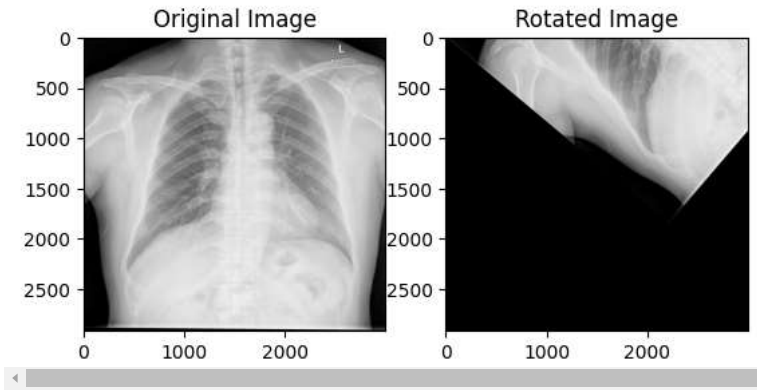
```
!pip install opencv-python matplotlib
```

```
Requirement already satisfied: opencv-python in /usr/local/lib/python3.10/dist-packages (4.10.0.84)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.10/dist-packages (3.7.1)
Requirement already satisfied: numpy>=1.21.2 in /usr/local/lib/python3.10/dist-packages (from opencv-python) (1.26.4)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (1.3.0)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (4.54.1)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (1.4.7)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (24.1)
Requirement already satisfied: pillow>=6.2.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (10.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (3.1.4)
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (2.8.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.7->matplotlib) (1.16.0)
```

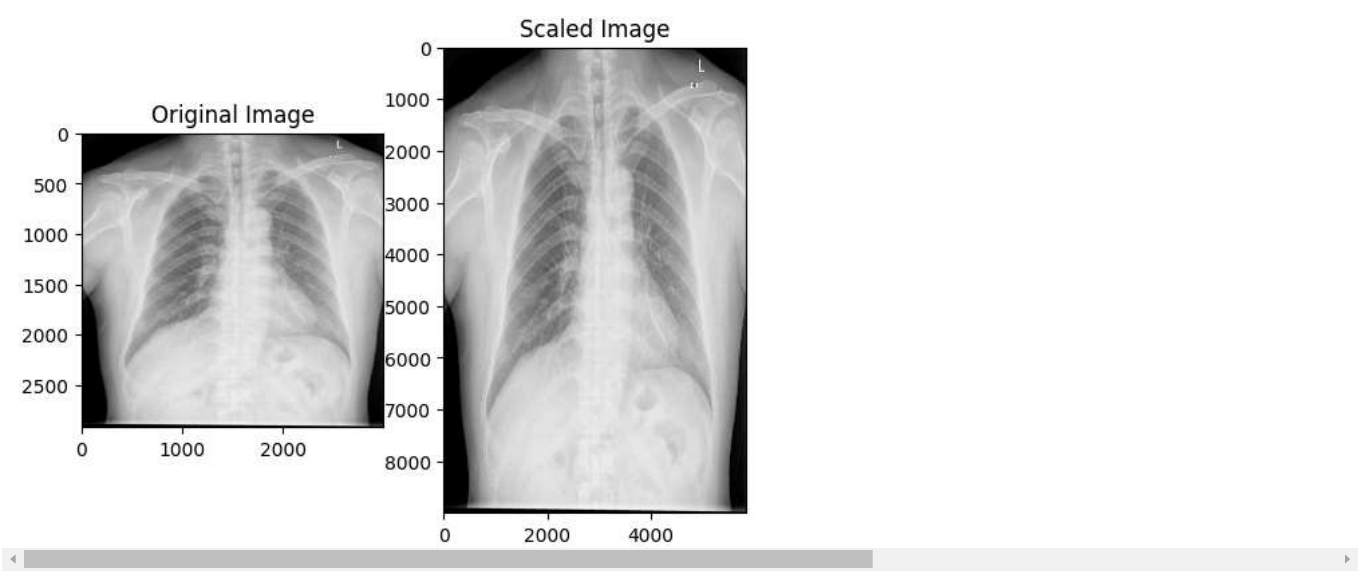
```
import cv2
import matplotlib.pyplot as plt


import numpy as np


from google.colab import files


uploaded = files.upload()
```

```
Choose Files No file chosen  Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.
```

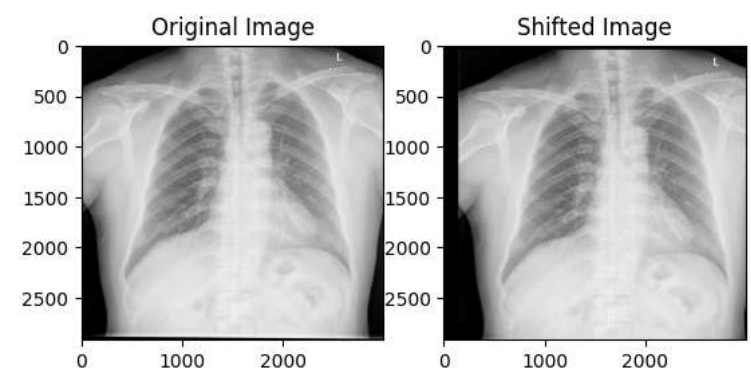
```
image = cv2.imread('ChestXRD.png')


image = cv2.imread('ChestXRD.png', 0)
row, col = image.shape


image_path = r'ChestXRD.png'  # Replace with your image path
img = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)


R = cv2.getRotationMatrix2D((0, 0), 50, 1)
rotated_image = cv2.warpAffine (image, R, (image.shape[1], image.shape[0]))


plt.subplot(1, 2, 1)
plt.imshow(image, cmap='gray')
plt.title('Original Image')
plt.subplot(1, 2, 2)
plt.imshow(rotated_image, cmap="gray")
plt.title('Rotated Image')
plt.show()
```



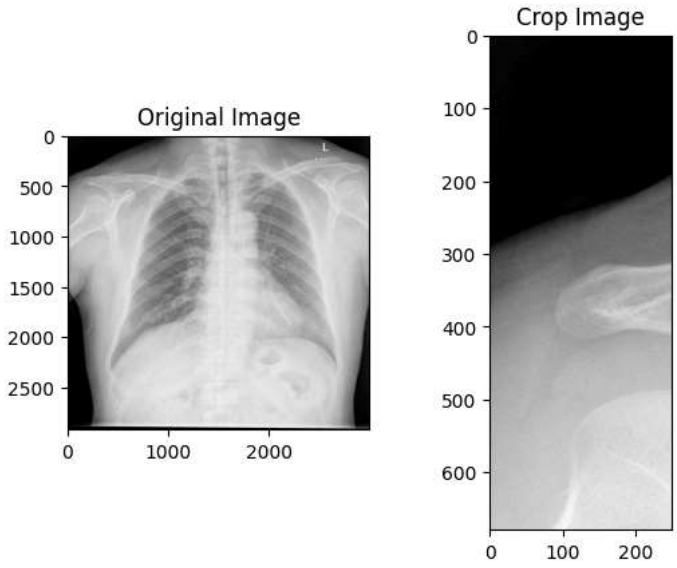

```
scaled_image = cv2.resize(image,(2*row, 3*col))


plt.subplot(1, 2, 1)
plt.imshow(image, cmap='gray')
plt.title('Original Image')
plt.subplot(1, 2, 2)
plt.imshow(scaled_image, cmap="gray")
plt.title('Scaled Image')
plt.show()
```




```
T = np.float32([[1, 0, 150], [0, 1, 50]])
shifted_image = cv2.warpAffine (image, T, (image.shape[1], image.shape[0]))


plt.subplot(1, 2, 1)
plt.imshow(image, cmap='gray')
plt.title('Original Image')
plt.subplot(1, 2, 2)
plt.imshow(shifted_image, cmap="gray")
plt.title('Shifted Image')
plt.show()
```




```
crop_image=image [100:780, 50:300]


plt.subplot(1, 2, 1)
plt.imshow(image, cmap='gray')
plt.title('Original Image')
plt.subplot(1, 2, 2)
plt.imshow(crop_image, cmap="gray")
plt.title('Crop Image')
plt.show()
```



Start coding or generate with AI.