**OCR Model Training and Image Processing**

**OCR Training:**

* Script: fastai\_training2.ipynb uses FastAI to train a CNN model using Resnet50 pretrained model
  + Loads images
  + Creates databunch
  + Runs initial training using CNN resnet50
  + Fine-tunes by adjusting the weights
  + Exports the model into ‘export.pkl’ file to be used in OCR recognition
* Training annotation library:
  + Library has folders with images labelled into separate classes
  + The expansion of the annotated library is an ongoing process to help to improve the model as more new drawings and new characters/combination of characters are detected

**Image Processing:**

Scripts:

* Process\_img.py – the main script which calls other scripts for detections, classification and generating text
  + Please change the location of the image files, output directory and other files as they are listed at the top
  + Can be run via terminal calling python process\_img.py
* Detect.py – the character detection script
  + Runs image processing
  + Contour detection
  + Outputting bounding box coordinates
  + Saves a detection image with all contours identified
* Classifier.py – the detected contour recognition using trained CNN model
  + Each contour is run through the export.pkl
  + Identified contours are output to be overlayed on an image to the relevant positions
* Generate\_text.py – script which groups recognized characters based on x position and angle proximity
  + Outputs lines of text in the order of appearance along the y axis
  + Generates a separate text files with the identified and grouped text

To do:

* Run the latest generation of the export.pkl
* Test and validate batch level processing to speed up the classification process
* Set up a regex search logic for specification patterns