Student: Elshan Naghizade

Project: Python Library To Transform Image Datasets in Query-able SQL Tables

Date: 25 June, 2023

RESEARCH STRATEGY

DEVELOPMENT STRATEGY

Data Extraction: I have built a loader capable of reading image datasets from a myriad of formats such as *JPEG, PNG, TIFF, BMP,* etc. Whereas utilizing the *PIL (Pillow)* and the *OpenCV* libraries facilitated this task. The function is able to handle both singular images and directories consisting of subdirectories of images by means of traversing the file tree fed to the library.

Feature Extraction: Upon successful loading of the images, the next phase involves extracting meaningful features. The feature extractors handle from basic features like color histograms, texture, shape to more complex ones like edges, corners, whereas deep learning features derived from pre-trained models will most likely be added in the upcoming weeks. *OpenCV* is the go-to for simple image features, whereas coupling with *scikit-image* caters to advanced features.

Data Transformation: Once features are extracted, they are converted into a query-able format such as SQL. I am leveraging libraries like SQLAlchemy and psycopg2 to spawn SQL tables and insert data into them.

QUANTITATIVE ANALYSIS

Memory Usage: The goal here is to measure how much memory the library uses since it mostly operates in RAM. Python's ***'memory-profiles'*** library is to be used as the main benchmark.

Scalability: This involves analyzing how the library's performance scales with increasing dataset size. It's critical to understand how the tool performs when processing small, medium, and larger datasets, and whether it maintains its speed proportionally as the data volume grows. Python's ***'time'*** library will be used to log the corresponding timestamps.

QUALITATIVE ANALYSIS

Flexibility: This review metric will involve measuring the variety of image formats and feature types the library can handle compared to manual methods. I am aiming to cover most popular image formats used in conventional Computer Vision.

DATA ACQUISITION STRATEGY

I will need any 3 image datasets of different sizes to benchmark its scalability. So simply sorting by dataset size on Kaggle does the job. I am expecting to observe proportional execution time growth depending on dataset size.

1) Small (9 Mb) - CAPTCHA Images:
   https://www.kaggle.com/datasets/fournierp/captcha-version-2-images
2) Medium (55 Mb) - Celebrity Face Image Dataset:
   https://www.kaggle.com/datasets/vishesh1412/celebrity-face-image-dataset
3) Large (264 Mb) - WeedCrop Image Dataset:
   https://www.kaggle.com/datasets/vinayakshanawad/weedcrop-image-dataset

DATA CLEANSING

The library is to be used in lieu of data loaders, therefore the actual cleansing of particular datasets is out of the scope of my project.