TA-2 Task

Object detection on Jetson Orin Nano steps:

**1. Set Up Jetson Orin Nano**

* **Flash the OS**: Flash JetPack (NVIDIA's software for Jetson devices) on your Jetson Orin Nano using the SDK Manager on a host system (Ubuntu).
* **Initial Boot Setup**: Power the device, connect it to a monitor via HDMI, and complete the initial setup (Wi-Fi, system configurations).
* **Install Dependencies**: Ensure CUDA, cuDNN, TensorRT, and OpenCV are installed through JetPack.

**2. Install Required Libraries**

Use apt-get or pip to install the required libraries.

**3. Set Up Object Detection Framework**

Choose a framework for object detection. For Jetson Orin Nano, you can use TensorFlow, PyTorch, or NVIDIA's **DeepStream**.

**Using PyTorch and YOLOv5:**

* **Install PyTorch** (Jetson-specific wheel).

**Clone YOLOv5 Repo**

**Using NVIDIA DeepStream SDK:**

* **Install DeepStream**

*sudo apt-get install deepstream-6.1*

This provides pre-built models like YOLO, SSD, etc., and is optimized for Jetson devices.

**4. Download Pre-trained Models**

You can either train a model or use a pre-trained one.

* For **YOLOv5**, download the pre-trained model:

*python3 yolov5/models/download\_weights.py*

**5. Train a Custom Object Detection Model (optional)**

* To train YOLOv5 on your dataset:
  1. **Prepare the Dataset**: Format it in YOLO format (images and label files).
  2. **Modify YOLO Config**: Adjust the config file for your number of classes.
  3. **Start Training**

*python3 train.py --img 640 --batch 16 --epochs 100 --data data.yaml --weights yolov5s.pt*

**6. Optimize the Model for Jetson**

NVIDIA Jetson uses TensorRT for acceleration. To convert your model:

* **Export YOLOv5 to ONNX**:

*python3 export.py --weights yolov5s.pt --img 640 --batch 1 --device 0 --simplify --opset 12*

* **Convert ONNX to TensorRT**

*trtexec --onnx=yolov5s.onnx --saveEngine=yolov5s.engine*

**7. Deploy the Model**

* **For YOLOv5**: Use the optimized TensorRT engine

*python3 detect.py --weights yolov5s.engine --source 0*

**8. Test and Fine-Tune**

Run inference on your test data or real-time video feed to ensure the object detection works efficiently on the Jetson Orin Nano. Optimize parameters like batch size, resolution, or hardware acceleration features if needed.