End-to-End License Plate Detection and Recognition Using Region Proposal Networks

Firas Naber

Computer Science Senior Seminar
Division of Science and Mathematics
University of Minnesota, Morris
Morris, Minnesota, USA

13 November 2021

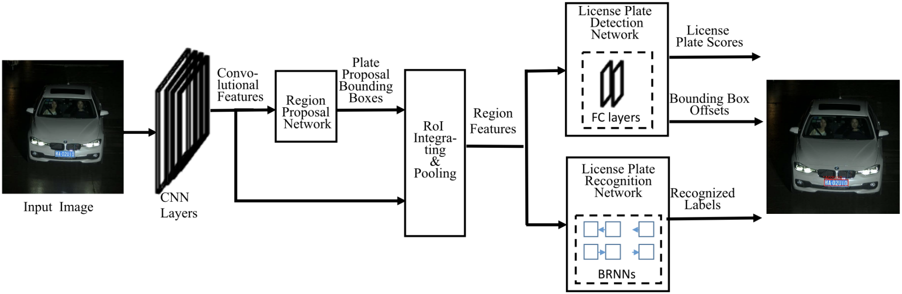
Outline

Outline

Li *et al.* license plate detection and recognition method

- Hui Li , Peng Wang, and Chunhua Shen proposed a license plate detection and recognition method [6]
- Detection: Find license plates in an input image
- Recognition: Read license plate characters/labels
- Takes an image as an input
- Simultaneously (Jointly-connected)
- End-to-end computation

The Model Structure by Li *et al.*



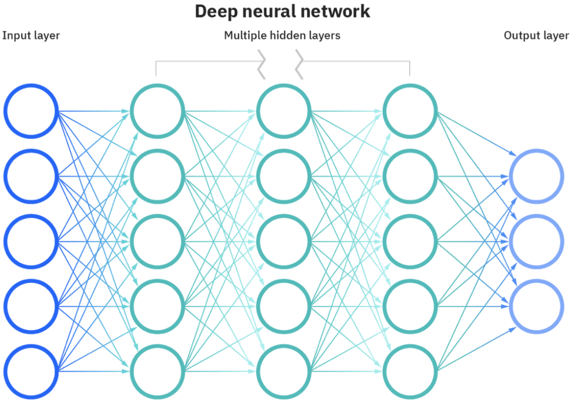
Outline

Neural Networks

What are neural networks?

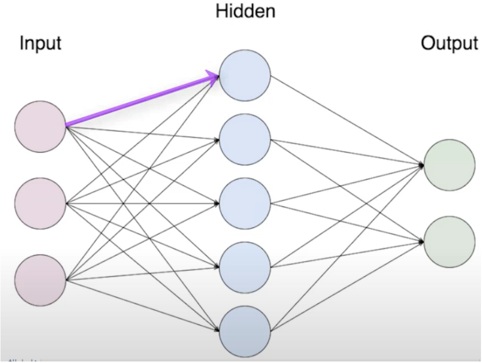
- Artificial neural networks (ANNs)
- Combination of nodes, or artificial neurons that connect together, forming node layers
- They allow computer programs to recognize patterns and solve common problems in different fields such as AI, machine learning, and deep learning.
- Each node contains an input, at least one “hidden” layer, and one output layer
- Deep neural networks (DNNs): ANNs with many hidden layer

Neural Networks - Visual



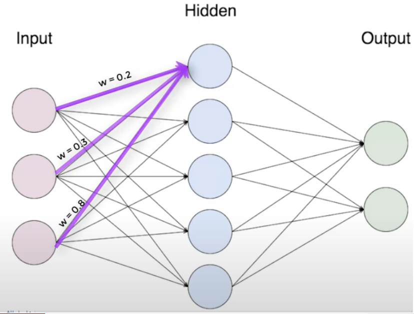
Passing Data Through Neural Networks

- Every node in a layer is connected to all the nodes in the next layer
- Every connection (arrow) transfers the output of the previous unit as input to the receiving unit



Passing Data Through Neural Networks - cont.

- Each connection would have its own “weight” (a value between 0 and 1)
- Weights represent the strength of the connections between the units
- The weight value would be multiplied by the value from the previous (output) unit, sending the multiplication result to the next unit as its input.



ANN Training

- Neural networks need to be trained in order to perform better
- Training is done by using specific algorithms that find the sets of weights that would map inputs to outputs at most efficiency possible.

Convolutional Neural Networks (CNNs)

- Derived from ANNs
- Superior performance with image inputs
- Convert images into numeric values (matrices)

CNN Layer Types

CNNs have 3 layer types:

- Convolutional Layer(s)
- Pooling Layer(s)
- Fully-connected (FC) Layer(s)

Convolutional Layer(s) - CNN Layer Type

- Convolutional layers tend to be at the front of a network
- Allowing network to look for specific patterns

Convolutional Layer(s) - Looking for Specific Patterns

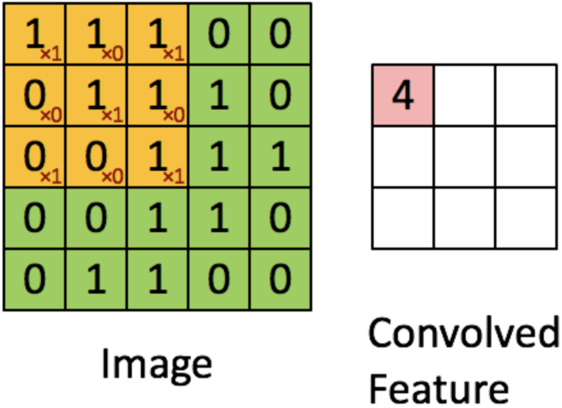
3 components required:

- Input data (e.g. an image)
- Filter/kernel (extracts specific features, such as edges from the input data)
- Feature map (the output from using a filter)

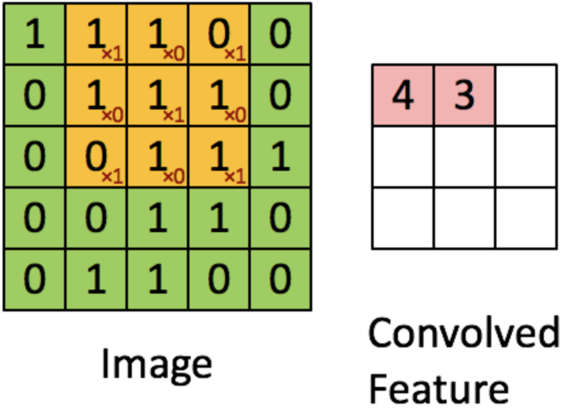
Convolutional Layer(s) - Filter

- 2-dimensional array
- Scan for features in an input (convolution)
- 3×3 (matrix) size is most common
- Filter is applied to an area of image
- Matrix values of filter $\times$ matrix values of area covered by filter (dot product)
- Dot product sent to output array
- Filter shifts
- Process repeated over entire image
- Output: feature map

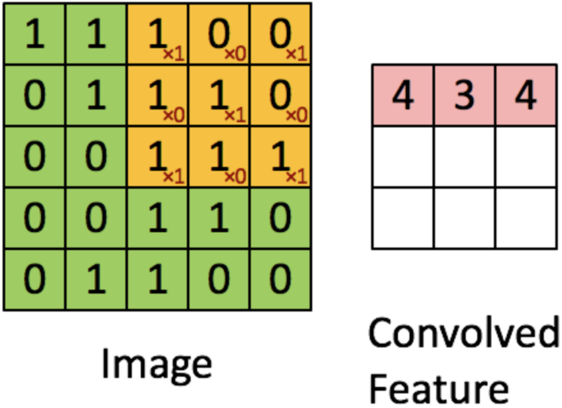
Filter - CNN Layer Type - Visual



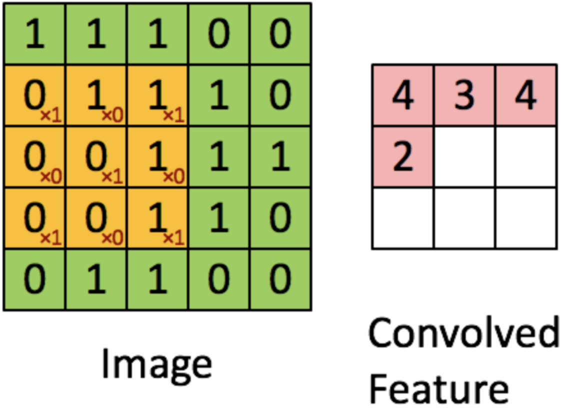
Filter - CNN Layer Type - Visual



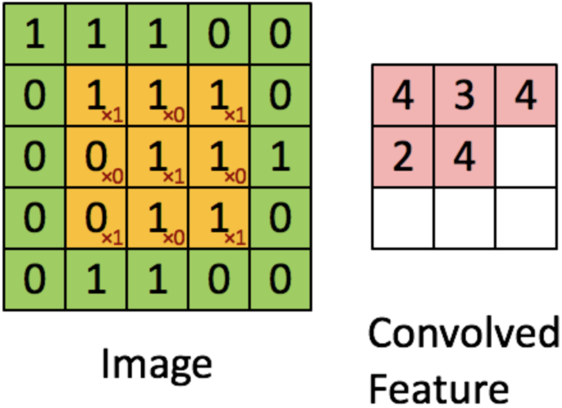
Filter - CNN Layer Type - Visual



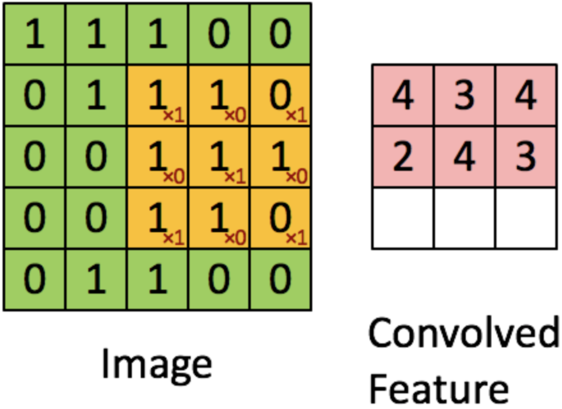
Filter - CNN Layer Type - Visual



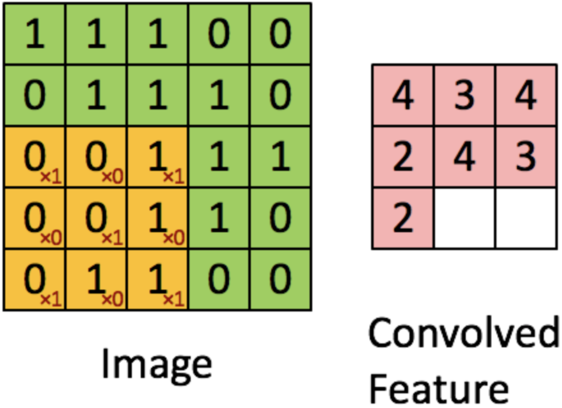
Filter - CNN Layer Type - Visual



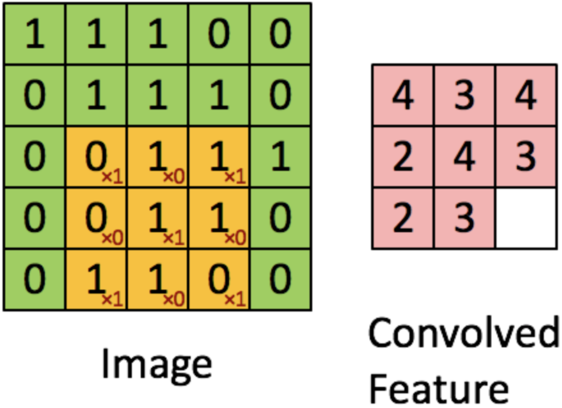
Filter - CNN Layer Type - Visual



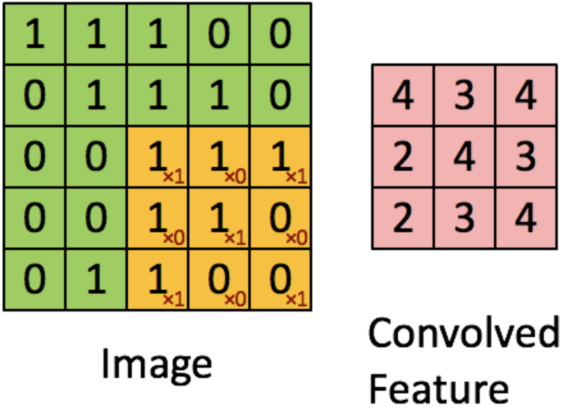
Filter - CNN Layer Type - Visual



Filter - CNN Layer Type - Visual



Filter - CNN Layer Type - Visual



Pooling Layer(s) - CNN Layer Type

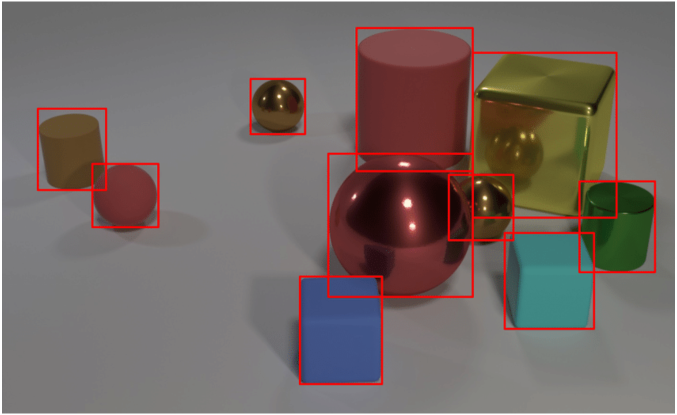
- Perform dimensionality reduction
- Scan filter across an input image
- Clusters values using aggregation
- Giving an output array
- Pooling reduces CNN complexity and improves efficiency

Region Proposal Network (RPN)

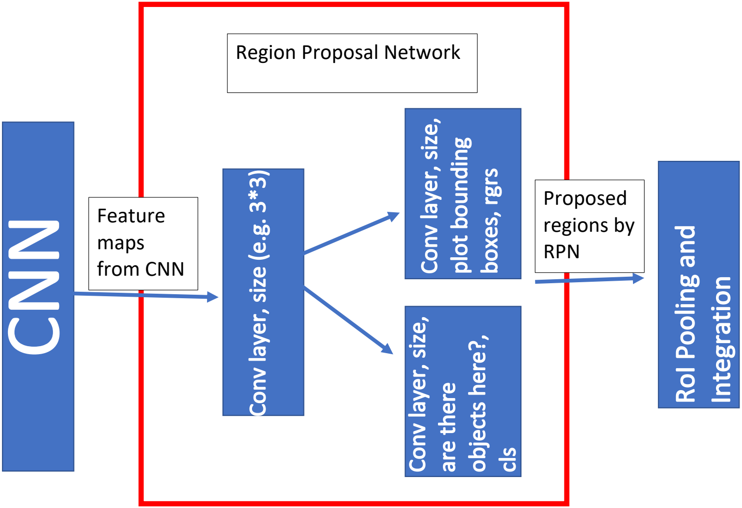
- RPN is an algorithm that identifies certain objects in an image, and places bounding boxes around them.
- Those objects are then “proposed” to the next layer connected to the RPN
- Developed by Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun [8].
- The detection and recognition of license plates model by Li *et al.* is based on RPN

Bounding Boxes

Bounding boxes are outline boxes placed around detected objects.



RPNs - How They Work



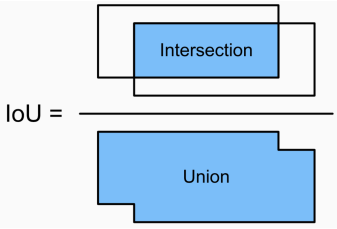
Ground-Truth

In the context of machine learning, ground-truth refers to checking the results of machine learning algorithms, against what is known in real life.

Intersection over Union (IoU)

- A statistic used for measuring the accuracy of object detectors.
- Calculated by dividing the area of intersection between the ground-truth bounding box (R_{gt}) and the detected bounding box (R_{det}) by the area of their union.

$$\text{IoU} = \frac{\text{area}(R_{det} \cap R_{gt})}{\text{area}(R_{det} \cup R_{gt})}$$



Outline

Plate Proposal Generation - Overview

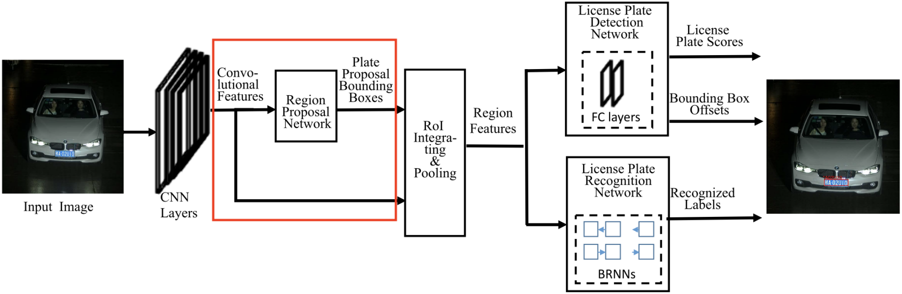


Plate Proposal Generation

- Li *et al.* modified the RPN by Ren *et al.*
- They designed 6 different scales for different license plate sizes
- With $k = 6$ anchors at each position of the input feature maps.

What Are Anchors?

- They are the center points of bounding boxes
- Anchors can be positive or negative
- Determined by IoU scores
- Anchors with IoU scores less than 0.5 are (usually) considered negative
- Anchors with IoU scores greater than 0.5 are (usually) considered positive

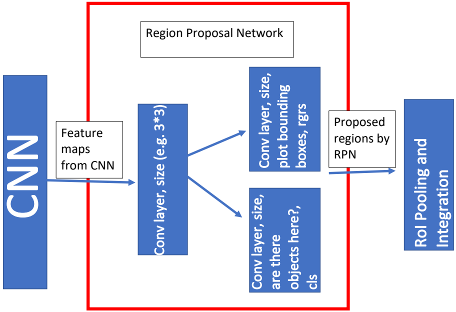
Li *et al.* decided:

- The IoU score used to determine positive anchors is 0.7 or more
- The IoU score used to determine negative anchors is 0.3 or less

Note: More details can be found in the paper.

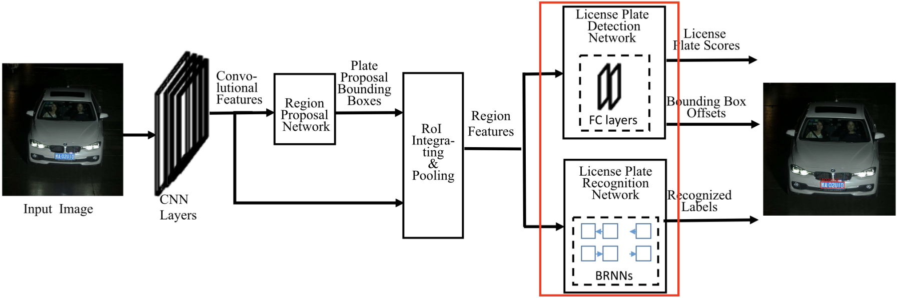
Plate Proposal Generation - cont.

- Two 256-dimensional filters (sizes 5×3 and 3×1)
- Placed simultaneously onto the feature map
- Classification scores that indicate probabilities of the anchors as license plates or not.
- Regression values refer to the offsets of anchor boxes to a nearby ground-truth.



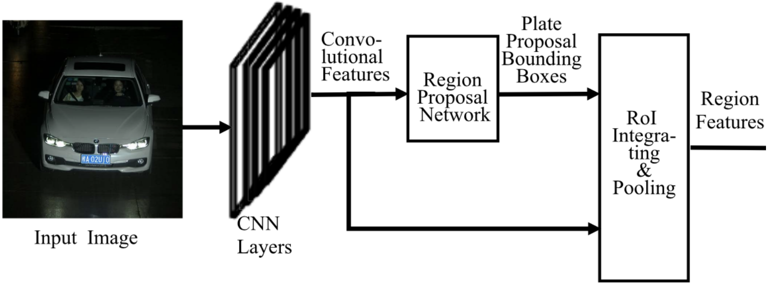
The Problem of Different Proposal Box Sizes

- Proposal boxes output from the RPN come in different sizes
- The license detection and recognition networks (in red box) can only process one box size.

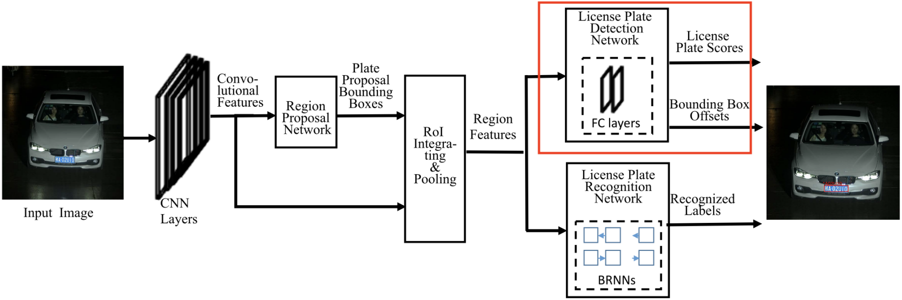


The Solution: RoI Pooling

- RoI pooling is performed to set one size (28×4) for all anchor boxes.
- RoI pooling requires two inputs; the proposals from RPN and the feature map from the initial CNN layers.
- The output of RoI is known as **region features**, which will be used in the next parts of the method by Li *et al.*.

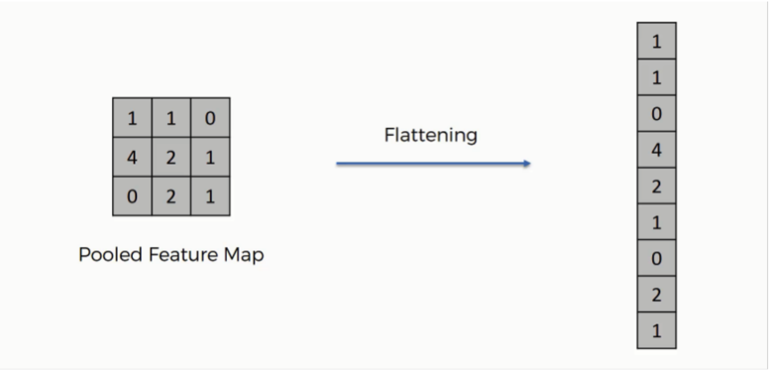


License Plate Detection Network - Overview



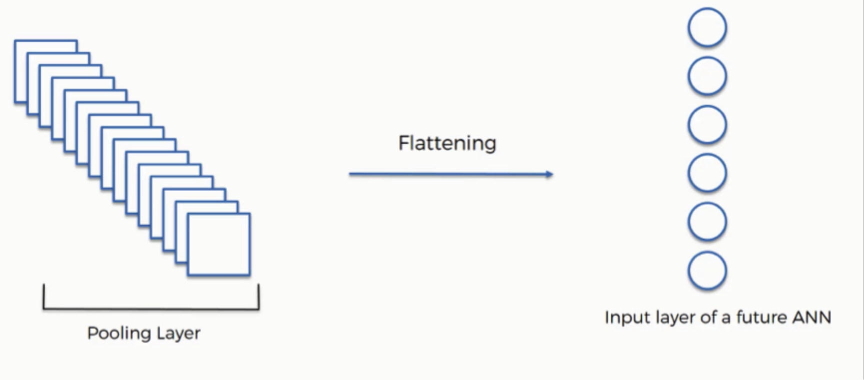
License Plate Detection Network

- The region feature map from RoI pooling is flattened (transformed) into a 1-dimensional a vector
- Then 2 fully connected (FC) layers with 2048 nodes are used to extract distinct features from the 1-dimensional vector.



Why is Flattening Done?

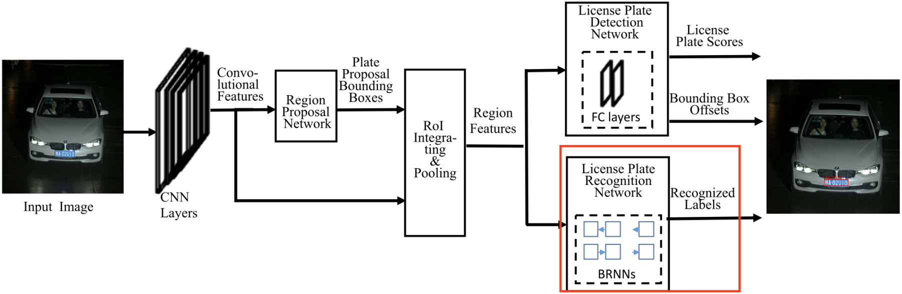
To be able to insert this data into an ANN.



License Plate Detection Network - cont.

- The features are then input into two other neural network layers (one classification and one regression layer)
- Giving two outputs
- License plate bounding box offsets for each proposal
- License plate scores (the probability of each RoI as a plate or non-plate)

License Plate Recognition Network - Overview



License Plate Recognition Network

- Take feature maps from RoI pooling and integration layer
- Inputs them into RNNs
- Connectionist temporal classification (CTC) neural networks are used to deocode license plate characters
- Outputting recognized labels (plate numbers)

Output - Visual

Outline

Test Results from PKUData Dataset

Li *et al.* tested their model using 4 datasets:

- Car license plates from China, known as CarFlag-Large.
- Application-Oriented License Plate (AOLP).
- Caltech-cars (Real) 1999.
- **PKUData by Yuan *et al.* [11]**

Detection Performance Average

Method	Detection Performance Average (%)	Detection Speed Per Image (ms)	End-to-end Speed Per Image (ms)
Zhou *et al.* [12]	90.22	475	-
Li *et al.* (2013) [5]	91.52	672	-
Yuan *et al.* [11]	97.69	**42**	-
Li *et al.* (2019) [Detection Only] [6]	99.51	283	-
Li (2019) *et al.* [Jointly-trained] [6]	**99.73**	279	310

- Li *et al.* test compare their method against 3 others
- Detection performance done on different images with different capturing conditions (such as how distant a plate is from the camera, or the angle at which the image was taken)
- Li *et al*'s method achieves an average detection ratio of 99.73%
- Which is 2% higher than the previous best performance (by Yuan *et al.*)

Detection Speed Per Image

Method	Detection Performance Average (%)	Detection Speed Per Image (ms)	End-to-end Speed Per Image (ms)
Zhou *et al.* [12]	90.22	475	-
Li *et al.* (2013) [5]	91.52	672	-
Yuan *et al.* [11]	97.69	**42**	-
Li *et al.* (2019) [Detection Only] [6]	99.51	283	-
Li (2019) *et al.* [Jointly-trained] [6]	**99.73**	279	310

- Detection speed per image is not the fastest (2nd place).
- Yuan *et al.* method is faster due to the use of simple support vector machines (SVMs) instead of CNNs and RNNs.
- Detection and recognition computational speed of Li *et al.* network is around $310ms$.

Outline

Conclusion

Li *et al.* proposed a jointly-trained network for simultaneously recognizing car license plate detection and recognition. The model achieved good test results, and has high accuracy and efficiency.

Acknowledgments

I would like to thank Nic McPhee and Elena Machkasova for their support, advice, feedback, and dedication to helping with my paper and this talk.

References I

[1] URL: http://ufldl.stanford.edu/tutorial/supervised/FeatureExtractionUsingConvolution/.

[2] *13.4. Anchor Boxes*. URL: http://d2l.ai/chapter_computer-vision/anchor.html.

[3] *Convolutional Neural Networks (CNN): Step 3 - Flattening*. Aug. 2018. URL: https://www.superdatascience.com/blogs/convolutional-neural-networks-cnn-step-3-flattening.

[4] IBM Cloud Education. *What are neural networks?* [Online; accessed 20-Oct-2021]. 2020. URL: https://www.ibm.com/cloud/learn/neural-networks.

References II

[5] Bo Li et al. "Component-based license plate detection using conditional random field model". In: *IEEE Transactions on Intelligent Transportation Systems* 14.4 (2013), pp. 1690–1699.

[6] Hui Li, Peng Wang, and Chunhua Shen. "Toward End-to-End Car License Plate Detection and Recognition With Deep Neural Networks". In: *IEEE Transactions on Intelligent Transportation Systems* 20.3 (2019), pp. 1126–1136. DOI: 10.1109/TITS.2018.2847291.

[7] Rayan Potter. *How Bounding Box Annotation Helps Object Detection in Machine Learning: Use Cases*. May 2020. URL: https://becominghuman.ai/how-bounding-box-annotation-helps-object-detection-in-machine-learning-use-cases-431d93e7b25b.

References III

[8] Shaoqing Ren et al. “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.6 (2017), pp. 1137–1149. DOI: 10.1109/TPAMI.2016.2577031.

[9] Adrian Rosebrock. *Intersection over Union (IoU) for object detection*. [Online; accessed 7-October-2021]. 2017. URL: https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/.

[10] Wikipedia contributors. *Artificial neural network — Wikipedia, The Free Encyclopedia*. [Online; accessed 13-November-2021]. 2021. URL: https://en.wikipedia.org/w/index.php?title=Artificial_neural_network&oldid=1054691619.

References IV

[11] Yule Yuan et al. "A Robust and Efficient Approach to License Plate Detection". In: *IEEE Transactions on Image Processing* 26.3 (2017), pp. 1102–1114. DOI: 10.1109/TIP.2016.2631901.

[12] Wengang Zhou et al. "Principal visual word discovery for automatic license plate detection". In: *IEEE transactions on image processing* 21.9 (2012), pp. 4269–4279.