**Problem Statement:** The aim of the project is to extract text from image by using Deep Learning models. The important application of OCRing images is we can learn more about the contents/words in the image by extracting the text inside that and also becomes easier to search. OCR (Optical Character Recognition) is an often discussed problem in the Expense Reporting Industry. There are lot of commercial OCR software packages like Nuance, Tesseract etc. They are priced quite heavily too. If this can be replaced by an in house open source OCR extractor, this would save lot of revenue for companies.

**Historical background :** OCR is a field of research in pattern recognition, artificial intelligence and computer vision. Early versions needed to be trained with images of each character, and worked on one font at a time. Advanced systems capable of producing a high degree of recognition accuracy for most fonts are now common, and with support for a variety of digital image file format inputs. Some systems are capable of reproducing formatted output that closely approximates the original page including images, columns, and other non-textual components.

**Similar Research :** When I was on the lookout for similar projects in this domain I found the below project useful. It extracts number plates from images. My project proposal would be of similar type, but should work for larger images and text, with the ability to handle paragraphs and sentences.

<https://matthewearl.github.io/2016/05/06/cnn-anpr/>

**Domain background**: The application of the project is huge and can be used in many domains. One of the domains, where I see this applicable is the Expense reporting domain, where machine learning models are used nowadays to extract the values from receipt such as Amount, Date, Vendor Name etc. To extract the tokens from the receipt image, many third party tools such as Tesseract or Nuance are being used.

**Dataset and Inputs:** I am planning to take two approaches to generate training data for this problem. I need a set of images and their corresponding text to build the deep learning model

* Generate own set of diverse images from text which would be varied in terms of coloring, rotation, placement of text and noise
* Use already available datasets such as <http://www.cs.cmu.edu/~yuntiand/sample.tgz>

The datasets would contain both word level and document level (containing multiple words) images. Inputs to the model would be the set of images and the output would be a sequence of characters numbers and symbols , which would be the text contained in the image. The problem is an application of Deep learning, where the CNN is used to encode the image into features(A variable-length context vector can be used instead of a ﬁxed-size vector) and then sequence to sequence to learning is used to decode the features into text. This might not be a typical classification or a regression problem. (An Attention mechanism can be used to produces a sequence of vectors from the encoder RNN from each time step of the input sequence. The Decoder learns to pay selective attention to the vectors to produce the output at each time step.)

The initial model would have 100000 training images generated to build the model and 20000 test images. These images would be in different dimensions and they would be padded to a constant length image. (dimensions of image would be decided later).

**Solution Statement:** The solution to solve this problem would be a deep learning model which has the ability to predict the OCR text from an image

**Design:**

* Generate training images with ground truth text
  + Generate training images using PIL library
    - Identify list of words from a well-known corpus
    - Identify list of fonts
    - Add visual effects to image such as warping, fake shadows, etc
  + Download available datasets such as <http://www.cs.cmu.edu/~yuntiand/sample.tgz>
  + Use labelled datasets from my current organization’s database
* **Encoder** : Add a CNN layer to convert the image to features
* **Decoder** : Add a stack of bi directional LSTM layers
  + A Visual Attention Decoder for producing final OCR text
* Experiment with different encoder/decoder models and with different deep learning architectures and loss functions

**Tools:**  The intention is to implement the project in Python using Keras with TensorFlow backend. For training large datasets, AWS GPU can be used. Additional packages such as PIL, NumPy, Scipy and ScikitLearn would be used.

**Benchmark model:** I am planning to use these two models as base and work on top of them

* <https://github.com/da03/Attention-OCR>
* <https://github.com/keras-team/keras/blob/master/examples/image_ocr.py>

**Evaluation Metrics:**

* + The difference in predicted and actual text in terms of distance (may be a Levenshtein distance) and computing them for the validation test. This would give us an idea of how far are the predictions from the actual text
  + Also accuracy can be tested using traditional metrics such as Accuracy, F1 score, and ROC-AUC scores. This would give us an idea in how many cases are we predicting the full text correctly.

**Reference :** <https://blogs.dropbox.com/tech/2017/04/creating-a-modern-ocr-pipeline-using-computer-vision-and-deep-learning/>

<https://en.wikipedia.org/wiki/Optical_character_recognition>