MaskingNet for Car Image Masking

Yini Duanmu

MS Defense

Advisor: Dr. Yi Shang

Outline

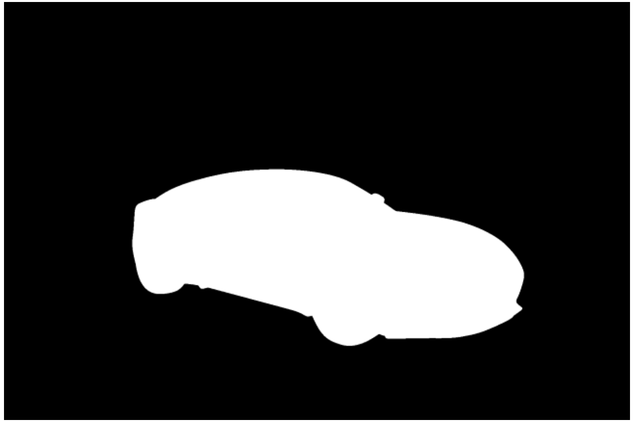
1. Research Problem Description
2. Existing semantic segmentation approaches
3. MaskingNet: A new deep learning architecture
4. Experimental results
5. Summary

Research Problem Description

- The car image masking problem
- Carvana Image Masking Challenge - Kaggle Data Science Competition
- Kaggle Dataset
- Technical Challenges

The car image masking problem

Input image



Target

Background: Online store segment out the products to display on website
Task: Automatically remove the background, keep the foreground

Carvana Image Masking Challenge - Kaggle

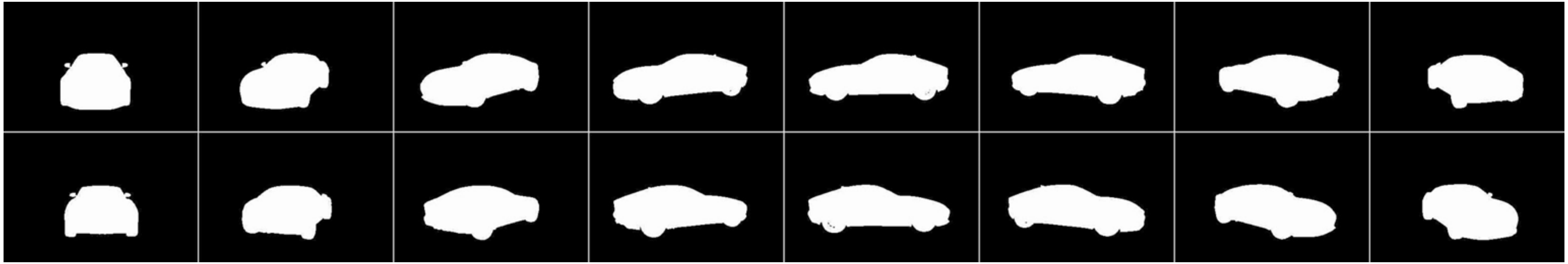
- Time: 7/26/2017~9/27/2017
- My submission ranked 12th / 735, 1st place of silver medal - my proposed Architecture #1 was used

#	△pub	Team Name	Kernel	Team Members	Score	Entries	Last
1	▲1	best[over]fitting			0.997332	80	1y
2	▼1	bestfitting			0.997331	78	1y
3	▲1	lyakaap			0.997264	43	1y
4	▲3	80 TFlops			0.997232	82	1y
5	▲7	Kyle			0.997209	59	1y
6	▼3	JbestDeepGooseFlops			0.997190	76	1y
7	▲1	deepsystems.io			0.997151	12	1y
8	▲17	jizs			0.997138	16	1y
9	▲13	lizy			0.997126	25	1y
10	▲20	David			0.997123	65	1y
11	▼5	Sukjae Cho			0.997115	42	1y
12	▲17	Onion x Potato			0.997085	20	1y

Kaggle Dataset

- Each vehicle has 16 images in different orientation
- Fixed camera position
- Different color, year, make, model combinations
- Training set: 5,088 images (318 vehicles)
- Test set: 100,064 images (6,254 vehicles)

Input images

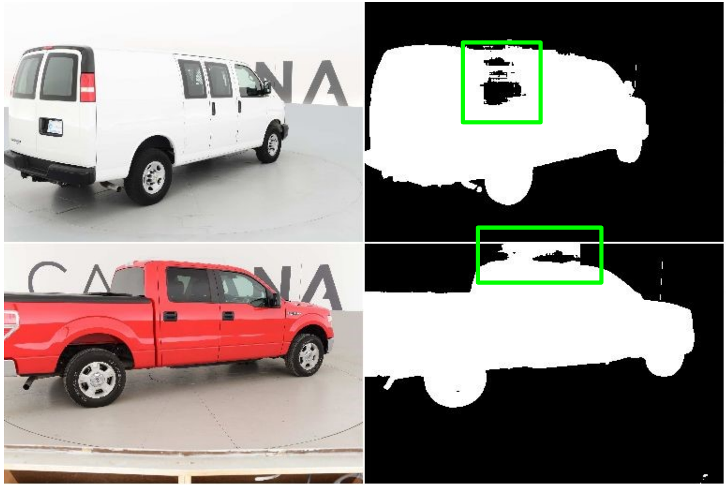


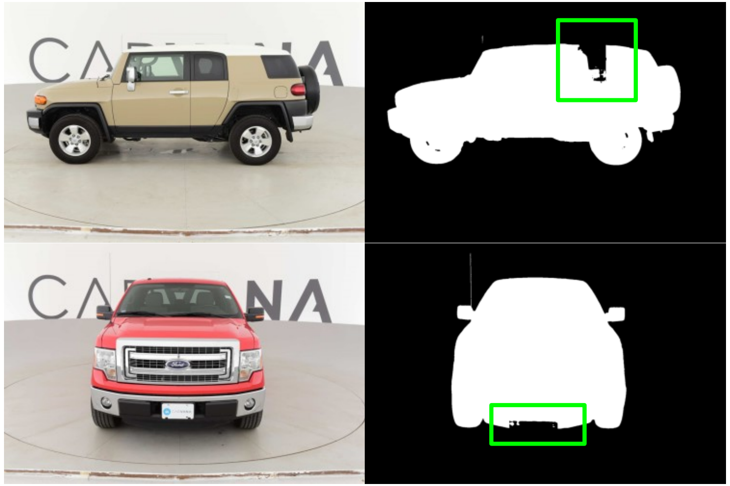
Target

Technical Challenges

- Foreground and background may have similar color
- Need to infer the boundary based on contextual info.

Issues in results generated by SegNet:





Existing semantic segmentation approaches

Bottom-up image segmentation followed by Region Classification

- SDS
- Zoom-Out
- RCNN

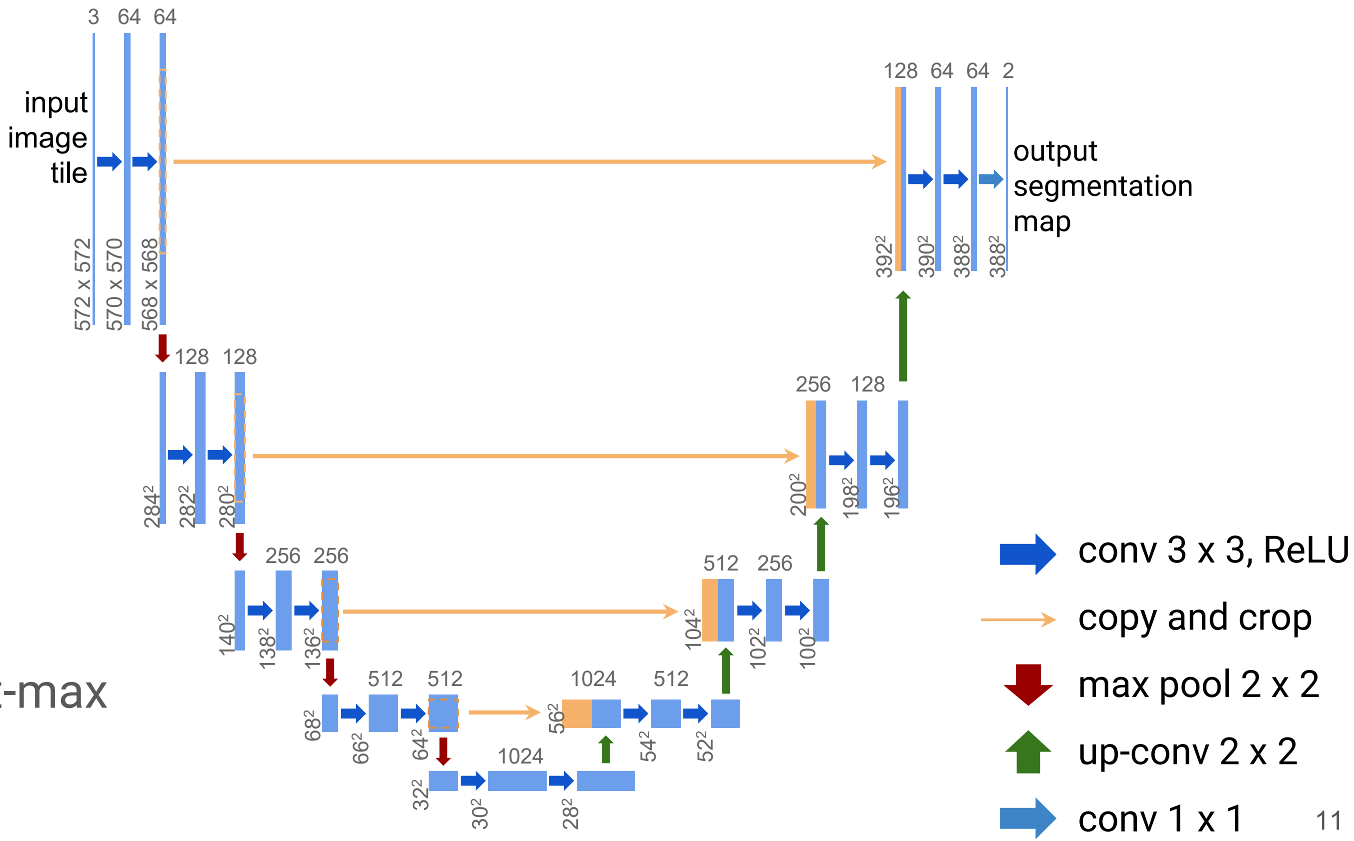
Independently extract CNN features and do segmentation, then couple the result

- Hypercolumns
- Convolutional Feature Masking

Directly do pixel-wise classification

- U-Net
- SegNet
- MaskingNet - a new approach proposed in my project

Architecture of U-Net



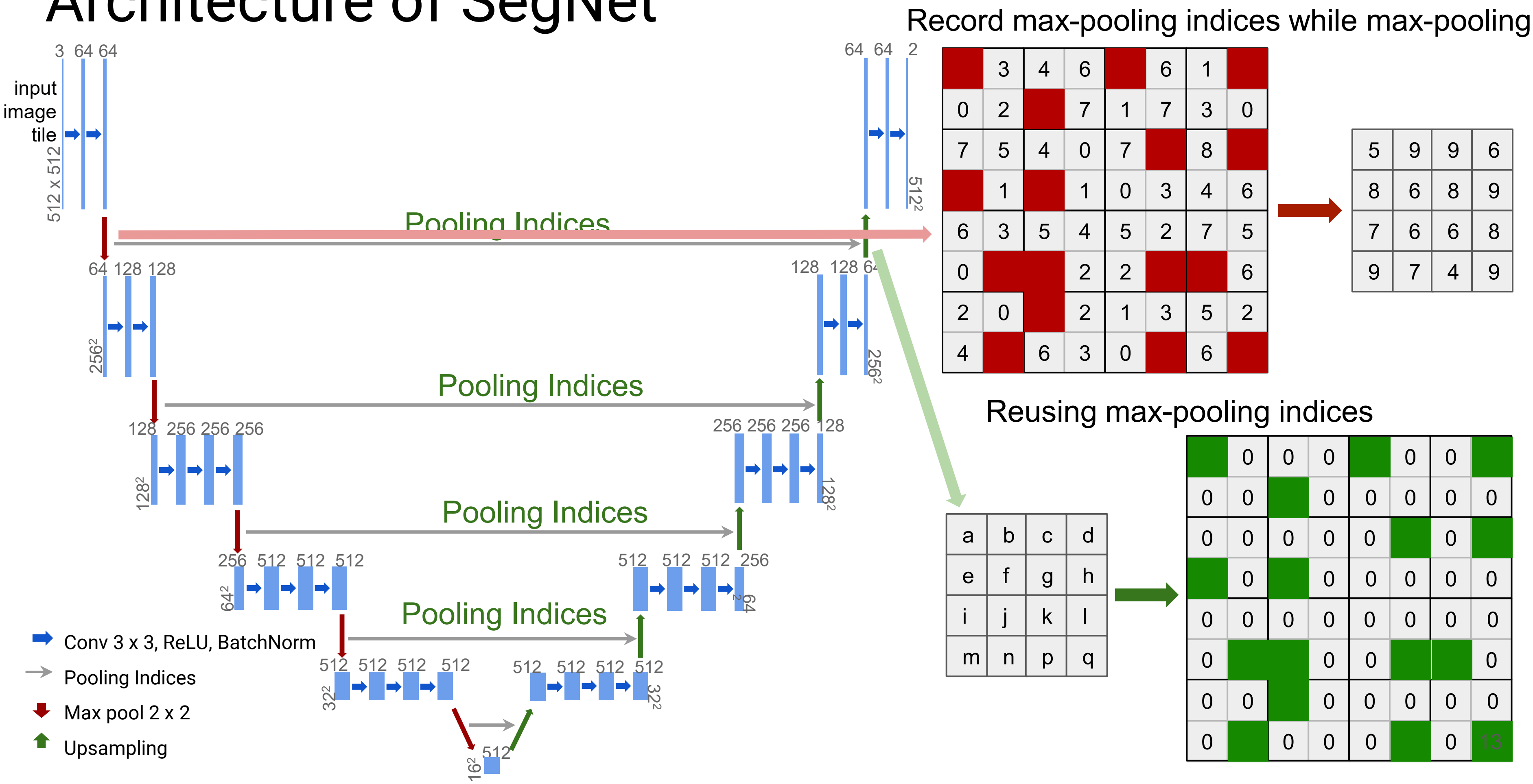
Loss function:

- Pixel-wised soft-max
- Cross Entropy

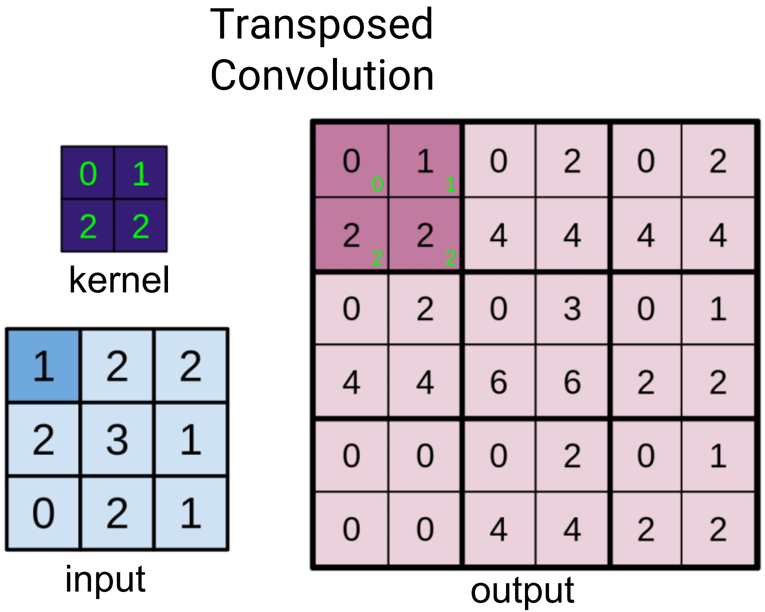
Disadvantages in U-Net

1. The spatial information loss lead by the Max-pooling
2. Lack of the ability to learn the correlation between the content and coordinates due to translation invariant.
3. Unet may not make good use of contextual information due to the limited receptive field.

Architecture of SegNet

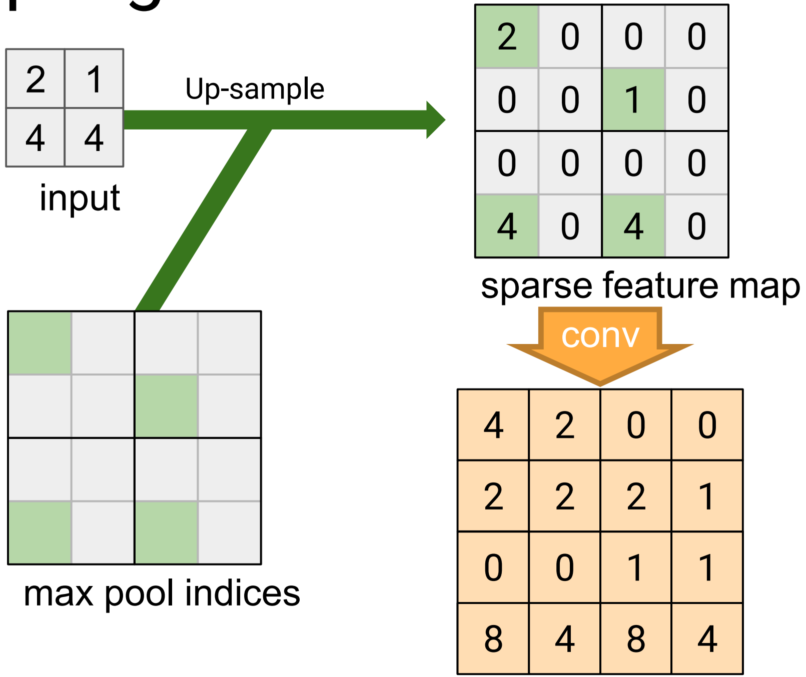


Comparison between upsampling methods



Up-sampling method of U-Net

Spatial info. lost due to the downsample, can never be recovered during the up-sampling



Up-sampling method of SegNet

Spatial info. is kept by reusing the pooling indices.

MaskingNet uses this method

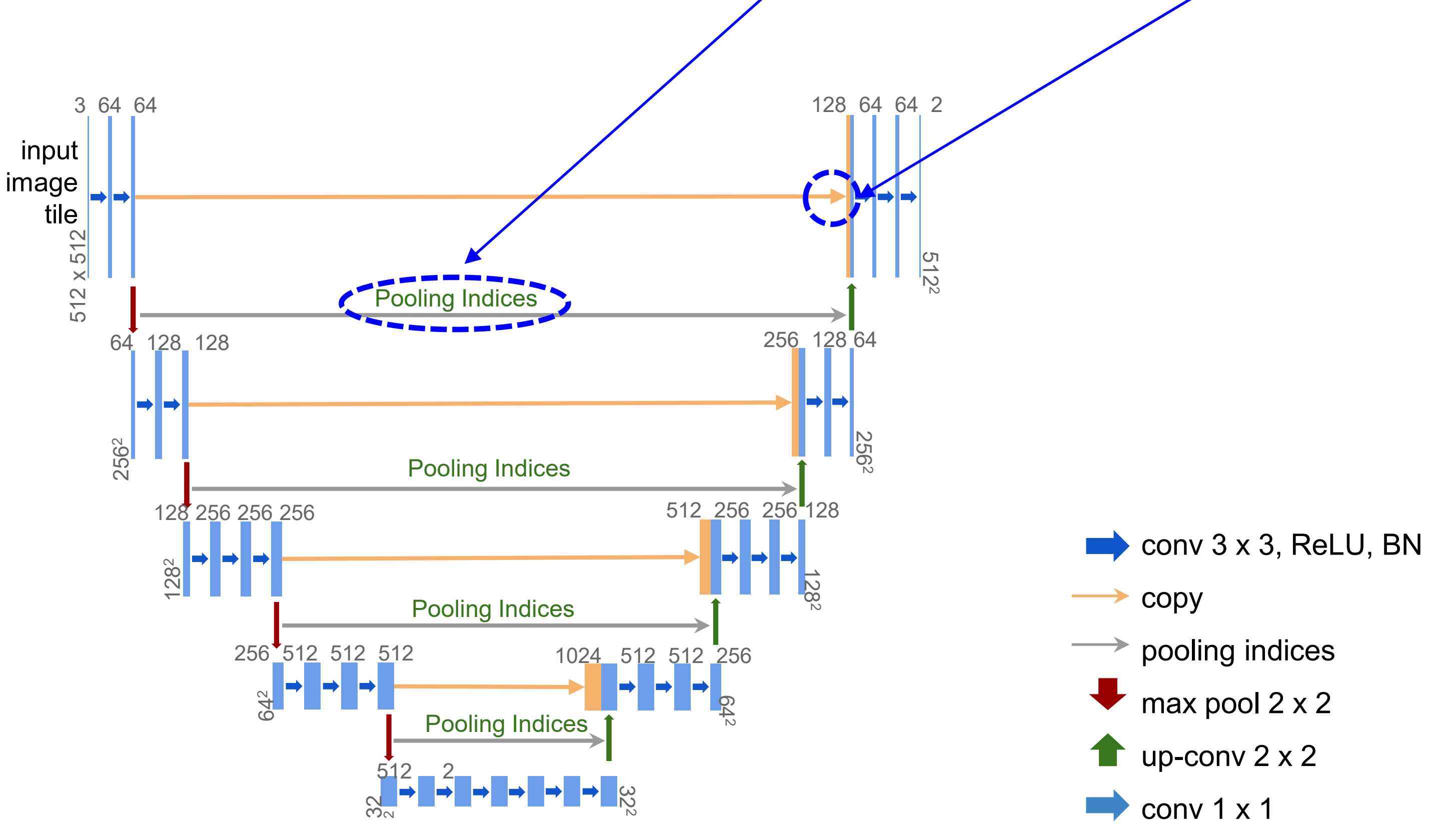
MaskingNet: a new deep learning architecture

1. New Combination of reusing pooling indices and concat. Encoder feature map
2. New Coordinate Maps method
3. New Dilated Convolution method

Model	**Max pool. indices**	**Coord. maps**	**Dilated Conv.**
Arc. #1	√		
Arc. #2	√	Input layer	
Arc. #3	√	Mid-layer	
MaskingNet	√	Mid-layer	√

My Architecture #1:

Combination of reusing pooling indices and concat. encoder feature map



New Coordinate Maps Method

1 Insight
2 Definition of Coordinate maps
3 My Architecture #2

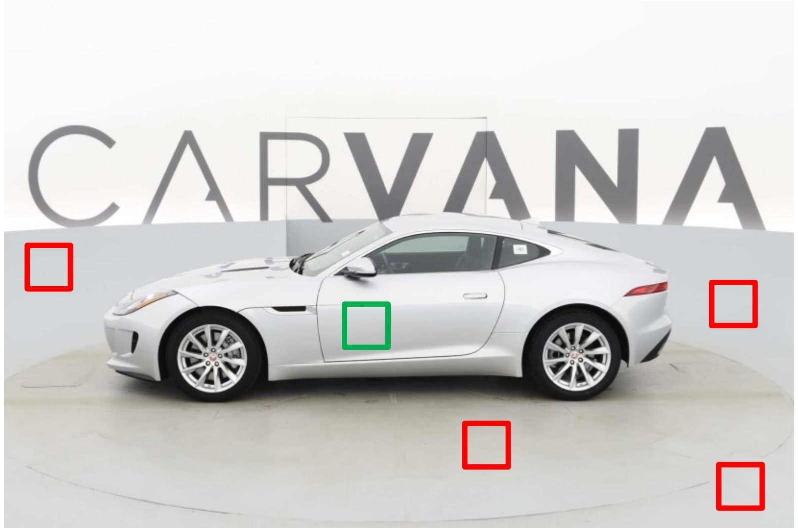
- New combination of reusing pooling indices and concat. encoder feature map
- New concat. coord. maps with input image

4 My Architecture #3

- New combination of reusing pooling indices and concat. encoder feature map
- New concat. coord. maps with mid-layer feature map

1. Insight

- Different parts have different spatial distributions



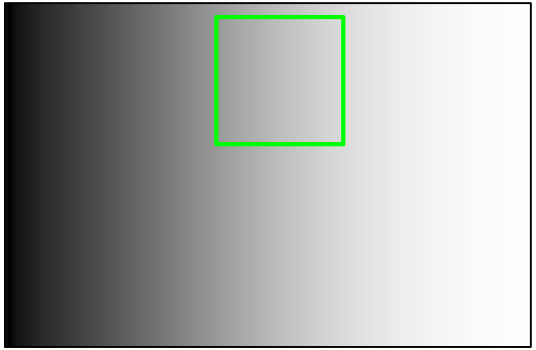
Foreground?

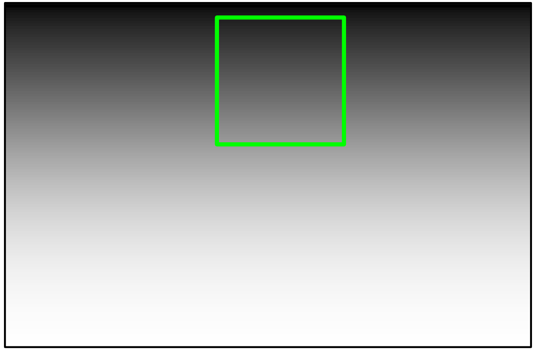
Background?

2. Definition of Coordinate maps

- Define X-Map as $I_X(x, y) = \frac{x}{w} - 0.5$, where w is the width of the image
- Define Y-Map as $I_Y(x, y) = \frac{y}{h} - 0.5$, where h is the height of the image
- Coordinate maps are constant, independent with the input image
- Subimage are cropped at the same position from input image, X-Map and Y-Map

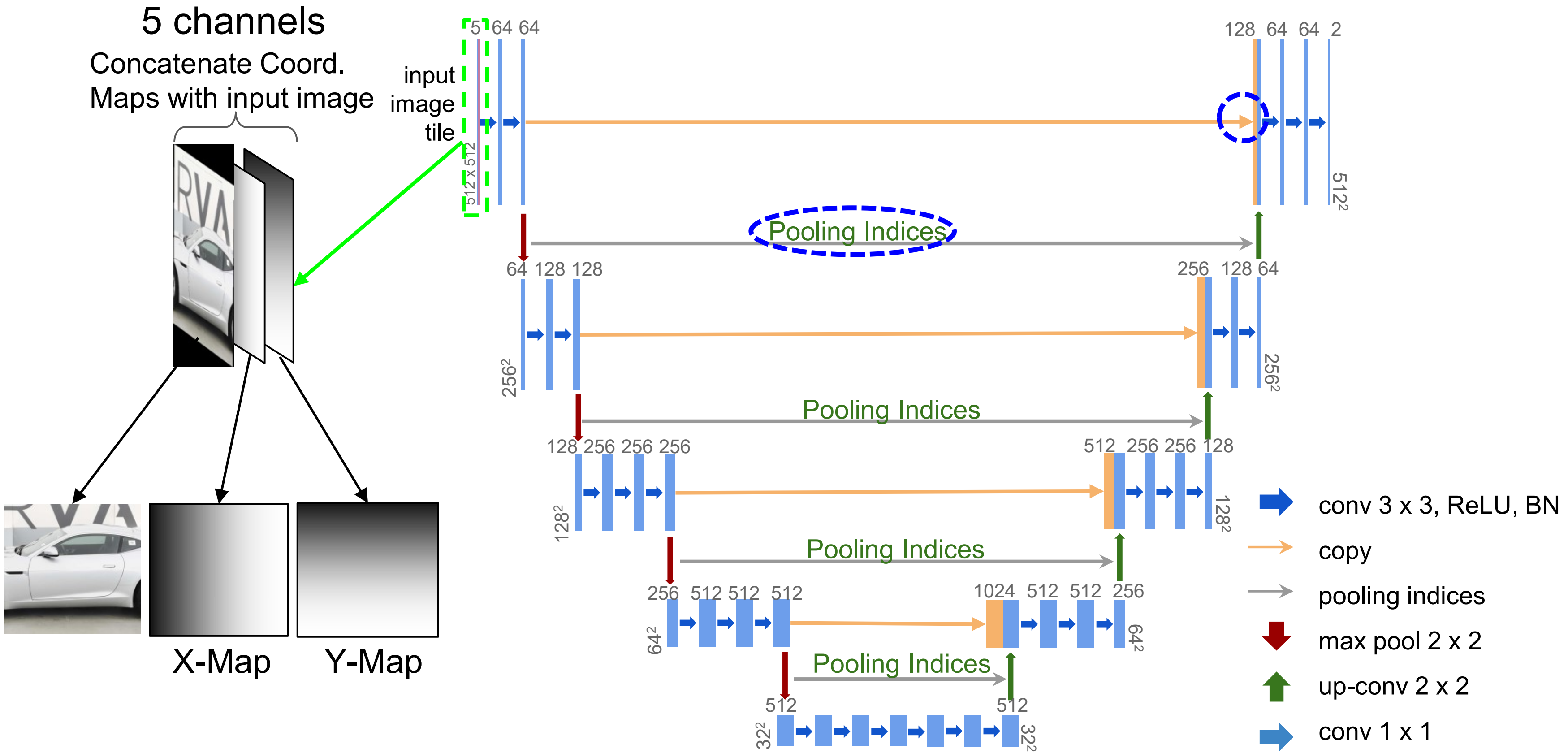
Input image


X-Map
The pixel value is its x coordinate


Y-Map
The pixel value is its y coordinate

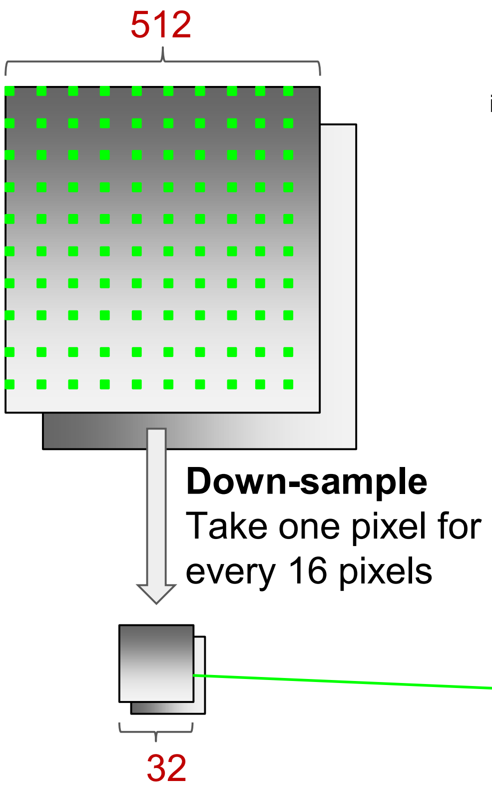
3. My Architecture #2

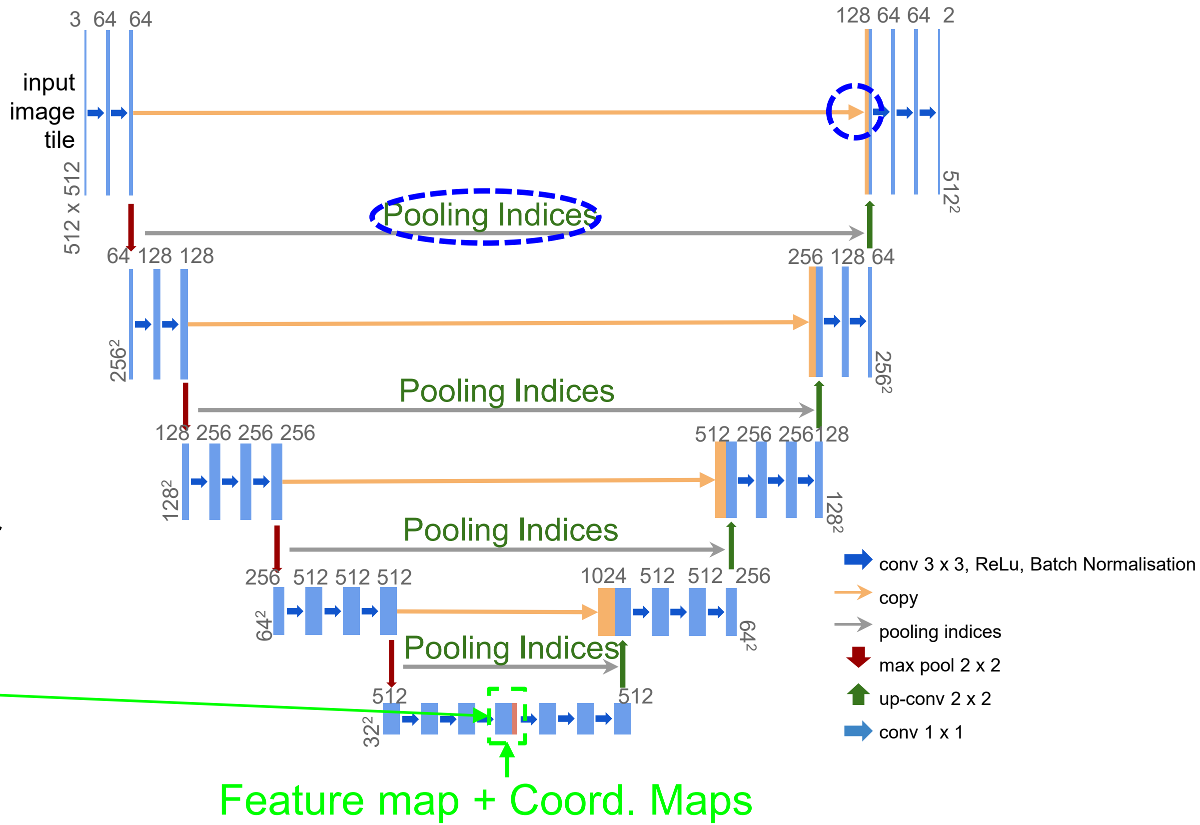
- New combination of reusing pooling indices and concat. encoder feature map
- New concat. coord. maps with input image



4. My Architecture #3

- New combination of reusing pooling indices and concat. encoder feature map
- New concat. coord. maps with mid-layer feature map

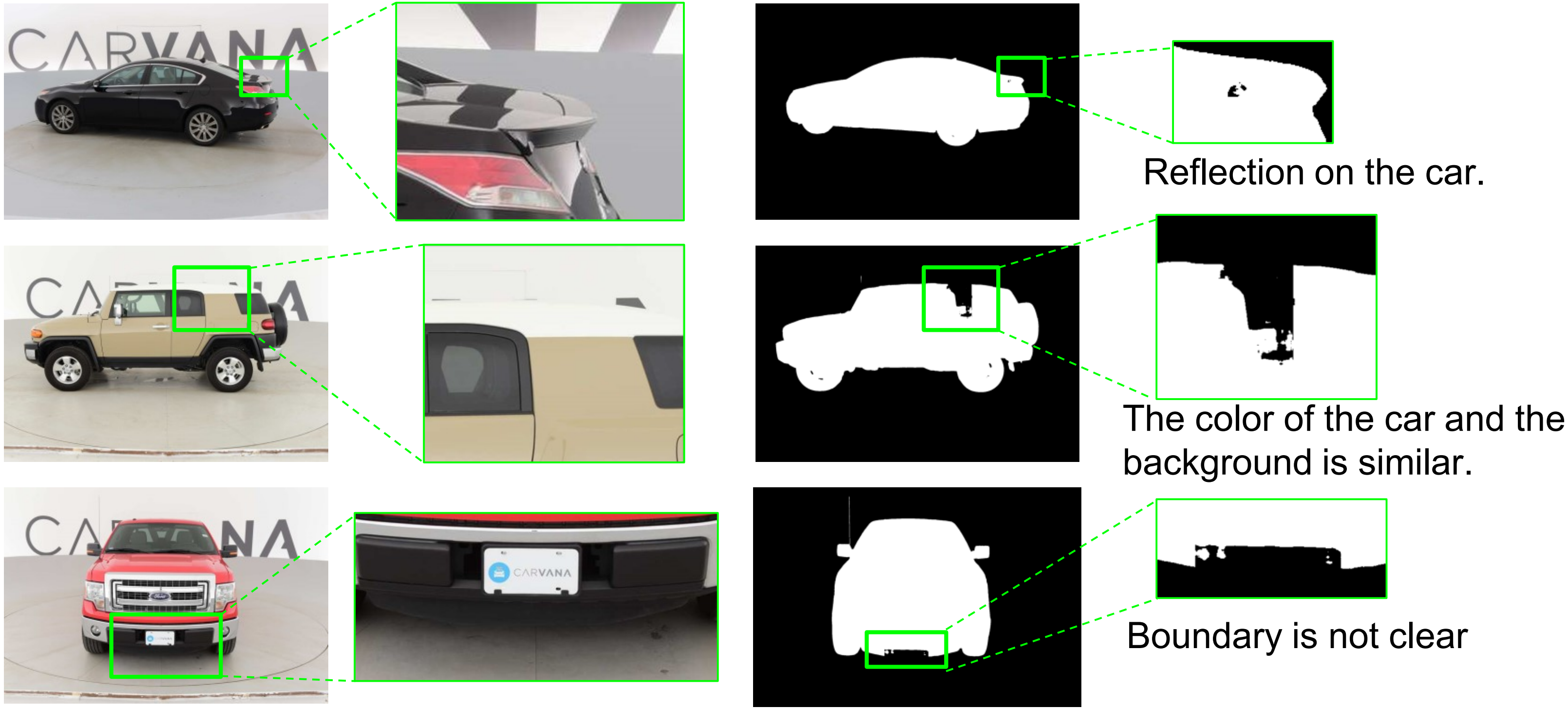




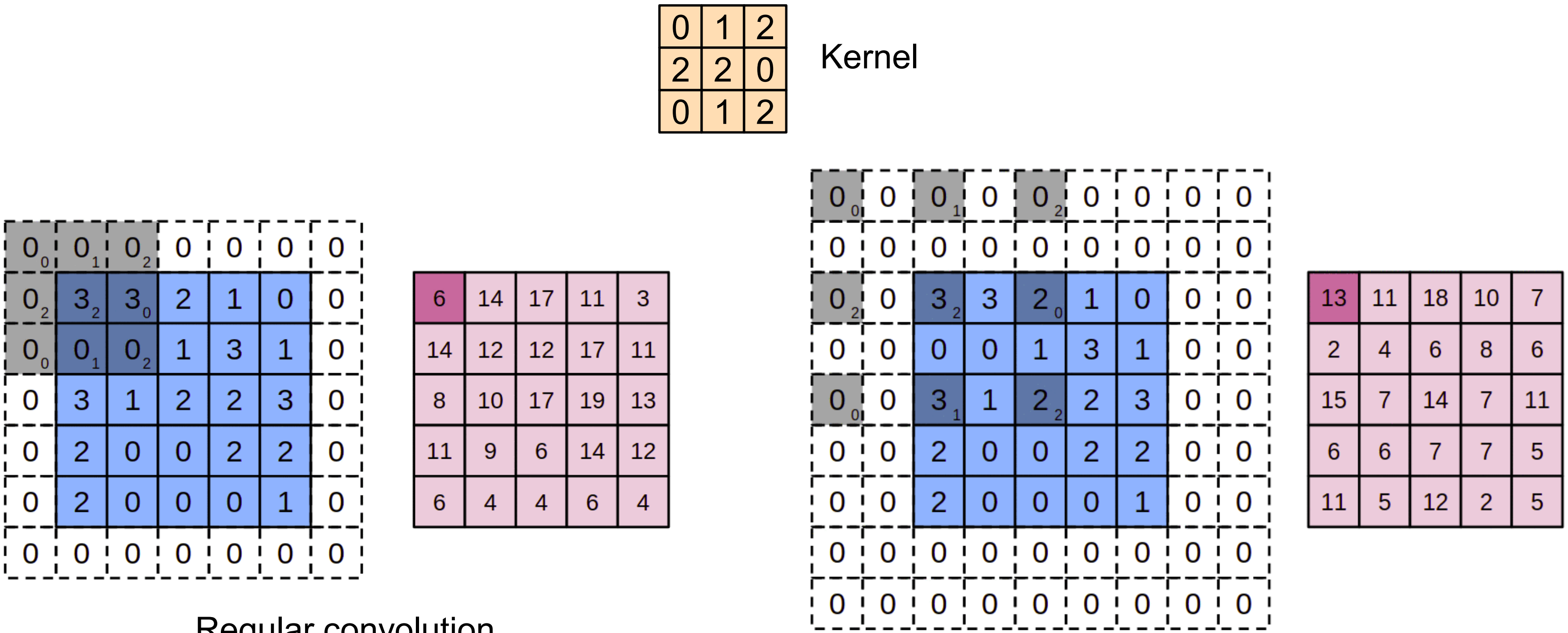
New Dilated Convolution Method

Motivation: Previous methods not good

Can’t infer the car shape due to the limited receptive field



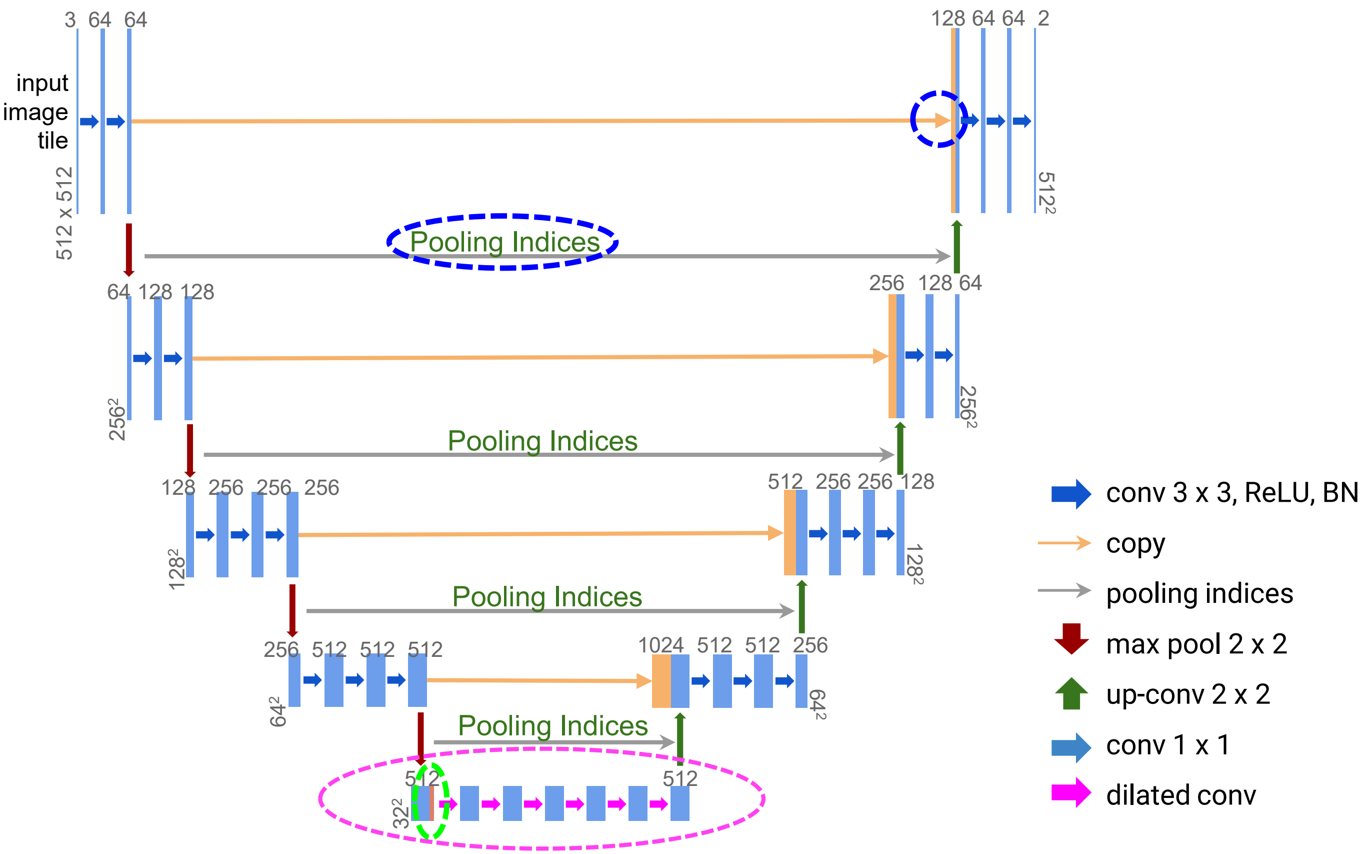
Regular Convolution vs. Dilated Convolution



Regular convolution
stride=1, padding=1

Dilated convolution
dilated=2, stride=1, padding=2

MaskingNet



3 64 64
input image tile
512 x 512
128 64 64 2
512^2
Pooling Indices
64 128 128
256^2
256 128 64
256^2
Pooling Indices
128 256 256 256
128^2
512 256 256 128
128^2
Pooling Indices
256 512 512 512
64^2
1024 512 512 256
64^2
Pooling Indices
512
32^2
512
conv 3 x 3, ReLU, BN
copy
pooling indices
max pool 2 x 2
up-conv 2 x 2
conv 1 x 1
dilated conv

Experimental Results

1. Training Parameters
2. Evaluation metric
3. Experimental results

1. Training Parameters

- Training set: 4071 images
- Validation set: 1017 images
- First 10 layers: pre-trained VGG-16
- Image size: 512x512
- Batch size: 4
- Optimizer: Adaptive Moment Estimation(Adam)
- Learning rate: initial 0.001, half for each 40 epochs
- Momentum: 0.9
- Total epochs: 250#
- Framework: Pytorch
- NVIDIA 1080Ti GPU, cuDNN v7

2. Evaluation metric

- Mean Dice Similarity Coefficient

Compare the agreement between predicted segmentation and ground truth

$$DSC = \frac{2 \times |X \cap Y|}{|X| + |Y|}$$

X: Predicted set of pixels
Y: Ground truth
The DSC is defined to be 1 with both X and Y are empty

3. Experimental results

Model	DSC Test set	Max pool. indices	Coord. maps	Dilated Conv.	#Parameters	Training Time ms/image	Test Time ms/image
U-Net	99.5604%				31,043,586	137.6	34.42
SegNet	99.6092%	√			29,444,162	165.13	30.50
Arc. #1	**99.6932%**	√			32,799,554	171.75	42.793
Arc. #2	99.6870%	√	Input layer		32,800,706	172.59	42.901
Arc. #3	**99.6937%**	√	Mid-layer		32,808,770	170.50	42.839
MaskingNet	**99.7023%**	√	Mid-layer	√	32,808,770	180.46	46.609

- Most of the pixels can be easily predicted gives the DSC very high number such as 99.56%, 99.60%
- 0.01% DSC difference/image ≈ 60 pixels difference/image(The car approximately occupied 600,000 pixels in each image)
- Much more easy case than hard case, so there are significance differences between hard cases with the difference of 0.01% DSC

#	△pub	Team Name	Kernel	Team Members	Score	Entries	Last
1	▲1	best[over]fitting			0.997332	80	1y
2	▼1	bestfitting			0.997331	78	1y
3	▲1	lyakaap			0.997264	43	1y
4	▲3	80 TFlops			0.997232	82	1y
5	▲7	Kyle			0.997209	59	1y
6	▼3	JbestDeepGooseFlops			0.997190	76	1y
7	▲1	deepsystems.io			0.997151	12	1y
8	▲17	jizs			0.997138	16	1y
9	▲13	lizy			0.997126	25	1y
10	▲20	David			0.997123	65	1y
11	▼5	Sukjae Cho			0.997115	42	1y
12	▲17	Onion x Potato			0.997085	20	1y

Improvement of the Up-sampling Method

Model	**1-DSC Test set**	**Max pool. indices**	**Coord. maps**	**Dilated Conv.**	**#Parameters**	**Training Time** ms/image	**Test Time** ms/image
U-Net	0.4396%				31,043,586	137.6	34.42
SegNet	0.3908%	√			29,444,162	165.13	30.50
Arc. #1	**0.3068%**	√			32,799,554	171.75	42.793
Arc. #2	0.313%	√	Input layer		32,800,706	172.59	42.901
Arc. #3	**0.3063%**	√	Mid-layer		32,808,770	170.50	42.839
MaskingNet	**0.2977%**	√	Mid-layer	√	32,808,770	180.46	46.609

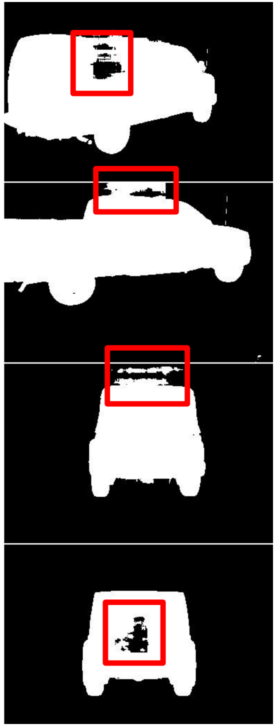
Up-sampling comparison: Reusing of max-pooling indices is better than using transposed convolution.
Helps generate more accurate segmentation

Improvement of the Coordinate Maps Method

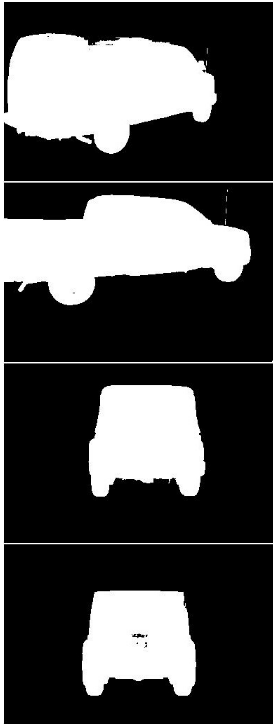
Model	1-DSC Test set	Max pool. indices	Coord. maps	Dilated Conv.	#Parameters	Training Time ms/image	Test Time ms/image
U-Net	0.4396%				31,043,586	137.6	34.42
SegNet	0.3908%	√			29,444,162	165.13	30.50
Arc. #1	**0.3068%**	√			32,799,554	171.75	42.793
Arc. #2	0.313%	√	Input layer		32,800,706	172.59	42.901
Arc. #3	**0.3063%**	√	Mid-layer		32,808,770	170.50	42.839
MaskingNet	**0.2977%**	√	Mid-layer	√	32,808,770	180.46	46.609

Encoding the Coord. Maps in the mid layer is better than not using them or encoding them in the input layer

Architecture-1
Reuse of pooling indices



Architecture-3
Reuse of pooling indices
Coord. Maps at middle



Improves the performance on some hard cases

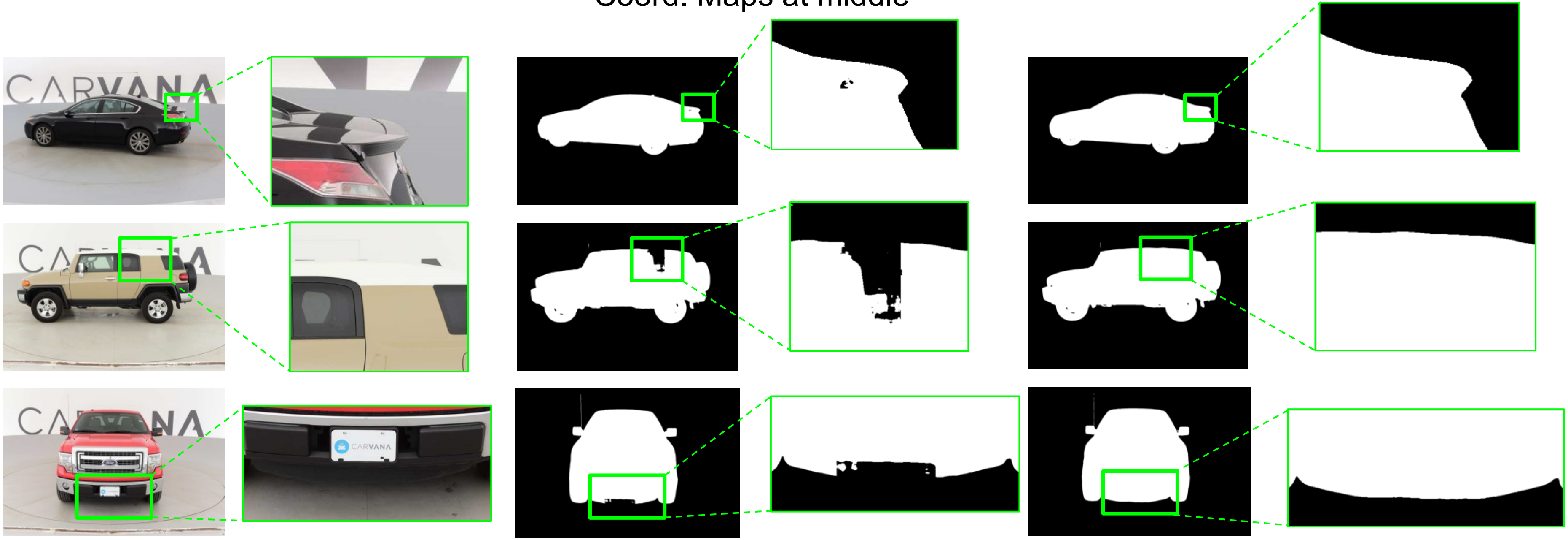
Improvement of the Dilated Conv. Method

Model	1-DSC Test set	Max pool. indices	Coord. maps	Dilated Conv.	#Parameters	Training Time ms/image	Test Time ms/image
U-Net	0.4396%				31,043,586	137.6	34.42
SegNet	0.3908%	√			29,444,162	165.13	30.50
Arc. #1	**0.3068%**	√			32,799,554	171.75	42.793
Arc. #2	0.313%	√	Input layer		32,800,706	172.59	42.901
Arc. #3	**0.3063%**	√	Mid-layer		32,808,770	170.50	42.839
MaskingNet	**0.2977%**	√	Mid-layer	√	32,808,770	180.46	46.609

MaskingNet is the best

Architecture-3
Reuse of pooling indices
Coord. Maps at middle

MaskingNet
Reuse of pooling indices
Coord. Maps at middle
Dilated Conv.



- Enlarge the receptive field to remove holes inside the car
- Dilated convolution helps to infer the shape of the car, even the boundaries are unclear in the RGB image

Model	DSC Test set	Max pool. indices	Coord. maps	Dilated Conv.
U-Net	99.5604%			
SegNet	99.6092%	√		
Arc. #1	**99.6932%**	√		
Arc. #2	99.6870%	√	Input layer	
Arc. #3	**99.6937%**	√	Mid-layer	
MaskingNet	**99.7023%**	√	Mid-layer	√

- Train only once
- Didn't adjust hyper-parameters
- Only train 4/5 training images

Team Members	Score	Entries	Last
	0.997332	80	1y
	0.997331	78	1y
	0.997264	43	1y
	0.997232	82	1y
	0.997209	59	1y
	0.997190	76	1y
	0.997151	12	1y
	0.997138	16	1y
	0.997126	25	1y
	0.997123	65	1y
	0.997115	42	1y
	0.997085	20	1y

- Ensembled by many models

Summary

- Proposed a new end-to-end deep fully convolutional neural network architecture for semantic segmentation named MaskingNet.
  - New combination of reusing pooling indices and concat. encoder feature map
  - New concat. coord. maps with mid-layer feature map method
  - New dilated conv. method
- Experimental results show that MaskingNet outperforms state-of-the-art methods U-Net and SegNet. The error decreased by 32.3% compared with U-Net and 23.8% compared with SegNet.

Question?

Thank you !