**A08 Binary Brains Cheat sheet**

Ambalika, Favour, Joseph, Misty

**Installation Instructions:**  **Basic Usage Examples:**

|  |  |
| --- | --- |
| !pip install tensorflow | TensorFlow: Import tensorflow as tf |
| !pip install keras | Keras: import keras |
| !pip install opencv-python | OpenCV: import cv2 |

**Object Detection Task Steps:**

|  |  |
| --- | --- |
| Data Collection: Obtain labeled images with annotations. | Preprocessing: Resize, normalize images. |
| Model Selection: Choose a suitable detection model. | Training: Fine-tune model on labeled dataset. |
| Evaluation: Measure performance using metrics like IoU | Inference: Apply trained model for object detection. |

**Common Challenges and Troubleshooting:**

|  |  |
| --- | --- |
| Overfitting: Regularization, data augmentation. | Data Imbalance: Class weighting, augmentation techniques. |
| Performance: Hyperparameter optimization, model architect | Runtime Efficiency:Model quantization, hardware acceleration |
| Poor Localization: Experiment with different network architectures or adjust hyperparameters. |  |

**Common Applications of Object Detection:**

|  |  |
| --- | --- |
| Self-driving cars: Identify pedestrians, vehicles and traffic signs. | Facial Recognition: Detect faces for identification. |
| Medical Scan (X-ray, CT scan): Detect anomalies like tumors. | Inventory Management: Automatic stock checks & reordering. |
| Robotics: Accurate perception for precision intensive tasks |  |

**Concepts:**

|  |  |
| --- | --- |
| Bounding boxes: Rectangular boxes indicating object location. | Annotations: Labels assigned to bounding boxes. |
| Confidence scores: Model’s certainty about detection. | Intersection over Union (IoU): Measure of overlap between boxes. |

**Common Algorithms:**

|  |  |
| --- | --- |
| R-CNN: Regions with CNN features | Fast R-CNN: Improved speed over typical R-CNN |
| Faster R-CNN: Improved speed & accuracy. (High precision) | SSD (Single Shot MultiBox Detector): 1 shot detection. |
| YOLO (You Only Look Once): Good for real-time applications. |  |

**Tools & Libraries:**

|  |  |
| --- | --- |
| TensorFlow: Open-source framework for machine learning. | Keras: High-level API simplifying development |
| OpenCV: Library for computer vision functions |  |

**Resources:**

|  |  |
| --- | --- |
| TensorFlow: [[API Documentation | TensorFlow v2.16.1](https://www.tensorflow.org/api_docs)] | Keras: [[Developer guides (keras.io)](https://keras.io/guides/)] |
| OpenCV: [[OpenCV: OpenCV modules](https://docs.opencv.org/4.x/index.html)] | R-CNN: [trainRCNNObjectDetector,](https://www.mathworks.com/help/vision/ref/trainrcnnobjectdetector.html) [rcnnObjectDetector,](https://www.mathworks.com/help/vision/ref/rcnnobjectdetector.html) |
| SSD: [trainSSDObjectDetector](https://www.mathworks.com/help/vision/ref/trainssdobjectdetector.html) | YOLO: [trainYOLOv3ObjectDetector](https://www.mathworks.com/help/vision/ref/trainyolov3objectdetector.html) |