YOLO

You Only Look Once: Unified, Real-Time Object Detection

Joseph Redmon
University of Washington
pjreddie@cs.washington.edu

Santosh Divvala
Allen Institute for Artificial Intelligence
santoshd@allenai.org

Ross Girshick
Facebook AI Research
rbg@fb.com

Ali Farhadi
University of Washington
ali@cs.washington.edu

Abstract

We present YOLO, a new approach to object detection. Prior work on object detection repurposes classifiers to perform detection. Instead, we frame object detection as a regression problem to spatially separated bounding boxes and associated class probabilities. A single neural network predicts bounding boxes and class probabilities directly from full images in one evaluation. Since the whole detection pipeline is a single network, it can be optimized end-to-end directly on detection performance.

Our unified architecture is extremely fast. Our base YOLO model processes images in real-time at 45 frames per second. A smaller version of the network, Fast YOLO, processes an astounding 155 frames per second while still achieving double the mAP of other real-time detectors. Compared to state-of-the-art detection systems, YOLO makes more localization errors but is far less likely to predict false detections where nothing exists. Finally, YOLO learns very general representations of objects. It outperforms all other detection methods, including DPM and R-CNN, by a wide margin when generalizing from natural images to artwork on both the Picasso Dataset and the People-Art Dataset.

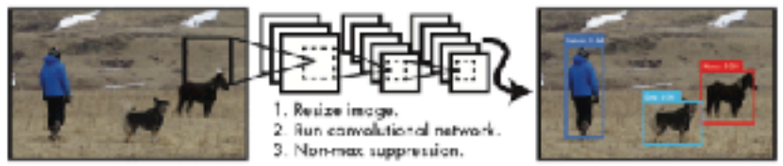


Figure 1: The YOLO Detection System. Processing images with YOLO is simple and straightforward. Our system (1) resizes the input image to 448 × 448, (2) runs a single convolutional network on the image, and (3) thresholds the resulting detections by the model's confidence.

classifier for that object and evaluate it at various locations and scales in a test image. Systems like deformable parts models (DPM) use a sliding window approach where the classifier is run at evenly spaced locations over the entire image [10].

More recent approaches like R-CNN use region proposal methods to first generate potential bounding boxes in an image and then run a classifier on these proposed boxes. After classification, post-processing is used to refine the bounding box, eliminate duplicate detections, and rescore the box based on other objects in the scene [13]. These complex

YOLO

(You Only Look Once)

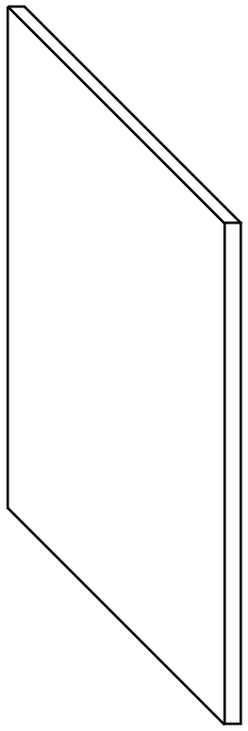
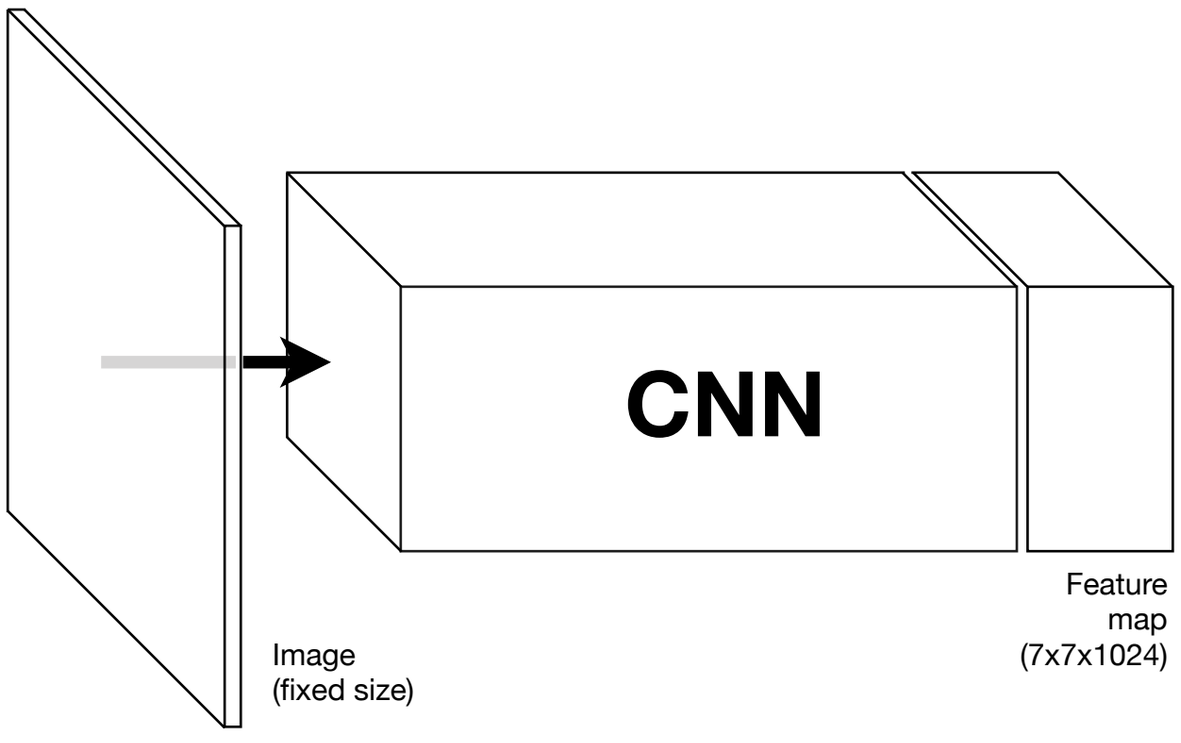
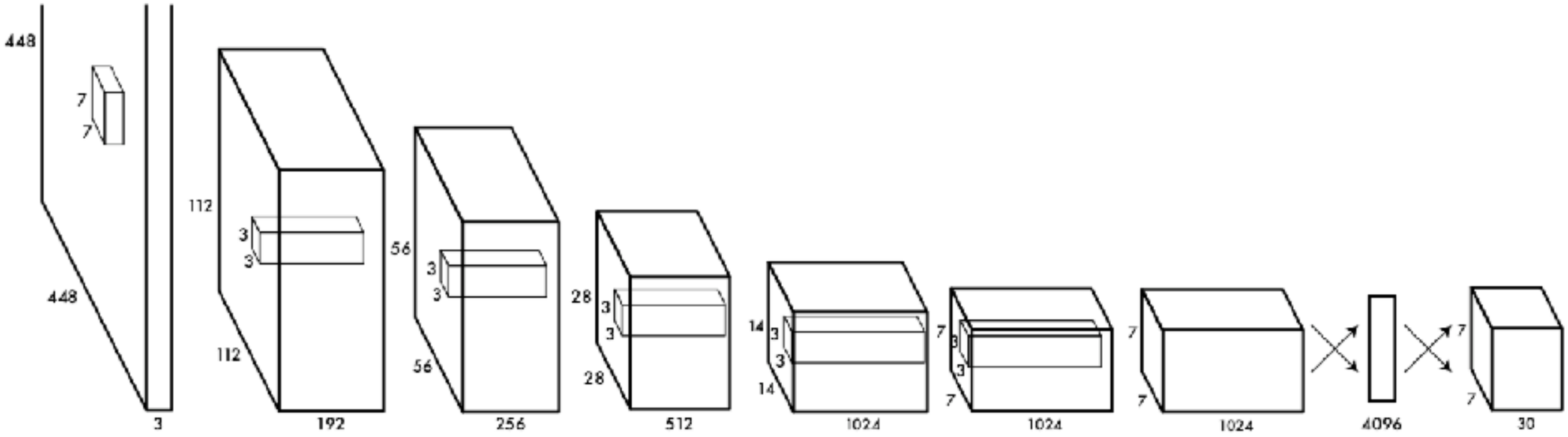


Image
(fixed size)

YOLO

(You Only Look Once)

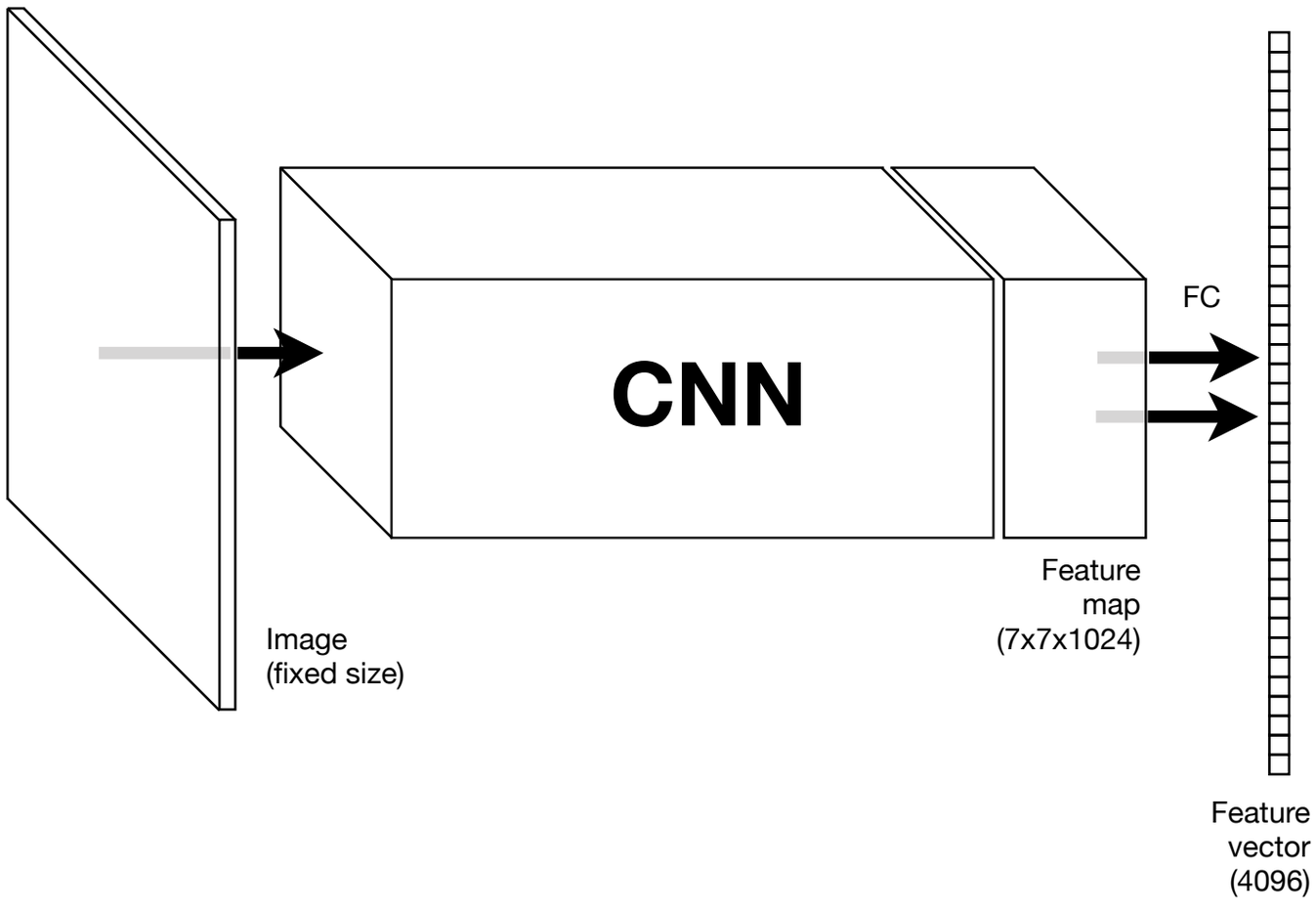




Conv. Layer	Conv. Layer	Conv. Layers	Conv. Layers	Conv. Layers	Conv. Layers	Conn. Layer	Conn. Layer
7x7x64-s-2	3x3x192	1x1x128	1x1x256 } ×4	1x1x512 } ×2	3x3x1024		
Maxpool Layer	Maxpool Layer	3x3x256	3x3x512	3x3x1024	3x3x1024		
2x2-s-2	2x2-s-2	1x1x256	1x1x512	3x3x1024			
		3x3x512	3x3x1024	3x3x1024-s-2			
		Maxpool Layer	Maxpool Layer				
		2x2-s-2	2x2-s-2				

YOLO

(You Only Look Once)

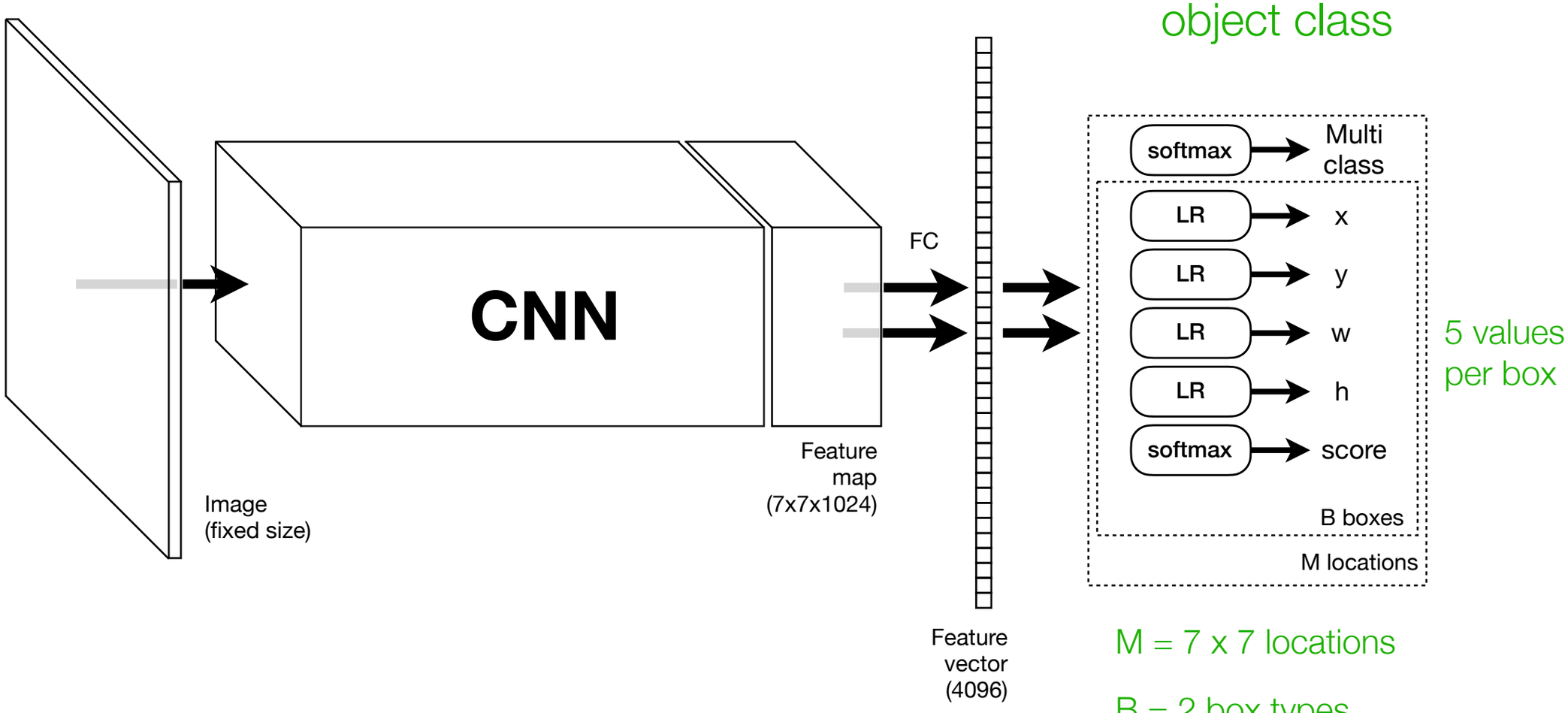


Expand to large
feature vector
instead of grid pooling

YOLO

(You Only Look Once)

Directly regress bounding boxes and object class



M = 7 x 7 locations

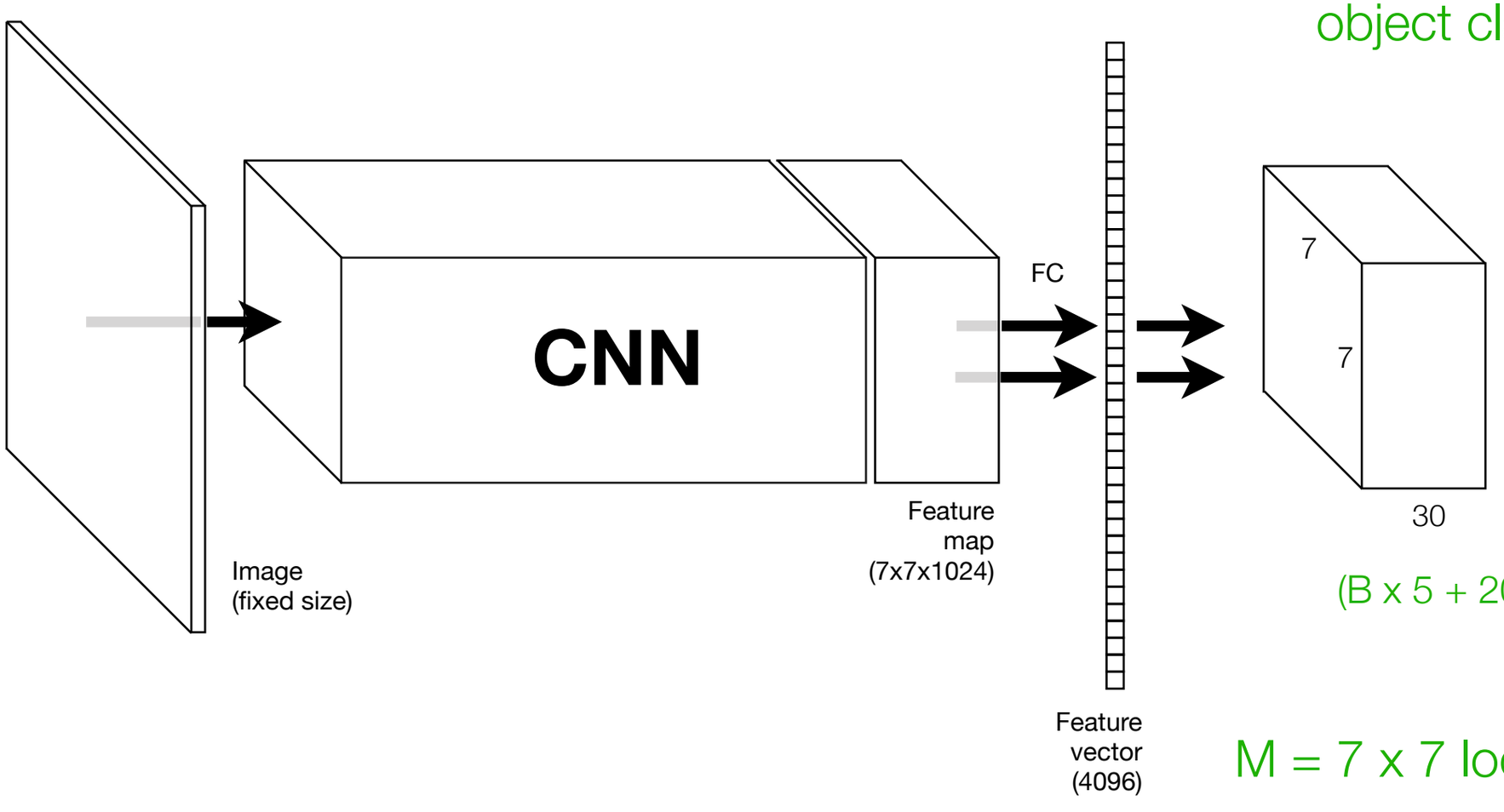
B = 2 box types

C = 20 classes

YOLO

(You Only Look Once)

Directly regress bounding boxes and object class



(B x 5 + 20)

M = 7 x 7 locations

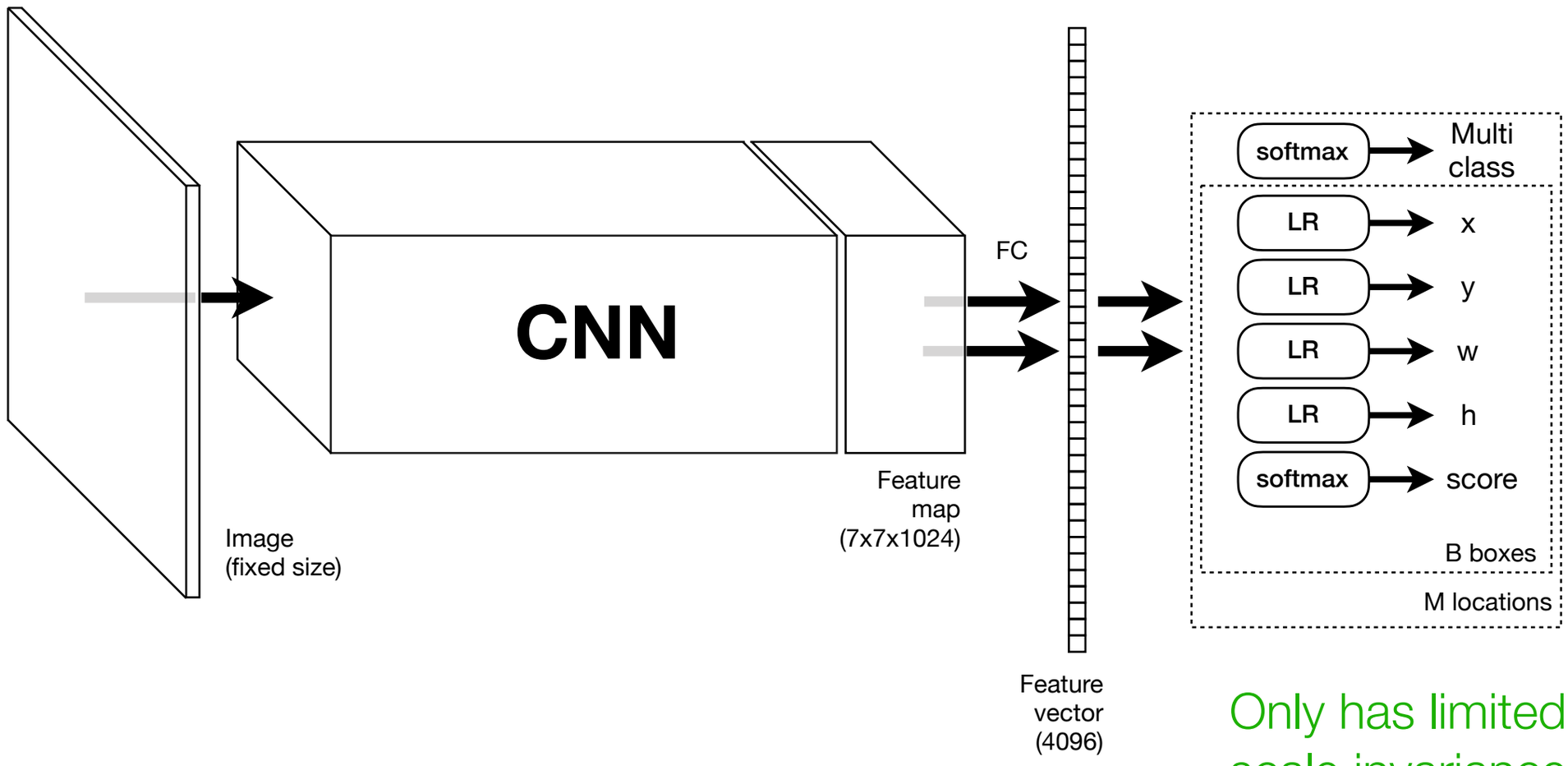
B = 2 box types

C = 20 classes


YOLO v2
http://pjreddie.com/yolo

YOLO

(You Only Look Once)



Only has limited scale invariance

(only two box sizes)

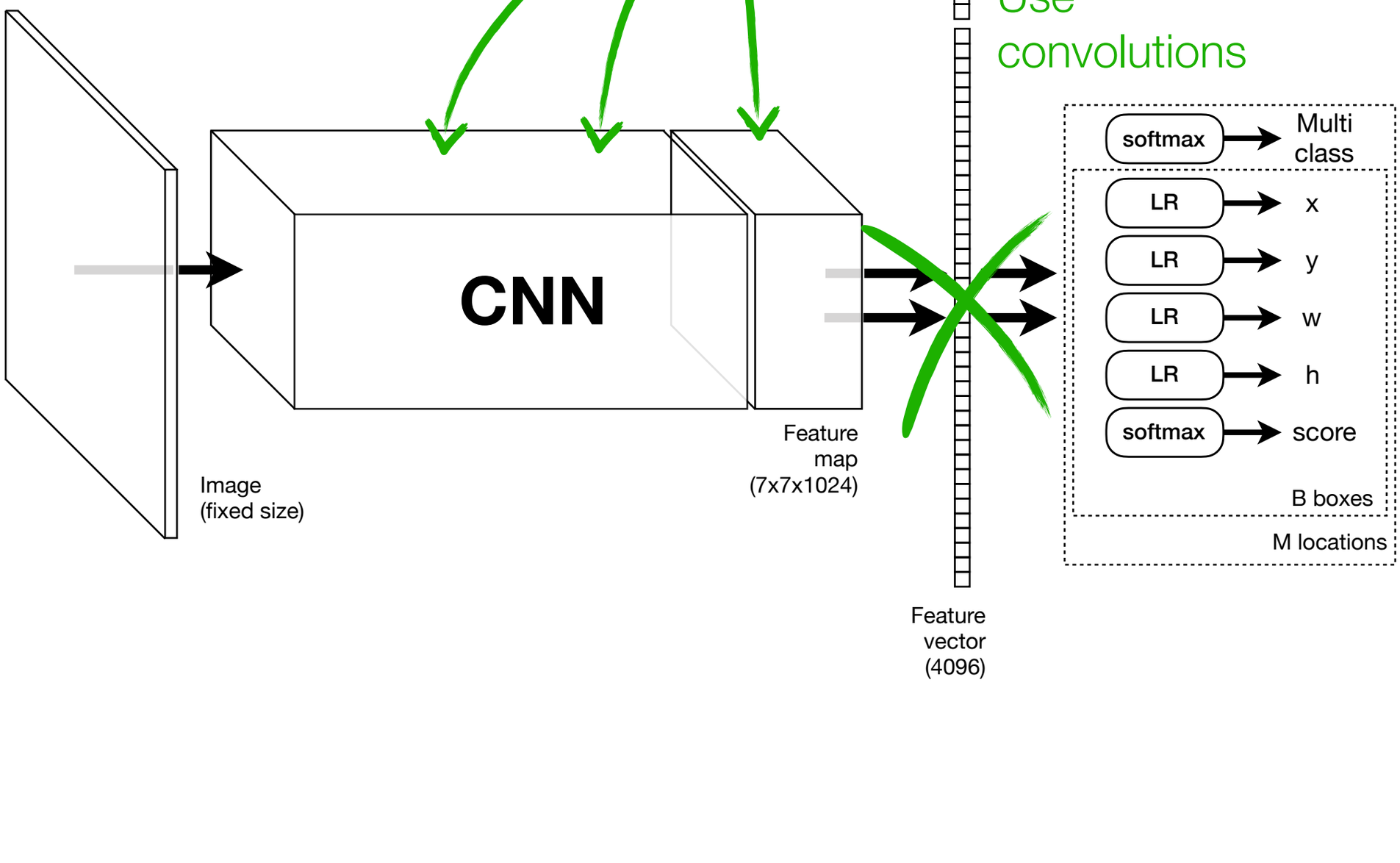
YOLO

(You Only Look Once)

Use multi-scale features

To avoid huge intermediate layer

Use convolutions



SSD

~~YOLO~~

(You Only Look Once)

Use multi-scale features

To avoid huge intermediate layer

Use convolutions