ANNOTATING OBJECT INSTANCES WITH A POLYGON RNN

08/11/2019

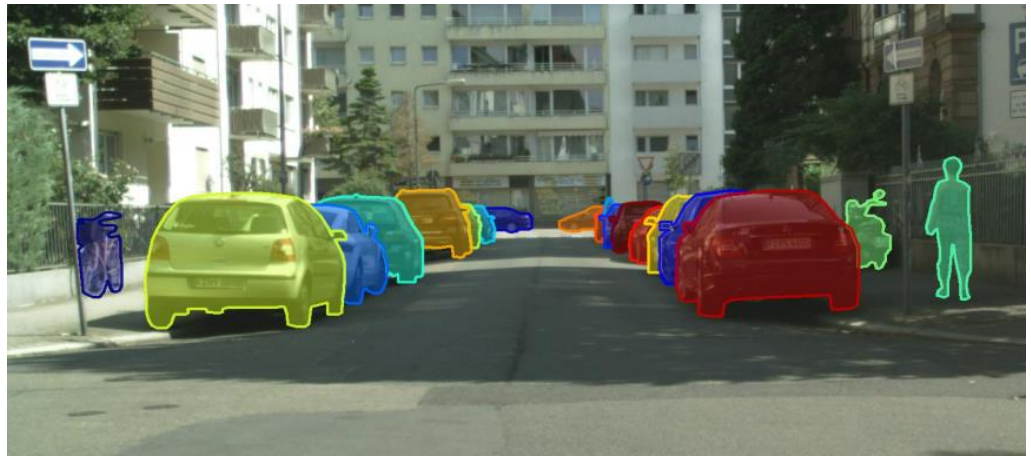
Paper by: Lluis Castrejon, Kaustav Kundu, Raquel Urtasun, and Sanja Fidler

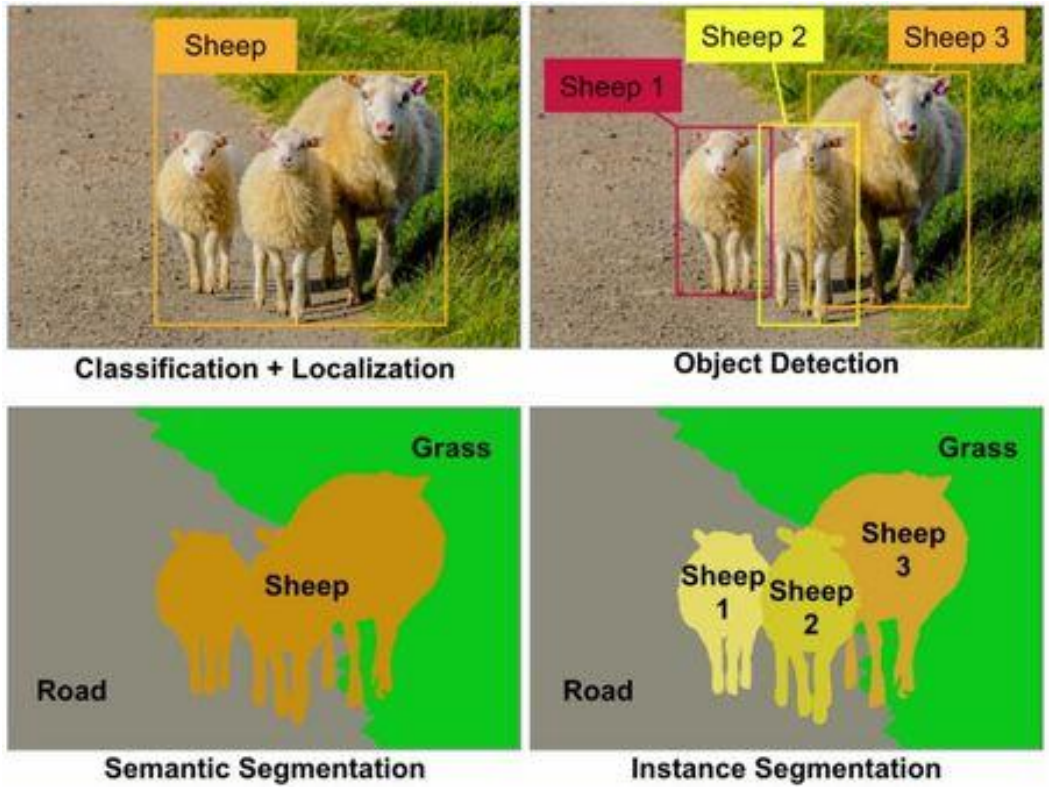
Presented by: Neel Bhatt

UNIVERSITY OF WATERLOO
FACULTY OF ENGINEERING

Context

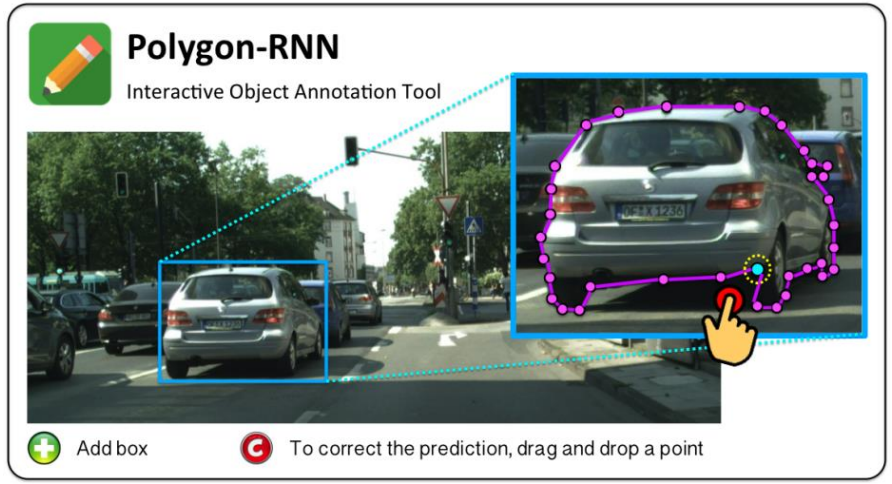
- 4 main types of image analysis pertaining to objects
- This paper focuses on **instance segmentation**
  - To train a model, we need to *generate* ground truth (GT) data by annotation





Motivation

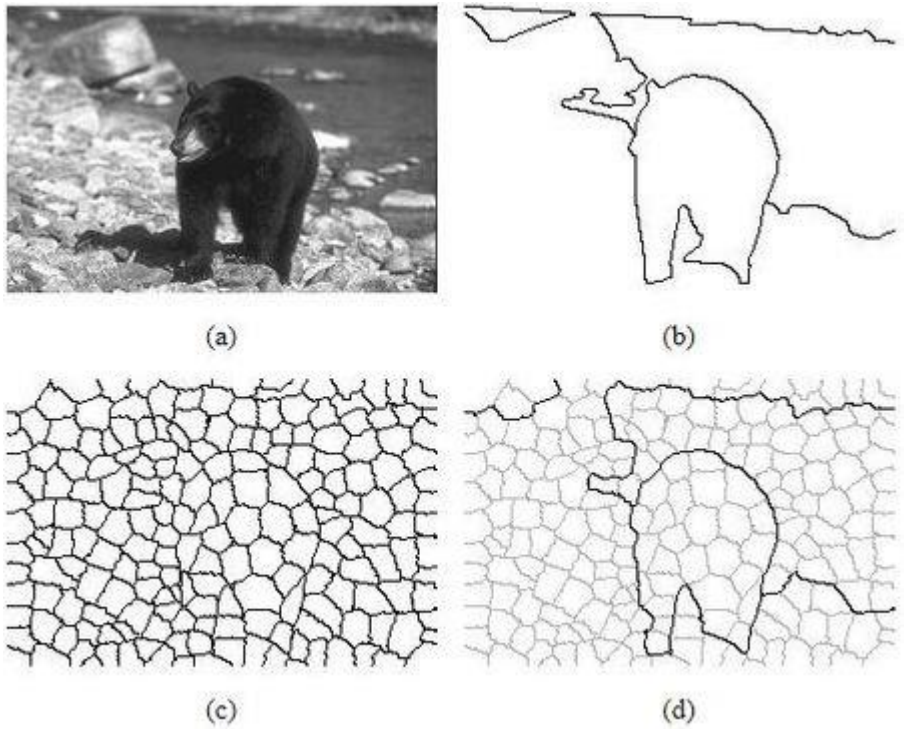
- Cityscape dataset contains 3500 images
  - 60,000 object instances
  - Will take about 500+ hours to annotate



- The main goal is to **reduce time spent** to annotate images
  - This paper presents a *semi-automatic* method for annotation

Related Work

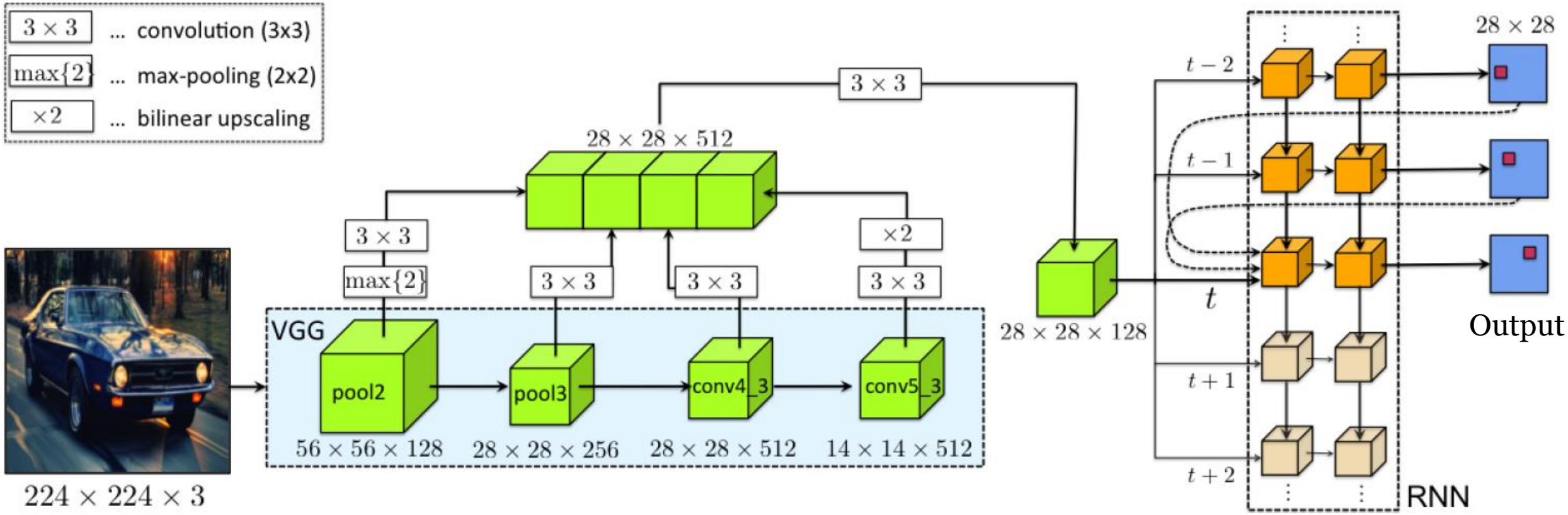
- Grabcut
- Superpixel approach



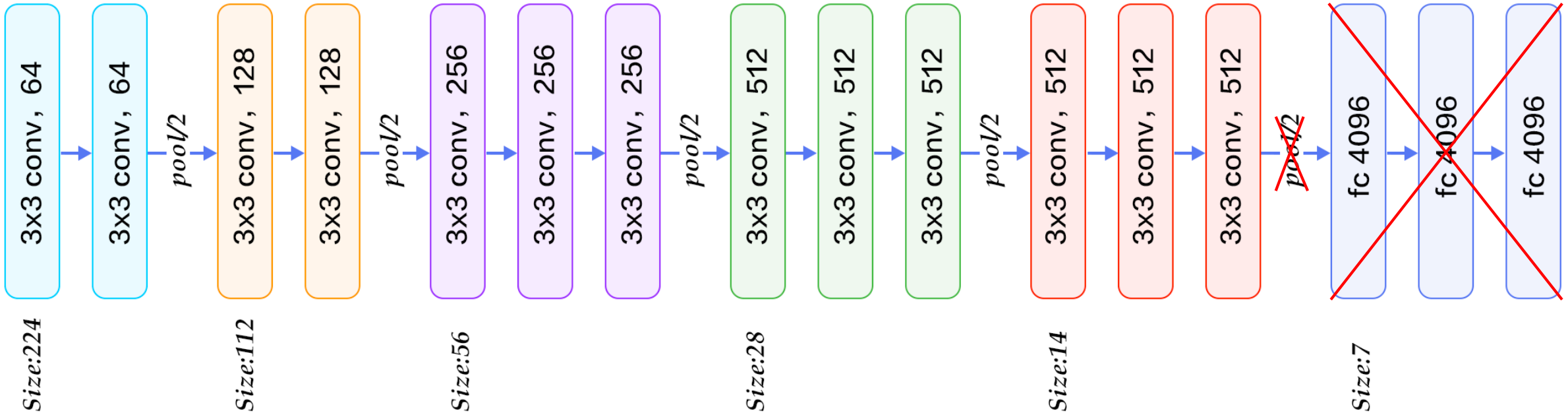
Polygon RNN – Why Polygons?

- It is a clustered representation of pixels
  - Implies: a sparse representation
- Follows what humans do to represent or annotate an object
- Easy to accommodate changes, if the object is not enclosed properly

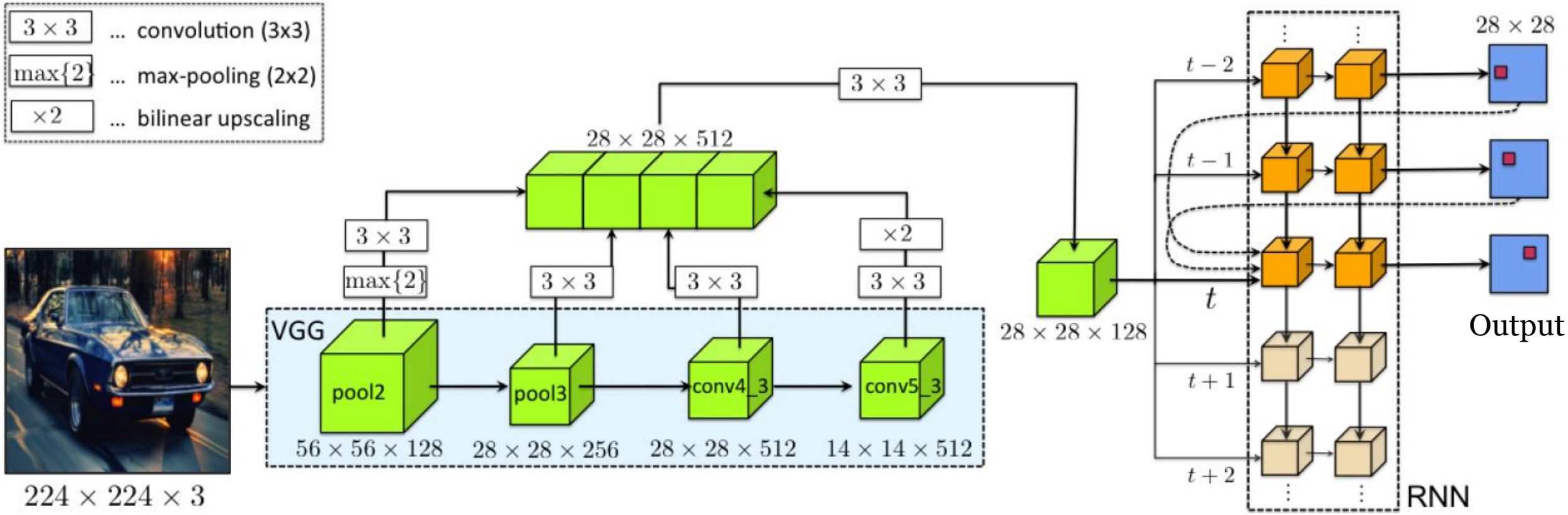
Model Overview



VGG16 Architecture



Model Overview



ConvLSTM

- Takes as input x_t from the CNN
  - Computes a hidden state h_t given:
  - i, f, o denote the input, forget, and output gate,
  - c_t is the cell state at time step t,
  - σ denotes the sigmoid function,
  - ⊙ indicates an element-wise product,
  - $*$ denotes a convolution,
  - W_h denotes the hidden-to-state convolution kernel and Wx the input-to-state convolution kernel.

$$\begin{pmatrix} i_t \\ f_t \\ o_t \\ g_t \end{pmatrix} = W_h * h_{t-1} + W_x * x_t + b$$

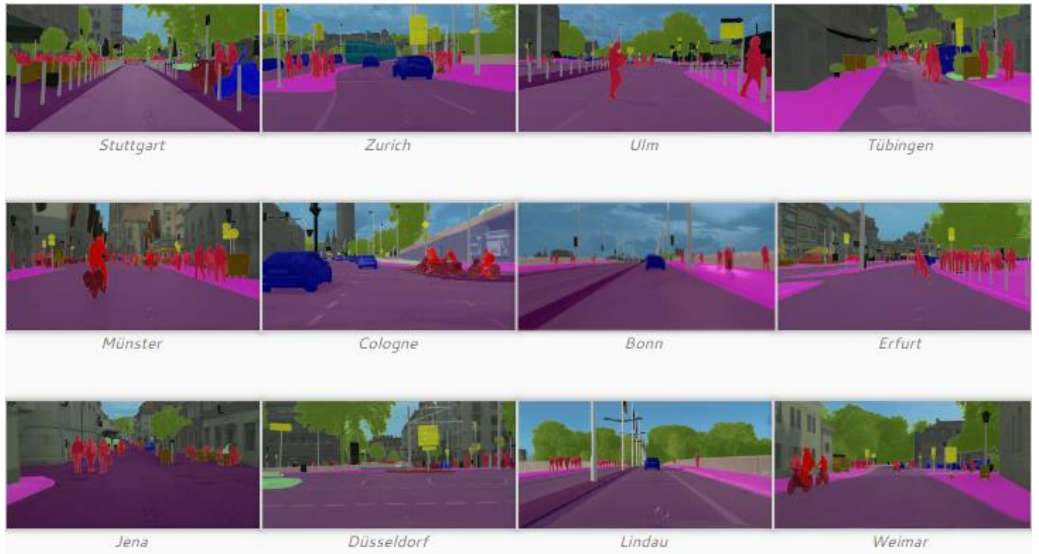
$$c_t = \sigma(f_t) \odot c_{t-1} + \sigma(i_t) \odot tanh(g_t)$$

$$h_t = \sigma(o_t) \odot tanh(c_t)$$

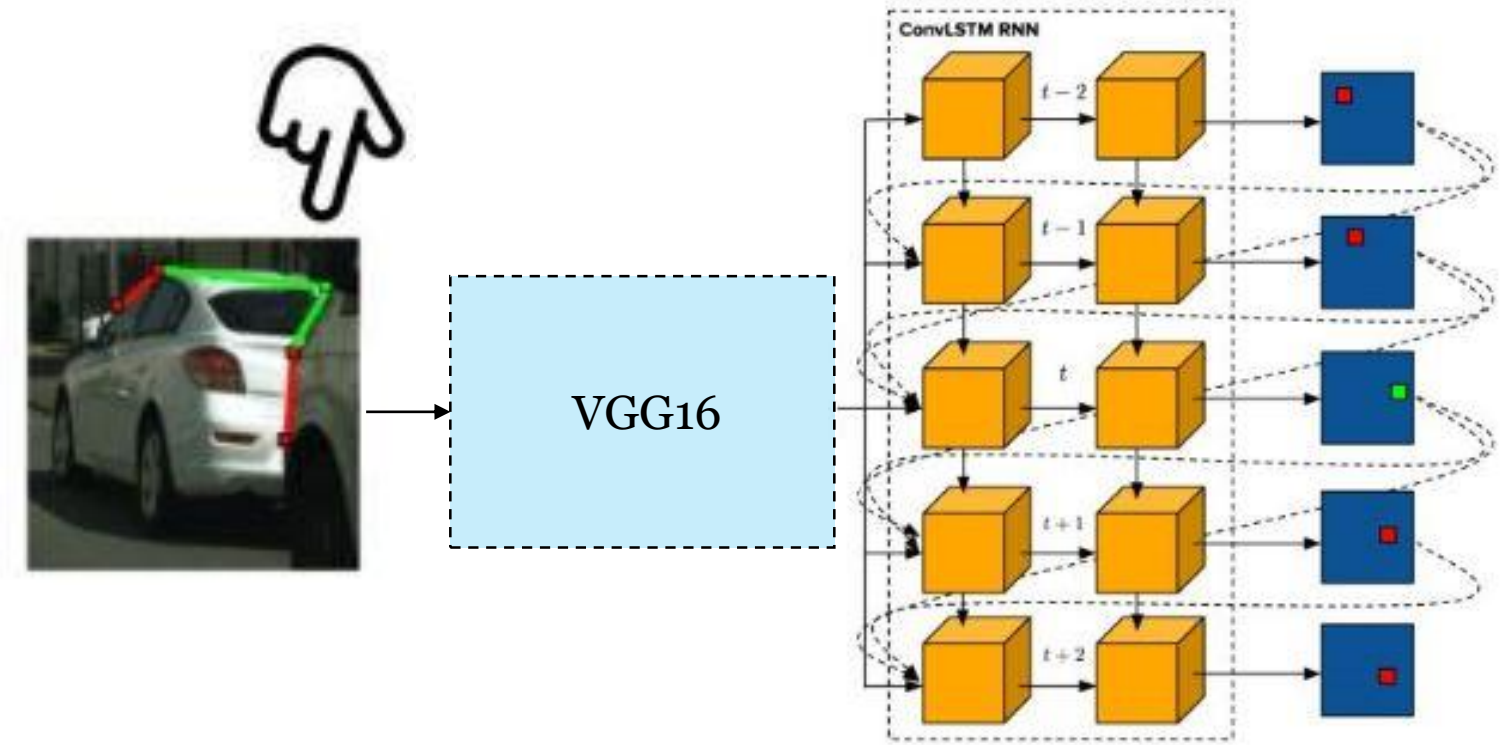
- Outputs one-hot representation of the predicted vertex

Training

- Cityscape Dataset
  - 2975 training, 500 validation, and 1525 test images
- KITTI Dataset
  - To measure generalization capability
- Implementation details:
  - Cross-entropy loss function
  - Adam optimizer (batch size = 8; decaying learning rate = 1^{-4}; decays by a factor of 10 after 10 epochs)
  - Training time: 1 full day on a NVIDIA Titan-X GPU
  - Data Augmentation: Flipping images, bounding box expansion, random starting vertex

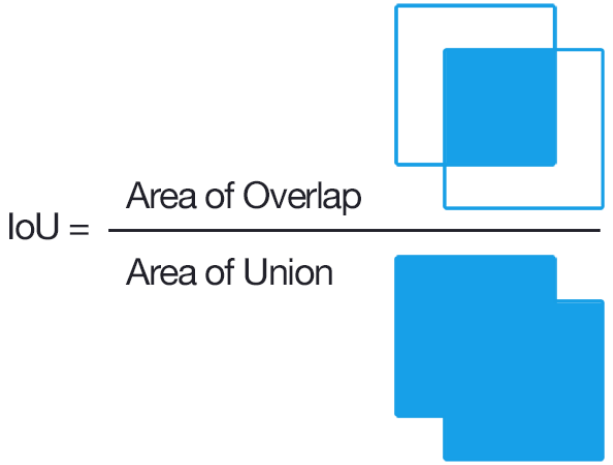


Importance of Human Annotator



Results

- Evaluation:
  - IoU
  - Number of Clicks
    - To evaluate speed up factor when human annotator is involved
    - Checkerboard distance threshold: $T \epsilon$ [1,2,3,4]
- Baseline:
  - **SharpMask** – 50 layer ResNet
  - **DeepMask** – SharpMask + additional CNN
  - **Dilation10** – Purely convolutional operations
  - **SquareBox** – Entire bounding box is labeled as an instance



Evaluation - IoU

- Superior performance in 6/8 categories without human corrections
  - IoU compared to SharpMask:
    - Car – 12%
    - Person – 7%
    - Rider – 6%

Model	Bicycle	Bus	Person	Train	Truck	Motorcycle	Car	Rider	Mean
Square Box	35.41	53.44	26.36	39.34	54.75	39.47	46.04	26.09	40.11
Dilation10	46.80	48.35	49.37	44.18	35.71	26.97	61.49	38.21	43.89
DeepMask [20]	47.19	69.82	47.93	62.20	63.15	47.47	61.64	52.20	56.45
SharpMask [20]	52.08	**73.02**	53.63	**64.06**	65.49	51.92	65.17	56.32	60.21
Ours	**52.13**	69.53	**63.94**	53.74	**68.03**	**52.07**	**71.17**	**60.58**	**61.40**

Evaluation - # of Clicks

- Importance of human annotator:
  - Under 5 clicks, speed up factor is 7 times compared to complete manual annotation

- Applied to KITTI dataset:
  - Still able to achieve similar IoU in about the same number of clicks

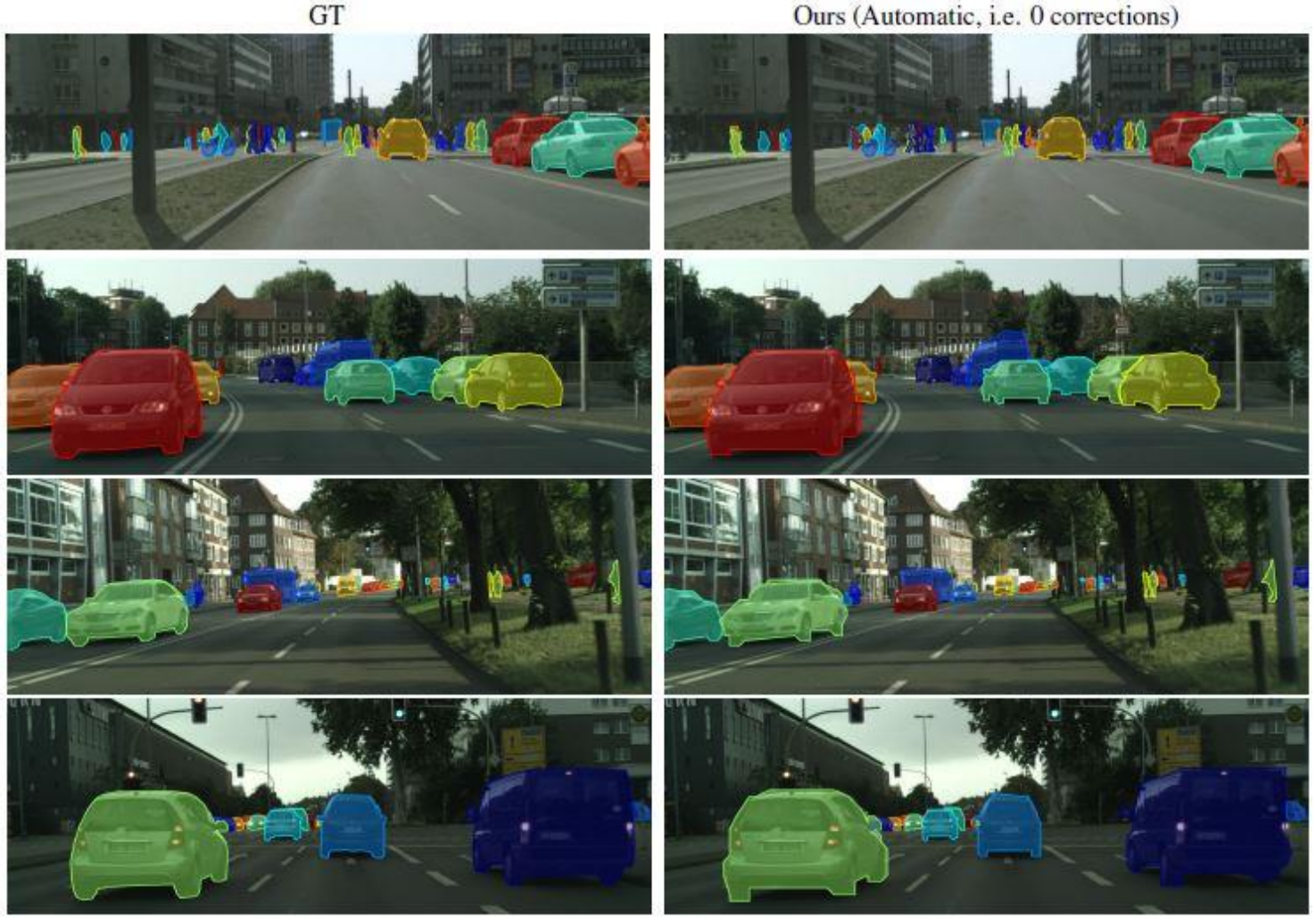
Speed-Up Factor

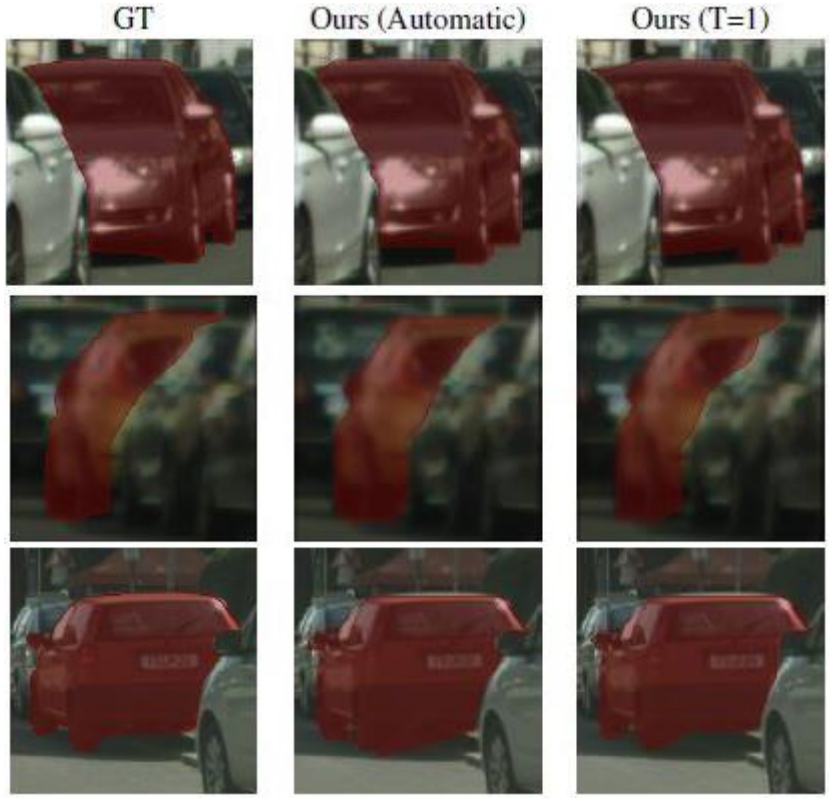
Method	Num. Clicks	IoU	Annot. Speed-Up
Cityscapes GT	33.56	100	-
Ann. full image	79.94	69.5	-
Ann. crops	96.09	78.6	-
Ours (Automatic)	0	73.3	No ann.
Ours (T=1)	9.3	87.7	x3.61
Ours (T=2)	6.6	85.7	x5.11
Ours (T=3)	5.6	84.0	x6.01
Ours (T=4)	4.6	82.2	x7.31

KITTI Dataset

Method	# of Clicks	IOU
DeepMask [20]	-	78.3
SharpMask [21]	-	78.8
Beat the MTurkers [4]	0	73.9
Ours (Automatic)	0	74.22
Ours (T=1)	11.83	89.43
Ours (T=2)	8.54	87.51
Ours (T=3)	6.83	85.70
Ours (T=4)	5.84	84.11

Qualitative Results





Critique

- Significant improvement in annotation time
  - Speed-up factor of 7
  - Cut down costs
  - Human annotation input helps increase IoU
- Approach works in other domains
  - Medical applications
- Model uses VGG16 instead of ResNet, but still quite efficient
- Baseline methods have an upper hand for larger objects: output resolution of baselines is higher
- Future work: remain insensitive to size of object, elimination of CNN for first vertex

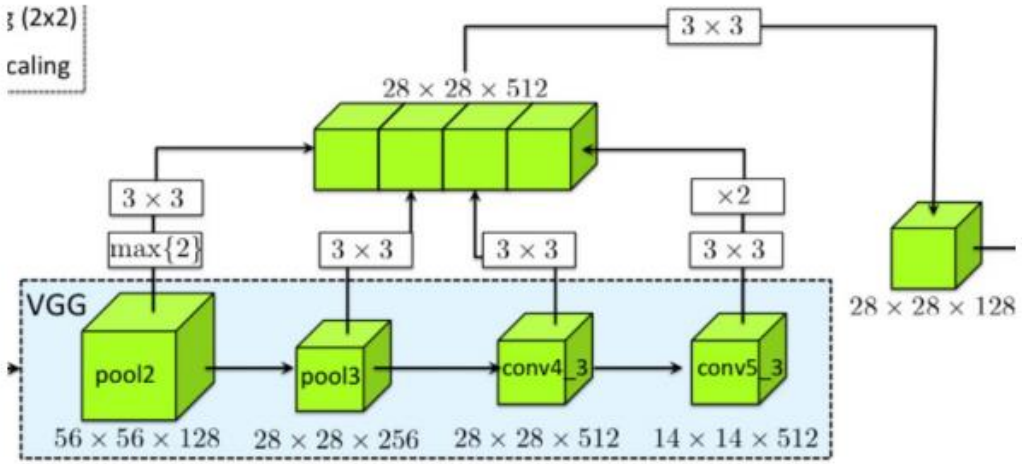
Questions

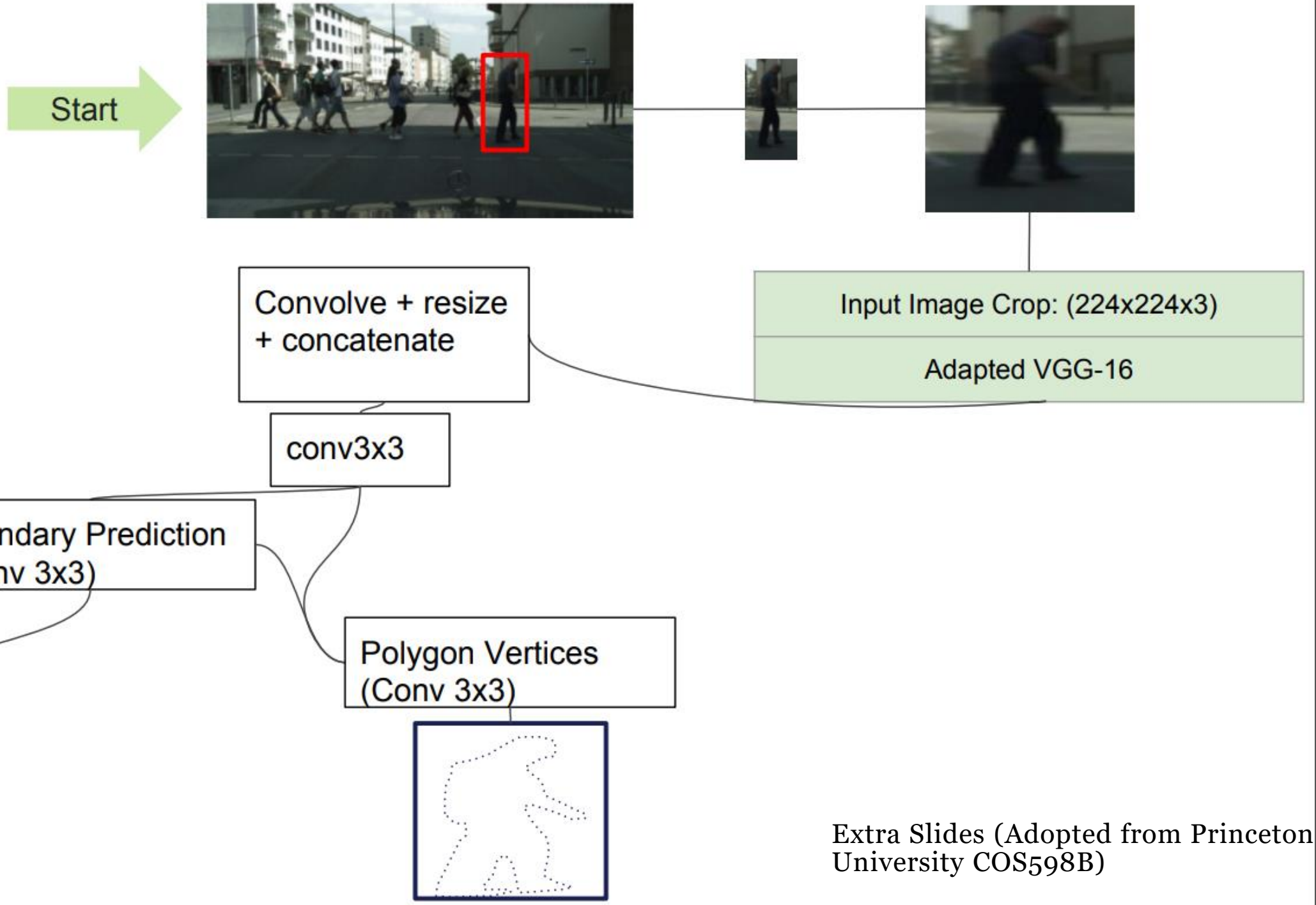
Castrejon, Lluis, et al. "Annotating Object Instances with a Polygon-RNN." CVPR. Vol. 1. 2017

Extra Slides (Adopted from Princeton University COS598B)

What about t=1?

- First vertex of polygon is not uniquely defined
  - Use a **second** (identical) CNN for initial inference
  - *Note: the weights are NOT shared between the two VGG nets





Extra Slides (Adopted from Princeton University COS598B)