Adv MM Assignment #3 – Testing face and pedestrian detection on short clip from "The Matrix" at different levels of compression

Khang Hui Chua, 2020280442
Kai Wen Yoke, 2020280598

January 7, 2021

1 Introduction

In this assignment, we test three types of face and pedestrian detectors (traditional and deep learning based) on a short clip from "The Matrix" movie, compressed to different levels. The video clip is challenging for these detectors because it is high action, with multiple characters running around with multiple poses in bad lighting conditions, gunfire and oftentimes partially occluded. With additional artifacts introduced by compression, the detection difficulty increases. We hence assess how detection performance varies at different compression levels, and also how performance varies across different detectors.

1.1 Background on face and pedestrian detection

Face and pedestrian detection are specific applications of object detection, which is an important computer vision task that deals with identifying and locating instances of particular visual objects in digital images. In the past two decades, object detection has progressed rapidly, and it is generally agreed that object detection can be separated into two periods: (1) traditional and (2) deep learning based, as shown in Figure 1. Traditional object detection algorithms were built based on handcrafted features, such as Haar wavelet features and HOG descriptors. As the performance of hand-crafted features hit a bottleneck, convolutional neural networks started to result in unprecedented improvement in object detection performance.

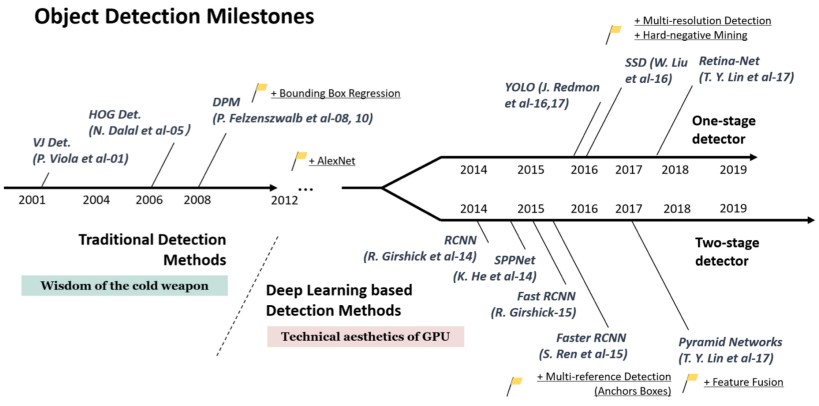

Figure 1: Progress timeline of object detection algorithms [1]

Face and pedestrian detection, as important object detection applications, are important prerequisites in many areas such as autonomous driving, video surveillance, criminal investigation etc. Face and pedestrian detection methods largely follow the same trend as general object detection algorithms, from the evolution of traditional feature designing methods to deep learning methods. Difficulties plaguing pedestrian detection include detection of small pedestrian bodies, similarity between pedestrian bodies and backgrounds, dense and occluded pedestrian; difficulties in face detection include variation in faces (as a result of varying expressions, skin colours, poses and movements), occlusion, small face sizes. We face some of these challenges in the short video clip to be tested for this assignment.

2 Methodology

We first prepare the test video clips by compressing them to different bitrates. Then, we prepare the ground truth labels by annotating the bounding boxes (of faces and bodies) in the original video clip. Subsequently, we apply three sets of face and body detectors on all the video clips to output detected bounding boxes for the faces and bodies. Finally, we evaluate the detector performance by comparing detected and ground truth bounding boxes.

2.1 Source video compression

We compressed the source videos to 12 different output bitrates (in kbps): {25, 50, 75, 100, 125, 150, 200, 300, 500, 1000, 2500, 5000}. The output bitrates were chosen based on visual inspection of the video quality and the original bitrate of the source video i.e. we found that artifacts started appearing very significantly at 100kbps and below, so we had finer intervals for smaller bitrates, and larger intervals for higher bitrates which showed no significant visual difference, up till 5000kbps which was around the original source bitrate (see Figure 2). Compression was executing using the following `x264` command:

```
x264 --bitrate $BITRATE_KBPS_TARG -o $FILENAME_OUT $FILE.
```

Figure 2: Image patches at different compression rates

2.2 Annotating bounding boxes for evaluation

We used the Computer Vision Annotation Tool (CVAT) [1] to label the ground truth bounding boxes in the required format. We manually drew bounding boxes for faces and bodies appearing in the video clip which had 2990 frames and labelled them as such, so that we can evaluate the detection accuracy quantitatively subsequently.

2.3 Face and pedestrian detection algorithms used

We tested three sets of face and pedestrian detectors: (1) Haar-cascade based; (2) HOG based; (3) Deep Learning based. First two were run on CPUs, and the last was run on GPU. Our choice of detectors follow the general classification of object detection algorithms, with the first two being traditional methods based on specially designed features, and the third being state-of-the-art deep learning algorithms. We do not bother comparing the running times since we are using different processors for the detectors, and we only compare the detection accuracy.

2.3.1 Haar cascade based detector

We used the OpenCV implementation [2] of the Viola-Jones (VJ) detector [2], which provided pretrained haar-cascade models that were able to detect faces and bodies. The VJ detector was a seminal development that marked the achievement of real-time detection of human faces for the first time in the 2000s. The VJ detector slides windows across the image to go through all possible locations and scales to see if any window contains a human face/body, and this process is sped up with three innovations: (1) Integral image: a computational method to speed up convolution computation independent of window size, hence speeding up the calculation of the Haar wavelets per window; (2) Feature selection: using Adaboost to select a small set of most relevant Haar features from a huge set; (3) Detection cascades: a multi-stage detection paradigm that reduce computation by spending less computations on background windows but more on face/body targets.

2.3.2 HOG based detector

We used the OpenCV implementation for the HOG-based body detector [3] and the dlib implementation for the HOG-based face detector [4]. The Histogram of Oriented Gradients (HOG) feature descriptor was proposed in 2005 [3], and is an important foundation of many object detectors. HOG was primarily motivated by the problem of pedestrian detection, and hence very applicable to this assignment. The HOG descriptor focuses on the shape and structure of an object by calculating the gradient and orientation of the edges in localised portions of an image. The HOG detector then rescales the input image multiple times while keeping the detection window size unchanged to detect objects of different sizes.

2.3.3 Deep learning based detector

Convolutional Neural Networks (CNN) have greatly improved object detection accuracy. CNN-based detectors can be separated into two-stage and the more recent one-stage detectors, the lat-

[1] https://cvat.org/
[2] https://github.com/opencv/opencv/blob/master/samples/python/facedetect.py
[3] urlhttps://github.com/opencv/opencv/blob/master/samples/python/peopledetect.py
[4] https://github.com/davisking/dlib/blob/master/python_examples/face_detector.py

ter of which is faster and more efficient. We used PaddlePaddle's implementation [5] of one-stage CNN-based object detectors. For the face detector, we used the PyramidBox model implemented in PaddlePaddle [6], and for the pedestrian detector, we used the YOLO-v3 model with Darknet backbone implemented in PaddlePaddle [7]. PyramidBox is a state-of-the-art context-assisted Single Shot Face Detector proposed in 2018 [4], and YOLO-v3, proposed in 2018 [5] with improvements over YOLO, which was the first one-stage detector in the deep learning era, which was remarkable for having good detection accuracy with much higher detection speed than comparable detectors.

2.4 Evaluation metrics

We evaluated the detectors' performances using precision and recall metrics, where true and false positives are determined by comparing the Intersection over Union (IoU) between the detected bounding boxes and the ground truth bounding boxes to a set threshold of 0.5.

[5] https://github.com/PaddlePaddle/Paddle

[6] https://github.com/PaddlePaddle/models/tree/develop/PaddleCV/face_detection

[7] https://github.com/PaddlePaddle/PaddleDetection/blob/release/2.0-beta/docs/featured_model/CONTRIB_cn.md

3 Results

We tested the three sets of detectors on the twelve compressed versions of the video clip. Figure 3 shows a quick comparison of the ground truth bounding boxes and the bounding boxes detected by the three detectors of a particular frame from the video clip. The red bounding boxes bound the bodies, while the blue bounding boxes bound the faces. From this one frame alone, it is clear that the PaddlePaddle deep learning detectors perform extremely well with near-perfect localisation accuracy, while the traditional detectors (Haar and HOG) performed extremely poorly for body detection but decently for face detection. The following sections will analyse in greater detail the detectors' performance.

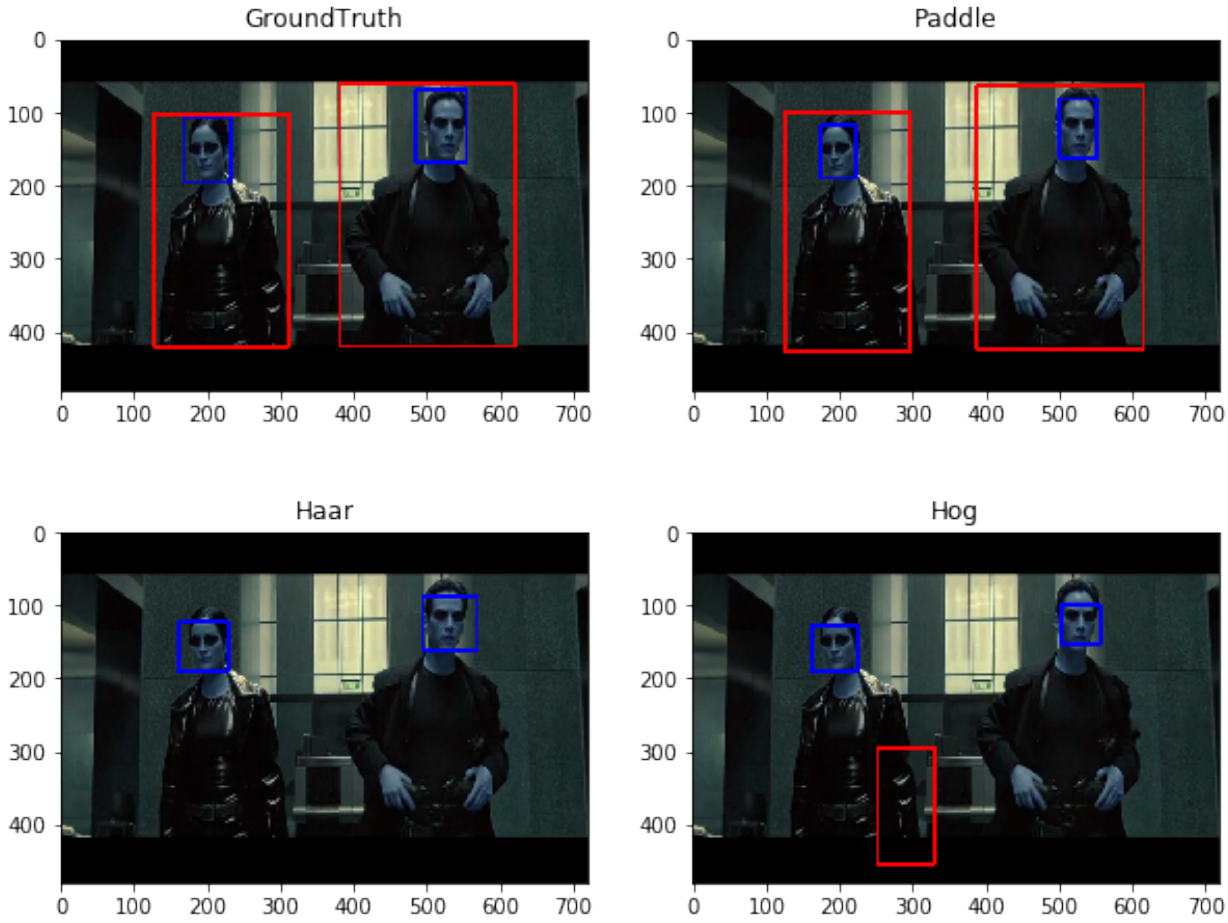

Figure 3: Bounding boxes from Ground Truth, Haar Cascade model, HOG based model, and Deep Learning based model

3.1 Quantitative analysis

Figure 4 compares the face detection and body detection precision and recall metrics across different compression levels, for each set of detectors.

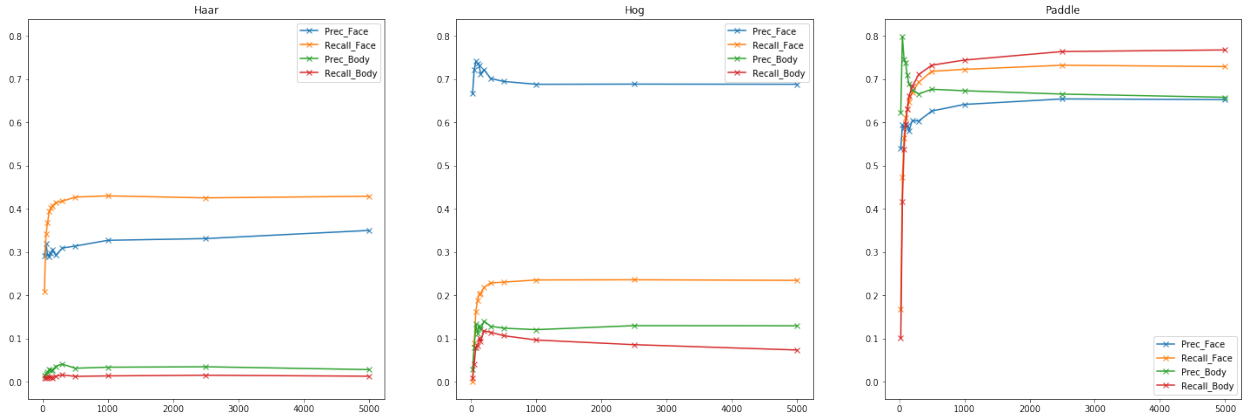

Figure 4: Precision and recall for all three sets of face and body detectors across different video compression levels

3.1.1 Performance between different detectors

All three face detectors work to some extent, while only the Paddle detector displayed decent performance for body detector. It is unsurprising that Paddle worked the best by far across both metrics, given that it is based on deep learning and state-of-the-art. The HOG model exhibited good precision for face detection, but extremely poor recall. Although HOG was first applied on pedestrian detection, it was meant for detecting pedestrians with a more fixed aspect ratio (i.e. upright pedestrians), but the video clip was full of bodies with different poses, resulting in poor performance in body detection for HOG. In fact, HOG detected many false positives. The Haar detector is very sensitive to lighting given the nature of the Haar wavelet features. As faces in the video clip still retains the standard face features such as lighter shades at the nose bridge and darker shades around the eyes, the Haar detector could still somewhat perform for face detection for both precision and recall, but poor lighting conditions in the video resulted in the Haar detector failing to detect bodies at all.

3.1.2 Performance across different compression levels

We focus on the graph for Paddle in Figure 4 since it is the only set of detectors that managed to achieve good performance for both face and body detection. We expected detection performance to improve with lower compression i.e. fewer compression artifacts, and indeed that is true for the recall metrics of both face and body detection. The sharp drop in recall performance for bitrates 100kbps and below is also consistent with our visual observation of the poor video quality at these bitrates. The almost constant recall at higher bitrates is also consistent with our visual perception that there is no significant difference when compression rate is 200kbps and above (see Figure 2). This suggests that CNN-based detectors are indeed closely related to human perception. For body detection, it was also interesting that precision was higher at lower bitrates, but this does not really mean much because at low bitrates, a lot of true body detections are missed, and it just implies that out of the small number of detected bodies, a greater proportion is indeed true. This might be because some of the compression artifacts blur the details within a body and thus makes the shape of the body more defined, and thus the small number of detected bodies have greater confidence of being true if they are even detected in the first place.

3.2 Visual analysis

Figure 5 shows the bounding boxes for the Paddle detectors at different compression levels for the same corresponding frame. Detection accuracy is near perfect for 100kbps and above, considering this is one of the easier frames with less noise.

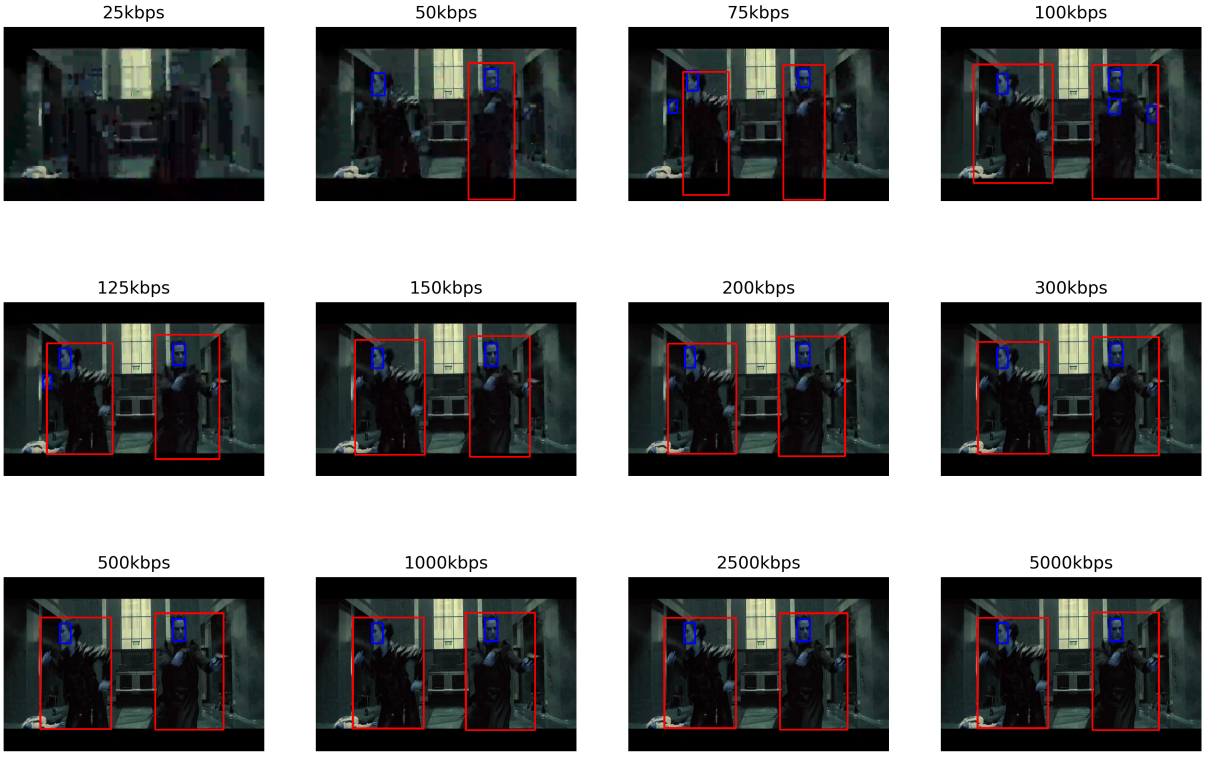

Figure 5: Sample detections across different compression levels for Paddle detectors for frame 469

Figure 6 shows several frames which had wrong detections by the Paddle detectors. The difficulty arises from heavy occlusion by gunfire and debris as in (a), different postures like crouching as in (b) and somersaulting as in (d) and partial body parts as in (c).

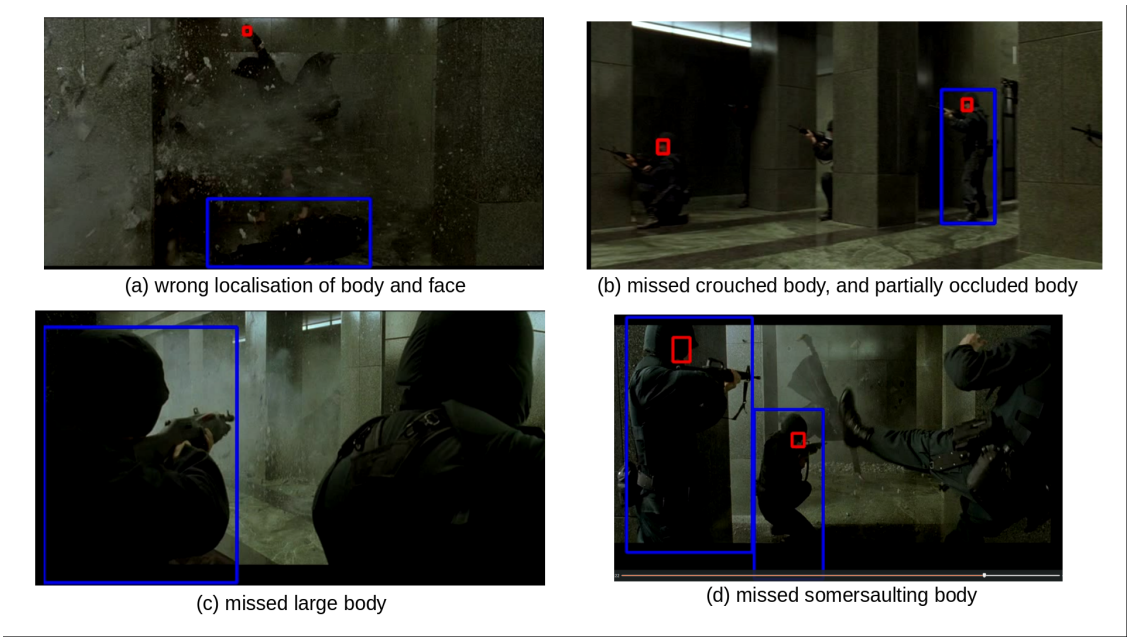

Figure 6: Particularly difficult frames for Paddle detectors

Figure 7 shows several impressive detections made by the Paddle detectors. Paddle was able to pick up a lying body as in (a), lower bodies of running soldiers as in (b), bodies and faces covered by debris and dust that are almost imperceptible for human eyes as in (c) and (d).

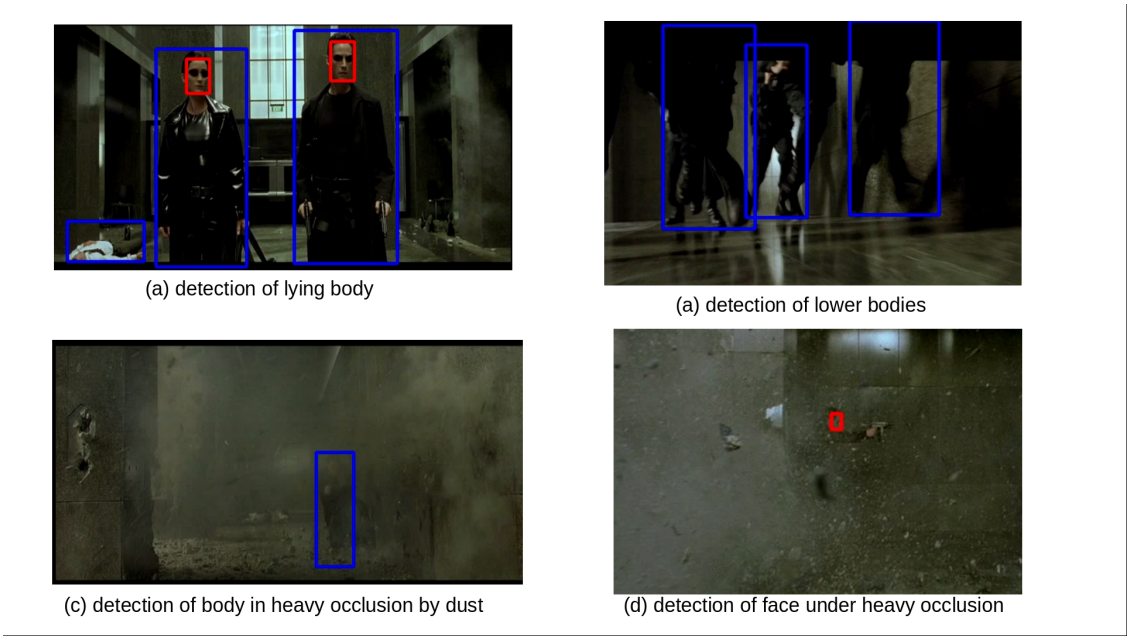

Figure 7: Impressive detection performance by Paddle detectors

4 Conclusion

Video compression rate does not have significant effect on object detection performance when bitrate is high enough, but below a certain compression threshold, object detection crumbles quickly. The compression threshold relates to human visual perception of the quality of the video, i.e. if we think after the compression, the video looks bad, then object detection will perform poorly. Also, CNN-based detectors outperforms traditional detectors by a huge margin, and even exceed human detection capabilities in some cases, while detection still remains problematic in very difficult conditions.

References

[1] Z. Zou, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *CoRR*, vol. abs/1905.05055, 2019. [Online]. Available: http://arxiv.org/abs/1905.05055

[2] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 1, 2001, pp. I–I.

[3] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, 2005, pp. 886–893 vol. 1.

[4] X. Tang, D. K. Du, Z. He, and J. Liu, "Pyramidbox: A context-assisted single shot face detector," 2018.

[5] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *CoRR*, vol. abs/1804.02767, 2018. [Online]. Available: http://arxiv.org/abs/1804.02767