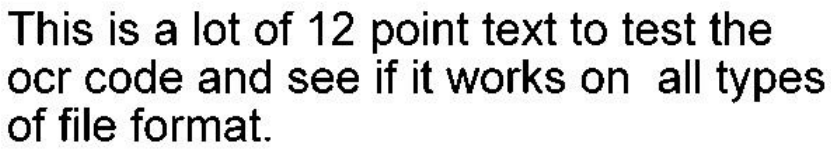
Lab 4 - Optical Character Recognition

Install an OCR library (Tesseract, for example; https://pyimagesearch.com/2017/07/10/using-tesseract-ocr-python/) and test its limitations:

1. Select two of the test files from directory ocr and sample21.jpg and create the "ground truth" information. For image:



   the "ground truth" is: This is a lot of 12 point text to test the ocr code and see if it works on all types of file format.
2. Add different amounts of noise and different types of noise to the image to be processed; (Add different noise to an image | TheAILearner)
3. Apply affine transformations: rotations, shear (vertical, horizontal); vary the values of the parameters for these type of transformations (angle for rotations, for example);
4. Resize the image - choose resizing parameters that shrink and enlarge the original image; consider keeping the aspect ratio or altering this ratio
5. Blur the image with average filters of different sizes and Gaussian filters with different values for the standard deviation parameter.
6. Use image enhancement preprocessing techniques: image sharpening, morphological operations (erosion, dilation, opening, closing) with different structuring elements, thresholding techniques. Does preprocessing help?

Evaluate the results of applying all these transformations computing the number of well recognized characters.