GROUP 4: KAIRE CAMPOMANES, DALLAS FOSTER, ELIJAH GHAYA, HISHAM MACARAYA, KATHERINE STANTON

OBJECT DETECTION

CHEAT SHEET

KEY CONCEPTS:

Object Detection **- Identifies and locates objects in an image.**

Bounding Boxes **- Rectangles around detected objects.**

Annotations **- Labels assigned to object in an image.**

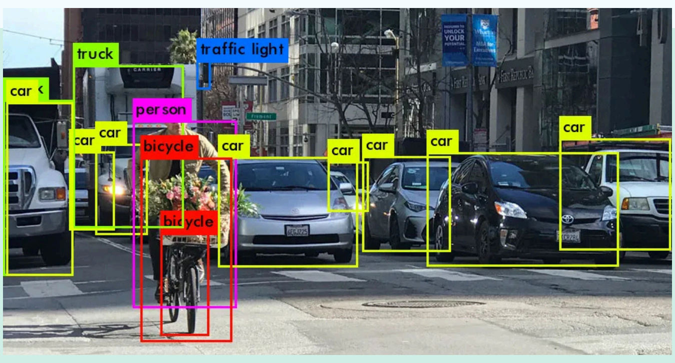
Confidence Score **- Indicates how sure the model is about the detection.**

IoU (Intersection over Union) **- Measures the overlap between predicted and truth boxes.**

$$IoU = \frac{(\text{Area of Intersection})}{(\text{Area of Union})} = \frac{|A \cap B|}{|A \cup B|}$$

STEPS IN AN OBJECT DETECTION TASK:

1. Data Collection & Annotation: Gather and label data.
2. Preprocessing: Resize and normalize images, split into train/ test sets.
3. Model Selection: Choose from R-CNN, SSD, or YOLO.
4. Training: Train the model using labeled data.
5. Evaluation: Use IoU and confidence scores for performance.
6. Fine-tuning: Adjust hyperparameters, use a different model if necessary.
7. Deployment: Applies the trained model.



COMMON ALGORITHMS

- R-CNN: Proposes regions and classifies them.
- Fast R-CNN: Processes entire image, using ROI pooling for effienciency.
- Faster R-CNN: Uses Region Proposal Networks for better speeds.
- SSD or Single Shot Multibox Detector: Detects objects in a single shot.
- YOLO (You Only Look Once): Processes entire image with one network pass.

TOOLS & LIBRARIES

- *TensorFlow & Keras*

Installation:

```
pip install tensorflow keras
```

Example:

```
model = tf.keras.applications.VGG16()
```

Docs: tensorflow.org, keras.io

- *Open CV*

Installation:

```
pip install opencv-python
```

Example:

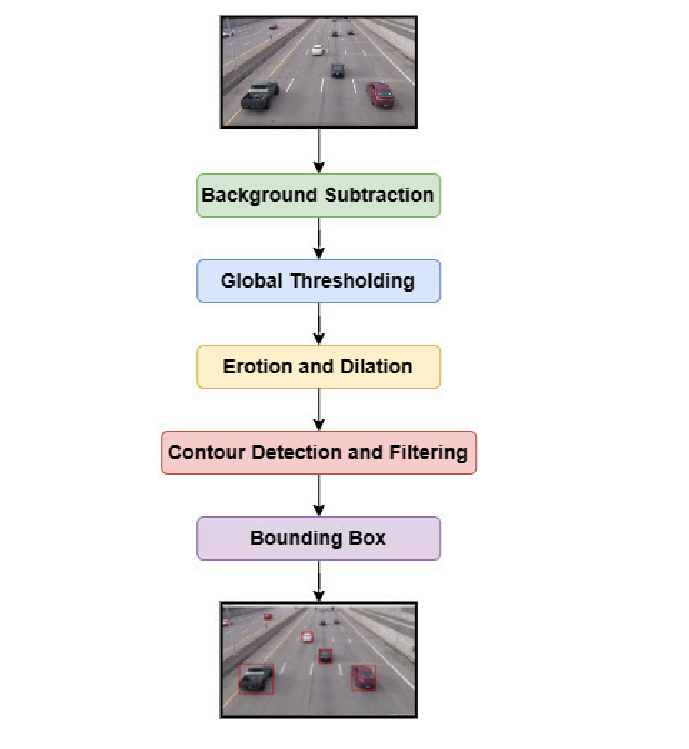
```
img = cv.2.imread(‘image.jpg’)
```

Docs: docs.opencv.org

COMMON CHALLENGES & TIPS

- Class Imbalance: Gives a balanced dataset for classes.
- Small Object Detection: Consider using Faster R-CNN or adjust YOLO layers.
- Overfitting: Use data augmentation, dropout, or early stopping.
- Slow Inference: For real-time tasks, consider using SSD or YOLO for faster speed.

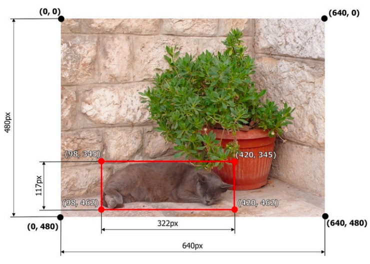
Moving Object Detection Pipeline using OpenCV



EVALUATION METRICS

- Intersection over Union (IoU)
- Precision
- Recall
- F-1 Score
- Mean Average Precision (mAP)

How different formats represent coordinates of a bounding box



pascal_voc
[x_min, y_min, x_max, y_max] ⟶ [98, 345, 420, 462]

albumentations
normalized [x_min, y_min, x_max, y_max] ⟶ [0.153125, 0.71875, 0.65625, 0.9625]

coco
[x_min, y_min, width, height] ⟶ [98, 345, 322, 117]

yolo
normalized [x_center, y_center, width, height] ⟶ [0.4046875, 0.8614583, 0.503125, 0.24375]

ADDITIONAL RESOURCES

- “Deep Learning for Computer Vision” by Adrian Rosebrock
- Papers with Code: Object Detection task
- Coursera: Object Detection with TensorFlow
- Towards Data Science: Various tutorials on object detection.