

Detection of Pneumonia in Chest X-Rays Using Generative Adversarial Networks

Sasane Sahil Sanjay¹, Wadgave Uttkarsh Ashok¹, Devika Sudhakar¹, Kale Pallavi Rohidase¹ and Dr.Satishkumar L Varma¹

¹PCE New Panvel, Navi Mumbai

Abstract

Pneumonia, an infection affecting the lungs, requires an accurate and timely diagnosis for effective treatment. However, detecting pneumonia in medical images presents challenges due to varying image qualities and subtle patterns, especially in low-clarity conditions influenced by inflammation and disease severity. One major challenge faced in this domain is the lack of datasets with sufficient training images. To address this challenge, this project proposes using Generative Adversarial Networks (GANs) to augment the training datasets by generating additional input images. GANs are particularly useful for this task as they can enhance image clarity and detect pneumonia-specific patterns through the use of a generator and discriminator. In this paper, the objective is to design and develop Generative Adversarial Networks for the detection of pneumonia using X-ray images by identifying image modalities and lung segmentation techniques used for detection of pneumonia. The influence of chest X-ray image enhancement and image augmentation by validating and evaluating network performance will be analyzed to classify the pattern of diffuse opacities (usually the result of fluid, damaged tissue, or inflammation) in a chest X-ray into alveolar or interstitial infiltration differentiating bacterial and viral pulmonary pneumonia using GANs. The proposed system will be developed using Python programming language and deep learning techniques. The experiments will focus on augmenting medical images and evaluating the performance of the model for pneumonia detection. This approach has the potential to improve the accuracy and reliability of pneumonia detection, ultimately benefiting patient care and outcomes.

Keywords: Pneumonia, pediatric pneumonia, Deep Learning, GAN, X-ray.

Corresponding author: E-mail address:

DOI:

Received: **Revised:** **Accepted:** **Published:**

1. Introduction

Pneumonia is a common and potentially life-threatening lung infection, with millions of deaths annually, particularly in children under five and the elderly. Current diagnostic methods, such as chest X-rays and CT scans, can be challenging to interpret and time-consuming. There is a need for an efficient and accurate diagnostic tool to assist healthcare professionals in detecting pneumonia. Deep learning (DL) techniques, including generative adversarial networks (GANs), have shown promise in medical imaging for disease detection. GANs can generate synthetic images that closely resemble real images, aiding in the diagnosis of pneumonia.

The high-level GAN architecture for pneumonia detection consists of multiple components, including a generator and a discriminator. The generator generates synthetic images of lung diseases, while the discriminator distinguishes between real and synthetic images. The architecture aims to improve the accuracy of pneumonia detection using generative adversarial networks.

2. Literature Survey

In recent years, considerable progress has been achieved in using Generative Adversarial Networks (GANs) for pneumonia detection in chest X-ray images. However, these advancements have also brought to light certain limitations and research gaps, emphasizing the crucial need for an advanced GAN framework tailored for automated pneumonia detection, such as the one proposed in this study.

The techniques in this category (GANs) are gaining traction for medical image analysis tasks like pneumonia detection in chest X-rays. These techniques offer the ability to generate synthetic data, potentially improving the performance of deep learning models.

Joby et al. [1] propose PneumoGAN, a Generative Adversarial Network (GAN)-based system for detecting pneumonia in chest X-rays. It addresses the challenge of limited training data by generating synthetic chest X-ray images depicting pneumonia. PneumoGAN achieved high accuracy (87

Ali et al. [2] propose a method for pneumonia detection in chest X-rays that tackles class imbalance. They use GANs to generate synthetic pneumonia X-rays and balance the dataset with undersampling. Transfer learning with pre-trained CNNs achieves high accuracy for pneumonia detection.

Saman Motamed et al. [3] propose a data augmentation technique called Inception-Augmentation GAN (IAGAN) specifically designed to improve the performance of Generative Adversarial Networks (GANs) used for anomaly detection in chest X-rays. IAGAN leverages both a noise vector and encoded real chest X-ray images as input to generate new, realistic X-rays. This allows IAGAN to learn the distribution of images within a specific class and generate new examples even for classes with limited labeled data. IAGAN achieved high accuracy (80

Jin et al. [4] propose using Conditional CycleGANs to generate realistic images depicting pneumonia progression in chest X-rays. This method offers advantages like improved disease visualization for patients and doctors and potential for data augmentation and clinical algorithms. However, the paper would benefit from comparisons with existing techniques, addressing data limitations, and incorporating metrics to assess generated image quality. Overall, this research builds on the use of GANs in medical imaging and shows promise for enhancing chest X-ray analysis.

Venu et al. [5] propose using Deep Convolutional Generative Adversarial Networks (DCGANs) to generate realistic chest X-ray images and address the limitations of small medical image datasets. DCGANs consist of a generator network creating images from noise and a discriminator network classifying real or generated images. Training these networks together improves the generator's realism and the discriminator's accuracy. The study showed the generated images resembled real data and a CNN classifier trained on the augmented dataset outperformed one trained on the original data, highlighting the potential of DCGANs for improving medical image analysis..

Ng et al. [6] used GANs to create synthetic COVID-19 chest X-ray images for training deep learning models when real data is scarce. They compared DCGAN and WGAN-GP, finding both generated realistic images. DCGAN performed well with just 1000 images. They noted DCGAN's training instability after 700 epochs and WGAN-GP's slightly lower FID scores. Augmenting real data with GAN-generated images improved a COVID-19 image classifier's performance, showing GANs' potential in medical image analysis for data-limited scenarios.

Yadav et al. [7] introduced Lung-GANs, a deep learning framework for unsupervised lung disease classification using chest X-ray and CT scan data. By leveraging Generative Adversarial Networks (GANs), Lung-GANs can learn from unlabeled data, addressing the limitations of expert interpretation and the need for large labeled datasets. The model achieves high classification accuracy (94.1

Motamed et al. [8] address the challenge of using GANs for one-class classification in pneumonia detection from chest X-rays. They propose Multi-class GAN (MCGAN), which utilizes limited labeled data from both classes during training to discourage the generation of images resembling the "other" class. Experimental results show MCGAN outperforming traditional GANs in classifying similar image classes, especially with limited labeled data for the second class. While the paper focuses on a specific challenge and lacks exploration of broader applications of MCGAN, it provides a promising solution for improving one-class classification performance in medical image analysis.

Table 1. Summary of Literature Survey on Image Enhancement Techniques

Literature	Techniques used for Image	Simple GAN	DCGAN	Pneumo GAN	Cycle SGAN	IAGAN
J. Joby et al. 2021 [1]	-	-	Yes	-	-	-
Ali et al. 2023 [2]	-	Yes	-	-	-	Yes
S. Motamed et al. 2021 [3]	-	-	-	-	-	Yes
S. Motamed et al. 2021 [9]	-	-	-	Yes	-	-
Y. Jin et al. 2023 [4]	-	-	Yes	-	-	-
Kora Venu et al. 2021 [5]	Yes	Yes	-	-	-	-
Ng et al. 2023 [6]	Yes	-	-	-	-	Yes

3. System Methodology

The process involves several key steps. A dataset of chest X-ray images with corresponding labels indicating pneumonia status is collected and preprocessed for consistency and diversity. Next, a GAN architecture comprising a generator network and a discriminator network is designed, with the generator producing synthetic chest X-ray images and the discriminator distinguishing between real and fake images. The GAN is then trained using adversarial training techniques on a split dataset, with adjustments made based on validation set performance. Features are extracted from the trained GAN, and a classification model, such as a CNN, is built on top of these features for pneumonia detection. The system's performance is evaluated on a separate test set, and optimizations are made for deployment, including speed and resource considerations.

3.1. System Architecture :

The architecture builds upon an existing framework by Lin et al., enhancing it to output images of size 512×512 . This enhancement

involves adding an extra transpose convolutional layer in the generator and a convolutional layer in the discriminator. The improved architecture allows for learning multiscale spatial information from higher-resolution images, leading to better results compared to existing methods.

3.1.1. Generator Architecture:

The generator architecture begins with an input of a 100-dimensional latent noise vector drawn from a uniform distribution, which is then reshaped by a dense layer into a $4 \times 4 \times 1024$ tensor. This tensor is processed through seven transpose convolutional layers, each designed to upsample the feature maps. ReLU activation functions are applied to all layers except the output layer, which uses a hyperbolic tangent (tanh) activation to constrain the output to the range [-1, 1]. Batch normalization is applied after each deconvolutional layer (except the last one) to stabilize and accelerate the training process. The use of kernel size and stride of 4 and 2, respectively, helps avoid uneven overlaps in the upsampling process, ensuring the generation of a single high-resolution $512 \times 512 \times 3$ image as the output.

3.1.2. Discriminator Architecture:

The discriminator architecture is based on a convolutional neural network (CNN) design, accepting both real and synthesized images with a shape of $512 \times 512 \times 3$ as input and outputting a probability value indicating the likelihood of an input image being real. Each convolutional layer incorporates batch normalization and LeakyReLU activation to enhance training stability and model performance. The model utilizes stridden convolutions for spatial downsampling, progressively reducing the dimensions of the input image. Additionally, 4×4 max pooling is applied on the last third convolutional layer, followed by 2×2 max pooling on the last second layer. The final convolutional layer employs the identity function to construct feature maps, which are then concatenated and flattened to be fed into a single sigmoid output for binary classification.

3.1.3. Training Details:

The training and implementation details of the proposed architecture involve the use of the TensorLayer library for deep learning and reinforcement learning tasks. The Adam optimizer is chosen for its efficiency in training neural networks. Hyperparameters such as momentum, learning rate, and batch size are set to 0.5, 0.0002, and 64, respectively, to optimize training performance. The network comprises approximately 29 million parameters and is trained on an Nvidia Tesla k80 GPU with 12 GB memory for 100 epochs. This setup ensures that the model is trained effectively and efficiently to achieve optimal performance.

3.2. Classifier:

The classifier architecture involves an ensemble approach using a stacked generalization. The base learners consist of random forest and linear support vector classification (SVC) models. The outputs of these base learners are then stacked together and used as input to a meta-classifier, which is implemented as logistic regression. The features used for classification are extracted from the multi-feature layer of the trained discriminator. Prior to training the SVC model, the input features are standardized to ensure consistency in the training process. This ensemble approach aims to combine the strengths of different classifiers to improve overall classification accuracy.

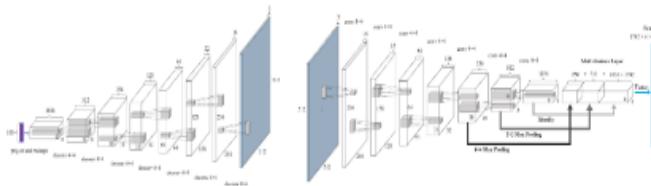


Figure 1. Existing system architecture used for Lung Disease Classification [7]

3.2.1. Proposed System Architecture:

[h] The objective of the proposed methodology is to further advance the generation of high-resolution lung disease images using Generative Adversarial Networks (GANs). Unlike simpler applications, the complexity of the X-ray datasets requires a more sophisticated approach. The proposed system aims to achieve this by enhancing the generator and discriminator with fine-tuning techniques. The generator will be fine-tuned to improve its mapping from random noise vectors to realistic images, while the discriminator will be fine-tuned to better differentiate between real and fake images, ultimately leading to the generation of more realistic high-resolution images.

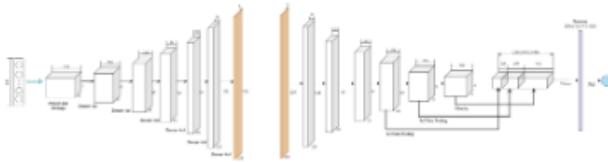


Figure 2. Proposed System Architecture for Lung Disease Classification

3.2.2. Generator Architecture:

The proposed generator architecture for the GAN is designed to output high-resolution 256×256×3 images of lung disease. It starts with a 100-dimensional latent noise vector drawn from a uniform distribution, which is reshaped by a dense layer into a 4 × 4 × 512 tensor. To improve image mapping, multiple layers are added for fine-tuning. Each deconvolutional layer upsamples the feature maps using a kernel size and stride of 4 and 2, respectively, to avoid uneven overlaps. ReLU activation functions are applied to all layers except the output layer, which uses a hyperbolic tangent (tanh) activation. Batch normalization is included after each deconvolutional layer, except the last one, to stabilize and accelerate training. Overall, this architecture is aimed at enhancing the generator's ability to produce realistic high-resolution images of lung disease.

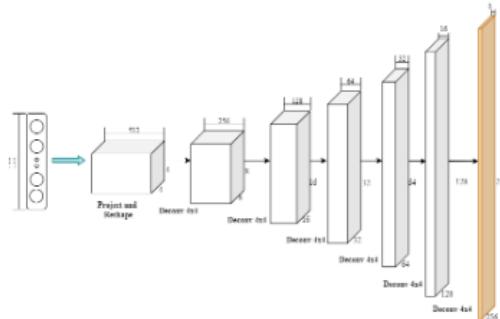


Figure 3. Generator Architecture

3.2.3. Discriminator Architecture:

[h] The proposed discriminator architecture is tailored for assessing the authenticity of high-resolution 256×256×3 lung disease images.

It accepts both real and synthesized images as input and outputs a probability value indicating the likelihood of the input image being real. The architecture utilizes a CNN structure with batch normalization and LeakyReLU activation functions to enhance learning and reduce overfitting. Strided convolutions are employed for spatial downsampling, allowing the model to learn its own downsampling process. Max pooling is used to construct feature maps, aiding in the identification of key image features. Finally, a sigmoid output layer is employed for binary classification, distinguishing between real and fake images.

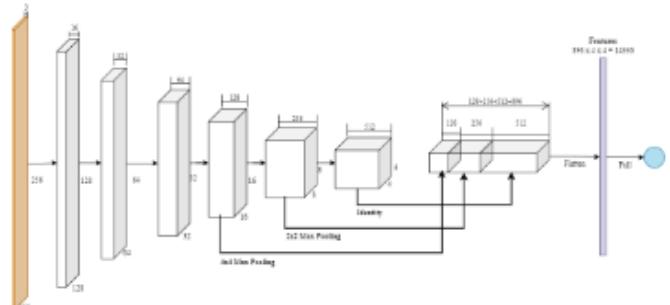


Figure 4. Discriminator Architecture

3.2.4. Training Details:

The training and implementation details of the proposed system for generating high-resolution 256×256 lung disease images involve the use of the PyTorch for deep learning tasks. The Adam optimizer is employed for efficient optimization of the network, with hyperparameters set to a momentum of 0.5, a learning rate of 0.0002, and a batch size of 64. The network comprises approximately 20 million parameters and will be trained on an Nvidia GPU T4 x2 of 16 GB memory for 100 epochs. These specifications ensure effective training and optimization of the generator and discriminator models for generating high-quality lung disease images.

3.2.5. Classifier Architecture:

The classifier architecture will consist of an ensemble of classifiers using a stacked generalization. The base classifiers will include LightGBM and XGBoost, with the meta-classifier being logistic regression. Features from the multi-feature layer of the trained discriminator will be used for classification, with input features standardized prior to training the classifiers. This ensemble approach aims to improve the classification accuracy of lung disease images generated by the proposed GAN architecture.

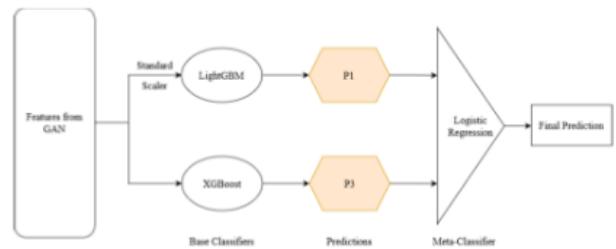


Figure 5. Proposed Classifier

4. System Implementation

The implementation of a Generative Adversarial Network (GAN) for pneumonia detection involves utilizing PyTorch for efficient implementation on NVIDIA GPUs. The GAN architecture consists of a

generator and discriminator model, trained on lung disease images preprocessed to a consistent size of 256×256 pixels and normalized pixel values. Training involves using the Adam optimizer and adjusting hyperparameters for optimal performance, with the generator trained to minimize the discriminator's ability to differentiate between real and fake images. Regularization techniques like dropout layers, batch normalization, and early stopping are employed to prevent overfitting.

Evaluation metrics such as accuracy, precision, recall, and F1-score are calculated on separate validation and test datasets. Fine-tuning and optimization are performed based on evaluation results, with the trained GAN models deployed for generating synthetic lung disease images and integrated into healthcare systems or research projects. Comprehensive testing and documentation ensure reliability, adherence to ethical guidelines, and responsible deployment to production environments.

4.1. Technique

This technique focuses on enhancing the resolution and quality of lung disease images through the implementation of Generative Adversarial Networks (GANs). The process begins with the preprocessing of the dataset, ensuring all images are of a consistent size (256×256 pixels) and pixel values are normalized for optimal performance. The dataset is then split into training, validation, and testing sets to train and evaluate the GAN models.

During training, the generator is optimized to minimize the discriminator's ability to differentiate between real and synthetic images, while the discriminator is optimized to better differentiate between the two. This adversarial training process iterates until the models achieve a satisfactory level of performance.

Evaluation of the trained models is conducted using various metrics including accuracy, precision, recall, and F1-score to assess the quality of generated images. Fine-tuning and optimization techniques are applied based on evaluation results to further improve the models' performance.

Overall, this technique leverages GANs to generate high-resolution lung disease images, with a focus on preprocessing, model architecture, training, and evaluation to achieve the desired outcome.

4.1.1. Use Case Diagram / Activity Diagram

Fig. 3.3 Proposed GAN (Generator and Discriminator) The diagram illustrates the architecture of a GAN, with the generator transforming random noise into fake images and the discriminator distinguishing between real and fake images. It shows the flow of data through the networks, highlighting the layers and connections involved in generating and evaluating images.

4.2. Sample Dataset Used

4.3. Evaluation Metrics

4.3.1. Accuracy:

Accuracy measures the proportion of correctly classified instances among all instances.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

(4.1)

4.3.2. Precision

Precision measures the proportion of true positive predictions among all positive predictions. It assesses the accuracy of positive predictions.

$$\text{Precision} = \frac{TP}{TP + FP}$$

(4.2)

Table 2. Sample Dataset Used for Experiment

Dataset	Count	Description
COVIDx CT	14,214	Two variants: "A" and "B". The "A" consists of cases with confirmed diagnoses (i.e., radiologist-confirmed).
ChestX-Ray8	112,120	Consists of 60% of all frontal chest X-rays. This dataset is comprehensive and widely used in various studies.
COVID-19 Radiography Database	13,808	Consists of 10,192 normal case images and 3,616 pneumonia case images. The database has been instrumental in research.
Labelled Optical Coherence Tomography (OCT) and Chest X-Ray Images for Classification - Mendeley Data	5,856	Contains 1,583 normal cases and 4,273 pneumonia case images. This diverse dataset supports various machine-learning tasks.
Extensive and Augmented COVID-19 X-Ray and CT Chest Images Dataset	17,099	Contains 5,500 non-COVID images and 4,044 COVID images. It provides a rich resource for training and testing models.

4.3.3. Recall (Sensitivity):

Recall measures the proportion of true positive predictions among all actual positive instances. It evaluates the ability of the model to find all positive instances.

$$\text{Recall} = \frac{TP}{TP + FN}$$

(4.3)

4.3.4. F1 Score:

The F1 Score is the harmonic mean of precision and recall. It provides a balance between precision and recall, considering both false positives and false negatives.

$$\text{F1 Score} = \frac{2 * (\text{Precision} * \text{Recall})}{\text{Precision} + \text{Recall}}$$

(4.4)

4.3.5. Area Under the ROC Curve (AUC-ROC):

The ROC Curve plots the True Positive Rate (TPR) against the False Positive Rate (FPR) at various threshold settings. The AUC-ROC is the area under the ROC Curve, which represents the model's ability to discriminate between positive and negative instances across different thresholds.

4.3.6. FID (Fréchet Inception Distance):

The FID is a metric used to evaluate the quality and diversity of images generated by GANs. It measures the distance between the feature representations of real and generated images in a pre-trained Inception model.

$$\text{FID} = \|\mu_{\text{Real}} - \mu_{\text{Fake}}\|^2 + \text{Tr}(C_{\text{Real}} + C_{\text{Fake}} - 2(C_{\text{Real}} \cdot C_{\text{Fake}})^{\frac{1}{2}})$$

(4.5)

4.4. DCGan

4.4.1. DCGan Model

Compares accuracy results of various DCGAN configurations, highlighting the differences in model performance. ReLU with RMSprop optimizer showed the highest accuracy of 95.4 using LGBM.

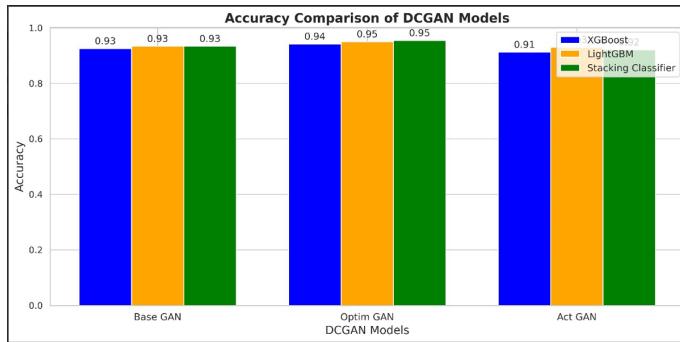


Figure 6. DCGAN Model

4.4.2. DCGAN Generated Image

[h] These generated images demonstrate DCGAN's ability to produce synthetic data that resembles real X-ray images, but the quality of these images is still slightly blurry, indicating room for improvement.

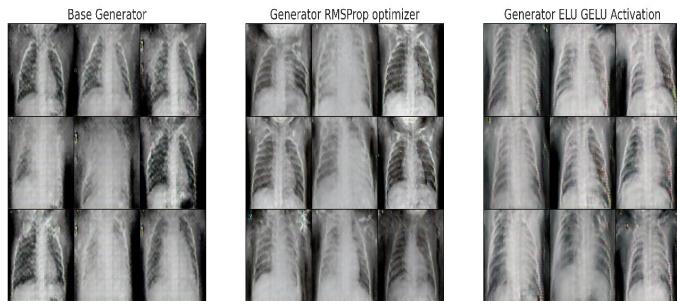


Figure 7. DCGAN Generated

4.4.3. DCGAN Loss

shows a graph comparing the loss function for various DCGAN configurations showing a gradual decrease in loss during training, but the fluctuations suggest that the DCGAN model struggles with convergence at certain points, leading to instability in the learning process.

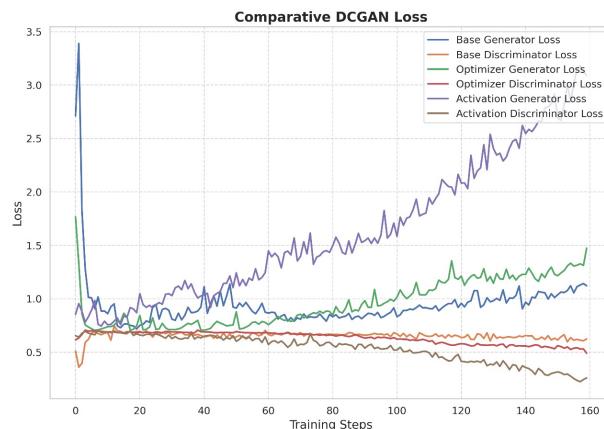


Figure 8. DCGAN Loss

4.4.4. DCGAN ROC curve

The ROC curve helps evaluate the overall performance of the DCGAN model at various classification thresholds. The area under the curve (AUC) serves as a metric for model performance, with a larger area indicating better performance.

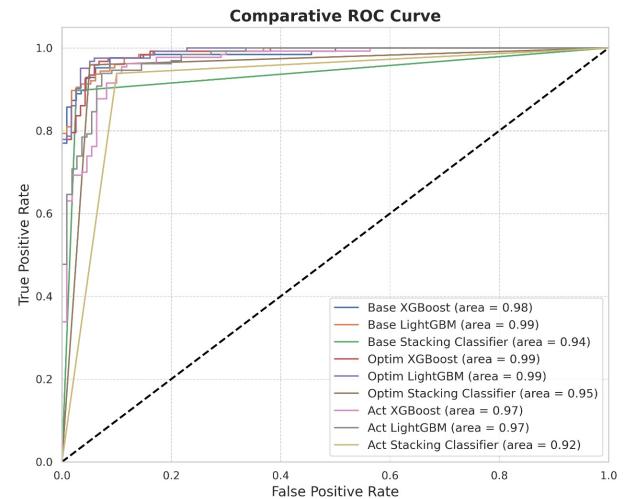


Figure 9. DCGAN ROC curve

4.4.5. DCGan Algorithm

Algorithm 1 This class includes the Mini Batch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k, is a hyperparameter. Here, k=1.

```
for number of training iterations do
    for k steps
        do
            • Sample minibatch of m noise samples {z(1), ..., z(m)} from noise prior pg(z).
            • Sample minibatch of m examples {x(1), ..., x(m)} from data generating distribution pdata(x).
            • Update the discriminator by ascending its stochastic gradient:
```

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^m [\log D(x^{(i)}) + \log (1 - D(G(z^{(i)})))]$$

```
end for
    • Sample minibatch of m noise samples {z(1), ..., z(m)} from noise prior pg(z).
    • Update the generator by descending its stochastic gradient:
```

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^m \log (1 - D(G(z^{(i)})))$$

```
end for
```

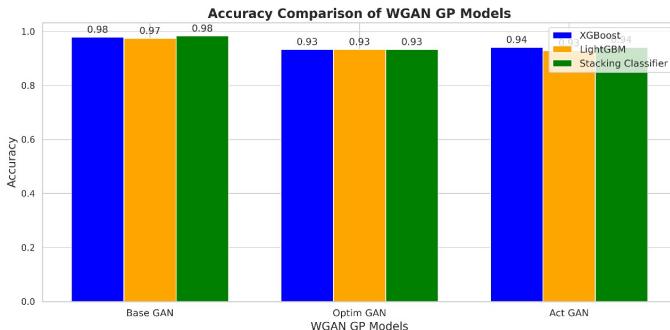
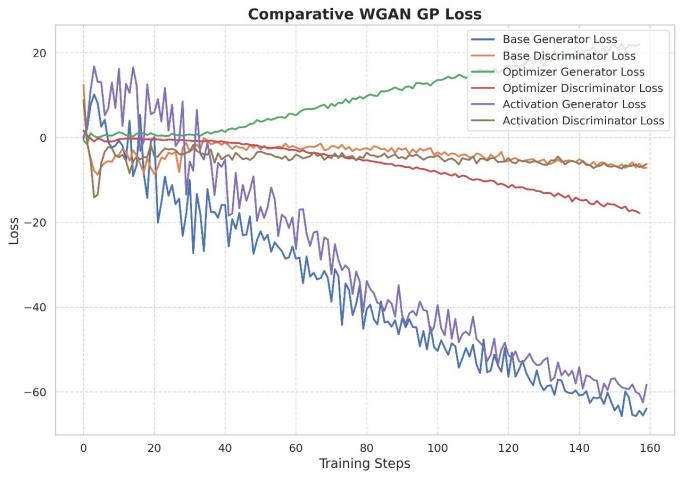
Table 3. DCGAN (Deep Convolutional GAN).

No.	Model	Activation	Loss function	Optimizer	Learning Rate	Batch size	Epoch	Classifier	n_estimators	Accuracy	Stacking Accuracy
1	DC GAN	ReLU, LeakyReLU	Binary Cross Entropy	Adam (Adaptive Moment Estimation)	0.0002	64	100	XGBClassifier	100	0.93	0.93
2	DC GAN	ReLU, LeakyReLU	Binary Cross Entropy	RMSprop	0.0002	64	100	XGBClassifier	100	0.942	0.954
3	DC GAN	GELU, ELU	Binary Cross Entropy	Adam (Adaptive Moment Estimation)	0.0002	64	100	XGBClassifier	100	0.91	0.921

4.5. WGan

4.5.1. WCGAN Model

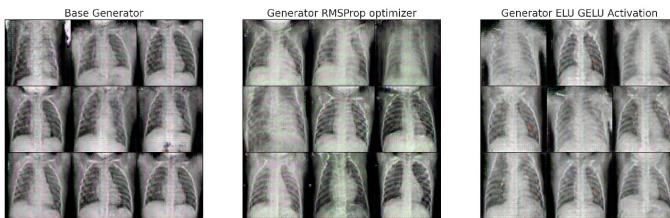
[h] Accuracy Comparison of WGAN GP Models, shows how different WGAN GP model configurations perform in terms of accuracy. WGAN GP model configuration using ReLU, LeakyReLU activation with Adam optimizer performs the best, achieving the highest accuracy of 98.3 using LGBM Classifier.

**Figure 10.** WCGAN Model**Figure 12.** WCGAN Loss

4.5.4. WGan Algorithm

4.5.2. WCGAN Generated Image

These generated images demonstrate DCGAN's ability to produce synthetic data that resembles real X-ray images, but the quality of these images is still slightly blurry, indicating room for improvement.

**Figure 11.** WCGAN Generated Image

4.5.3. WCGAN Loss

shows a graph that compares the loss function for various WGAN GP configurations, showing a steady decrease in loss during training. However, occasional fluctuations indicate that the WGAN GP models experience some instability in convergence, though overall performance improves with time.

Algorithm 2 This class includes the WGAN. All experiments in the paper used default values $\alpha = 0.00005$, $c = 0.01$, $m = 64$, $n_{\text{critic}} = 5$.

Require: α , the learning rate; c , the clipping parameter; m , the batch size; n_{critic} , the number of critic iterations per generator iteration.

Require: w_0 , initial critic parameters; θ_0 , initial generator parameters.

- 1: **while** θ_0 has not converged **do**
- 2: **for** $t = 0, \dots, n_{\text{critic}}$ **do**
- 3: Sample $\{x^{(i)}\}_{i=1}^m \sim P_r$, a batch from the real data.
- 4: Sample $\{z^{(i)}\}_{i=1}^m \sim p(z)$, a batch of prior samples.
- 5: $g_w \leftarrow \nabla_w \left[\frac{1}{m} \sum_{i=1}^m f_w(x^{(i)}) - \frac{1}{m} \sum_{i=1}^m f_w(g_\theta(z^{(i)})) \right]$
- 6: $w \leftarrow w + \alpha \cdot \text{RMSProp}(w, g_w)$
- 7: $w \leftarrow \text{clip}(w, -c, c)$
- 8: **end for**
- 9: Sample $\{z^{(i)}\}_{i=1}^m \sim p(z)$, a batch of prior samples.
- 10: $g_\theta \leftarrow -\nabla_\theta \frac{1}{m} \sum_{i=1}^m f_w(g_\theta(z^{(i)}))$
- 11: $\theta \leftarrow \theta - \alpha \cdot \text{RMSProp}(\theta, g_\theta)$
- 12: **end while**

Table 4. WGAN GP (Wasserstein GAN Gradient Penalty).

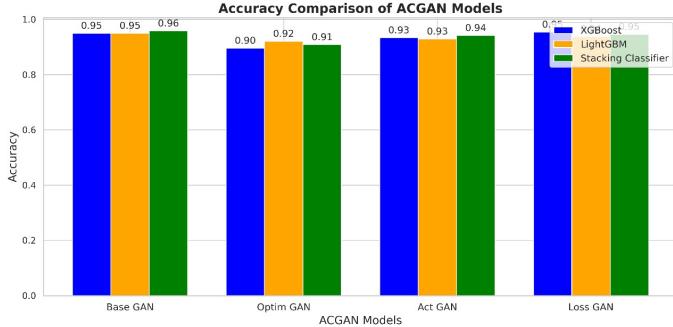
No.	Model	Activation	Loss function	Optimizer	Learning Rate	Batch size	Epoch	Classifier	n_estimators	Accuracy	Stacking Accuracy
1	WGAN GP	ReLU, LeakyReLU	Binary Cross Entropy	Adam (Adaptive Moment Estimation)	0.0002	64	100	XGBClassifier	100	0.93	0.93
2	WGAN GP	ReLU, LeakyReLU	Binary Cross Entropy	RMSprop	0.0002	64	100	XGBClassifier	100	0.942	0.954
3	WGAN GP	GELU, ELU	Binary Cross Entropy	Adam (Adaptive Moment Estimation)	0.0002	64	100	XGBClassifier	100	0.91	0.921

Note: Obtained from L^AT_EX tables [1].

4.6. Auxillary Gan

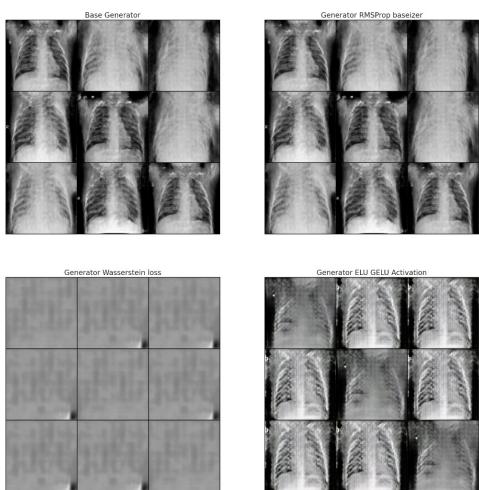
4.6.1. Auxillary Gan Model

Accuracy Comparison of ACGAN Models, shows how different ACGAN model configurations perform in terms of accuracy. ACGAN model configuration using ReLU, LeakyReLU activation with Adam optimizer performs the best, achieving the highest accuracy of 95.8 for XGB and LGBM classifiers.

**Figure 13.** Auxillary Gan Model

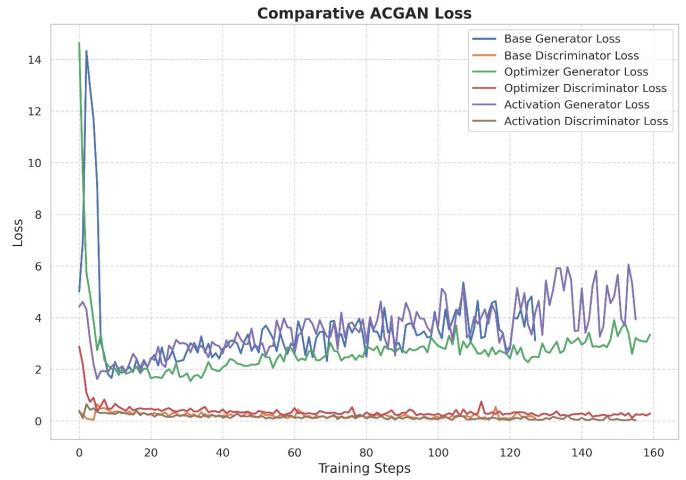
4.6.2. Auxillary Gan Generated Image

ACGAN models produce images that are more distinguishable between different classes of pneumonia, which is beneficial for medical image classification and diagnosis.

**Figure 14.** Auxillary Gan Generated Image

4.6.3. Auxillary Gan Loss

It highlights how the loss decreases steadily over time, indicating improved model performance, with fluctuations reflecting the challenges in maintaining stability between the generator and discriminator.

**Figure 15.** Auxillary Gan Loss

4.6.4. Auxillary Gan ROC

presents the ROC curve for the ACGAN model. This curve shows the model's ability to distinguish between two classes by plotting the true positive rate against the false positive rate.

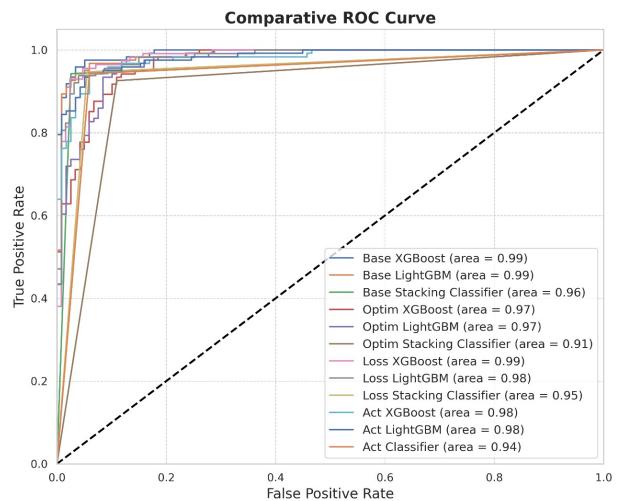
**Figure 16.** Auxillary Gan ROC

Table 5. Auxiliary Classifier GAN.

No.	Model	Activation	Loss function	Optimizer	Learning Rate	Batch size	Epoch	Classifier	n_estimators	Accuracy	Stacking Accuracy
1	DC GAN	ReLU, LeakyReLU	Binary Cross Entropy	Adam (Adaptive Moment Estimation)	0.0002	64	100	XGBClassifier	100	0.93	0.93
2	DC GAN	ReLU, LeakyReLU	Binary Cross Entropy	RMSprop	0.0002	64	100	XGBClassifier	100	0.942	0.954
3	DC GAN	GELU, ELU	Binary Cross Entropy	Adam (Adaptive Moment Estimation)	0.0002	64	100	XGBClassifier	100	0.91	0.921

[1].

Table 6. Comparison of different GAN architectures and their performance.

No.	Model	Activation	Loss Function	Optimizer	Learning Rate	Batch Size	Epoch	Classifier	Accuracy	Stacking Accuracy
1	DC GAN	ReLU, LeakyReLU	Binary Cross Entropy	RMSprop	0.0002	64	100	XGBClassifier	0.928	0.92
								LGBMClassifier	0.936	0.92
2	WGAN GP	ReLU, LeakyReLU	Wasserstein Loss	Adam	0.0002	64	100	XGBClassifier	0.944	0.952
								LGBMClassifier	0.960	0.952
3	ACGAN	ReLU, LeakyReLU	Binary Cross Entropy	Adam	0.0002	64	80	XGBClassifier	0.920	0.92
								LGBMClassifier	0.904	0.92

5. Conclusion

Lung diseases are a severe matter of concern all over the world. Early diagnosis is crucial to help in faster recovery and improve long-term survival rates. From the literature survey, we found that the limited size of datasets in medical imaging often leads to overfitting of training data, hampering detection performance. To address these challenges, we will utilize Generative Adversarial Networks (GANs) as a solution. The objective of this paper is to design and develop GANs for the detection of pneumonia using X-ray images by identifying image modalities and lung segmentation techniques for pneumonia detection.

6. References

- J. Joby, R. J. Raju, R. Job, S. M. Thomas and A. K. S, "PneumoGAN: A GAN based Model for Pneumonia Detection," 2021 2nd International Conference on Advances in Computing, Communication, Embedded and Secure Systems (ICACCESS), Ernakulam, India, 2021, pp. 102-106, doi: 10.1109/ACCESS51619.2021.9563341.
- Ali, Wardah, et al. "Pneumonia Detection in Chest X-Ray Images: Handling Class Imbalance." arXiv preprint arXiv:2301.08479 (2023). <https://doi.org/10.48550/arXiv.2301.08479>
- Saman Motamed, Patrik Rogalla, Farzad Khalvati, Data augmentation using Generative Adversarial Networks (GANs) for GAN-based detection of Pneumonia and COVID-19 in chest X-ray images, *Informatics in Medicine Unlocked*, Volume 27, 2021, 100779, ISSN 2352-9148, <https://doi.org/10.1016/j.imu.2021.100779>
- Y. Jin, W. Chang and B. Ko, "Generating Chest X-Ray Progression of Pneumonia Using Conditional Cycle Generative Adversarial Networks," in *IEEE Access*, vol. 11, pp. 88152-88160, 2023, doi: 10.1109/ACCESS.2023.3305994 Available <https://arxiv.org/abs/2011.13775>
- Kora Venu, S.; Ravula, S. Evaluation of Deep Convolutional Generative Adversarial Networks for Data Augmentation of Chest X-ray Images. *Future Internet* 2021, 13, 8. <https://doi.org/10.3390/fi13010008>
- Ng, M.F.; Hargreaves, C.A. Generative Adversarial Networks for the Synthesis of Chest X-ray Images. *Eng. Proc.* 2023, 31, 84. <https://doi.org/10.3390/ASEC2022-13954>
- P. Yadav, N. Menon, V. Ravi and S. Vishvanathan, "Lung-GANs: Unsupervised Representation Learning for Lung Disease Classification Using Chest CT and X-Ray Images," in *IEEE Transactions on Engineering Management*, vol. 70, no. 8, pp. 2774-2786, Aug. 2023, doi: 10.1109/TEM.2021.3103334.
- S. Motamed and F. Khalvati, "Multi-class Generative Adversarial Networks: Improving One-class Classification of Pneumonia Using Limited Labeled Data," 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Mexico, 2021, pp. 3817-3822, doi: 10.1109/EMBC46164.2021.9629980.
- S. Motamed and F. Khalvati, "Inception-GAN for Semi-supervised Detection of Pneumonia in Chest X-rays," 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Mexico, 2021, pp. 3774-3778, doi: 10.1109/EMBC46164.2021.9630473.
- Kermany, Daniel; Zhang, Kang; Goldbaum, Michael (2018), "Labeled Optical Coherence Tomography (OCT) and Chest X-Ray Images for Classification", Mendeley Data, V2, doi: 10.17632/rscbjbr9sj.2
- Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural*

- Information Processing Systems, pages 2672–2680, 2014.
- 12 S. Vashisht, S. Lamba, B. Sharma and A. Sharma, "Pneumonia Classification using CNN-GAN," 2023 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), Erode, India, 2023, pp. 456-461, doi: 10.1109/ICSCDS56580.2023.10104733.
 - 13 D. Srivastav, A. Bajpai and P. Srivastava, "Improved Classification for Pneumonia Detection using Transfer Learning with GAN based Synthetic Image Augmentation," 2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 2021, pp. 433-437, doi: 10.1109/Confluence51648.2021.9377062.
 - 14 V. Bhagat and S. Bhaumik, "Data Augmentation using Generative Adversarial Networks for Pneumonia classification in chest X Rays," 2019 Fifth International Conference on Image Information Processing (ICIIP), Shimla, India, 2019, pp. 574-579, doi: 10.1109/ICIIP47207.2019.8985892.
 - 15 D. Srivastav, A. Bajpai and P. Srivastava, "Improved Classification for Pneumonia Detection using Transfer Learning with GAN based Synthetic Image Augmentation," 2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 2021, pp. 433-437, doi: 10.1109/Confluence51648.2021.9377062.
 - 16 Muralikrishna Puttagunta, Ravi Subban, Nelson Kennedy Babu C, A Novel COVID-19 Detection Model Based on DCGAN and Deep Transfer Learning, Procedia Computer Science, Volume 204, 2022, Pages 65-72, ISSN 1877-0509, <https://doi.org/10.1016/j.procs.2022.08.008>.
 - 17 Putri, Kania Ardhani and Wikky Fawwaz Al Maki. "Enhancing Pneumonia Disease Classification using Genetic Algorithm-Tuned DCGANs and VGG-16 Integration." Journal of Electronics, Electromedical Engineering, and Medical Informatics (2023): n. Pag.
 - 20 Chandranegara, Didih Sari, Zamah Dewantoro, Muhammad Wibowo, Hardianto & Suharso, Wildan. (2023). Implementation of Generative Adversarial Network (GAN) Method for Pneumonia Dataset Augmentation. *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*. 10.22219/kinetik.v8i2.1675.