**Algorithm for YOLOv4 for object detection:**

**1. Dataset Preparation:**

Collect a dataset of tomato images and annotate them with bounding boxes.

Ensure annotations are in the YOLO format (text files containing class index and bounding box coordinates).

**2. Download YOLO Model:**

Download the pre-trained weights for YOLOv3 or YOLOv4 from the official YOLO website or GitHub.

**3. Configuration:**

Modify the YOLO configuration file (e.g., yolov3.cfg or yolov4.cfg) to set the number of classes to 1 (for tomatoes).

**4. Training:**

Use the YOLO model's architecture and load the pre-trained weights.

Replace the output layer with a new layer for one class.

Fine-tune the model on the tomato dataset.

**5.Inference:**

Load the trained model.

Input an image to the model for inference.

**6. Post-processing:**

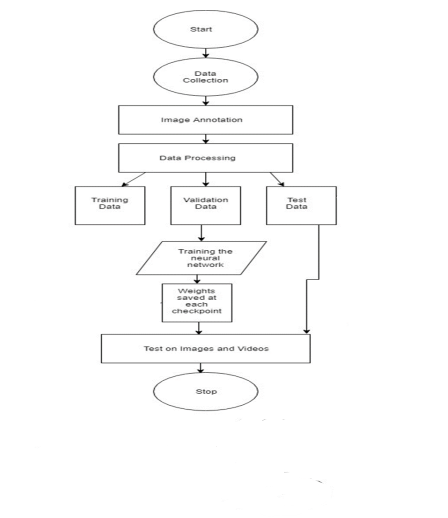
Filter out predictions with low confidence scores (e.g., below 0.5).

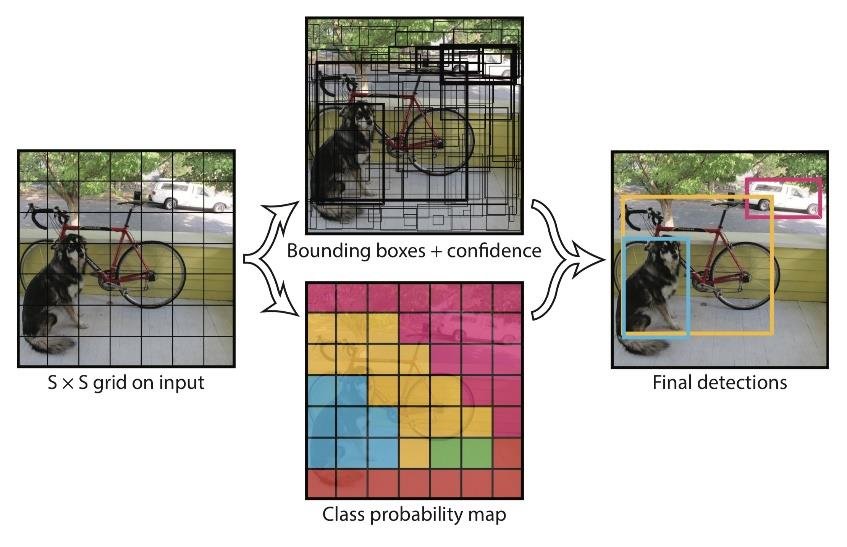
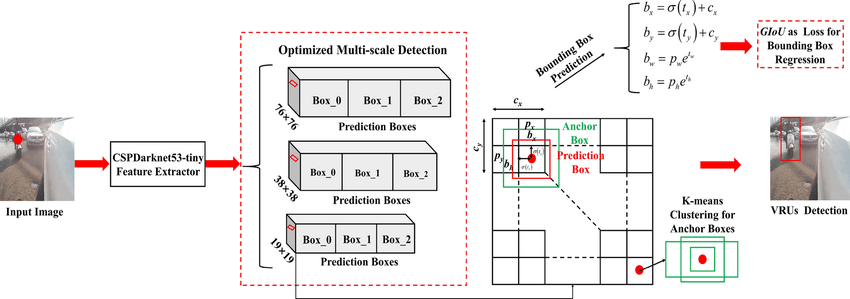
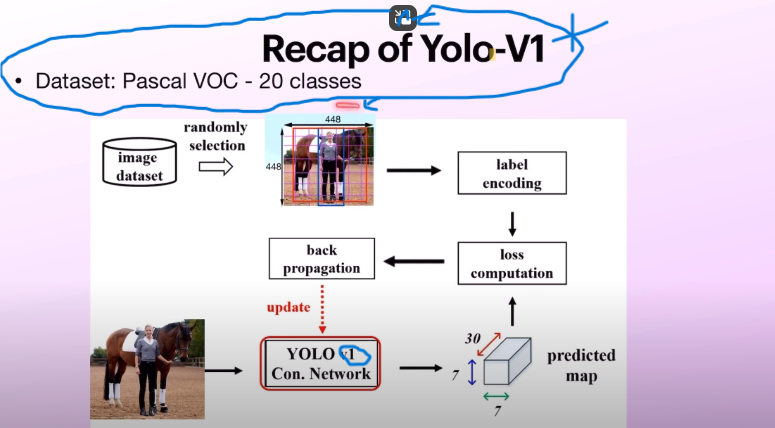
Extract bounding box coordinates for the remaining predictions.

**7. Visualization:**

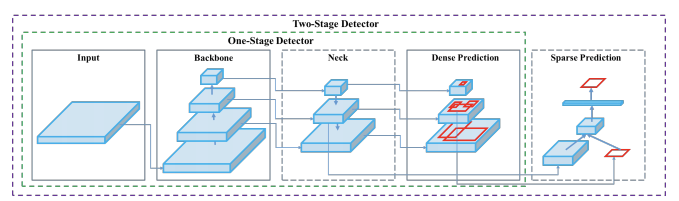
Draw bounding boxes on the original image to visualize tomato detections.

**Flowchart: Image detection**





**Architecture**

****