YOLOv3 Realtime Sign Detection & Classification

Lewis Collum

Contents

1 Results 1

2 How YOLOv3 was Implemented 2

3 Easy. Detect Signs Then Classify. 2

1 Results

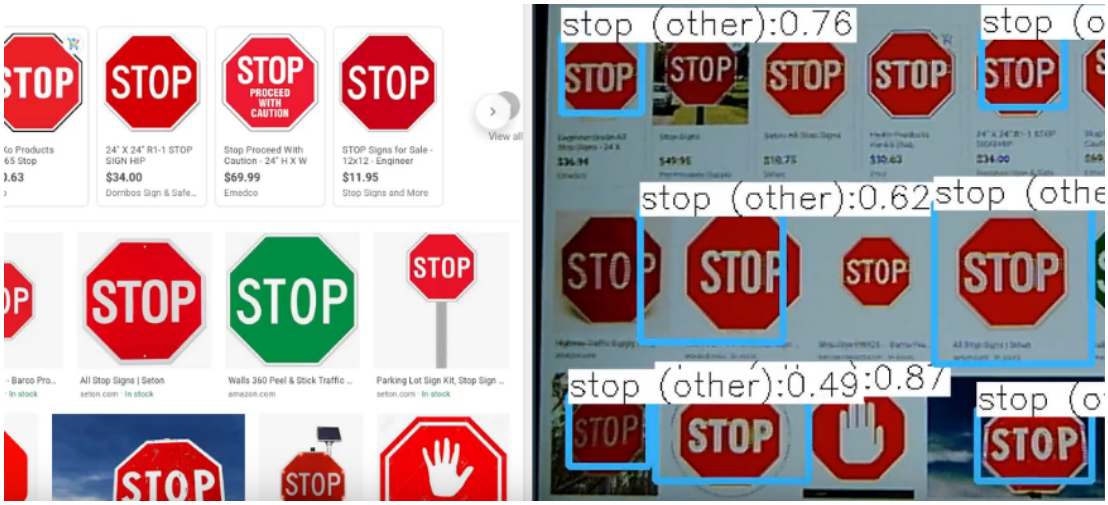
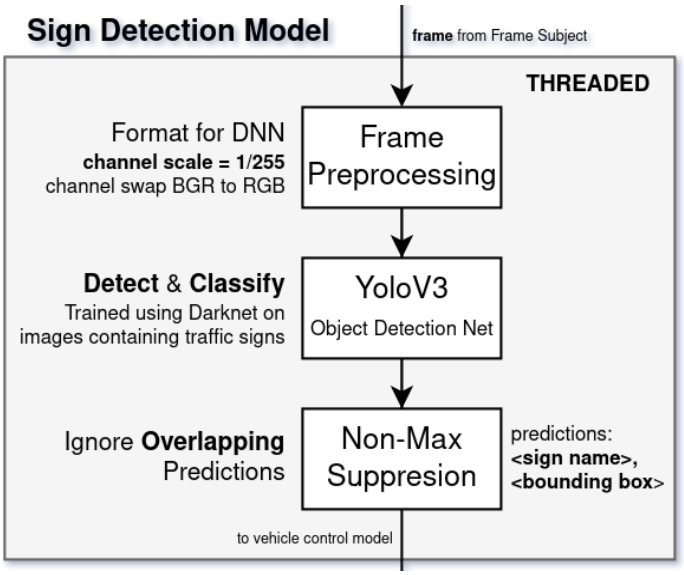


Fig. 1. Stop sign images on a monitor (left), and the processed video feed (right).

2 How YOLOv3 was Implemented



3 Easy. Detect Signs Then Classify.

No. This is how object detection[1] algorithms started (e.g. R-CNN), but they can be too slow for real-time (and embedded) object detection. Those looking for speed, use algorithms that extract classified objects from the frame in a single pass, as opposed to two passes. YOLOv3 is an algorithm that detects in a single pass.

There are multiple versions of YOLOv3. We are using YOLOv3-tiny-prn, which has the highest frames per second (FPS) compared to other commonly used algorithms (see figure 2). The sacrifice to speed is accuracy, as YOLOv3-tiny-prn borders around 35% average precision. Since we implemented sign detection on a Raspberry Pi as a proof-of-concept, this accuracy is acceptable.

1. Classification is not detection. Objects first need to be detected before they can be classified. But, for briefness, we say "object detection" when we really mean "object detection and classification."

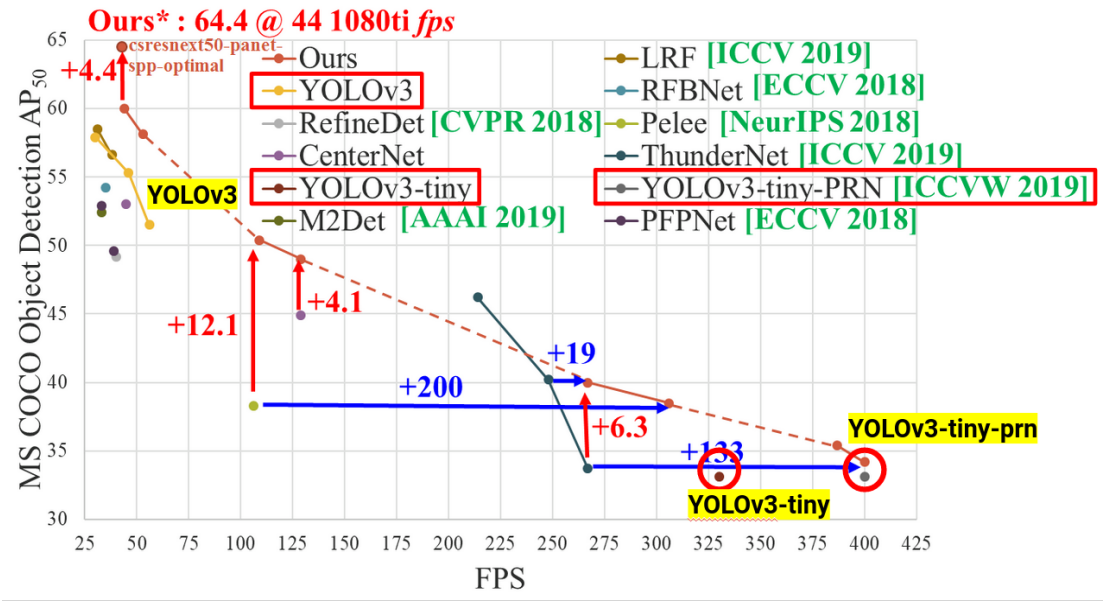


Fig. 2. YOLOv3-tiny-prn has the highest FPS, with an acceptable 35% average precision.