CliniScan: Lung-Abnormality Detection on Chest X-rays using AI

Kevin A Dobbin

TABLE OF CONTENTS

PROJECT DESCRIPTION & DATASET

Project Description

Medical deep learning models require clean, standardized, high-quality input data. However, raw medical images in DICOM format often contain variations in resolution, contrast, metadata, borders, or noise.

This project automates the preprocessing pipeline for chest X-ray DICOM images. It includes:

- **Automatic dataset scanning**
- **Visualization of sample medical images**
- **Image enhancement and denoising**
- **Cropping and padding for shape consistency**
- **Normalization and preparation for neural network models**
- **Annotation conversion to YOLO format**
- **Safe data augmentation for improving training performance**
- **Train-test-validation dataset splitting**

This ensures that every image sent into the model is consistent, normalized, and ready for training.

Dataset Used

The dataset consists of chest X-ray images stored in Digital Imaging and Communications in Medicine (DICOM) format. These contain both pixel data and patient metadata.

Characteristics of the dataset:

- **Medical imaging standard widely used in hospitals**
- **Single-channel grayscale images**
- **May contain variable resolution and exposure**
- **Useful for tasks like detection, classification, or segmentation**

ENVIRONMENT SETUP

This system uses Python with a medical image processing and deep learning environment.

Core Frameworks Installed

- **pydicom – used for reading `.dcm` medical images and extracting metadata**
- **OpenCV (cv2) – used for image resizing, denoising, and enhancement**
- **NumPy – used for array operations**
- **Matplotlib – used for visualization**
- **PyTorch + torchvision – used for tensor conversion and deep learning preprocessing**
- **Albumentations – used for image augmentation**
- **Pandas & Scikit-learn – used for dataset splitting and metadata handling**

This environment ensures a stable and standardized pipeline for processing medical imaging data.

DATA EXPLORATION

Automatic File Discovery

The system automatically scans the dataset directory and collects every file with:

- `.dcm`
- `.dicom`

This avoids the need for manual file listing and ensures that even nested directories inside hospital imaging folders are detected.

Image Visualization

A sample of the first 10 images is displayed to:

- **Verify data integrity**
- **Confirm that pixel values are readable**
- **View positioning metadata (e.g., PA, AP lateral views)**

This step helps validate that the input data is properly formatted and meaningful before continuing.

DATA PREPROCESSING

1. Loading DICOM Images

Each file is loaded using pydicom, extracting the image pixel array. Since DICOM stores 12-bit or 16-bit images, raw pixel values may vary greatly.

2. Image Normalization

Pixel intensity ranges differ between scanners and patients. To standardize this, Min–Max normalization converts every image to a scale of:

`0.0 → black`
`1.0 → white`

This ensures consistent brightness and contrast across all images.

3. Conversion to RGB

Deep learning models pre-trained on ImageNet expect 3-channel input. Medical scans are grayscale, so the single channel is duplicated to create:

`(H, W) → (H, W, 3)`

without modifying visual content.

4. Image Resizing

All images are resized to standardized dimensions (e.g., 512×512). This ensures:

- **Every image fits in the model**
- **Batch processing becomes efficient**
- **No dynamic shape handling is needed**

5. Tensor Conversion and ImageNet Normalization

After resizing:

- **The image is converted to a PyTorch tensor**
- **Standard ImageNet mean & standard deviation normalization is applied**

This prepares the data for modern CNN-based models and accelerates convergence.

6. Contrast Enhancement using CLAHE

Medical scans may appear low-contrast due to soft tissue density.
CLAHE (Contrast Limited Adaptive Histogram Equalization):

- **Enhances local contrast**
- **Prevents over-amplification of noise**
- **Improves visibility of diagnostic structures**

This technique is widely used in radiology preprocessing pipelines.

7. Image Denoising

To remove minor pixel noise without losing detail, median filtering is applied. This:

- **Removes “salt-and-pepper” noise**
- **Preserves edges better than Gaussian blur**
- **Improves clarity for subtle medical features**

8. Cropping and Padding

Medical images often contain black borders or scanning margins. The system:

1. **Detects the real image content**
2. **Crops unnecessary borders**
3. **Rescales the image proportionally**
4. **Pads it back to a uniform dimension**

This ensures every image:

- **Remains centered**
- **Maintains spatial accuracy**
- **Has consistent input size**

9. YOLO Bounding Box Conversion

Original bounding boxes (COCO-style) require conversion to YOLO format:

```
(x_center, y_center, width, height)
```

All values are normalized to `[0, 1]`.
This ensures compatibility with YOLO detection models.

10. Safe Medical Image Augmentations

The system applies conservative augmentations such as:

- **Horizontal flipping**
- **Small rotations**
- **Light contrast changes**
- **Slight zoom**
- **Controlled elastic deformation**

These augmentations:

- **Increase dataset variability**
- **Improve generalization**
- **Do not distort diagnostic features**

They also ensure bounding boxes remain accurate after transformation.

11. Train–Validation–Test Split

To avoid data leakage:

- **The dataset is split into training, validation, and test sets**
- **Stratified splitting ensures equal class distribution**
- **IDs are saved to text files so experiments remain reproducible**

A typical split might be:

```
80% → Training
10% → Validation
10% → Testing
```

This division ensures the model is evaluated objectively on unseen d