AI Summer School 2024
Medical Imaging Informatics

University of Pittsburgh

Non-max Suppression, YOLO, SSD

Instructor: Nick Littlefield, MS

Pitt SHRS
School of Health and Rehabilitation Sciences

CPAiCE
Computational Pathology & AI Center of Excellence

IEEE COMPUTER SOCIETY

Learning Objectives

After completing this lecture, you should be able to:

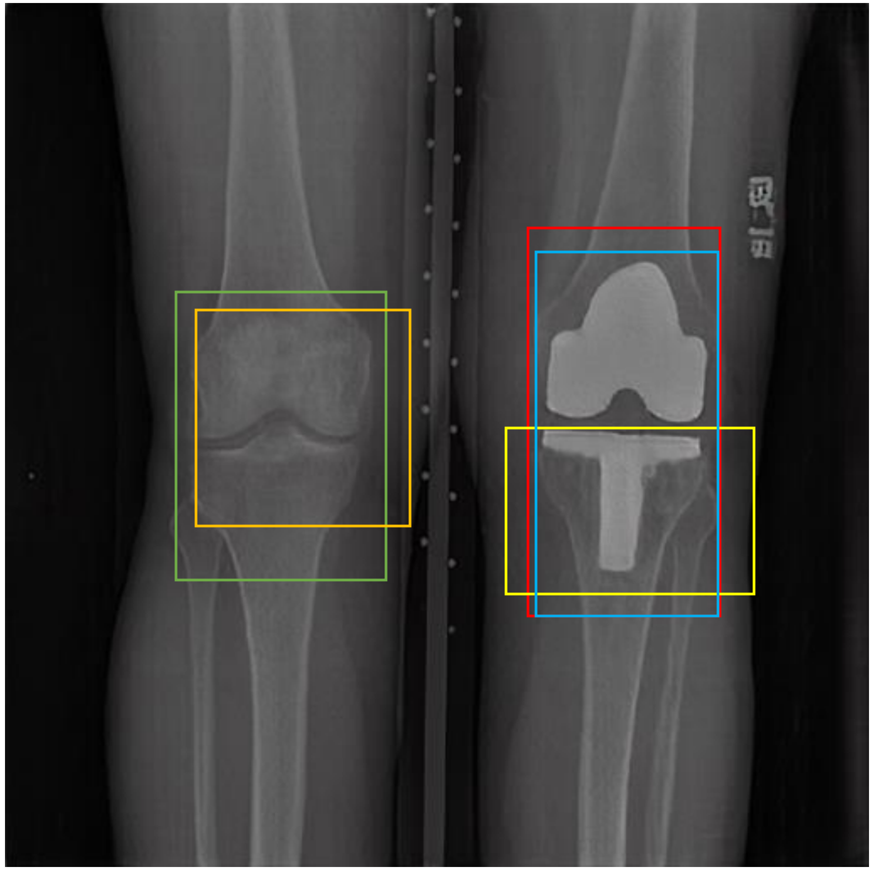
- Understand how non-max suppression can be used to select the best bounding box
- Understand the architecture and how “You Only Look Once” (YOLO) works
- Understand the architecture and how Single Shot Detection (SSD) works
- Explain the strengths and weaknesses of YOLO and SSD and their applications

Outline

- Non-max Suppression
- YOLO (You Only Look Once)
- SSD (Single Shot Detector)

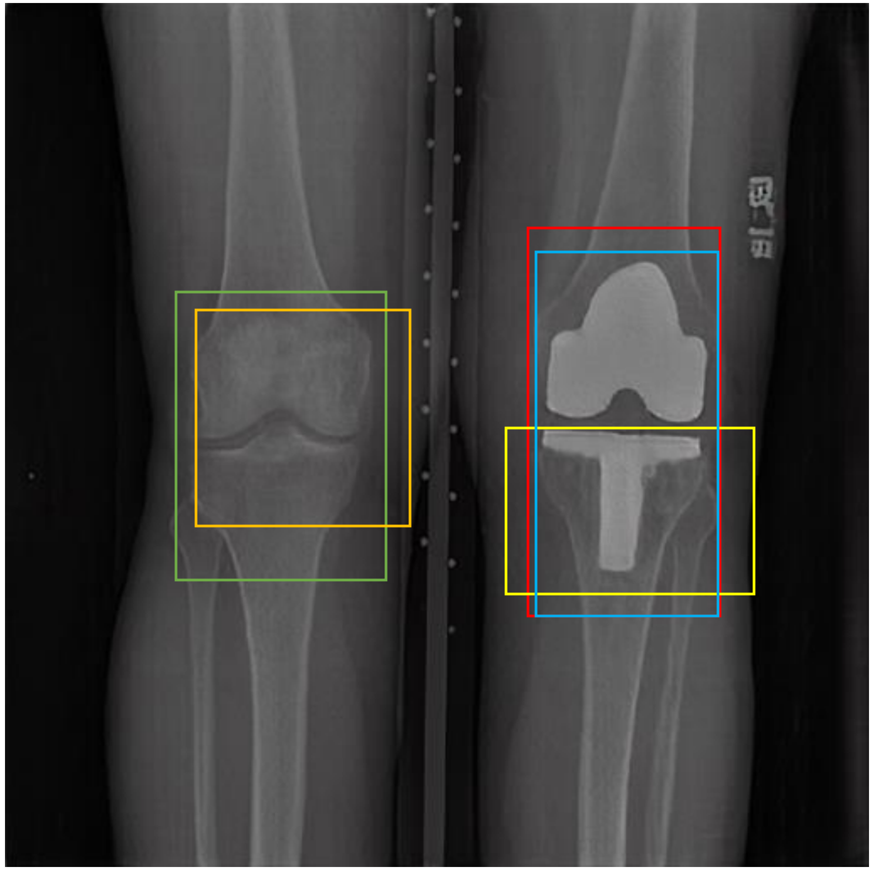
Non-max Suppression (NMS)

- **Problem:** Object detection methods will often output many detections that are overlapping



Non-max Suppression

- **Problem:** Object detection methods will often output many detections that are overlapping
- **Solution**: Eliminate redundant boxes that overlap (suppress) ensuring only the bounding boxes that remain is the best one



Non-max Suppression

- **Algorithm:**
    1. Remove all bounding boxes with a confidence score below a certain threshold
    2. Sort remaining bounding boxes by their confidence score
    3. Select the bounding box with the highest confidence score (primary bounding box)
    4. For the remaining bounding boxes calculate the IoU with the primary bounding box and discard all bounding boxes above some IoU threshold, removing all redundant predictions (Suppression step)
    5. Repeat steps 3 and 4 until no more bounding boxes remain

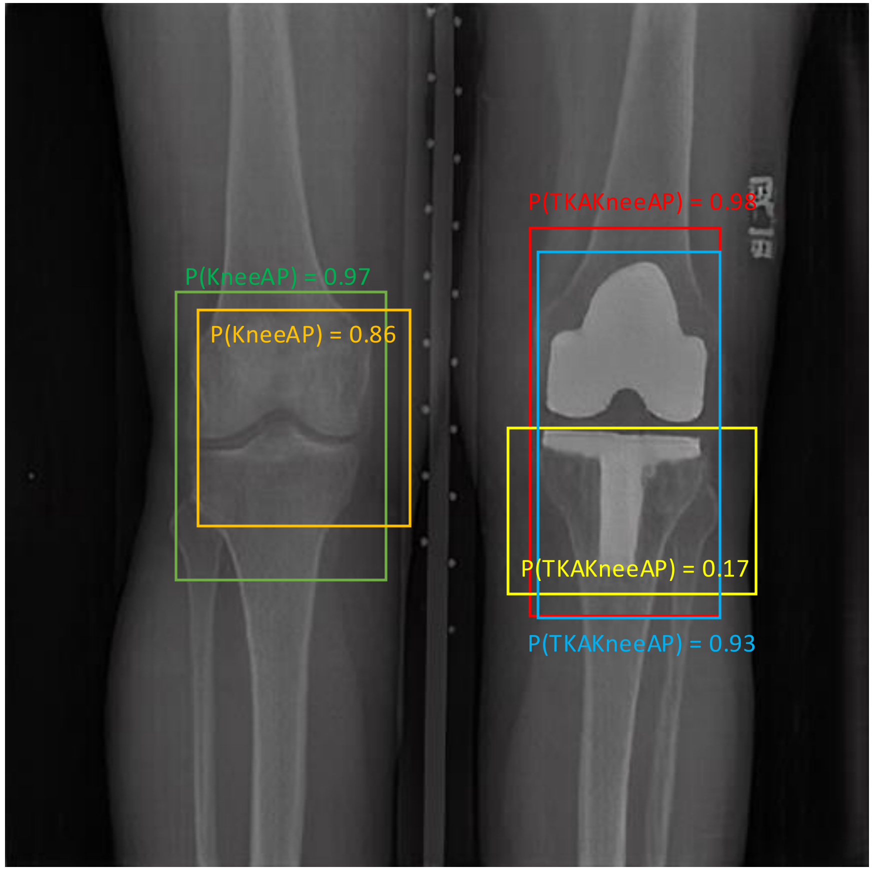
Non-max Suppression: Example

Confidence Threshold: 0.5

IoU Threshold: 0.7

Step 1: Remove all bounding boxes with a confidence score below a certain threshold

Remove Yellow bounding box (0.17 < 0.5)



Non-max Suppression: Example

Confidence Threshold: 0.5

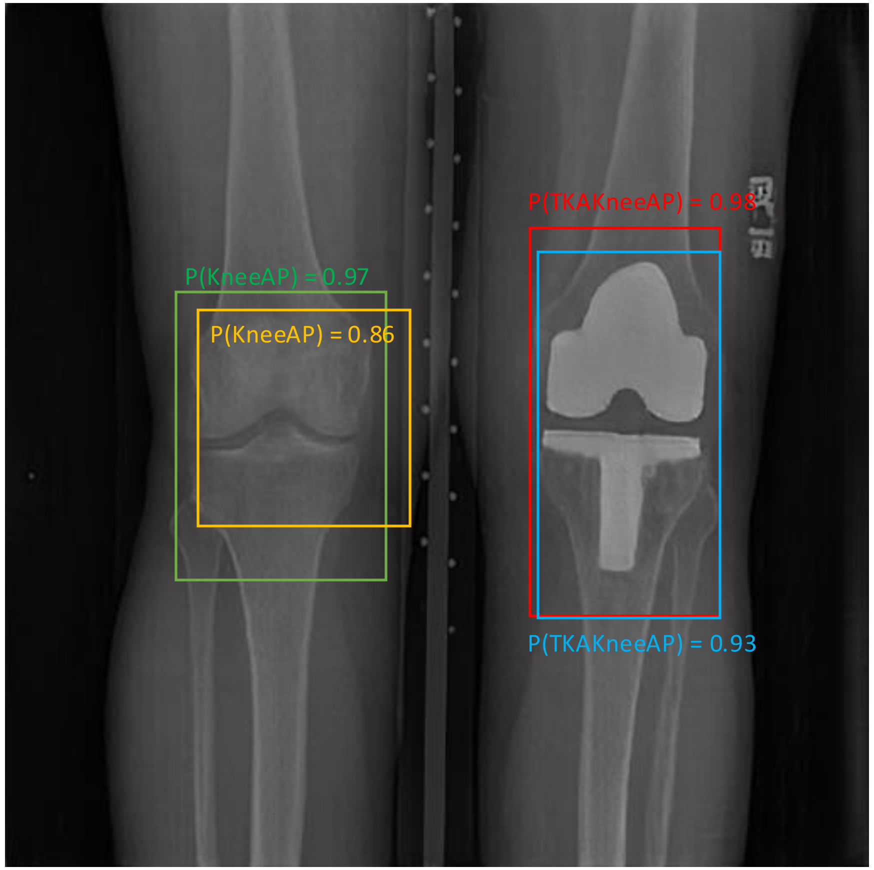
IoU Threshold: 0.7

Step 2: Sort remaining bounding boxes by their confidence score

0.98 0.97 0.93 0.86

Step 3: Select the bounding box with the highest confidence score (primary bounding box)

0.98



Non-max Suppression: Example

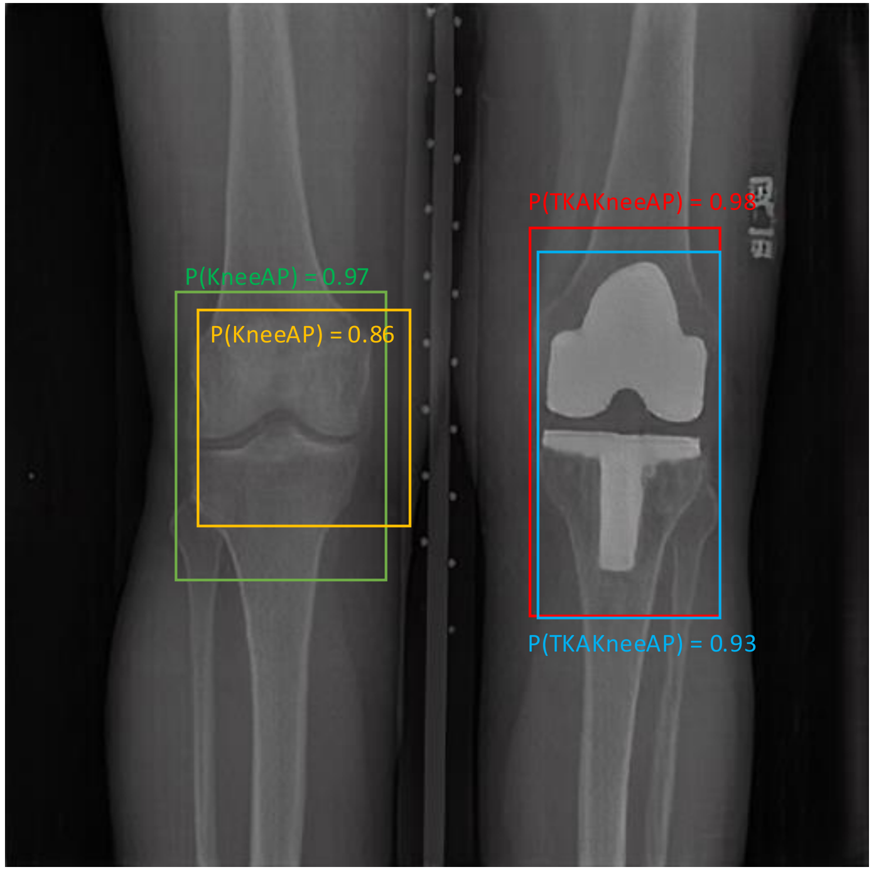
Confidence Threshold: 0.5

IoU Threshold: 0.7

Step 4: For the remaining bounding boxes calculate the IoU with the primary bounding box and discard all bounding boxes above some IoU threshold, removing all redundant predictions

IoU(Red, Blue) = 0.91
IoU(Red, Green) = 0
IoU(Red, Orange) = 0

IoU(Red, Blue) > 0.7, remove this box



Non-max Suppression: Example

Confidence Threshold: 0.5

IoU Threshold: 0.7

Step 5: Repeat steps 3 and 4 until no more bounding boxes remain

Step 3:

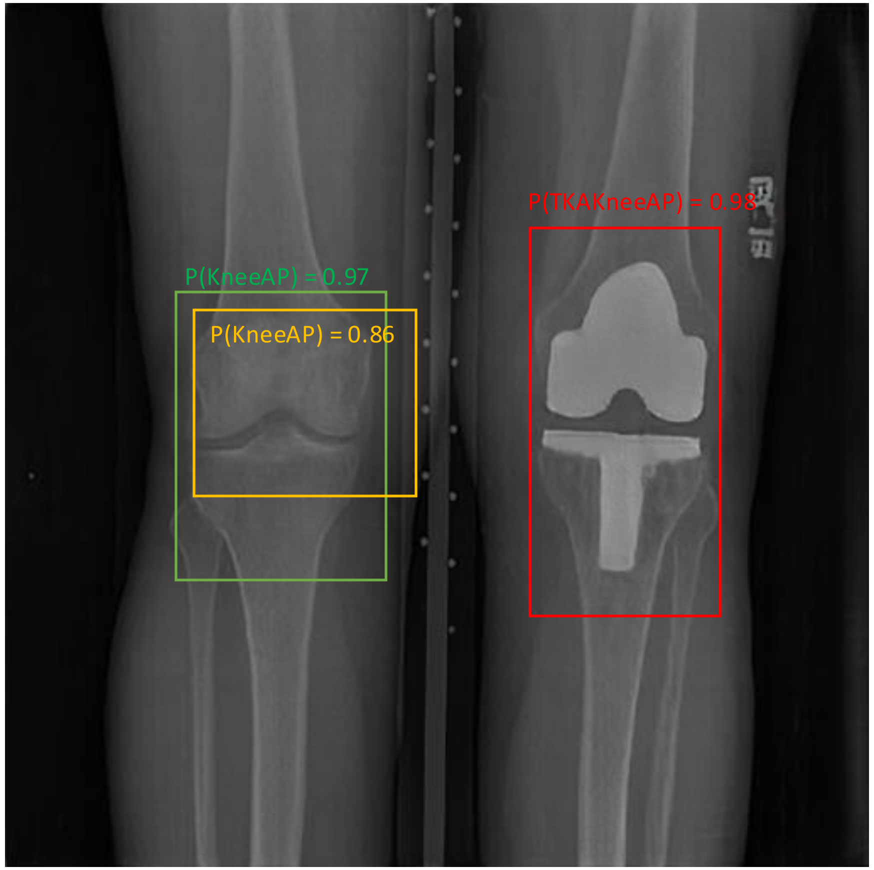
0.97 **0.93** **0.86**

Max Score: **0.97**

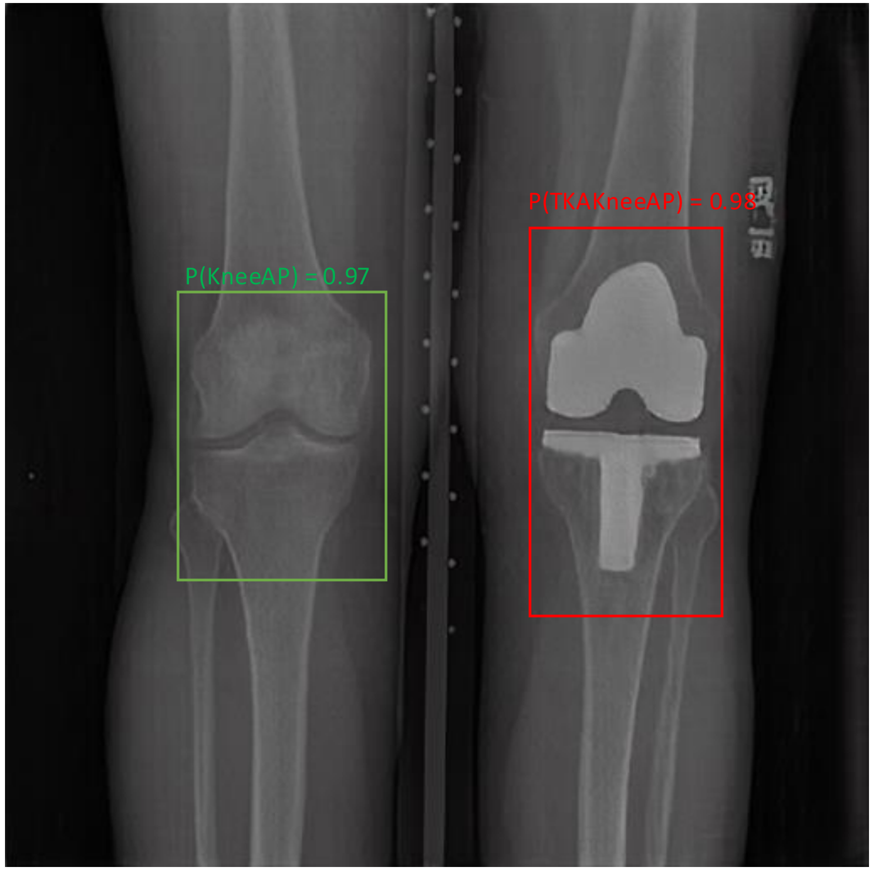
Step 4:

IoU(Green, Orange) = 0.65

IoU(Green, Orange) **> 0.65, remove this box**

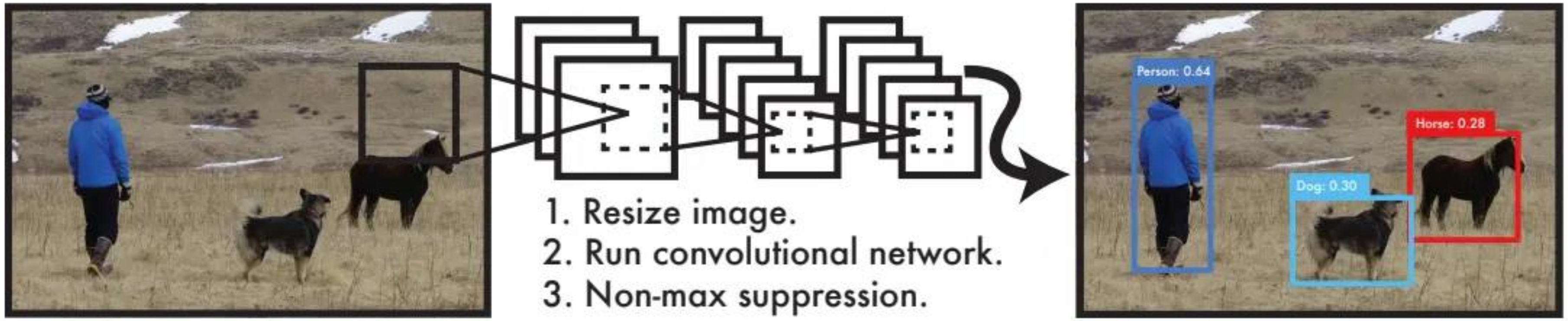


Non-max Suppression: Example



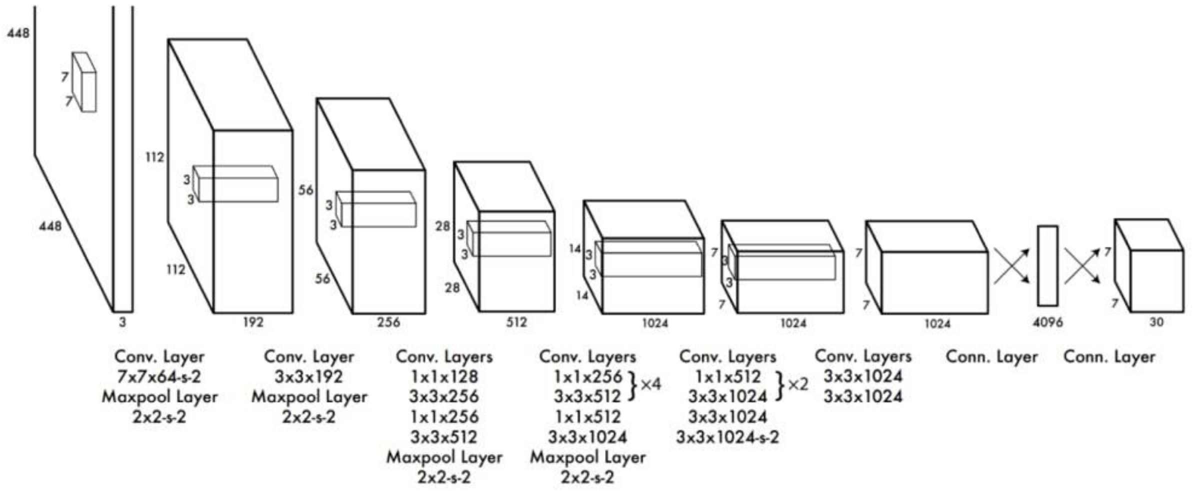
YOLO (You Only Look Once)

- **YOLO** is a state-of-the-art (SOTA) object detection algorithm that frames object detection tasks as a regression problem that spatially separates bounding boxes and their class probabilities
- Known for its speed allowing for use in real-time
- Uses a single CNN



YOLO Architecture

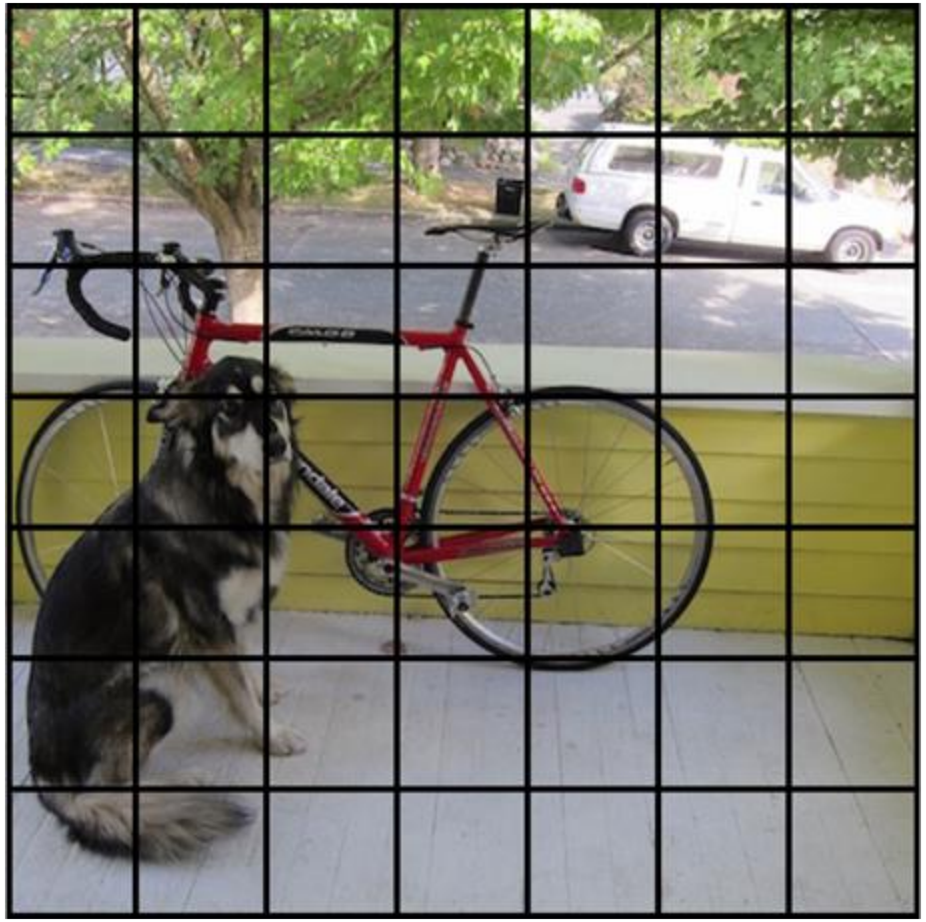
- A single CNN responsible for dividing an image into a grid
- Each grid predicts a certain number of bounding boxes and a class probability
- Class probabilities indicate the likelihood of a specific object being in present in the bounding box



The Architecture. Our detection network has 24 convolutional layers followed by 2 fully connected layers. Alternating 1×1 convolutional layers reduce the features space from preceding layers. We pretrain the convolutional layers on the ImageNet classification task at half the resolution (224×224 input image) and then double the resolution for detection.

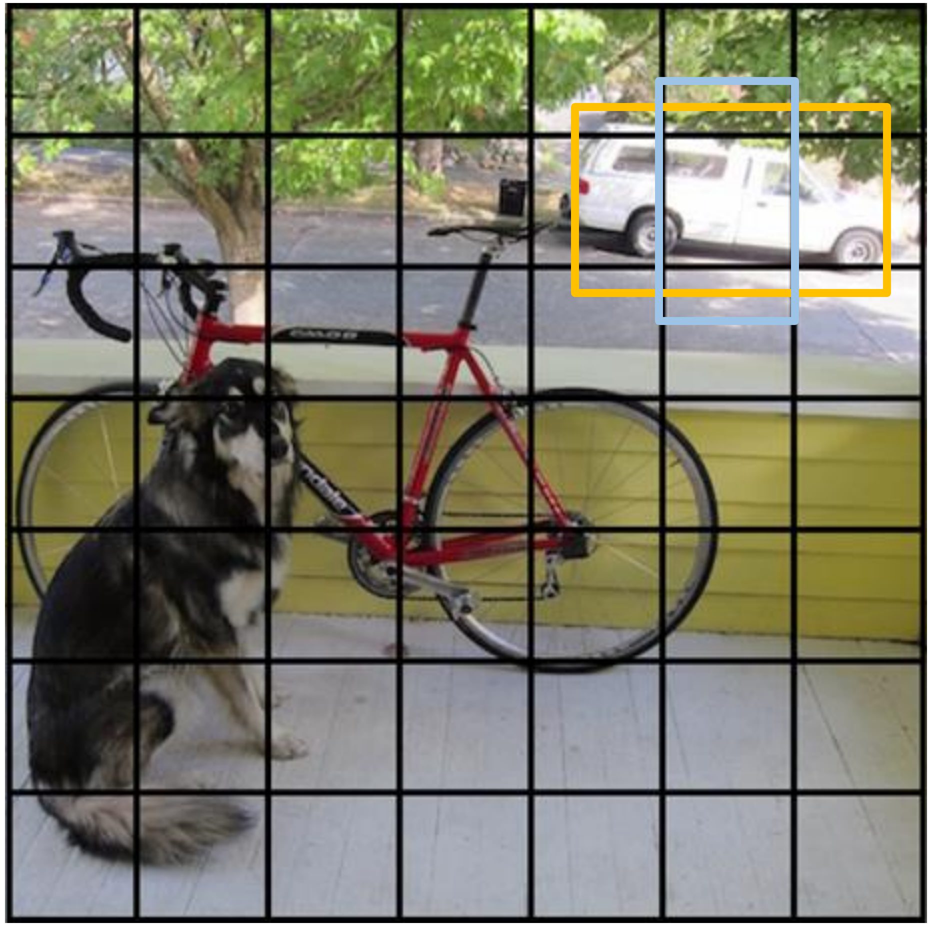
Bounding Box Detection

- An image is divided in an *S x S* grid, where each cell is responsible for predicting an object if its center is within it



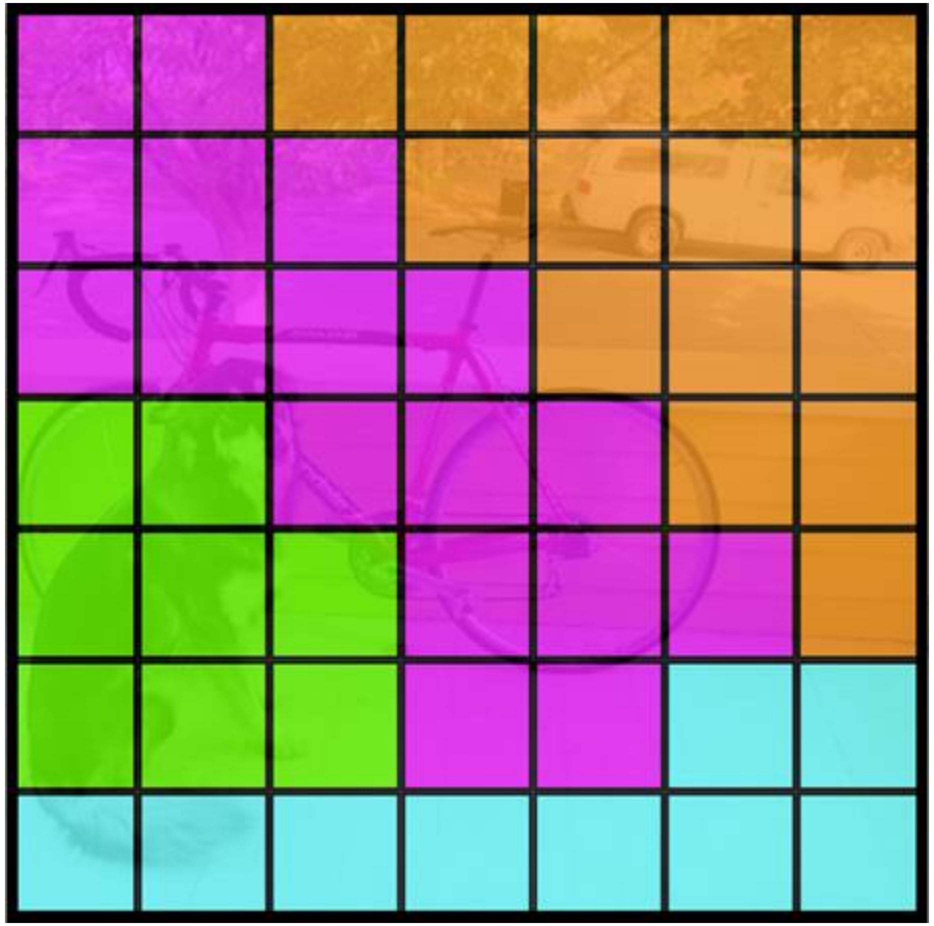
Bounding Box Detection

- Each cell predicts *B* bounding boxes and their corresponding confidence scores
- Confidence scores reflect how certain the model is that the bounding box contains a given object and how accurate the bounding box is



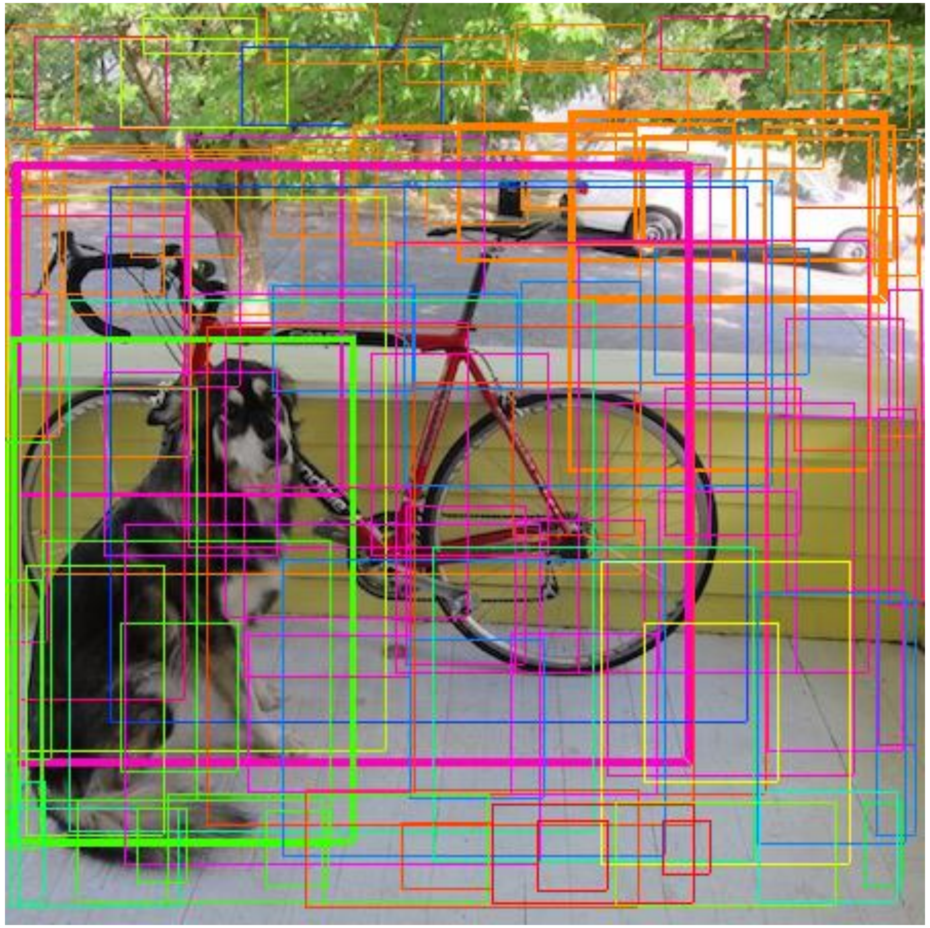
Bounding Box Detection

- Each cell Predicts *C* conditional class probabilities (one per class) representing the probability of the object being in the bounding box



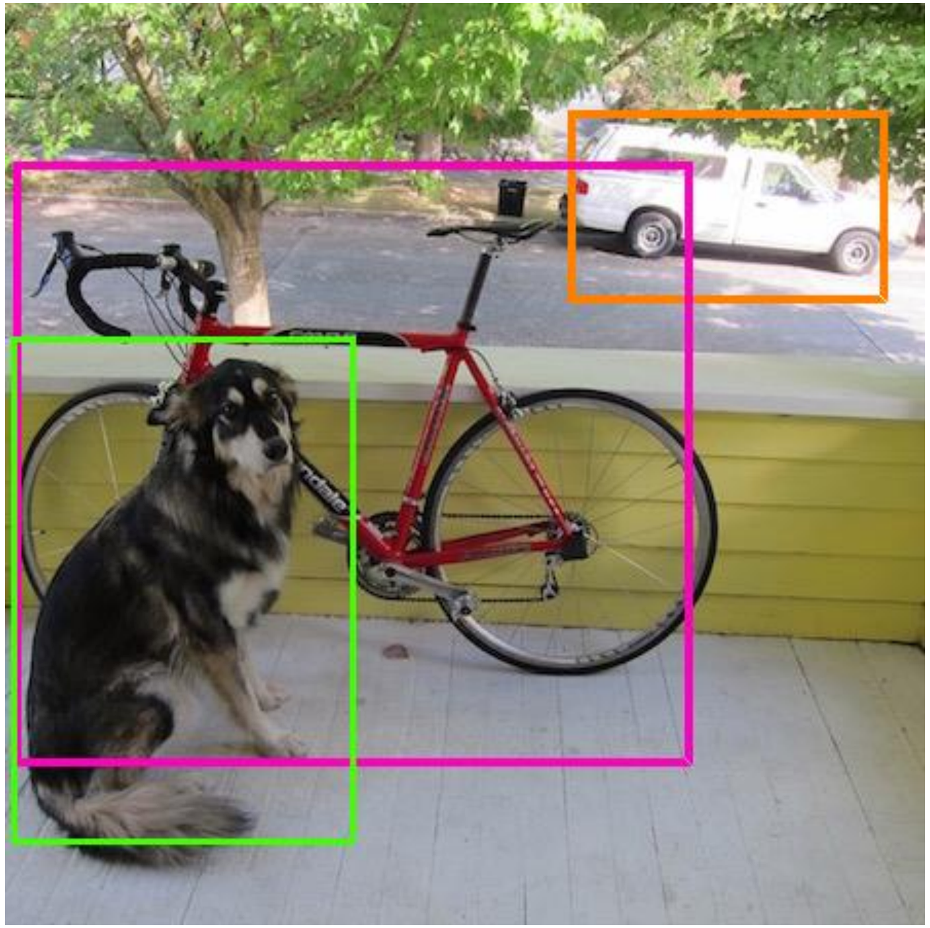
Bounding Box Detection

- Box and class probabilities are combined



Bounding Box Detection

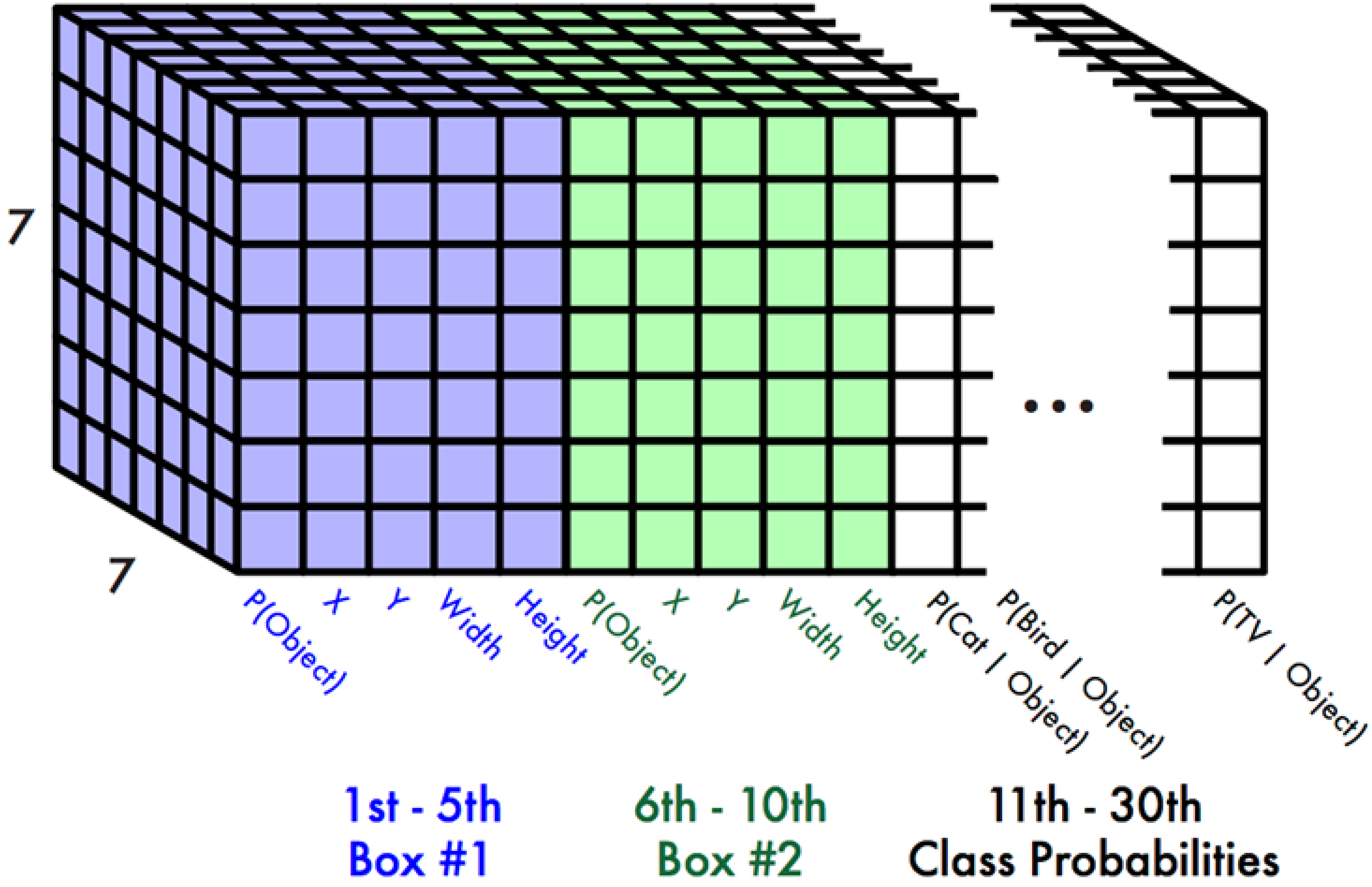
- NMS is then performed to get final predictions



YOLO Output

- Output is fixed based on parameterization
- For each bounding box:
  - 4 coordinates (x, y, w, h)
  - 1 confidence value

- Some number of class probabilities

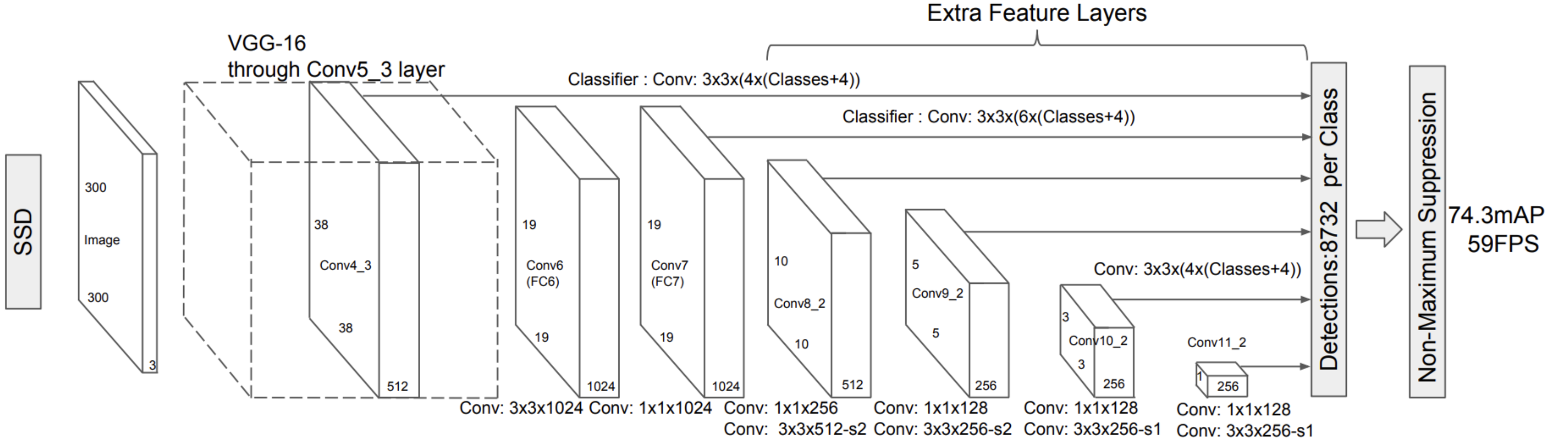


YOLOv9

- **YOLOv9** is the latest version of the YOLO architecture
- Overcomes the information loss challenges that occur in deep learning
- Overcomes the Information Bottleneck Principle:
  - As data passes through successive layers in a network the potential for information loss increases.
  - Programmable Gradient Information (PGI) aids in preserving the essential data across the network
- Reversible Functions: Mitigates the risk of information degradation by ensuring preservation of critical data needed for object detection tasks
  - A function is reversible if it can be inverted without any loss of information

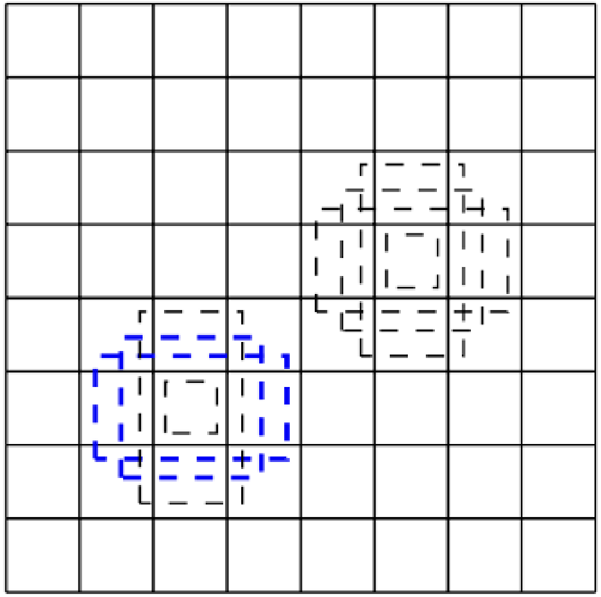
SSD (Single Shot Detector)

- An alternative to YOLO that can be used for real time detection
- Based on a feed-forward CNN that produces a fixed-size collection of bounding boxes (default boxes) and probabilities for the presence of an object
- Multi-scale feature maps are used for detection to allow for prediction of objects at different scales
- Predicted outputs are combined and NMS is applied to get final bounding box predictions



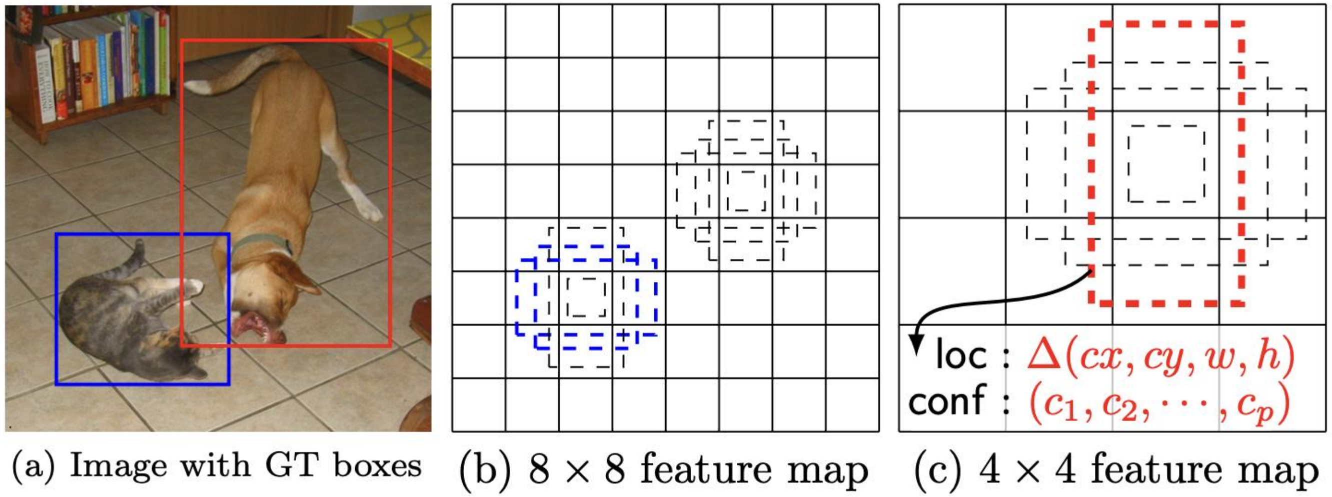
Default Boxes

- For each feature map, a set of default boxes for every cell
- Each cell predicts predicts the offsets relative to the shape of the default box and the scores for each class indicating whether a class object is present.
- Default boxes are used at different aspect ratios



Training: Matching Default Boxes to Ground Truths

- During training we need to determine which of the default boxes in the feature maps correspond to the ground truth bounding boxes
- We match each ground truth box to the default box with the best IoU score
- Any default box that matches with the ground truth and has an IoU score > 0.5 is additionally match allowing the network to predict higher scores for overlapping default boxes rather than picking only one


loc : $\Delta(cx, cy, w, h)$
conf : $(c_1, c_2, \cdots, c_p)$

(a) Image with GT boxes (b) 8 × 8 feature map (c) 4 × 4 feature map

Summary: YOLO and SSD

	YOLO	SSD
Speed vs. Accuracy	Prioritizes speed and real-time performance over accuracy	Achieves good balance between both speed and accuracy
Small Objects	Struggles with detection	Can detect due to the multi-scale feature maps
Flexibility	More flexible wit input size during inference	Requires predefined default boxes for both training and inference

Thank you!

Questions!