Training Region-based Object Detectors with Online Hard Example Mining

Abhinav Shrivastava[1] Abhinav Gupta[1] Ross Girshick[2]

[1]Carnegie Mellon University [2]Facebook AI Research

Work done when Abhinav Shrivastava and Ross Girshick were at Microsoft Research.

Object Detection

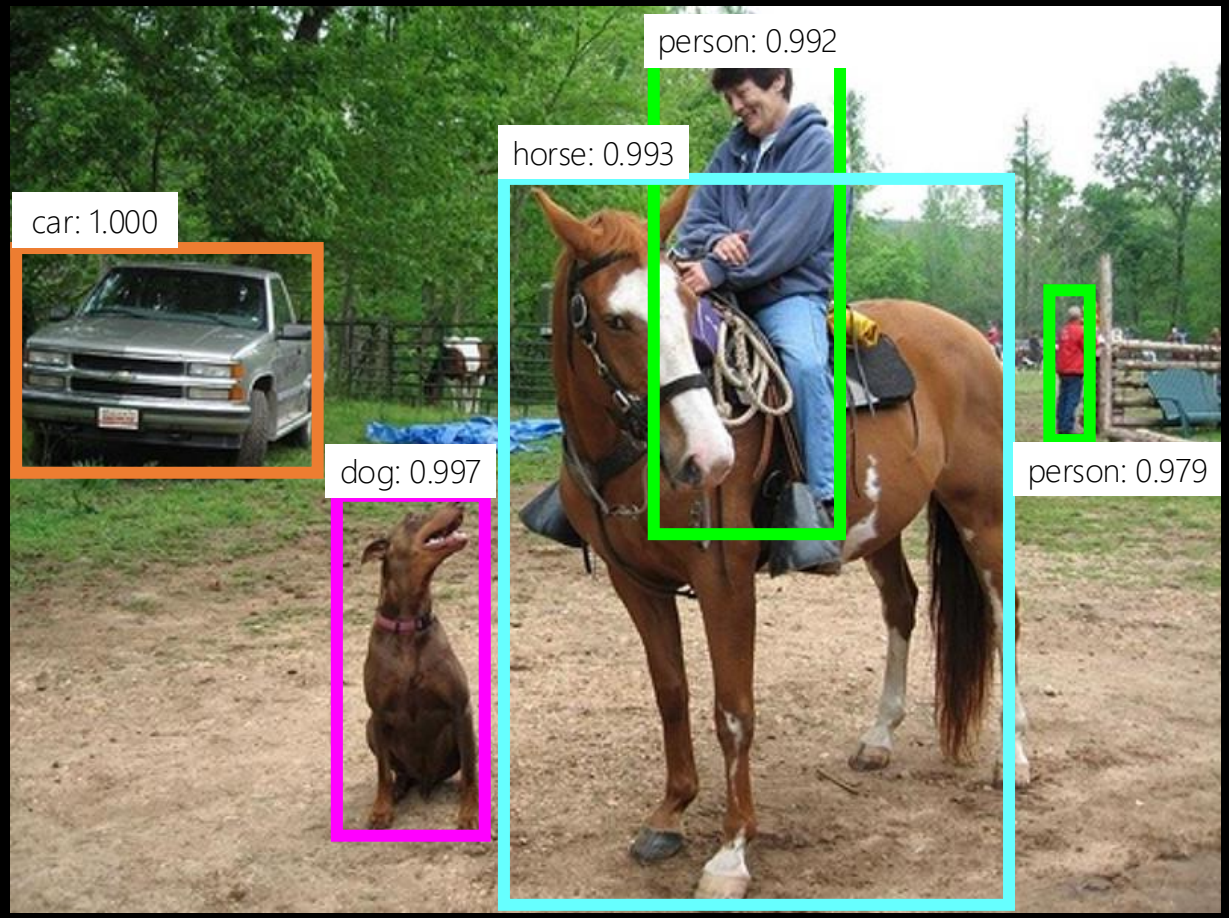


Image from Ross Girshick & Kaiming He

Object Detection

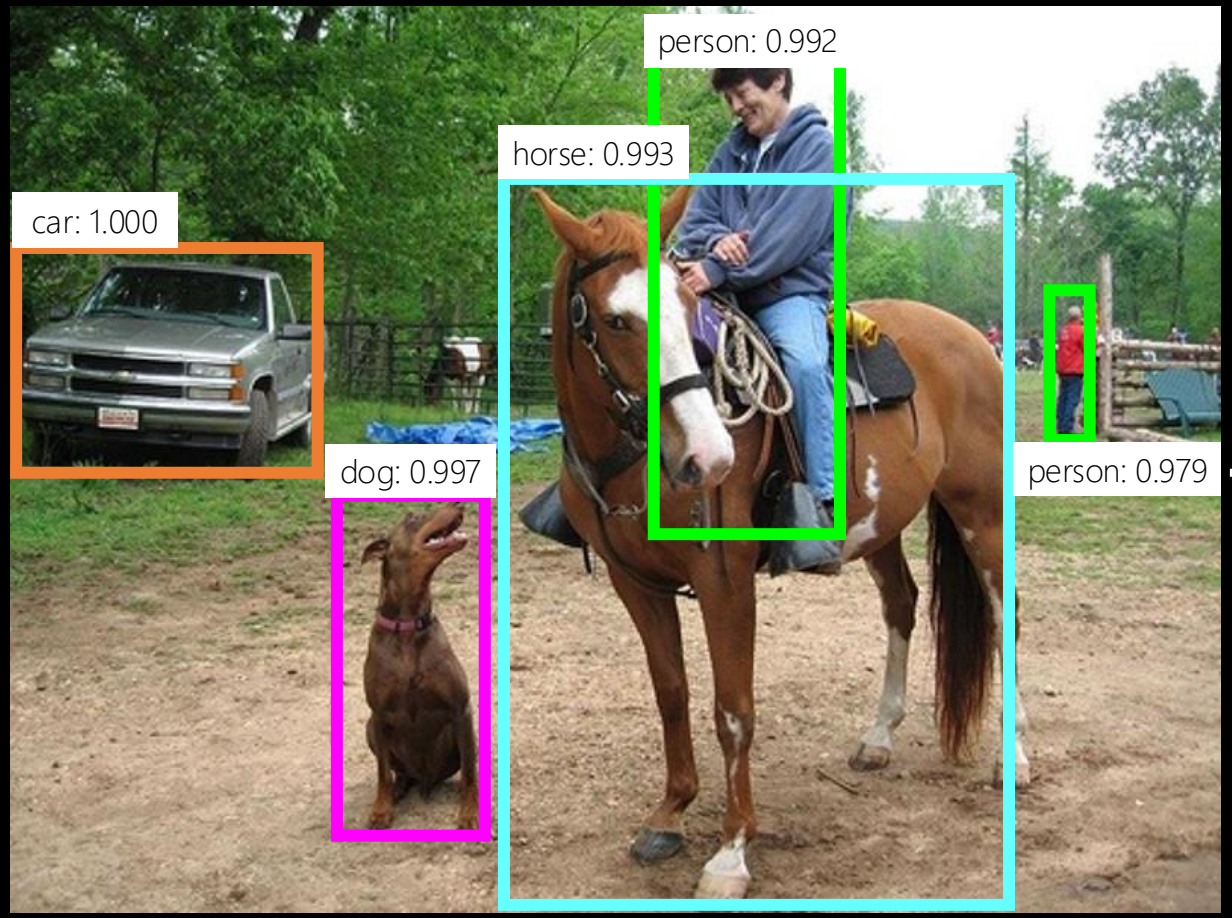
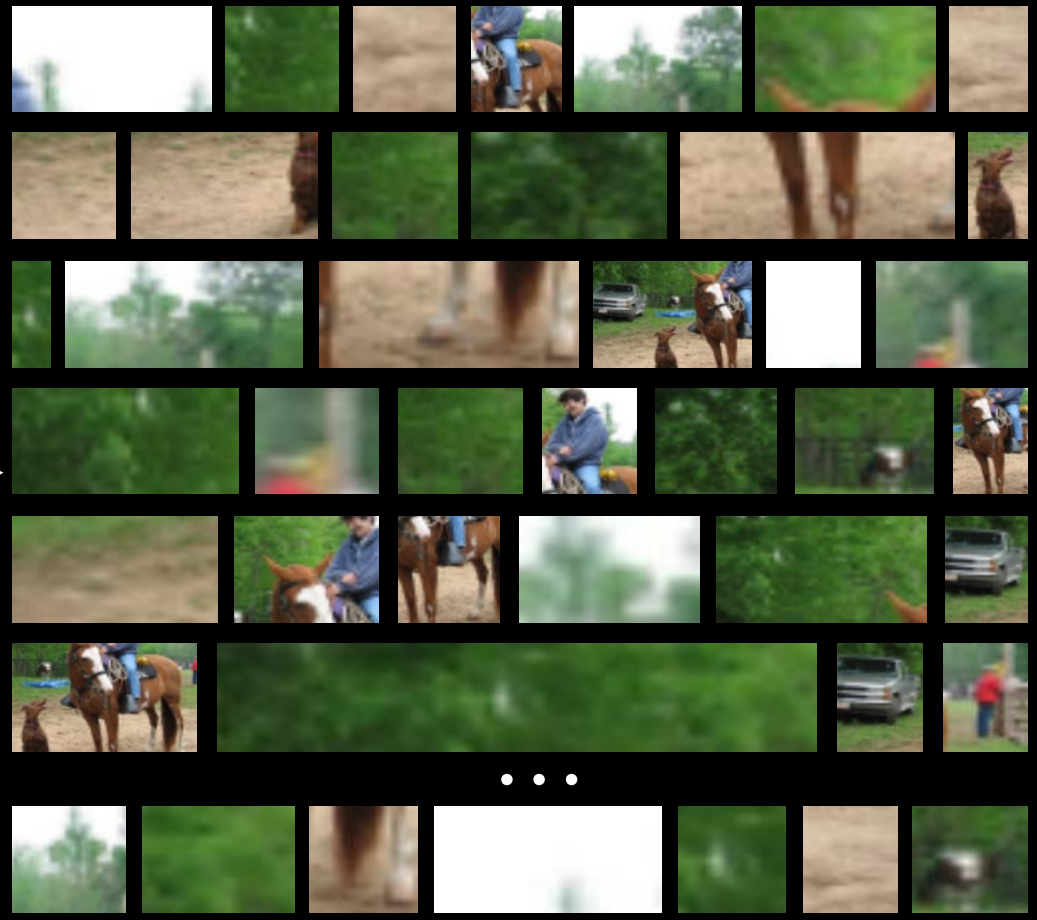


Image Classification Reduction



Object Detection

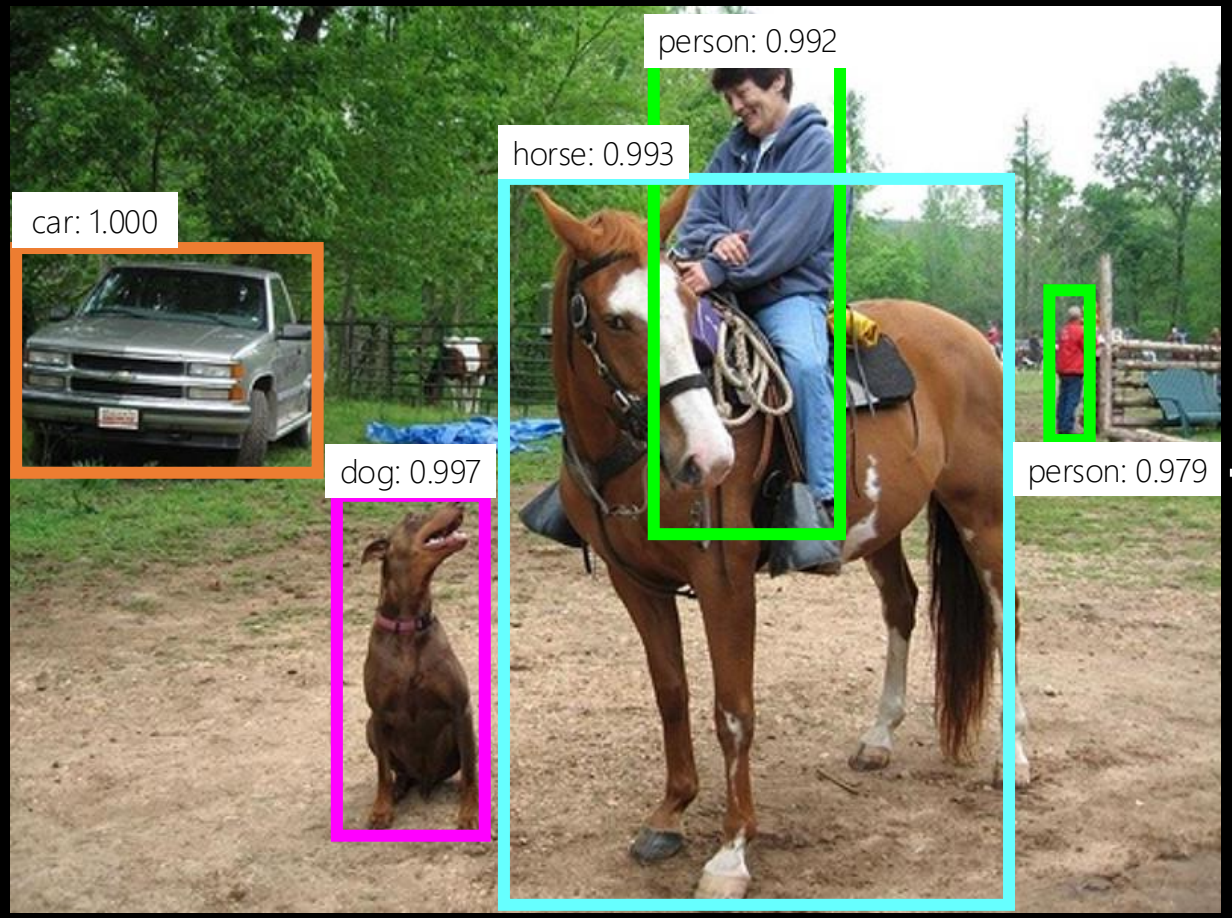
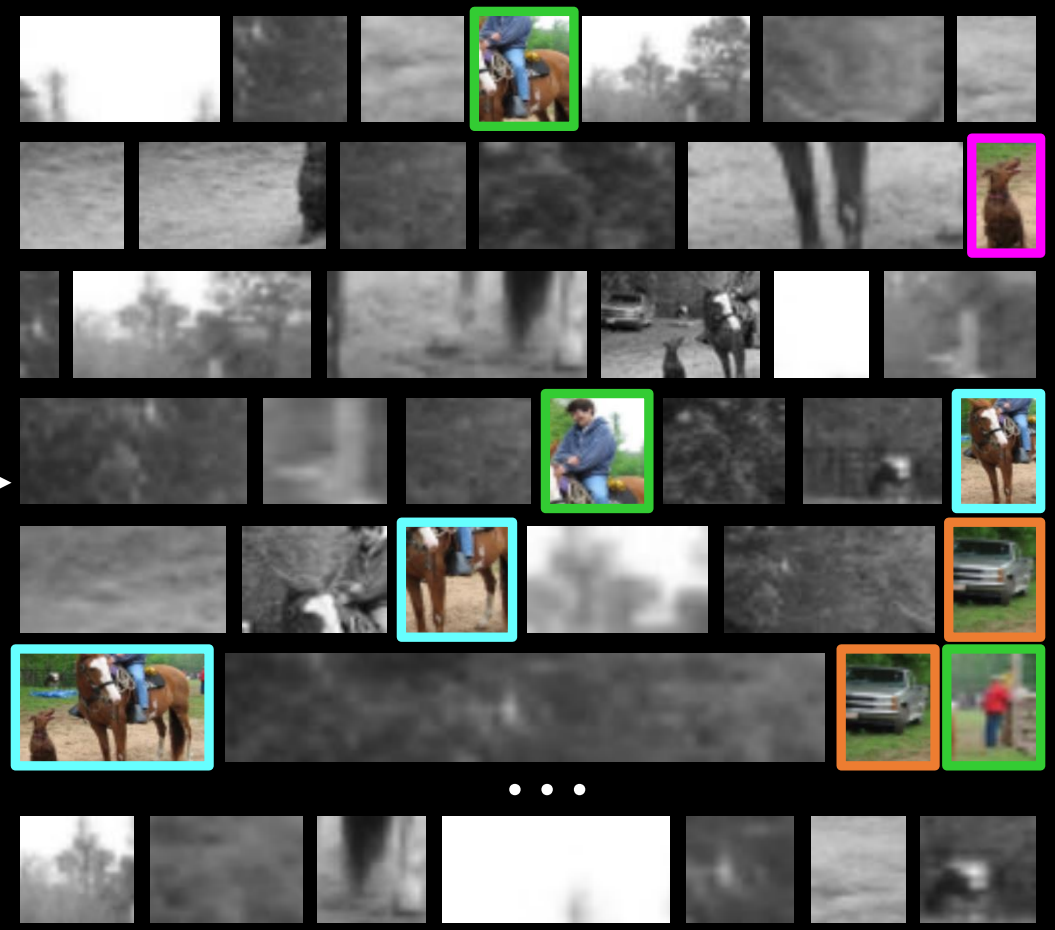


Image Classification Reduction



Object Detection

Foreground << Background

	fg : bg
Sliding-window Object Detector (e.g., DPM)	1 : 100,000
Region-based Object Detector (e.g., R-CNN)	1 : 70

Image Classification Reduction

R-CNN [Girshick, 2014], DPM [Felzenszwalb, 2010]

Bootstrapping

A “bootstrap” paradigm for
selecting useful training examples and for
incrementally training object and pattern detection systems
to arbitrary levels of robustness.

[Sung and Poggio, 1994]

Bootstrapping

A “bootstrap” paradigm for
selecting useful training examples and for
incrementally training object and pattern detection systems
to arbitrary levels of robustness.

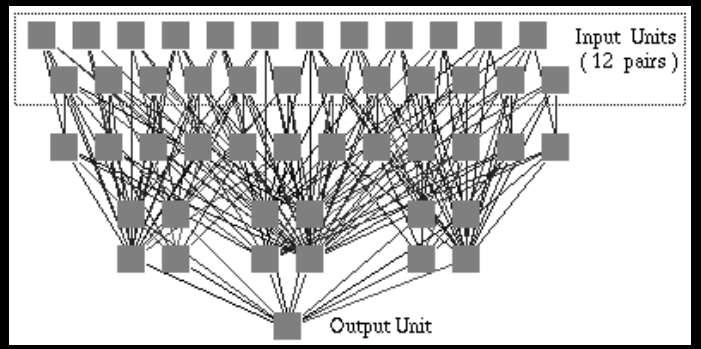
[Sung and Poggio, 1994]

Bootstrapping

[Sung and Poggio, 1994] version

Face Detector

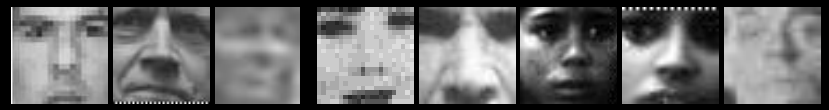
(Multi-layer Perceptron)



Train

Training Set

`face' examples



`non-face' examples

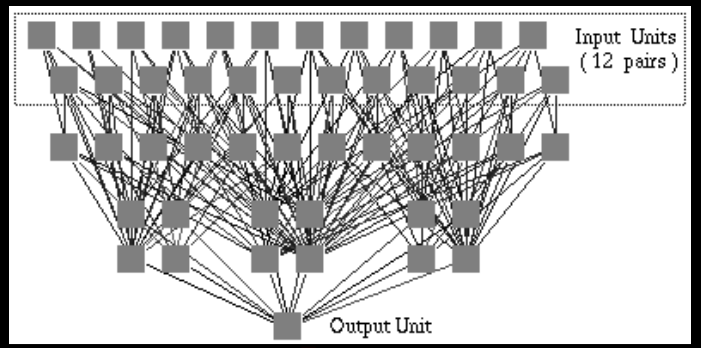
Images from [Sung and Poggio, 1994]

BOOTSTRAPPING

[Sung and Poggio, 1994] version

Face Detector

(Multi-layer Perceptron)

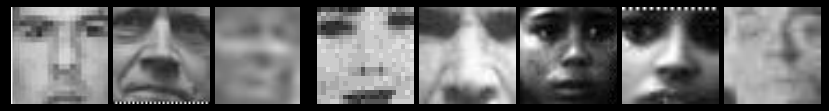


Train

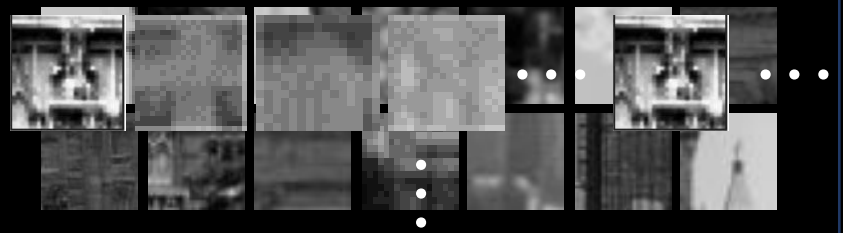
`non-face' images

Training Set

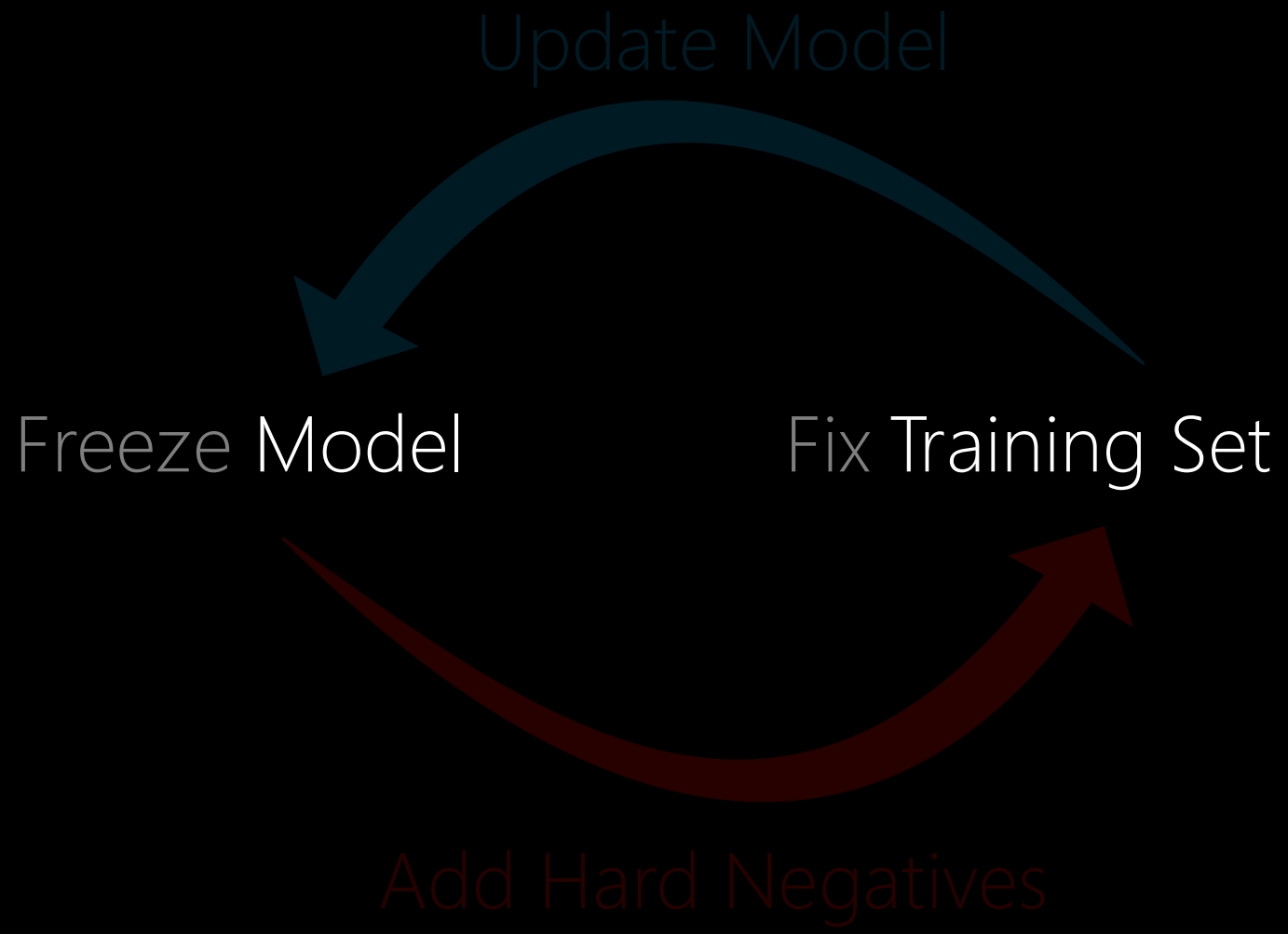
`face' examples



`non-face' examples



Bootstrapping



1. Fix Training Set
   Update Model
2. Freeze Model
   Add Hard Negatives
3. Redo

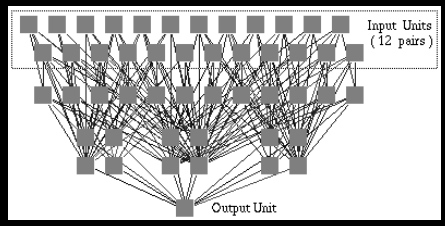
Bootstrapping

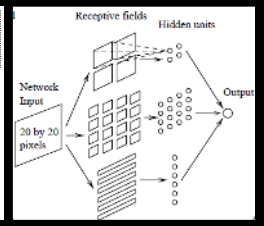
Hard Negative Mining

Shallow Neural Networks

[Sung & Poggio, 1994]

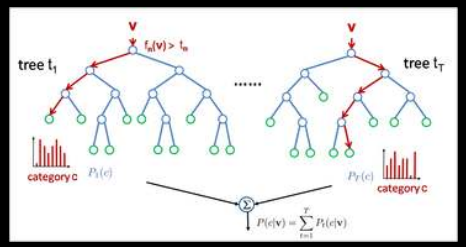
[Rowley et al., 1998]





Boosted Decision Trees

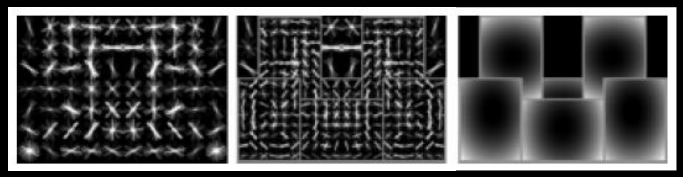
[Dollár et al., 2009]



SVMs, LSVMs

[Dalal & Triggs, 2005]

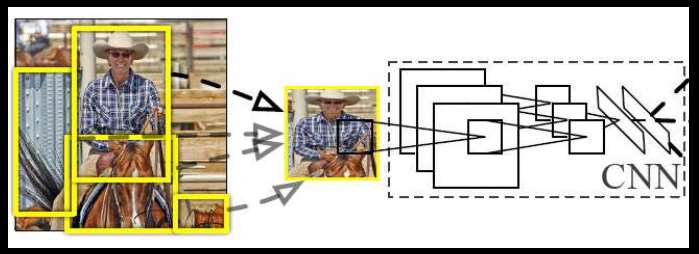
DPM [Felzenszwalb et al., 2010]



ConvNet Features + SVMs

R-CNN [Girshick et al., 2014]

SPPNet [He et al., 2014]



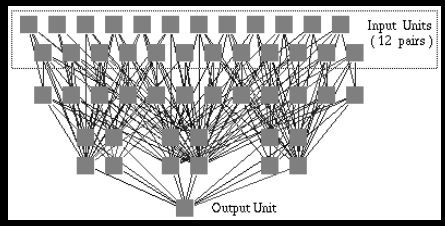
Bootstrapping

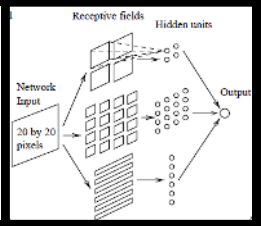
Hard Negative Mining

Shallow Neural Networks

[Sung & Poggio, 1994]

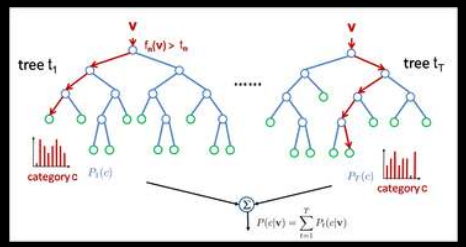
[Rowley et al., 1998]





Boosted Decision Trees

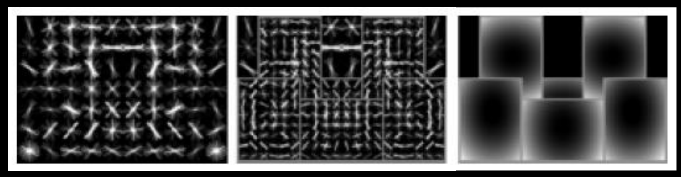
[Dollár et al., 2009]



SVMs, LSVMs

[Dalal & Triggs, 2005]

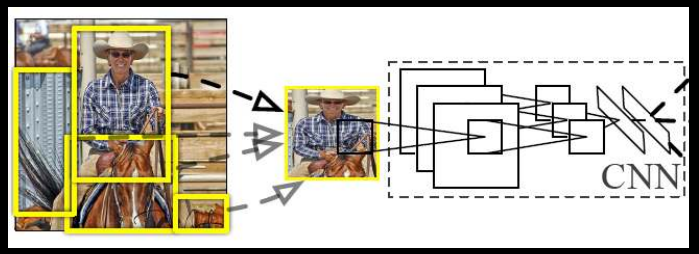
DPM [Felzenszwalb et al., 2010]



ConvNet Features + SVMs

R-CNN [Girshick et al., 2014]

SPPNet [He et al., 2014]



Multi-stage Training → End-to-end Training

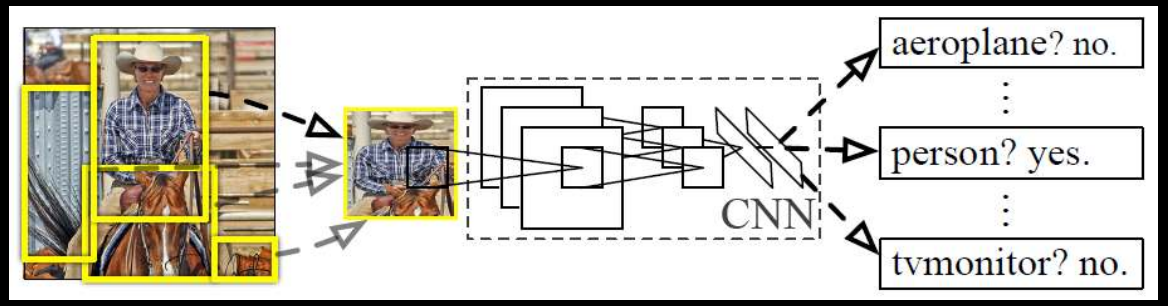
ConvNet Features + SVMs

R-CNN [Girshick et al., 2014]

SPPNet [He et al., 2014]

Stage 1: Features

Stage 2: SVMs



R-CNN [Girshick et al., 2014]

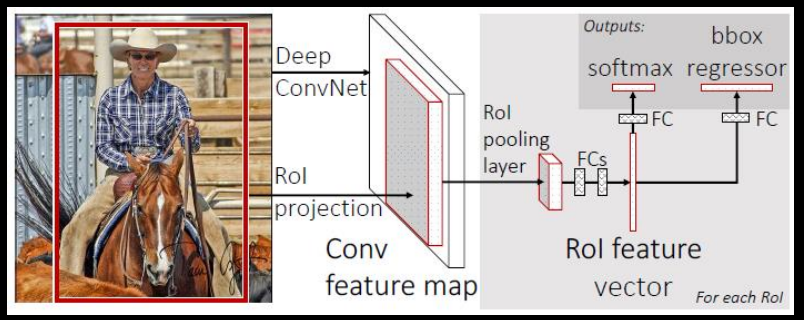
ConvNet Detectors

Fast R-CNN [Girshick, 2015]

Faster R-CNN [Ren et al., 2015]

End-to-end

Multi-task Loss



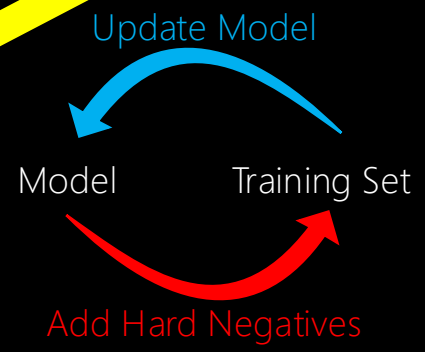
Fast R-CNN [Girshick, 2015]

Stochastic Gradient Descent (SGD)

Why is the traditional Hard Mining not trivial?

1. Fix Training Set
2. Update Model
3. Freeze Model
4. Add Hard Negatives
5. Redo

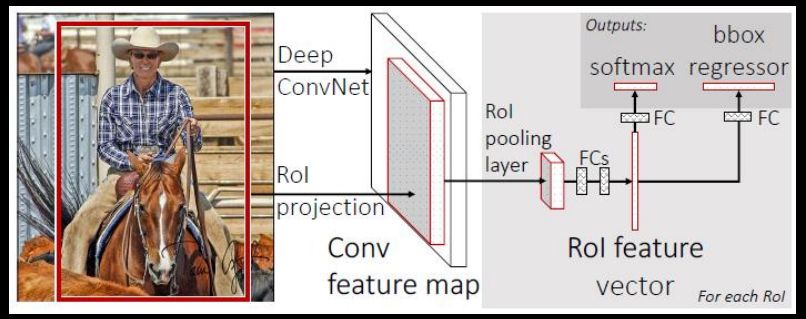
ONLINE



End-to-end ConvNet Detectors

Fast R-CNN [Girshick, 2015]

Faster R-CNN [Ren et al., 2015]



Fast R-CNN [Girshick, 2015]

Stochastic Gradient Descent (SGD)

100,000s iterations

OHEM: Online Hard Example Mining

Simple modification to SGD

- Sample from non-uniform, non-stationary distribution

Simplifies Training

- Removes several heuristics & hyperparameters

Consistent & Significant Performance Gains

- Applicable to state-of-the-art ConvNet Detectors

Fast R-CNN with SGD

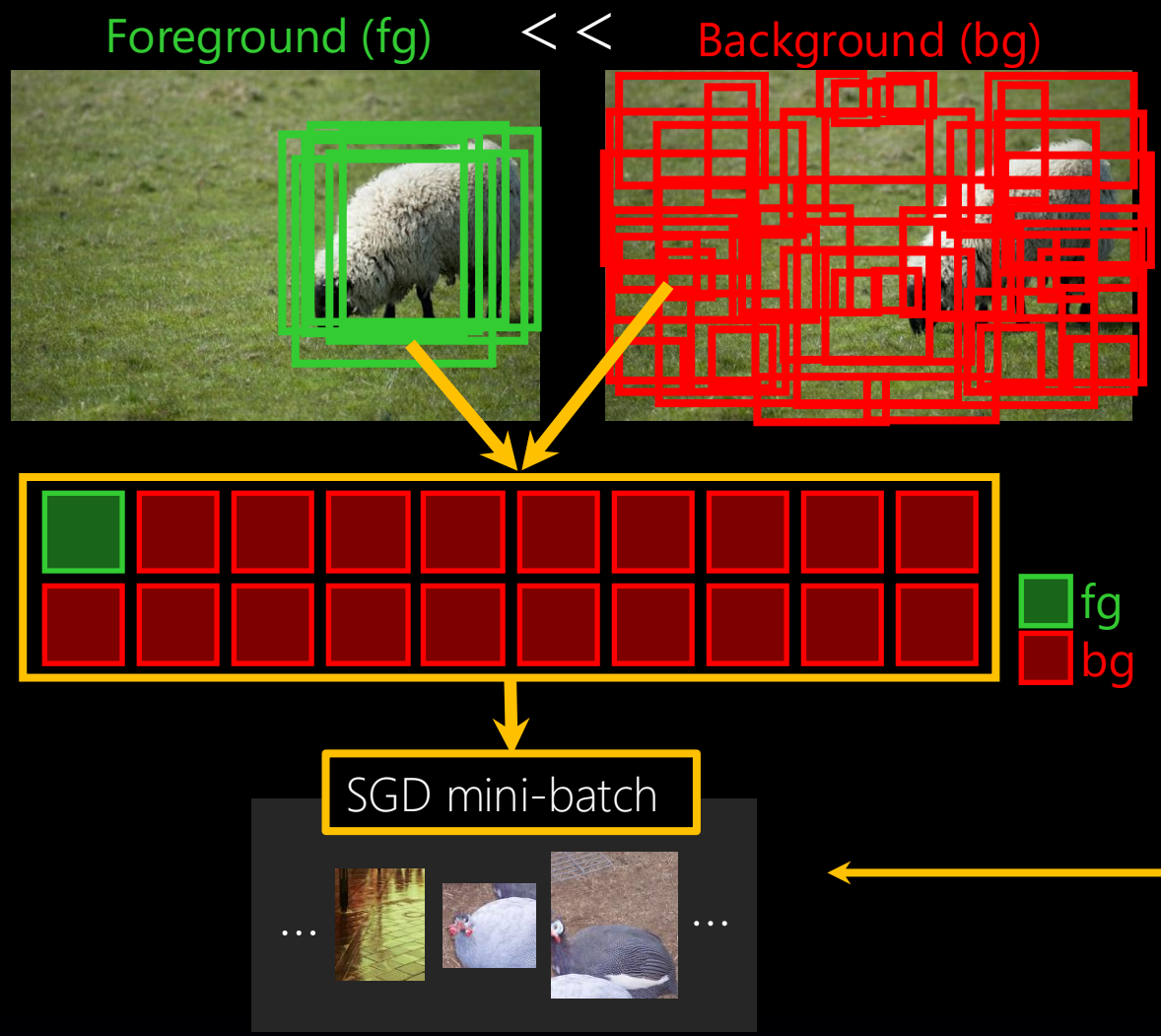
Hierarchical Sampling

1. Sample a small number of images, **Randomly**

2. Sample many examples from each image

Fast R-CNN [Girshick, 2015]

Fast R-CNN with SGD

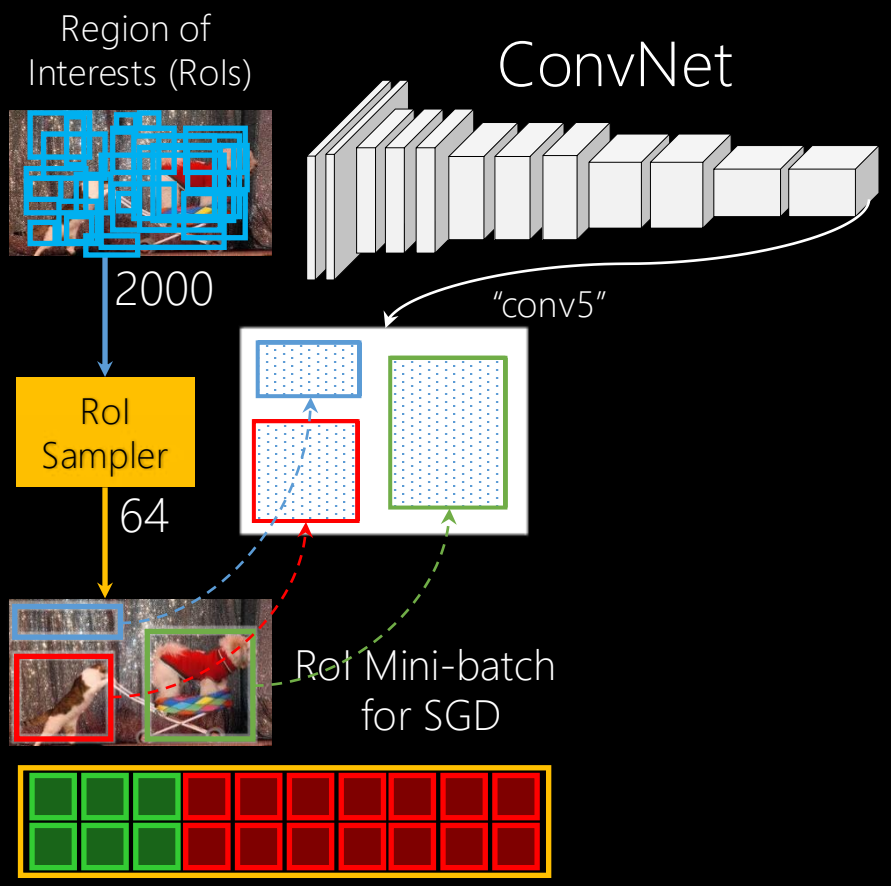


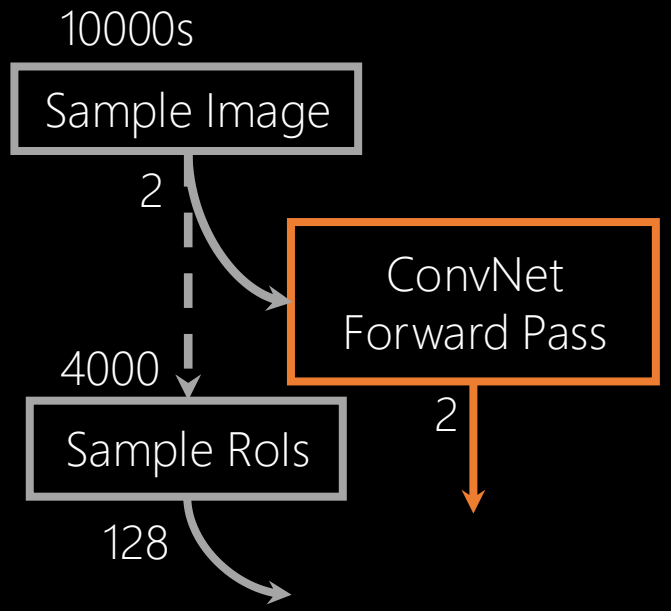
Hierarchical Sampling

1. Sample a small number of images, **Randomly**

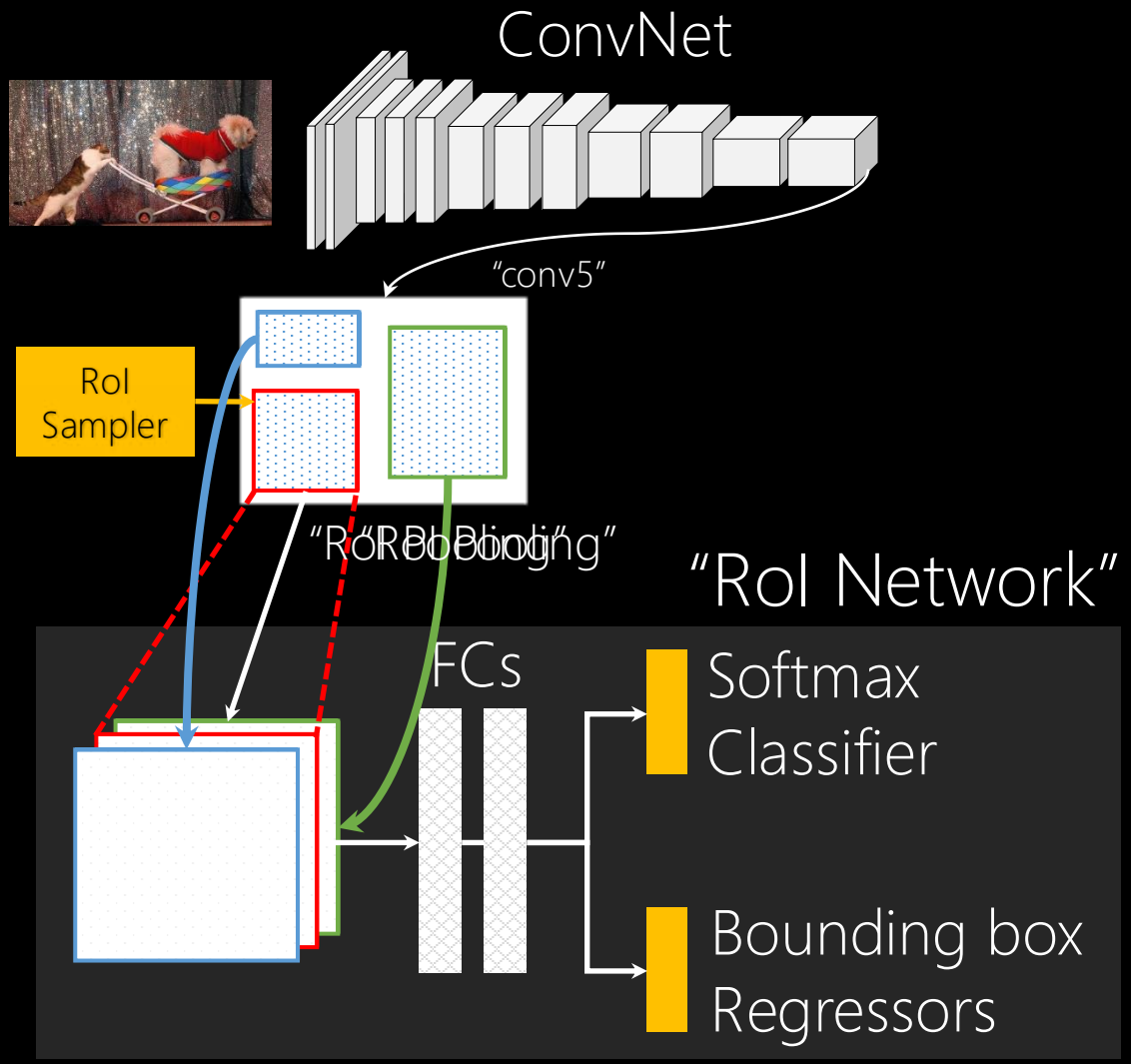
~~Many heuristics & hyperparameters~~

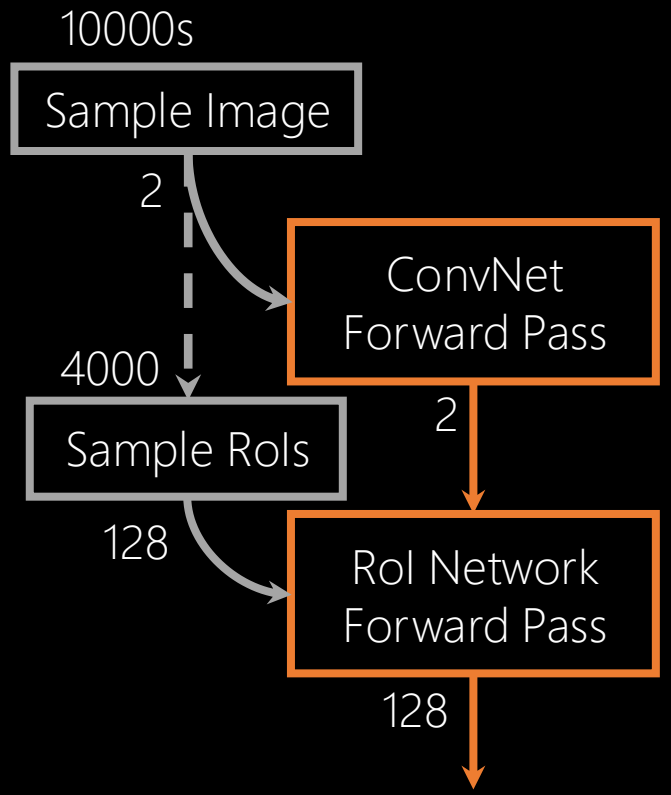
Fast R-CNN with SGD



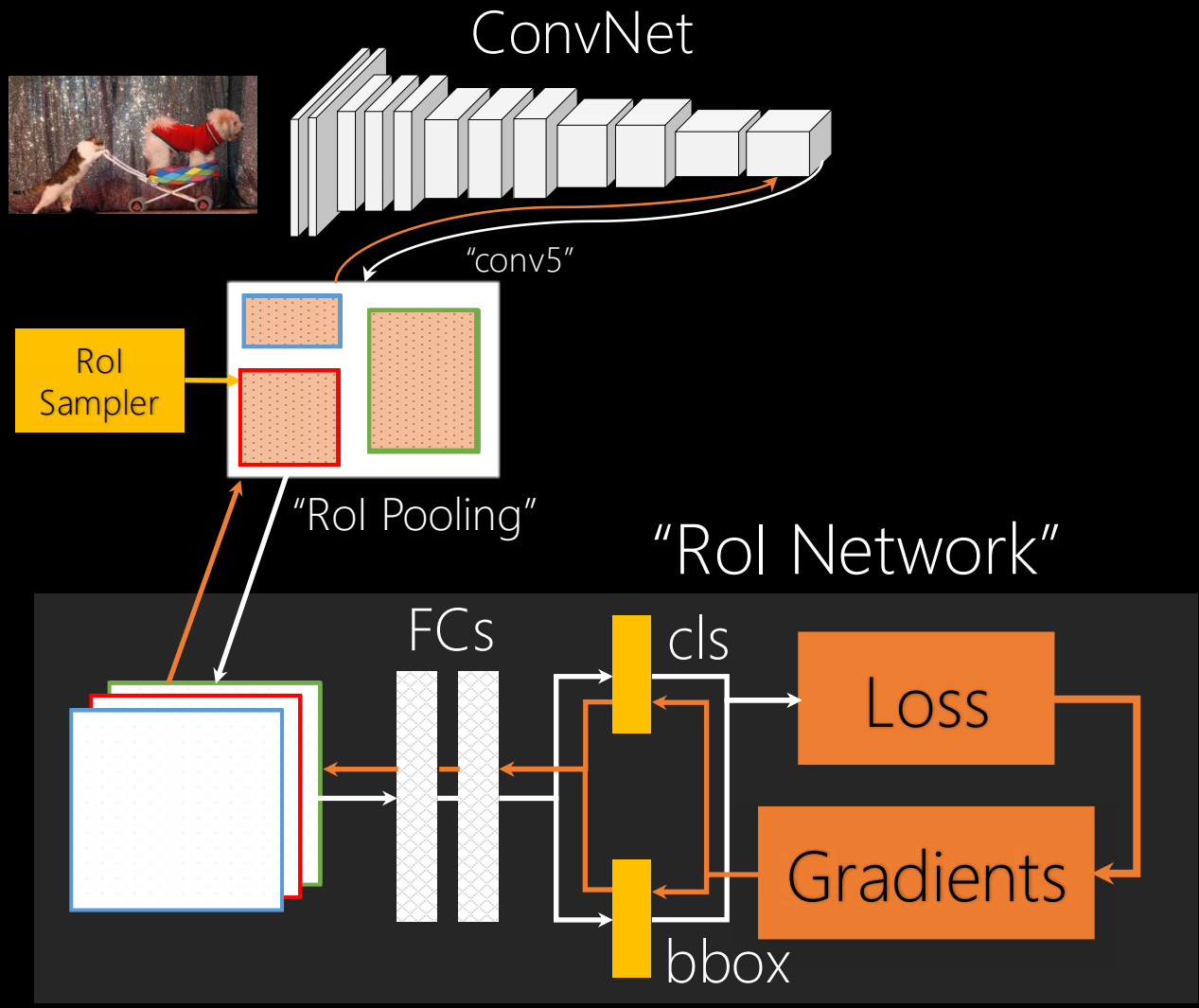


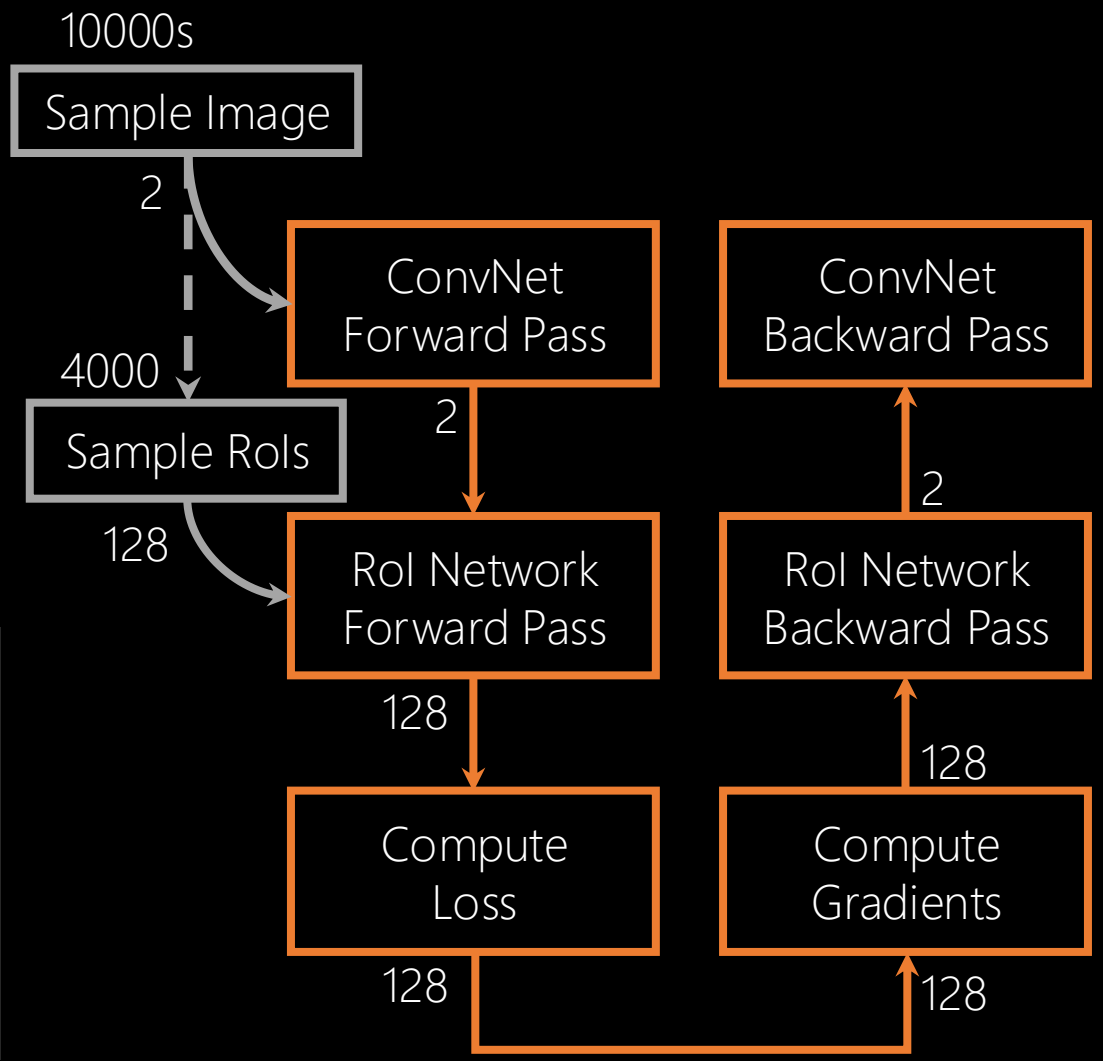
Fast R-CNN with SGD





Fast R-CNN with SGD



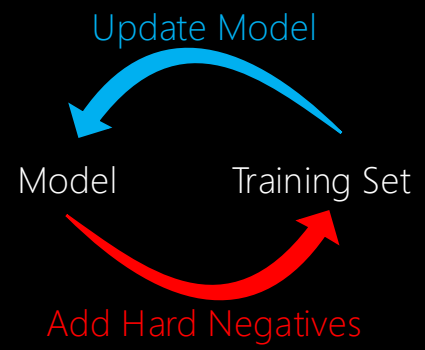


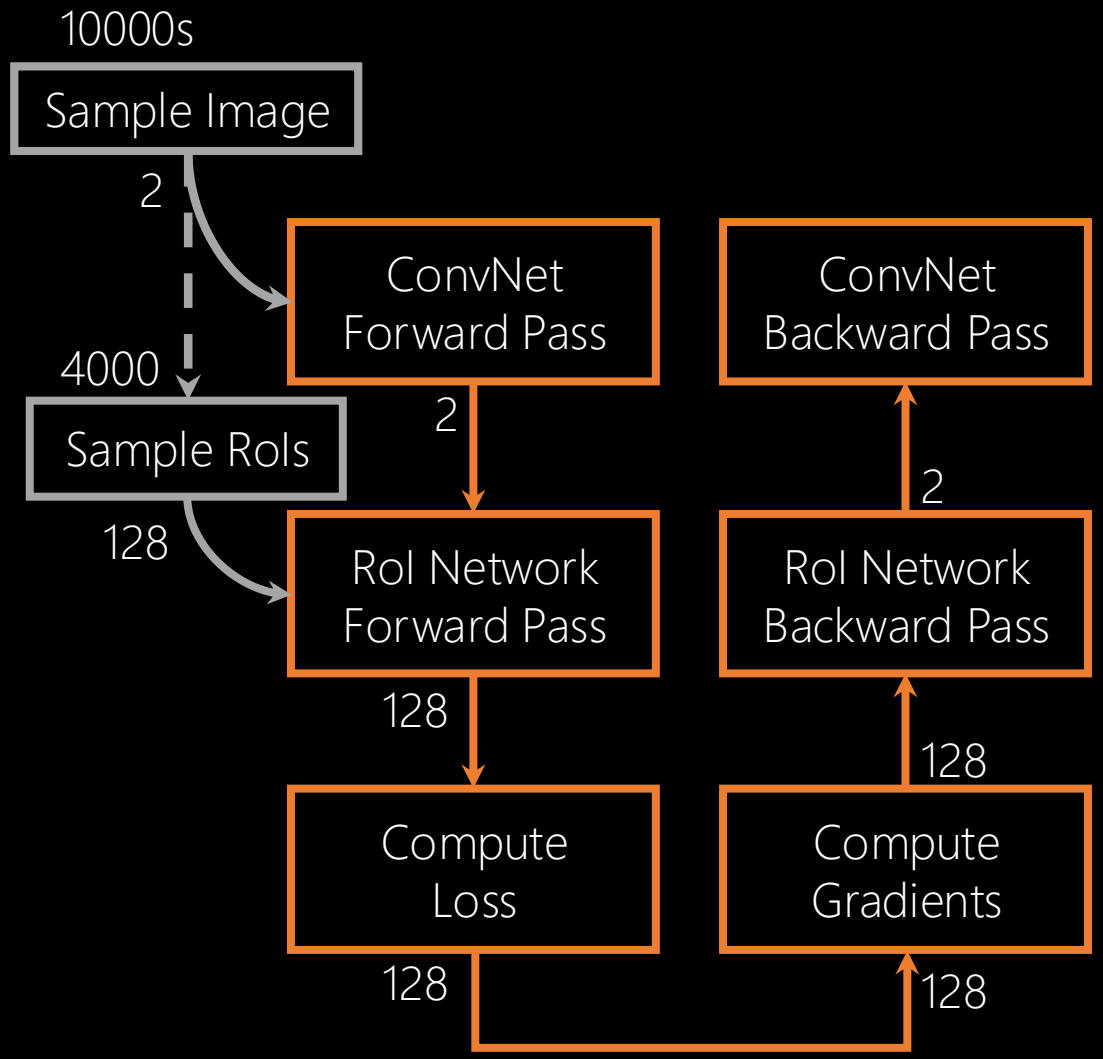
Fast R-CNN with Online Hard Example Mining

1. Fix Training Set
   Update Model

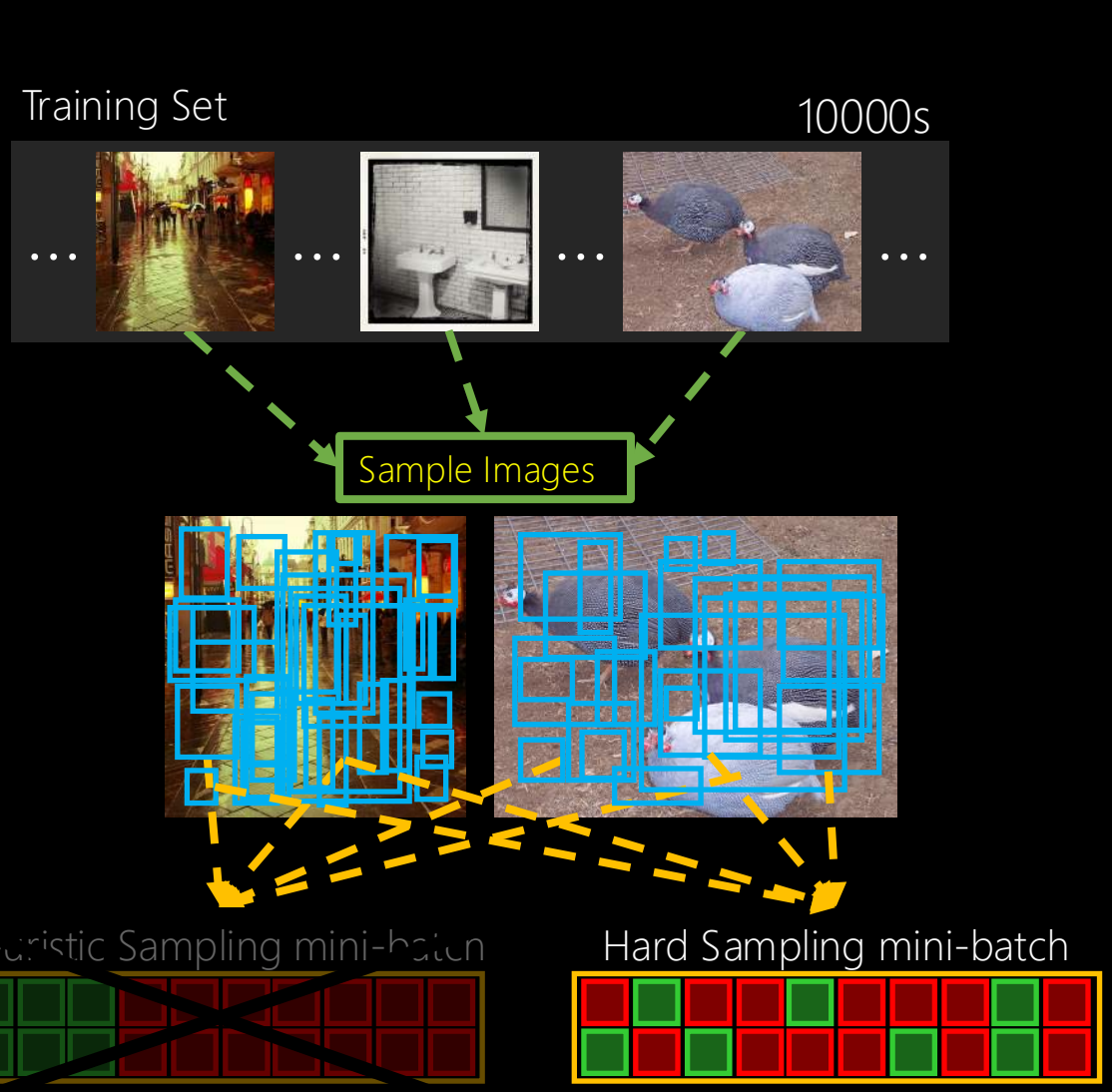
2. Freeze Model
   Add Hard Negatives

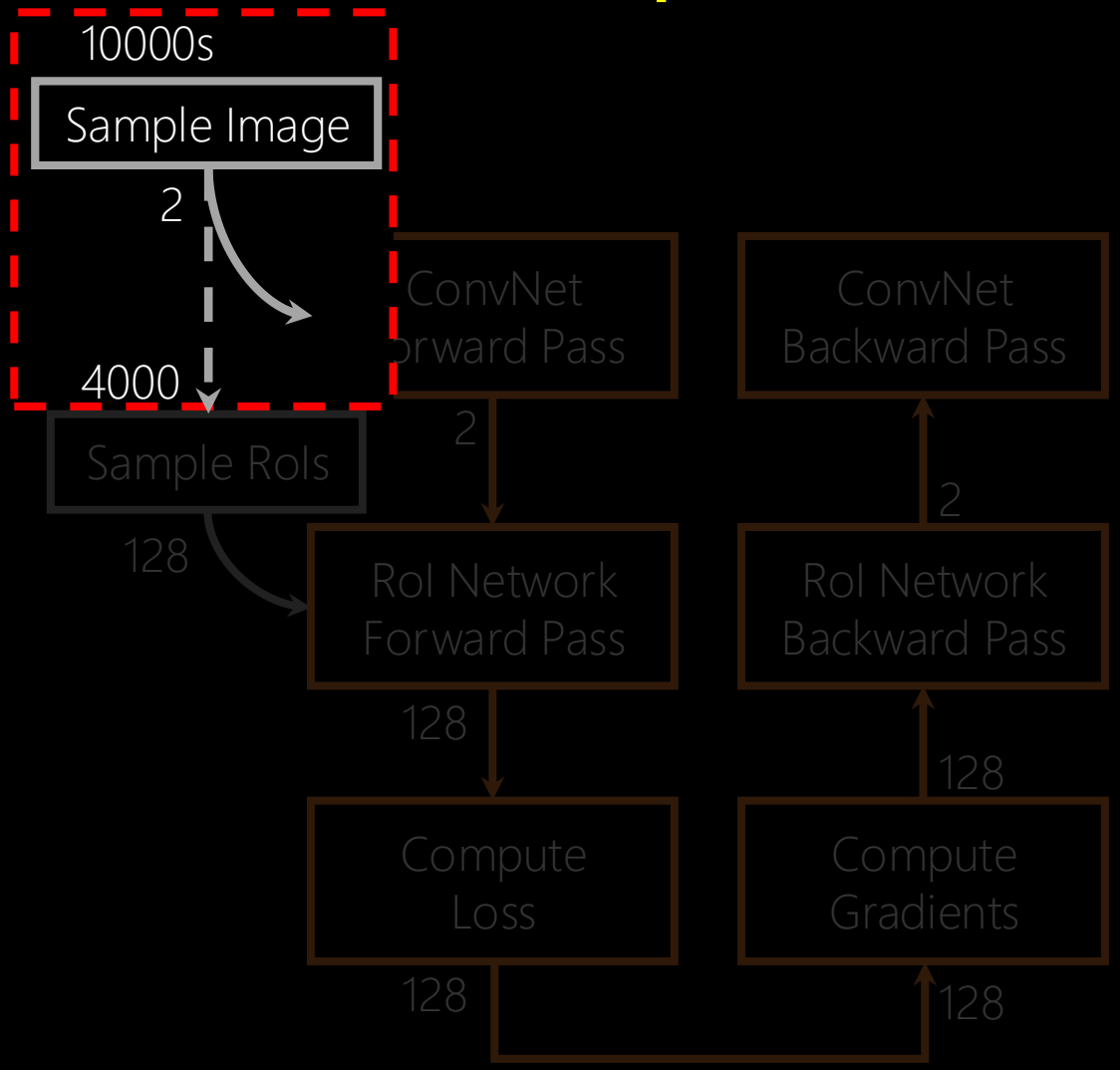
3. Redo





Fast R-CNN with Online Hard Example Mining



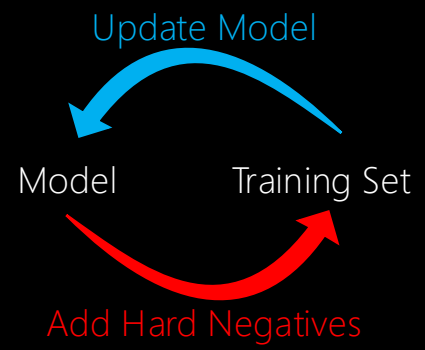


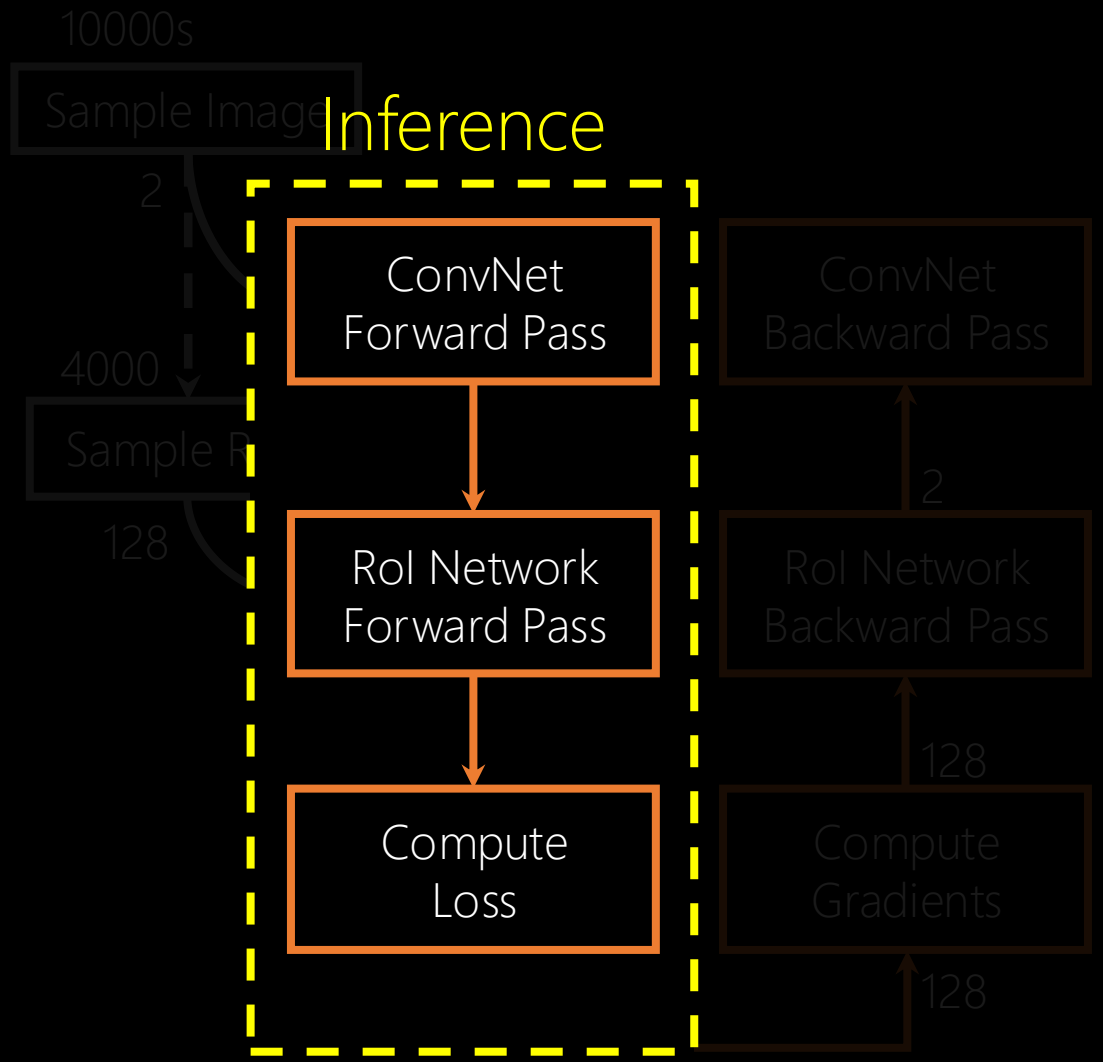
Fast R-CNN with Online Hard Example Mining

1. Fix Training Set
   Update Model

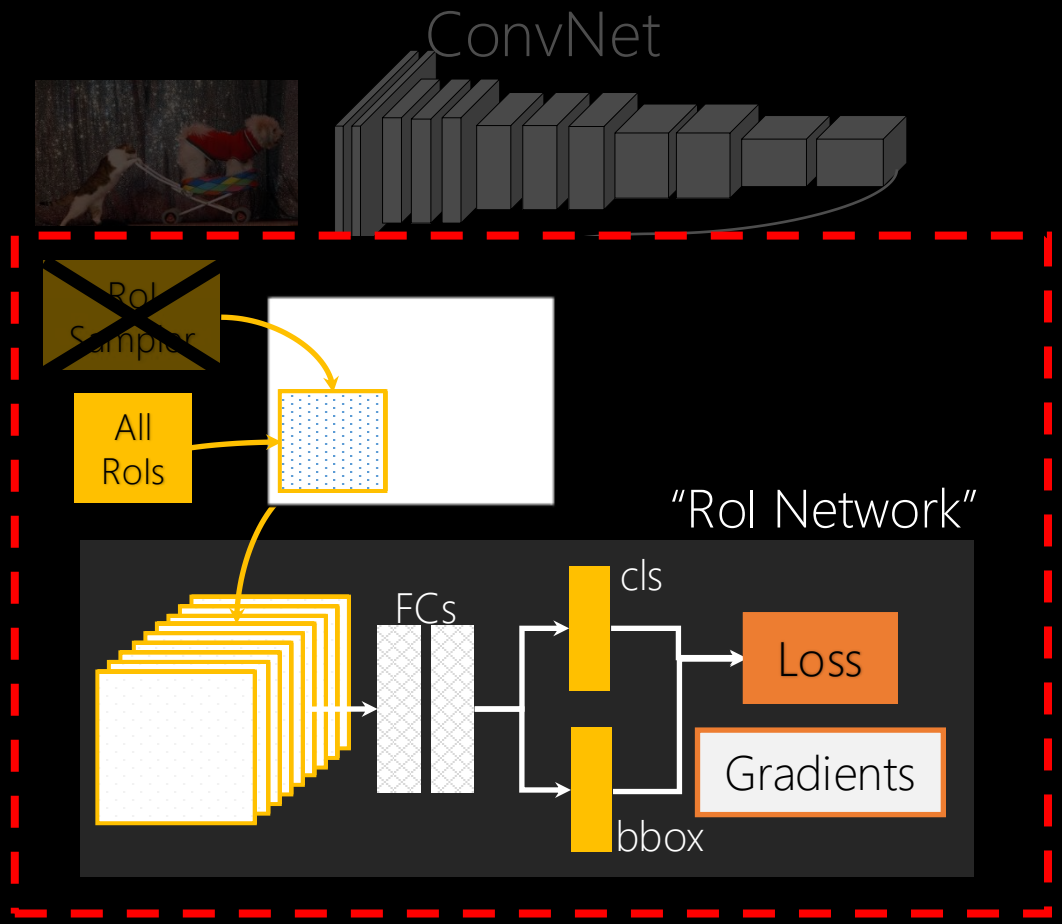
2. Freeze Model
   Add Hard Negatives

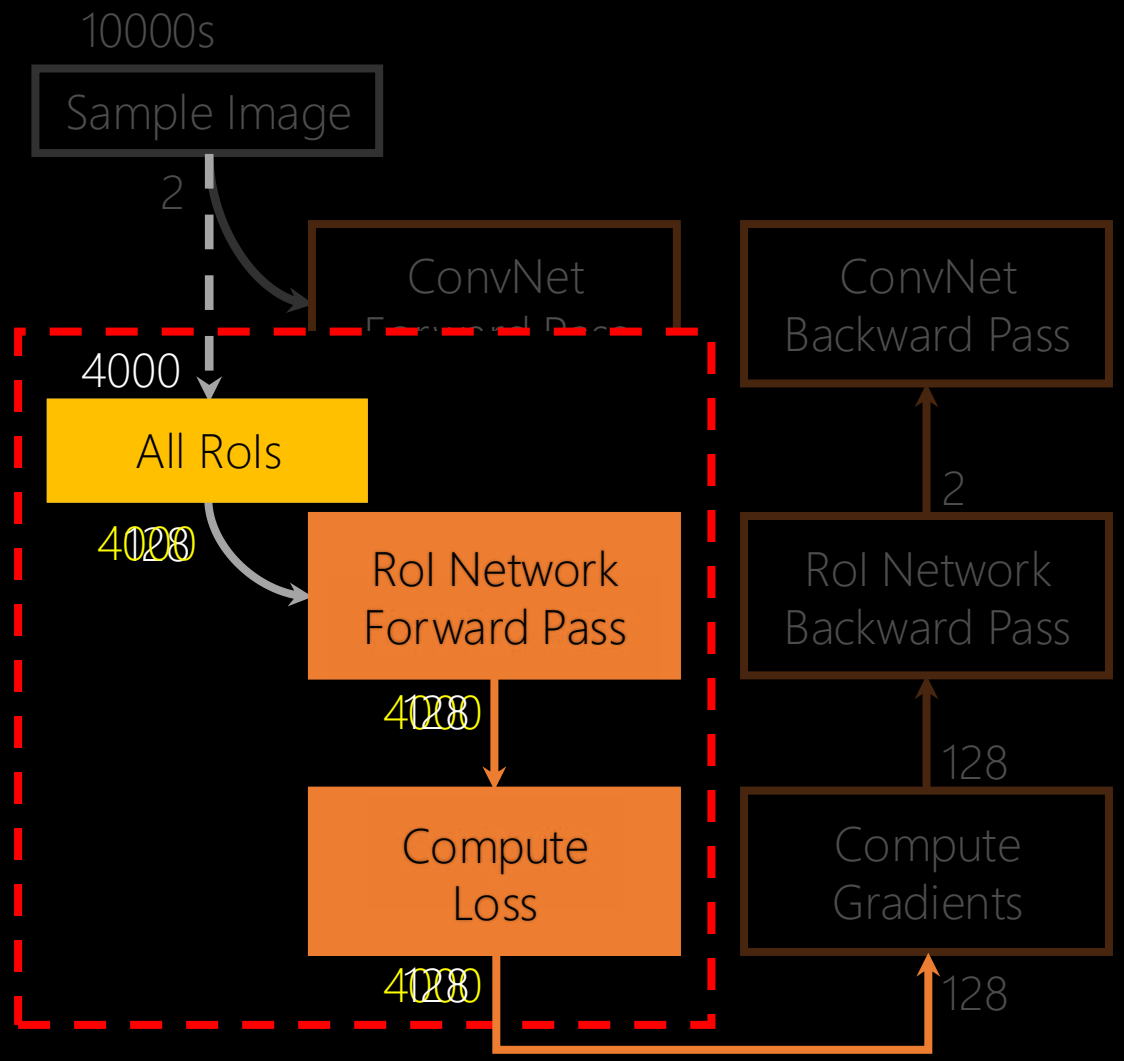
3. Redo



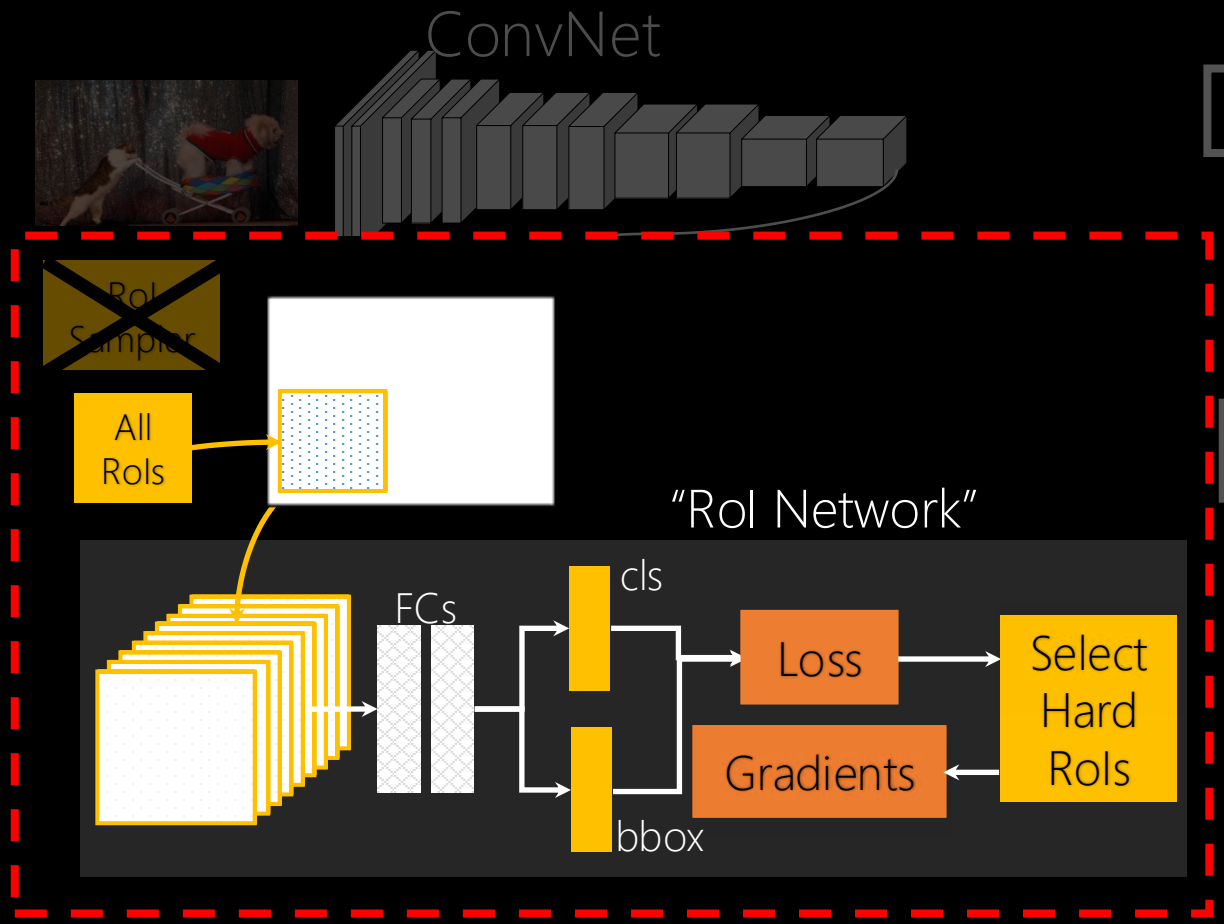


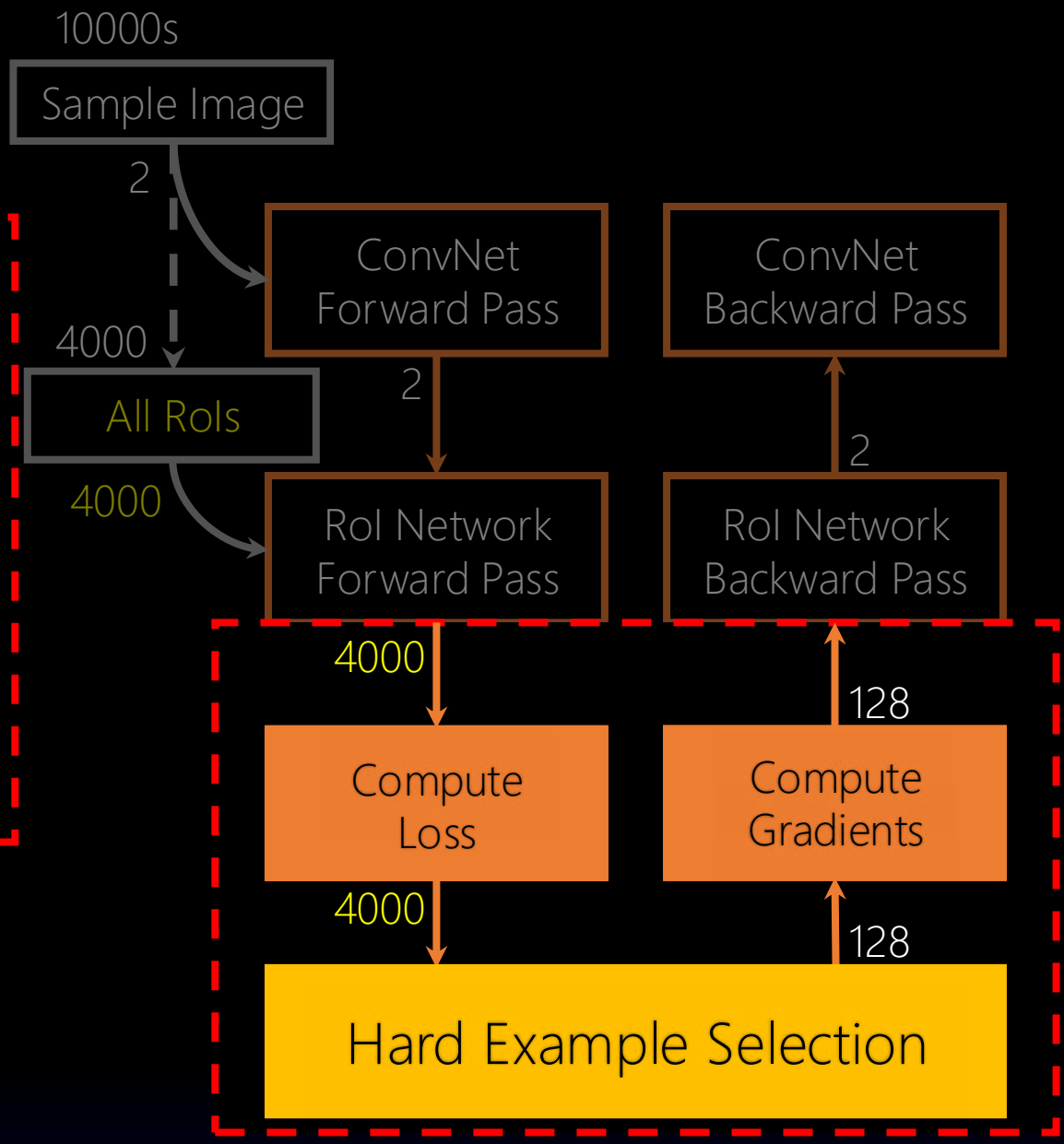
Fast R-CNN with Online Hard Example Mining



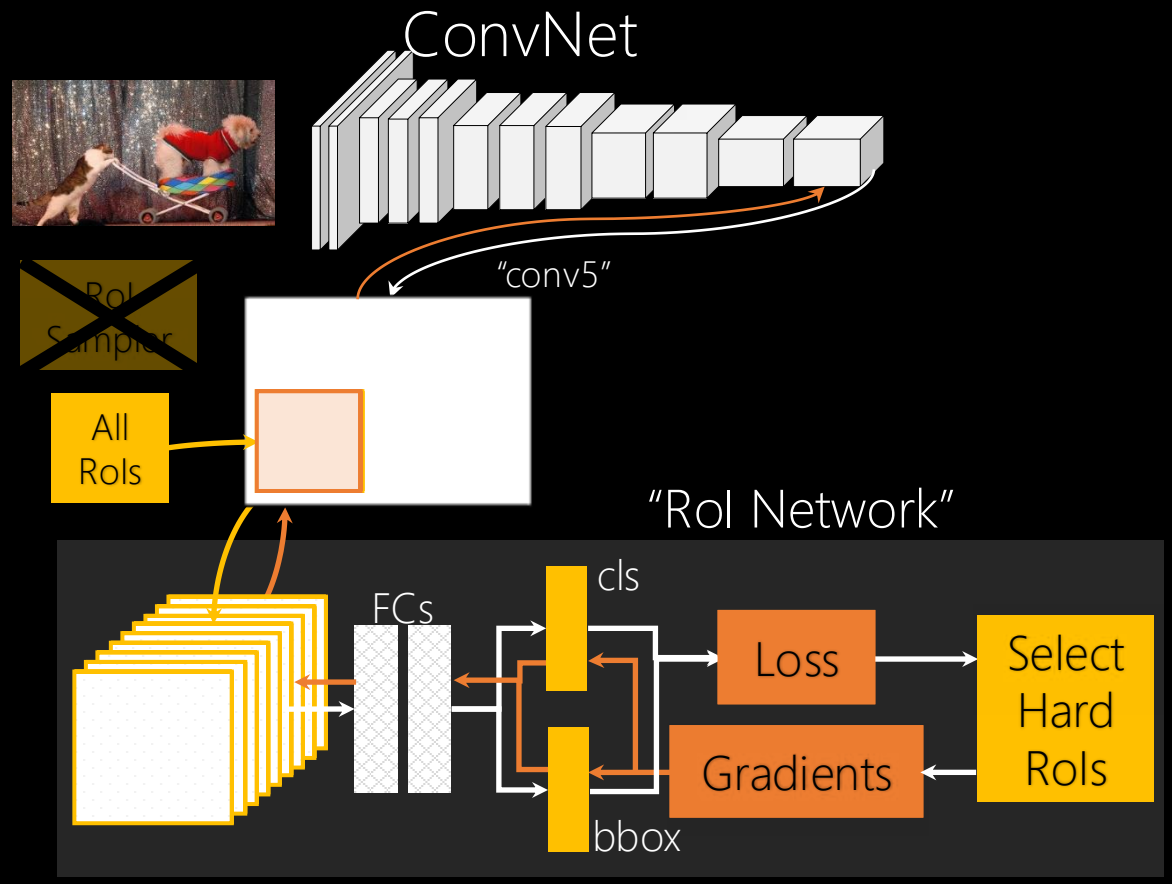


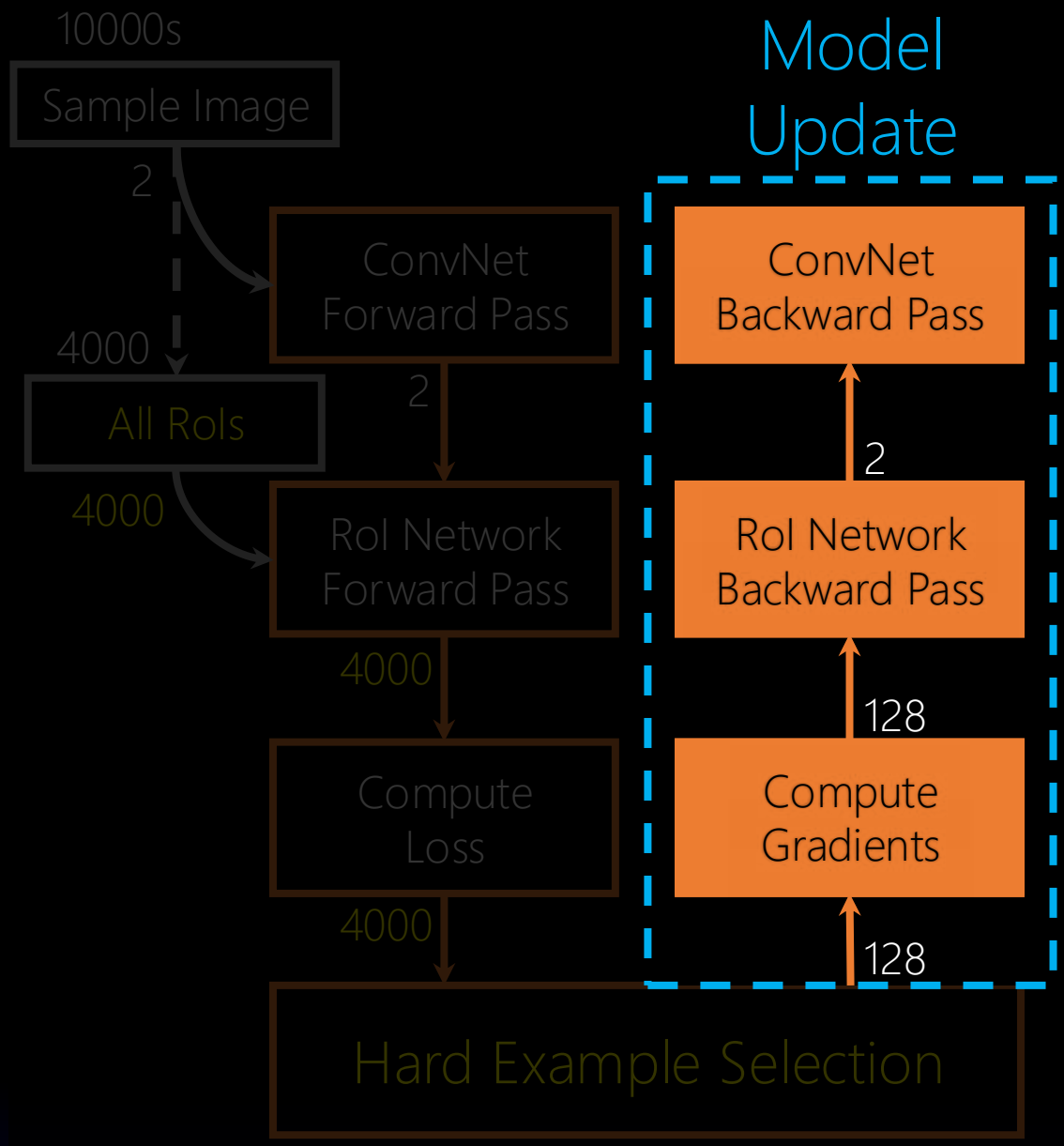
Fast R-CNN with Online Hard Example Mining



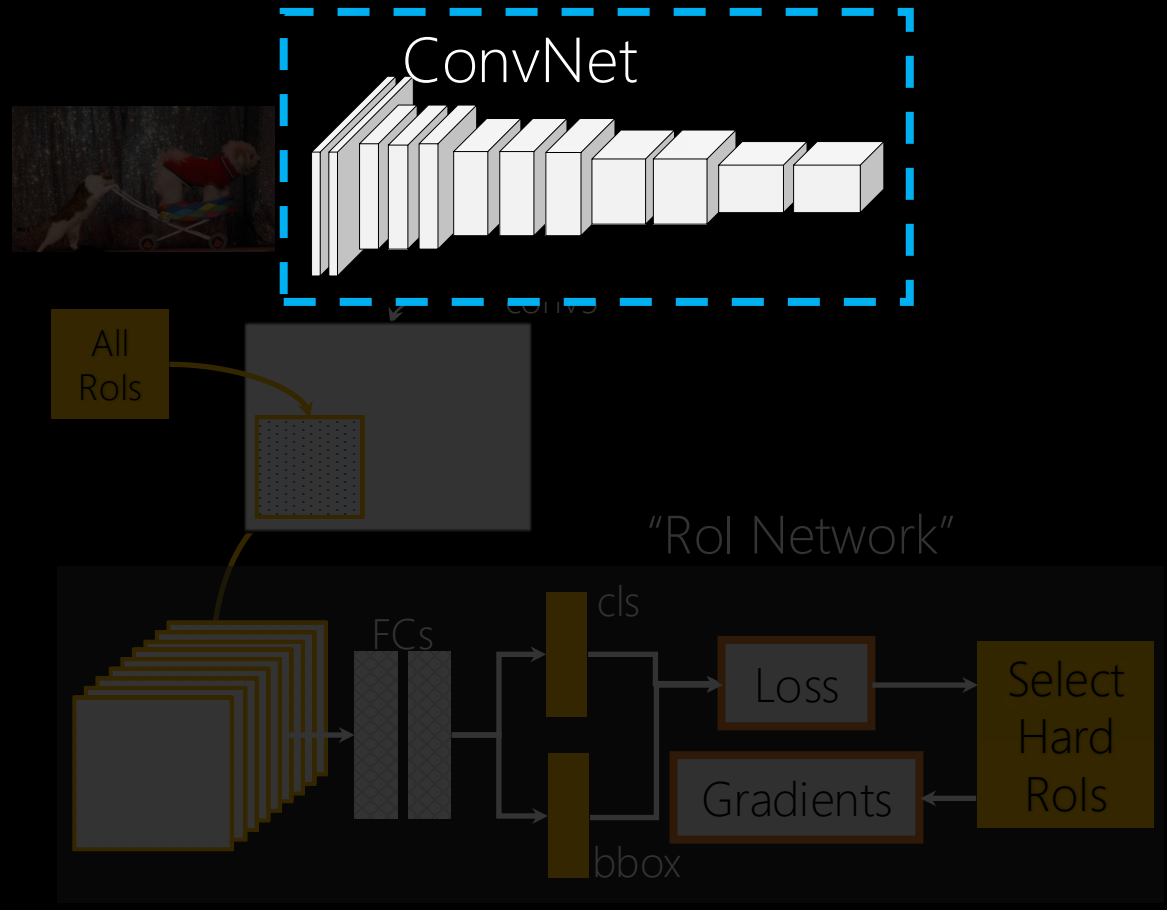


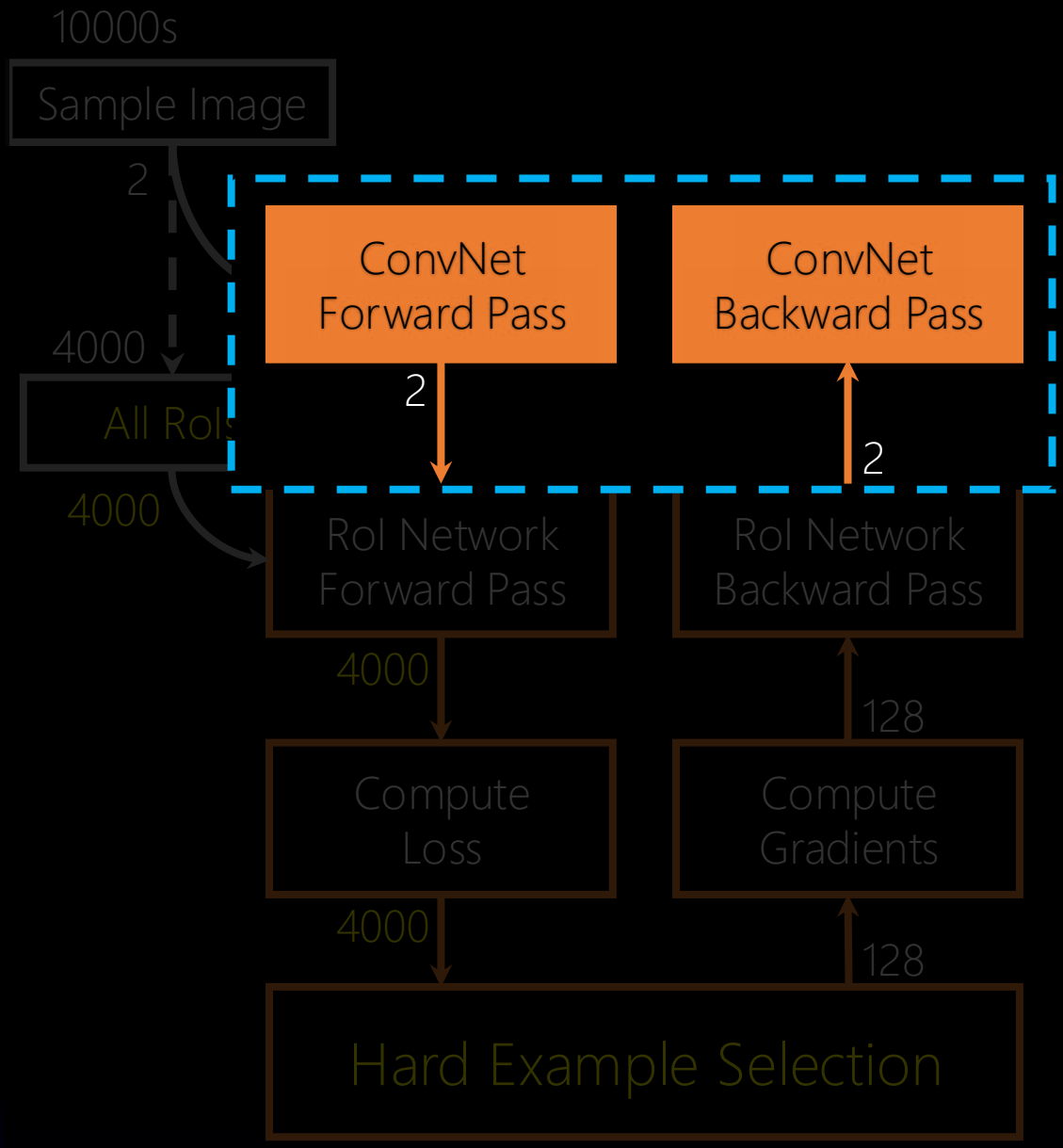
Fast R-CNN with Online Hard Example Mining



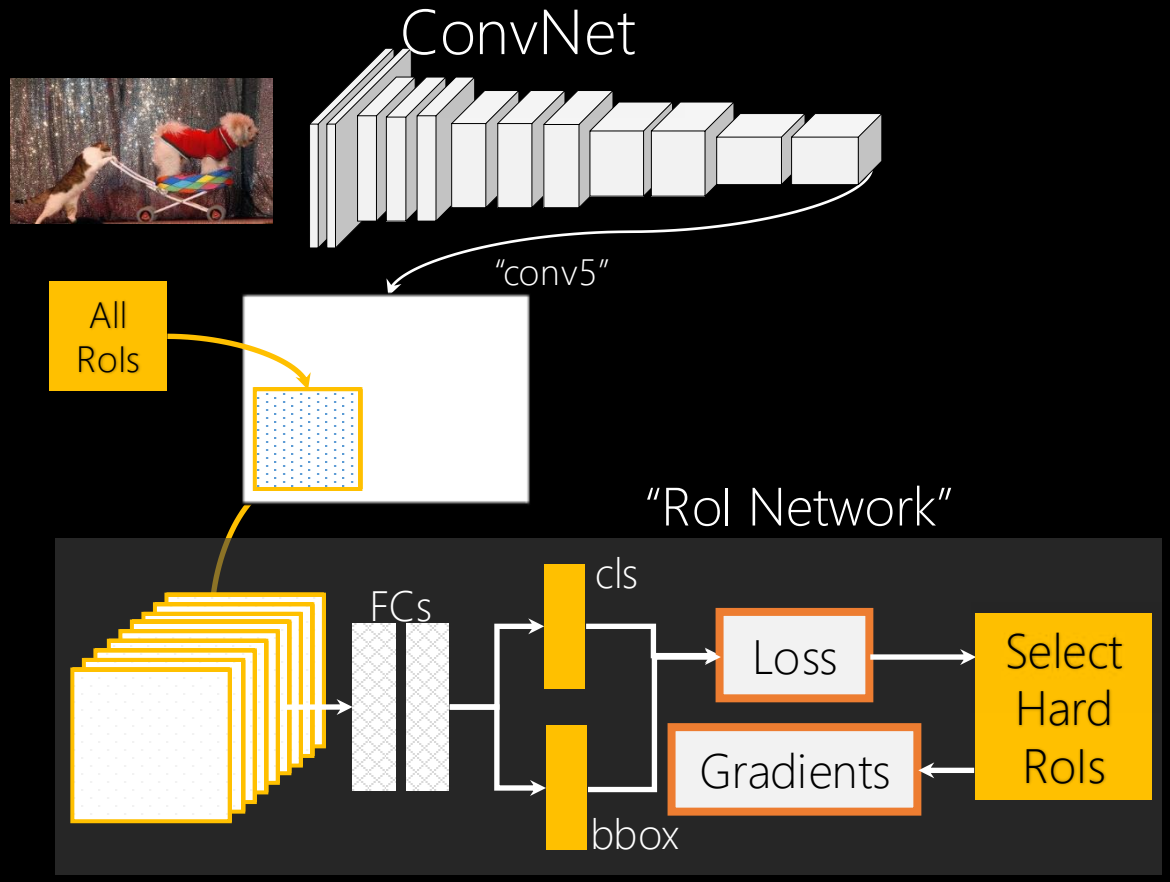


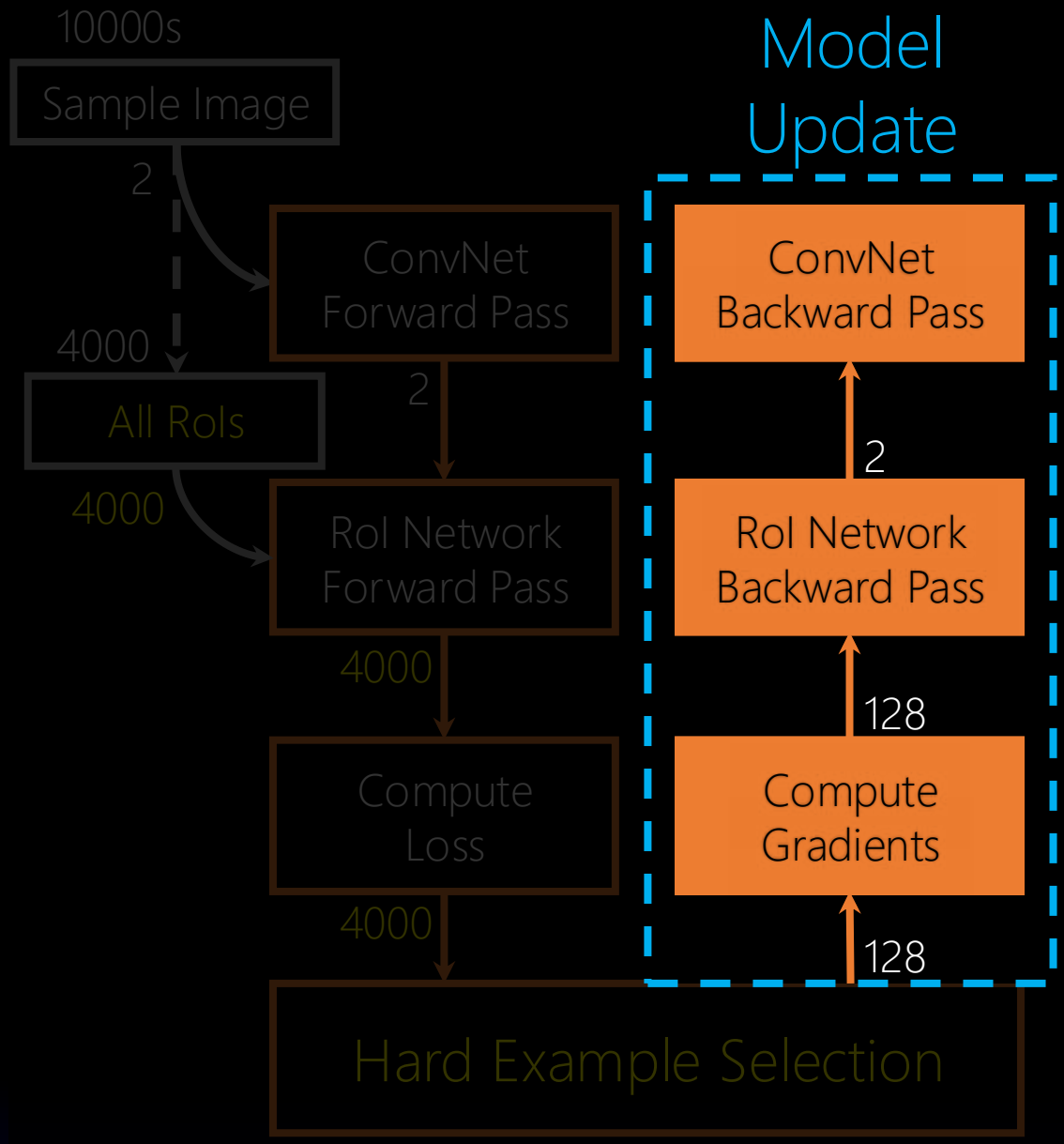
Fast R-CNN with Online Hard Example Mining



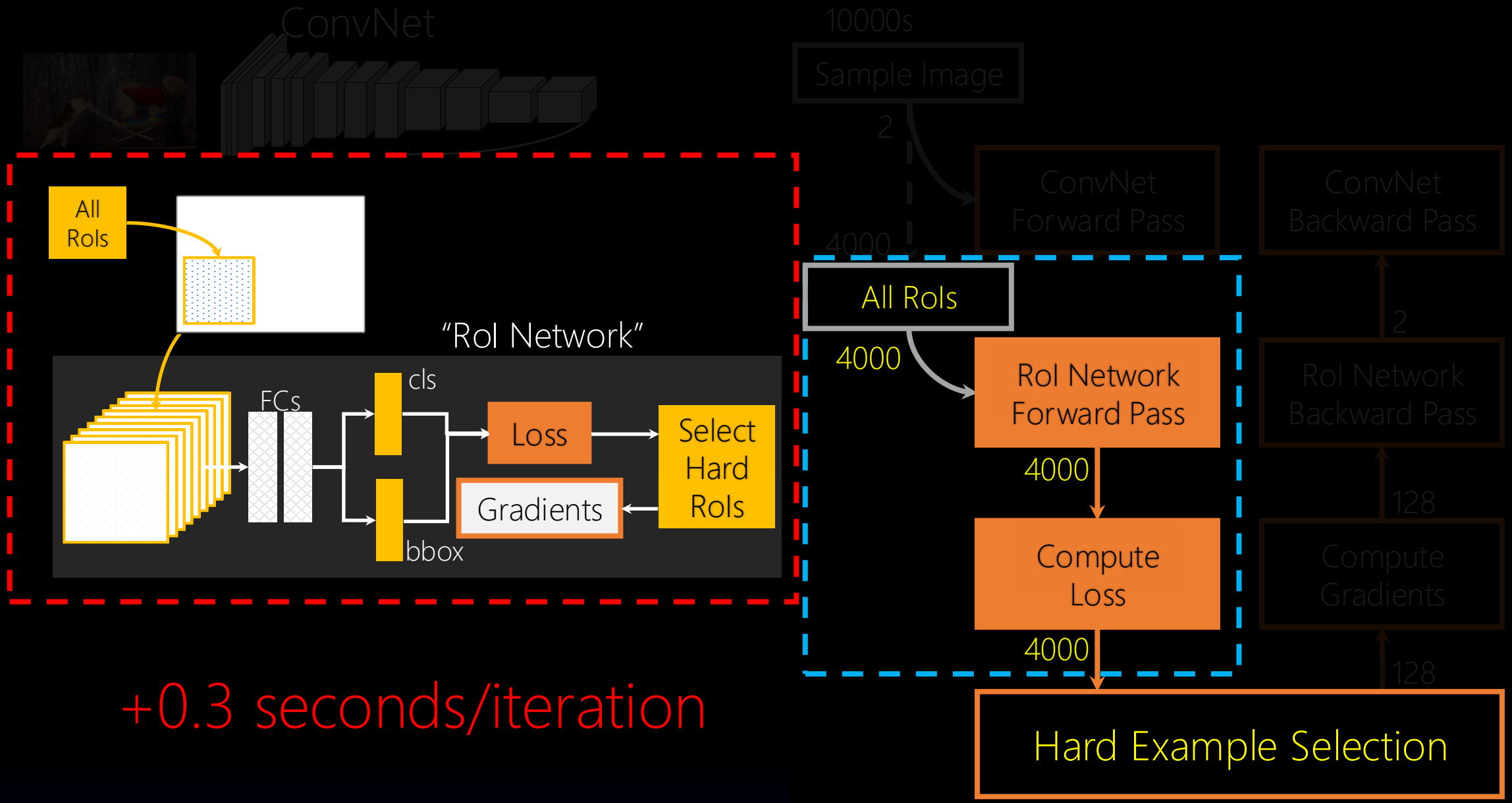


Fast R-CNN with Online Hard Example Mining





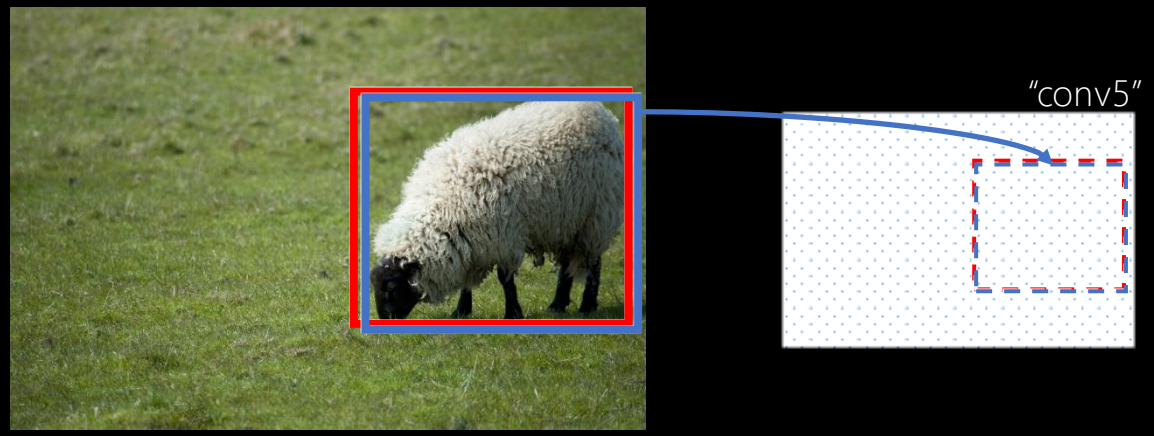
Fast R-CNN with Online Hard Example Mining



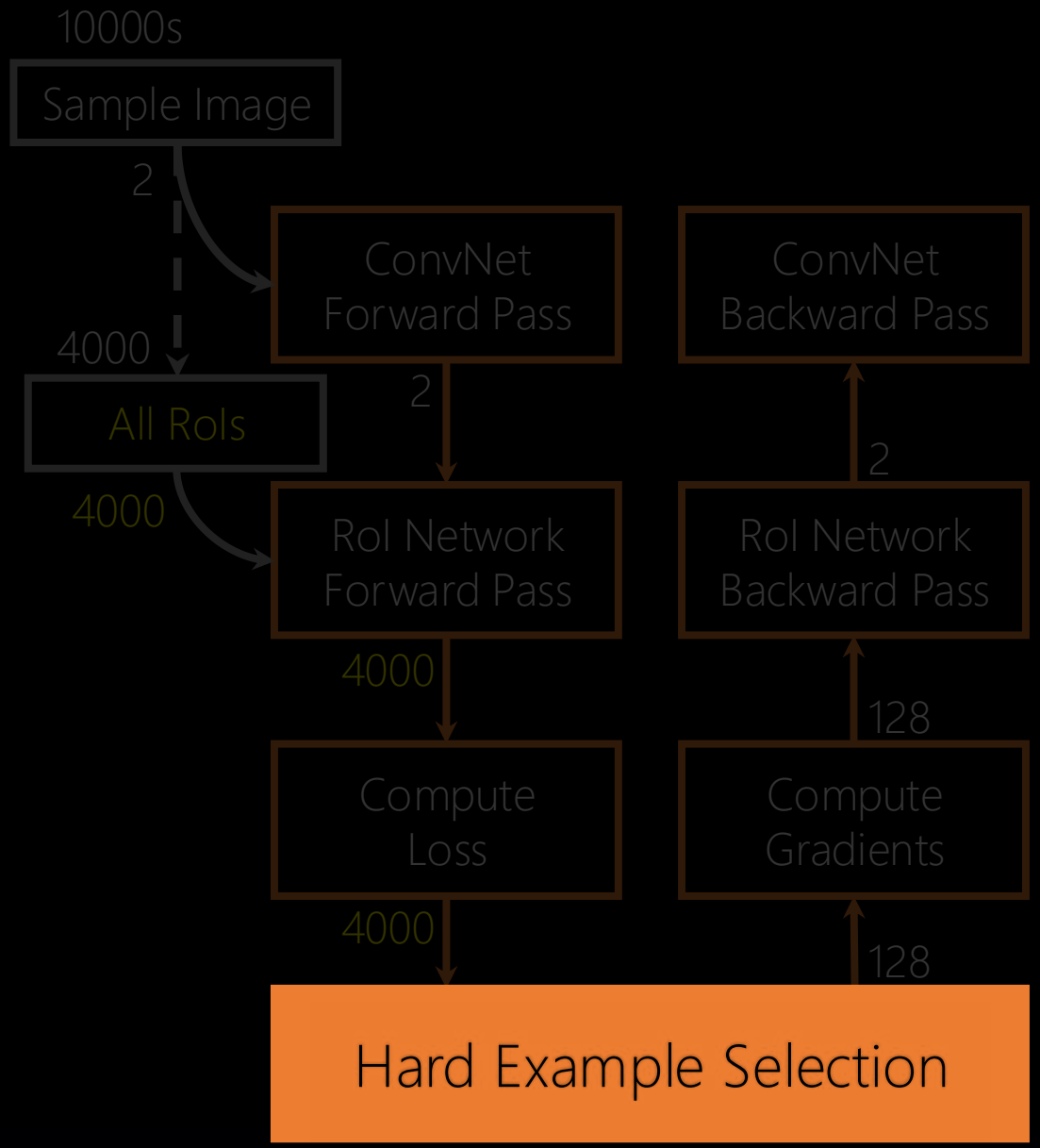
Fast R-CNN with Online Hard Example Mining

Hard Example Selection:

1. Sort RoIs based on Loss
2. Select top 128 RoIs



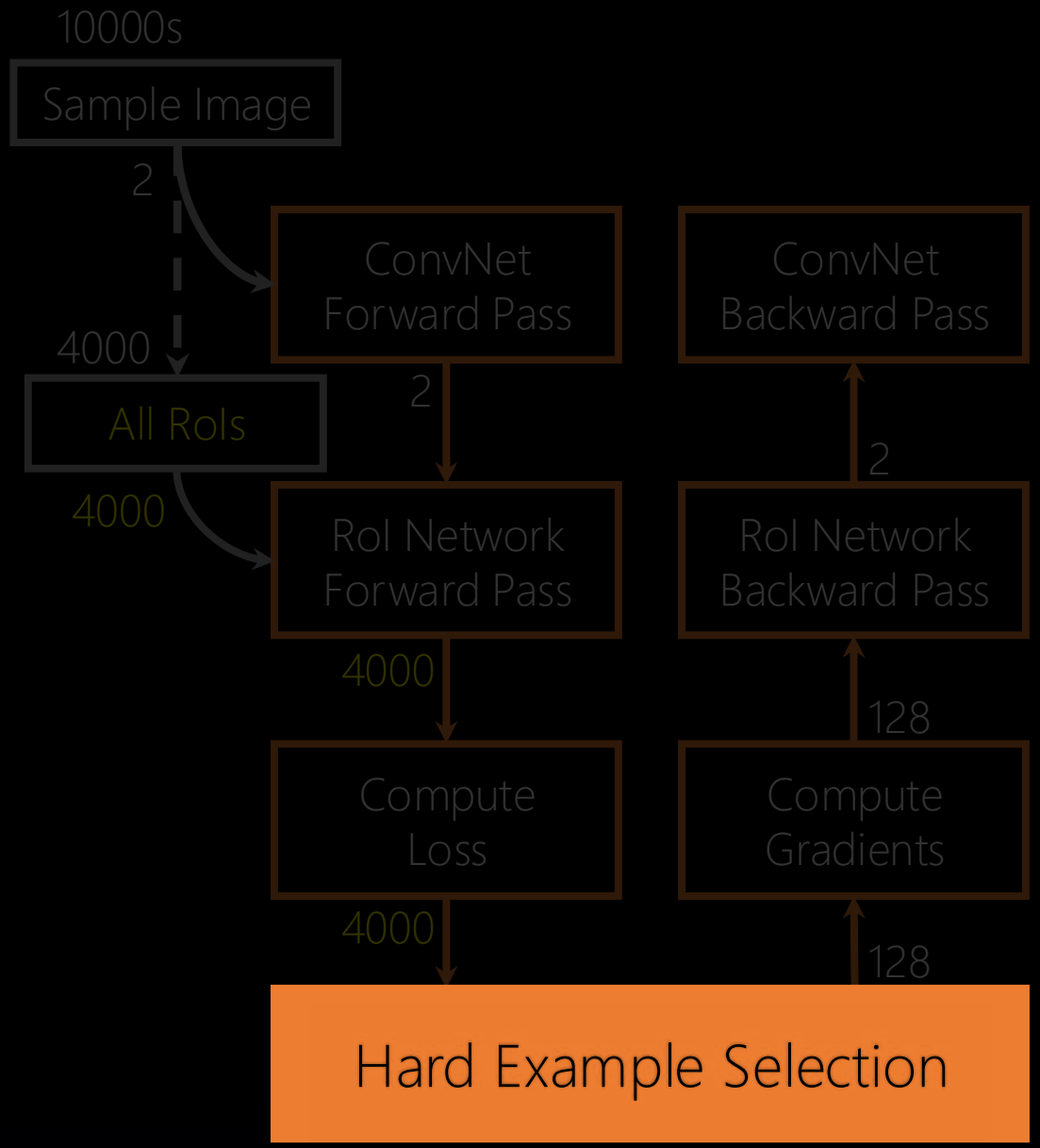
- Co-located RoIs = Co-related Loss
- Resolution disparity = Loss double counting



Fast R-CNN with Online Hard Example Mining

Hard Example Selection:

1. Sort RoIs based on Loss
2. Do NMS for de-duplication
3. Select top 128 RoIs



Main Results

Pascal VOC & MS COCO

Main Results: Pascal VOC

Test	Train	Fast R-CNN	Fast R-CNN W/ OHEM
07 mAP			
12 mAP			

Fast R-CNN [Girshick et al., 2015], Pascal VOC [Everingham et al., 2010].
All methods use VGG16 [Simonyan & Zisserman, 2014]. See paper for training set key.

Main Results: Pascal VOC

Test	Train	Fast R-CNN	Fast R-CNN W/ OHEM	
07 mAP	07	66.9	69.9	+3.0
12 mAP	12	65.7	69.8	+4.1

Fast R-CNN [Girshick et al., 2015], Pascal VOC [Everingham et al., 2010].
All methods use VGG16 [Simonyan & Zisserman, 2014]. See paper for training set key.

Main Results: Pascal VOC

Test	Train	Fast R-CNN	Fast R-CNN W/ OHEM	
07 mAP	07	66.9	69.9	+3.0
	07+12	70.0	74.6	+4.6
12 mAP	12	65.7	69.8	+4.1
	07++12	68.4	71.9	+3.5

Fast R-CNN [Girshick et al., 2015], Pascal VOC [Everingham et al., 2010].
All methods use VGG16 [Simonyan & Zisserman, 2014]. See paper for training set key.

Main Results: MS COCO

AP @ IoU	area	Fast R-CNN	Fast R-CNN W/ OHEM
[0.50 : 0.95]	all		
0.50	all		
0.75	all		
[0.50 : 0.95]	small		
[0.50 : 0.95]	med.		
[0.50 : 0.95]	large		

Testing on "Test-dev" split. All methods use VGG16 [Simonyan & Zisserman, 2014].
Fast R-CNN [Girshick et al., 2015], MS COCO [Lin et al., 2010].

Main Results: MS COCO

AP @ IoU	area	Fast R-CNN	Fast R-CNN W/ OHEM	
[0.50 : 0.95]	all	19.7	22.6	+2.9
0.50	all	35.9	42.5	+6.6
0.75	all	19.9	22.2	+2.3
[0.50 : 0.95]	small	3.5	5.0	+1.5
[0.50 : 0.95]	med.	18.8	23.7	+4.9
[0.50 : 0.95]	large	34.6	37.9	+3.3

Testing on "Test-dev" split. All methods use VGG16 [Simonyan & Zisserman, 2014].
Fast R-CNN [Girshick et al., 2015], MS COCO [Lin et al., 2010].

Main Results: MS COCO

AP @ IoU	area	Fast R-CNN	Fast R-CNN W/ OHEM	
[0.50 : 0.95]	all	19.7	22.6	+2.9
0.50	all	35.9	42.5	+6.6
0.75	all	19.9	22.2	+2.3
[0.50 : 0.95]	small	3.5	5.0	+1.5
[0.50 : 0.95]	med.	18.8	23.7	+4.9
[0.50 : 0.95]	large	34.6	37.9	+3.3

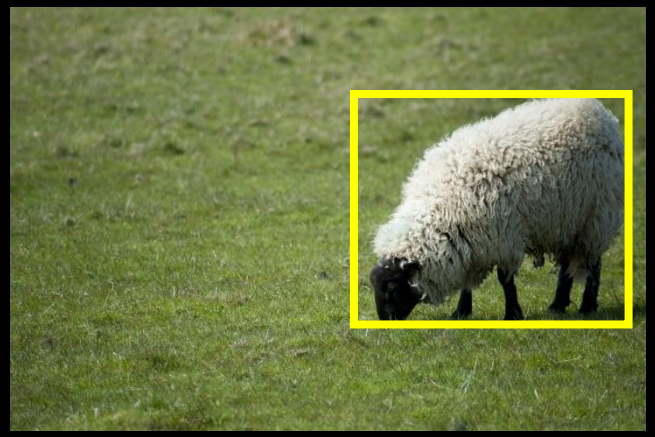
Testing on "Test-dev" split. All methods use VGG16 [Simonyan & Zisserman, 2014].
Fast R-CNN [Girshick et al., 2015], MS COCO [Lin et al., 2010].

Other Alternatives?

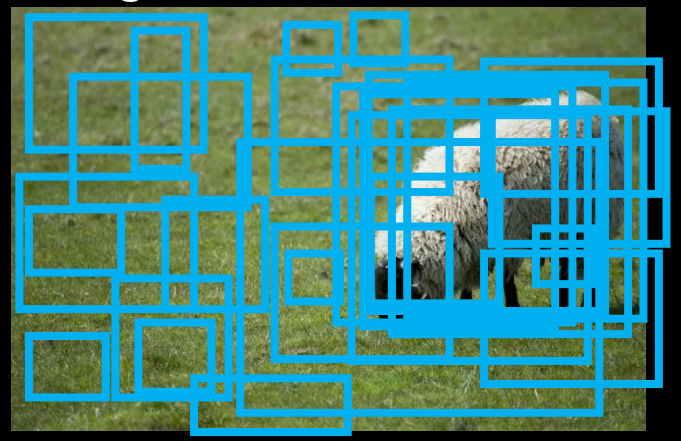
Heuristics for Hard Mining

OHEM vs. Heuristic Sampling

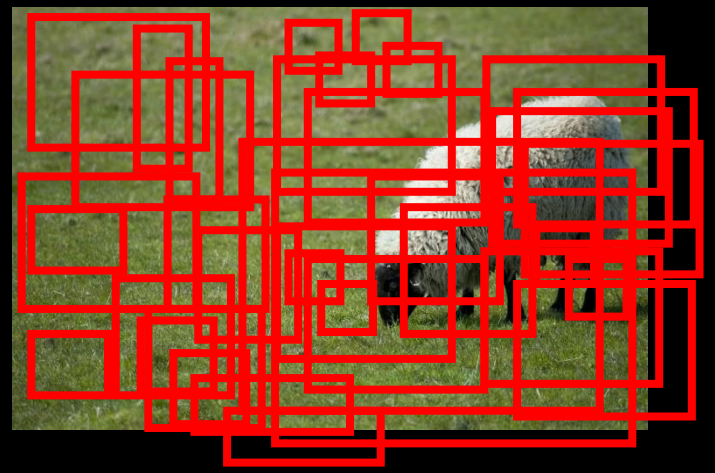
Ground-truth (GT)



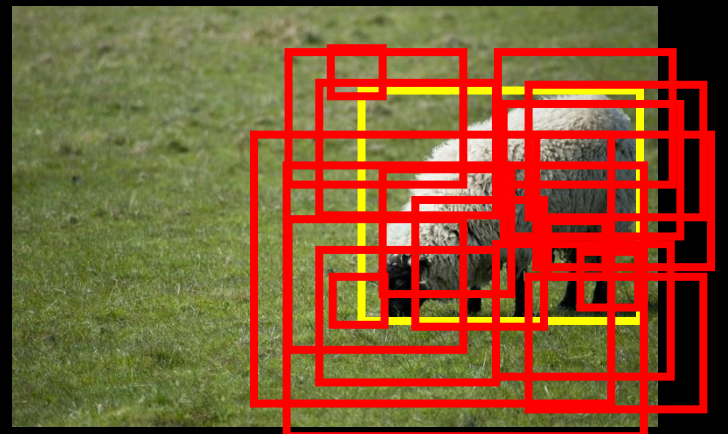
Regions of Interest (RoIs)



Background



Background with GT overlap



Heuristic Hard Negative Mining

Only considers bg regions that overlap with the GT

SPPNet [He et al., 2014],
Fast R-CNN [Girshick, 2015]

OHEM vs. Heuristic Sampling

	VGG_M	VGG16
Random	57.2	67.5
Heuristic [1, 2]		
OHEM		

[1] SPP-Net [He et al., 2014]
[2] Fast R-CNN [Girshick, 2015]
Testing on VOC 07 Test.
All Systems use Fast R-CNN. VGG_M and VGG16 from [Simonyan & Zisserman, 2014].

OHEM vs. Heuristic Sampling

	VGG_M		VGG16	
Random	57.2		67.5	
Heuristic [1, 2]	59.6	+2.4	67.2	-0.3
OHEM				

[1] SPP-Net [He et al., 2014]
[2] Fast R-CNN [Girshick, 2015]
Testing on VOC 07 Test.
All Systems use Fast R-CNN. VGG_M and VGG16 from [Simonyan & Zisserman, 2014].

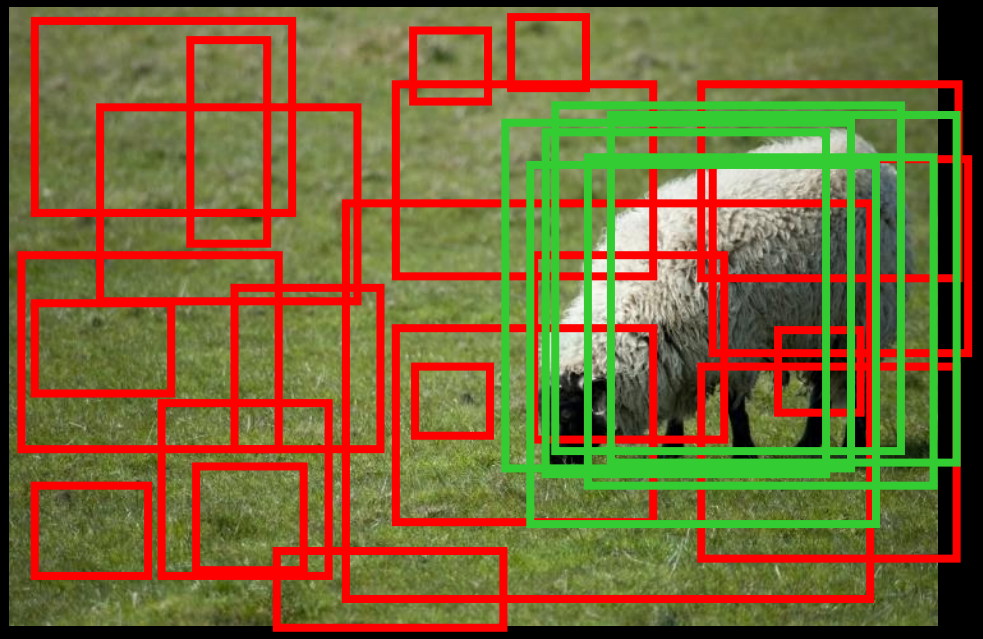
OHEM vs. Heuristic Sampling

	VGG_M		VGG16	
Random	57.2		67.5	
Heuristic [1, 2]	59.6	+2.4	67.2	-0.3
OHEM	62.0	+4.8	69.9	+2.4

[1] SPP-Net [He et al., 2014]
[2] Fast R-CNN [Girshick, 2015]
Testing on VOC 07 Test.
All Systems use Fast R-CNN. VGG_M and VGG16 from [Simonyan & Zisserman, 2014].

Why not use all Regions?

All RoIs Mini-batch



□ fg □ bg

Why not use all Regions?

Batch Size	Learning Rate	Fast R-CNN	Fast R-CNN W/ OHEM
128	0.001	67.2	69.9
2048			

Testing on VOC 07 Test.
Fast R-CNN [Girshick, 2015]. All methods use VGG16 [Simonyan & Zisserman, 2014].

Why not use all Regions?

Batch Size	Learning Rate	Fast R-CNN	Fast R-CNN W/ OHEM
128	0.001	67.2	69.9
2048	0.003		

Testing on VOC 07 Test.
Fast R-CNN [Girshick, 2015]. All methods use VGG16 [Simonyan & Zisserman, 2014].

Why not use all Regions?

Batch Size	Learning Rate	Fast R-CNN	Fast R-CNN W/ OHEM
128	0.001	67.2	69.9
2048	0.003	68.7	
		+1.5	

Testing on VOC 07 Test.
Fast R-CNN [Girshick, 2015]. All methods use VGG16 [Simonyan & Zisserman, 2014].

Why not use all Regions?

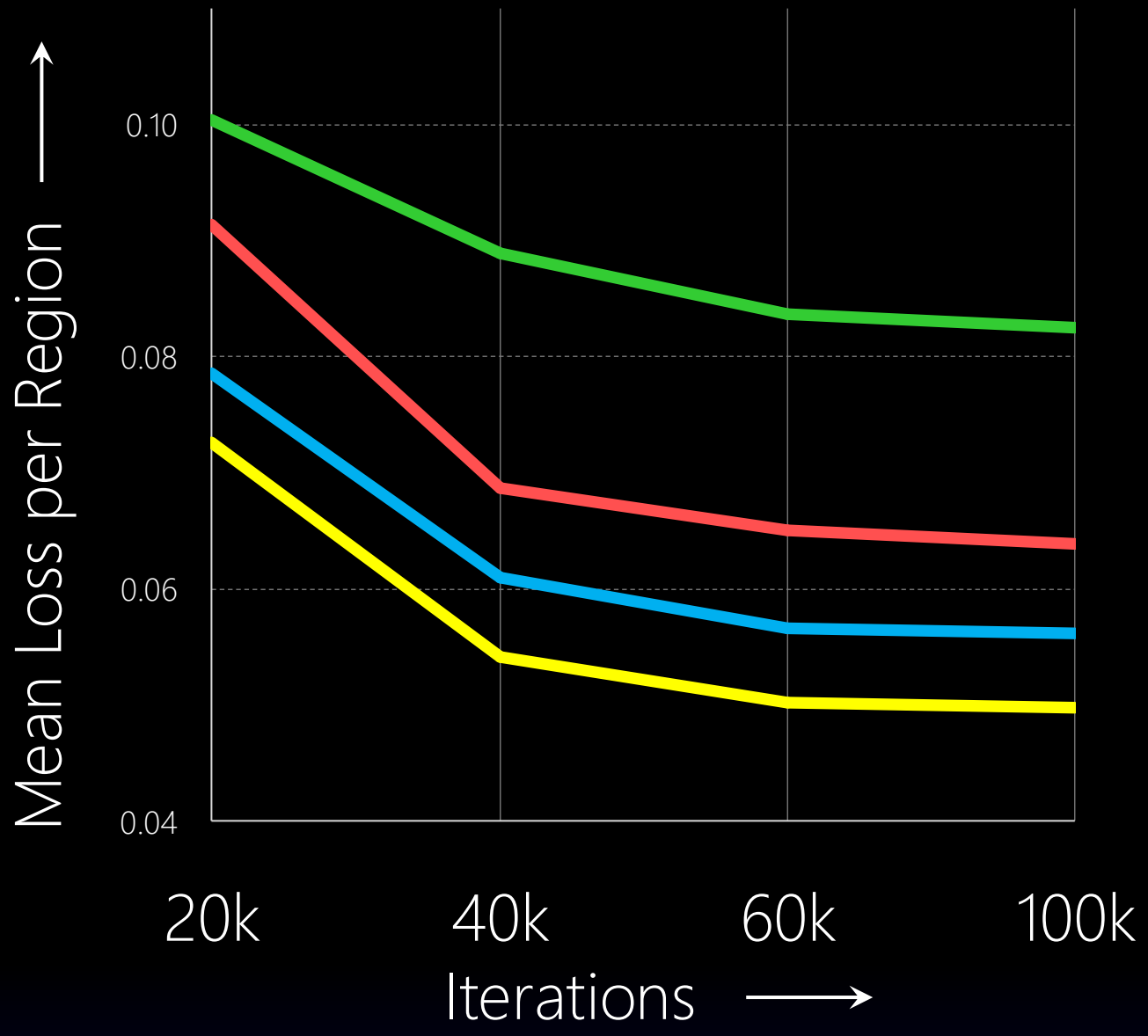
Batch Size	Learning Rate	Fast R-CNN	Fast R-CNN W/ OHEM
128	0.001	67.2	69.9
2048	0.003	68.7	
		+1.5	+1.2

Testing on VOC 07 Test.
Fast R-CNN [Girshick, 2015]. All methods use VGG16 [Simonyan & Zisserman, 2014].

Better Optimization

Compared to alternatives

Training Loss



Bells and Whistles

1. Multi-scale [1, 2]
2. Iterative Localization [3]

[1] SPP-Net [He et al., 2014]
[2] Fast R-CNN [Girshick, 2015]
[3] MR-CNN [Gidaris & Komodakis, 2015]

Fast R-CNN W/ OHEM

Test	Train		+Multi-scale [1, 2] +Iterative Loc. [3]
VOC 07 mAP	07	69.9	
	07+12	74.6	
VOC 12 mAP	12	69.8	
	07++12	71.9	

[1] SPP-Net [He et al., 2014], [2] Fast R-CNN [Girshick, 2015], [3] MR-CNN [Gidaris & Komodakis, 2015]

Fast R-CNN W/ OHEM

Test	Train		+Multi-scale [1, 2] +Iterative Loc. [3]	
VOC 07 mAP	07	69.9	75.1	+5.2
	07+12	74.6	78.9	+4.3
VOC 12 mAP	12	69.8	72.9	+3.1
	07++12	71.9	76.3	+4.4

[1] SPP-Net [He et al., 2014], [2] Fast R-CNN [Girshick, 2015], [3] MR-CNN [Gidaris & Komodakis, 2015]

Fast R-CNN W/ OHEM

Test	Train		+Multi-scale [1, 2] +Iterative Loc. [3]	
VOC 07 mAP	07	69.9	75.1	+5.2
	07+12	74.6	78.9	+4.3
VOC 12 mAP	12	69.8	72.9	+3.1
	07++12	71.9	76.3	+4.4
	+COCO		80.1	

[1] SPP-Net [He et al., 2014], [2] Fast R-CNN [Girshick, 2015], [3] MR-CNN [Gidaris & Komodakis, 2015]

Other Detectors and Networks

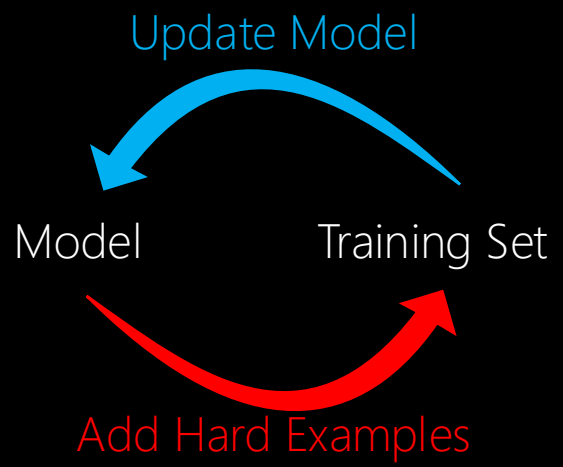
Recent works employing OHEM

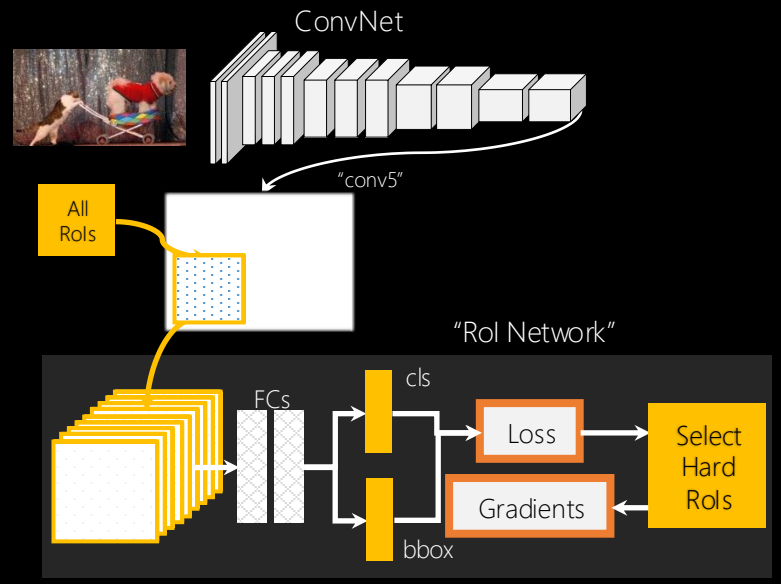
	07 mAP
Faster R-CNN	76.4
Faster R-CNN w/ OHEM	79.3
	+2.9
R-FCN	76.6
R-FCN w/ OHEM	79.5
	+2.9

Both methods use ResNet-101 [He et al., 2016]

Thanks to:
R-FCN: Object Detection via Region-based Fully Convolutional Networks
Jifeng Dai, Yi Li, Kaiming He, Jian Sun
arxiv:1605.06409, 2016

OHEM: Online Hard Example Mining





Test	Train	Fast R-CNN	Fast R-CNN W/ OHEM	
07 mAP	07	66.9	69.9	+3.0
	07+12	70.0	74.6	+4.6
12 mAP	12	65.7	69.8	+4.1
	07++12	68.4	71.9	+3.5

AP @ IoU	area	Fast R-CNN	Fast R-CNN W/ OHEM	
[0.50 : 0.95]	all	19.7	22.6	+2.9
0.50	all	35.9	42.5	+6.6
0.75	all	19.9	22.2	+2.3
[0.50 : 0.95]	small	3.5	5.0	+1.5
[0.50 : 0.95]	med.	18.8	23.7	+4.9
[0.50 : 0.95]	large	34.6	37.9	+3.3