Object Detection

By
Subhadip Samanta | CSE Data Analytics |TNU2020021100001
Wrishav Sett | CSE Data Analytics |TNU2020021100004
Sudipta Saha | CSE Data Analytics |TNU2020021100005
Subarna Das | CSE AI&ML |TNU20200531000111

Contents

Introduction

- Object detection is a field in computer vision that focuses on the development of algorithms and systems capable of identifying and classifying objects in digital images or video. It enables machines to understand and interpret visual information, mirroring human perception.
- Object detection originated from early research in computer vision, which began in the 1960s.
- Initial approaches were based on simple features and heuristics, but significant progress was made with the advent of deep learning in the 2010s.
- Deep neural networks, especially convolutional neural networks (CNNs), revolutionized object recognition by achieving remarkable accuracy and scalability.
- The Viola-Jones algorithm, proposed by Viola and Jones in 2001, marked a significant milestone in object detection.

Popular Object Detection Algorithms

- Histogram of Oriented Gradients (HOG),
- Scale-Invariant Feature Transform (SIFT),
- Deformable Part Models (DPM),
- Deep Learning-based Approaches such as R-CNN, Fast R-CNN Faster R-CNN,
- Single Shot MultiBox Detector (SSD),
- You Only Look Once (YOLO),
- Two-Stage Detectors such as Mask R-CNN,
- EfficientDet and
- DETR (DEtection TRansformer).

Definition of Problem Statement

- Through this project work we set out to achieve an algorithm that would be able to detect objects not only from still frames but also from live video-feed with enough accuracy to be used in real world scenarios without sacrificing on the speed.
- We had several algorithms to work with such as YOLO, SSD and Faster R-CNN.
- However amongst the above we decided to move forward using the YOLO algorithm not only because of its high accuracy but also because of its high speed.
- Although SSD provides a similarly accurate result through use of predefined anchor boxes, it’s speed is not up to the mark when taking into considerations YOLO algorithms.
- Faster R-CNN provides a much more accurate result than both YOLO and SSD algorithms, since it uses a two-stage approach for object detection but it’s not nearly as fast as either in real-world scenarios.

Accuracy and Speed

Algorithm	Precision%	Recall%	F1%	mAP%
YOLOv3	69.13	80.19	70.14	80.17
SSD	63.17	88.69	72.13	82.41
Faster R-CNN	62.19	94.24	78.23	87.69

Algorithm	mAP%	FPS	Model Size
YOLOv3	78.52	69	236M
SSD	78.69	41	445M
Faster R-CNN	79.63	3	1129M

*The above data is for the MS COCO dataset comprising of 80 classes

Challenges

While the project development period we faced multiple challenges especially since this was a new field of study for the majority of the individuals of the group. Some of them are:

- Learning about Deep Learning and CNN in detail
- Knowledge of how to prepare a dataset
- Knowledge of the multiple detection algorithms and their advantages and disadvantages and their ease of use for first time developers
- Connecting the GUI with the code for deployment which is still under way
- Acquiring the necessary files for running the algorithm and use these pretrained files for transfer learning
- Resource availability for training models on custom dataset

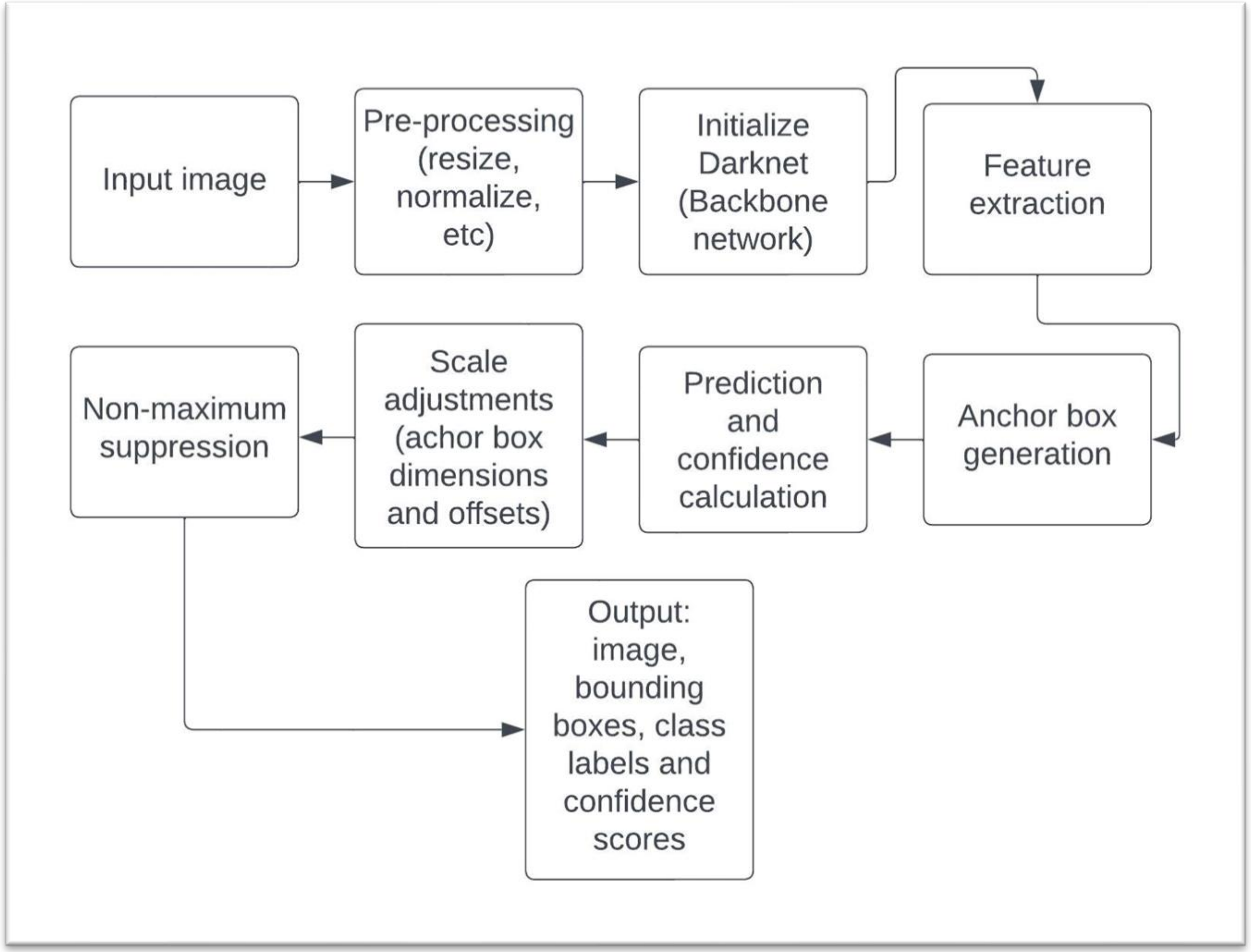
Methodology

The methodology behind a YOLOv3 algorithm:

1. Multi-scale detection
2. Input Image
3. Feature Extraction
4. Anchor boxes
5. Prediction refinement
6. Non-maximum suppression
7. Output

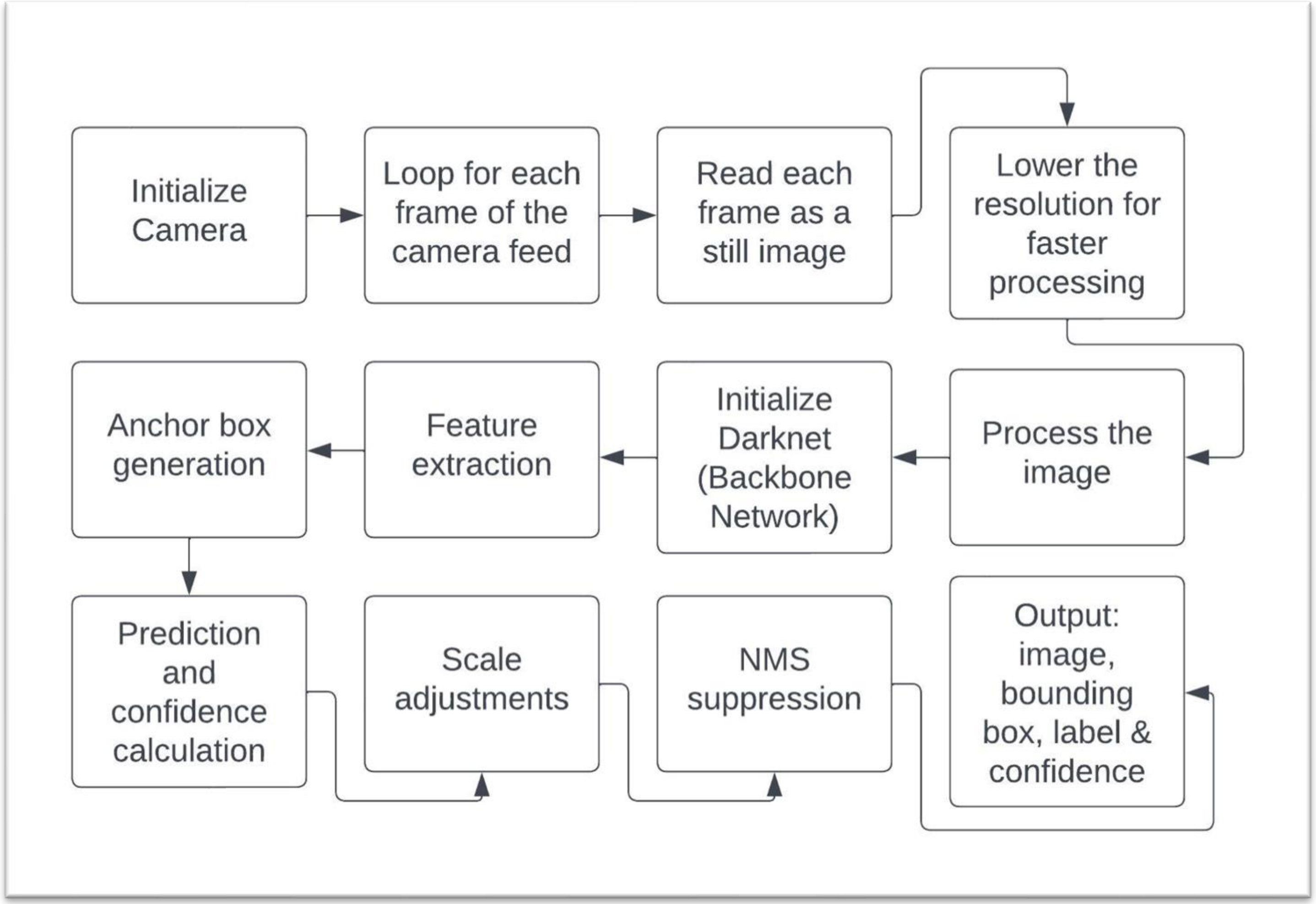
Methodology cont.

Flow chart for the working of YOLOv3 using a pretrained dataset for still images for object detection:



Methodology cont.

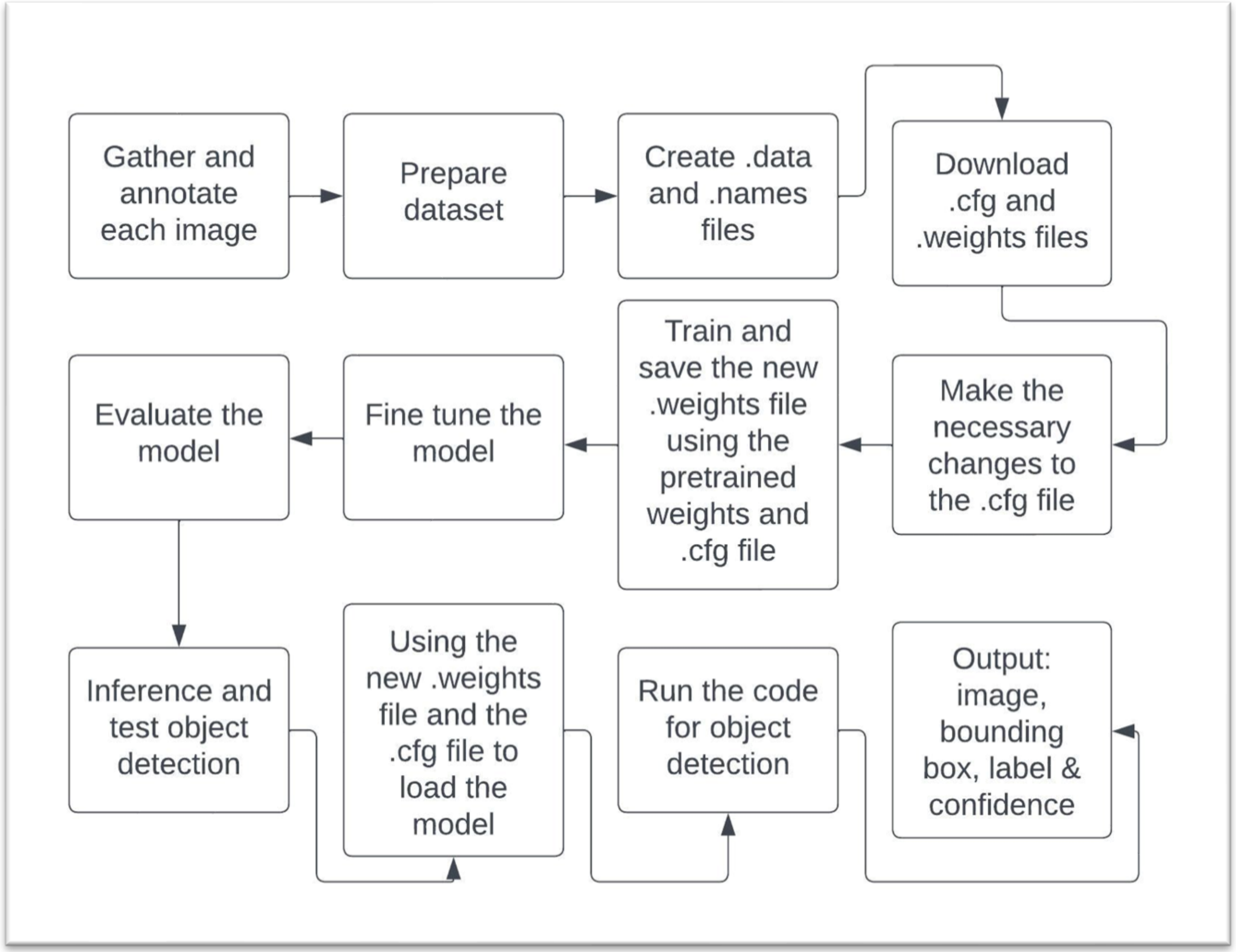
Flow chart for the working of YOLOv3 using a pretrained dataset for live video for object detection:



Methodology cont.

Flow chart for the working of YOLOv3 using a custom dataset for still images for object detection:

Once the model is trained to the desired level of accuracy and speed we can proceed to use the model for object detection using the same steps mentioned in slide **8** from still images.



*For further detail refer to page no. **57** to **68**, modules **6.9**, **6.9.1**, **6.9.2** of the project report provided.

Software Engineering Model

A SE Model is a description of the software process that represents one view, such as the activities, data or roles of people involved.

We have used the Incremental Model for our project.

The incremental or iterative and incremental model involves dividing the process into smaller modules and working and delivering said modules as we progress.

Once all the modules are developed they are put together to give the end product or in our case the working project.

The modules that we divided our project into are as follows:

- Developing of the GUI
- Creating the custom dataset
- Training the model for the custom dataset
- Using the trained model for object detection
- Object detection for still images
- Object detection for live video
- Connecting the GUI (frontend) and the code (backend) and deployment

Demonstration

Q&A

The End