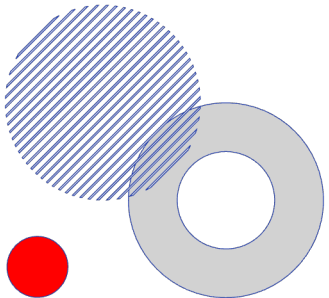
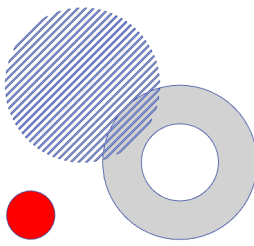


ms Detection 1111 Real-time

Real-Time Object Detection

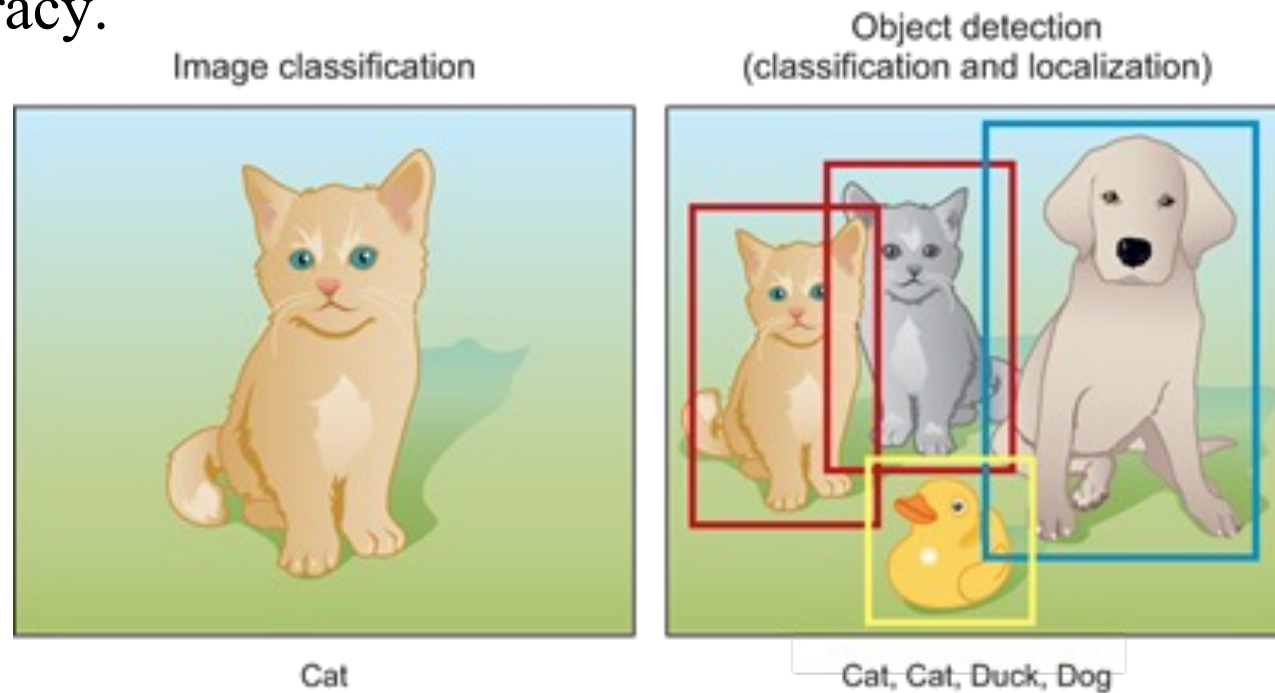
Lecturer: Dr. Thittaporn Ganokratanaa

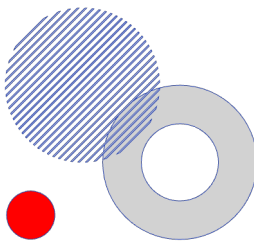




❖ Problem Addressed: Object Detection

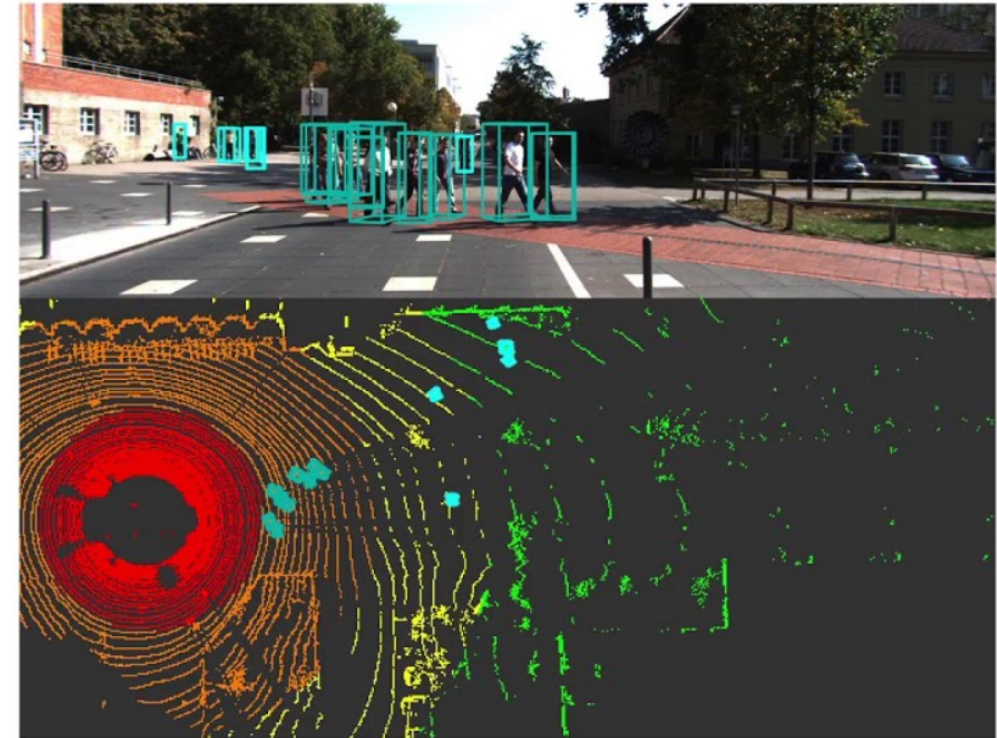
- Object detection is the problem of both locating AND classifying objects
- Goal of object detection algorithm is to do object detection both fast AND with high accuracy.

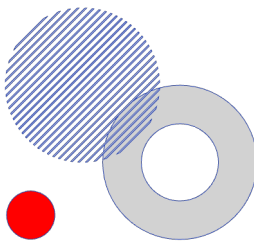




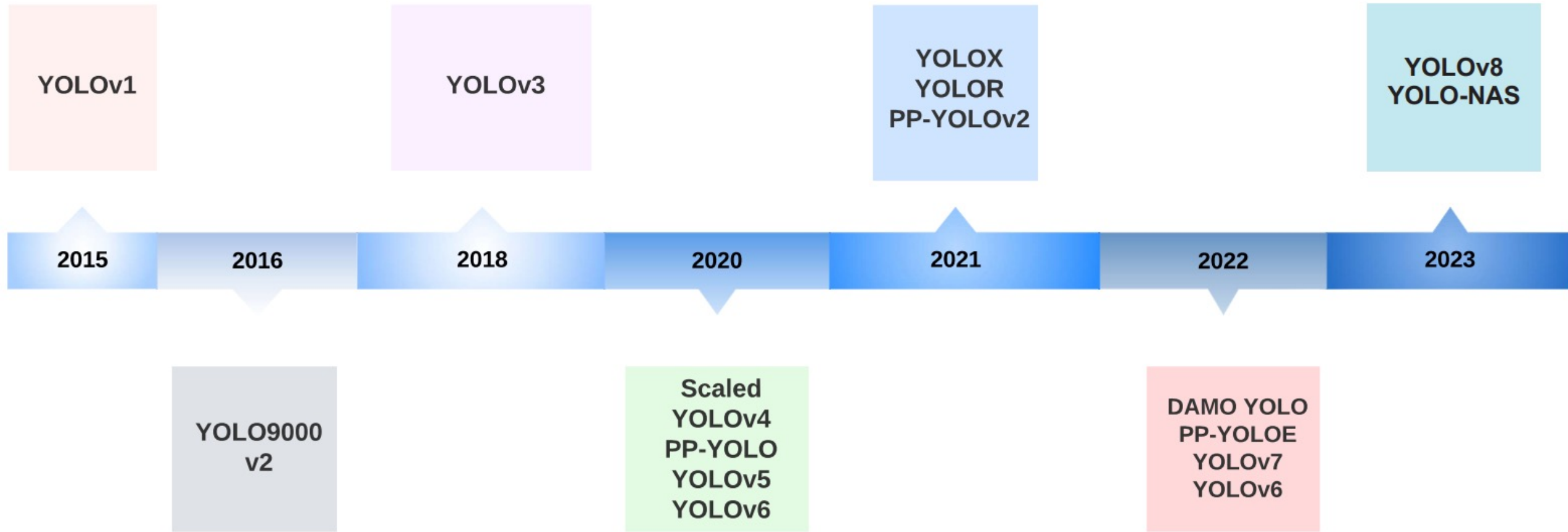
❖ Importance of Object Detection

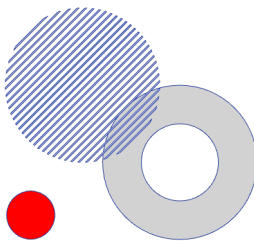
- Visual modality is very powerful
- Humans are able to detect objects and do perception using just this modality in real-time (not needing radar)
- If we want responsive robot systems that work real-time (without specialized sensors), almost real-time vision based object detection can help greatly.





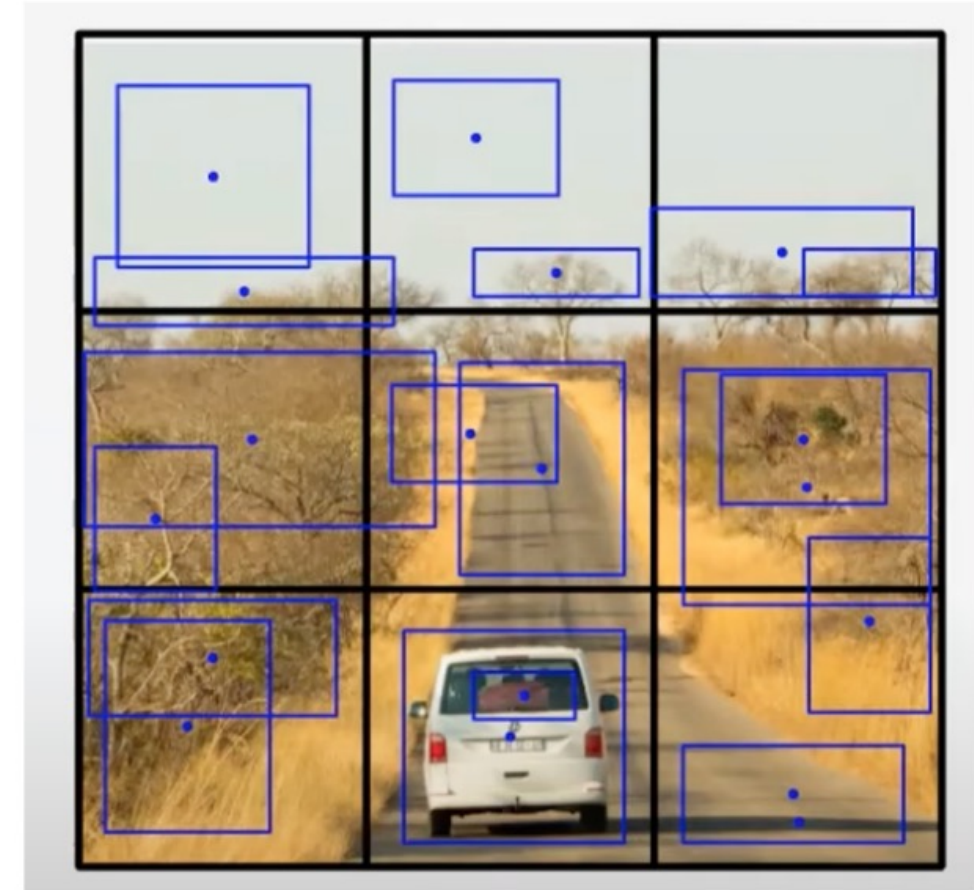
❖ A timeline of YOLO versions





❖ YOLO Overview

- First, image is split into a $S \times S$ grid
- For each grid square, generate B bounding boxes
- For each bounding box, there are 5 predictions:
 x, y, w, h , confidence



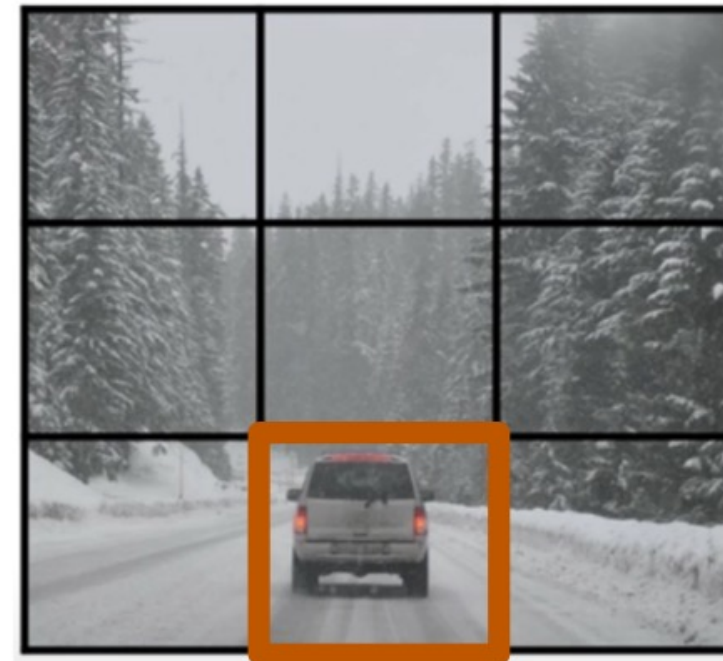
$$S = 3, B = 2$$

❖ YOLO Training

- YOLO is a regression algorithm. What is X? What is Y?
- X is simple, just an image width (in pixels) * height (in pixels) * RGB values
- Y is a tensor of size $S * S * (B * 5 + C)$
- $B * 5 + C$ term represents the predictions + class predicted distribution for a grid block

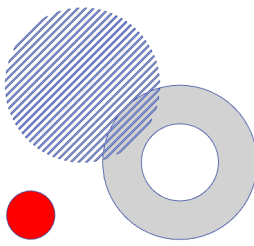
For each grid block, we have a vector like this. For this example B is 2 and C is 2

ρ_1
b_{x_1}
b_{y_1}
b_{h_1}
b_{w_1}
ρ_2
b_{x_2}
b_{y_2}
b_{h_2}
b_{w_2}
c_1
c_2



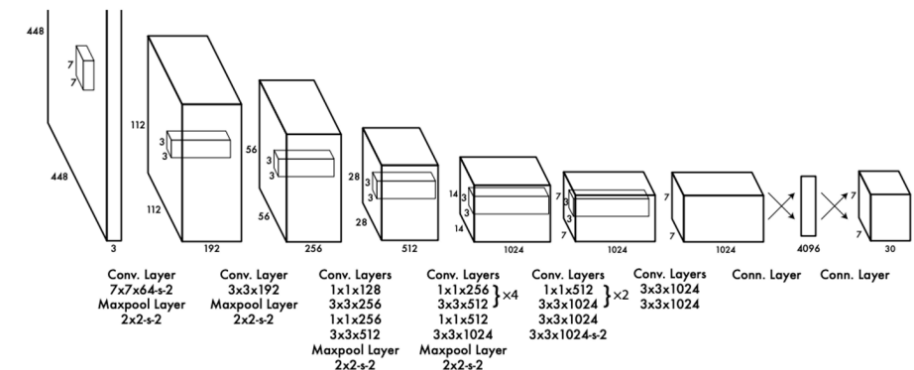
GT label
example:

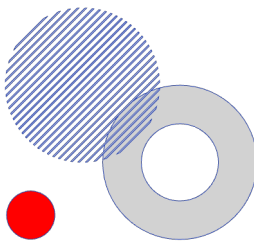
1
b_{x_1}
b_{y_1}
b_{h_1}
b_{w_1}
0
?
?
?
?
$c_1 = 1$
$c_2 = 0$



❖ YOLO Architecture

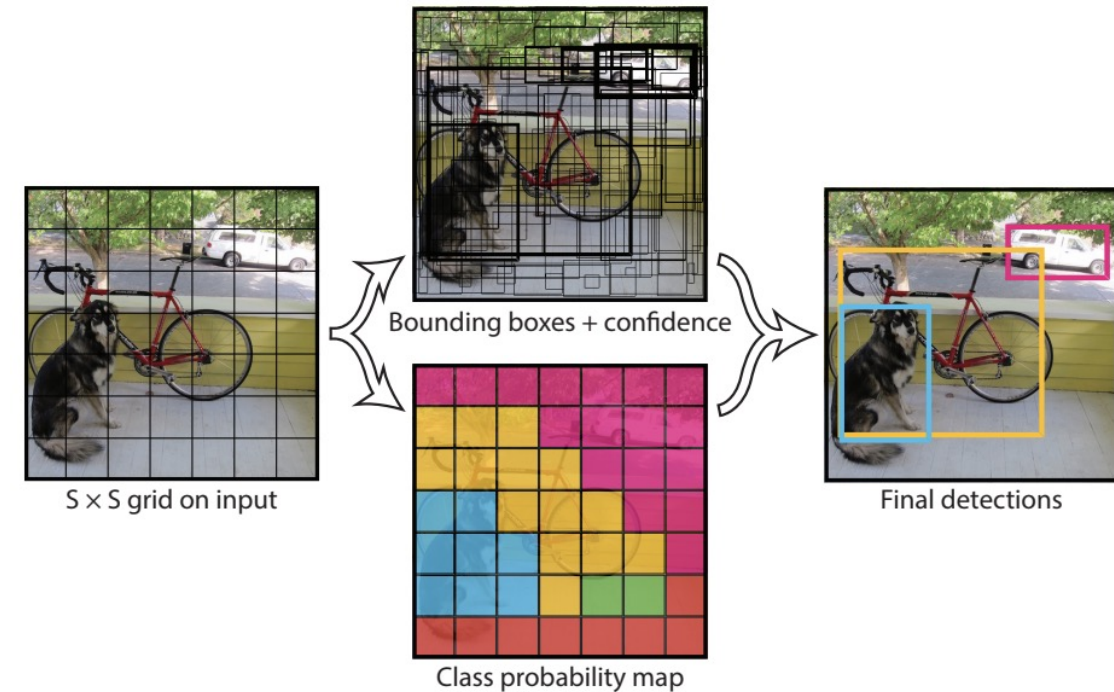
- Now that we know the input and output, we can discuss the model
- We are given 448 by 448 by 3 as our input.
- Implementation uses 7 convolution layers
- Paper parameters: $S = 7$, $B = 2$, $C = 20$
- Output is $S*S*(5B+C) = 7*7*(5*2+20) = 7*7*30$



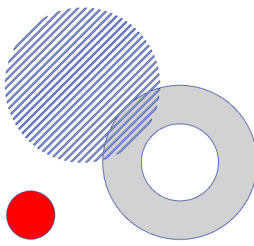


❖ Non-maximal suppression

- We then use the output to make final detections
- Use a threshold to filter out bounding boxes with low $P(\text{Object})$
- In order to know the class for the bounding box compute score take argmax over the distribution $\text{Pr}(\text{Class}|\text{Object})$ for the grid the bounding box's center is in

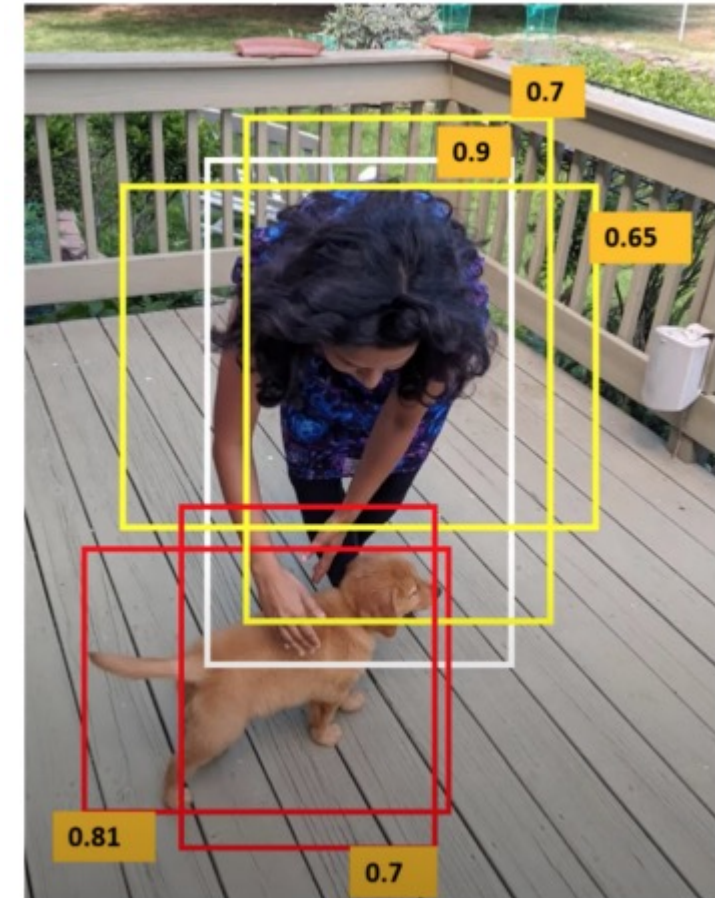


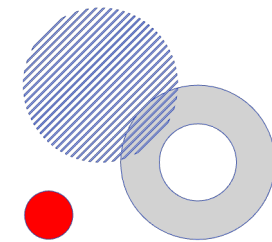
$$\text{Pr}(\text{Class}_i|\text{Object}) * \text{Pr}(\text{Object}) * \text{IOU}_{\text{pred}}^{\text{truth}} = \text{Pr}(\text{Class}_i) * \text{IOU}_{\text{pred}}^{\text{truth}}$$



❖ YOLO Prediction

- Most of the time objects fall in one grid, however it is still possible to get redundant boxes (rare case as object must be close to multiple grid cells for this to happen)
- Discard bounding box with high overlap (keeping the bounding box with highest confidence)
- Adds 2-3% on final mAP score





❖ YOLO Objective Function

Localization loss

Set to 5 to increase the loss of bounding box predictions

$$\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{obj} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right]$$

GT bbox x-coordinate in the i th cell

Predicted bbox x-coordinate in the i th cell

GT bbox y-coordinate in the i th cell

Predicted bbox y-coordinate in the i th cell

Sum-squared error

For each grid cell

For each grid box

'1' if object appears in the i th cell and the j th box detect it, '0' otherwise

$$+ \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{obj} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right]$$

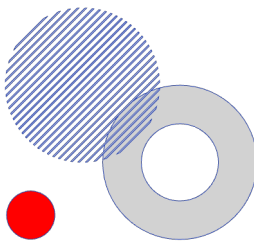
Square root to reduce the range of the values

GT bbox width in the i th cell

Predicted bbox width in the i th cell

GT bbox height in the i th cell

Predicted bbox height in the i th cell



❖ YOLO Objective Function (Cont.)

Confidence loss

$$+ \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{obj} \left[(C_i - \hat{C}_i)^2 \right]$$

GT confidence score Predicted confidence score

Confidence error when an object is detected in the i th cell

Set to 0.5 to decrease the loss for empty boxes

$$+ \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{noobj} \left[(C_i - \hat{C}_i)^2 \right]$$

'1' if there is no object in the i th cell, '0' otherwise

Confidence error when an object not detected in the i th cell

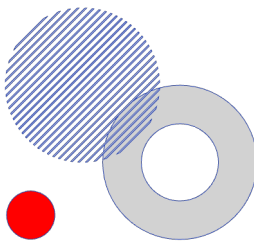
Classification loss

$$+ \sum_{i=0}^{S^2} \mathbb{1}_i^{obj} \sum_{c \in \text{classes}} \left[(p_i(c) - \hat{p}_i(c))^2 \right]$$

For each grid cell For each class

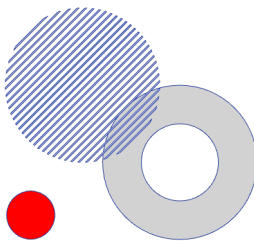
Predicted conditional probability of an object of class c appearing in the i th cell

GT conditional probability of class c appearing in the i th cell

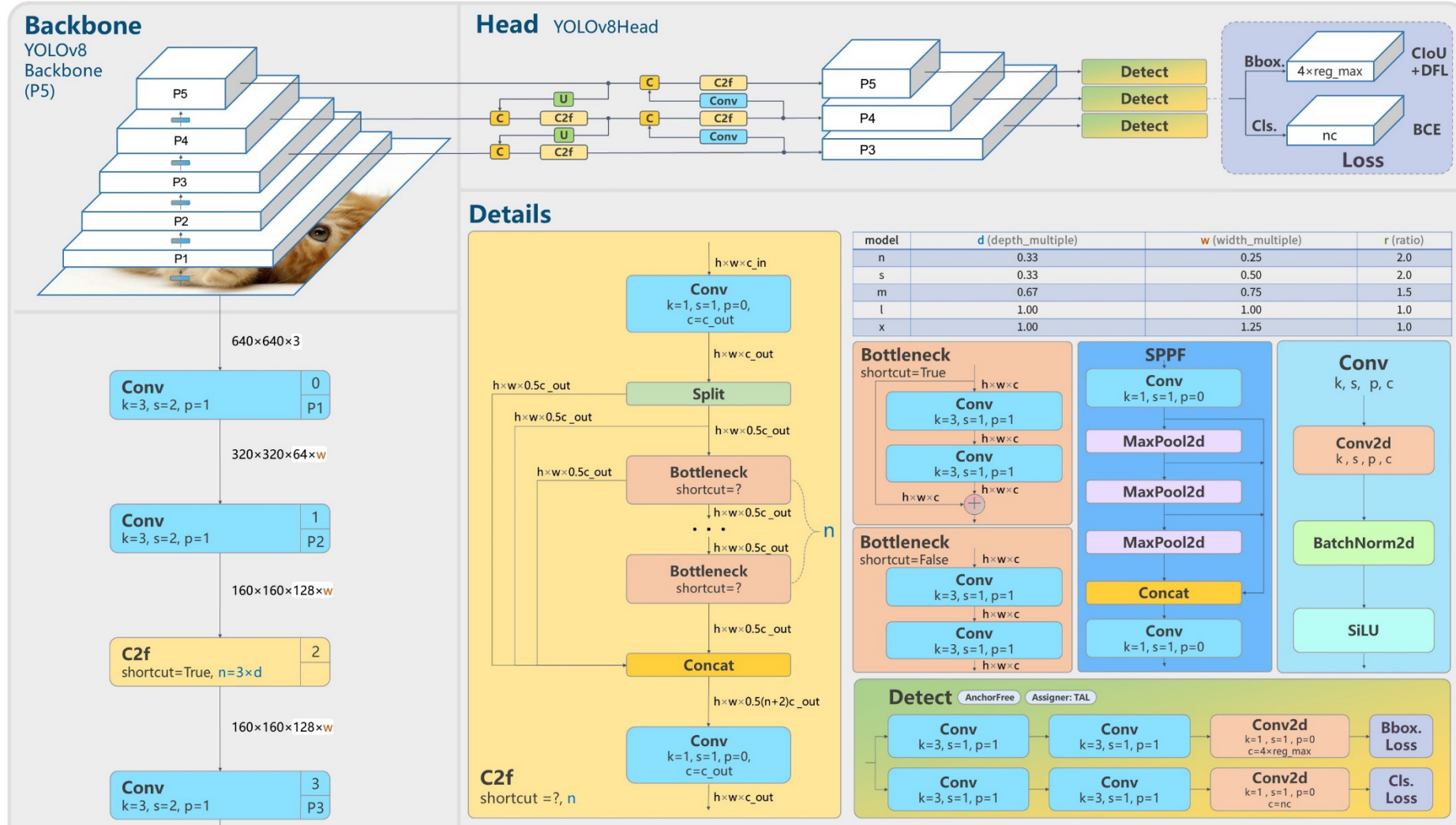


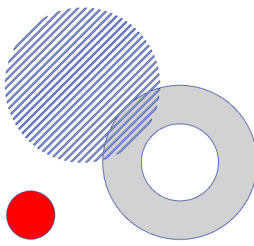
❖ YOLO V8

- YOLOv8 uses a similar backbone as YOLOv5 with some changes on the CSPLayer, now called the C2f module.
- The C2f module (cross-stage partial bottleneck with two convolutions) combines high-level features with contextual information to improve detection accuracy

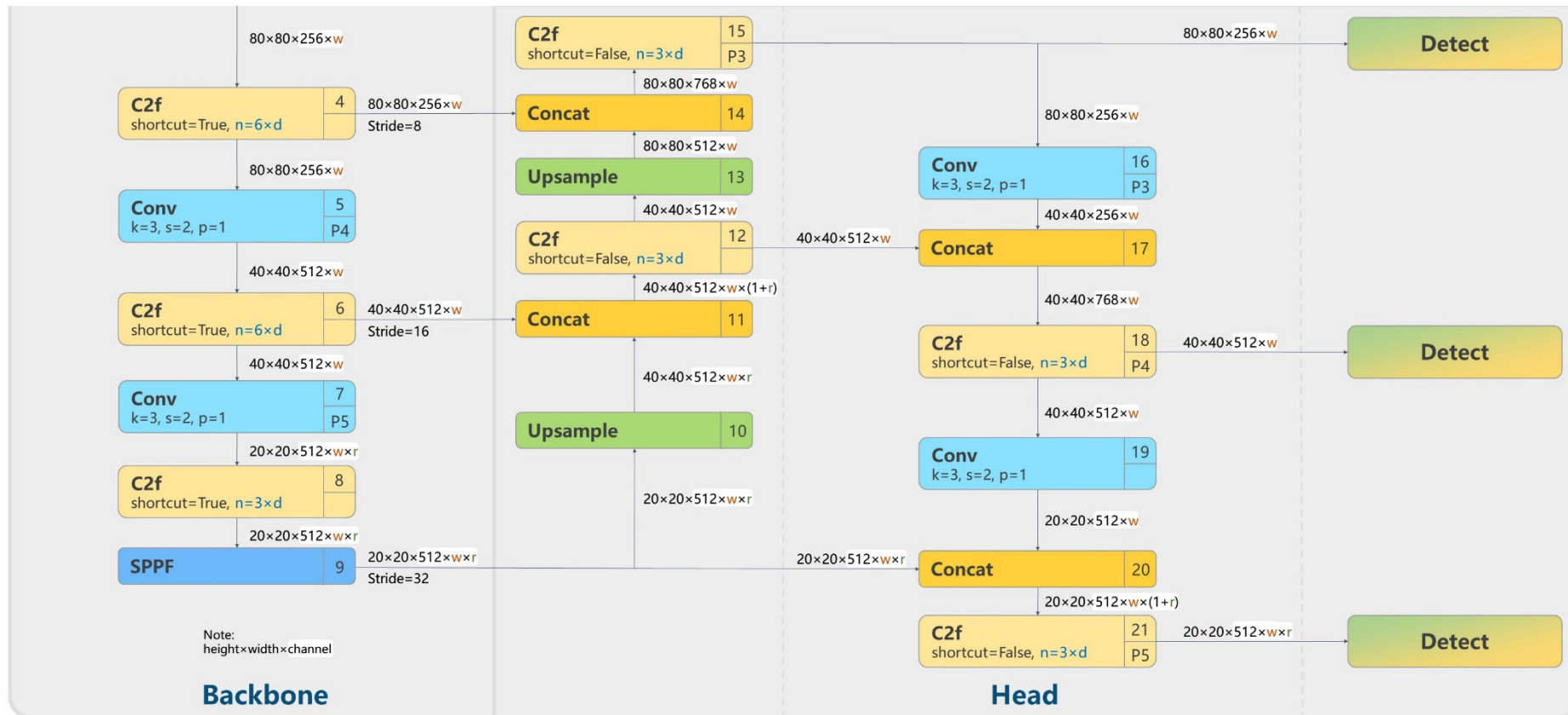


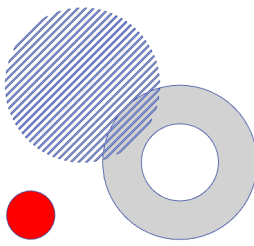
❖ YOLO V8 Architecture





❖ YOLO V8 Architecture (Cont.)





❖ YOLO V8 Experiment

➤ Using this Google Colab:

https://colab.research.google.com/drive/14x7_B44tBvAe8RzuETDVJ14cYWstnT2D?usp=sharing

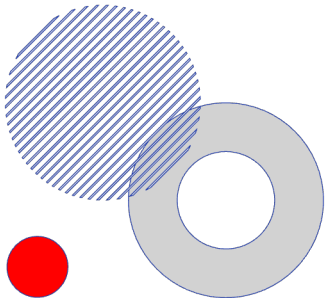
Exercise

Extract this video into frame and label it into four classes (bus, taxi, car, and pedestrian), then generate the model to classify those four classes using yolov8



Conclusion

- The research focused on utilizing AI technology to augment police efficiency in Thailand.
- We aimed to enhance law enforcement capabilities and bolster public trust in crime prevention measures.
- By employing AI in crime data analysis, leveraging intelligent CCTV technology for crime monitoring, and integrating real-time alerts for suspicious activities to police.



Q&A

