# Assignment No:5

Practical Exercise: Text Identification using OpenCV, Tesseract (OCR), and Deep Neural Networks

## 1) Problem Statement

Implement a text identification system using OpenCV for image processing, Tesseract for Optical Character Recognition (OCR), and a Deep Neural Network (DNN) for detecting and recognizing text in images. The system should be able to preprocess the image, detect text regions, and extract the text.

## 2) Libraries Used

Python:  
1. **OpenCV**: For preprocessing the image and applying transformations such as grayscale and noise removal.  
2. **Tesseract (pytesseract)**: For Optical Character Recognition (OCR) to extract text from the processed image.  
3. **NumPy**: For numerical operations on image arrays.  
4. **Google Colab**: For handling file uploads and displaying images.

## 3) Theory

Text identification in images is a crucial task in many real-world applications, such as document scanning, number plate recognition, and text-based data extraction from images. OpenCV is used to preprocess the image (convert to grayscale, remove noise, and apply thresholding), while Tesseract OCR is used to detect and recognize text regions. The deep learning model (DNN) helps in enhancing the accuracy of text detection and recognition by learning features from the data.  
  
Tesseract uses a combination of pattern recognition, machine learning, and statistical techniques to extract text from images.

## 4) Methods

1. **Image Preprocessing**:  
 - The input image is converted to grayscale and denoised using OpenCV's `fastNlMeansDenoising` method.  
 - Thresholding is applied to create a binary image, enhancing the text regions and removing unnecessary background details.  
   
2. **Text Detection**:  
 - Tesseract's `image\_to\_data` function is used to detect text regions in the image. It returns the bounding boxes around detected words along with confidence levels.  
   
3. **Drawing Bounding Boxes**:  
 - For text regions with a confidence level greater than 60%, bounding boxes are drawn around the detected text using OpenCV's `rectangle` method.  
   
4. **Text Extraction**:  
 - The detected text is extracted and returned along with its corresponding bounding box coordinates for further use or display.

## 5) Advantages and Disadvantages

- **Advantages**:  
 - OpenCV provides efficient image preprocessing techniques that enhance the accuracy of text recognition.  
 - Tesseract is an open-source OCR engine with robust text extraction capabilities.  
 - Deep Neural Networks (DNNs) can improve the accuracy of text detection by learning complex features.  
  
- **Disadvantages**:  
 - Tesseract's accuracy depends on the quality of the input image. Poor lighting or skewed images may result in inaccurate text recognition.  
 - Preprocessing techniques may need to be fine-tuned for different types of images to achieve optimal performance.

## 6) Diagram

## OpenCV OCR and text recognition with Tesseract - PyImageSearch

## OpenCV EAST model and Tesseract for detection and recognition of text in natural scenes | by Jaafar Benabderrazak (Human/Not A Robot) | Medium

## 7) Conclusion

Text identification using OpenCV, Tesseract (OCR), and Deep Neural Networks (DNN) provides an effective solution for recognizing and extracting text from images. By combining image preprocessing with machine learning techniques, this system is capable of handling various real-world applications, such as document scanning and automated text extraction from images.