Burn Segmentation - folder explanation

Dataset - Folder containing all images and annotations - currently in use data folders may be left here and later categorized into one of the subfolders

- burns - We crafted much of our own data, these were the base 20 images that we then overlaid burns onto
- csvs - Possibly empty depending on gitlfs cost, but used to hold the crafted dataframes storing each pixels identification
- Extra Images - Images used to augment the initially crafted dataset. Since we moved from neural networks to decision tree classifiers, a set size of image was no longer required
- full_coco - parent folder containing the crafted images, class masks (category 1 is healthy skin, category 2 is burned skin), and the original polygon annotations created by CVAT after labeling
- labeledburnsegments - CVAT's failed export of image masks, not used but if CVAT fixes the export it could be used later
- OriginalImages - original image that would later be resized and burns placed on them
- OriginaResized - images were originally resized to 250x250 so we had the option of using neural networks
- saved_models - copy of the saved decision tree classifier with different number of images trained off of. "Dtree_additional_images2.save" was the most currently trained as of June 24th 2022
- tmp_files - folder for storing misc files when cleaning up but too scared to delete
- training_data - Images with burns placed on original images

Documents - Store of related white papers / documentation

old_notebooks - notebooks to create the original dataset - later when we used dtree classifiers the image size was no longer important so we moved to placing pixels of a specific class in their own images and added by "add_data.ipynb"

add_data.ipynb - Jupyter notebook that will take 3 lists (background pixels, healthy skin, and burned skin) and converts each of their pixels to an HSI representation. Each HSI value is then placed with it's classification into a pandas dataframe and saved to CSV

dtree_skin_segmentation.py - Python file that the mobile app will use to import/identify pixels. Code tested and verified in skin_segmentation_dtree.ipynb will be placed here after confirmation with the team

requirements.txt - Required libraries to run the code

skin_segmentation_dtree.ipynb - Working code that trains the model with 5 fold cross validation and the place where changes to classification/image processing were tested before being brought into the app

SkinOnlyExample.png - Image used for testing model output tested in "skin_segmentation_dtree.ipynb"