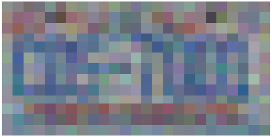
Can you enhance it?

Forensic Reconstruction of Severely Degraded License Plates

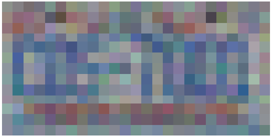
Benedikt Lorch[1], Shruti Agarwal[2], Hany Farid[2]
[1] Friedrich-Alexander University; Erlangen, Germany
[2] Dartmouth College; Hanover, NH, USA

January 14, 2019

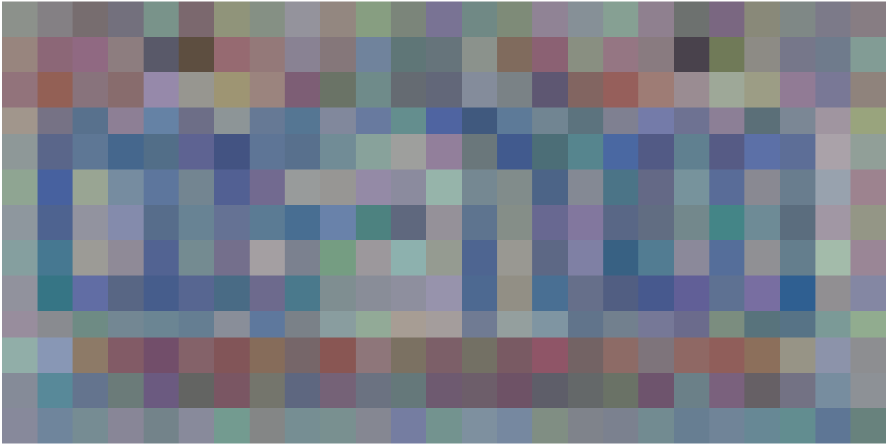
Can you read this?



Can you read this?



Can you read this?



Turning the image enhancement problem on its head

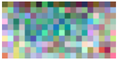
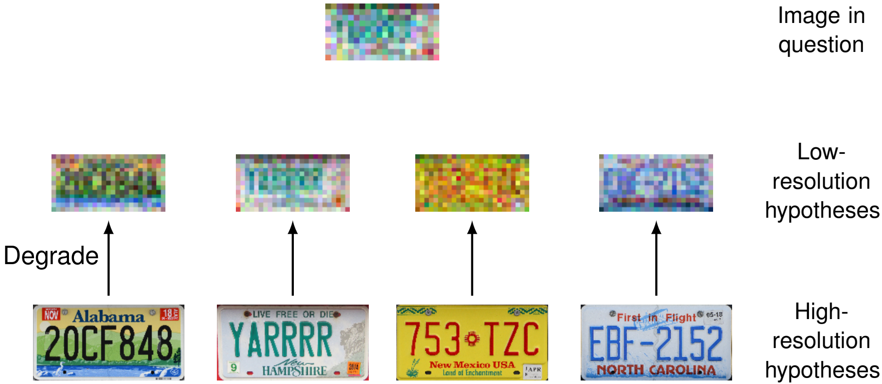


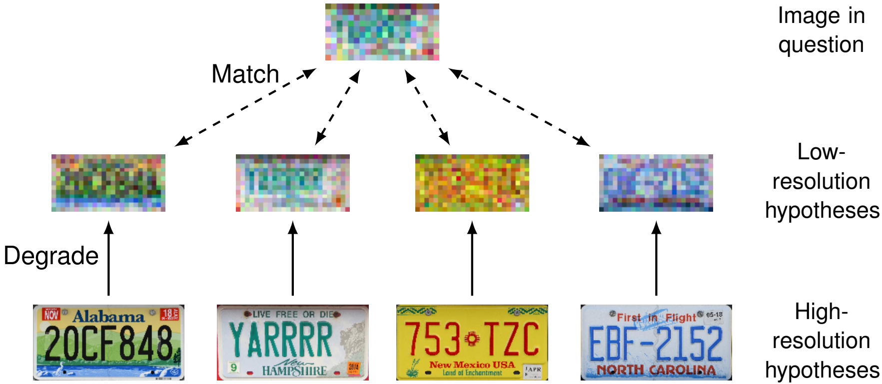
Image in question

High-resolution hypotheses

Turning the image enhancement problem on its head



Turning the image enhancement problem on its head



Conventional recognition[1]

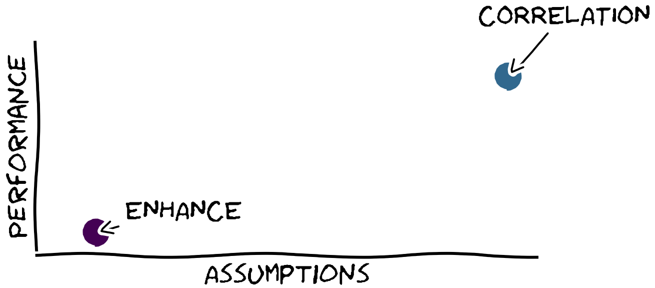
1. License plate detection
2. Character segmentation
3. Character recognition

[1] M. Safraz, M. J. Ahmed, S. A. Ghazi, “Saudi Arabian license plate recognition system”, in *2003 International Conference on Geometric Modeling and Graphics*, Jul. 2003.

Related work

Correlation by Hsieh et al.[2]

- Slide character templates across image
- Drawback: Requires knowledge of font style and character placement

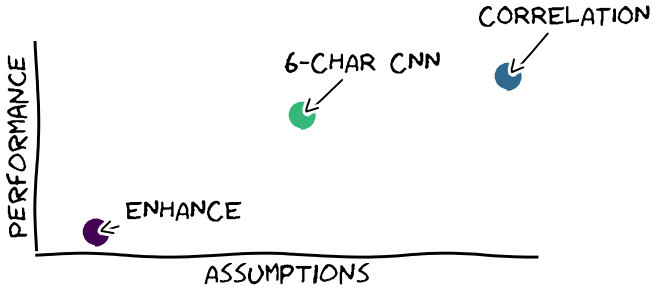


[2] P.-L. Hsieh, Y.-M. Liang, H.Y. M. Liao, "Recognition of blurred license plate images", in *2010 IEEE International Workshop on Information Forensics and Security*, Dec. 2010.

Related work

CNN-based recognition by Agarwal et al.[3]

- Recognize the first and last three characters separately
- Drawback: Assumes six-character license plates

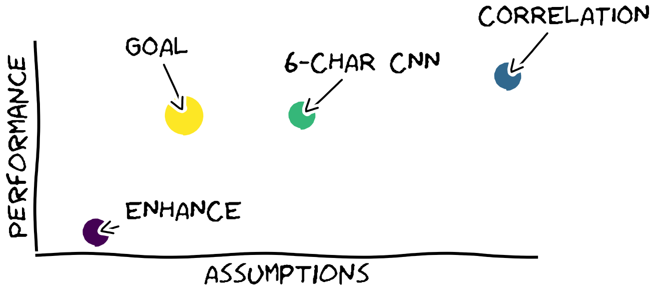


[3] S. Agarwal, D. Tran, L. Torresani, H. Farid, "Deciphering severely degraded license plates", *Electronic Imaging*, Jan. 2017.

Goal

Proposed method

- Work with variable number of characters (5 to 7)



Performance evaluation

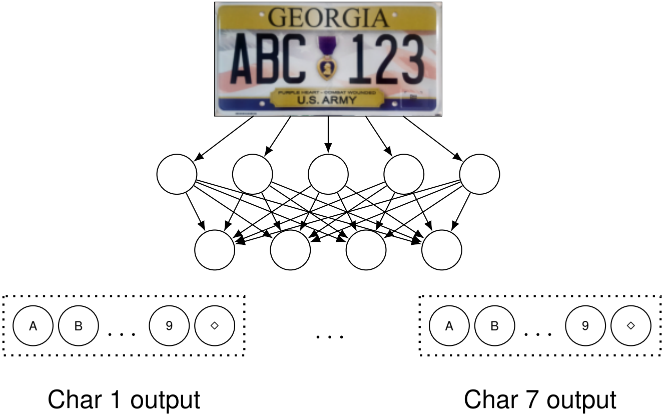
- Fill license number label with null character “◇” up to length seven
- Assign integer score out of a maximum of seven points

True	Predicted	Top-1 accuracy
ABC123◇	ABC123◇	7/7 (100%)
ABC123◇	ABC12◇◇	6/7 (86%)
ABC123◇	ABC1234	6/7 (86%)
ABC123◇	AXXXX3◇	3/7 (43%)
ABC123◇	XABC123	0/7 (0%)

- Report Levenshtein distance to take shifted recognition into account
- Address similar characters, e.g., “O” and “0”, “I” and “1”, by Top-5 accuracy

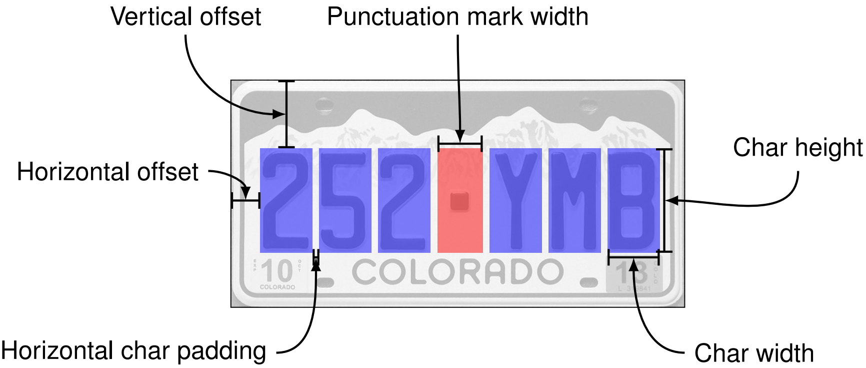
Implementation as CNN architecture I

- VGG-like architecture with seven output layers consisting of 37 units each



Synthetic data rendering

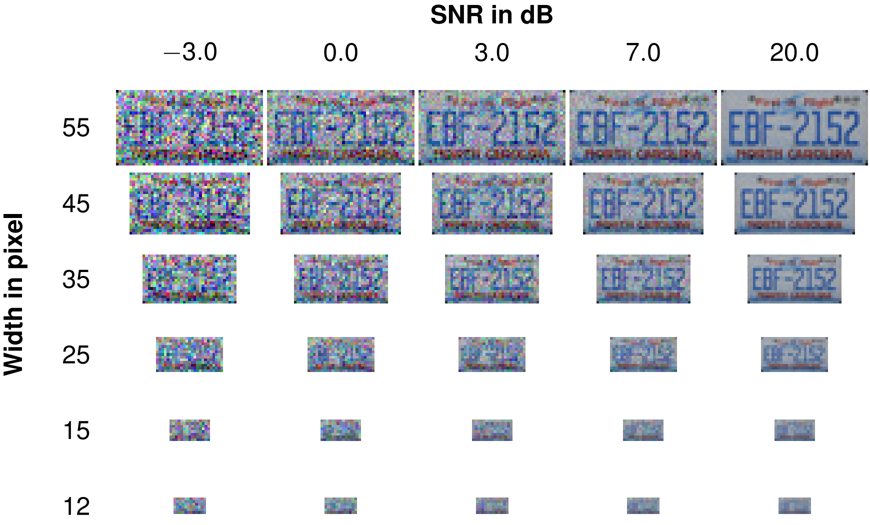
- Optimizing 55M parameters requires a lot of training data
- Generate synthetic training images based on measurements from real-world license plates



Synthetic-I training image examples

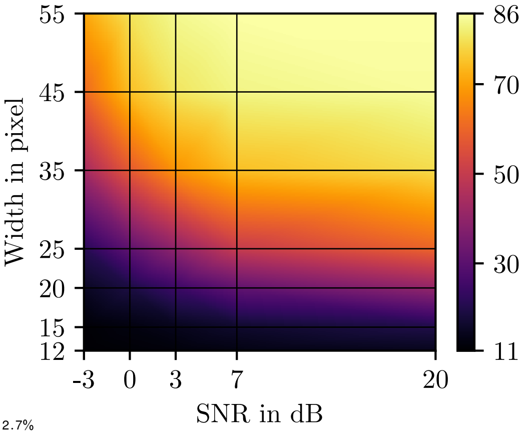
- Vary characters, gap position, font, font size, character placement, contrast, background

Degradation levels



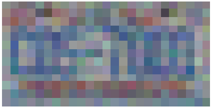
Synthetic-I results on 1,000 real-world images

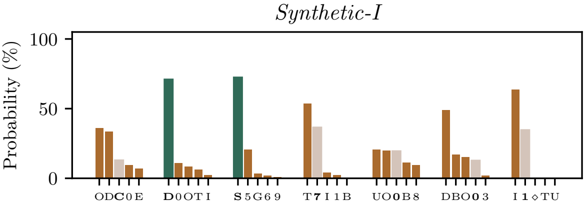
Top-1 accuracies



Chance performance: $1/37 \approx 2.7\%$

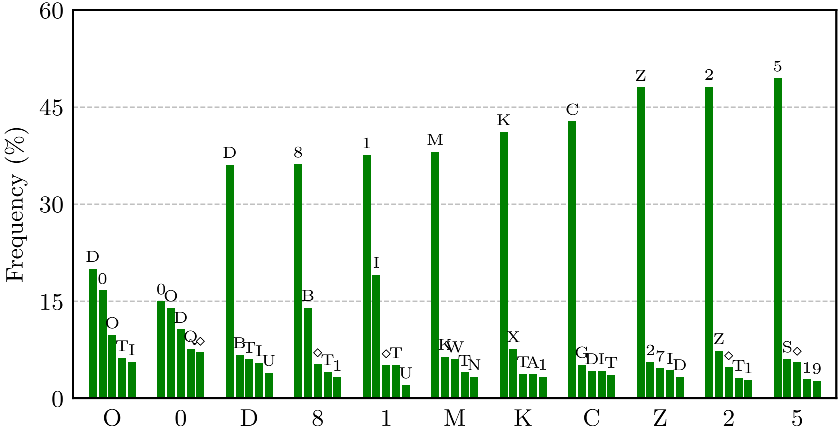
Example: “CDS7001”





Synthetic-I results on 1,000 real-world images

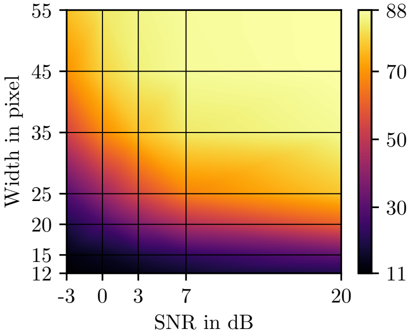
Confusion chart



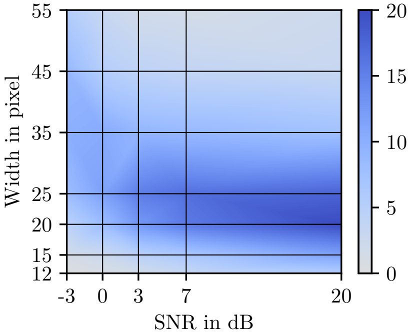
Improving synthetic data rendering

- Place distracting object
- Add embossing effect
- Add frame and drop shadow
- Colorize background, font, and frame and blend with natural images

Synthetic-II results on 1,000 real-world images

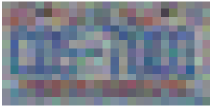


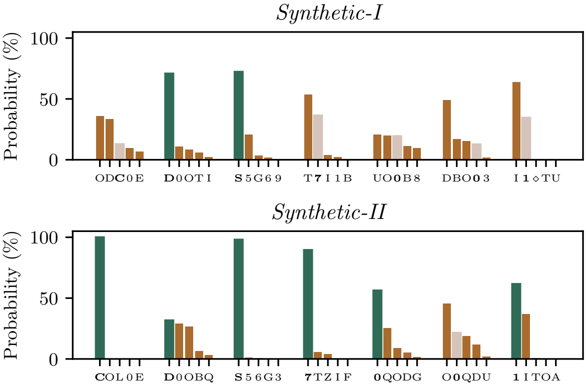
(a) Top-1 accuracies



(b) Difference in Top-1 accuracy compared to *Synthetic-I*

Example: “CDS7001”





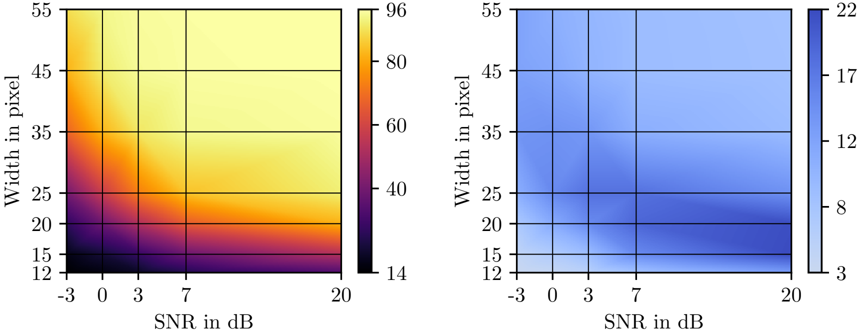
Real-world images from Plateshack

Real-world: Fine-tuning on Plateshack images

- Train on synthetic images, fine-tune all layers on Plateshack images with reduced learning rate

Real-world: Fine-tuning on Plateshack images

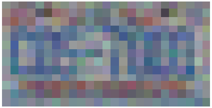
- Train on synthetic images, fine-tune all layers on Plateshack images with reduced learning rate

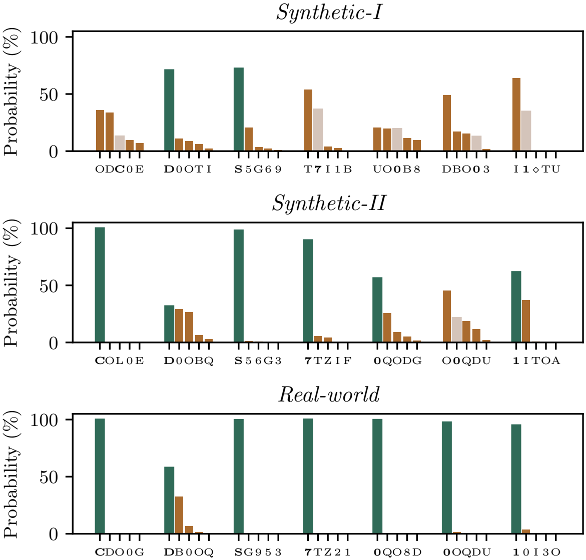


(a) Top-1 accuracies

(b) Difference in Top-1 accuracy compared to *Synthetic-II*

Example: “CDS7001”

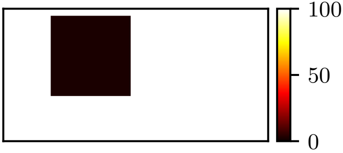




Which image parts are important?

Idea: Highlight important regions for recognition by sliding occluding patch across image

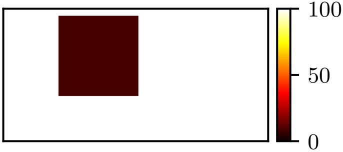
What is the probability that the first character is “E”?



Which image parts are important?

Idea: Highlight important regions for recognition by sliding occluding patch across image

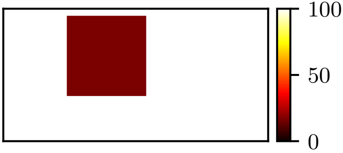
What is the probability that the first character is “E”?



Which image parts are important?

Idea: Highlight important regions for recognition by sliding occluding patch across image

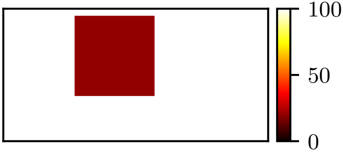
What is the probability that the first character is “E”?



Which image parts are important?

Idea: Highlight important regions for recognition by sliding occluding patch across image

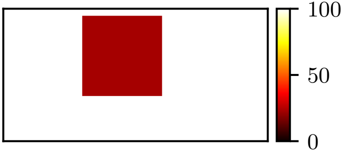
What is the probability that the first character is “E”?



Which image parts are important?

Idea: Highlight important regions for recognition by sliding occluding patch across image

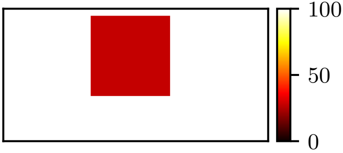
What is the probability that the first character is “E”?



Which image parts are important?

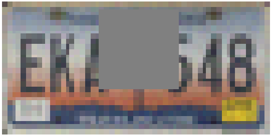
Idea: Highlight important regions for recognition by sliding occluding patch across image

What is the probability that the first character is “E”?

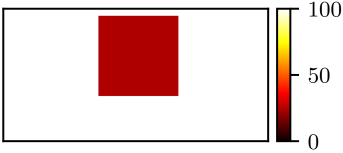


Which image parts are important?

Idea: Highlight important regions for recognition by sliding occluding patch across image



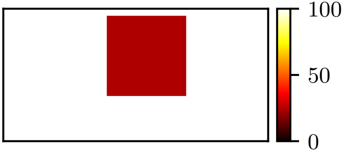
What is the probability that the first character is "E"?



Which image parts are important?

Idea: Highlight important regions for recognition by sliding occluding patch across image

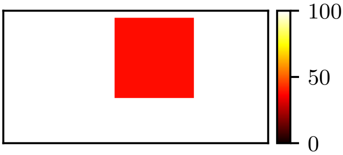
What is the probability that the first character is “E”?



Which image parts are important?

Idea: Highlight important regions for recognition by sliding occluding patch across image

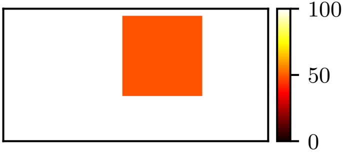
What is the probability that the first character is “E”?



Which image parts are important?

Idea: Highlight important regions for recognition by sliding occluding patch across image

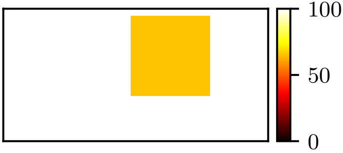
What is the probability that the first character is “E”?



Which image parts are important?

Idea: Highlight important regions for recognition by sliding occluding patch across image

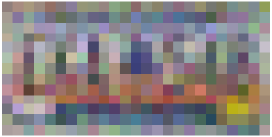
What is the probability that the first character is “E”?



Which image parts are important?

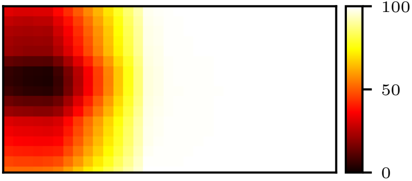
Idea: Highlight important regions for recognition by sliding occluding patch across image

License number “EKA548”, resolution 25 pixels, SNR 7 dB



True character probability

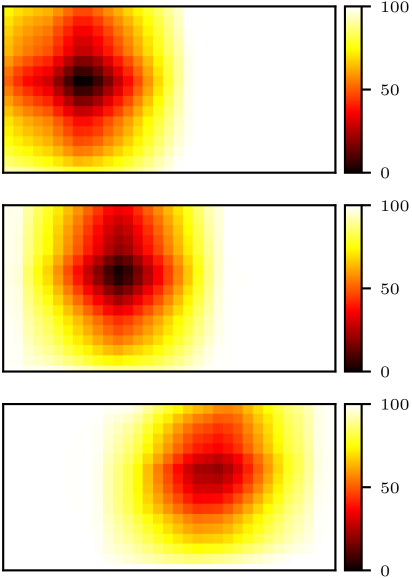
(1)
E



(2)
K

(3)
A

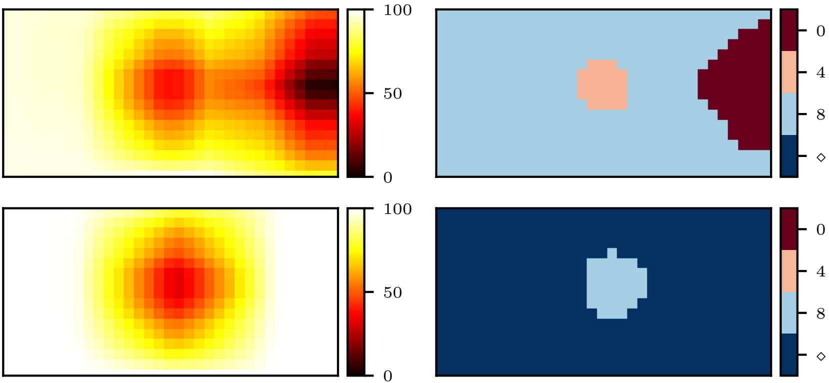
(4)
5



Most likely charаracter

(6)
8

(7)
◇



Conclusion

- Even from low-resolution images we can extract useful information
- CNN outperforms humans at low-resolution and high amounts of noise
- Performance increases with more realistic training data

Limitations

- Bad performance on special license plate configurations
- CNN not robust to unknown degradations, e.g., motion blur

Outlook

- Include more types of degradation
- Better training data generation, e.g., using GANs
- Train combined denoising and recognition CNN end-to-end

github.com/btlorch/license-plates

Image sources

- https://9gag.com/gag/23853/enhance-that-license-plate
- http://plateshack.com/y2k/