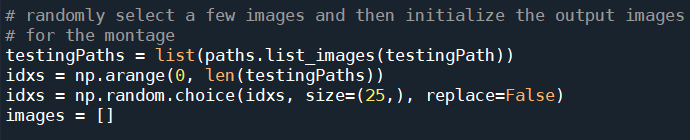
**Testing The Model**

After training the model, the model should be tested by using the test data which is been separated while splitting the data for checking the functionality of the model.

Here we are selecting 25 images from the test data and initialize the output images for montage



we’re going to create a montage so that we can share our work visually

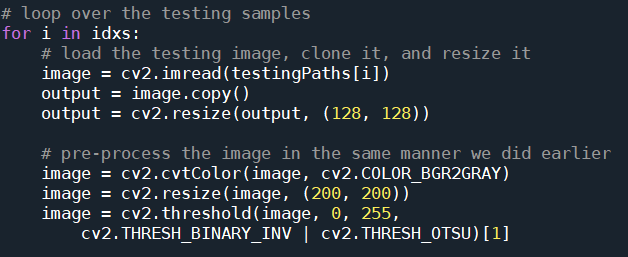
* First, we randomly sample images from our testing set

* Our images  list will hold each spiral image along with annotations added via OpenCV drawing functions .

* We proceed to loop over the random image indices.

* Inside the loop, each image is processed in the same manner as during training(convert to gray scale, resize, threshold) .

* From there we’ll automatically classify the image using our new HOG + Random Forest based classifier and add color-coded annotations

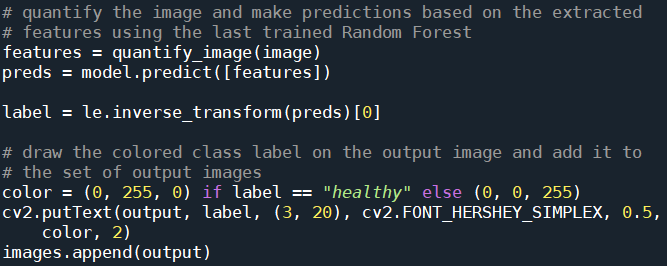


* Each image is quantified with HOG features.

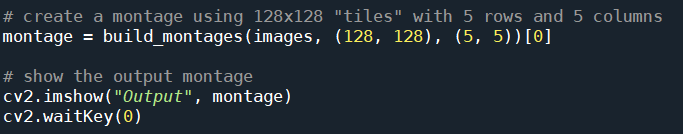
* Then the image is classified bypassing those features to model.predict .

* The class label is colored <strong>green</strong> for “healthy”  and <strong>red</strong> otherwise .The label  is drawn in the top left corner of the image using cv2.putText function .

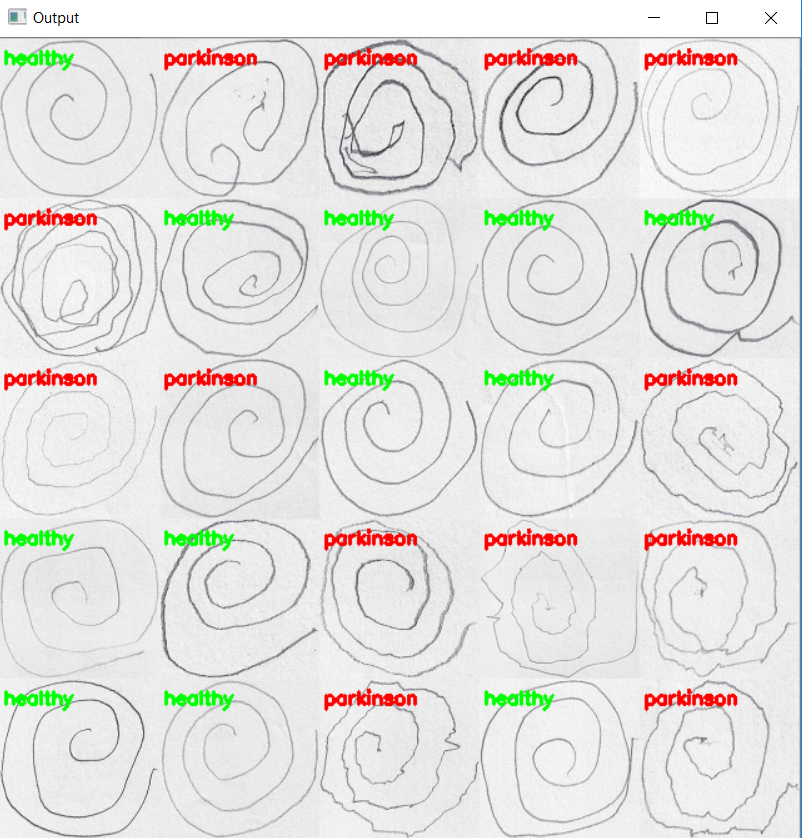
* Each output  image is then appended to an images  list so that we can develop a montage



* The montage is then displayed until a key is pressed



The **cv2.imshow()**function always takes two more functions to load and close the image. **cv2.waitKey()** function, you can provide any value to close the image and continue with further lines of code.



From the above montage, we can clearly differentiate between the geometric patterns drawn by healthy and Parkinson patient.