

Use of Python Scripts to Compare and Contrast Photo Images

Kevin Benton, Disaiah Bennett

Center of Excellence in Remote Sensing Education
and Research

Elizabeth City State University
Elizabeth City, North Carolina 27909
kevinbenton1996@gmail.com,
lavontae.bennett@gmail.com

Derek Morris

Center of Excellence in Remote Sensing Education
and Research

Elizabeth City State University
Elizabeth City, North Carolina 27909
Dmow4454@gmail.com

Abstract- In python there is an ability to write python scripts for a variety of projects. For this project the python script was written to perform an image comparison. This was done with various libraries that are included in the python language. These libraries include Image, OpenCV, and etc. Thus, when importing these the program would ask the user for two images and the size for the images the program would display the pixel values for the images as well as display an image were if the pixels of the two images matched the image would be "black out" for that section. The team later found out that to compare the images the team needed to resize the two images as well as that to have the program run on different computer the necessary libraries needed to be on the new computer. Therefore, in the future the team decided that the program would need to be more robust and efficient.

Keywords – Python, Image Comparison

I. INTRODUCTION

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms and can be freely distributed. [1]

A. PIL

PIL is the Python Imaging Library which provides the python interpreter with image editing capabilities. It was developed by Fredrik Lundh and several other contributors. Pillow is the friendly PIL fork and an easy to use library developed by Alex Clark and other contributors. [2]

B. OpenCV

OpenCV was started at Intel in 1999 by Gary Bradsky, and the first release came out in 2000. Vadim Pisarevsky joined Gary Bradsky to manage Intel's Russian software OpenCV team. In 2005, OpenCV was used on Stanley, the vehicle that won the 2005 DARPA Grand Challenge. Later, its active development continued under the support of Willow Garage with Gary Bradsky and Vadim Pisarevsky leading the project. OpenCV now supports a multitude of algorithms related to Computer Vision and Machine Learning and is expanding day by day. OpenCV supports a wide variety of programming languages such as C++, Python, Java, etc., and is available on different platforms including Windows, Linux, OS X, Android, and iOS. [3]

C. Numpy

NumPy is the fundamental package for scientific computing with Python. It contains among other things like a powerful N-dimensional array object, sophisticated (broadcasting) functions, tools for integrating C/C++ and Fortran code, useful linear algebra, Fourier transform, and random number capabilities. Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-

types can be defined. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases. [4]

II. METHODOLOGY

For the python program, the team researched many python topics such as how to open an image and also how to get the pixel values for both images. This knowledge provided a foundation that was used to build a python program that would ask the user what two images that the user wanted to compare, display the pixel data for the two images, and, finally, output a picture that showed the differences between both photos. It was discovered that, unless the two images were the same size, the program could not compare the two images. The root of this cause was the fact that the pixel data was passed through an array the lengths of the two images needed to be the same. Thus, the program was altered to ask the user what size picture the user would want.

III. RESULTS

The python script utilized constructed a separate compare/contrast image from the two original photos selected by the user. Printing out each image individually, the pixel's value is displayed on the screen and organized in a descending order by rows and columns.[5] With the current python script, images of the same width and height are recommended when comparing. As the team advanced in testing the python script, two large images were compared but of different sizes. However, the overwhelming amount of data being produced within the CPU resulted in the system crashing on various occasions. When two images of a different size are compared the image's matrix ratio will proceed in an error, due to the image sizes not being compatible. To proceed comparing multiple images, the python script grants the user the option to select between resizing an image or maintaining its current form. By resizing the photo to an equal size, the images are reduced or enlarged by the CPU and are subtracted by the each other to produce an image that is the contrast.

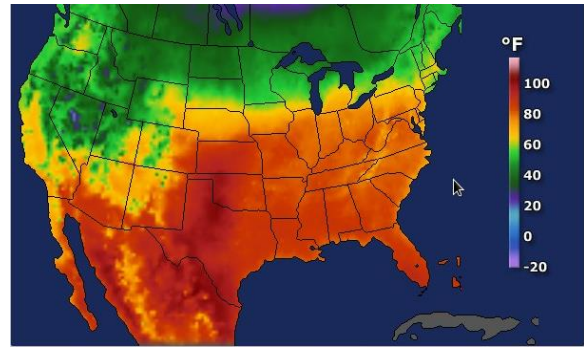


Fig. 1. First image

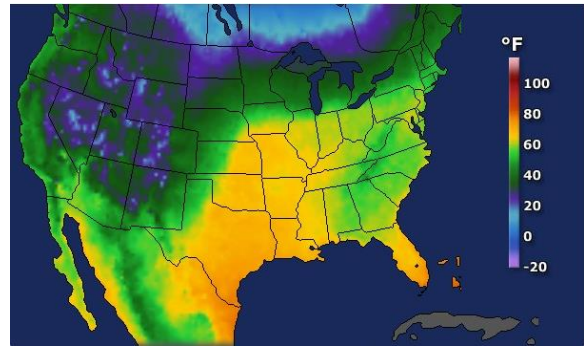


Fig. 2. Second image

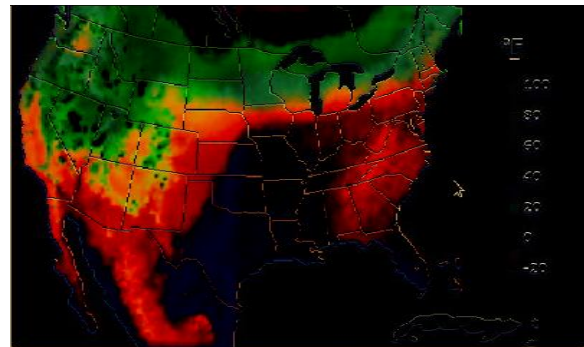


Fig. 3. Image that shows the differences between the two images

IV. CONCLUSION

After testing the code, the team determined that multiple images of different file types could be compared based on pixel values. However, images must be resized to the same width and height to the opposing image. By reflecting on the data, python image libraries and modules were rated to produce higher statics and image variables over majority of the programming languages. The team also concluded that image comparison is not the same across different operating system platforms. Each operating system contains their own modules and necessary drivers required to execute the program, even when they can utilize the same python library.

IV. FUTURE WORK

In the future, the team plans on manipulating the code to compare multiple images at once, rather than two images and producing a more efficient result. Based on the data produced by the python program, the script utilization limited the user to operate on two images to reduce the amount of system errors/crashes. In a more professional setting, operating at a lower and more balanced time complexity would be recommended. However, developing a method that would deliver faster and sharper image comparisons would be ideal for researcher in the STEM field. Image processing/comparison programs are developed to detect anomalies within various and diverse images; therefore, the development of the script is critical towards certain performances

V. ACKNOWLEDGEMENTS

We would like to thank Dr. Linda B. Hayden for her support during this project. The team would also like to thank Derek Morris, for his assistance and his guidance with the research and execution of the project.

VI. REFERENCE

- [1] Python.org. (2018). What is Python? Executive Summary. [online] Available at: <https://www.python.org/doc/essays/blurb/>.
- [2]"EXTRACTING PIXEL VALUES OF AN IMAGE IN PYTHON - Akshay Pai", HackerEax *rth*, 2018. [Online]. Available: <https://www.hackerearth.com/practice/notes/extracting-pixel-values-of-an-image-in-python/>.
- [3]"OpenCV library", *Opencv.org*, 2018. [Online]. Available: <https://opencv.org>.
- [4]"NumPy — NumPy", *NumPy.org*, 2018. [Online]. Available: <http://www.numpy.org>.

[5]"scikit-learn: machine learning in Python — scikit-learn 0.19.1 documentation", *Scikit-learn.org*, 2018. [Online]. Available: <http://scikit-learn.org/stable/>.