Task-Driven Convolutional Tuberculosis Detection Model with Low-Dose X-Ray

Adrian McIntosh  
*Department of Electrical Engineering  
University of Cape Town*Cape Town, South Africa   
[adrianmcintosh@myucla.edu](mailto:adrianmcintosh@myucla.edu)

Dan Ruan, PhD   
*Department of Radiation Oncology   
David Geffen School of Medicine, UCLA*Los Angeles, United States   
[DRuan@mednet.ucla.edu](mailto:DRuan@mednet.ucla.edu)

*Abstract*—Balancing dose and image quality is a prevalent trade-off in medical imaging with ionizing radiation. Therefore, it is highly desirable to leverage modern algorithms to push the frontier of As Low As Reasonably Achievable (ALARA) principle. To achieve this goal with low-dose tuberculosis screening, we propose a tailored approach for task-driven image denoising and enhancement.

# Introduction

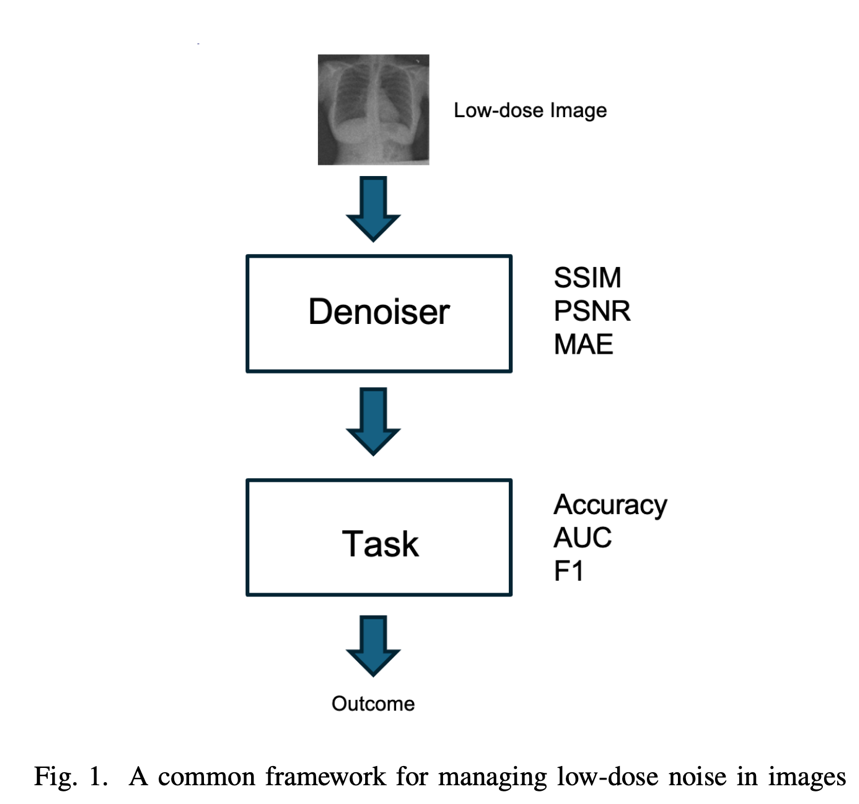
The resulting decrease in quality resulting from the use of low-dose images, often leads to reduced performance in tasks such as classification and segmentation. As shown in Fig. 1, a common strategy is to process the images using a denoising algorithm (’denoiser’) and feed the resulting denoised images into the task, whether this be a manual or automatic process.

Fig. 1: A common framework for managing low-dose noise in images

However, these denoisers are commonly assessed using certain image quality metrics (including SSIM, PSNR and MAE, for example) while the ”task” is optimized for different metrics related directly to the overall goal of the model (such as task-based accuracy, F1 score and AUC). Thus, choosing a denoiser based on these standard image quality metrics, may not align with the denoiser that would produce the best results in terms of the task.

The purpose of this paper is to further investigate this misalignment. The goals of this paper are described as follows:

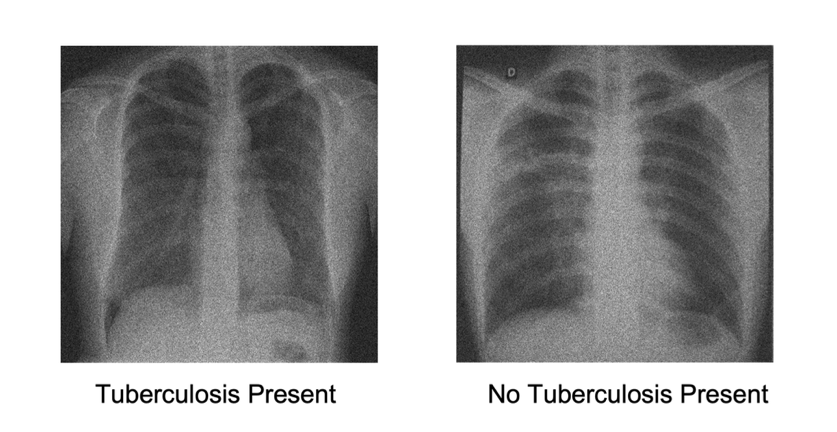
1. Quantify the misalignment between standard image quality metrics and task-based metrics: This was done by setting up various denoisers paired with classifiers, assessing and comparing these metrics in each case. The goal is not to show that the standard metrics always misalign with the task, but rather that there are tasks and situations where they do.
2. Design a system that accommodates for the mis- alignment between quality and task-based metrics: This was done using an auto-encoder-based architecture, combining the ”denoising” step and the ”task” or in this case, classification step.

The resulting architecture involves an end-to-end system for image-based TB detection, incorporating noise as a feature to the input, in contrast to sequential image-quality-driven denoising as a preprocessing step. The system would be appli- cable in the contexts where cheaper (lower dose) machinery or with limited power supplies, but still producing results of similar accuracy.

While this work works with X-Ray in particular, the system could easily be applied to other imaging modalities, such as Computed Tomography.

# Methodology

## A. Data set & Problem Setup

The data set used in this project comes from [1], examples of which are found in Fig. 2. It consists of about 4000 Chest X-ray images, a quarter of which depict X-ray projections with Tuberculosis, and the rest are Healthy (or non-Tuberculosis) which are labelled accordingly. The task in this context was thus classification. 80% of the images were used for training, and 20% were used for testing.

A set of starter code found in [2] was used as a reference for the initial preprocessing and classification framework for the images. With this, the specific preprocessing steps were adjusted (including noise simulation and setting up the de- noising).

Fig. 2 Example Images from the Dataset

## B. Simulating the Noise

For the project, simulated noise was applied to Chest X-ray images using the Poisson-Gaussian Mixture Model, described in [3] and [4]. The model was used to simulate noise for a similar purpose in [4] and the function used to implement this model was adapted from that work.

The parameters were chosen to moderately high values in specified in the original paper. These were α = 32 and β = 0.1 were chosen in Equations 3 and 4 in [4].

## C. Overall Framework

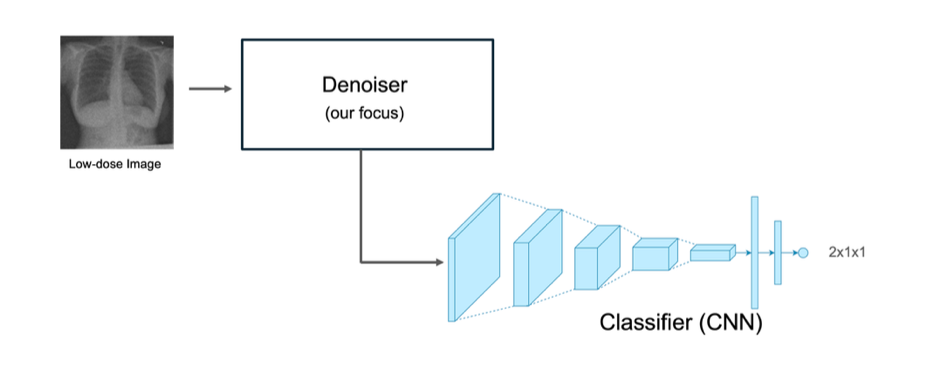
The framework for the experiment used can be described in two parts that the images are moved through as they are processing, namely, the Denoiser and the Classifier. This is shown in Fig. 3.

Fig. 3 General Framework for Investigation

The Denoiser consists of preprocessing specifically aimed at reducing the noise in the image, based on standard image metrics (for NLM and Wavelet) or alternatively focusing on the task-based metrics, in which cases this step is more closely combined with the Classifier.

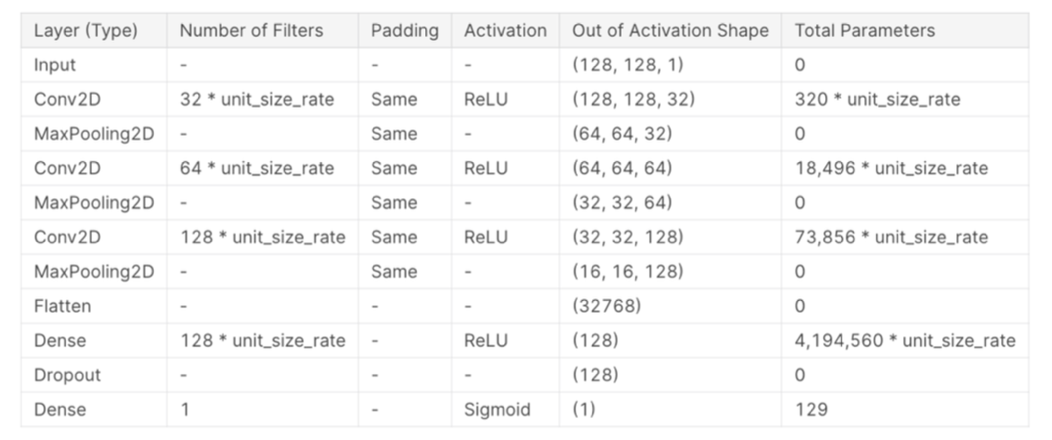
The Classifier step consists of a Convolutional Neural Network, trained on the output of the Denoiser step. The architecture of this Classifier was kept relatively consistent for each denoiser, except with slight variations when the dimensionality of the input to this step changed for a specific denoiser used.

TABLE I: Architecture of CNN Classifier

The general architecture of this Classifier was based on [2] and is shown in TABLE I. Other image preprocessing techniques applied to the images are described as follows:

**1) Resolution Standardization:** the images were converted to a standard resolution of 128x128 pixels.

**2) Data Type conversion:** the images were converted to NumPy arrays to save memory.

**3) Data Augmentation:** Due to the imbalanced proportions of TB and Normal on the dataset, a more even distribution was created by adding scaled and rotated versions of the TB images to the dataset.

## D. Baseline & Performance Metrics

Two models for comparison were used. The first consisted of the Classifier trained and tested on the simulated low-dose images (with the applied noise) and the second consisted of the Classifier trained and tested on the original high-quality images without the simulated noise. The accuracy results for these two models were 0.903 and 0.954 respectively.

The standard image quality metrics chosen for investigation are PSNR, SSIM and Mean Absolute Error. Where relevant these metrics were calculated and saved for the denoisers.

## E. Denoisers

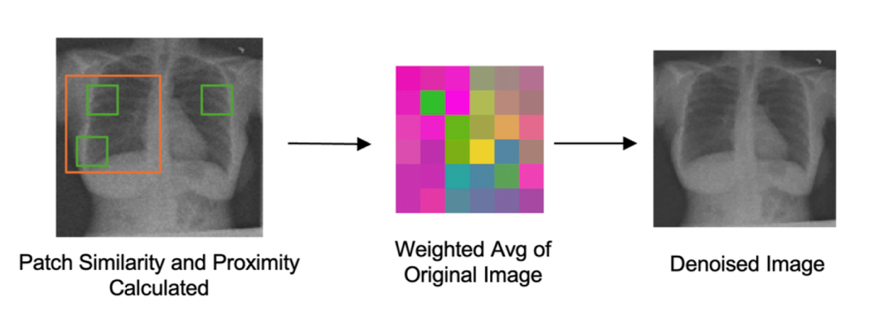
Several different Denoisers (denoising algorithms or pre- processing steps) were used, assessed and compared. These include the Non-Local Means Algorithm, Wavelet Denoising, 2D Fast Fourier Transform and the Adapted Autoencoder Model.

Fig. 4 Non-Local Means Denoising

1. ***Non-Local Means:***

This algorithm, described in [5] essentially works by separating the image into pixel centers and their surrounding “patches”, and comparing nearby and similar patches. Each output pixel is then calculated as a weighted average of the image’s original pixel values, with higher weights associated with similarity and proximity (to the corresponding pixel in the original image). Combining these output pixels results in a ”denoised” version of the input image. A flowchart explanation is shown in Fig. 4.

1. ***Wavelet Denoising:***

This algorithm works by computing the wavelet transform of an image, applying soft-thresholding, to the wavelet coefficients, and then computing the inverse wavelet transform to get the denoised image [6].

The noise level of the image is first computed using the Mean Absolute Deviation (MAD) method provided by the skimage library [7]. Using the VisuShrink method and dividing the estimated noise level, σ, by 4 (to avoid over-aggressive denoising), soft-thresholding is applied. VisuShrink uses a universal threshold that is applied to all coefficients and is based on the noise level provided.

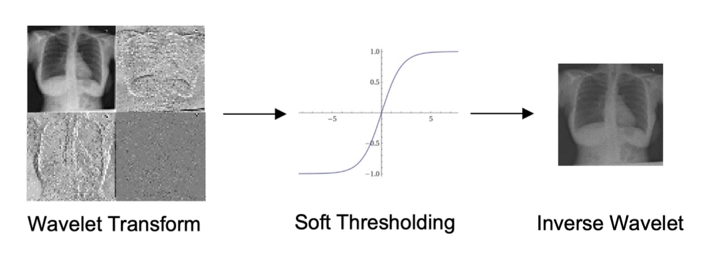


Fig. 5 Wavelet Denoising

1. ***Two-Dimensional Fast Fourier Transform (2DFFT):***

Having worked with the Wavelet Denoiser, the use of representational domains is was shown to be useful. The 2DFFT version of the Denoiser step involved computing the two- dimensional Fast Fourier Transform of the images and applying the resulting representation (in the form of an image) as the input to the Classifier. An example of this is shown in Fig. 6. In this case, the ’Denoiser’ step is more of a preprocessing step that is used to represent the data in a different form rather than an algorithm created with the specific intention of denoising.

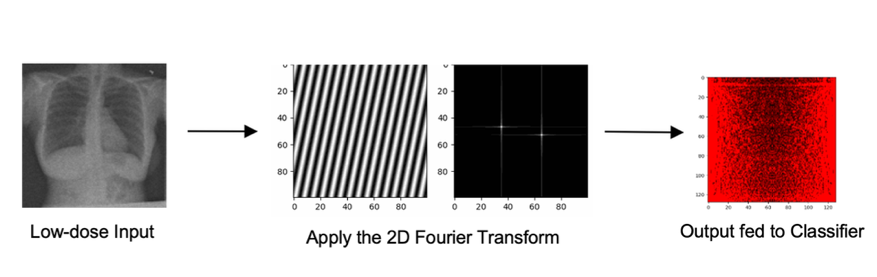


Fig. 6 2DFFT 'Denoiser'

1. ***Auto-encoder:***

An auto-encoder is a type of neural network typically used for compression, but it can also be used for denoising. It consists of a shallow CNN with the same images used as input and output. The hidden layers of the network have a lower dimensionality and consist of a set of weights that represent a summary of the image, and contain the most essential information.

When used for denoising, an auto-encoder is set up similarly, with the low-dose images as both input and output. A diagram representing this is shown in Fig. 7. Note that this diagram shows a fully connected feed-forward network, as a simplified diagram - a CNN was used in our case. In the general setup as shown in Fig. 3, the auto-encoder output layer will be used to output the denoised image. However, in our case, the lower-dimensional hidden layer is used as input to the CNN. This is shown in Fig. 8. Using this framework, the noise (and information embedded into the noise) is not smoothed out or removed, but rather incorporated into the classification model.

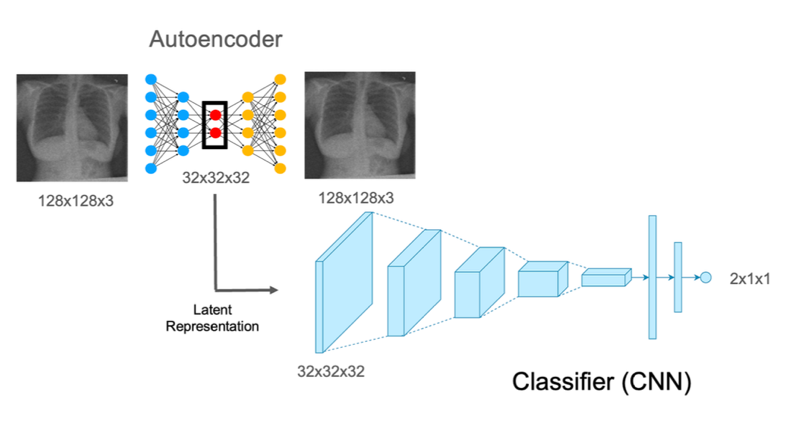
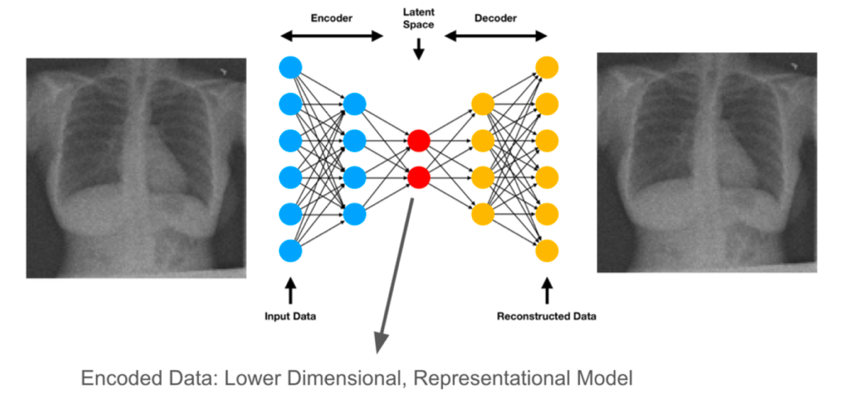


Fig. 7 Auto-encoder Model

Fig. 8 Adapted Auto-encoder

# Results & Discussion

Each of the denoisers was trained and tested using the simulated low-dose images. The standard image quality metrics as well as the task-based metrics were calculated for each case.

## A. Standard Image Quality Metrics vs Task-based Metrics

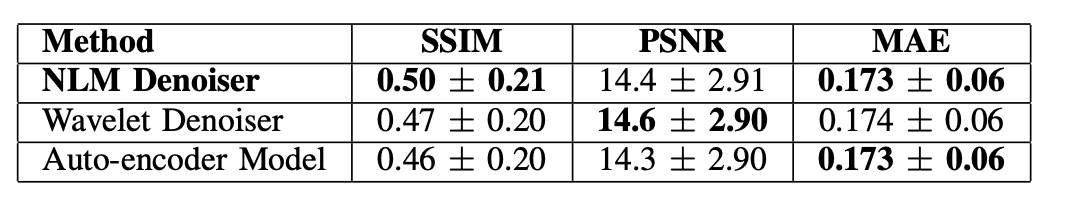
The average SSIM, PSNR and MAE were calculated for the three relevant denoisers. The results of these are shown in TABLE II. These values are calculated as an average over the training dataset. Note that the images used for this computation for the auto-encoder were the unprocessed simulated low-dose images fed into the overall framework.

TABLE II Standard Image Quality Metrics for Different Denoisers

According to these metrics, the best-performing model would be the NLM Denoiser, getting the best, second-best and joint-best scores for SSIM, PSNR and MAE respectively. Similarly, the second-best performing model would be the Wavelet Denoiser and the worst of these three would be the Auto-encoder Model.

A table with numbers and letters

Description automatically generatedThe task-based metrics were assessed using the classification performance of each model on the test set. The results of this is shown in TABLE III.

TABLE III Task-Based Metrics for Different Denoisers

Based on these metrics, the best model would be the Wavelet Denoiser, the second-best the Auto-encoder and the worst was the NLM Denoiser.

With these results, the misalignment between task-based and standard image quality metrics is demonstrated.

## B. Task-based Metrics & Model Design

The overall task-based results for each model were calculated based on their task-based accuracy. The classification results are shown in TABLE IV.

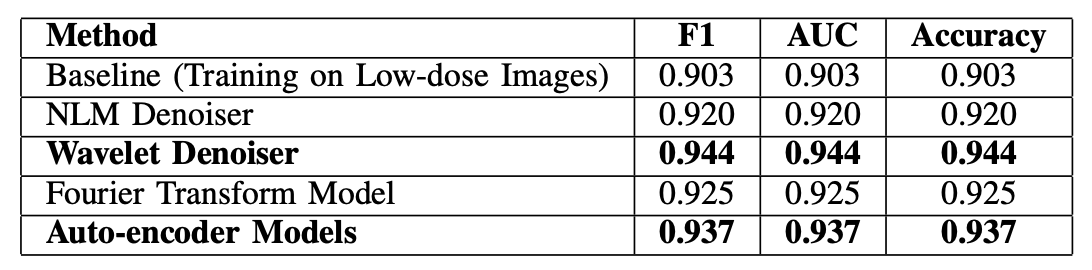


TABLE IV Task-based Metrics for All Methods

The best-performing model was the Wavelet Denoiser, with the Auto-encoder coming in at a close second. Interestingly, the top 3 performing models, all involved the use of a transform or representational version of the original image. The performance of the auto-encoder model suggests that this focus on task-based metrics and design is promising for the purpose of accommodating low-dose image quality issues.

# Conclusion

Both goals of the paper, outlined in the Introduction, were achieved. They are reiterated and described below:

## A. Main Objectives

1. **Quantify the misalignment between standard image quality metrics and task-based metrics:**

The misalignment between TABLE II and TABLE III successfully demonstrates this misalignment. Although the same results may not be found for every task, the purpose was to demonstrate that the standard image quality metrics do not work for every task, which is shown here. Seeing that the task chosen for this paper was binary classification, which is relatively simple for the dataset provided, the range and differences in task-based metrics were limited to 5% (between the worst-performing low-dose base model 0.903 and the classification model based on the clean images 0.954)

1. **Design a system that accommodates for the misalignment between quality and task-based metrics:**

The results shown in TABLE IV demonstrate that the Auto-encoder model performed very closely (0.937) to the overall best-performing Wavelet Denoiser model (0.944). Furthermore, this was very close to the best possible result for this task, 0.954, which was the classification model trained and tested on high-dose clean images (no simulated noise).

## B. Future Developments

Further investigation of these standard image quality metrics is necessary. We would like to apply this paradigm to the following:

1) Other Imaging Modalities   
An example of this would be the application of Computed Tomography (CT).

2) Extending to Other Tasks  
Segmentation (in the CT modality for example) could be another task that is investigated.

# REFERENCES

[1] S. Jaeger, S. Candemir, S. Antani, Y.-X. J. Wa ́ng, P.-X. Lu, and G. Thoma, “Two public chest x-ray datasets for computer-aided screening of pul- monary diseases,” *Quantitative imaging in medicine and surgery*, vol. 4, no. 6, p. 475, 2014.

[2] P. Janthawee, “Tuberculosis detection cnns with lime,” https://www.kaggle.com/code/pisitjanthawee13/tuberculosis-detection- cnns-with-lime, accessed: 2024-08-05.

[3] S. Lee, M. S. Lee, and M. G. Kang, “Poisson-gaussian noise analysis and estimation for low-dose x-ray images in the nsct domain,” *Sensors*, vol. 18, no. 4, p. 1019, 2018.

[4] N. Ba ̈hler, M. E. Helou, Objois, K. Okumus ̧, and S. Su ̈sstrunk, “Pogain: Poisson-gaussian image noise modeling from paired samples,” *IEEE Signal Processing Letters*, vol. 29, pp. 2602–2606, 2022.

[5] A.Buades,B.Coll,andJ.-M.Morel,“Non-localmeansdenoising,”*Image Processing On Line*, vol. 1, pp. 208–212, 2011.

[6] A. Halidou, Y. Mohamadou, A. A. A. Ari *et al.*, “Review of wavelet denoising algorithms,” *Multimed Tools Appl*, vol. 82, pp. 41 539–41 569, 2023.

[7] S. van der Walt, J. L. Scho ̈nberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, T. Yu, and the scikit-image contributors, “scikit-image: image processing in python,” *PeerJ*, vol. 2, p. e453, jun 2014. [Online]. Available: https://doi.org/10.7717/peerj.453