Real-time object detection and tracking

Samuele Sermisoni s212060, Ruixin Chen s210312, Xianhao Liu s202724, Carlos Barragan s211821

Introduction

A problem with deep learning algorithms is that they require high-performance hardware (such as GPUs) to run in real time. The purpose of our project is therefore to develop an algorithm capable of performing real-time object detection and tracking on a Jetson-Nano, a low-power single board computer. In particular, the goal is to identify cans of beer and coke and trace their movement within the static camera's field of view. Such an algorithm could be used inside a factory to track products moving on conveyor belts, in order to preventively identify errors that would lead to the blocking of the line.

Key Points

- We created a new dataset using a camera to record cans of beer and coke rolling on the floor and then we labelled the images with bounding boxes using CVAT.
- We analyzed our dataset by identifying and solving weaknesses through data augmentation techniques such as:
  - roto-translation of the images
  - addition of Gaussian noise
  - balance of the number of images
  - contrast and brightness correction
- We build a dataloader to import the images and the bounding boxes both in VOC Pascal and COCO format
- We trained a Faster R-CNN net using different kind of backbone (ResNet50 and MobileNetV2) for feature extraction.
- We coded an accuracy algorithm to evaluate the quality of the trained net
- We trained the YOLOv5s net to make a comparison between the two different nets

Data Preparation

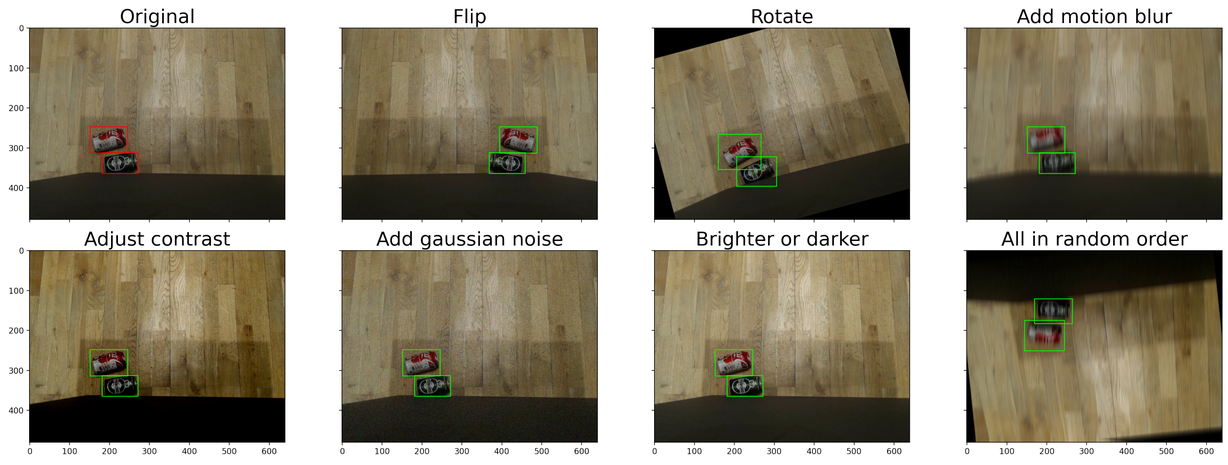

Figure 1: Data augmentation with different methods.

The overfitting problem arises when training a Faster R-CNN network with limited training data. Various methods can solve this, among which data augmentation is one of the most direct and efficient ways. By flipping, rotating, and shearing the image, the pose of the object can be manipulated. But rotating or shearing too much leads to the new bounding box covering over much incorrectly targeted space. A small angle can ensure the compactness of the bounding boxes while changing the direction of objects. Multiplication to all channels and contrast adjustment can simulate different illumination. Injecting the right amount of noise is also a fundamental tool for data augmentation. Last but not least, introducing motion blur imitates the object moving fast or under a low frame rate camera.

Object Detection -Faster R-CNN with MobileNetV2 Backbone

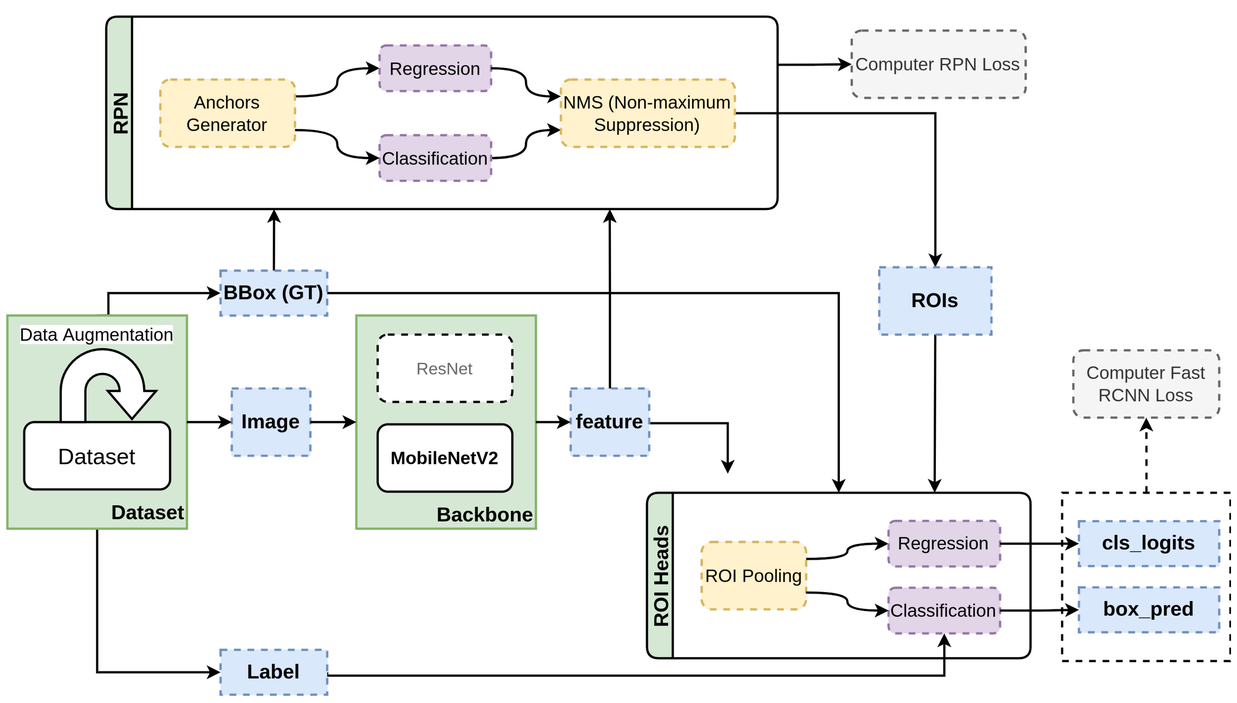

Figure 2: Faster R-CNN architecture.

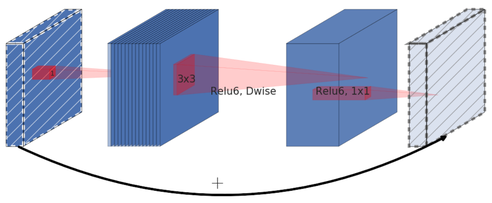

Figure 3: Inverted residual block[2].

MobileNetV2–deep nets on mobile devices

- Depthwise Separable Convolutions
- Linear Bottlenecks
- Inverted Residual Structure, see in Fig. 3

Object Detection -Yolo

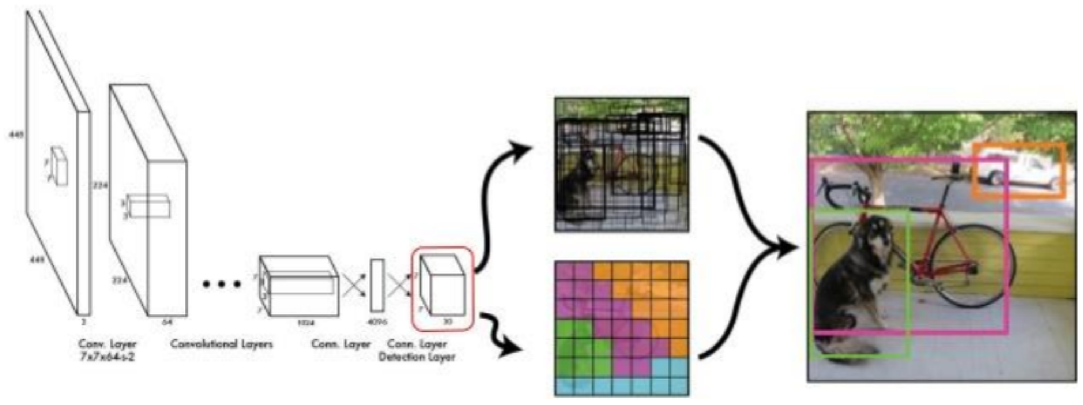

Figure 4: Yolo Architecture and Procedures

Yolo -You Only Look Once[1]

1. Split the image into cells.
2. Each cell predicts boxes and confidences(probability of the box contains an object)
3. Each cell also predicts a class probability.
4. Finally we do NMS and threshold detections

It's trained to do classification and bounding box regression simultaneously rather than by order. Thus with the simpler architecture, it runs faster than Faster R-CNN.

Object Tracking

Object tracking is the task of taking an initial set of object detections, creating a unique ID for each of the initial detections, and then tracking each of the objects as they move around each subsequent frames in a video, maintaining the ID assignment. We implemented so using a simple algorithm based on computing the Euclidean distance between the objects detected.

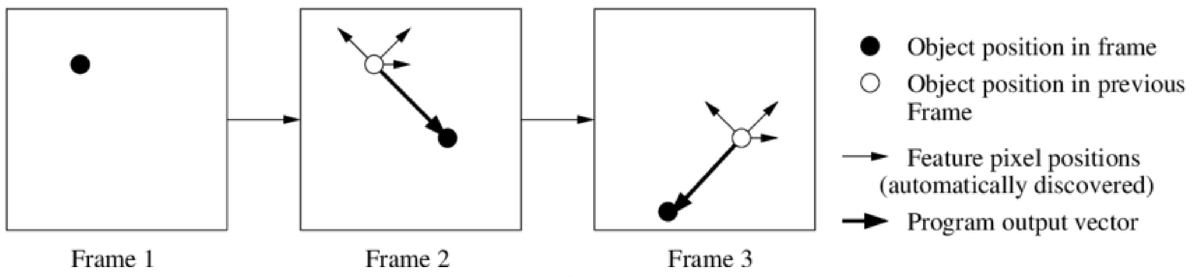

Figure 5: Object tracking.

Performance & Evaluation

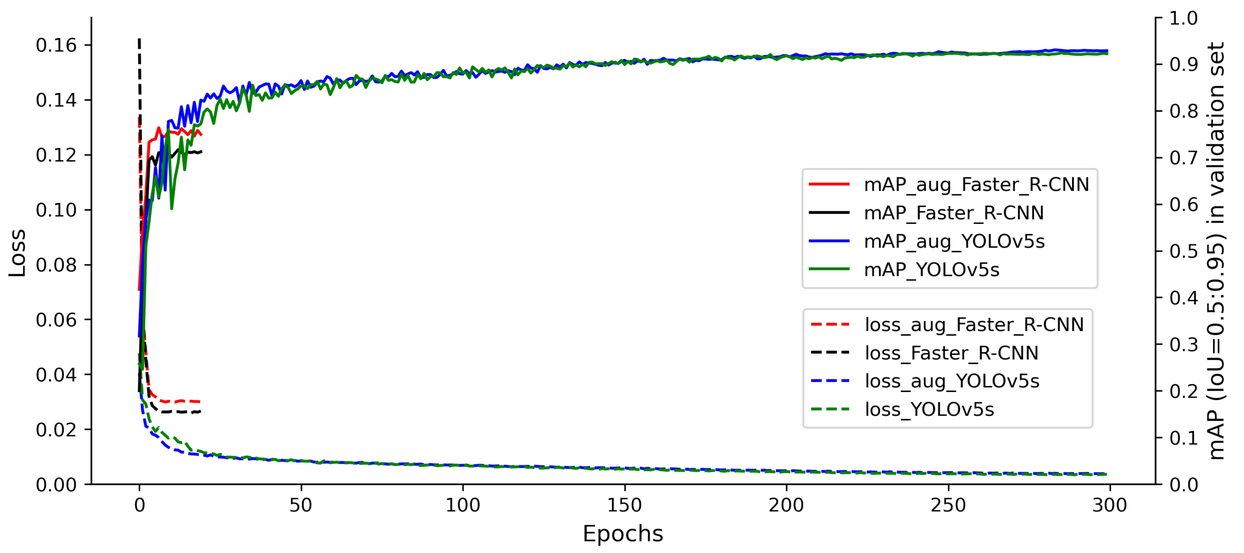

Figure 6: YOLOv5s's average loss and mAP in validation set compared with Faster R-CNN.

Model	Speed(FPS)	Accuracy(%)	mAP@[0.5:0.95]	mAP@[0.5]
Faster R-CNN	53.3	98.6	0.763	0.980
YOLOv5s	83.3	99.2	0.930	0.995

Table 1: Performance of Faster R-CNN and YOLOv5s based on NVIDIA's RTX 3070:

We have successfully developed a real-time object detection and tracking system, which can automatically detect beer cans and cola cans in real time with high precision, and accurately track and count objects at the same time.

In our test, the system can work with 99% accuracy for videos which utmost 80FPS. When there are two close cans that are moving fast, in some of the frames the bounding box is wrong.

Future Work

- The object tracking algorithm can be further upgraded by extra features:
  - introducing the Kalman filter for more precise tracking and motion prediction, which can also handle the potential occlusion situation
  - registering the object as valid only when it enters the camera frame from four sides around
- Use the time-series motion prediction from the tracking algorithm to alleviate the motion blur effect and improve the detection, inspired by [3]
- Test the software on a Jetson-Nano to evaluate the real performance

References

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[2] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[3] M. Sayed and G. Brostow. Improved handling of motion blur in online object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1706–1716, 2021.

s212060@student.dtu.dk