Computer Vision: Fall 2022 — Lecture 14

Dr. Karthik Mohan

Univ. of Washington, Seattle

November 16, 2022

References

Generic ML/DL

1. Good Book for Machine Learning Concepts
2. Deep Learning Reference

CNN

1. Convolutional Neural Networks for Visual Recognition
2. Convolutional Neural Net Tutorial
3. CNN Transfer Learning
4. PyTorch Transfer Learning Tutorial

CNN Publication References

CNN surveys

1. Convolutional Neural Networks: A comprehensive survey, 2019
2. A survey of Convolutional Neural Networks: Analysis, Applications, and Prospects, 2021

CNN Archs

1. GoogLeNet
2. Top models on ImageNet
3. ResNet ILSVRC paper

Object Detection and Image Segmentation References

Object Detection

CNN

1. A survey of modern deep learning based object detection methods
2. R-CNN
3. Fast R-CNN
4. Faster R-CNN

Mid Course Survey

Mid-course Survey Results

Learnings from the Mini-Project - Breakout Session!

Discuss This next Lecture

Breakout and Discuss - Peer Learning (5 mins)

Breakout and discuss in your zoom room - What were your key learnings from the mini-project? What strategies worked and what didn't? How much did hyper-param tuning play a role in the result? Did you get to build your intuition with the models you tested?

Last Lecture

1. Transfer learning in CNN (a.k.a how to not reinvent the wheel with CNN training!)

Last Lecture

1. Transfer learning in CNN (a.k.a how to not reinvent the wheel with CNN training!)
2. Pre-trained models

Last Lecture

1. Transfer learning in CNN (a.k.a how to not reinvent the wheel with CNN training!)
2. Pre-trained models
3. Strategies to transfer the learning from a pre-trained model to a new data set

Last Lecture

1. Transfer learning in CNN (a.k.a how to not reinvent the wheel with CNN training!)
2. Pre-trained models
3. Strategies to transfer the learning from a pre-trained model to a new data set
4. PyTorch Tutorial on Transfer Learning

Last Lecture

1. Transfer learning in CNN (a.k.a how to not reinvent the wheel with CNN training!)
2. Pre-trained models
3. Strategies to transfer the learning from a pre-trained model to a new data set
4. PyTorch Tutorial on Transfer Learning
5. How many of us tried out a transfer learning model for Mini-Project? Thoughts??

Today!

1. Introduction to Object Detection and Instance Segmentation
2. Models and Architectures for Object Detection and Instance Segmentation

Computer Vision Topics

1. Image Processing using convolutions
2. Image De-noising
3. Image Smoothing
4. Image Clustering
5. Image Classification
6. Object Detection
7. Semantic Segmentation
8. Instance Segmentation (maybe)
9. Image Embeddings
10. Image to Text
11. Image Captioning
12. Text to Image (high-level)

NOW

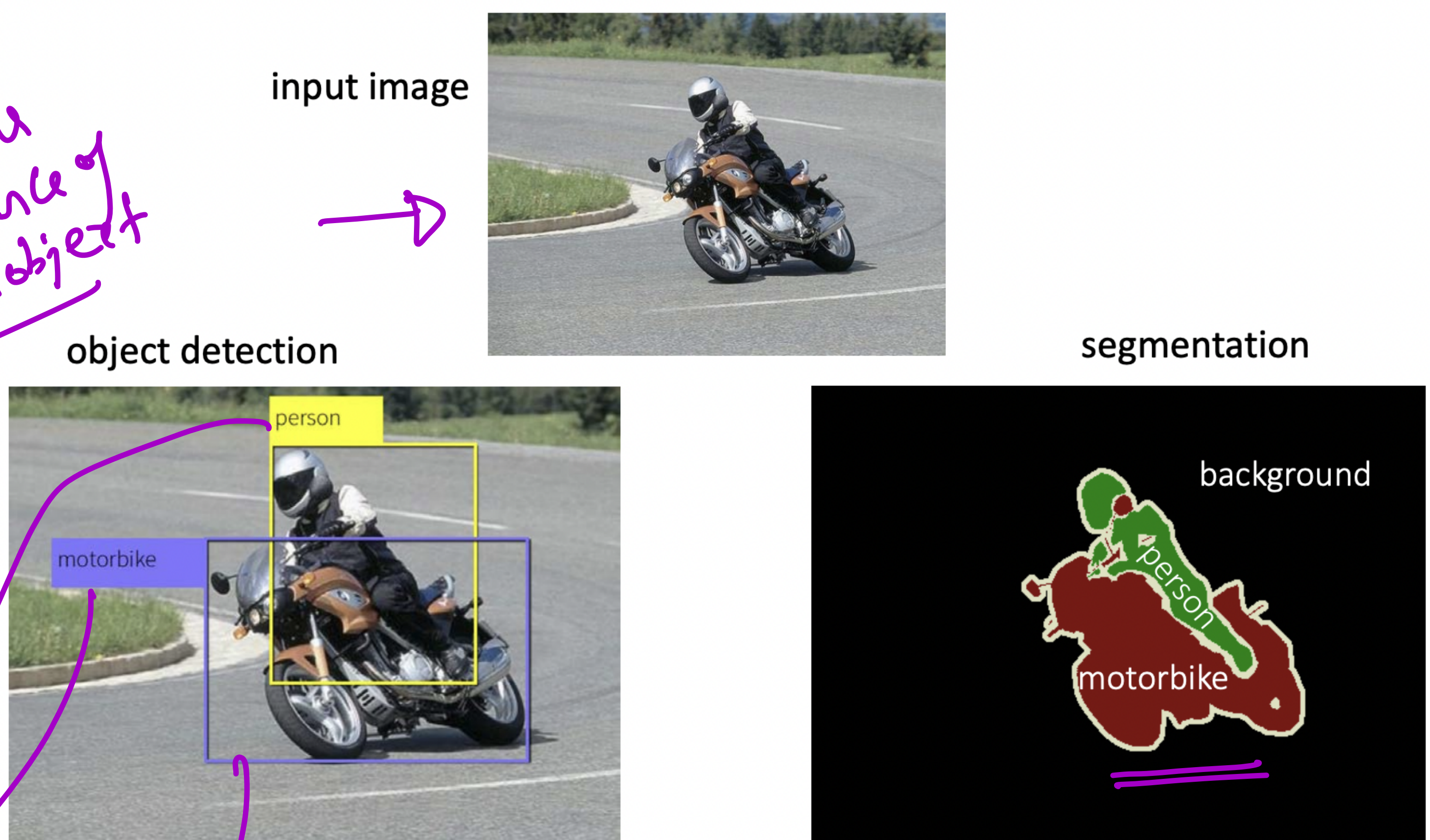
Object and Instance Detection Motivation

Traffic Instance Segmentation

Object Detection vs Image Segmentation

Every single instance of an object

Instance Detection

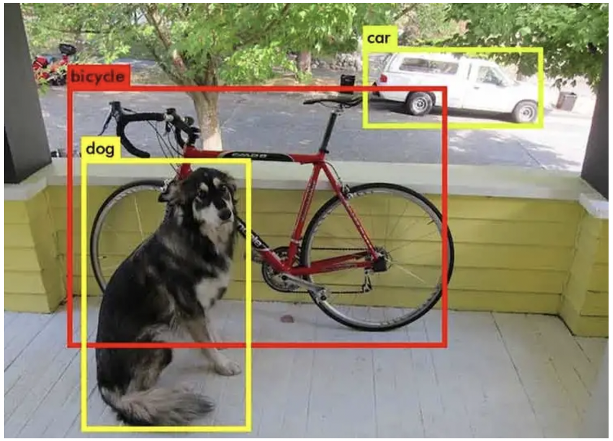


Classification

Bounding Box → width, height, center

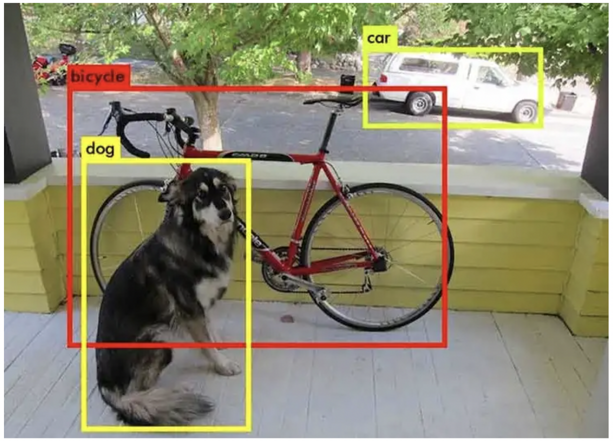
(x, c, w, h) → center, width, height

Object Detection History



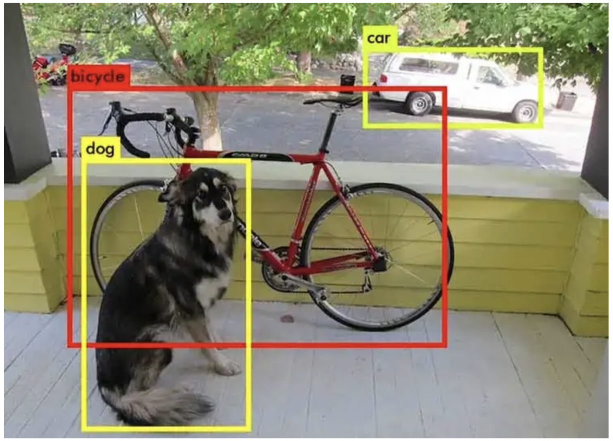
1. Has been an uphill task until 2012

Object Detection History



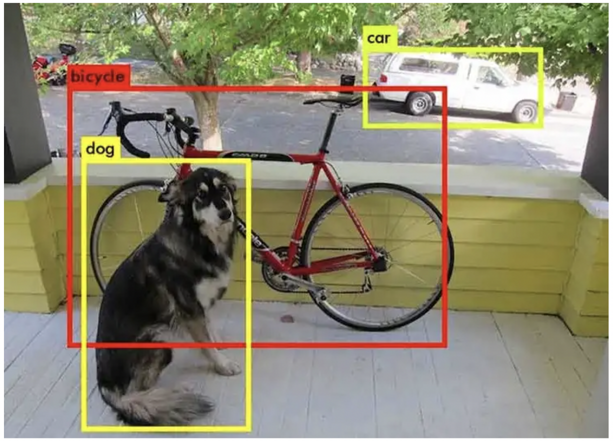
1. Has been an uphill task until 2012
2. Early detectors for objects - Ensemble of hand-crafted ones

Object Detection History



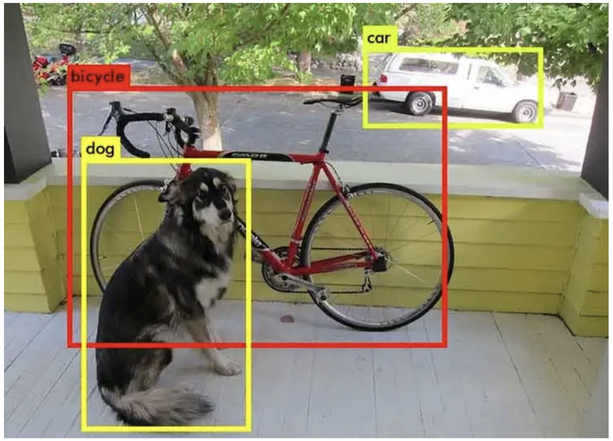
1. Has been an uphill task until 2012
2. Early detectors for objects - Ensemble of hand-crafted ones
3. Early detectors: Low accuracy and cumbersome/time-consuming

Object Detection History



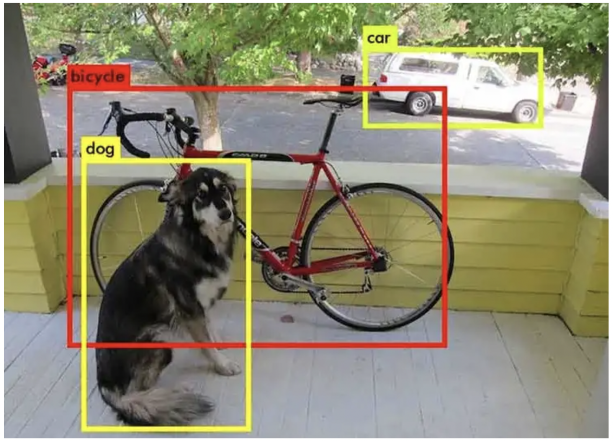
1. Has been an uphill task until 2012
2. Early detectors for objects - Ensemble of hand-crafted ones
3. Early detectors: Low accuracy and cumbersome/time-consuming
4. CNN changed the landscape - Better Accuracy, faster train, generalizability

Object Detection History



1. Has been an uphill task until 2012
2. Early detectors for objects - Ensemble of hand-crafted ones
3. Early detectors: Low accuracy and cumbersome/time-consuming
4. CNN changed the landscape - Better Accuracy, faster train, generalizability
5. AlexNet (2012) - First CNN archs to be applied to Obj. Detection

Object Detection History



1. Has been an uphill task until 2012
2. Early detectors for objects - Ensemble of hand-crafted ones
3. Early detectors: Low accuracy and cumbersome/time-consuming
4. CNN changed the landscape - Better Accuracy, faster train, generalizability
5. AlexNet (2012) - First CNN archs to be applied to Obj. Detection
6. **Real world application:** Self-driving cars

Multi-class Classification vs Object Detection

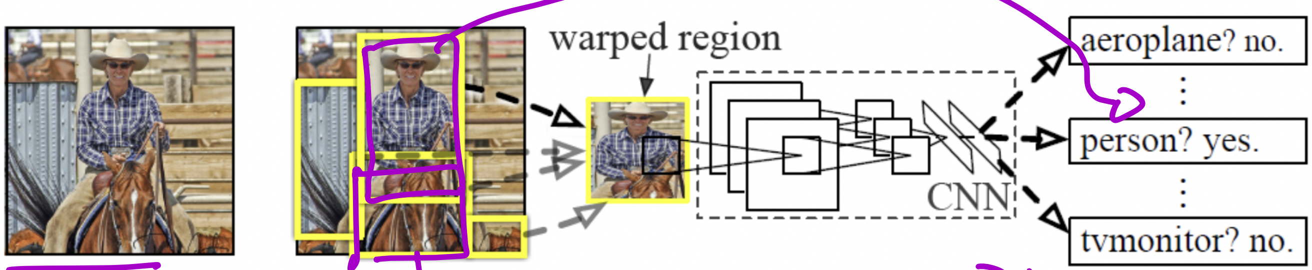
Multi-Class Classification

Object Detection

1) **Localization** → where are the bounding boxes??

2) **Multi-class Classification!**

Multi-label Classification vs Object Detection



- Person
- Jockey

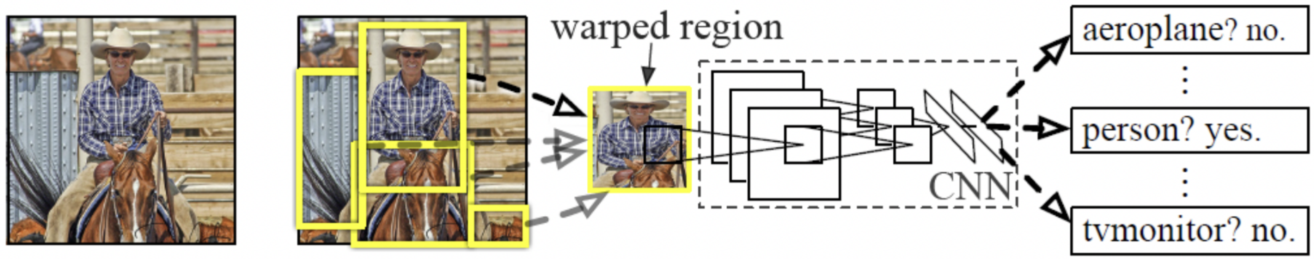
Multi-Class Classification

Object Detection

Localization

Multi-class Classification!

Multi-class Classification vs Object Detection



Multi-Class Classification

Person

Multi-label Classificaiton

Person, Horse, House

multi-labels

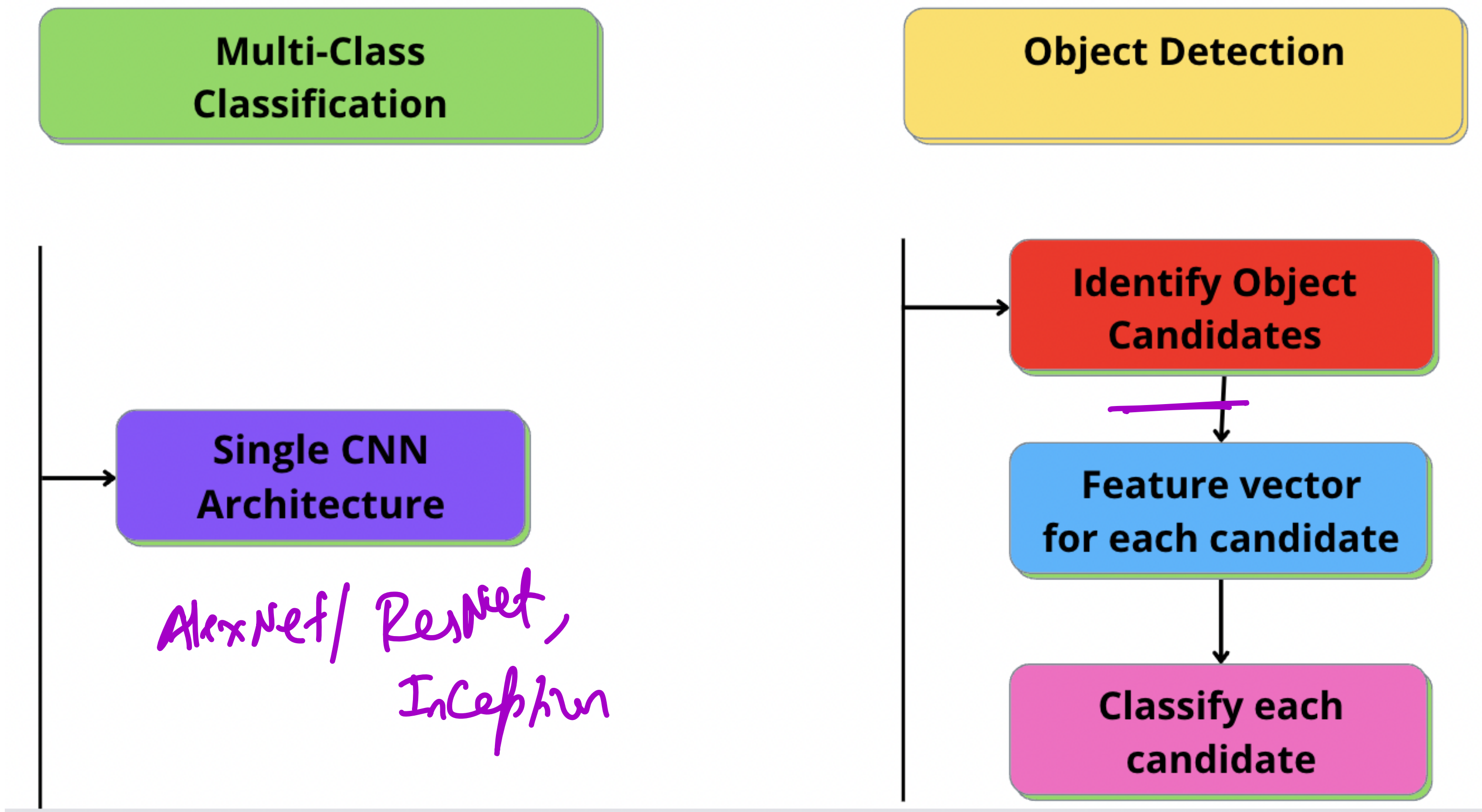
Object Detection

multi -label +

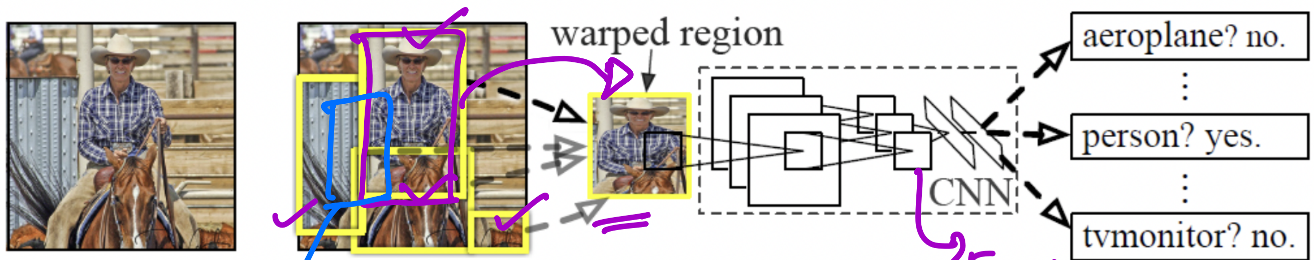
Localization

Multi-class Classification!

Multi-class Classification vs Object Detection



Object Detection Intuition



Feature vector (CNN Code)

Have a person?

Object Detection

Identify Object Candidates

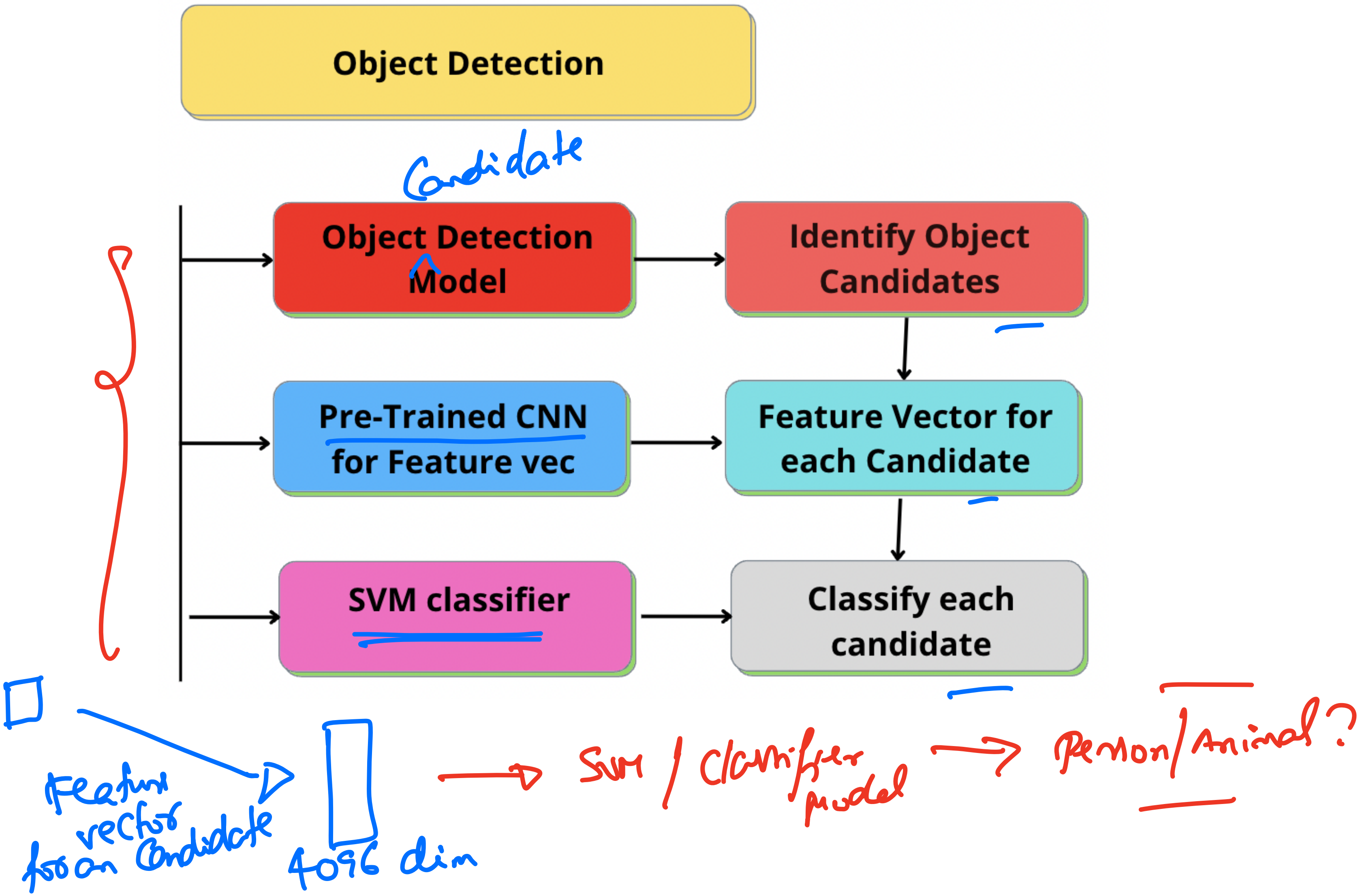
Feature vector for each candidate

Classify each candidate

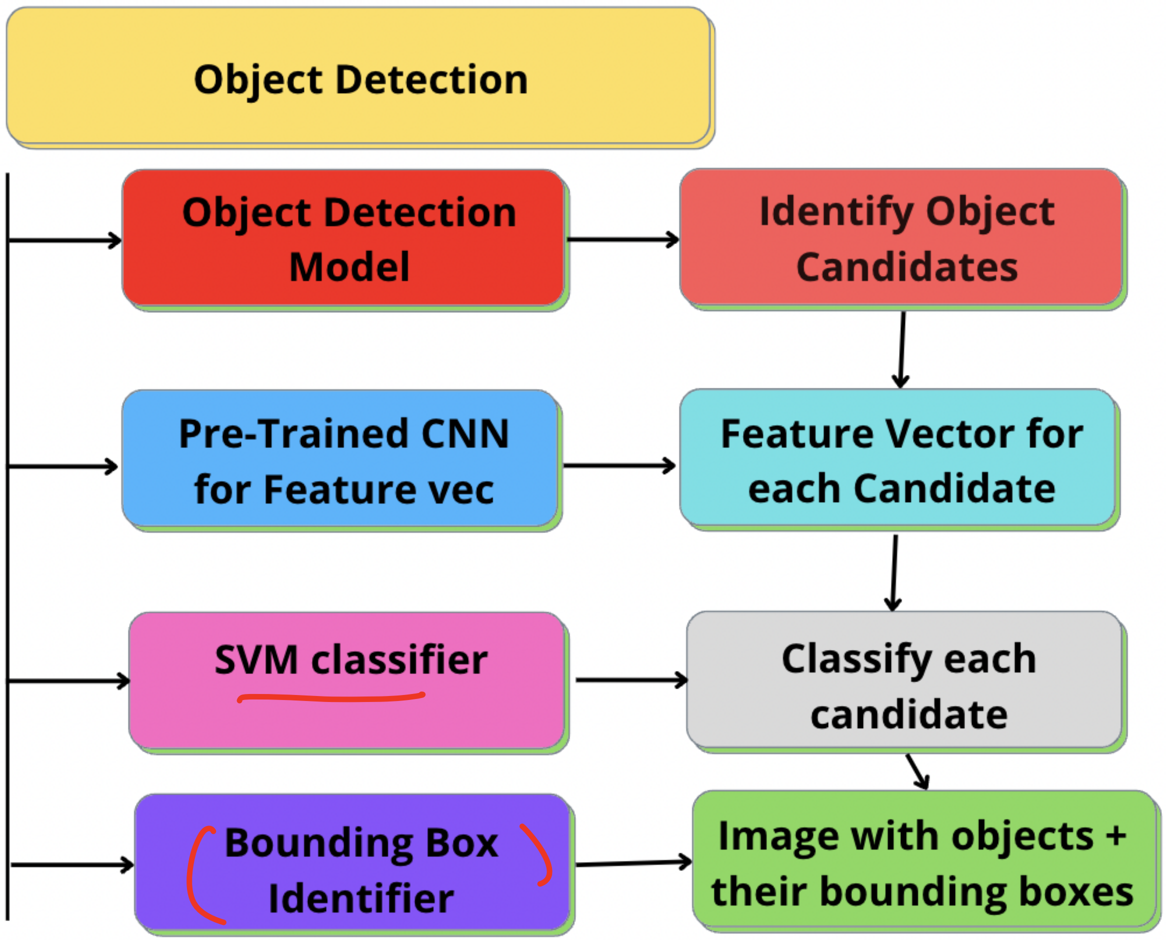
selection search

→ 100's or 1000's from Image

Object Detection Model Framework

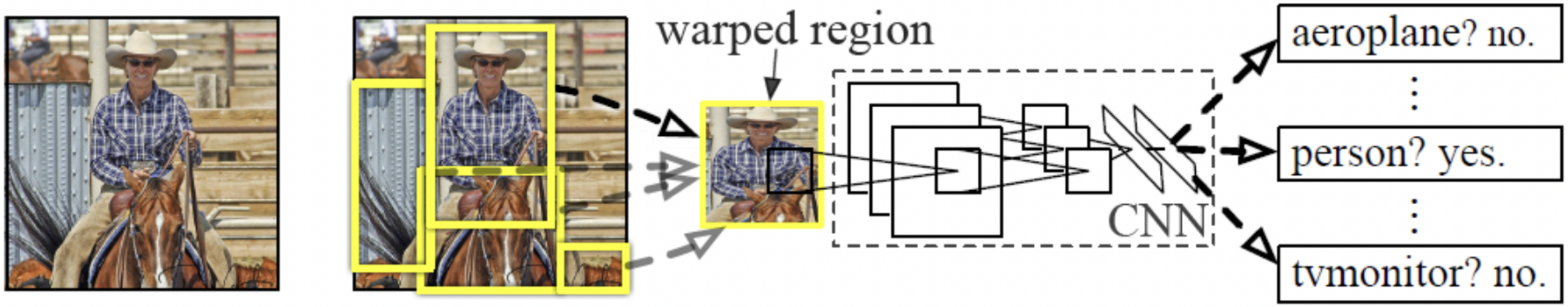


Object Detection Model Framework

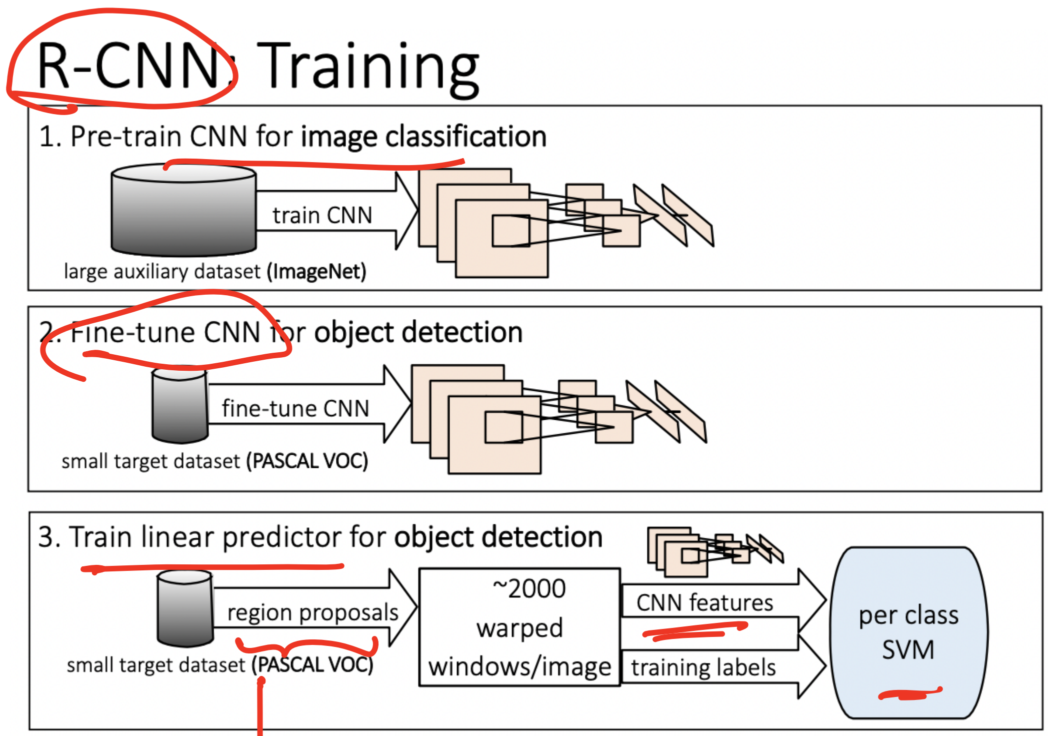


First CNN Model for Object Detection: R-CNN model

Region of Interest



First CNN Model for Object Detection: R-CNN model



Region proposals (candidate)

Object Detection Dimensions

ImageNet for Multi-class

1. Data Sets for benchmarking

Object Detection Dimensions

1. Data Sets for benchmarking
2. DL/CNN Models (R-CNN, Fast R-CNN, etc)

Object Detection Dimensions

1. Data Sets for benchmarking
2. DL/CNN Models
3. Metrics

Let's take a look at the Data Sets

Let's take a look at the Data Sets

Dataset	Classes	Train			Validation			Test
		Images	Objects	Objects/Image	Images	Objects	Objects/Image	
1. PASCAL VOC 12	20	5,717	13,609	2.38	5,823	13,841	2.37	10,991
MS-COCO	80	118,287	860,001	7.27	5,000	36,781	7.35	40,670
ILSVRC	200	456,567	478,807	1.05	20,121	55,501	2.76	40,152
OpenImage	600	1,743,042	14,610,229	8.38	41,620	204,621	4.92	125,436

Microsoft

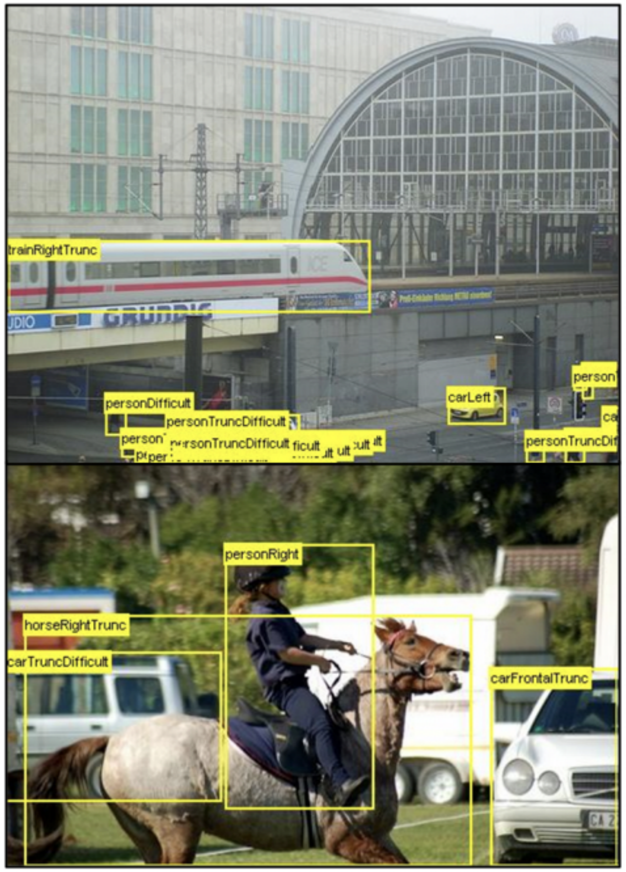
Google

(Largest Data sets for Obj. Detection)

#images in val < #images in test

Data Sets

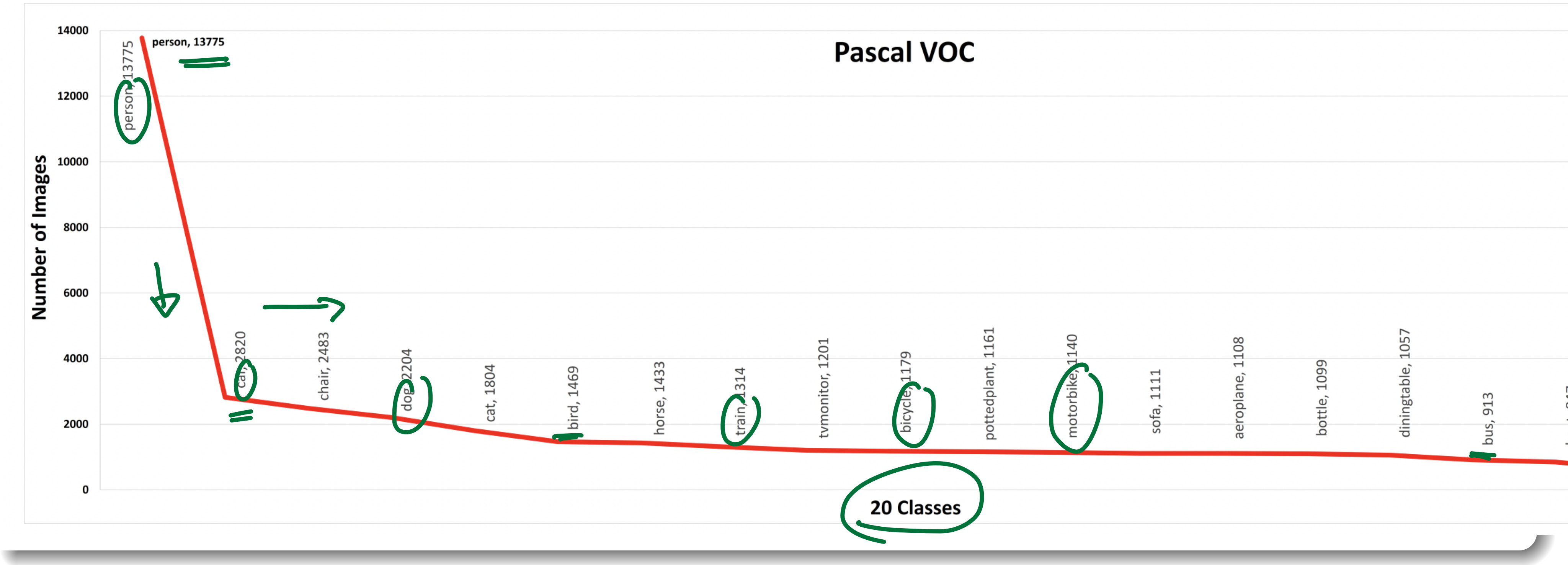
Pascal Data Set



(a) PASCAL VOC 12

Data Sets

Pascal Data Set



Let's take a look at the Data Sets

Dataset	Classes	Train			Validation			Test
		Images	Objects	Objects/Image	Images	Objects	Objects/Image	
PASCAL VOC 12	20	5,717	13,609	2.38	5,823	13,841	2.37	10,991
MS-COCO	80	118,287	860,001	7.27	5,000	36,781	7.35	40,670
ILSVRC	200	456,567	478,807	1.05	20,121	55,501	2.76	40,152
OpenImage	600	1,743,042	14,610,229	8.38	41,620	204,621	4.92	125,436

Data Sets

ILSVRC Data Set

1 object

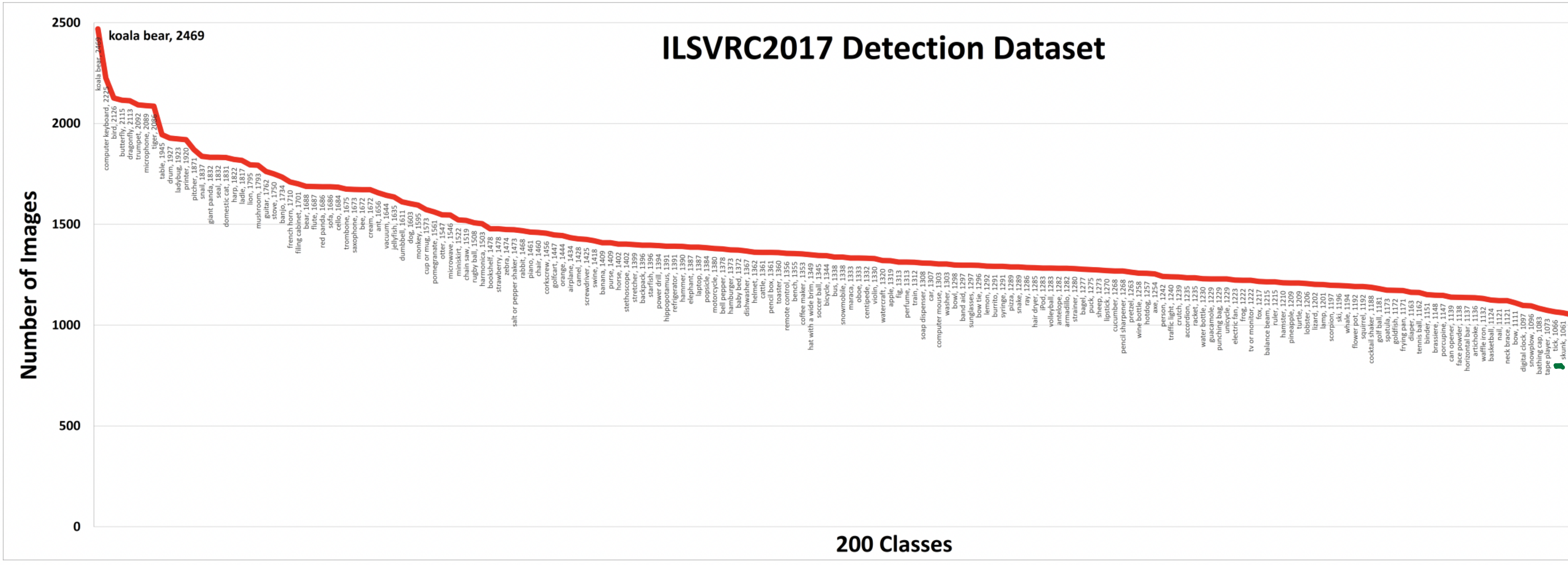
← 1 obj

← Person
← train
etc

(c) ILSVRC

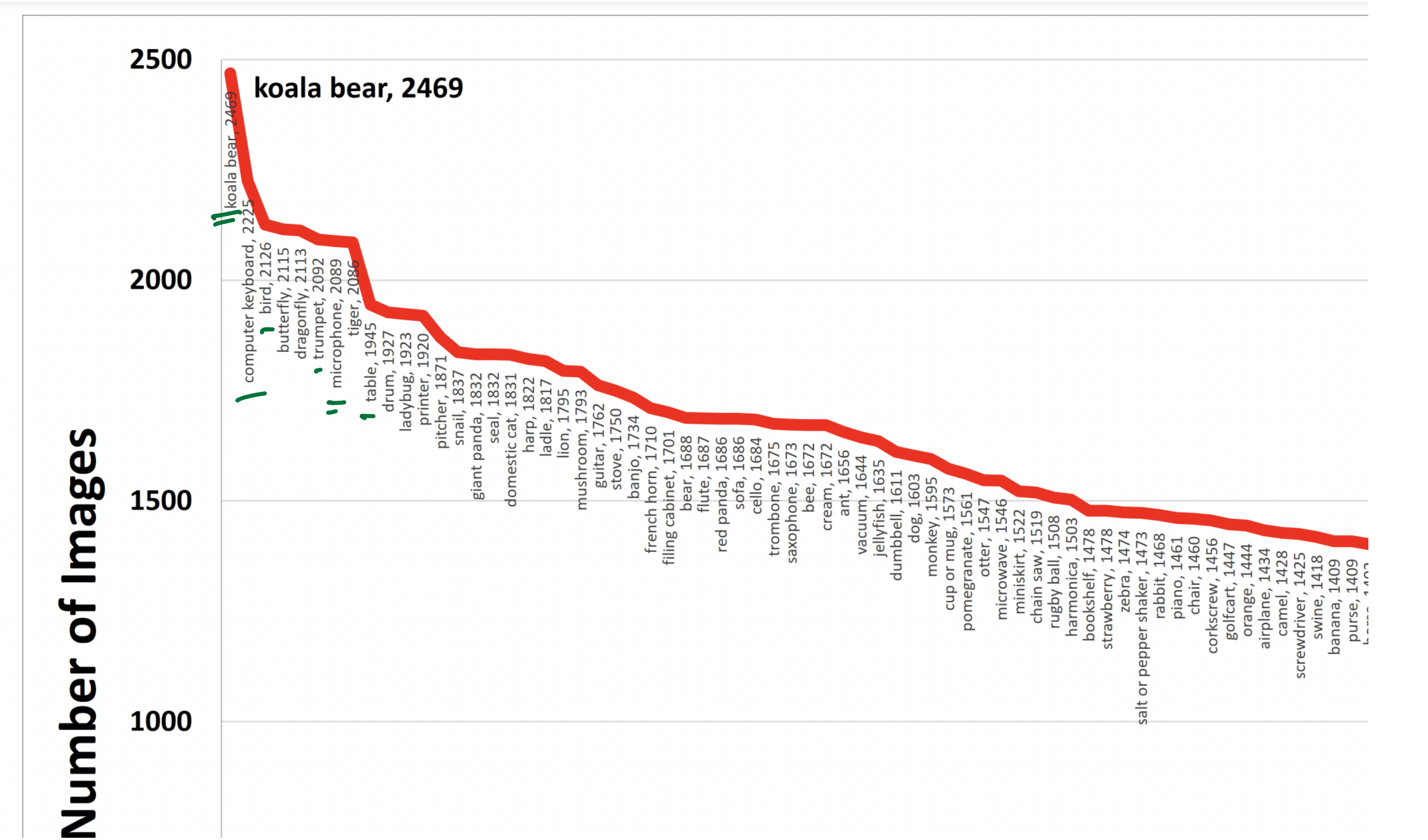
Data Sets

ILSVRC Data Set



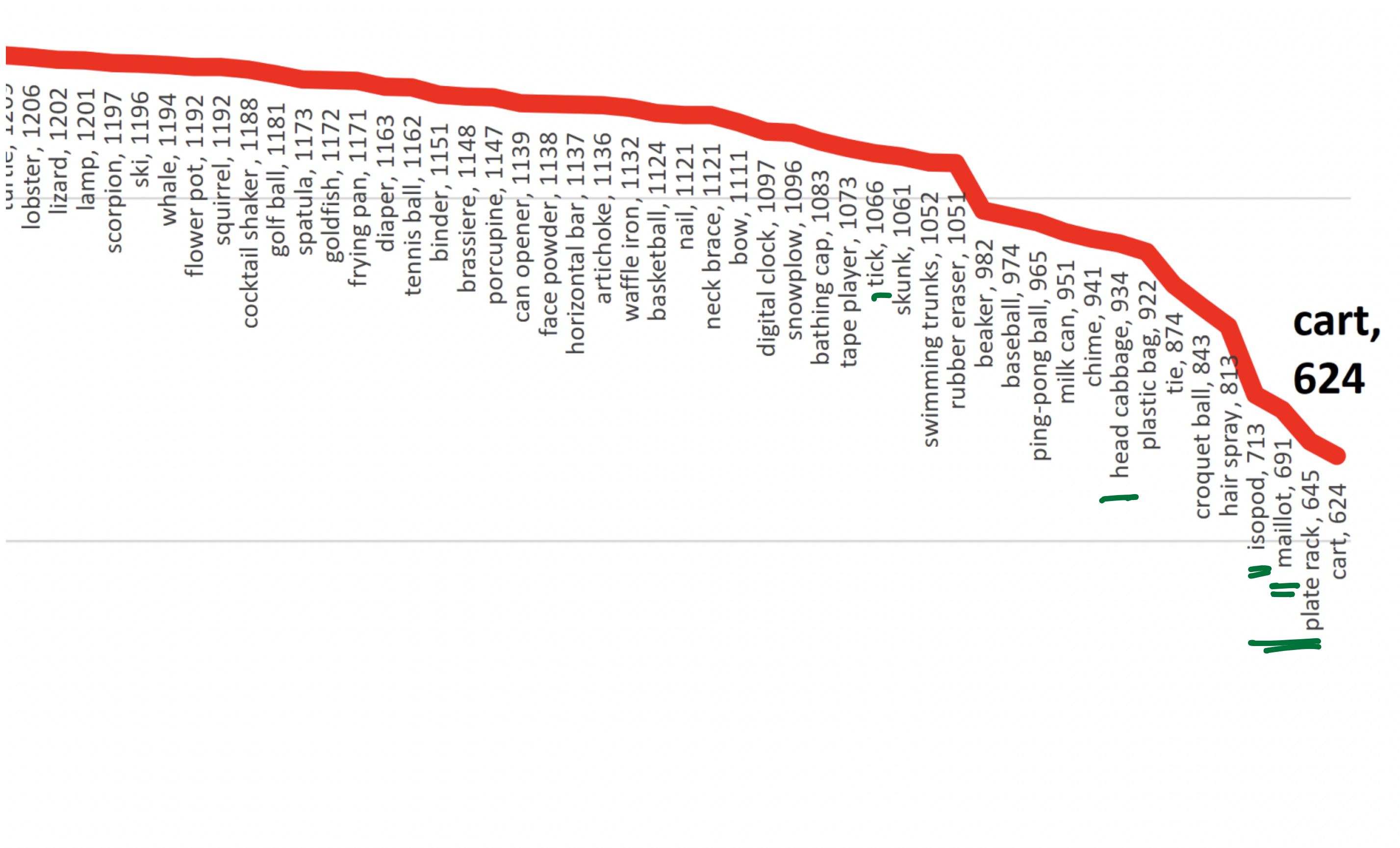
Data Sets

ILSVRC Data Set (Zoomed In)



Data Sets

ILSVRC Data Set (Zoomed In)



Let's take a look at the Data Sets

Dataset	Classes	Train			Validation			Test
		Images	Objects	Objects/Image	Images	Objects	Objects/Image	
PASCAL VOC 12	20	5,717	13,609	2.38	5,823	13,841	2.37	10,991
MS-COCO	80	118,287	860,001	7.27	5,000	36,781	7.35	40,670
~~ILSVRC~~	200	456,567	478,807	1.05 ?	20,121	55,501	2.76	40,152
OpenImage	600	1,743,042	14,610,229	8.38	41,620	204,621	4.92	125,436

Data Sets

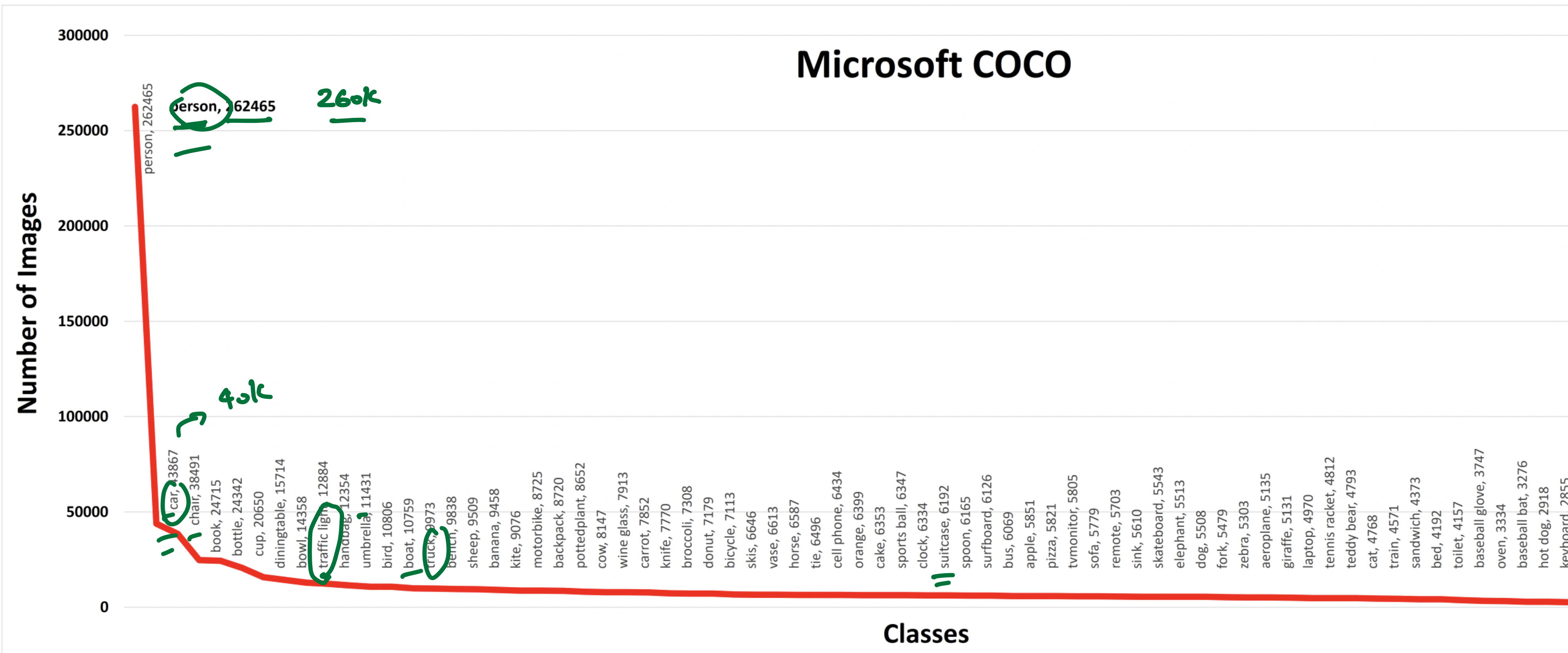
MS Coco Data Set

(b) MS-COCO

1
← Aeroplane
multiple Instances?

Data Sets

MS Coco Data Set

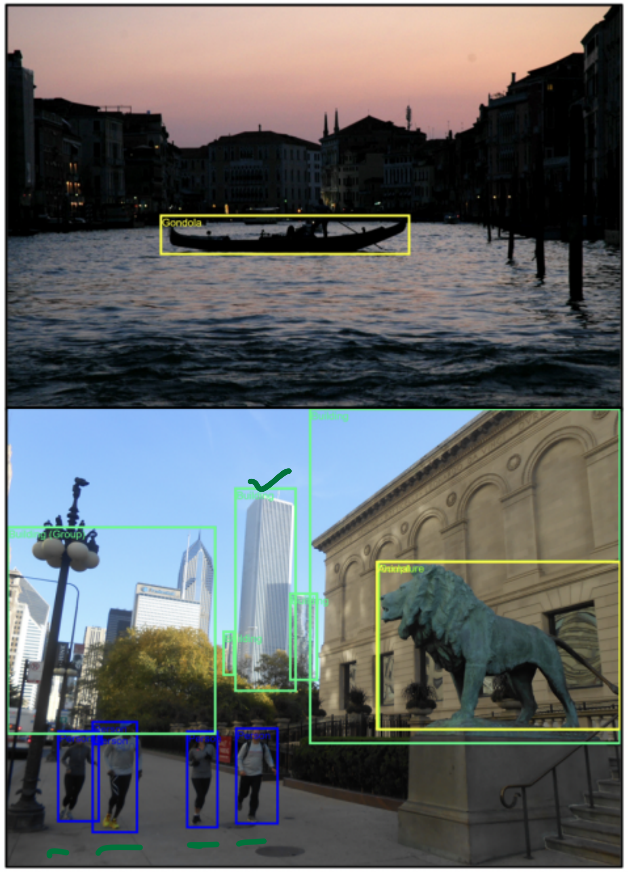


Let's take a look at the Data Sets

Dataset	Classes	Train			Validation			Test
		Images	Objects	Objects/Image	Images	Objects	Objects/Image	
PASCAL VOC 12	20	5,717	13,609	2.38	5,823	13,841	2.37	10,991
MS-COCO	80	118,287	860,001	7.27	5,000	36,781	7.35	40,670
ILSVRC	200	456,567	478,807	1.05	20,121	55,501	2.76	40,152
OpenImage	600	1,743,042	14,610,229	8.38	41,620	204,621	4.92	125,436

Data Sets

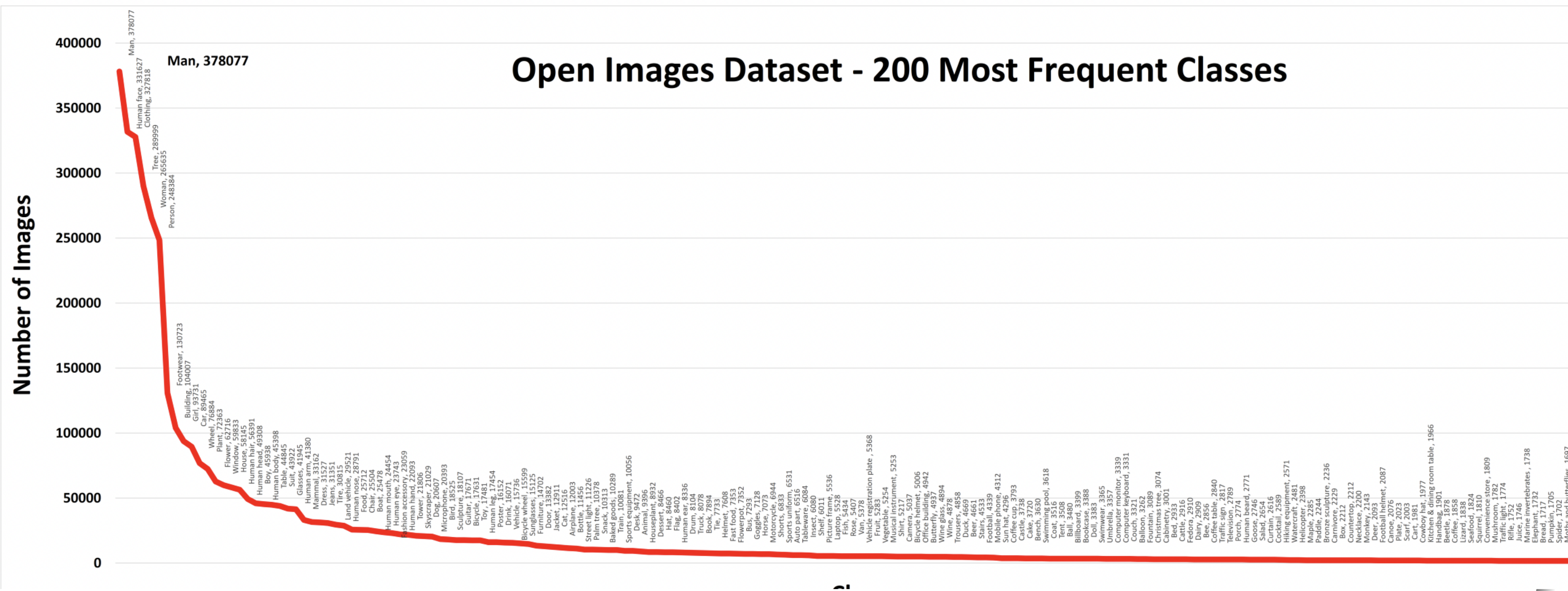
OpenImage Data Set



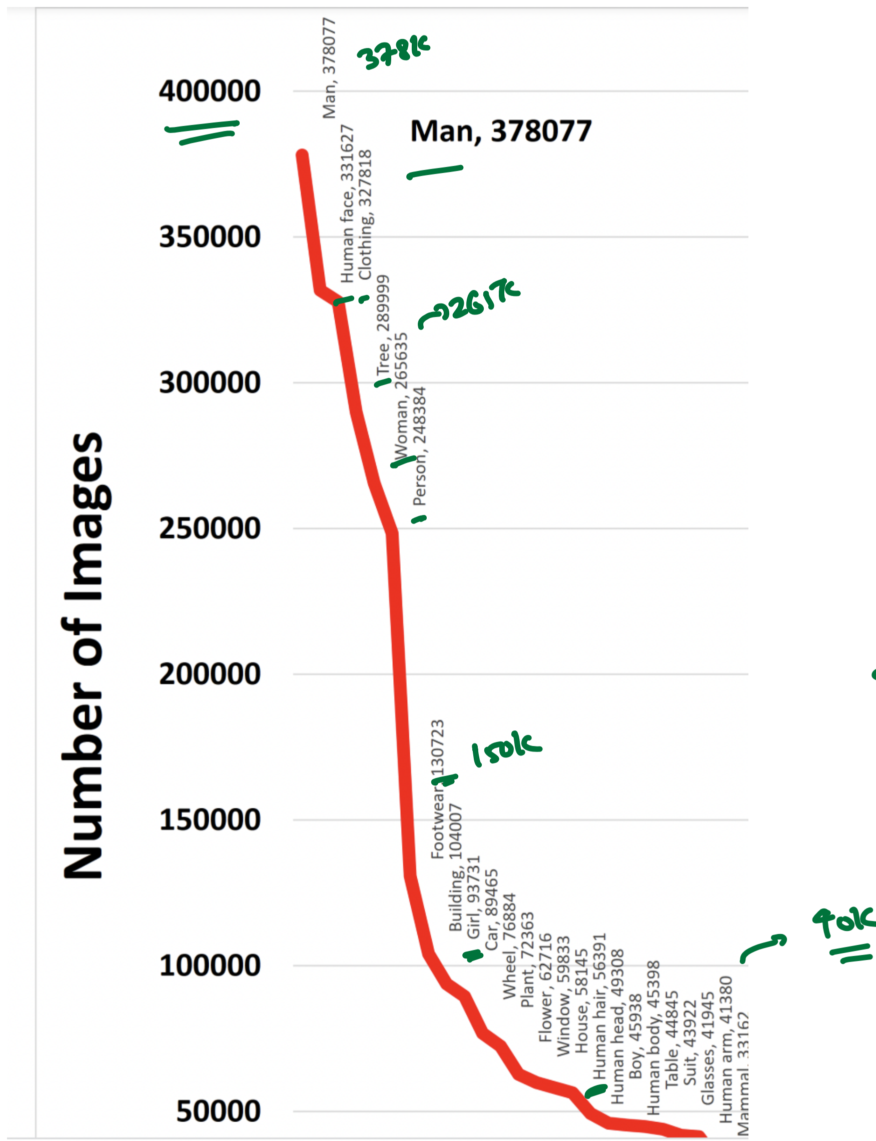
(d) OpenImage

Data Sets

OpenImage Data Set



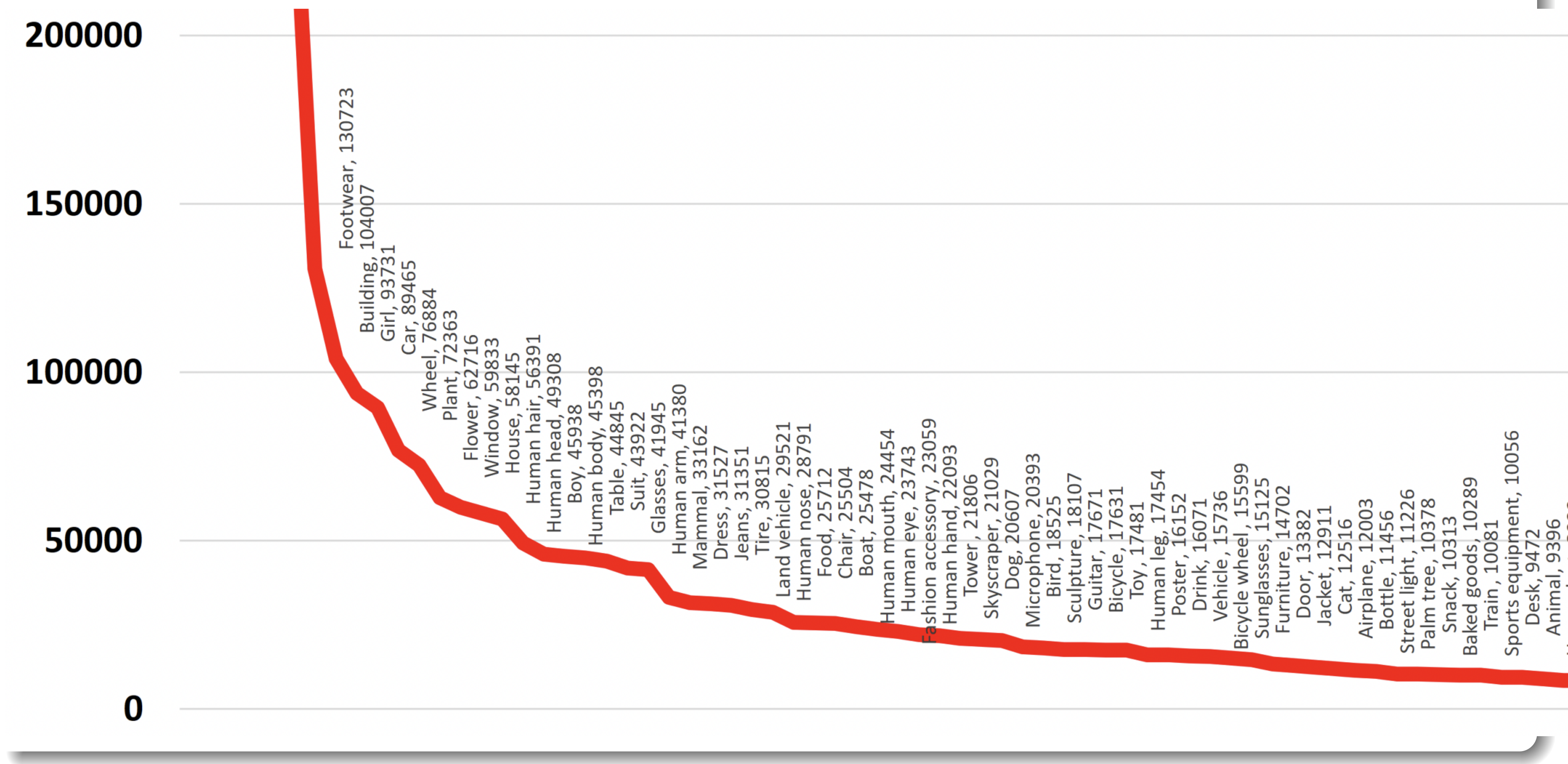
OpenImage Data Set (Zoomed In)



Significant class Imbalance

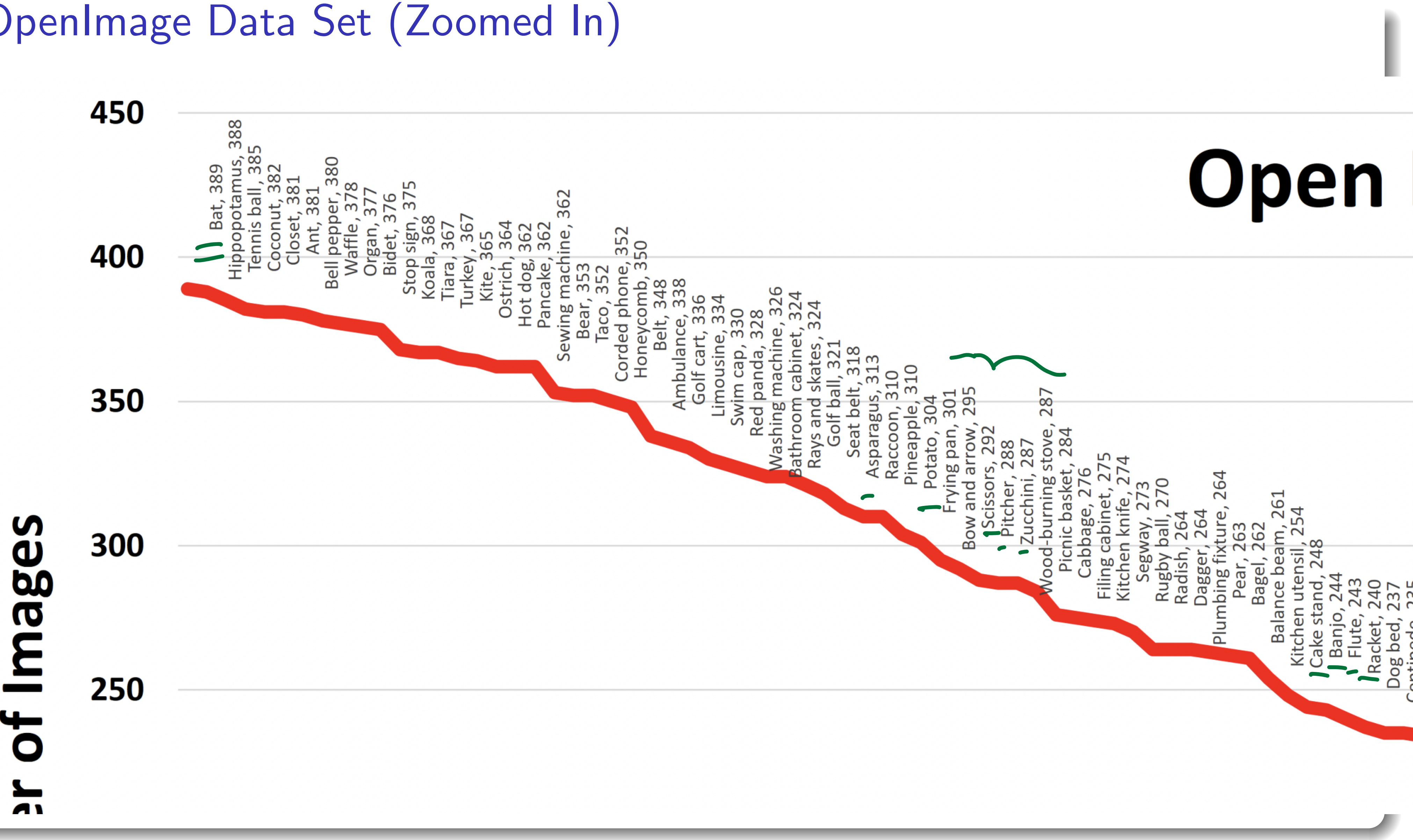
Data Sets

OpenImage Data Set (Zoomed In)



Data Sets

OpenImage Data Set (Zoomed In)



What does the data sets deep-dive show us?

1. **Intra-class variation** can exist - E.g. Husky dog vs Wolf? Regular Husky vs Siberian Husky? Robustness?

What does the data sets deep-dive show us?

How sensitive is the model to perturbations (Noise, Low-light, etc)

1. **Intra-class variation** can exist - E.g. Husky dog vs Wolf? Regular Husky vs Siberian Husky? Robustness?

~~Robustness~~
overfitting

2. **Too many classes**

Data Skew
(Koala Bears!)
Not a Real-world use-case (for obj. detection)

Data Bias
low acc. on minority class
May need Data Augmentation
Class Imbalance

Majority ⟷ minority class
Gap

What does the data sets deep-dive show us?

1. **Intra-class variation** can exist - E.g. Husky dog vs Wolf? Regular Husky vs Siberian Husky? Robustness?
2. **Too many classes**
3. **Efficiency** of object detection - Esp. with on-device inference

Prediction or Process of predicting

ICE #1

What's the issue from a machine learning stand point with having thousands of classes to detect in *object detection* on images? (more than one answer may apply)

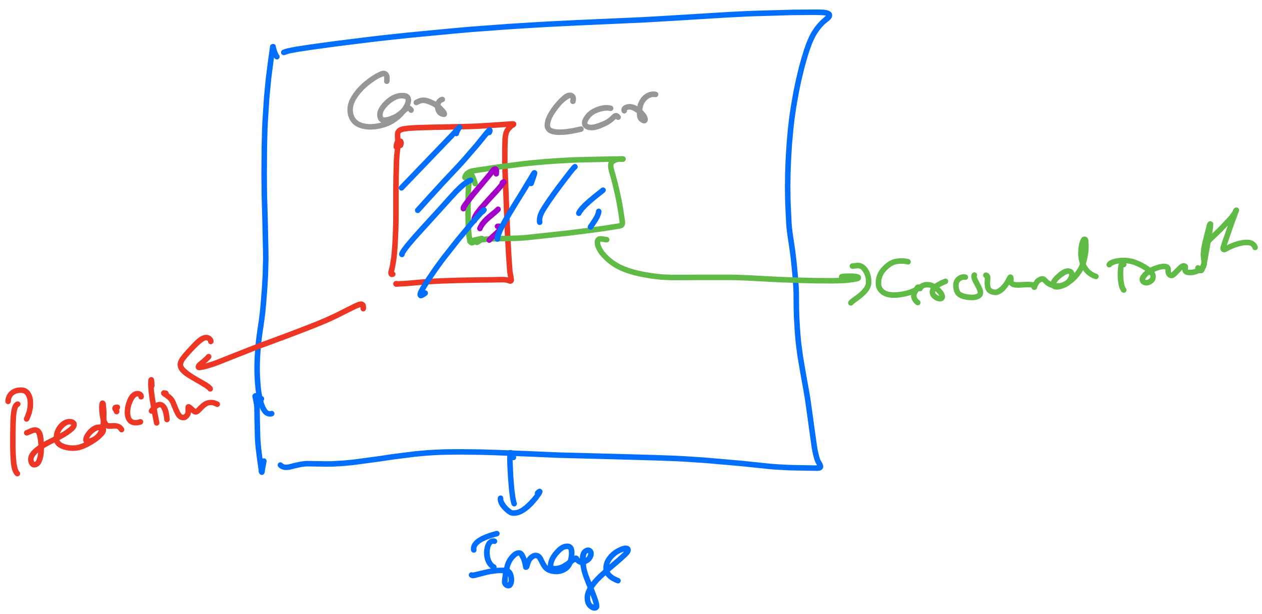
1. If the number of examples per class has a high variance, this can lead to model biasing towards the more frequent class being detected in images
2. If the there are not enough examples for a class, the model may not have accurate predictions for that class
3. The model may overfit on the more frequent class as compared to the less frequent class
4. Robustness issues of the model to small data perturbations can get exacerbated in this scenario

Object Detection Dimensions

1. Data Sets for benchmarking ✓
2. DL/CNN Models ✓ (R-CNN)
3. **Metrics**

IOU

Intersection over Union: Is the ratio of the area of the *intersection between the predicted bounding box and the ground truth bounding box* **over** *the union of the area between the predicted bounding box and the ground truth bounding box*



Metrics for Classification

IOU

Intersection over Union: Is the ratio of the area of the *intersection between the predicted bounding box and the ground truth bounding box* **over** *the union of the area between the predicted bounding box and the ground truth bounding box*

Threshold to use for Precision

MAP @ 0.5 IOU

Average Precision (AP) @ 0.5 IOU: If IOU

$$>$$

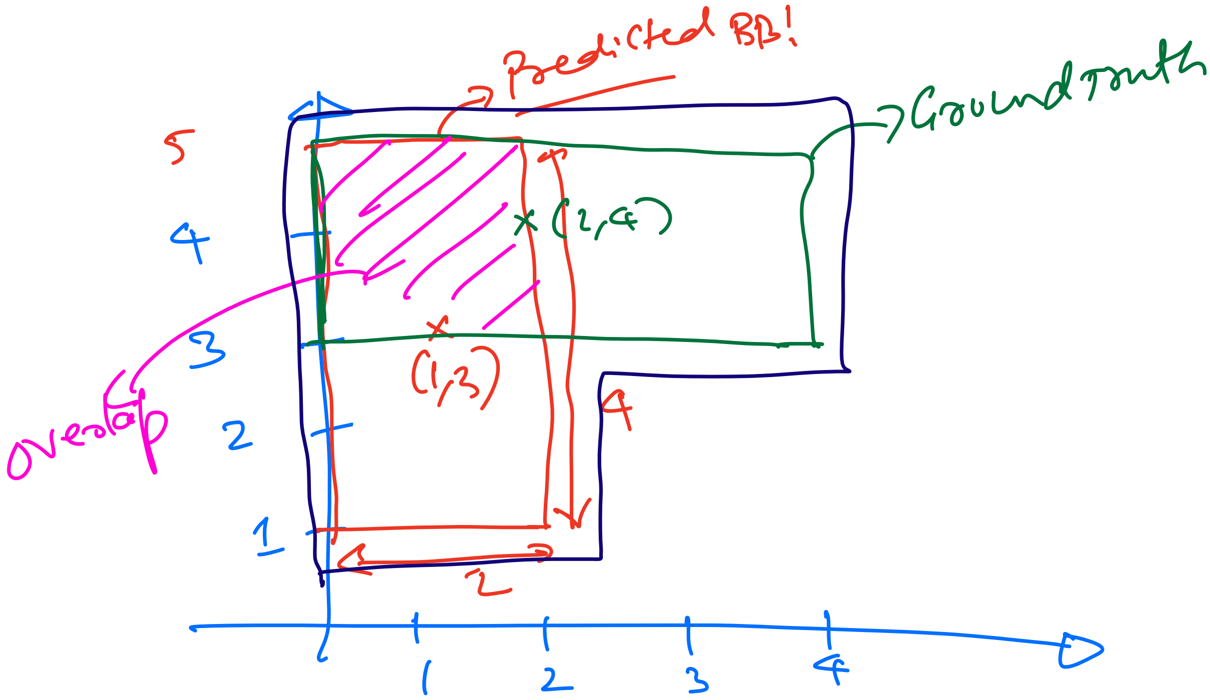
0.5 across examples of a given class, count the precision as 1, else 0. Average Precision is the average of all the precisions in a given class.

ICE #2

IOU computation

Let's say we have an image as above. The model predicts a horse centered at coordinates $(1, 3)$ with a bounding box around it of width 2 and height of 4. The ground truth predicts a horse centered at coordinates $(2, 4)$ with a bounding box around it of width 4 and height of 2. What's the IOU ratio between the model's prediction and the ground truth? **Hint:** Plot the coordinates and bounding box on paper and then compute the IOU!

1. 0.2
2. 0.3
3. 0.4
4. 0.5



$$IoU = \frac{2 \times 2}{2 \times 4 + 2 \times 2} = \frac{4}{12} = \frac{1}{3}!$$

ICE #3

AP @ 0.5 IOU

So we want to compute the AP or *Average Precision* @ 0.5 IOU, a unique metric that the $R - CNN$ paper came up with as a unique metric for the Object Detection problem. Suppose we want to measure the AP for a given class - Say the class of horses. For every candidate object in the horse class (note there could be multiple candidates per image!), we look at the IOU and threshold it compute precision and subsequently the average precision. Let's say the IOU for the horse class over 10 candidate objects (for which the model predicted horse) looks as follows: $\{0.7, 0.3, 0, 0.51, 0.49, 0.2, 0.8, 0.6, 0, 0.57\}$ What's the AP @ 0.5 IOU in this scenario?

Precision

$0.7 > 0.5 \Rightarrow 1$

$0.3 < 0.5 \Rightarrow 0$

⋮

1. 0.3
2. 0.4
3. 0.5
4. 0.6

Breakout for Takeaways!

Discuss Takeaways (5 mins)

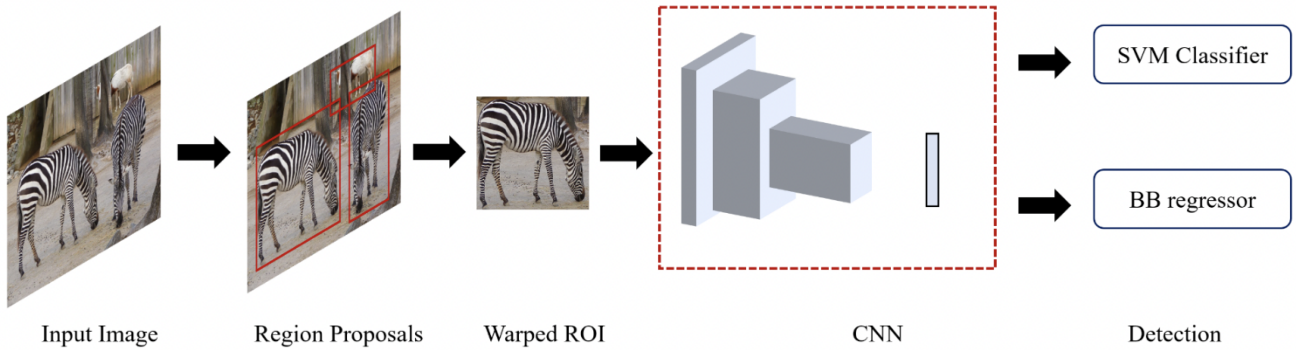
From today's lecture in your zoom group

Next Class

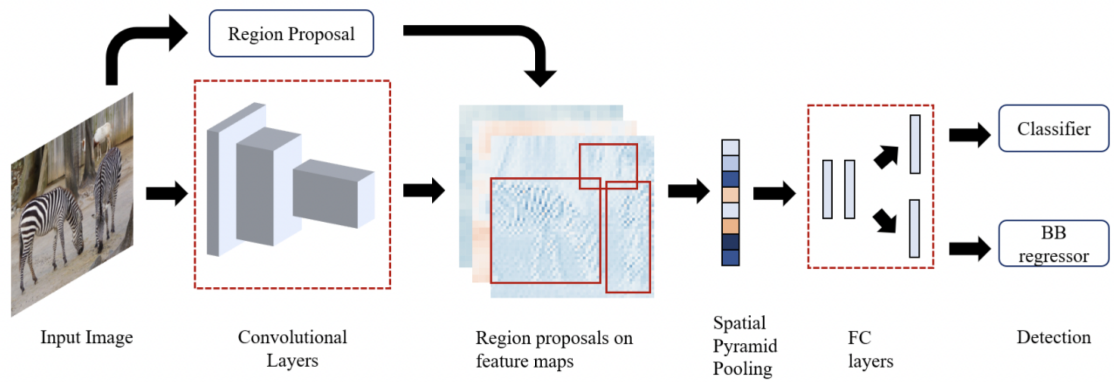
1. Object Detection Recap
2. R-CNN variants - Fast and Faster R-CNN
3. Results and Benchmarking on the data sets
4. Image Segmentation (Maybe)

R-CNN variants

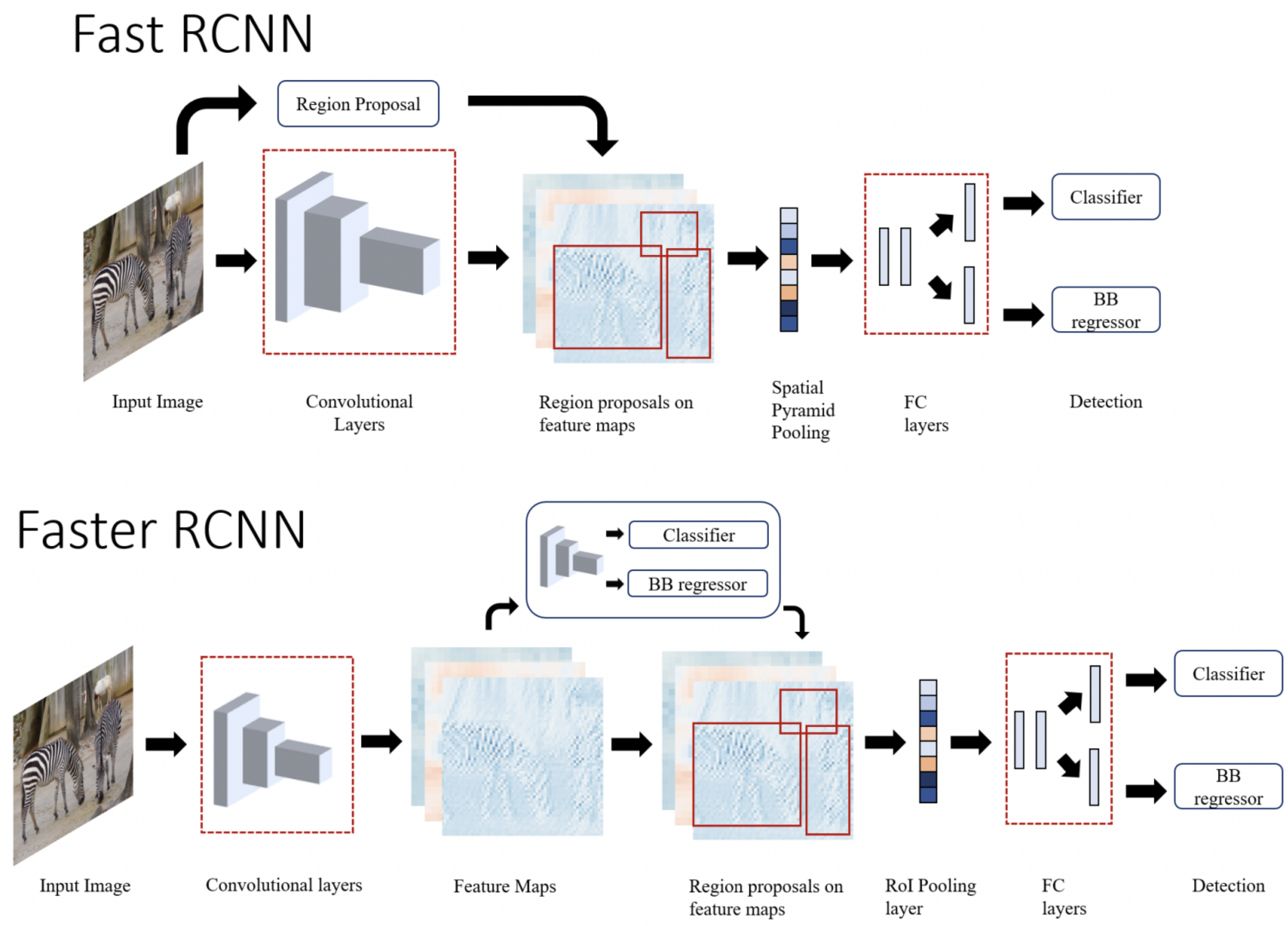
RCNN



Fast RCNN



R-CNN variants



Results for Object Detection

Model	Year	Backbone	Size	$AP_{[0.5:0.95]}$	$AP_{0.5}$
R-CNN*	2014	AlexNet	224	-	58.50%
SPP-Net*	2015	ZF-5	Variable	-	59.20%
Fast R-CNN*	2015	VGG-16	Variable	-	65.70%
Faster R-CNN*	2016	VGG-16	600	-	67.00%
R-FCN	2016	ResNet-101	600	31.50%	53.20%
FPN	2017	ResNet-101	800	36.20%	59.10%
Mask R-CNN	2018	ResNeXt-101-FPN	800	39.80%	62.30%
DetectoRS	2020	ResNeXt-101	1333	53.30%	71.60%
YOLO*	2015	(Modified) GoogLeNet	448	-	57.90%
SSD	2016	VGG-16	300	23.20%	41.20%
YOLOv2	2016	DarkNet-19	352	21.60%	44.00%
RetinaNet	2018	ResNet-101-FPN	400	31.90%	49.50%
YOLOv3	2018	DarkNet-53	320	28.20%	51.50%
CenterNet	2019	Hourglass-104	512	42.10%	61.10%
EfficientDet-D2	2020	Efficient-B2	768	43.00%	62.30%
YOLOv4	2020	CSPDarkNet-53	512	43.00%	64.90%
Swin-L	2021	HTC++	-	57.70%	-

[a]Models marked with * are compared on PASCAL VOC 2012, while others on MS COCO.Rows colored gray are real-time detectors