



AI and Deep Learning

4-4 More Examples

Zhonglei Wang

WISE, SOE, CHOW

XMU

(Version v1)

Contents

1. MobileNet

2. YOLO

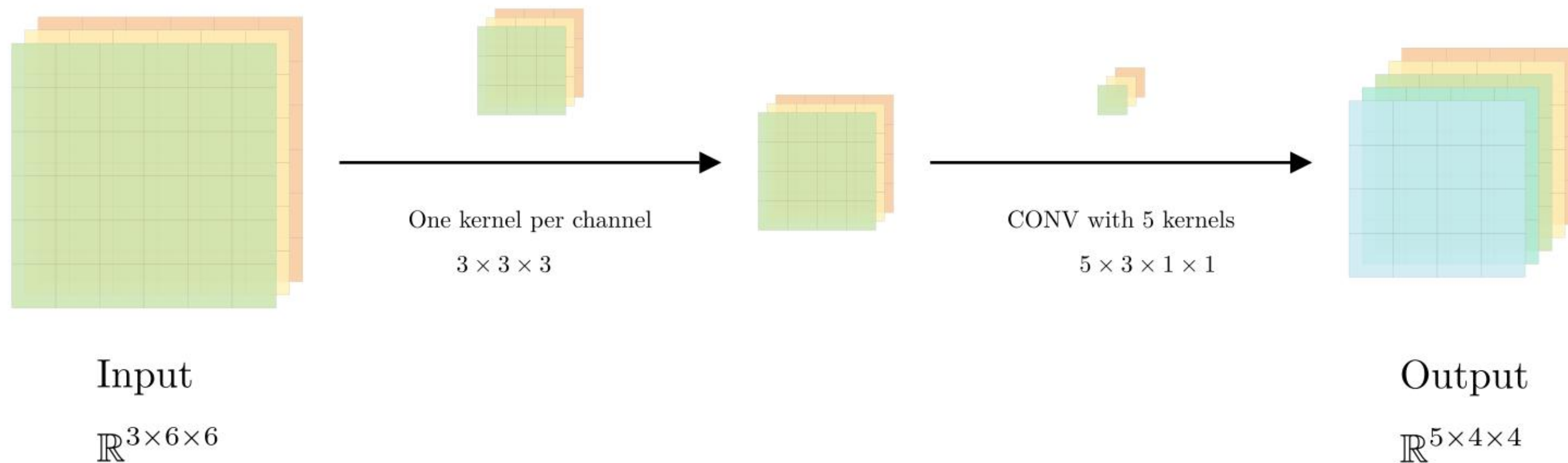
MobileNet

1. A computationally efficient model, which can be deployed on mobiles
 - It is based on “a timely fasion on a computationally limited platform”
 - Goal: Make CNN smaller and faster, without sacrificing performance
 - Key component: Depth-wise separable convolutions



Figure 1. MobileNet models can be applied to various recognition tasks for efficient on device intelligence.

Intuition



1. There are only $3 \times 3 \times 3 + 3 \times 5 = 42$ parameters for this layer

Remarks

1. Replace the traditional convolution with two separate parts
 - Less model parameters and faster computation
 - Keep satisfactory performance

YOLO

1. Up to now, we only consider classification tasks
2. Object detection is also an important task in practice
3. Next, we focus on classification as well as location detection

YOLO



[<https://www.superannotate.com/blog/yolo-object-detection>]

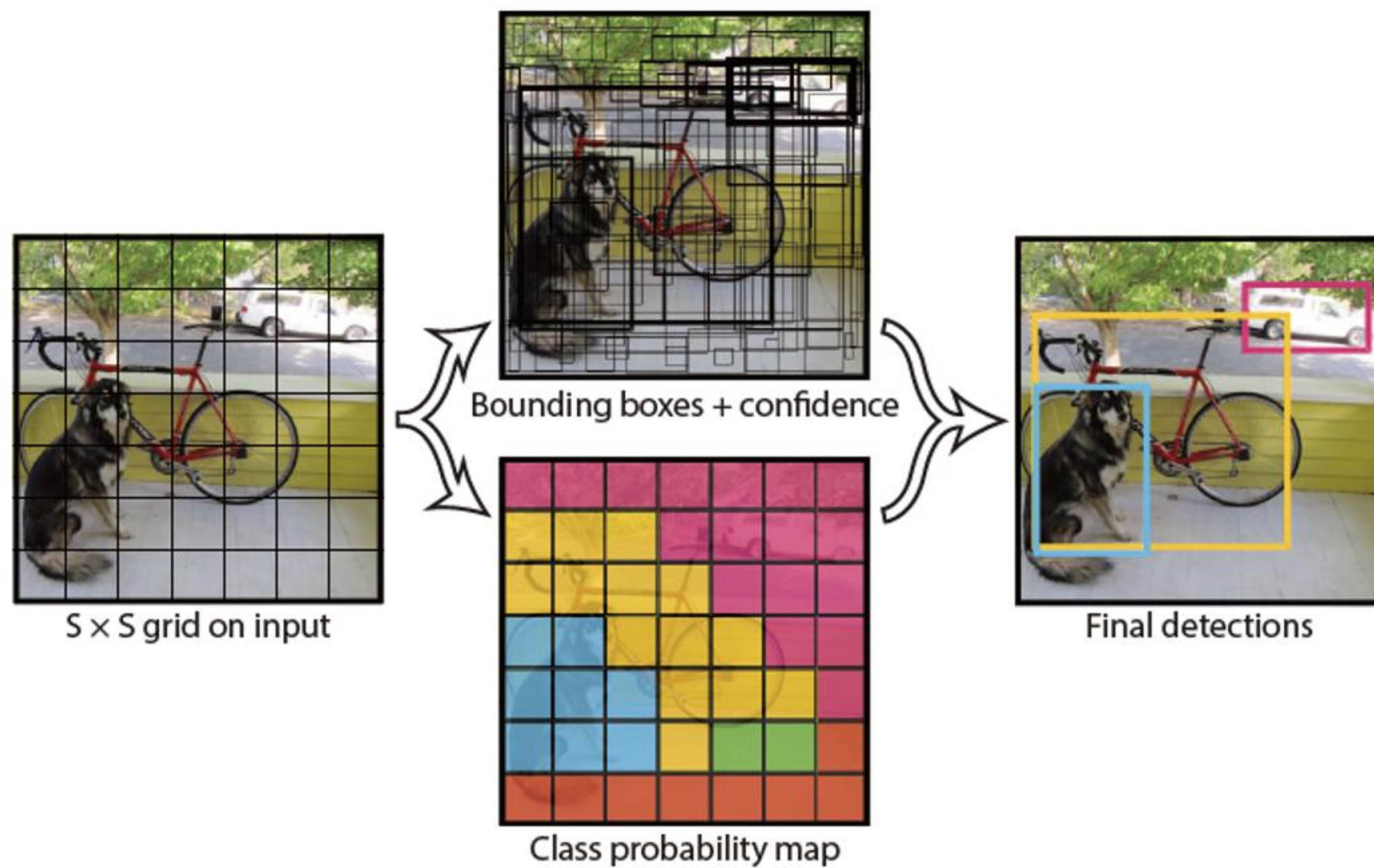
YOLO

1. We are interested in classifying C classes and identifying locations
2. YOLO (You Only Look Once) provides a good solution
3. We only focus on YOLO v1 (Redmon et al., 2016)
4. See <https://www.bilibili.com/video/BV1JT411j7MR?p=2> for more details

YOLO v1

1. Partition the image into $S \times S$ grid cells
2. Form B bounding boxes for each grid cell
3. For each grid cell, use IoU to select one “representative” bounding box
4. Minimize a specific cost function
5. Use non-max suppression algorithm to finalize the boundary

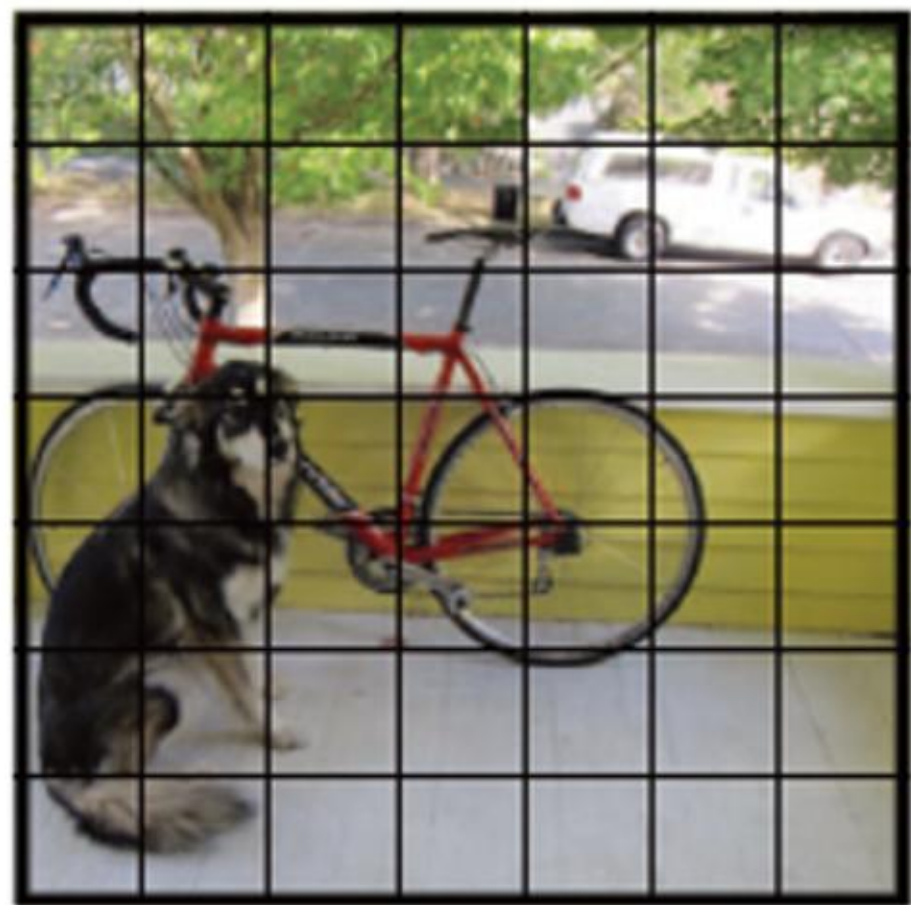
YOLO v1



[Redmon, J., Divvala, S., Girshick, R., Farhadi, A. (2016), You Only Look Once: Unified, Real-Time Object Detection, CVPR, 779–788]

YOLO v1

1. Partition the image into $S \times S$ grid cells



$S \times S$ grid on input

YOLO v1

1. For each grid cell, consider two candidate bounding boxes

- Only one bounding box is representative for the object, and IoU is used to select it
- Each bounding box is represented by **five** elements
 - ▷ x, y : Center of the box relative to **the bounds of that grid cell**
 - ▷ w, h : Width and height of the box relative to **the whole image**
 - ▷ *confidence*: Confidence score that the box contains an object and how accurate it thinks the box is that it predicts.

YOLO v1

1. IoU is short for “Intersection over Union”
2. During the training procedure, we observe labels, so we can choose a box with larger IoU for each grid cell
3. This bounding box can be further tuned for the object

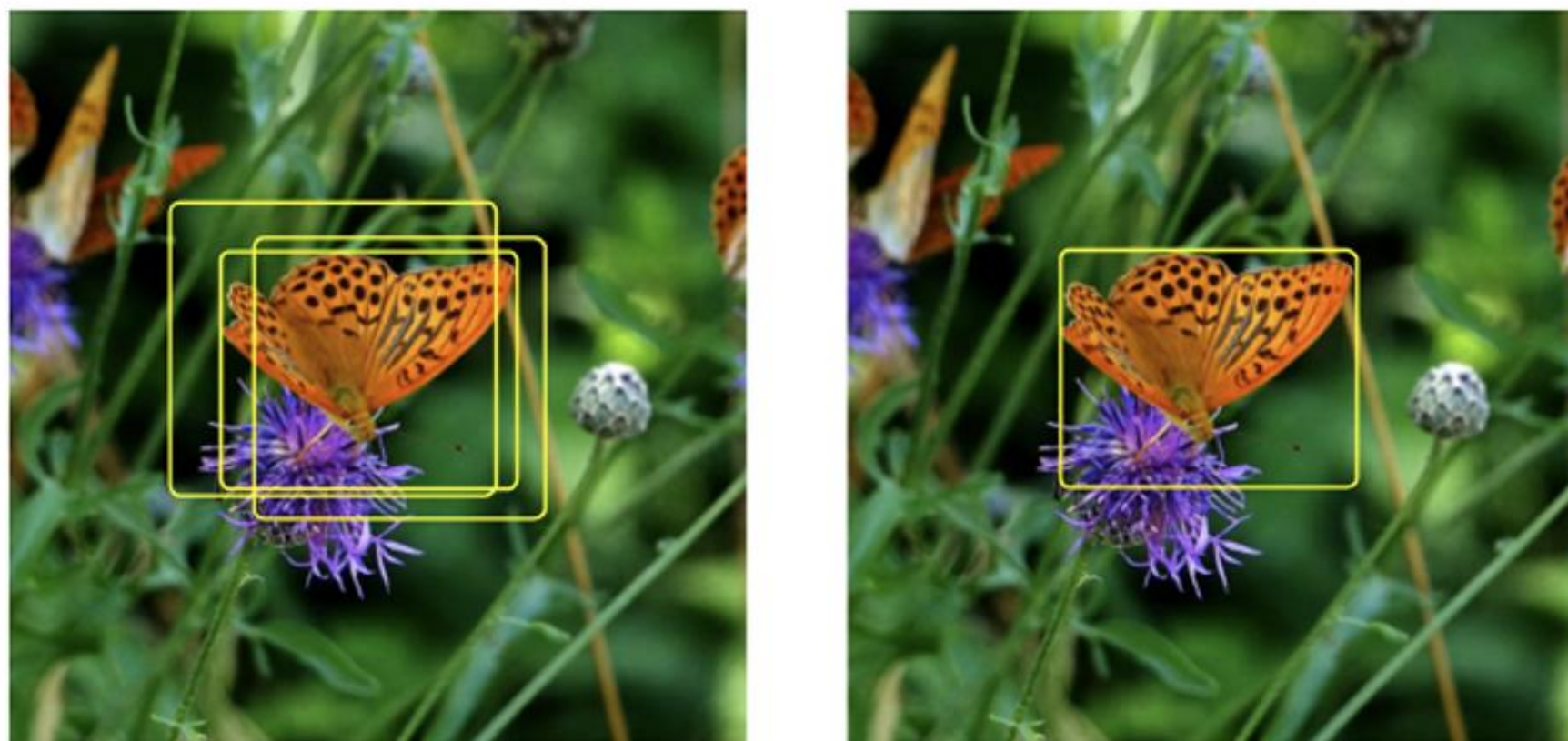
YOLO v1

1. The cost function, with $\lambda_{\text{coord}} = 5$ and $\lambda_{\text{noobj}} = 0.5$ is

$$\begin{aligned} & \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right] \\ & + \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right] \\ & + \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left(C_i - \hat{C}_i \right)^2 \\ & + \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i \right)^2 \\ & + \sum_{i=0}^{S^2} \mathbb{1}_i^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2 \end{aligned} \quad (3)$$

YOLO v1

1. For each class, discard all bounding boxes with confidence less than a threshold
2. Among the remaining overlapping ones, only keep the one with the largest confidence, and discard those with IoU larger than a threshold



[<https://www.oreilly.com/library/view/practical-machine-learning/9781098102357/ch04.html>]

Summary

1. MobileNet: Make CNN lighter and more suitable for deployment in small device
2. YOLO: Unify category classification and location prediction into a single efficient framework