

MIC-Assignment-1

[Image 11_12.jpg](#)

[Image 44_19.jpg](#)

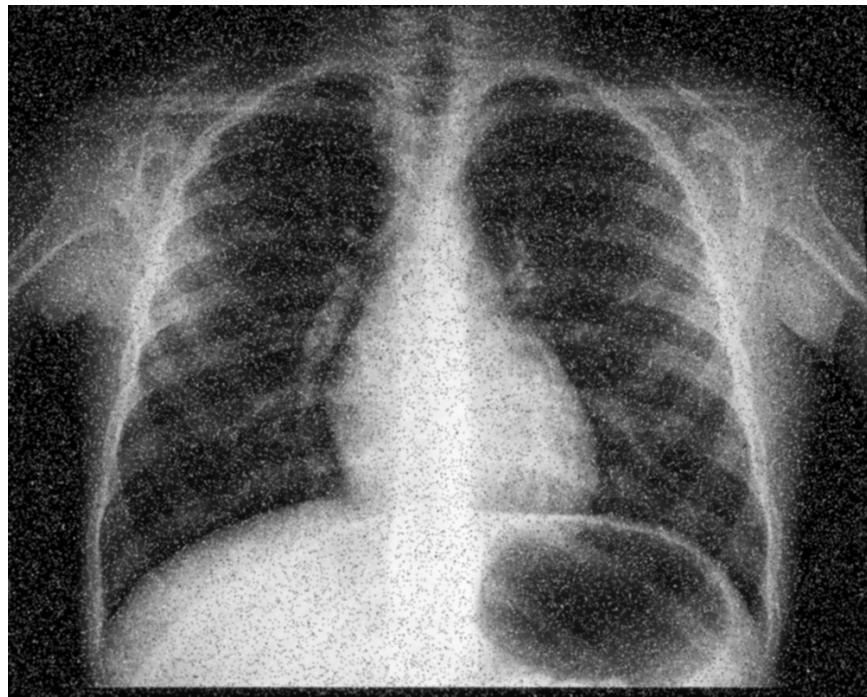
[Image 99_28.jpg](#)

[Image 101_20.jpg](#)

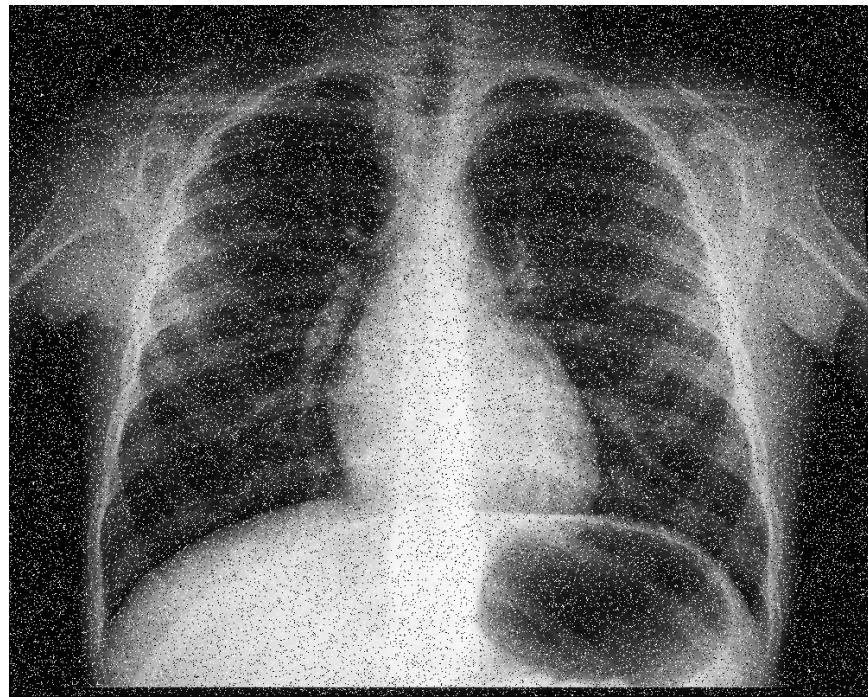
[Image 102_32.jpg](#)

Image 11_12.jpg

- For this image, I experimented with morphological operators but one of them removed the noise
- Later I tried, different smoothing techniques but they didn't remove the noise either, regardless of the size of the kernel

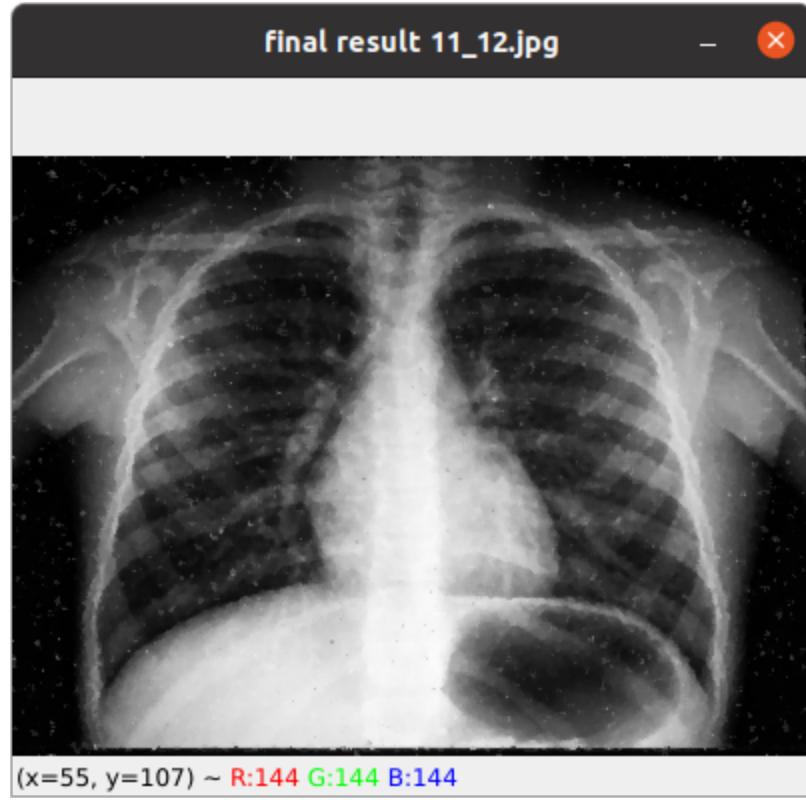


Gaussian blur with k=5

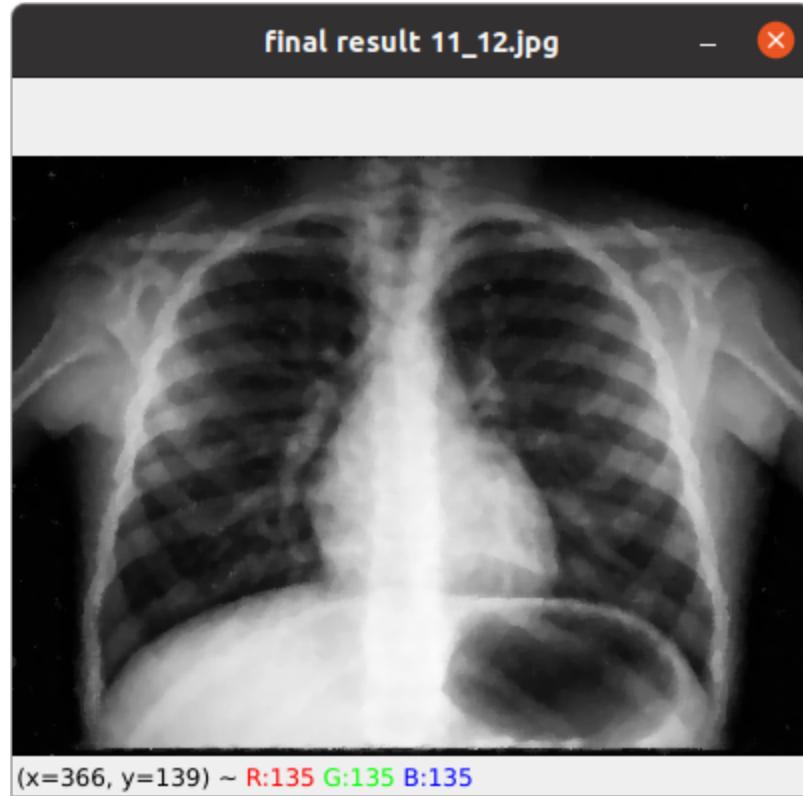


Bilateral blur with k=5

- Best result was provided by Median Filter (Blur)
- Kernel size was set to 7 as it yielded the best result result in terms of clarity and noise reduction. Anything less than 7 would leave some noise and a greater value would erode the details of rib cage.



median blur with k=3. notice that image still has roughly 5% noise left



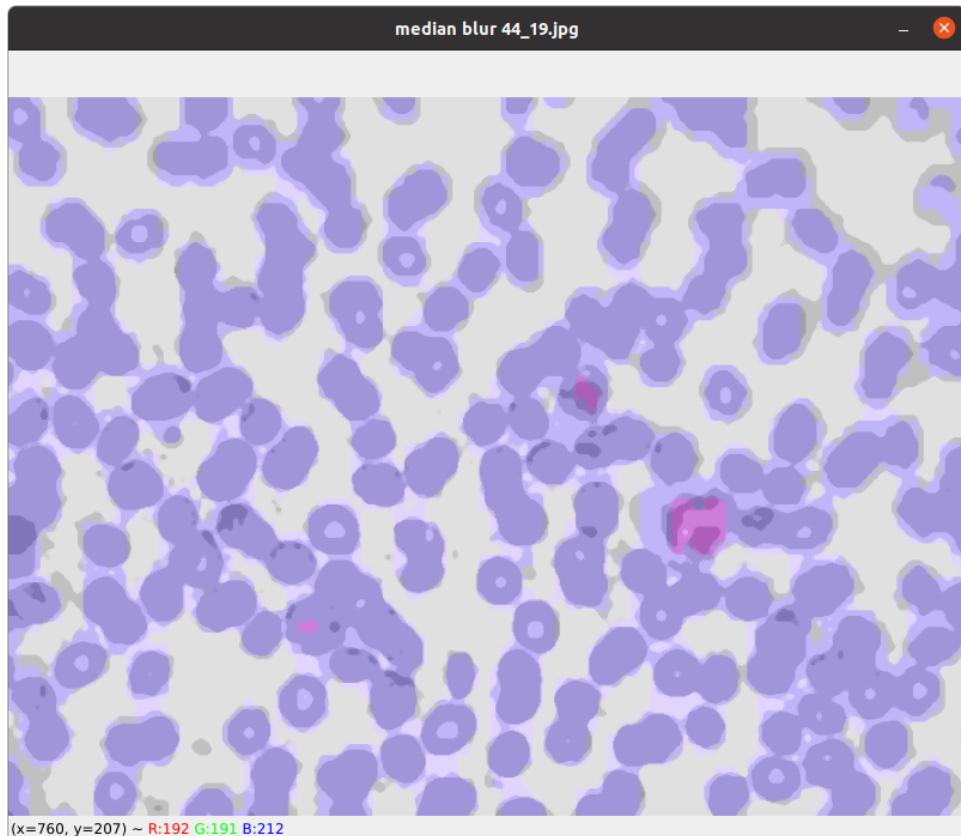
median blur with k=5. noise got reduced to 1%.



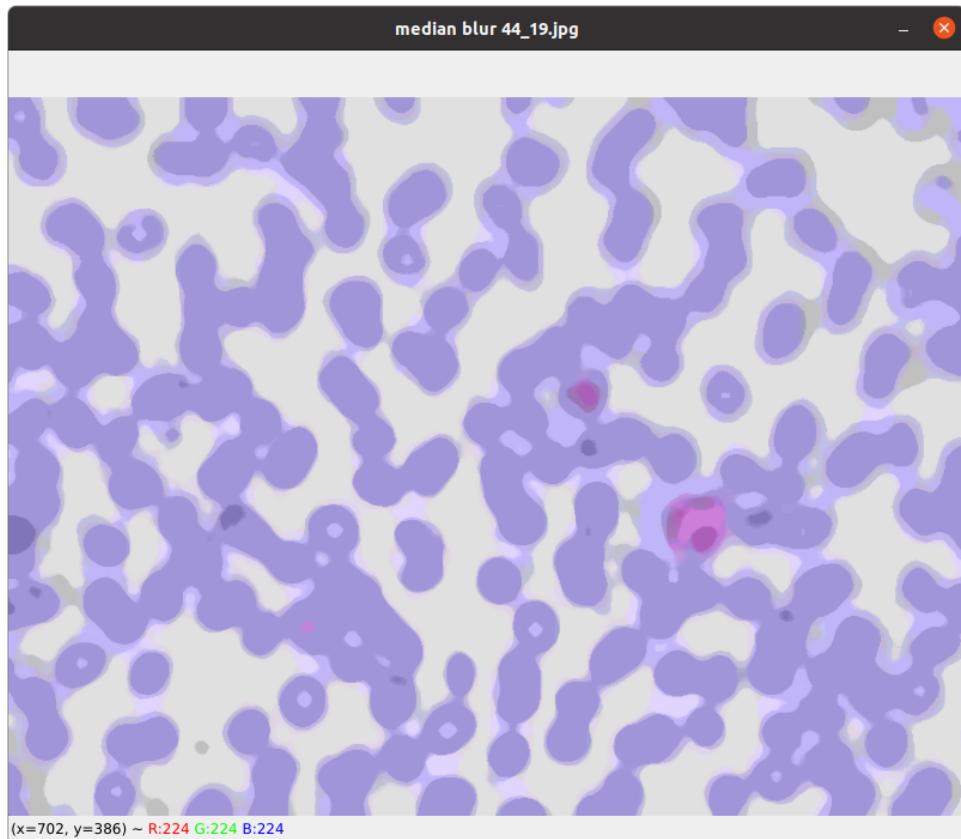
median filter with k=7

Image 44_19.jpg

Similar to 11_12.jpg, I used median blur with kernel size = 7



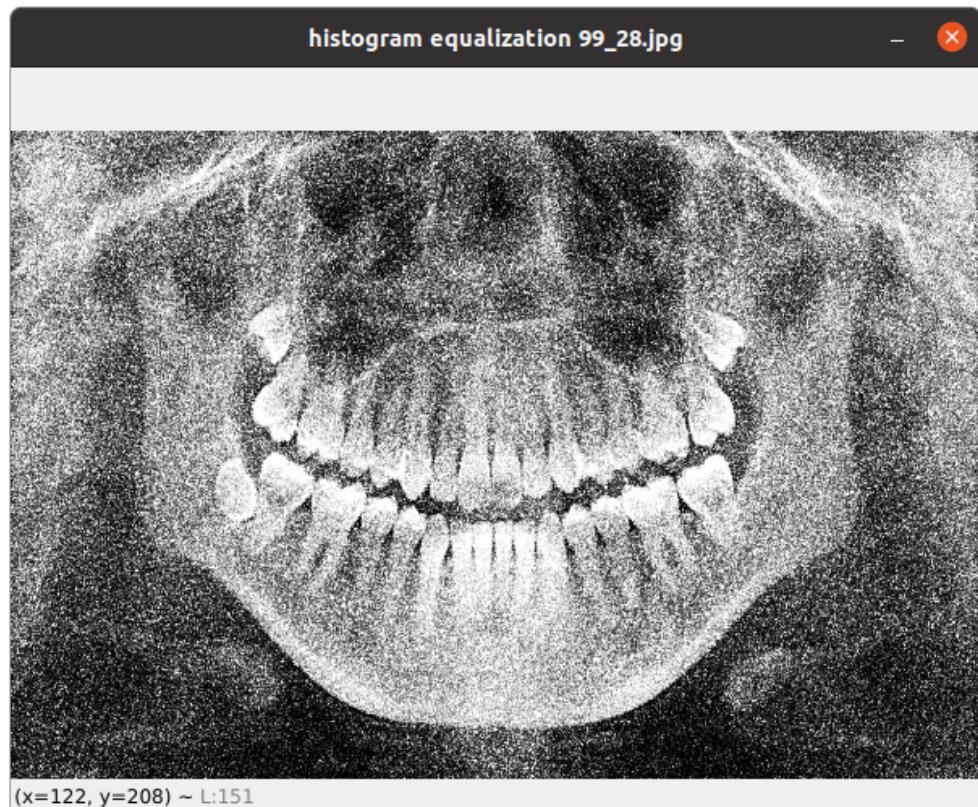
median blur with k=7



median blur with k=15

Image 99_28.jpg

In the case of 99_28.jpg, I combined histogram equalization (takes only gray scale image as an argument) with Median Blur of kernel size 7.



Output with histogram equalization.



Output with histogram equalization with median blur (k=7)



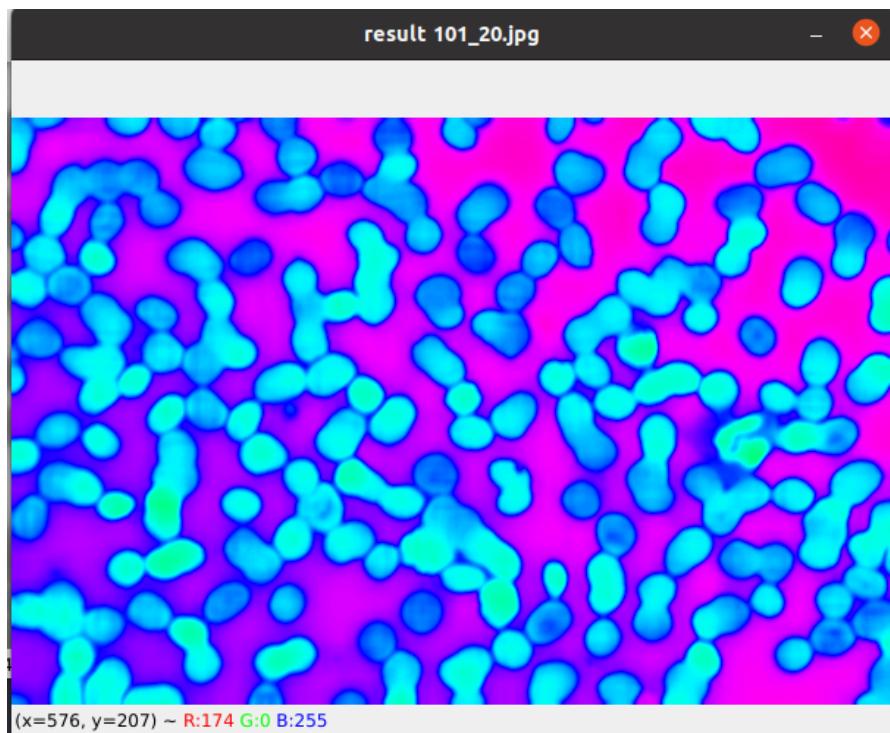
Output with histogram equalization with median blur (k=3)



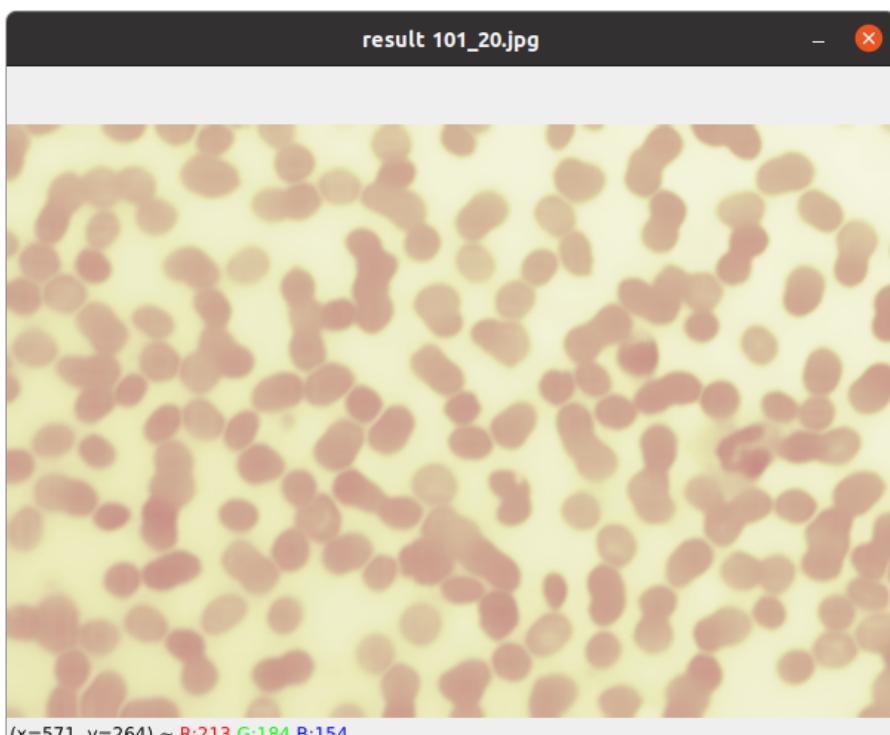
Output with histogram equalization with median blur (k=13)

Image 101_20.jpg

- For 101_20.jpg, I used Fast N Means Denoising for smoothing the image
- **h** factor was set to 10 as recommended in the documentation. Higher value will provide a more blurry and less detailed image.
- For colorization, I experimented with almost of the maps but none of the provided a pinkish result as shown in the [image11.jpg](#) and [image12.jpg](#). I ended up using [COLORMAP_PINK](#).



COLORMAP_HSV color mapping after median blur



COLORMAP_PINK color encoding after median blur

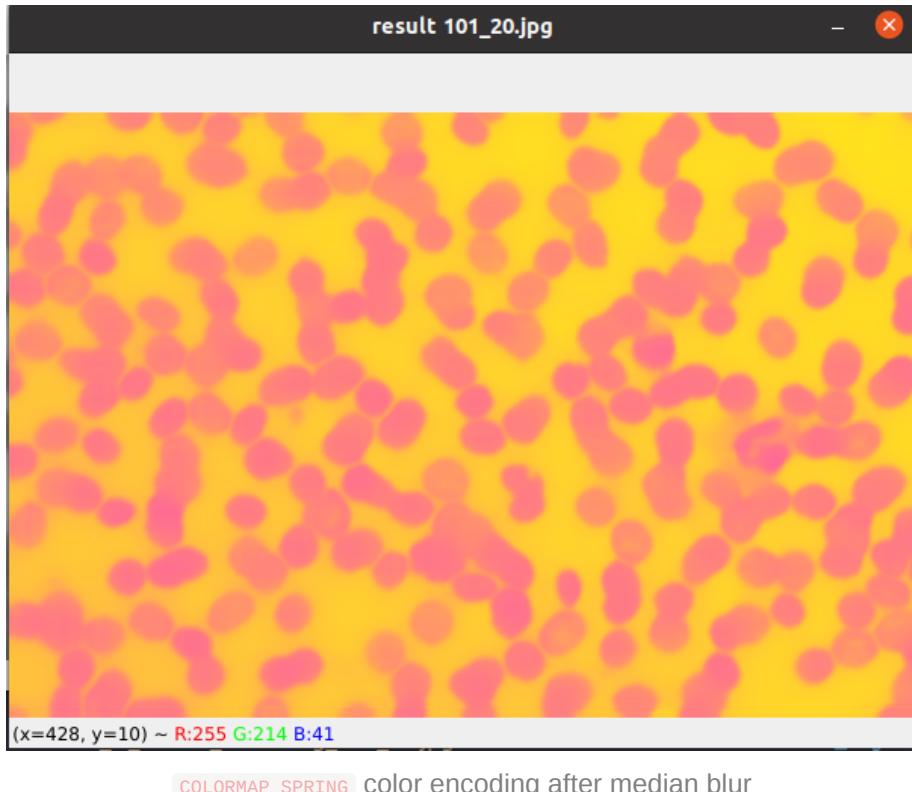


Image 102_32.jpg

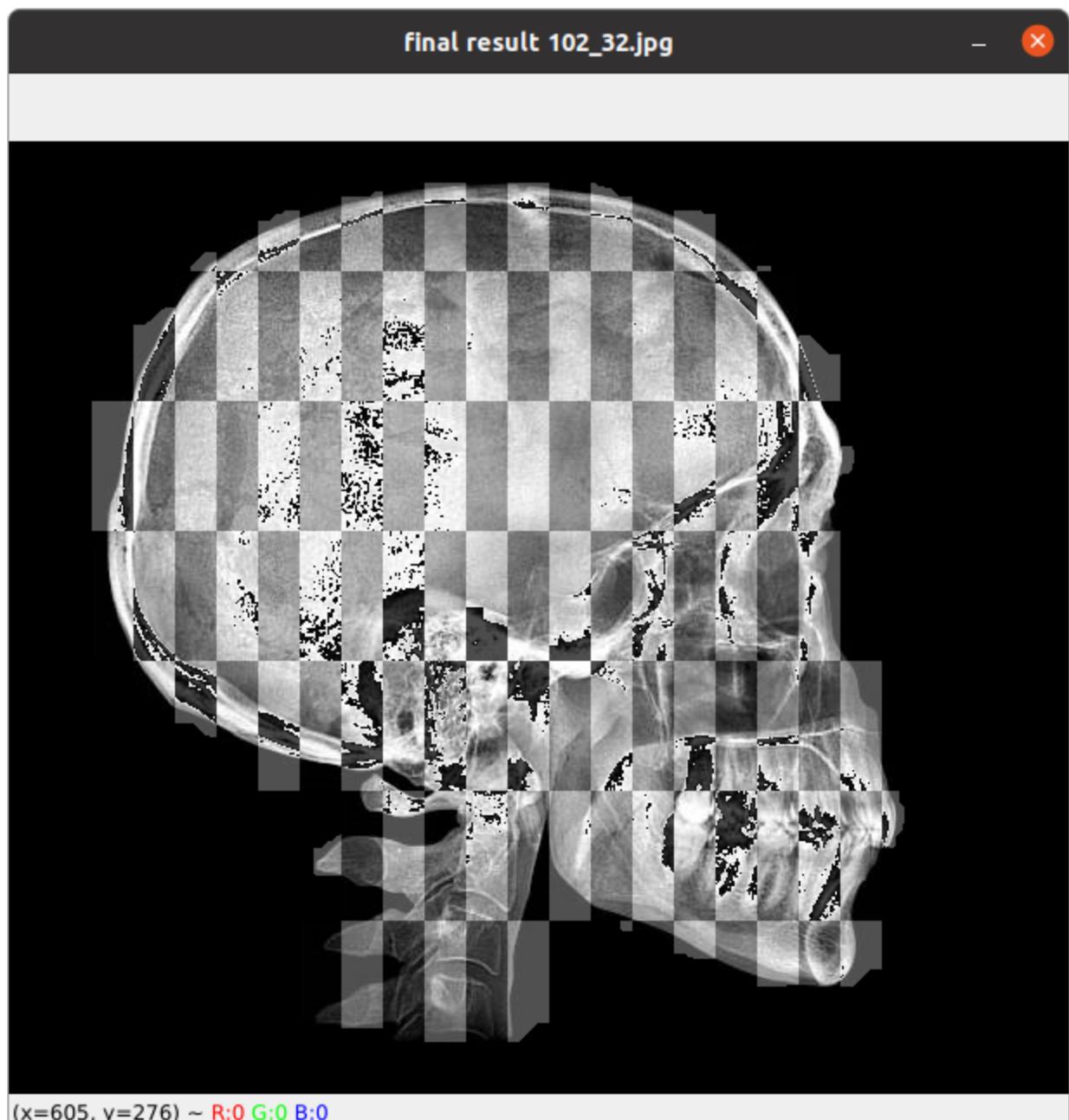
Processing 102_32.jpg involved Grab Cut methodology. This approach relies on couple of factors:

- Rectangle's coordinates (in case of rectangular mode):
 - In order to segment a particular region in an image, you should know the starting X and Y coordinates, along with the width and the height.
 - Any variation in any of these values will lead to missing/extra area in the segmented image.

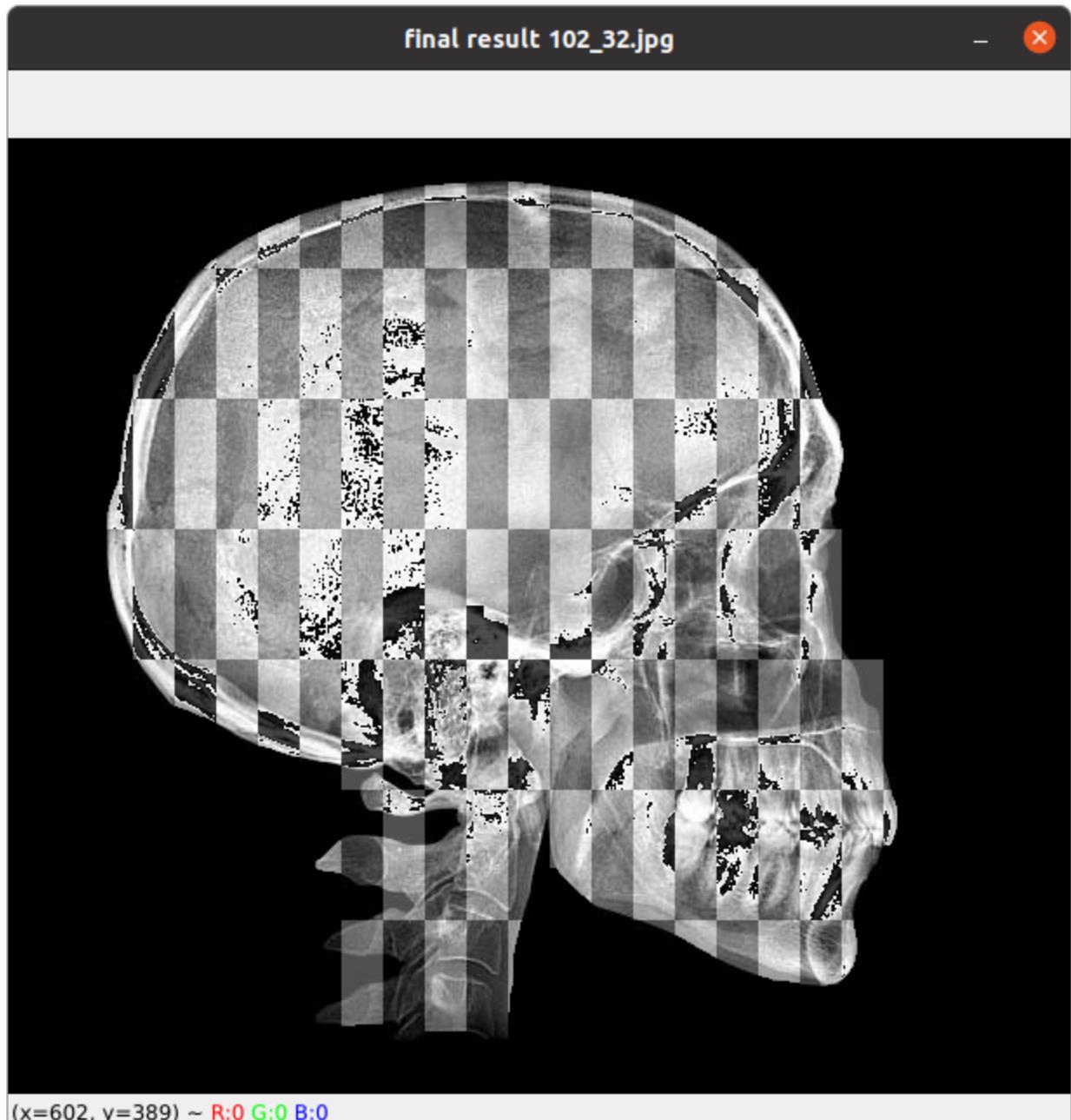


grab cut with wrong rectangle dimensions

- Iterations: Applying Grab Cut algorithm with lower iterations will result in an image having edges on the borders



grab cut algorithm with 1 iteration. notice how the left side of skull has edges remaining



grab cut with 8 iterations. edges on left side are now gone