Image Manipulation Python

Project Test Code

Kirlan Merchant

This Document refers to the “project test code” file in my GitHub repository, (not the final code). Here I will explain what troubles I came across when working with this specific code.

First, I want to list my parameters as shown in my design document:

* Apply optional PIL Filters to image
  + Blur
  + Contour
  + Detail
  + Edge Enhance
  + Emboss
  + Find Edges
  + Sharpen
  + Smooth
* Apply optional manipulations to image
  + Resize
  + Crop
  + Rotate
  + Flip
  + Convert

**Explanation:**

* **Step 1**: Open the image from a given path and store it in a variable.
* **Step 2**: Display the image for debugging.
* **Step 3 to 8**: Apply various transformations to the image (resize, crop, rotate, flip, convert to grayscale, and apply blur).
* **Step 9**: Save the final image and display a message.

**Issues in the Original Code:**

1. **Saving the Wrong Image**: The code saves the original image, not the manipulated one. The final image saved should be one of the manipulated images (like resized\_image or blurred\_image), not the original image.
   * **Fix**: Save the manipulated image (e.g., image.save(resized\_image)).
2. **Incorrect File Name**: The file path for saving is 'newhands,jpg', which contains a comma instead of a period. This should be corrected to 'newhands.jpg'.
   * **Fix**: Use 'newhands.jpg' as the output path.

**Summary**:

My main goal for the code was to make all the manipulations optional, which in the first addition of the design document they were, blurring, merging two images, contrasting, adding a grayscale filter, inverting the colors, and resizing the image. I believed this code would have worked if I used the correct image logic, I was using img1 ang img2 to distinguish between multiple images needed to be manipulated. I was overwriting the function rather than modifying the “img” variable. I also used the “Image.show()” function incorrectly and had an incorrect output path string for the manipulated image as well as the print statement for the manipulated image. I ended up scraping this code and completely started over. I felt that this code lacked structure, and I also wanted to make it simple to follow for a user to manipulate an image using the optional functions.