Anthony Pichardo

Scope of Project

Tool Architecture:

Python

Face++

Follow Google Coding Standard

Performance requirements speed vs. Quality:

My platform will place quality of the results over speed of performance.

Error handling philosophy:

Since my platform is about accuracy on error we will ask the user to resubmit or make any changes that are causing the error to occur.

POC Interface:

For the proof of concept of this project it will have a command line interface as opposed to a GUI. This is because the core functionality is about the ability to categorize races using facial recognition software. The usability and accuracy of that functionality has a higher importance than a GUI interface for loading in sample images and reading out results. The CLI will take in a folder of images and the file to output to as parameters.

The goal for the POC is to successfully distinguish one race from another on a high level. Meaning if given an image of a person the system should be able to identify which continent(s) an individual derives from.

MVP Interface:

For the MVP I will have a fully functioning GUI with the ability to drag and drop images as well as import files from the file explorer.

The MVP will be able to identify people into three categories: White, Black, Asian and follow the same philosophies outlined above for the POC.

There will be a pop-up to display the results of the tool.