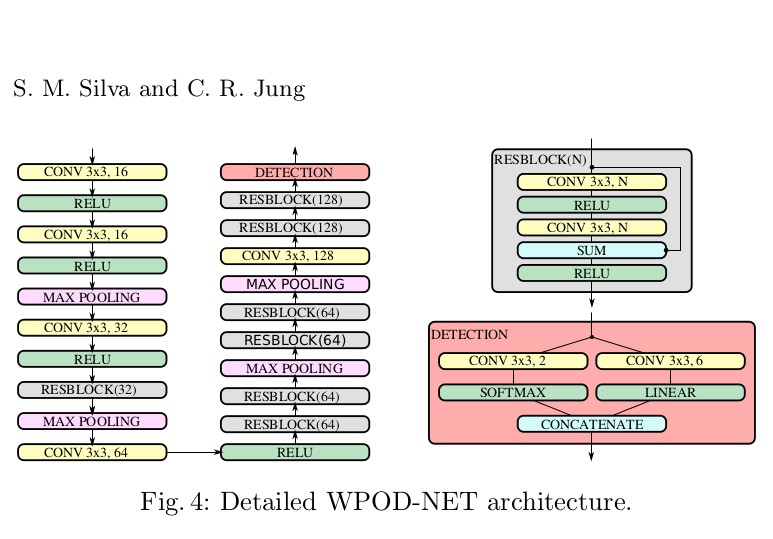
PROPOSED SYSTEM

The proposed system consists of three stages;

* License Plate ROI extraction
* Text Region Detection
* OCR

Licence Plate ROI Extraction

For license plate region of interest(ROI) extraction we used WPOD-NET(Warped Planar Object Detection Network) by Sérgio M. Silva et al. The network detects License Plates from a images containing cars in a variety of different different distortions and regresses the coefficients of an affine transformation that is used to present the license plate as a frontal view. Other methods like Christos N. A. et al use sliding window methods which can be computionally inefficent when running on embedded systems.



Architecture

The proposed architecture has a total of 21 convolutional layers, where 14 are inside residual blocks . The size of all convolutional filters is fixed in 3 × 3. ReLU activations are used throughout the entire network, except in the detection block. There are 4 max pooling layers of size 2 × 2 and stride 2 that reduces the input dimensionality by a factor of 16. Finally,the detection block has two parallel convolutional layers:

(i) one for inferringthe probability, activated by a softmax function

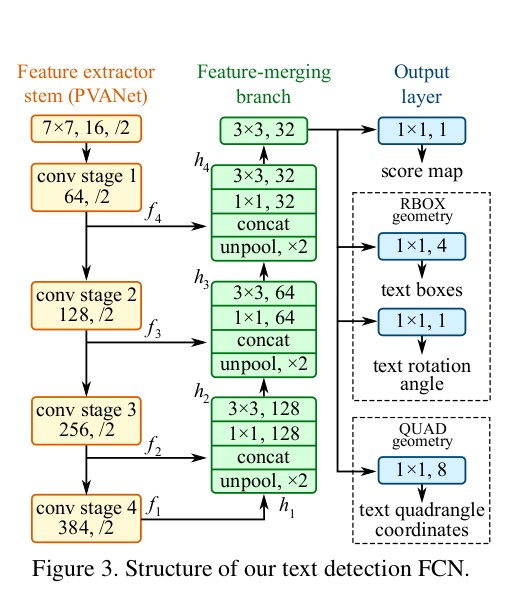
(ii) another for regressing the affine parameters, without activation (or, equivalently, using the identity F (x) = x as the activation function).

Text Region Detection

For text region extraction we employ EAST(An Efficient and Accurate Scene Detector) by Xinyu Zhou et al. Their method employs a two stage Fully Convolutional Neural Network.

Conventional methods for text region detection employed hand crafted features. Stroke Width Transform by B. Epshtein et al seek to predict regions using edge detection. Fastext by M. Busta et al modified the FAST key point detector to detect strokes. But most the non-deep learning methods perform poorly in terms of adaptability and accuracy. We employed EAST which is based on deep learning because it produces an F-Score of 0.7830 on the ICDAR-2015 dataset. It also employs a light weight neural network as a feature extractor which makes it efficient and ideal for use in embedded systems. It is also capable of detecting text regions in either word level or line level predictions, whose geometric shapes can be rotated boxes or quadrangles, depending on specific

applications.



Architecture

EAST consists of three stages as shown in the figure above;

1. Feature Extractor Stem
2. Feature Merging Branch
3. Output Layer
4. Feature Extractor Stem

The feature extractor stem can be a convolutional neural network consisting of blocks of convolution and pooling layers pretrained on the ImageNet dataset. The authors of EAST used PVAnet. Features are extracted at four level in the convnet. The different levels is to accomodate the varying sizes of text regions. This is denoted by where . The features have sizes of the input image.

1. Feature Merging Branch

In this branch the features from the levels of feature maps are gradually merged. In this stage, we first feed the feature map from the last stage of feature extractor into the unpooling layer to double its size. The ouput is then concatenated with the current feature map followed by a convolution. This reduces the number of channels and computation K. He et al. This is then followed by a convolution. The output feature map is then fed to the output layer.

3. Output Layer

The final output of the feature merging branch which consists of 32 channels is fed into the output layer where several convolutions are performed to project the 32 channels of the input feature map to output layer into 1 channel of score map and a multichannel geometry map . The geometry output can either be one of RBOX(Rotated Box) or Qaudrangle.

OCR

For Optical Character Recognition we use the tesseract v4 ocr engine to extract text from the license plate. Tesseract v4 uses an LSTM(Long Short Term Neural Network) to recognize text. For each text region detected by EAST in the license plate extracted by WPOD-NET we passed this into the tesseract OCR engine to perform text extraction.

Development Environment and Tools

The system was written in python computer programming language. We used Keras which is an open source deep learning library which runs on many low level deep learning libraries in our case Tensorflow. We also used OpenCV, an open source computer vision library. Pytesseract which runs on the Tesseract OCR engine was also used in our project. We also used Numpy, a numerical computation library for python. The system was tested and run on Ubuntu Linux 18.04.

Constraints

An appropriate scaling factor had to be chosen for each image in order to achieve more accurate text extraction results. We had manually

Conclusion

We presented a system that does License plate extraction, Text Region Detection and License plate number OCR. This system can be improved upon with car detection to further narrow down the ROI space for the License plate detector.