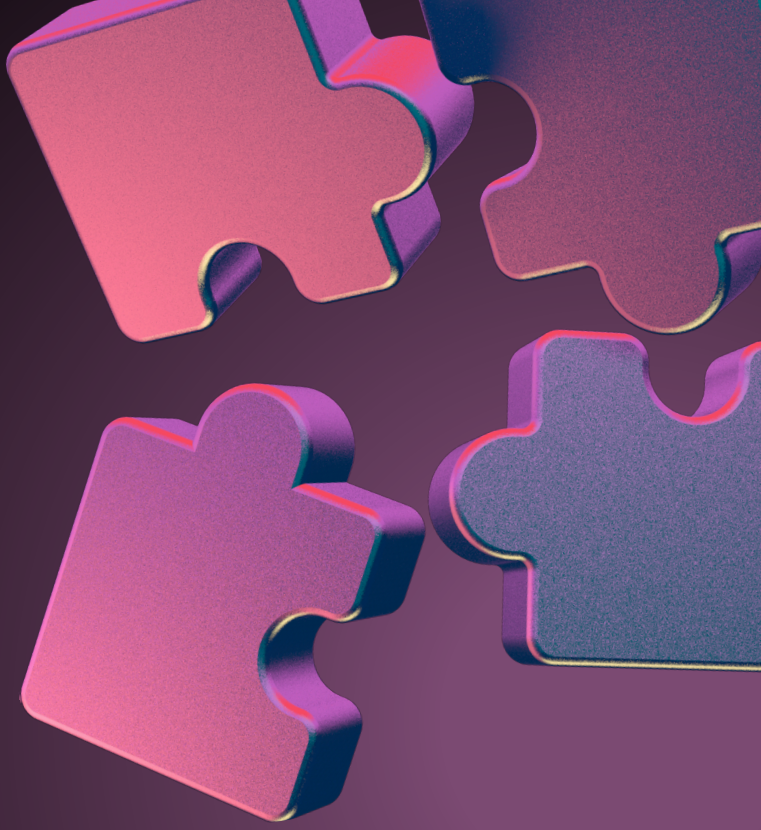
# Vehicles counting

## with YOLO + SSD + Faster-RCNN



Phạm Quốc Anh Khoa, Huỳnh Nhật Hoà, Nguyễn Vũ Hoàng Long

# TABLE OF CONTENTS

# PROBLEM

**Context:**

- Urbanization and Traffic Congestion
- Smart Cities: real-time data collection and traffic management.

**Goal:**

- counting traffic volume

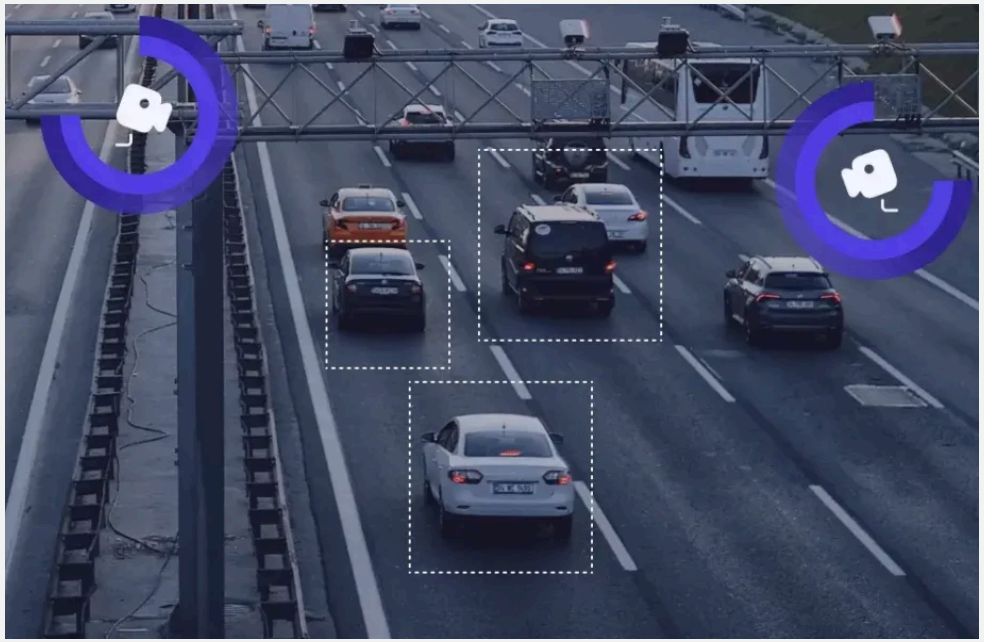
# PROBLEM

**Input:**

- labeled dataset
- video

**Output:**

- a number of vehicles



# DATASET

**Training:**

Vehicle Counting AIC HCMC 2020

**Experiment:**

AI Challenge HCMC 2020 Vehicle Counting data

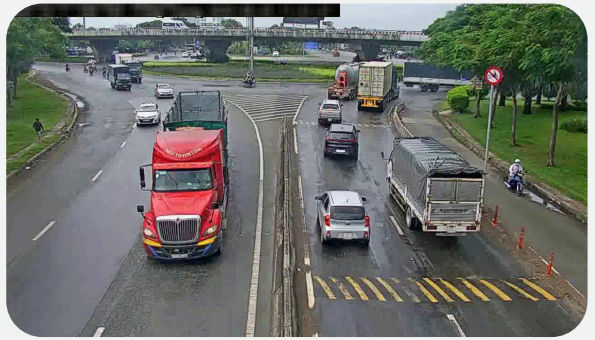
# DATASET

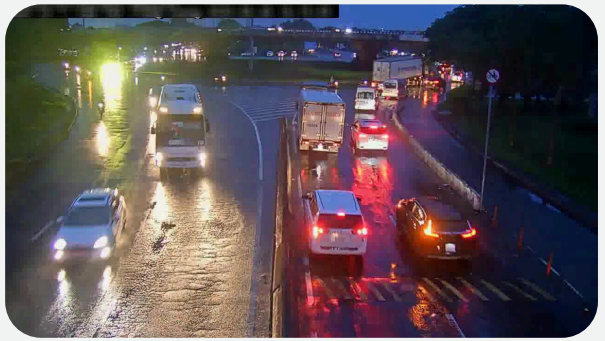
**Training and testing:**

- train: 20956
- test: 5239

**Classes:**

- car
- bus
- truck
- motorcycle





# DATASET

**Experiment**

- 10 videos, each one lasts 10 mins

# METHOD

1. Set up counting zones
2. Detection + Tracking
3. Count vehicles inside these zones

# METHOD

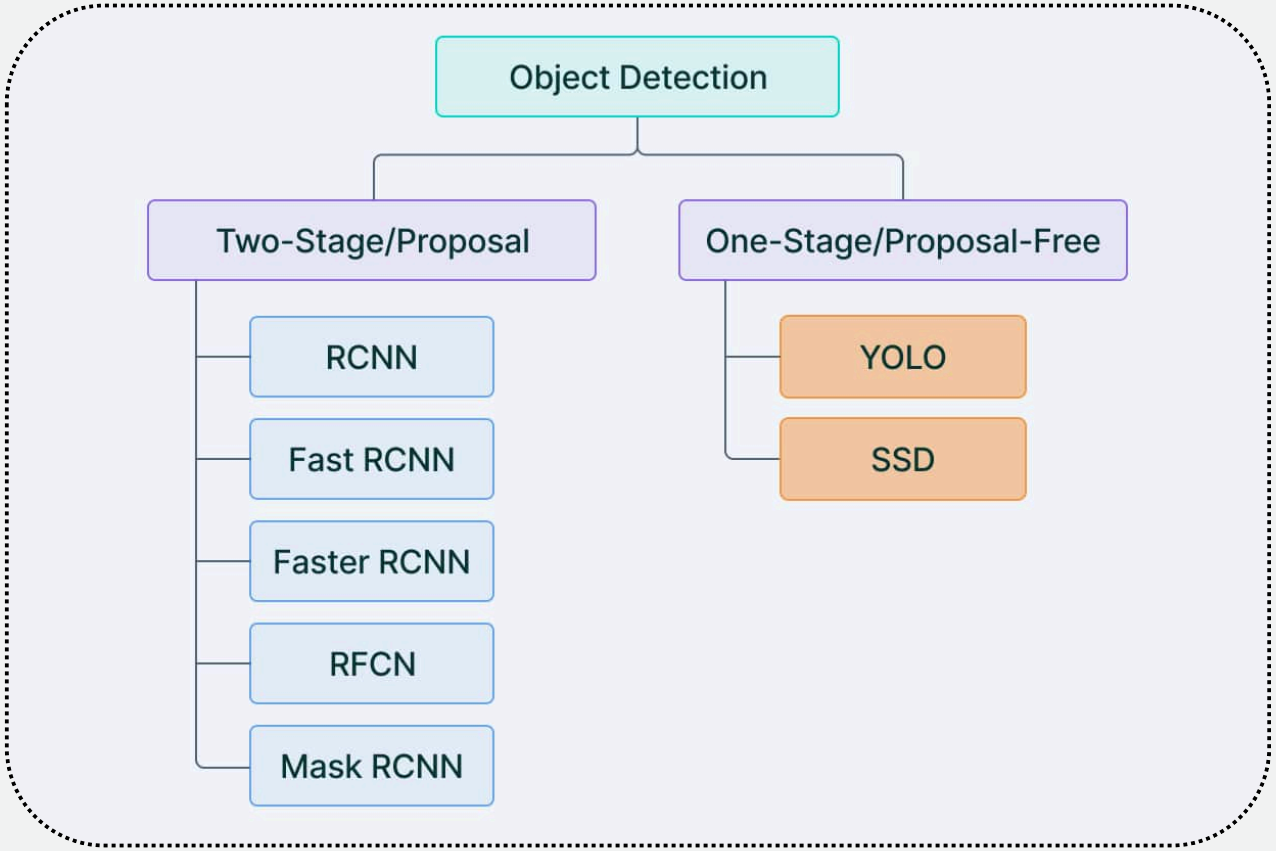
detector

**Faster RCNN**
**YOLOv8**
**SSD**

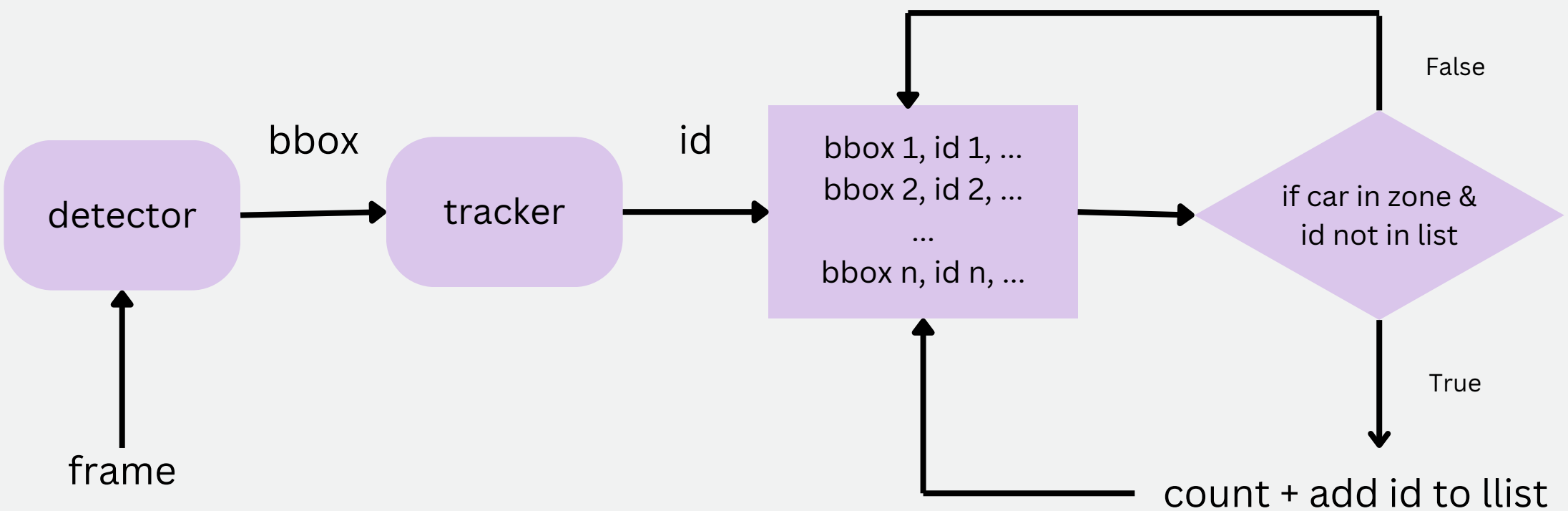
tracker

**ByteTrack**
**DeepSort**

# METHOD - DETECTOR



# METHOD



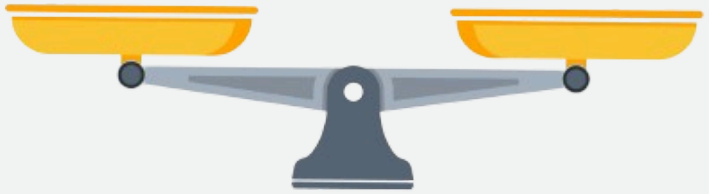

list:  include counted id (initialize an empty list)

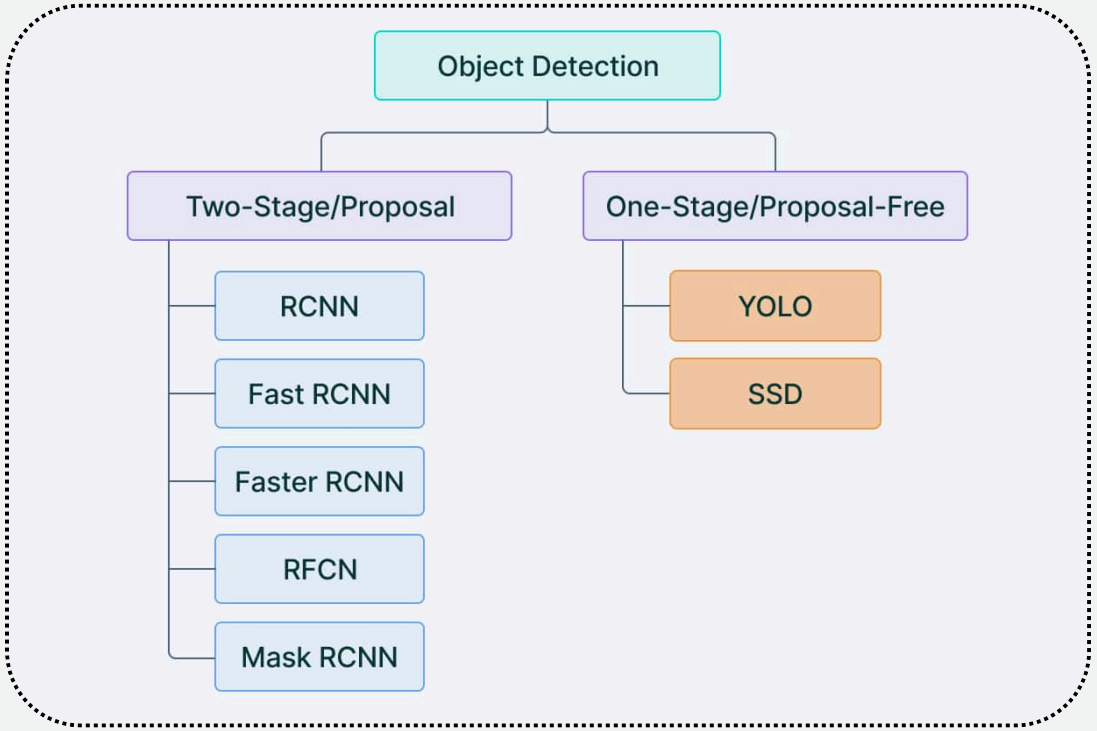
# METHOD - DETECTOR

**Requirement**

- **Real-time inference**

One stage or Two stage





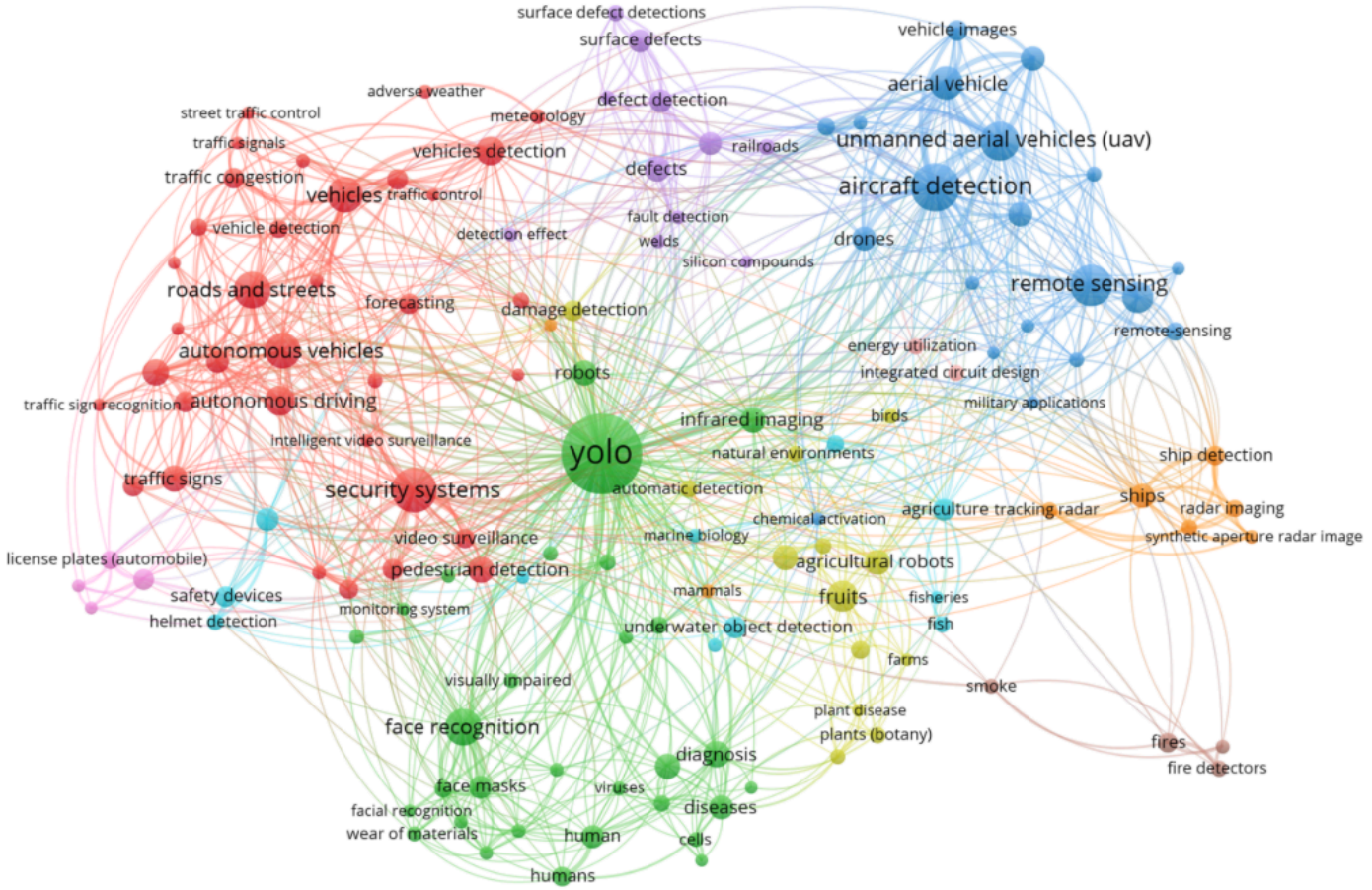
# METHOD - YOLO


**Models**

- YOLOv3
- YOLOv4
- YOLOv5
- YOLOv6
- YOLOv7
- YOLOv8
- YOLOv9
- YOLOv10

source: [ultralytics doc]

# METHOD - YOLO




source: [paper]

# METHOD - YOLO

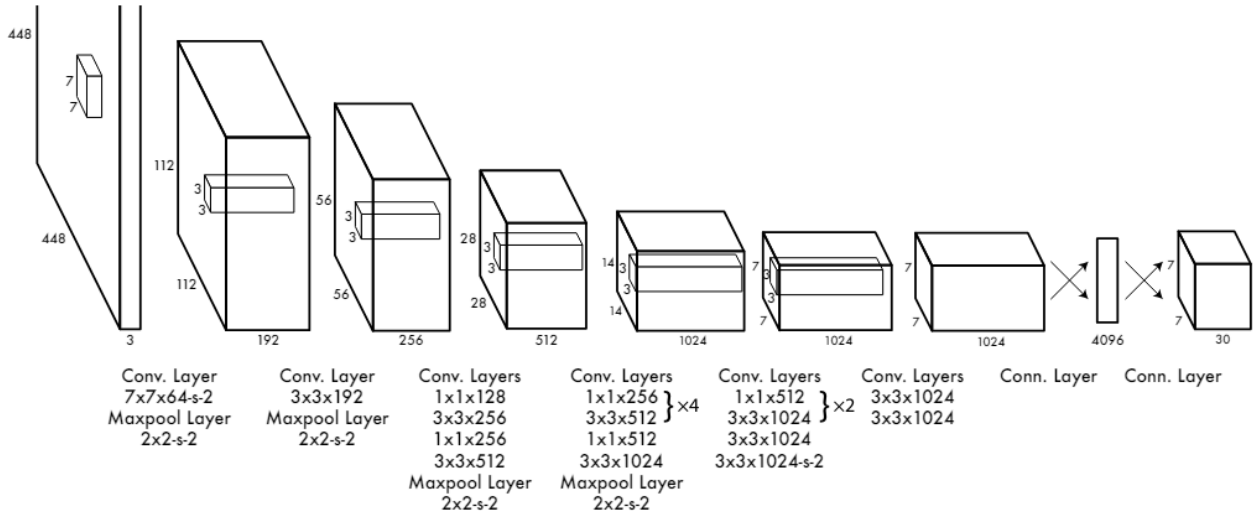



**Figure 3: The Architecture.** Our detection network has 24 convolutional layers followed by 2 fully connected layers. Alternating $1 \times 1$ convolutional layers reduce the features space from preceding layers. We pretrain the convolutional layers on the ImageNet classification task at half the resolution ($224 \times 224$ input image) and then double the resolution for detection.

YOLO stands for “You Only Look Once”

source: paper

# METHOD - YOLO

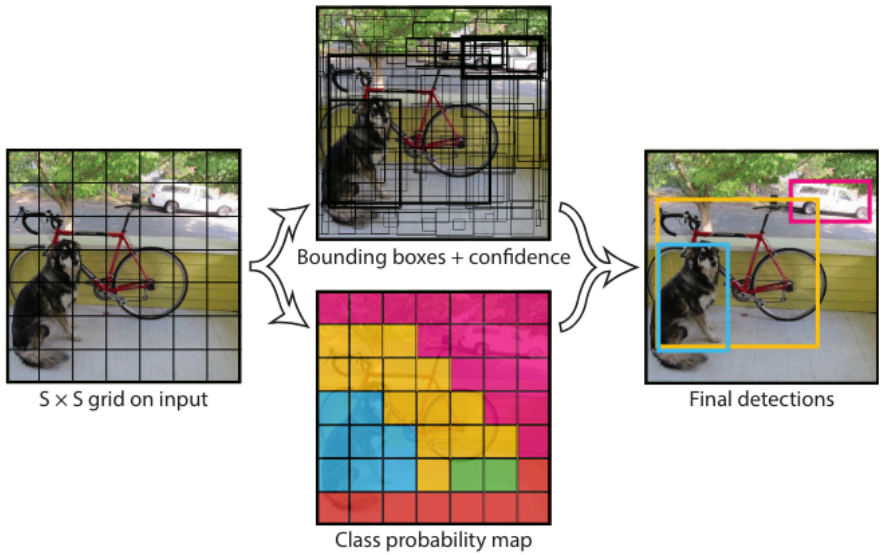
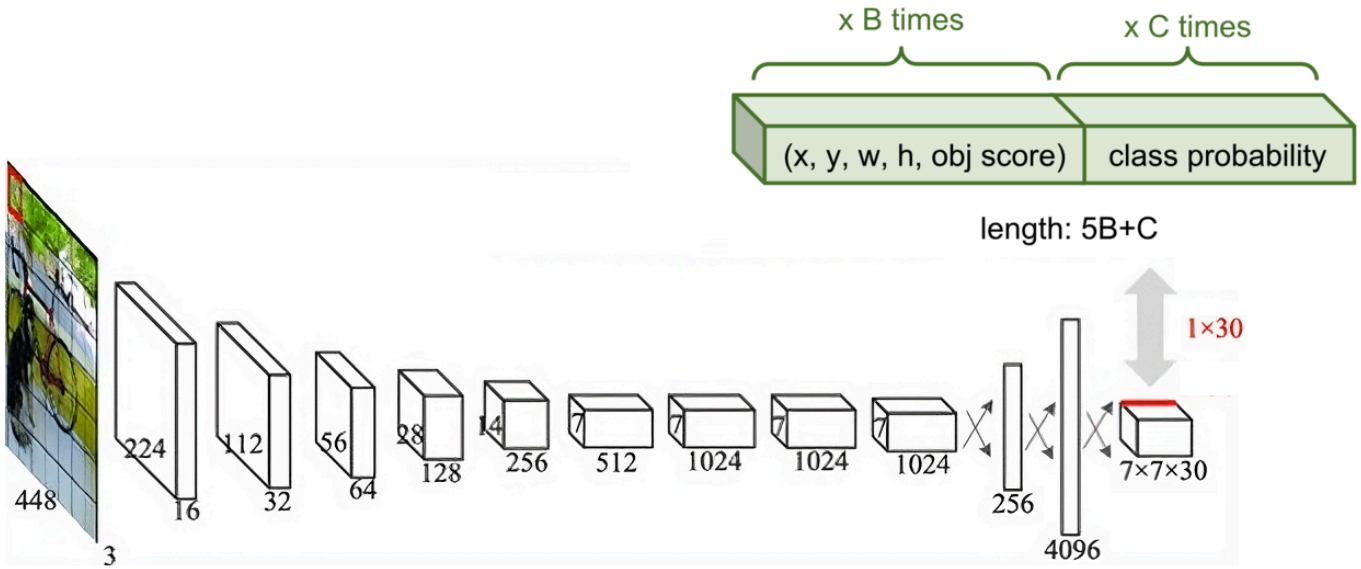



**Figure 2: The Model.** Our system models detection as a regression problem. It divides the image into an $S \times S$ grid and for each grid cell predicts $B$ bounding boxes, confidence for those boxes, and $C$ class probabilities. These predictions are encoded as an $S \times S \times (B * 5 + C)$ tensor.

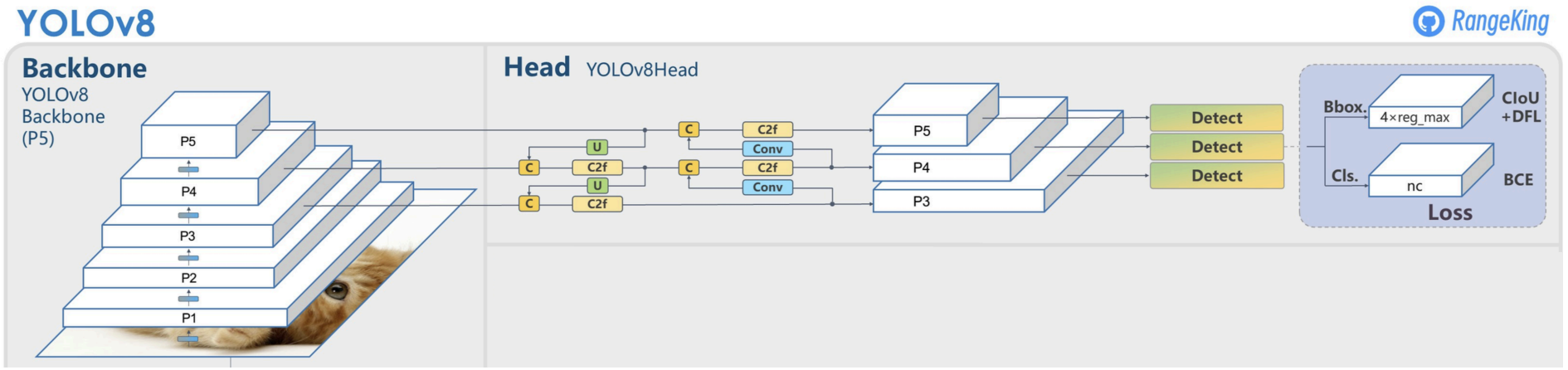



source: [paper]

# METHOD - YOLOV8

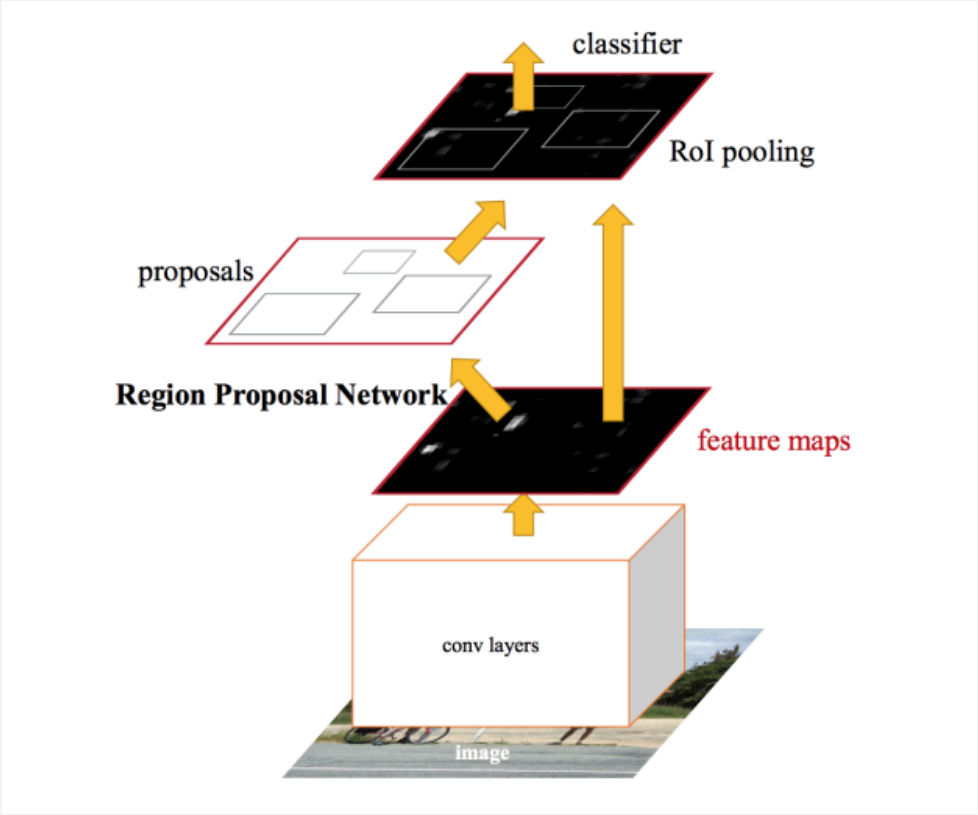
**Primary improments**

- Anchor-free
- Feature Pyramid Network (FPN)
- Soft-NMS




source: [paper]

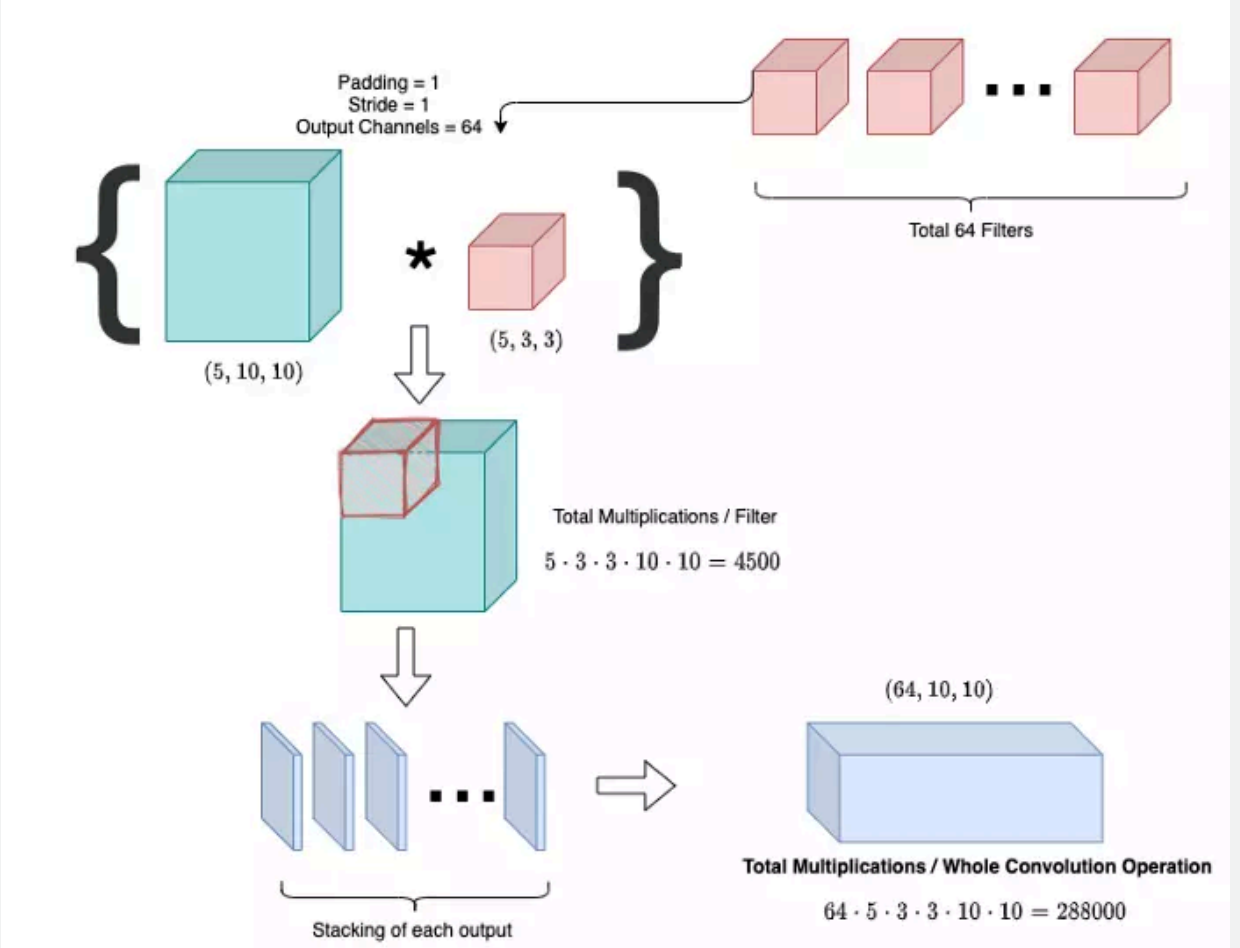
# FASTER RCNN


classifier
RoI pooling
proposals
Region Proposal Network
feature maps
conv layers
image

# SSD_MOBILENET

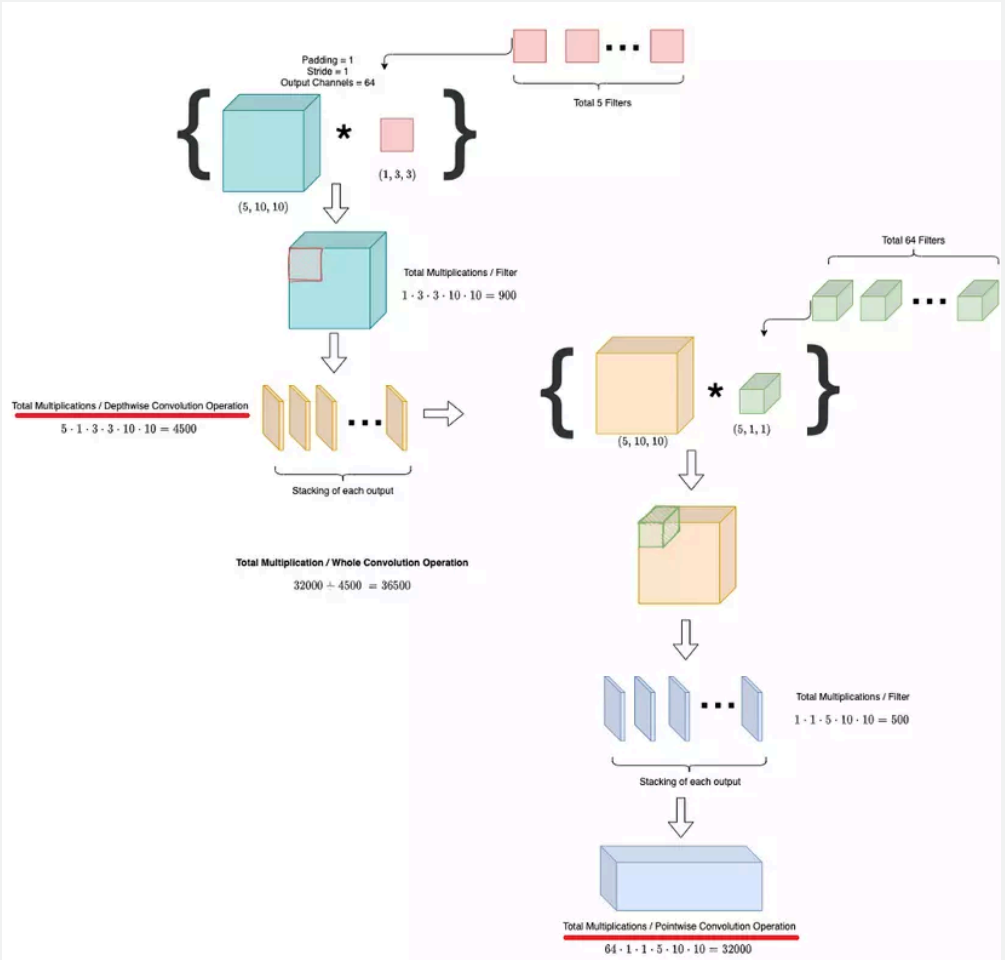

source: paper

# SSD_MOBILENET




source: paper

# SSD_MOBILENET




source: paper

# SSD_MOBILENET

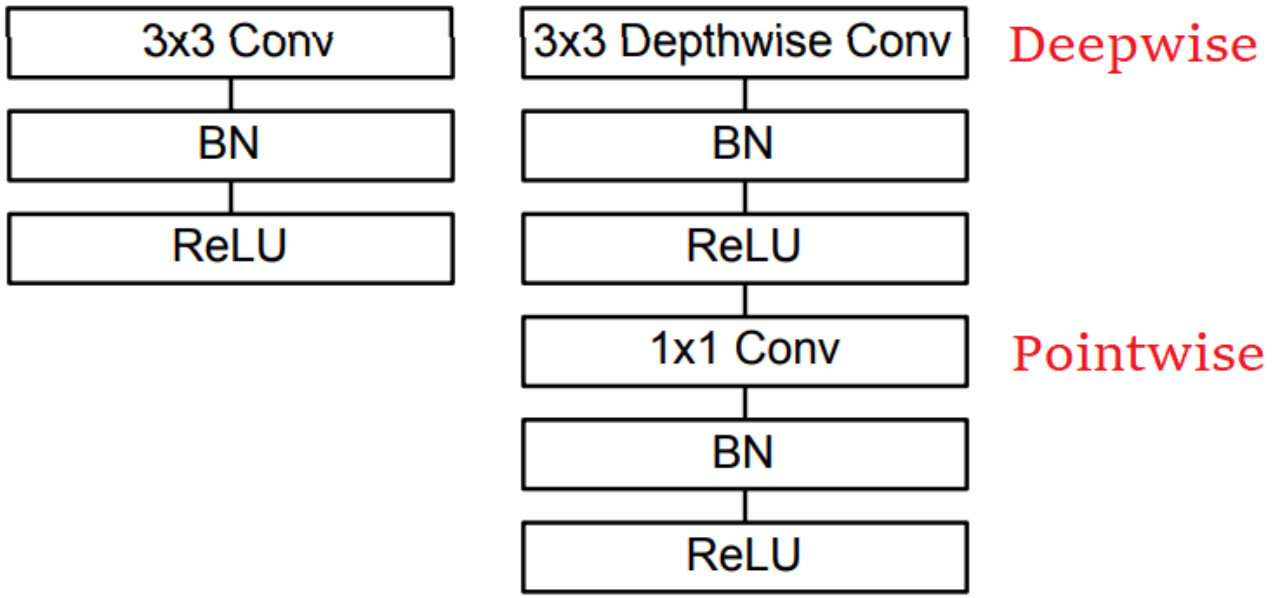
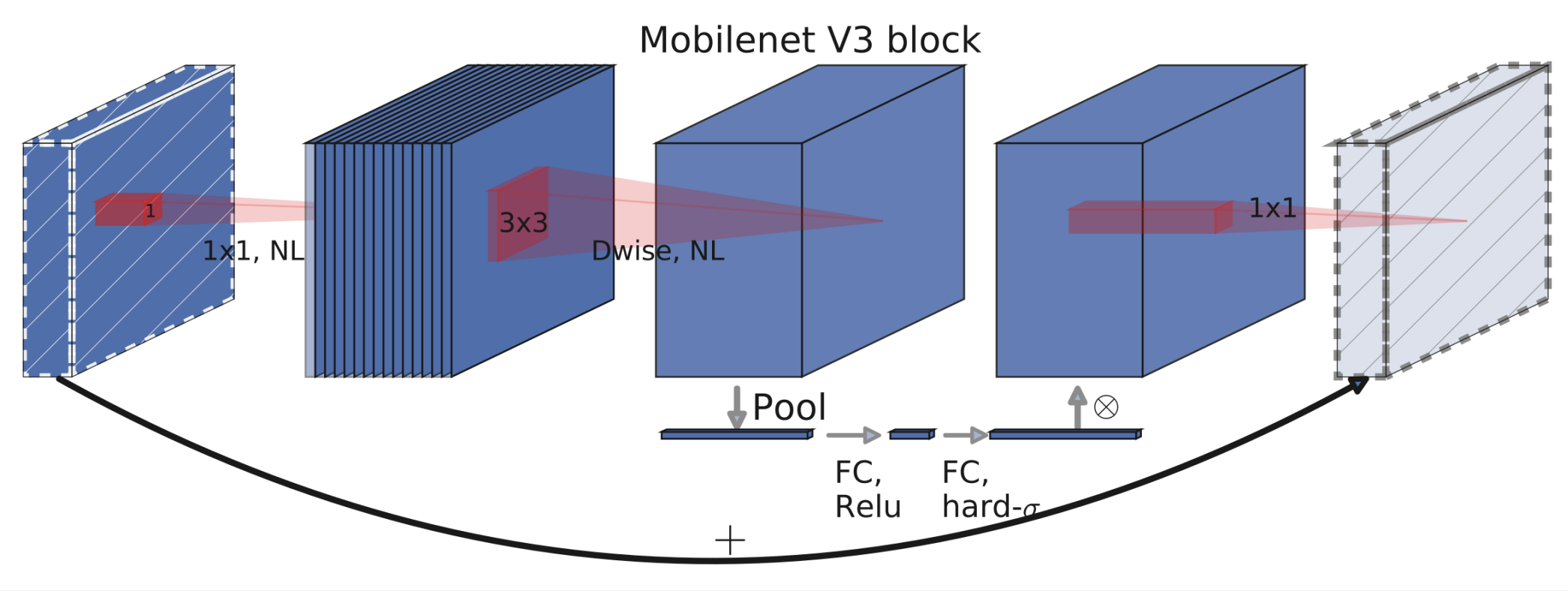



Figure 3. Left: Standard convolutional layer with batchnorm and ReLU. Right: Depthwise Separable convolutions with Depthwise and Pointwise layers followed by batchnorm and ReLU.

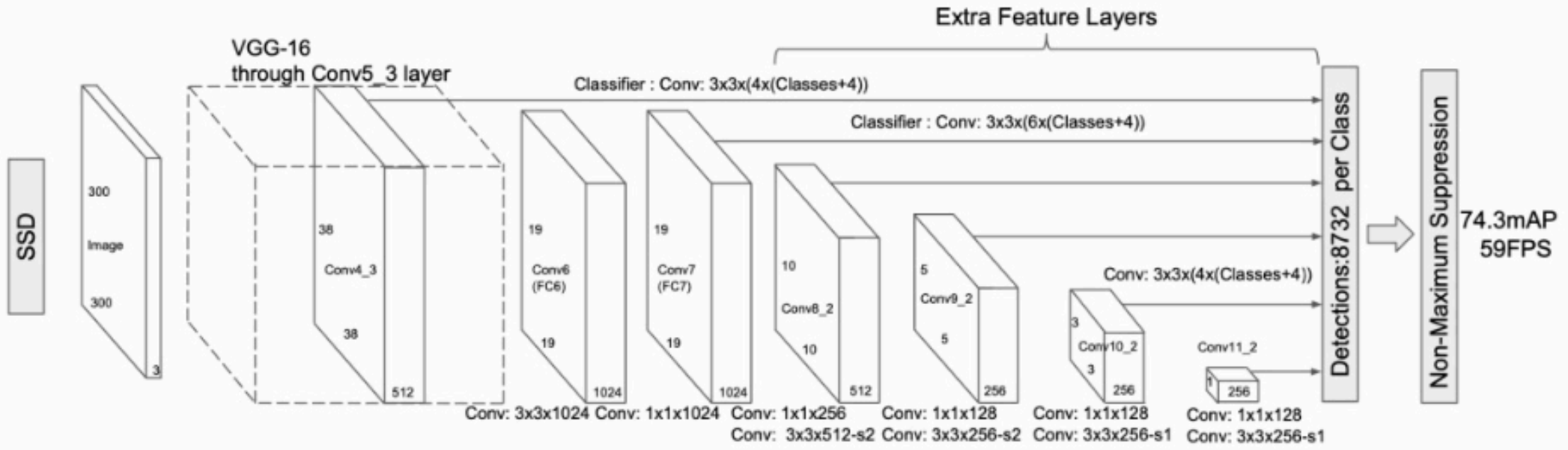
source: paper

# SSD_MOBILENET



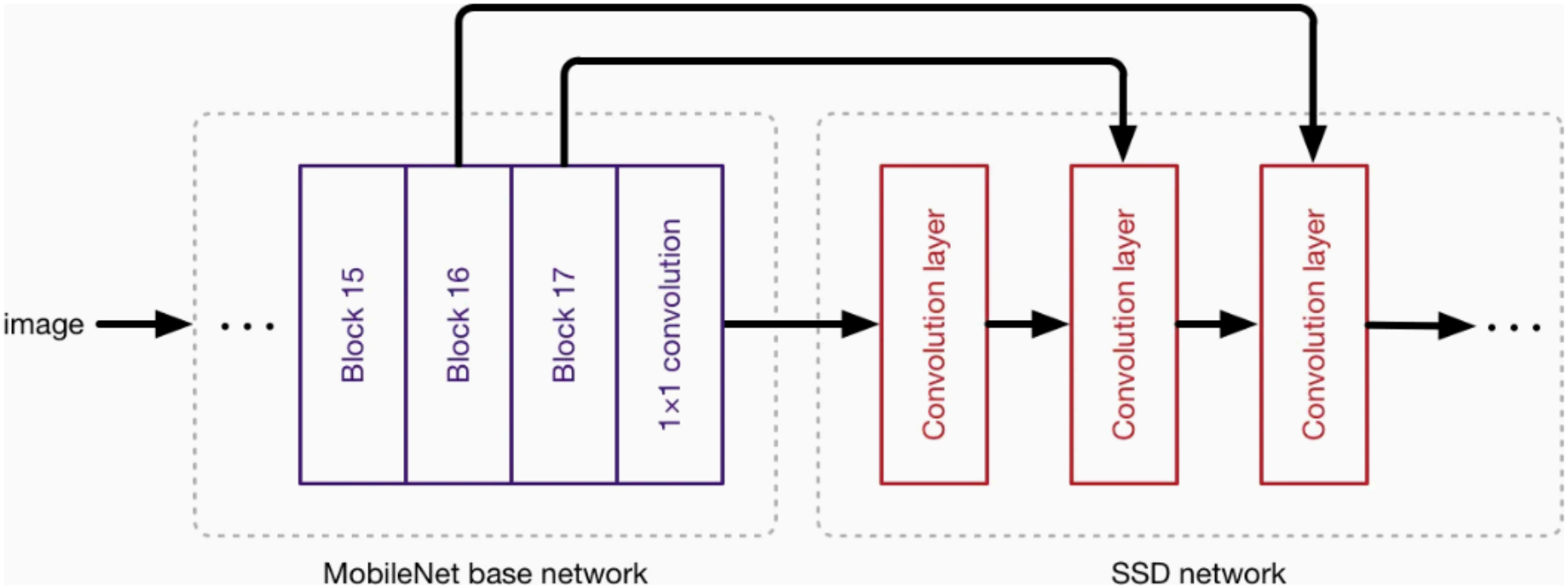

source: paper

# SSD_MOBILENET




source: paper

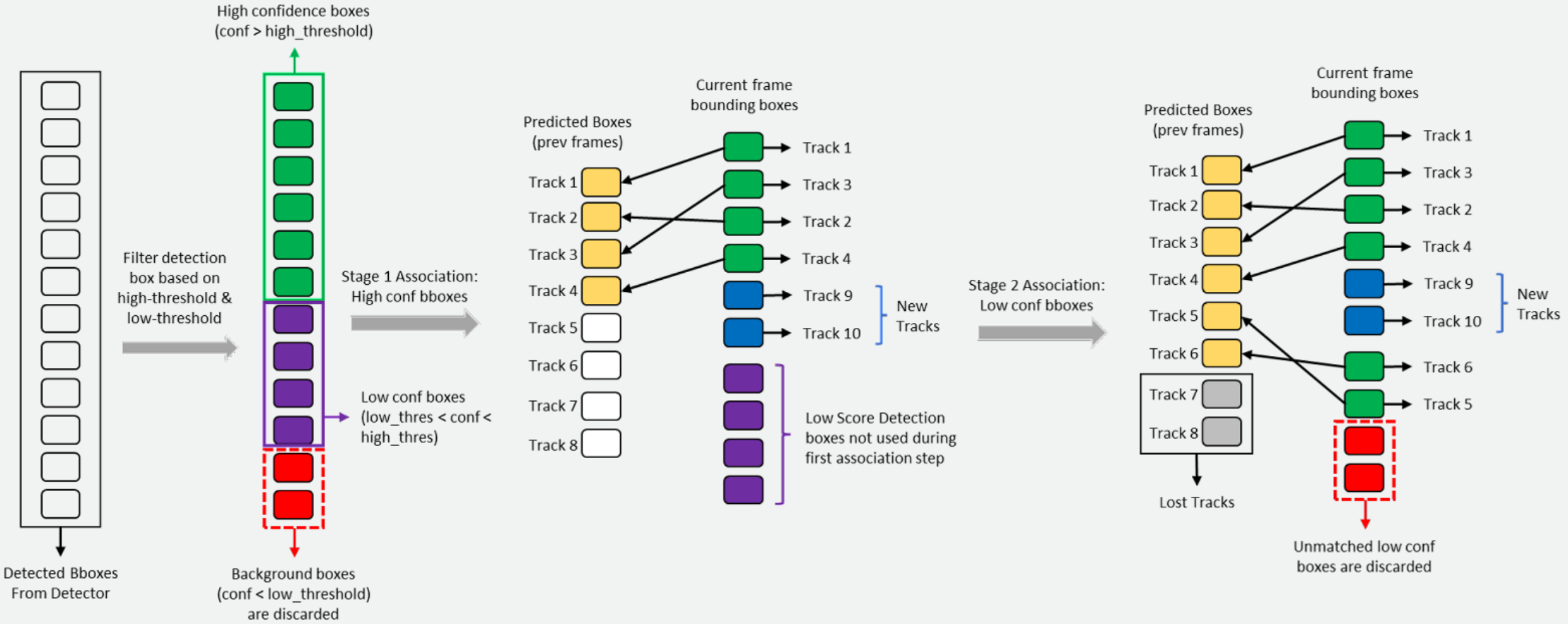
# SSD_MOBILENET




MobileNet SSD overview [7]

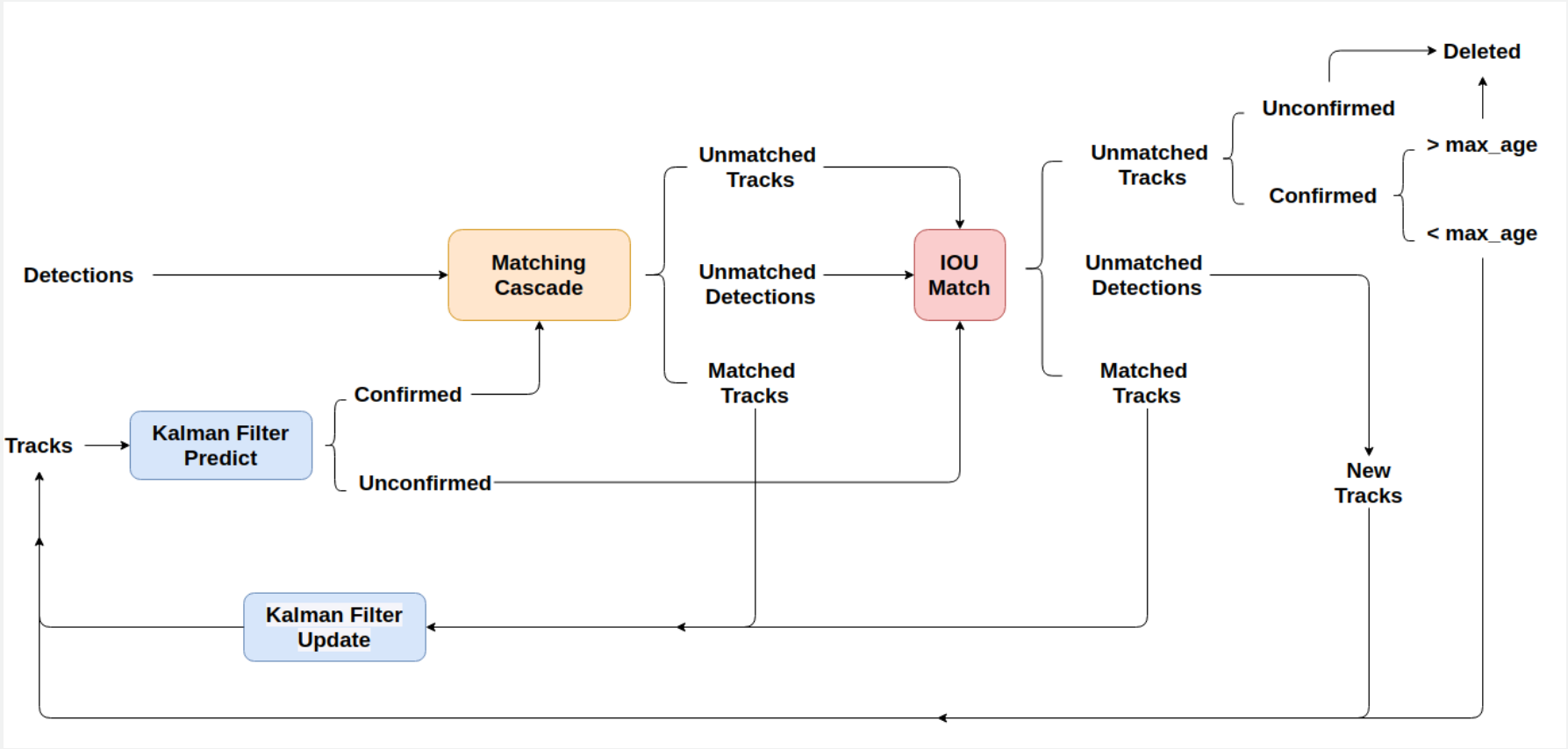
source: paper

# METHOD - BYTETRACK




source: [blog 1](), [blog 2](), [paper]()

# METHOD - DEEPSORT




[Paper]

# EVALUATION

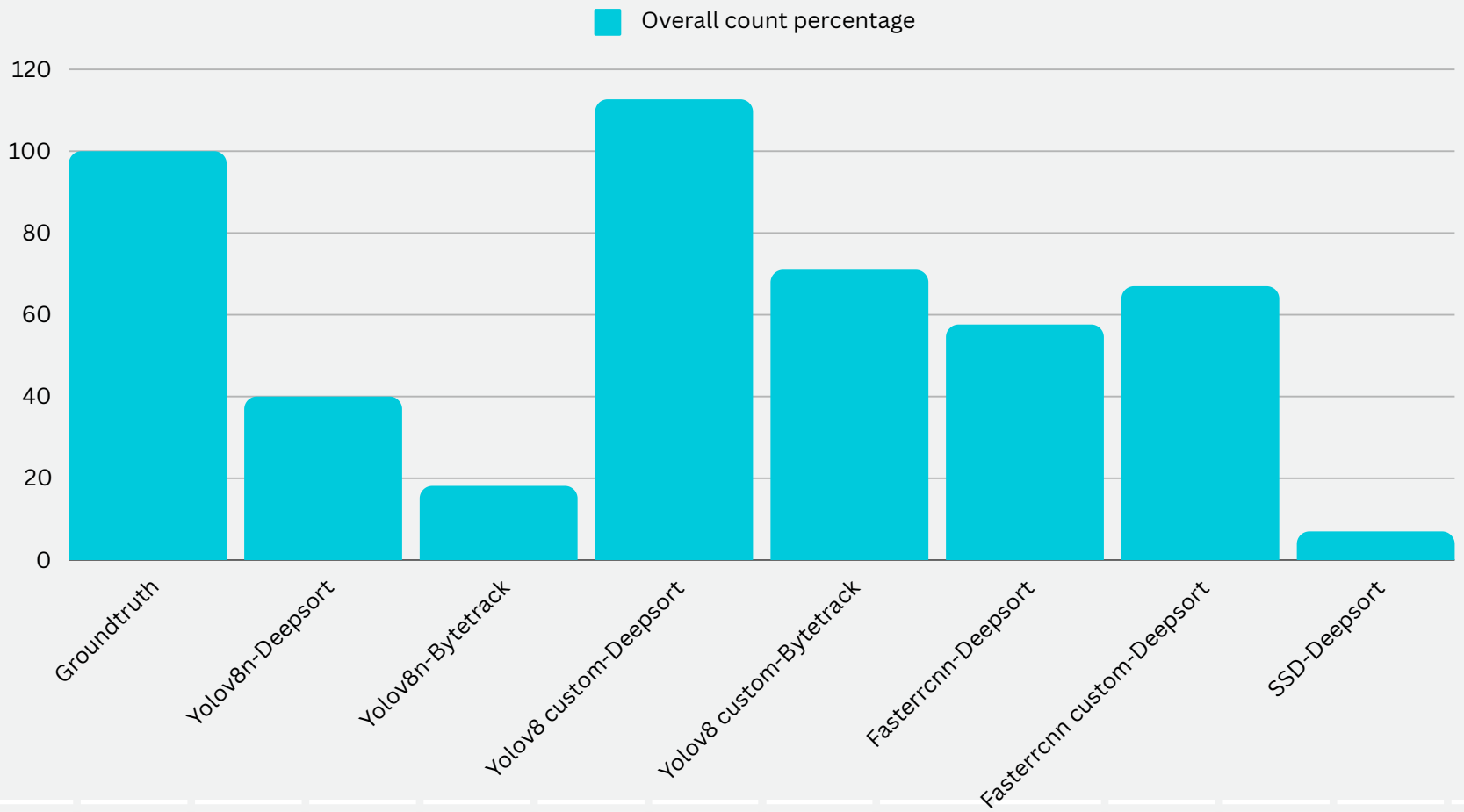
**Metric: Mean absolute error**

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i|$$

# EVALUATION

| Detector | MAE |
| --- | --- |
| YOLOv8n | 5.828678376043766 |
| YOLO Custom | 1.8070832133602073 |
| Faster RCNN | 3.4189461560610424 |
| Faster RCNN custom | 5.242441693060754 |
| SSD | 6.367405701122949 |

# COMPARISION



# COMPARISION