GAN-Amis: Evaluating Clustering of GAN-Generated Medical Images Using Custom and Pre-trained CNN Architectures to Identify GAN Fingerprints

Notebook for ImageCLEF Lab at CLEF 2024

Aman Upganlawar[1,*], Aarti Lad[1] and Arnav Desai[1]

[1]*Pune Institute of Computer Technology, Pune, India.*

Abstract

ImageCLEF is an annual evaluation forum that addresses research tasks in image analysis and cross-language annotation. In ImageCLEF 2024, a challenging task named "Detect Generative Model's Fingerprints" was introduced, focusing on identifying unique fingerprints left by generative models on synthetic images. In this paper, we present our approach to this task, which involves exploring the hypothesis that generative models imprint distinct fingerprints on their synthetic outputs. We describe the task setup, dataset composition, and related works in detail. Our methodology involves employing various deep learning architectures, including a custom CNN architecture, EfficientNet, ResNet50, MobileNetV2, VGG19, and Xception, to extract features from synthetic images and perform clustering using K-means algorithm. We conducted experiments on both development and test datasets, evaluating the effectiveness of different architectures in detecting model fingerprints. Our results reveal varying performance across architectures, with challenges encountered in accurately clustering synthetic images. Through this study, we contribute insights into the complexities of detecting generative model fingerprints and discuss potential avenues for improvement in future research endeavors.

Keywords

Clustering, GAN Fingerprint detection, Convolutional neural networks(CNNs), Generative models

1. Introduction

ImageCLEF is an evaluation forum organized annually that encompasses research tasks oriented towards image analysis and cross-language annotation. ImageCLEF 2024 [1] focused on various challenges aimed at improving research contributions in visual analysis, annotation, classification, and retrieval tasks. Medical-based tasks have been included since the second edition of ImageCLEF under the tag ImageCLEFMedical[2], which has annually hosted several medical domain-based tasks for significant achievements since 2004. Amongst the tasks proposed for the year 2024, Detect Generative Model's Fingerprints is indeed a challenging task within the track.

In the healthcare domain, medical imaging plays a pivotal role in disease diagnosis and treatment planning. Lung cancer is one of the leading causes of cancer-related deaths worldwide. Computed tomography (CT) scans are widely used for lung cancer screening, diagnosis, and treatment response assessment. The application of GANs in lung CT imaging has shown promising results in various tasks, including image denoising, segmentation, and synthesis[3]. However, the detection of GAN-generated fingerprints on lung CT scans remains an under-explored research area.

The detection of GAN-generated images is a challenging task due to the high quality and realistic nature of the generated images. Several methods have been proposed for detecting GAN-generated images, including the use of statistical features[4], deep learning-based approaches[5], and frequency-domain analysis[6]. However, these methods have limitations, such as the requirement of a large number of images for training, the inability to generalize to unseen GAN architectures, and susceptibility to image compression and post-processing operations.

CLEF 2024: Conference and Labs of the Evaluation Forum, September 09–12, 2024, Grenoble, France

*Corresponding author.

aman.upg27024@gmail.com (A. Upganlawar); aarti.lad@gmail.com (A. Lad); arnavdesai235@gmail.com (A. Desai)

We have employed a CNN architecture and several other widely used classification architectures to detect complex patterns within each generated image. We then used standard K-means clustering using the extracted features from these architectures to cluster the images from the test dataset.

In the following sections, we first describe the task and the dataset provided for ImageCLEF Medical 2024 for the task DETECT GENERATIVE MODELS' FINGERPRINTS in detail in Section 2, followed by the related works which discuss approaches to this task in Section 3. In Section 4, we describe the details of the methods employed, and Section 5 presents the experiments, results, and discussion. Section 6 elucidates the conclusion for this task.

2. Task Description

The primary objective of this task is to explore the hypothesis that generative models imprint unique fingerprints on the synthetic images they produce. This investigation focuses on understanding whether different generative models or architectures leave discernible signatures within the synthetic images they generate.

Participants are provided with a set of synthetic images generated through various generative models. The task is to identify and detect the distinct "fingerprints" associated with each model. This involves analyzing the characteristics, patterns, or features embedded in the synthetic images to determine the specific traits that define each model's output. The ultimate goal is to distinguish between images created by different models and to uncover the unique imprints left by each generative model, facilitating model attribution recognition.

This task is fundamentally a clustering problem, where the aim is to group images based on the unique fingerprints left by different generative models. It is important to note that the number of clusters identified in the training and development datasets may differ from those in the testing dataset, adding a layer of complexity to the task.

To achieve this task, we had access to two datasets:

Development Dataset: The development dataset consists of 600 images generated using three different generative models. Each model is represented by 200 images of size 256x256 and are organized in annotated folders.

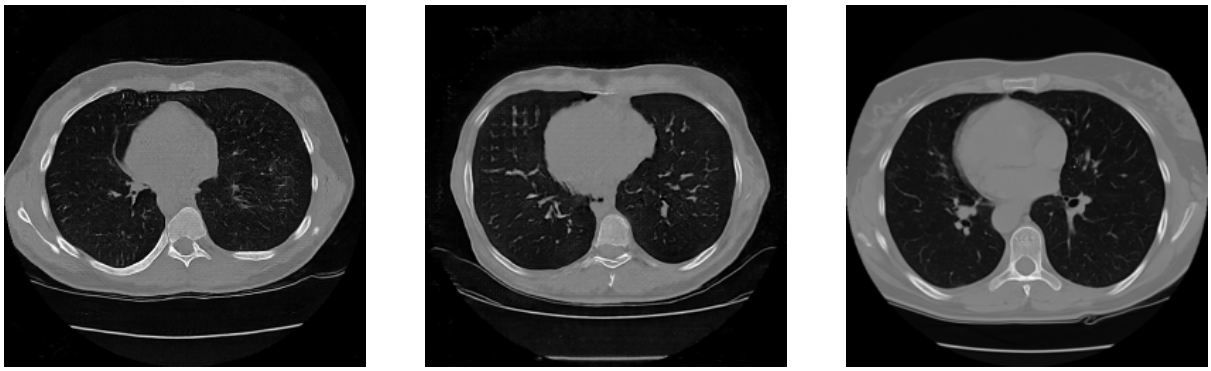
Test Dataset: This task involves working with a dataset comprising 3000 computed tomography (CT) slices, each sized at 256x256 pixels and grayscale. These slices were generated using four distinct generative models. For the tasks, participants must refer to these models as [1, 2, 3, 4].

The subsets of real images are composed of axial slices of 3D computed tomography (CT) images taken from a dataset of approximately 8,000 lung tuberculosis patients. No real data was used in this task in either the development or the test dataset and the images obtained were solely generated by the generative models. Data Description The benchmarking image dataset consists of axial slices of 3D CT images from approximately 8,000 lung tuberculosis patients. These images, stored as 8-bit PNG files with dimensions of 256x256 pixels, vary in appearance; some may look relatively "normal," while others exhibit lung lesions, including severe cases.

In addition to these real CT images, participants are provided with artificial slice images of the same size (256x256 pixels) generated using different generative models, including Generative Adversarial Networks (GANs) and Diffusion Neural Networks. The challenge is to analyze these synthetic images to identify and differentiate the unique fingerprints imprinted by each generative model. The figures 1 and 2 represent some sample images from the datasets for better insight into the nature of images in this task.

3. Related works

There have been several attempts to discern real images from fake(generated images) when it comes to GAN detection in generated face images. Matern et al.[7] extracted several geometric facial features which were then fed to a Support Vector Machine (SVM) classifier to distinguish between real and



(a) Class 1 (b) Class 2 (c) Class 3

Figure 1: Sample images from the three classes in the development dataset

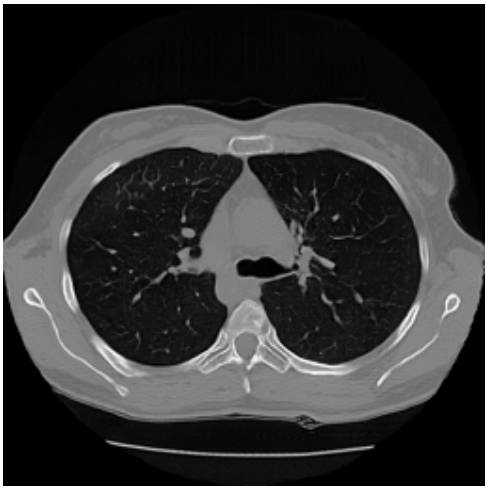


Figure 2: Sample image from the test dataset

synthetic face images. Yang et al.[8] exploited the weakness of GANs in generating consistent head poses and trained a SVM to distinguish between real and synthetic faces based on the estimation of the 3D head pose. Sinitsa and Fried[9] introduce a new method for detecting synthetic images and analyzing model lineage using deep image fingerprints. Their approach enables the detection of images from known generative models and establishes relationships between fine-tuned models. Furger et al.[10] explore the applications of GANs in dermatologic imaging, emphasizing the detection of unique patterns in synthetic images. Their work underscores the importance of fingerprint detection in ensuring the authenticity of medical images. Tang et al.[11] investigate the synthesis of fingerprint images using deep generative models, focusing on the statistical features and deep learning approaches required for effective fingerprint detection in generated images. However these approaches cannot be generalized for detection of fingerprints in other use cases, especially here due to a problem caused by domain shifting in the applications. For CT images, the work has been very limited, however the works presented in the last edition of the ImageCLEF[12][13][14][15][16] give us an insight into tackling this problem. Since our task involves clustering of GANs generated images, training a robust classifier whose features could then be extracted for clustering seems to show good outcomes. M. Russo, M. Stella et al.[17] proposed a CNN method on which they compared VGG16 and ResNet50 architectures on the histopathology images of lungs to detect and classify lung cancer, S. Hoo-Chang, et al.[18] implemented 5 different CNN architecture based neural networks methods to identify the interstitial lung disease using the dataset of 2D images of CT scan slices. The employment of CNNs to identify deep details within the GAN generated images as well as on medical image classification tasks proves a strong case to use these

architectures for performing classification the development dataset provided.

4. Methodology

4.1. Convolutional Neural Network

The first model devised for this task is based on a Convolutional Neural Network (CNN) architecture. CNNs are widely used for image classification tasks due to their ability to effectively capture spatial features from images. In this study, we first constructed and preprocessed the datasets for training and validation. The training dataset was created by loading images from the specified directory, with images automatically labeled based on the directory structure. The images, in grayscale format with a resolution of 256x256 pixels, were loaded in batches of 16. Similarly, the validation dataset was prepared using images from a separate directory with identical specifications. To facilitate model training, we applied a preprocessing function that normalized the image pixel values to a range between 0 and 1 by casting the images to float32. Additionally, the labels were one-hot encoded to represent the three different classes, ensuring compatibility with our classification model. This preprocessing step was applied to both the training and validation datasets. The resulting architecture of our CNN model for detecting fingerprints in synthetic lung CT images begins with an input layer for grayscale images of size 256x256 pixels. The input is followed by a series of convolutional layers that progressively increase the number of filters, capturing increasingly complex features. The first stage consists of two convolutional layers with 64 filters each, followed by batch normalization and ReLU activation. This pattern is repeated, with the number of filters doubling in each subsequent stage: 128, 256, and 512 filters, respectively. Max pooling layers follow each pair of convolutional layers to downsample the feature maps, reducing their spatial dimensions while preserving crucial information. After the four stages of convolution and downsampling, the model includes a bottleneck layer with two convolutional layers having 1024 filters each, continuing the pattern of batch normalization and ReLU activation. From the bottleneck layer, the feature maps are flattened into a one-dimensional vector, which is then passed through two fully connected (dense) layers with 256 and 64 units, respectively, each employing ReLU activation to introduce non-linearity and enable the model to learn complex patterns. The architecture concludes with a dense output layer with three units, corresponding to the three classes of generative models, using a softmax activation function to generate class probabilities. the training process with a batch size of 16 and a learning rate of 10-4, over a span of 200 epochs. The model was compiled using the Adam optimizer with categorical cross-entropy loss, and accuracy as the evaluation metric. We incorporated several callbacks: ModelCheckpoint to save the best-performing model, CSVLogger to record the training log, TensorBoard for visualization, and EarlyStopping to halt training if the validation loss did not improve for 50 consecutive epochs. The last layer of the model was removed to create a feature extractor, which outputs the penultimate layer's activations. This modified model was used to predict features for the validation dataset. These features were then clustered using K-means clustering with four clusters, corresponding to the four generative models used to create the test dataset. To validate our approach, we also generated clusters for a smaller subset of the test dataset. The same feature extractor was employed to predict features from this dataset, and K-means clustering was applied to these features as well. Finally, we performed clustering on the full test dataset. The image files were processed similarly, and the features were extracted using the same feature extractor. These features were clustered into four groups using K-means, and the resulting cluster labels were analyzed to assess the performance of our approach in distinguishing between the synthetic images generated by different models.

4.2. Existing architectures

In addition to the custom CNN architecture, we leveraged several pre-trained deep learning models to enhance feature extraction and clustering performance, specifically EfficientNet, ResNet50, MobileNetV2, VGG19, and Xception. These architectures, each with unique strengths, were fine-tuned

on the development dataset to adapt to the specific nuances of synthetic lung CT images. EfficientNet, for its ability to balance accuracy and computational efficiency, scales depth, width, and resolution uniformly, making it versatile for various image recognition tasks. This model's compound scaling approach enables it to extract a diverse set of features. ResNet50, with its deep residual learning framework, excels in capturing intricate patterns and mitigating the vanishing gradient problem. Its ability to maintain performance with increased depth ensures that it captures detailed and hierarchical features. MobileNetV2, optimized for mobile and embedded vision applications, offers a lightweight yet effective feature extraction capability. Its inverted residuals and linear bottlenecks allow it to efficiently process images. Despite its efficiency, MobileNetV2 maintains robust feature extraction performance, which is beneficial for our clustering task. VGG19, characterized by its deep and uniform architecture, provides a straightforward yet powerful approach to feature extraction. Its simplicity in design, with sequential convolutional layers, enables it to capture hierarchical features effectively. The depth of VGG19 allows it to learn complex representations, which can be particularly useful for distinguishing fine-grained differences in the synthetic images. Xception, an extension of the Inception architecture, utilizes depth wise separable convolutions, which decouple the learning of spatial and channel-wise features. This approach significantly reduces the number of parameters while maintaining high performance, making Xception both efficient and powerful. Each of these pre-trained models was custom-trained on the development dataset to fine-tune their weights for our specific task. This custom training ensured that the models were well-adapted to the characteristics of the synthetic lung CT images generated by different models. After training, the final classification layers of these models were removed to use the deep feature representations generated by the preceding layers. The extracted features from each model were then subjected to K-means clustering, grouping the images based on the unique fingerprints left by different generative models. This multi-architecture approach allowed us to comprehensively evaluate and utilize the strengths of different deep learning models, enhancing the robustness and reliability of our detection methodology. By comparing the clustering results across these architectures, we aimed to identify the most effective model for detecting generative model fingerprints in synthetic lung CT images.

5. Results and discussion

The performance of the clustering was evaluated using the Adjusted Rand Index (ARI), a standard metric for comparing the similarity between two data clusterings. The ARI is a measure of the similarity between two clusterings, adjusted for the chance grouping of elements. It ranges from -1 to 1, where: 1 indicates perfect agreement between the two clusterings, 0 indicates a random clustering result, Negative values indicate less agreement than expected by chance. The formula for the Adjusted Rand Index is given by:

$$\text{ARI} = \frac{\text{RI} - \mathbb{E}[\text{RI}]}{\max(\text{RI}) - \mathbb{E}[\text{RI}]}$$

where:

- RI is the Rand Index, which measures the similarity between two clusterings.
- $\mathbb{E}[\text{RI}]$ is the expected value of the Rand Index for a random clustering.
- $\max(\text{RI})$ is the maximum value of the Rand Index.

The ARI is particularly useful because it adjusts for the chance of random clusterings, providing a more accurate measure of clustering performance.

The following table shows the ARI scores for different architectures:

The ARI scores indicate the effectiveness of each model in clustering the images generated by different generative models. An ARI score close to 0 indicates that the clustering is random and does not effectively capture the underlying structure. Negative ARI scores, as seen in several of the models, suggest that the clustering results are even less consistent than what would be expected by chance. EfficientNet and

Table 1
ARI Scores for Different Architectures along with their corresponding submission IDs

Architecture	ARI Score
EfficientNet(ID#: 520)	-0.0005467941
MobileNetV2(ID#: 518)	-0.0000102128
Xception(ID#: 517)	-0.0020193309
ResNet50(ID#: 516)	0.0000795212
VGG19(ID#: 513)	-0.0009935105
Custom CNN(ID#: 277)	-0.0006152185

VGG19 produced slightly negative ARI scores, indicating poor clustering performance. EfficientNet's more complex scaling might not have aligned well with the synthetic image features, while VGG19's simpler architecture might have missed intricate patterns. MobileNetV2 achieved a near-zero ARI score, suggesting random clustering performance. Despite its efficiency and effectiveness in other tasks, its lightweight design might not have captured enough discriminative features for this task. Xception had the most negative ARI score, possibly due to its complex architecture failing to generalize well to the specific synthetic features of the images. ResNet50 produced a slightly positive ARI score, indicating that it performed better than random clustering. Its residual connections likely helped in preserving more relevant features, making it somewhat more effective for this task. Custom CNN also resulted in a negative ARI score, suggesting that it might not have captured the generative model fingerprints as effectively as was expected. The varying ARI scores across different architectures highlight the differences in their capabilities to capture and distinguish the synthetic image features. ResNet50's slight positive score shows some promise due to its residual learning capabilities, which help in retaining more complex patterns. In contrast, Xception's lower performance might be attributed to its more sophisticated architecture not aligning well with the specific dataset characteristics. MobileNetV2's near-zero score suggests that its efficient, lightweight structure did not capture enough details necessary for effective clustering. The relatively poor performance of EfficientNet and VGG19 could be due to their architectural designs not being optimal for the type of features present in the synthetic lung CT images. Overall, these results indicate that while pre-trained models provide powerful feature extraction capabilities, their effectiveness in this specific task of clustering generative model fingerprints varies significantly. Custom tuning and perhaps hybrid approaches combining multiple architectures might be necessary to achieve better clustering performance.

6. Conclusion

In this paper, we explored the challenging task of detecting generative models' fingerprints on synthetic images, particularly focusing on lung CT scans. Our proposed method, utilizing modified CNN architectures for feature extraction followed by K-means clustering, showcased limitations in effectively clustering the images based on the unique fingerprints of different generative models. Despite custom training on the development dataset, our approach yielded unsatisfactory results.

However, our study highlights several important insights and avenues for improvement in this domain. Firstly, while our method struggled to distinguish between images generated by different models, it underscores the complexity of the task and the need for more sophisticated techniques. Future research could explore ensemble approaches or hybrid models that combine features from multiple architectures to leverage their respective strengths. Additionally, incorporating domain-specific knowledge, such as lung anatomy and pathology, into the feature extraction process could enhance the model's ability to discern subtle differences in synthetic images.

Furthermore, our study sheds light on the importance of dataset diversity and size. The limited size of the development dataset may have hindered the generalization ability of our model. Therefore, expanding the dataset to include a wider range of synthetic images generated by various models could lead to more robust and generalizable results.

Moreover, exploring alternative clustering algorithms beyond K-means could offer valuable insights. Hierarchical clustering or density-based clustering methods may better capture the underlying structure of the data, especially in scenarios where the number of clusters is unknown or varies.

In conclusion, while our proposed method demonstrated limitations in effectively detecting generative model fingerprints on synthetic images, it provides a foundation for future research in this area. By addressing the identified shortcomings and leveraging advancements in machine learning techniques, we can pave the way towards more accurate and reliable methods for attributing synthetic images to their respective generative models, thus ensuring the integrity and authenticity of medical imaging data.

References

[1] B. Ionescu, H. Müller, A. Drăgulinescu, J. Rückert, A. Ben Abacha, A. Garcıa Seco de Herrera, L. Bloch, R. Brüngel, A. Idrissi-Yaghir, H. Schäfer, C. S. Schmidt, T. M. Pakull, H. Damm, B. Bracke, C. M. Friedrich, A. Andrei, Y. Prokopchuk, D. Karpenka, A. Radzhabov, V. Kovalev, C. Macaire, D. Schwab, B. Lecouteux, E. Esperança-Rodier, W. Yim, Y. Fu, Z. Sun, M. Yetisgen, F. Xia, S. A. Hicks, M. A. Riegler, V. Thambawita, A. Storås, P. Halvorsen, M. Heinrich, J. Kiesel, M. Potthast, B. Stein, Overview of ImageCLEF 2024: Multimedia retrieval in medical applications, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction, Proceedings of the 15th International Conference of the CLEF Association (CLEF 2024), Springer Lecture Notes in Computer Science LNCS, Grenoble, France, 2024.

[2] A. Andrei, A. Radzhabov, D. Karpenka, Y. Prokopchuk, V. Kovalev, B. Ionescu, H. Müller, Overview of 2024 ImageCLEFmedical GANs Task – Investigating Generative Models' Impact on Biomedical Synthetic Images, in: CLEF2024 Working Notes, CEUR Workshop Proceedings, CEUR-WS.org, Grenoble, France, 2024.

[3] X. Yi, E. Walia, P. Babyn, Generative adversarial network in medical imaging: A review, Medical Image Analysis 58 (2019) 101552. doi:10.1016/j.media.2019.101552.

[4] Y. Li, S. Lyu, Exposing deepfake videos by detecting face warping artifacts, arXiv preprint arXiv:1811.00656 (2018).

[5] M. Barni, K. Kallas, E. Nowroozi, B. Tondi, Cnn detection of gan-generated face images based on cross-band co-occurrences analysis, in: 2020 IEEE international workshop on information forensics and security (WIFS), IEEE, 2020, pp. 1–6.

[6] J. Frank, T. Eisenhofer, L. Schönherr, A. Fischer, D. Kolossa, T. Holz, Leveraging frequency analysis for deep fake image recognition, in: International conference on machine learning, PMLR, 2020, pp. 3247–3258.

[7] S. McCloskey, M. Albright, Detecting gan-generated imagery using color cues, arXiv preprint arXiv:1812.08247 (2018).

[8] X. Yang, Y. Li, S. Lyu, Exposing deep fakes using inconsistent head poses, in: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2019, pp. 8261–8265.

[9] S. Sinitsa, O. Fried, Deep image fingerprint: Towards low budget synthetic image detection and model lineage analysis, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2024, pp. 4067–4076.

[10] F. Furger, L. Amruthalingam, A. Navarini, M. Pouly, Applications of generative adversarial networks to dermatologic imaging, in: F.-P. Schilling, T. Stadelmann (Eds.), Artificial Neural Networks in Pattern Recognition, Springer International Publishing, Cham, 2020, pp. 187–199.

[11] W. Tang, D. Figueroa, D. Liu, K. Johnsson, A. Sopasakis, Enhancing fingerprint image synthesis with gans, diffusion models, and style transfer techniques, arXiv preprint arXiv:2403.13916 (2024).

[12] H. Montenegro, P. Neto, C. Patrício, I. Rio-Torto, T. Gonçalves, L. F. Teixeira, Evaluating privacy on synthetic images generated using gans: Contributions of the vcmi team to imageclefmedical gans 2023, Challenge (2023) 8.

[13] D. Subburam, S. M. SathyaNarayanan, B. Anand, K. Srinivasan, M. Subramaniam, Dmk-ssn at imageclef 2023 medical: Controlling the quality of synthetic medical images created via gans using machine learning and image hashing techniques (2023).
[14] M. M. Ghazi, M. M. Ghazi, Gan-isi: Generative adversarial networks image source identification using texture analysis (2023).
[15] H. Bharathi, A. Bhaskar, V. Venkataramani, K. Desingu, L. Kalinathan, Correlating biomedical image fingerprints between gan-generated and real images using a resnet backbone with ml-based downstream comparators and clustering: Imageclefmed gans, 2023 (2023).
[16] A.-G. Andrei, B. Ionescu, Aimultimedialab at imageclefmedical gans 2023: determining "fingerprints" of training data in generated synthetic images, in: CLEF2023 Working Notes, CEUR Workshop Proceedings, Thessaloniki, Greece, 2023.
[17] M. Šarić, M. Russo, M. Stella, M. Sikora, Cnn-based method for lung cancer detection in whole slide histopathology images, in: 2019 4th International Conference on Smart and Sustainable Technologies (SpliTech), IEEE, 2019, pp. 1–4.
[18] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, R. M. Summers, Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning, IEEE transactions on medical imaging 35 (2016) 1285–1298.