Part 1a

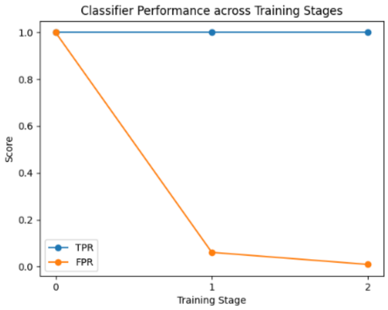


Figure 1 – Classifier performance per stage.

I trained a strong classifier on the dataset of dart images and suitable negative examples. Each stage in the training process adds an additional classifier. After training, the classifier has three layers. Throughout training, the true positive rate (TPR) remains constant at 1.0. The false positive rate (FPR) also starts at 1.0 but decreases to 0.009. The cascade classifier is designed around identifying potential pixel windows where a dartboard may be present and exploring these with more advanced classifiers. This leads to rapid detection as many negative regions can be disqualified cheaply. Positive samples require positive classification by all classifiers, which become more advanced in later layers. In order achieve this, classifiers typically classify a high proportion of samples as positive instances, particularly early-stage classifiers. This happens to prevent positive samples being discounted too quickly, but necessarily means lots of negative samples are erroneously classified positively in early stages. This high positivity rate accounts for the high TPR rate across all layers. As layers deepen, the classifiers become more advanced and hence can more reliably discard negative samples. Hence, later stage classifiers have a lower FPR as they are more selective (hence the model FPR decreases).

Part 1b

Image Number	TPR	F1
0	1	0.125
1	1	0.25
2	1	0.143
3	1	0.222
4	1	0.222
5	1	0.211
6	0	0
7	1	0.25
8	1	0.174
9	1	0.222
10	0	0
11	1	0.133
12	1	1
13	1	0.133
14	1	0.063
15	1	0.333

Table 1 – Cascade classifier performance.

Figure 2 – Cascade classifier performance on three dartboard images.

Average TPR	Average F1
0.875	0.218

Table 2 – Mean cascade classifier performance.

The classifier achieves a mean TPR score of 0.875 and an average F1 score of 0.218. The classifier achieves an impressive recall score but has a lower precision rating. This is because the classifier is prone to many false positive results. The model effectively identifies dartboards, even in cases of partial occlusion and when multiple boards exist within an image. However, the model is prone to identifying dartboard regions that are correct but too large to be considered accurate. The training performance for the TPR is marginally better as the model is being trained on similar data of dartboards. However, the test TPR is rather impressive too, which suggests that the model has effectively learnt dartboard features. The training FPR is much better than the test FPR. This can be explained by the fact that the model is trained on 500 negative images. However, the variety of potential negative images is extremely large, and therefore this training set is unlikely to capture enough of that variety to generalise well. Hence, the model learns the negative instances effectively on the training set, but this fails to generalise effectively to the test data items.

Part 2a

Figure 3 – (Left to right) Thresholded gradient magnitude of image, circular Hough space of image and detector prediction for two dartboard images. Detector combines Hough space data with cascade classifier to make predictions.

Part 2b and 2c

Image Number	TPR	F1	TPR Difference	F1 Difference
0	1	1	0	+0.875
1	1	0.4	0	+0.15
2	1	0.25	0	+0.107
3	1	0.333	0	+0.111
4	0	0	-1	-0.222
5	1	0.5	0	+0.289
6	0	0	0	0
7	1	0.5	0	+0.25
8	1	0.571	0	+0.397
9	1	0.333	0	+0.111
10	0	0	0	0
11	0	0	-1	-0.133
12	1	1	0	0
13	1	0.286	0	+0.153
14	1	0.143	0	+0.08
15	1	1	0	+0.667

Table 3 – Cascade classifier with circle filtering performance. Relative performance to no circle filtering given.

Average TPR	Average F1	Average TPR Difference	Average F1 Difference
0.75	0.395	-0.125	+0.177

Table 4 – Mean cascade classifier with circle filtering performance. Relative performance to no circle filtering given.

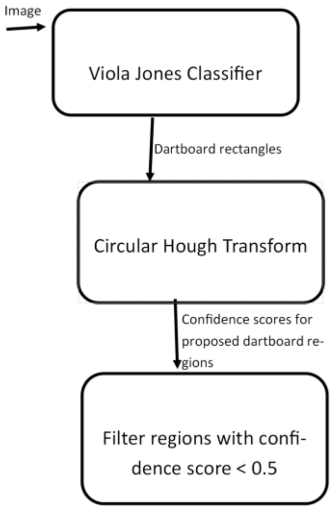


Figure 4 – Flowchart describing how circular Hough transform is incorporated with cascade classifier to detect dartboards.

Given the high TPR rate of the cascade classifier, I use proposed rectangular dartboard regions as input to my Hough algorithm. Each region is assigned a confidence score, and regions that have a confidence score of less than 0.5 are removed. To calculate this confidence score, I utilise the circular

Hough transform to detect circles within an image. The confidence score considers gives a high confidence score to images with multiple circles. This approach helps filter the false positives produced by the viola jones method and only leads to a small (but non-negligible) reduction in TPR. This approach proves largely effective, however it struggles with regions where circles are present that aren't dartboards. A better algorithm would use line and ellipse data in combination with this to combat occlusion and camera perspective more effectively. It does help to filter out clearly spurious proposal regions which contain no circular data. Additionally, this approach is very slow as the Hough space calculation creates a large accumulator matrix.

<u>Part 3</u>

I decided to use a deep learning detection approach, as such techniques exhibit state-of-the-art performance for object detection. I combined the provided dartboard image ('dart.bmp') and the negative images used for training the cascade classifier to create a synthetic dataset of 2000 dartboard images. The dartboard pictures are distorted with various augmentation techniques including contrast adjustment, image scaling and object occlusion.

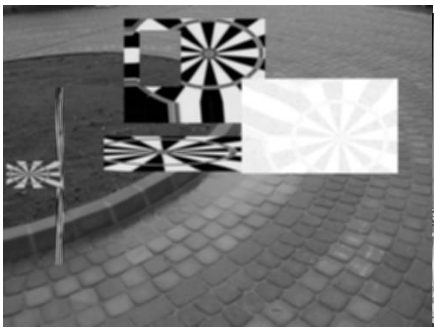
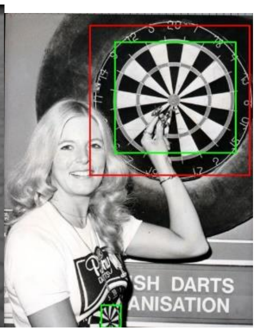


Figure 5 – Example of synthetic training image



Average TPR	Average F1	Average TPR Difference	Average F1 Difference
0.948	0.813	+0.198	+0.418

Table 5 – Mean performance of deep learning dartboard detector. Relative performance to cascade classifier with circle filtering given.

Figure 6 – Final detector performance on two images.

I finetuned a pretrained YOLOv8 model with my synthetic dataset. I trained my model for 60 epochs, saving model weights every 20 epochs. Each trained model achieves a very low false positive rate and accurately detects most dartboards. My best model achieves a TPR of 0.88 and an F1 score of 0.90. Given the low false positive rate, I combined the model predictions together to achieve a TPR of 0.95 and an F1 score of 0.81. This combined approach is effective because of each model's low false positive rate, accurate dartboard bounding box and differing understanding of what constitutes a dartboard. In *Figure 6* the detector produces accurate bounding boxes and even identifies a dartboard pattern on a shirt.

My approach is superior to my previous detector as it identifies slightly more true positives and produces far fewer false positives. Additionally, the bounding boxes produced are significantly more precise. The approach is 11 times faster than utilising the cascade classifier in combination with the Hough algorithm. My approach could be improved by combining synthetic data with labelled real-world images. Additionally, my approach does not consider colour information, which could help further elevate performance.

Image Number	TPR	F1	TPR Difference (Circle Cascade)	F1 Difference (Circle Cascade)
0	1	1	0	0
1	1	0.666	0	+0.266
2	1	1	0	+0.75
3	1	1	0	+0.667
4	1	0.8	+1	+0.8
5	1	1	0	+0.5
6	1	0.666	+1	+0.666
7	1	1	0	+0.5

Image Number	TPR	F1	TPR Difference (Circle Cascade)	F1 Difference (Circle Cascade)
8	0.5	0.666	-0.5	+0.095
9	1	0.666	0	+0.333
10	0.666	0.571	+0.666	+0.571
11	1	0.8	+1	+0.8
12	1	1	0	0
13	1	0.5	0	+0.214
14	1	1	0	+0.857
15	1	0.666	0	-0.334

Table 6 –Performance of deep learning dartboard detector. Relative performance to cascade classifier with circle filtering given.