**Python Client Script**

**Description**

The Python script acts as a client and communicates with the FPGA that acts as a server. The script is documented very well with (almost excessive) comments so that the user knows what is happening at each step of the way. This document provides a high-level overview of the Python script because in-depth logic can be figured out from the script itself, and also because the script is changing on a daily basis and the modifications will be noted in the script itself while the high-level goal remains the same.

The primary goal of the Python Client script is to: transmit user-defined parameter data to the FPGA server, figure out how many images/frames exist within a directory, extract the data from the image, transmit the data to the server on the FPGA, send new frames to the FPGA server as requested, receive the results from the FPGA server after DIC has completed, and lastly format all of those results into scientific notation within a solutions.txt file. The script provides timing statistics that print to the console for the user so they have an idea of how long each step of the processing took. The script converts some of the user-defined parameters in the Subsets.txt file to floating-point format as needed based on what the DIC requires. It creates a socket to establish TCP communications between it and the FPGA. TCP was chosen because it is more reliable in sending data and our DIC requires every pixel value and can’t afford data/packet loss via UDP. Python version 2.7.15 was used for the script. This decision was made for previous operations within the script, but it has since been streamlined, so converting it to 3.7.2 is an option. Libraries installations are required for: bitstring, PIL, and numpy; these are done with ease using “pip” in the terminal. Currently the script has support for 14 subsets and up to 99,999 frames but this can easily be modified within the noted sections of the code.