Learning.

1 Introduction

As a non-invasive process, medical imaging plays essential roles in diagnosis and treatment of diseases by creating visual representations of the interior of a body or the function of some organs or tissues such as the commonly used Magnetic Resonance Imaging (MRI) and CT imaging. While machine learning and artificial intelligence technologies are developed for many online applications of medical imaging analysis [11, 22], data security (i.e. vulnerability) becomes a main concern [13]. Patients' medical images can be accessed and manipulated by attackers for multitude of reasons, including financial gain through holding the real data ransom or through insurance fraud [15].

Image tampering can take on many forms. The simplest methods just perform copy-move tampering, resampling, sharpening, blurring, and compression. More intricate methods use classical inpainting algorithms such as Navier-Stokes inpainting, image melding, or patchmatch [3, 5, 7]. More recent deep learning-based methods use Generative Adversarial Networks (GANs) to generate or change the content of images with high visual realism [12, 17]. All these methods can be applied to medical images [10, 15, 20]. Unlike natural scene images which contains rich texture and color information in high resolutions, most medical images are

gray scale with relatively low resolutions which makes the detection of tampered images more challenging for human beings as well as for algorithms.

Some approaches have been proposed to detect non-medical tampered images. Bayar and Stamm proposed a convolutional neural network (CNN) based method to suppress the image content and emphasize the relationship of a pixel with its neighbor [4]. Rao and Ni developed a CNN-based approach to guide the network to detect copy-move tampering by initializing the first layer to only contain high-pass filters [19]. Recently, a few GAN tampering detection methods were reported. Marra *et al.* tested ideas from different areas of tampering detection [14]. Cozzolino *et al.* developed an encoder-decoder network with a latent space that, during training, manually separated the untampered images from the tampered images [6]. Wang *et al.* trained a ResNet-50 to predict whether an image is forged or not [21].

For medical imaging, recently Mirsky *et al.* proposed a deep generative network, CT-GAN, to generate tampered images by producing and inserting visually realistic patches into medical CT images [15]. These images have been reviewed by radiologists in both an open and blind trial respectively and demonstrated misdiagnosis [15]. Although there are some studies exploring medical image tampering detection such as embedding extra information (watermark) into images before transmission [2, 8, 16] as well as non-intrusive techniques to detect image forgery [10, 18, 20], currently there is no existing method to detect the more advanced and realistic tampered medical images generated by deep learning methods.

This paper attempts to detect realistic tampered medical images in lung cancer CT scans (see examples in Fig. 1.) To the best of our knowledge, this is the first work to study how to prevent deep learning based medical image tampering. The contributions of this paper are summarized in the following three aspects: (1) We generate a large-scale dataset consisting of $7,202$ total tampered CT scans with $356,217$ slices by different tampering methods including copy-move forgery, classical inpainting, and deep inpainting. This dataset will serve as the most comprehensive testbed for comparative studies of data security in healthcare and directly benefit the research of the medical image analysis community. We will release the dataset and annotations of the forged regions through our research website; (2) We propose a novel framework, ConnectionNet, to detect tampered images by effectively propagating fine-grained features to the decision function; (3) Experimental results demonstrate that our proposed ConnectionNet is effective at detecting tampered images generated by different methods.

2 Tampered Medical Image Dataset Generation

Medical image tampering detection is a burgeoning field. However, researchers create and conduct experiments on their own private datasets [10, 20]. The CT-GAN tampered dataset is generated by a GAN for testing and evaluation of tampered images [15], but it is small and only contains 41 CT scans and 821 CT

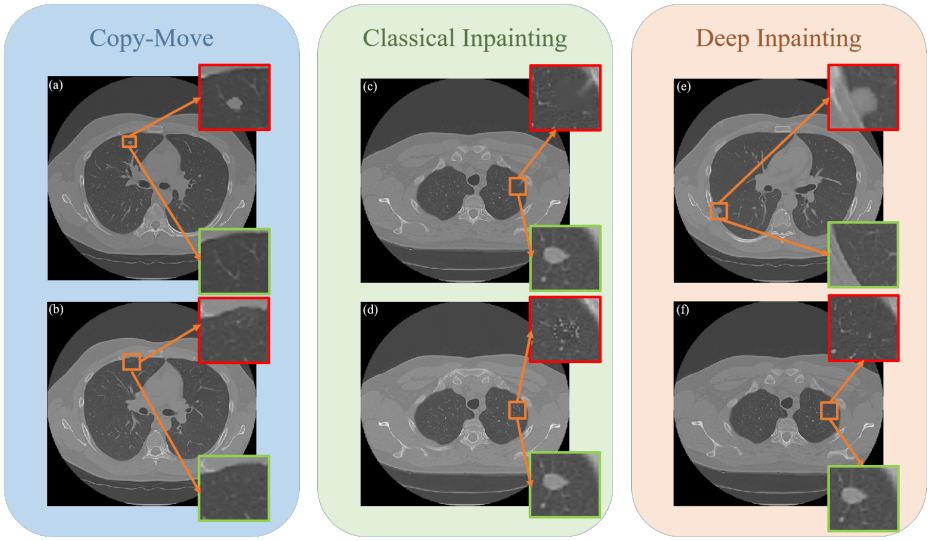

Fig. 1. Examples of our collected LuNoTim-CT dataset which contains tampered lung cancer CT slices generated by three methods: Copy-move, classical inpainting, and deep inpainting. The green patches are original while the red red patches are tampered. (a) copying outer lung tissue and moving it into the inner lung; (b) copying inner lung tissue and moving it to another location in the inner lung; (c) removing a nodule by navier-stokes inpainting; (d) removing a nodule by patchmatch inpainting; (e) adding a nodule by deep inpainting; and (f) removing a nodule by deep inpainting. Note that each tampered slice is only changed in one or more small regions.

slices. To train deep learning-based tampered image detection methods, large-scale datasets are needed for networks to capture the real distribution of the data. Therefore, we have generated a large-scale dataset, LuNoTim-CT (Lung Nodule Tampered Images), consisting of $7,225$ scans with $356,217$ CT slices (see details in Table 1) which can serve as the most comprehensive testbed for comparative studies of data security in healthcare. The LuNoTim-CT dataset will be released through our research website[1].

Our LuNoTim-CT dataset is generated based on the LIDC-IDRI dataset [1], which contains $1,020$ lung CT scans with 883 of them having lung nodules. The CTs in our dataset are tampered by three different tampering methods including copy-move, classical inpainting, and deep inpainting by removing and adding nodules from/to the original CT scans in the original LIDC-IDRI dataset. For each tampered slice, only one tampering method is used at once while the same CT scan can be tampered by different tampering methods at different time. The scans that are excluded from the dataset are the ones where the random process repeatedly led to unrealistic tamperings, either due to what the random process decided or the output of the algorithms used. On average

[1] http://media-lab.ccny.cuny.edu/wordpress/datecode.

about 50 slices are tampered per CT scan. There are no restrictions to how many regions are tampered in one slice, however, tampered regions should not overlap if there is more than one region. In particular, the copy-move tampering method is employed to add tampered regions. The classical inpainting tampering method is used to remove nodules in CT slices. The deep inpainting tampering method is employed to do both adding and removing. It is worth noting that only the slices with nodules present in the base LIDC-IDRI dataset can have nodules removed, thus limiting the total number of slices with removals in our database. For adding, there is no such limitation. Some examples of tampered images generated by different methods in our dataset can be found in Fig. 1.

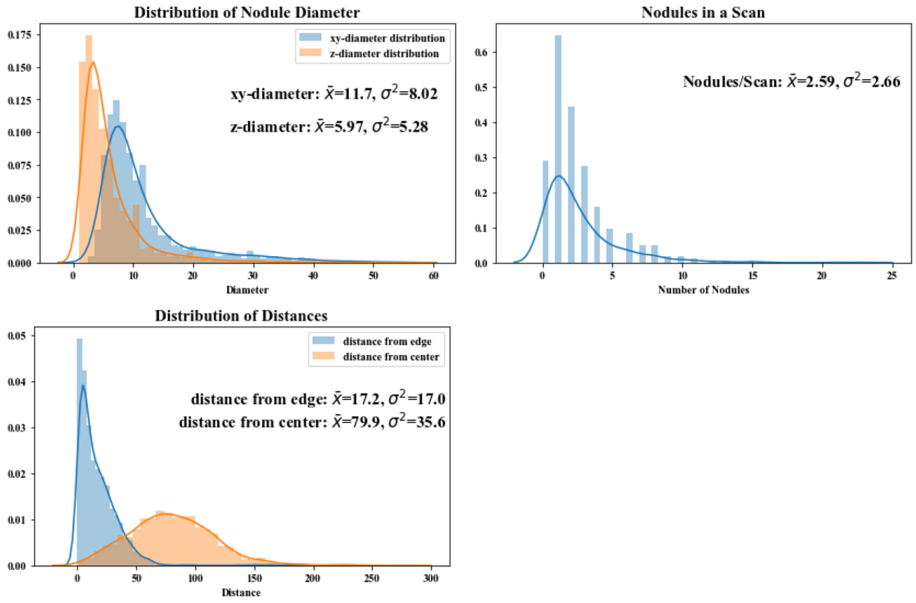

Fig. 2. The statistics of nodule size, number, and location in the LIDC-IDRI dataset which are used as guidance to generate our tampered medical image LuNoTim-CT dataset. Left: Nodule size (unit: pixels); Middle: Number of nodules per scan; Right: Nodule location with distance in pixels.

Tampering location and size selection. To generate realistic fake nodules in lung CT scans, we first calculate the statistics of the nodule size, location, and the number of successive slices a nodule appears in the LIDC-IDRI dataset as shown in Figure 2. There are on average 2 nodules per CT scan and each nodule may appear on six slices. In addition, we observe that more nodules are located closer to the boundary of the inner lung regions. Then guided by the distributions of size and location, a diverse set of forged nodules are generated

in three ways: a) removing the existing nodules; b) randomly adding nodules; and c) randomly moving normal tissue to different areas of the CT slice.

Copy-Move Tampering (CMT). The copy-move tampering method copies an area of an image and moves it to another area. In our LuNoTim-CT dataset, two strategies of copy-move forgeries are performed: 1) moving an outer non-nodule lung area to an inner lung region [see Fig. 1(a)]. As these tampered regions are sufficiently different, it is possible to be observed by human eyes. 2) moving an inner non-nodule lung area to a different position of the inner lung [see Fig. 1(b)]. Since the textures of inner lung regions are self-similar, this type of tampering would be much harder to observe. In both strategies the boundary between the copied patch and its neighborhood is not changed which may presents edge artifacts. The average size of a patch that was copied and moved is between 17×17 pixels. Note that these patches helps disassociate the occurrence of tampering from those of lung nodules. The copy-move method contributes $3,823$ scans ($124,367$ tampered slices) where non-nodule areas are changed.

Classical Inpainting Tampering (CIT). Inpainting algorithms are a class of algorithms that fill in missing patches of an image. Two classical inpainting algorithms are employed to generate tampered CT slices by removing lung nodules: Navier-Stokes inpainting and PatchMatch guided inpainting. Navier-Stokes inpainting is a physics based algorithm that uses ideas of flow from fluid dynamics to propagate the gradient of image intensity smoothly into the inpainted area [5] [see Fig. 1(c)]. PatchMatch inpainting uses a random algorithm to efficiently find patches of images that are similar [3]. Patches with nodules are substituted with similar patches without nodules [see Fig. 1(d)]. The average size of tampered regions is 31×31. The two classical inpainting methods generated $1,753$ CT scans with $29,132$ tampered slices where nodules are removed.

Deep Inpainting Tampering (DIT). Deep inpainting uses deep neural networks to determine how a missing patch of an image should be filled. Compare to copy-move and classical inpainting methods, deep inpainting generates more realistic tampered regions which are harder to detect. In our paper, CT-GAN , a method verified to cause misdiagnosis by radiologists, is employed to add and remove nodules [15]. CT-GAN combines GAN with additive white gaussian noise to blend the generated patch which is further blended by combining the GAN generated cuboid with the original cuboid [15]. The average patch size for these blending procedures is 50×50. Deep inpainting method contributes 758 scans where nodules ($77,898$ slices) are removed and 891 scans ($124,495$ slices) where nodules are added.

In order to verify that the generated dataset is similar to the original dataset that it is based on, Principal Component Analysis (PCA) is applied to image patches from both the generated LuNoTim-CT dataset and the original LIDC-IDRI dataset. PCA aims to find the orthogonal vectors of the training data that explains the most amount of variance between samples [9]. The PCA model was trained using patches from both datasets with the goal of reducing a patch to data along two orthogonal vectors. Then samples from different parts of the

Table 1. Amount of CT scans and slices generated by different methods in our LuNoTim-CT dataset. "CMT": Copy-Move Tampering; "CIT": Classical Inpainting Tampering; and "DIT": Deep Inpainting Tampering.

Tampering Method	Adding		Removing	
	# CT scans	# CT slices	# CT scans	# CT slices
CMT	3,823	124,692	-	-
CIT	-	-	1,753	29,132
DIT	891	124,495	758	77,898
Total	4,714	249,187	2,511	107,030

LuNoTim-CT and the LIDC-IDRI datasets that were withheld during training were inputted to the PCA model to visualize where in the reduced, two dimensional space they would appear. The results shown in Figure 3 show that there is a high degree of overlap between the untampered and the tampered datasets. The CT-GAN added portion of the dataset overlaps the most with the untampered dataset, whereas the narrowest overlap comes from the Patchmatch and Navier-Stokes portion of the dataset.

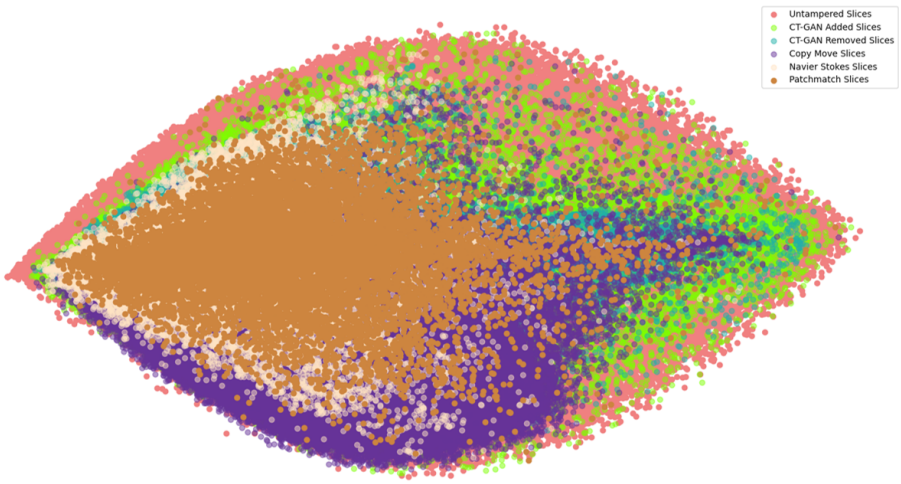

Fig. 3. The overlap of the untampered slices and the tampered slices is demonstrated in two dimensions reduced by applying PCA. The highest degree of overlap is observed for tampered CTs with added patches, whereas the narrowest overlap in the removal part of the dataset.

3 Framework for Medical Tampering Detection

3.1 Architectures

A natural choice for the network $F(x|\theta)$ would be a classification network. The vanilla VGG network makes predictions based on high-level features (fixed dimension vectors) which are extracted by hierarchical convolution layers. The max-pooling layer and global average pooling layer helps the network to capture high-level global features, but leaves low-level features ignored. However, detecting tampering artifacts requires the network to identify tiny regions (size) from a full size image. To augment the capability of capturing fine-grained features, as shown in Figure 4, we propose ConnectionNet to forward the fine-grained features from shallow layers to the fully-connected layers to aid in prediction.

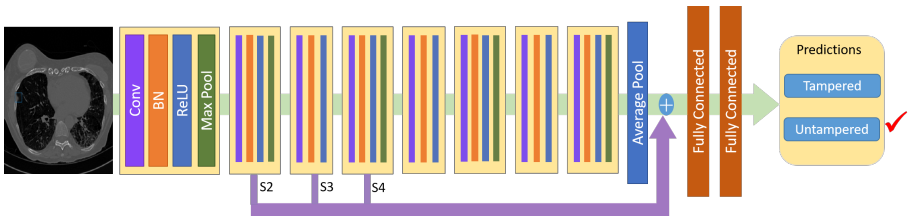

Fig. 4. The proposed framework of ConnectionNet for tampered medical image detection. The backbone of the network is VGG-11. S_n indicates a skip connection that forwards the *nth* convolutional layer to a 1×1 convolution layer that is then sent to an average pooling layer. These fine-grained features from these skip connections are then appended to the output of the network's average pooling layer.

3.2 Model Parameterization

Let $\mathcal{D} = \{(X_1, y_1), (X_2, y_2), ..., (X_n, y_n)\}$ denotes training data of size N, and the i-th datapoint (X_i, y_i) indicate an image from the dataset and its corresponding label represent whether it is tampered or not. The ConnectionNet, $F(x|\theta)$, takes an input image x and predicts if it is tampered by optimizing the parameters of the network θ using D. Cross-entropy loss is employed to optimize the network. Given a medical image x_i, the cross entropy loss is formulated as:

$$loss(x_i|\theta) := -\sum_i (y_i \log(F(x_i|\theta)) + (1 - y_i) \log(1 - F(x_i|\theta))). \quad (1)$$

Given a set of N training pairs $D = \{x_i\}_{i=0}^N$, the overall training loss function is defined as:

$$loss(D) = \min_\theta \frac{1}{N} \sum_{i=1}^{N} loss(x_i|\theta). \quad (2)$$

4 Experiments and Results

4.1 Experimental Setup

The dataset was split in a 64%-16%-20% scheme for training, validation, and testing, respectively. In our experiments, Stochastic Gradient Descent (SGD) optimizer is used with an initial learning rate of 0.001, a momentum of 0.9, and weight decay of 0.0005. It is trained over 30 epochs using a batch size of 6, evenly sampling from the tampered and untampered dataset. To thoroughly evaluate the performance, we evaluated the models using different criteria including precision, accuracy, recall, and Area Under the Curve (AUC).

4.2 Ablation Study of Backbone Networks

Table 2. Using VGG as a backbone network results in better performance in all metrics except AUC other than the ResNet backbone, therefore, it is adopted as the backbone of our framework.

Backbone	Input	Accuracy	Precision	Recall	F1 Score	AUC
Baseline Networks						
VGG	512×512 Image	**0.83**	**0.88**	**0.73**	**0.80**	0.59
ResNet-18	512×512 Image	0.67	0.72	0.57	0.63	**0.75**
ResNet-50	512×512 Image	0.798	0.87	0.66	0.75	0.51

As shown in Table 2, for full resolution images (512×512), ResNet50 outperforms ResNet18 by more than 10% on the DIT portion of the dataset. The VGG network achieves better performance than ResNet in all metrics including classification accuracy, precision, recall, and F1 score. In the ImageNet dataset, the performance of VGG is 68.9% while the performance of ResNet is 76% which is almost 7% higher than the VGG network. This observation indicates that ResNet is more powerful on natural images, however, for medical image tampering detection, VGG significantly outperforms ResNet.

Different from natural image tampering, only one or more small regions are tampered (usually about 50×50 pixels) in each image in our tampered medical image LuNoTim-CT dataset. To capture global features, the current mainstream neural networks employ a max-pooling layer or stride convolution to extract high-level invariant features. With these networks, the information of the tampered region is overwhelmed in the global features. So we tested another straightforward method by training and testing networks with patches cropped from the images. As shown in Table 2, the patch-based method has a relatively high accuracy from being able to identify untampered images, but has very low precision and recall compared to the full resolution method. Thus, a patch-based approach is not optimal in this situation.

4.3 Results of Our Framework

Table 3. The ConnectionNet model achieves a much higher AUC, precision, and recall, depending on the amount and location of skip connections, all while maintaining a similar accuracy to VGG.

Backbone	Input	Accuracy	Precision	Recall	F1 Score	AUC
VGG	512×512 Image	0.83	0.88	0.73	0.80	0.59
VGG-S1	512×512 Image	0.79	0.84	0.72	0.77	0.87
VGG-S2	512×512 Image	0.82	0.81	**0.83**	0.82	0.91
VGG-S1S2	512×512 Image	0.81	0.82	0.80	0.81	0.90
VGG-S2S3S4	512×512 Image	**0.84**	**0.96**	0.72	**0.82**	**0.91**

As shown in Table 3 by forwarding the features from the second-convolution layer to the fully connected layer, the recall of the VGG-S2 network significantly improved by 7%. The VGG-S2S3S4 network greatly improves on the precision (+8%) and AUC (+0.32) of the vanilla VGG network while also slightly increasing the overall accuracy (+1%) and F1 score (+2%).

4.4 Generalizability of the ConnectionNet

Table 4. All networks trained on the DIT portion of the dataset and tested on the CMT, CIT, and DIT datasets generalize pretty well by keeping the same level accuracy, precision, and recall while increasing in the AUC metric. Varying ConnectionNet networks outperform VGG in the different metrics used.

Backbone	Input	Accuracy	Precision	Recall	F1 Score	AUC
VGG	512×512 Image	0.83	0.90	0.75	0.82	0.90
ResNet-18	512×512 Image	0.70	0.74	0.62	0.68	0.78
ResNet-50	512×512 Image	0.81	0.90	0.71	0.79	0.89
VGG-S1	512×512 Image	0.82	0.85	0.79	0.82	0.90
VGG-S2	512×512 Image	0.81	0.79	**0.83**	0.81	0.90
VGG-S1S2	512×512 Image	0.83	0.83	0.82	**0.83**	**0.91**
VGG-S2S3S4	512×512 Image	**0.85**	**0.96**	0.72	0.82	**0.91**

We further evaluate the generalizability of the proposed ConnectionNet on tampered medical images generated by different methods. ConnectionNet is trained only on the DIT portion of the dataset and tested on the three types of tampered images including DIT, CMT, and CIT. As shown in Table 4, our

Table 5. Granular accuracy/recall results of each network on each section of the dataset.

Backbone	Untampered Scans	CT-GAN A	CMT	CT-GAN R	PatchMatch	Navier-Stokes R	Overall
VGG	0.83/**0.75**	0.78/0.64	**0.83/0.76**	0.90/**0.89**	0.89/**0.86**	0.87/0.82	0.83/**0.75**
VGG-S1S2	**0.88**/0.72	0.75/**0.65**	0.78/0.69	0.89/0.85	0.85/0.80	0.91/**0.88**	0.84/0.72
VGG-S2S3S4	0.85/0.73	**0.79**/0.62	**0.83**/0.70	**0.92**/0.88	**0.90**/0.84	**0.92**/0.86	**0.85**/0.73

proposed method performs consistently well and improves on vanilla VGG by 2% in terms of accuracy on a testing set that consists of three different tampering methods. Compared to vanilla VGG, the recall of our VGG-S1S2 and the precision of our VGG-S2S3S4 is significantly higher than other networks. The consistency of the VGG-S1S2 and VGG-S2S3S4 across the different datasets shows that the proposed framework has strong generalization ability across different types of tampering. Table 5 shows the results of the proposed framework and baselines across each section of the dataset. We observe that the lowest accuracy and recall occur when tumors are added to CT scans as opposed to when tumors are removed from CT scans. This suggests that it is more difficult to spot when elements are added to a CT scan than when they are removed. As shown in Figure 3, the portion of the dataset where elements are added has a high degree of overlap (less distinguishable) with the untampered portion of the dataset. On the other hand, the removed portion of the dataset has a narrower range of overlap with the untampered portion of the dataset (more distinguishable).

5 Conclusion

This paper tackles the important medical data security problem of how to detect realistic tampered medical images generated by advanced deep learning methods. We have generated a large scale dataset of tampered chest CT scans and proposed the ConnectionNet framework for detecting tampered CT slices. Our ConnectionNet framework achieves better accuracy and a higher AUC score than the vanilla VGG network. This demonstrates that propagating fine-grained features to the decision function is an effective way to learn the small scale features that helps to detect the removed patches. Our future work includes extending the dataset to more different types of images in addition to CT scans, conducting independent evaluations by radiologists, and further improve the accuracy of unauthorized alteration detection in medical images.

6 Acknowledgements

This work was supported in part by National Science Foundation under award numbers IIS-1400802 and IIS-2041307, Memorial Sloan Kettering Cancer Center Support Grant/Core Grant P30 CA008748, and Intelligence Community Center of Academic Excellence (IC CAE) at Rutgers University.

References

1. Armato III, S.G., McLennan, G., Bidaut, L., McNitt-Gray, M.F., Meyer, C.R., Reeves, A.P., Zhao, B., Aberle, D.R., Henschke, C.I., Hoffman, E.A., et al.: The lung image database consortium (lidc) and image database resource initiative (idri): a completed reference database of lung nodules on ct scans. Medical physics **38**(2), 915–931 (2011)
2. Arsalan, M., Malik, S.A., Khan, A.: Intelligent reversible watermarking in integer wavelet domain for medical images. Journal of Systems and Software **85**(4), 883–894 (2012)
3. Barnes, C., Shechtman, E., Finkelstein, A., Goldman, D.B.: PatchMatch: A randomized correspondence algorithm for structural image editing. ACM Transactions on Graphics (Proc. SIGGRAPH) **28**(3) (Aug 2009)
4. Bayar, B., Stamm, M.C.: A deep learning approach to universal image manipulation detection using a new convolutional layer. In: Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security. pp. 5–10 (2016)
5. Bertalmio, M., Bertozzi, A.L., Sapiro, G.: Navier-stokes, fluid dynamics, and image and video inpainting. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001. vol. 1, pp. I–I. IEEE (2001)
6. Cozzolino, D., Thies, J., Rössler, A., Riess, C., Nießner, M., Verdoliva, L.: Forensictransfer: Weakly-supervised domain adaptation for forgery detection. arXiv preprint arXiv:1812.02510 (2018)
7. Darabi, S., Shechtman, E., Barnes, C., Goldman, D.B., Sen, P.: Image Melding: Combining inconsistent images using patch-based synthesis. ACM Transactions on Graphics (TOG) (Proceedings of SIGGRAPH 2012) **31**(4), 82:1–82:10 (2012)
8. Das, S., Kundu, M.K.: Effective management of medical information through roi-lossless fragile image watermarking technique. Computer methods and programs in biomedicine **111**(3), 662–675 (2013)
9. Geladi, P., Isaksson, H., Lindqvist, L., Wold, S., Esbensen, K.: Principal component analysis of multivariate images. Chemometrics and Intelligent Laboratory Systems **5**(3), 209 – 220 (1989). https://doi.org/https://doi.org/10.1016/0169-7439(89)80049-8, http://www.sciencedirect.com/science/article/pii/0169743989800498
10. Ghoneim, A., Muhammad, G., Amin, S.U., Gupta, B.: Medical image forgery detection for smart healthcare. IEEE Communications Magazine **56**(4), 33–37 (2018)
11. Gong, E., Pauly, J.M., Wintermark, M., Zaharchuk, G.: Deep learning enables reduced gadolinium dose for contrast-enhanced brain mri. Journal of Magnetic Resonance Imaging **48**(2), 330–340 (2018)
12. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems. pp. 2672–2680 (2014)
13. Jalali, M.S., Kaiser, J.P.: Cybersecurity in hospitals: a systematic, organizational perspective. Journal of medical Internet research **20**(5), e10059 (2018)
14. Marra, F., Gragnaniello, D., Cozzolino, D., Verdoliva, L.: Detection of gan-generated fake images over social networks. In: 2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR). pp. 384–389. IEEE (2018)
15. Mirsky, Y., Mahler, T., Shelef, I., Elovici, Y.: Ct-gan: Malicious tampering of 3d medical imagery using deep learning. In: 28th {USENIX} Security Symposium ({USENIX} Security 19). pp. 461–478 (2019)

16. Nyeem, H., Boles, W., Boyd, C.: A review of medical image watermarking requirements for teleradiology. Journal of digital imaging **26**(2), 326–343 (2013)
17. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: Feature learning by inpainting. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2536–2544 (2016)
18. Qureshi, M.A., Deriche, M.: A bibliography of pixel-based blind image forgery detection techniques. Signal Processing: Image Communication **39**, 46–74 (2015)
19. Rao, Y., Ni, J.: A deep learning approach to detection of splicing and copy-move forgeries in images. In: 2016 IEEE International Workshop on Information Forensics and Security (WIFS). pp. 1–6. IEEE (2016)
20. Ulutas, G., Ustubioglu, A., Ustubioglu, B., Nabiyev, V.V., Ulutas, M.: Medical image tamper detection based on passive image authentication. Journal of digital imaging **30**(6), 695–709 (2017)
21. Wang, S.Y., Wang, O., Zhang, R., Owens, A., Efros, A.A.: Cnn-generated images are surprisingly easy to spot... for now. arXiv preprint arXiv:1912.11035 (2019)
22. Yi, X., Walia, E., Babyn, P.: Generative adversarial network in medical imaging: A review. Medical image analysis p. 101552 (2019)