CITS4402 – Computer Vision Project

Amos Viray (23729527), Cameron Waddingham (23737222)

Phase 1: Dataset Building

For the training set, we aimed for at least 250 human (positive) and at least 250 non-human (negative) samples. Initially the positive and negative dataset were gathered from public datasets provided by the project description where we randomly picked images of individuals only for the positive dataset and no people or animals for the negative dataset. For the testing dataset we decided to use a dataset that hadn't been seen before to truly test the model. As the model was going to be tested to detect if a person in in an image, we decided to use images that had more than one person or had more in the background. We obtained these images through other resources [https://universe.roboflow.com/m3-ytsk5/m3finalclass] and randomly selected the images for our test data. We noticed that our model was not performing as we expected, as shown in Table 1, the metrics of the first dataset show its accuracy and miss rate are 53% and 59% respectively. This was not enough to start our ablation study as we wanted a more accurate model so it would be more obvious as to the effects of the ablation study. We had realised that we were training the model of a different dataset than what it would be tested on. To rectify this we sampled more images from the new dataset for the positive and negative datasets so that we had a wider range of images to train and test the model but to also have more data points to train the model on and get a higher accuracy from the model.

Table 1: Metrics comparing the first and second dataset

	First Dataset	Second Dataset
Accuracy	53.00%	74.00%
Precision	53.95%	91.38%
Recall	41.00%	53.00%
Miss Rate	59.00%	47.00%
FPPW	0.175	0.025

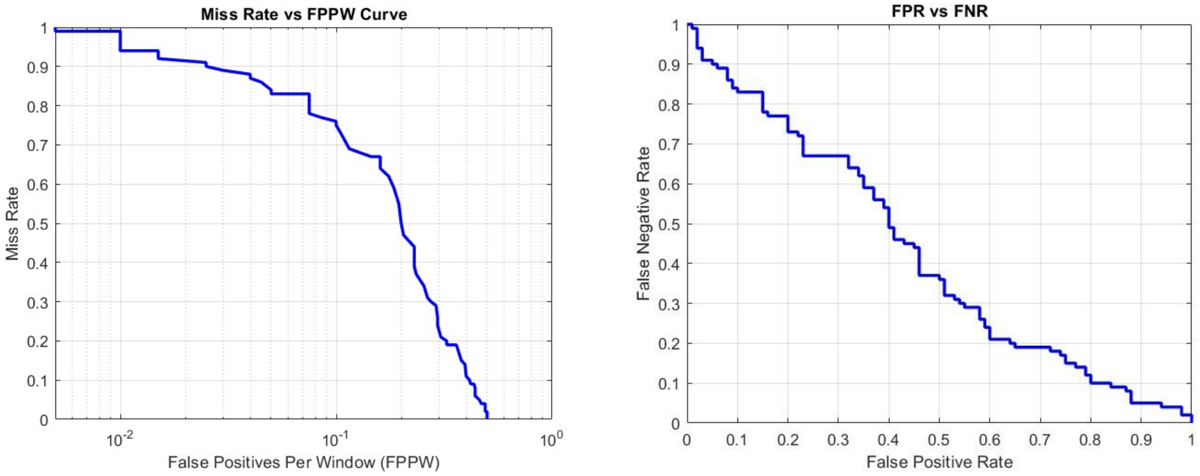

Figure 1: Miss rate vs false positives per window and false negative rate vs false positive rate of the model for the first training dataset

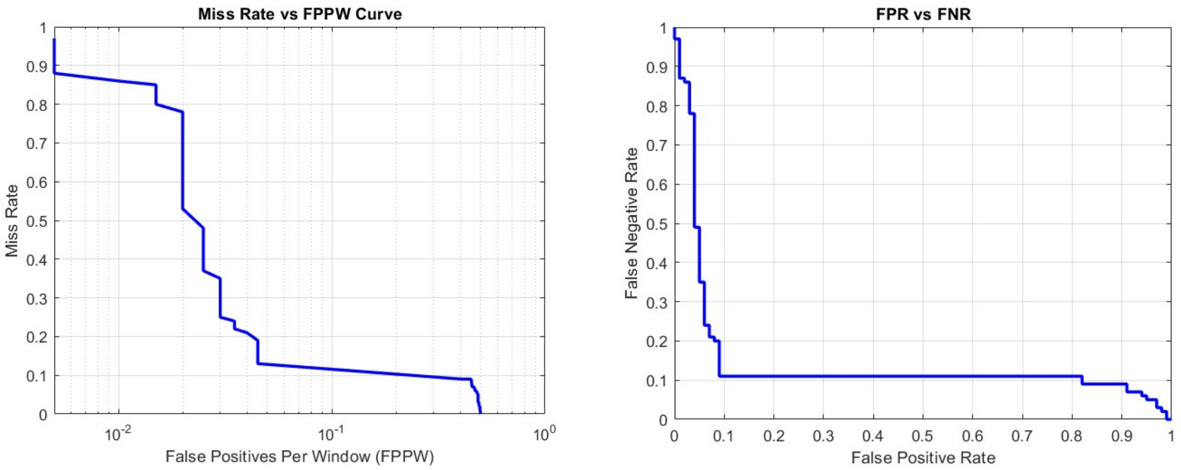

Figure 2: Miss rate vs false positives per window and false negative rate vs false positive rate of the model for the second training dataset

After adjusting the dataset, the model performed much better with a significantly higher accuracy and precision and a significantly lower false positives per window. This was used as our final dataset for training the model for our ablation studies.

The positive and negative images are stored in their respective directories are in the .png and .jpg format. All photos in both the training and testing sets are pre-processed in our MATLAB program, where it is resized to 64x128 pixels and converted to greyscale.

Phase 2: Feature Extraction and Model Training

For the HOG feature extraction, we implemented our own version from scratch as the in-built function for the HOG feature extraction in MATLAB is very limited in terms of how much we are able to modify its parameters. The best parameters we found and used were the default parameters required in the original implementation, which are:

1. [-1,0,1] filter without smoothing for the gradient computation
2. 9 bins for orientation binning to cover angles from 0°-180°
3. 8x8 pixels for the cell size
4. 16x16 pixels for the block size
5. L2-Hys normalisation for block normalisation
6. 8-pixel spacing for block stride
7. 64x128 pixel detection window

To create our model we used the built-in fitcsvm function in MATLAB to train our SVM model, which uses a linear kernel. The feature extraction and model training and testing utilises the same method in terms of iterating through their respective datasets and processing the images.

- In general, after pre-processing, our program iterates through each image, extracts their HOG features, transpose the features matrix from a column vector to a row vector in order for each element to correspond to each of the images' feature where it is assigned a ground truth or false label (0 for non-human and 1 for human).

The program then generates prediction labels using our trained model and the testing dataset – using the predict function – which we use alongside the ground labels of our testing set to create the elements for a confusion matrix:

- true positive (TP) – sum of human images correctly predicted as human
- true negative (TN) – sum of non-human images correctly predicted as non-human
- false positive (FP) – sum of non-human images incorrectly predicted as human
- false negative (FN) – sum of human images incorrectly predicted as non-human

These elements are then utilised to calculate the metrics:

- Accuracy is the proportion of total predictions that were correct: $\frac{TP+TN}{TP+TN+FP+FN}$
- Precision is the proportion of actual human images, predicted as humans: $\frac{TP}{TP+FP}$
- Recall is the proportion of actual humans image that were predicted correctly: $\frac{TP}{TP+FN}$
- Miss Rate is the proportion of actual human images that were missed: $\frac{FN}{TP+FN}$
- False Positive Per Window (FPPW) is the proportion of total windows (images) incorrectly identified as human: $\frac{FP}{TP+TN+FP+FN}$

Phase 3: Ablation Studies

Sobel Filter

For one of the ablation studies conducted on our program, I decided to change the gradient filter from [-1,0,1] to Sobel. To compare the performances of the default filter and the Sobel filter, I decided to duplicate the original code and implement two HOG feature extraction functions – one with the default filter and one with the Sobel filer. The model is then retrained with the Sobel filter.

Table 2: Metric comparison of default and Sobel filters

	Default [-1,0,1]	Sobel
Accuracy (%)	74	75
Precision (%)	91.38	93.1
Recall (%)	53	54
Miss Rate (%)	47	46
FPPW	0.025	0.02

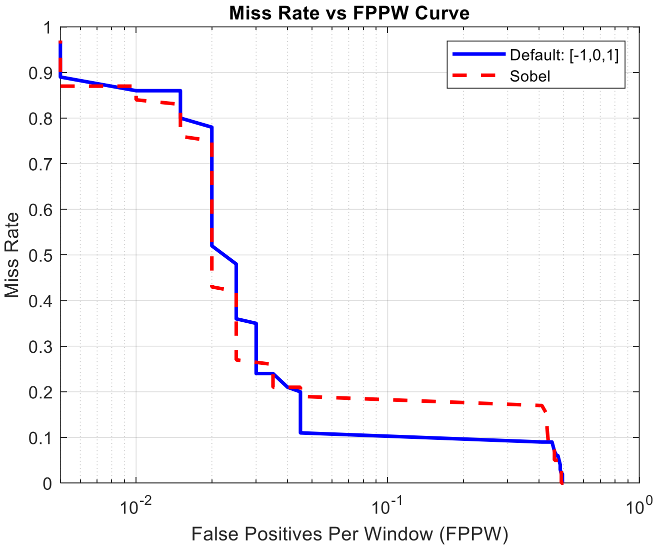

Figure 3: Comparison of DET curves of default and Sobel gradient filters

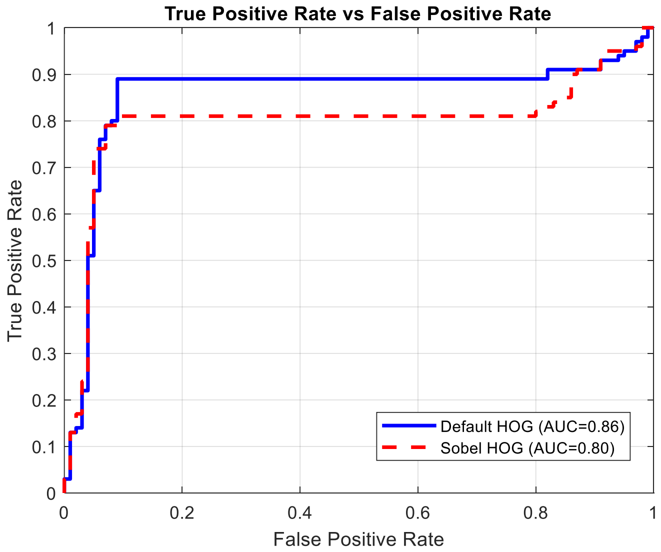

Figure 4: Comparison of ROC curves of default and Sobel gradient filters

By simply observing Table 2, one may conclude that the Sobel gradient filter outperforms the default gradient filter by a couple of percentages. In a DET curve, the curve closes to the bottom-left corner indicates better performance as there are lower false positives (FPR) and false negatives (FNR). In an ROC curve, the curve closest to the top left corner indicates better performance as the true positive rate (TPR) is higher and the false positive rate (FPR) is lower. Figure 3 shows that the Sobel slightly outperforms the default filter at lower FPRs but is outperformed by the default filter as the FPR increases. Figure 4 shows that the Sobel filter slightly outperforms the default in low FPRs (below ~0.1 FPR), where its TPRs are slightly higher. The Sobel filter is immediately outperformed by the default filter as the default filter's curve suddenly spikes at around 0.1 of FPR and retains that TPR as the FPR increases. The area under the curve (AUC) is also an indication of a curve's performance where a higher AUC indicates better performance. In this case, the default filter has an AUC of 0.86 compared to the Sobel filter's AUC of 0.8, indicating that the default filter is the better performing HOG feature extraction.

L1 Normalisation:

In this ablation study, I chose to implement L1 normalisation in comparison to the original implemented L2-Hys normalisation. L1 normalisation uses the formula $v' = \frac{v}{|v|+\epsilon}$, while L2-Hys normalisation uses three steps. Step one uses the formula $v' = \frac{v}{\sqrt{v^2+\epsilon^2}}$, step two clips all values that are above a threshold, then step three is to use the formula again to normalise for a second time. The model was duplicated and the normalisation technique in the HOG feature extraction was changed so that there are two models with two separate normalisation techniques. Both models were trained and tested on the same datasets that were used to train and test the original model.

Table 3: Comparison of metrics from two different normalisation techniques, L1 and L2-Hys

	L2-Hys Normalisation	L1 Normalisation
Accuracy	74.00%	70.00%
Precision	91.38%	90.00%
Recall	53.00%	45.00%
Miss Rate	47.00%	55.00%
FPPW	0.025	0.025

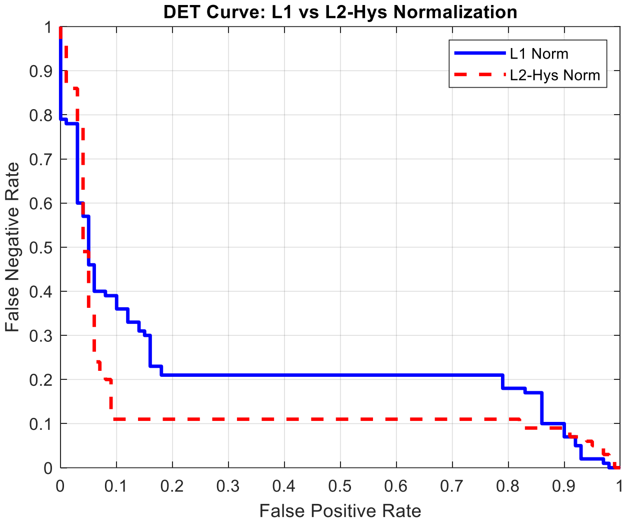

Figure 5: DET curve comparing both normalisation techniques, L1 and L2-Hys

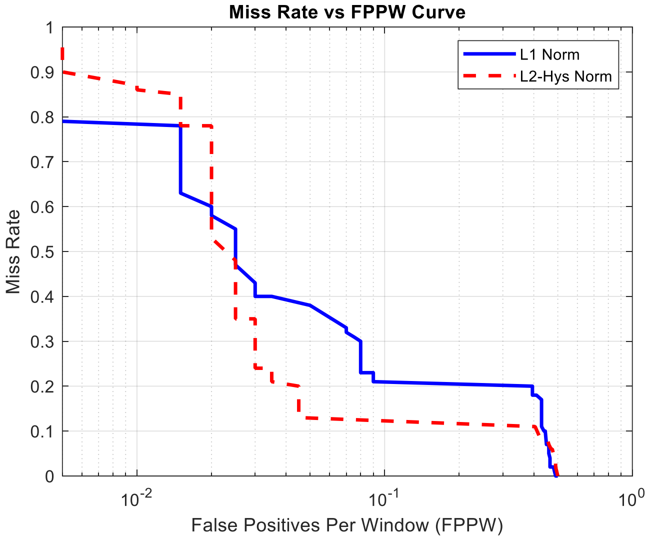

Figure 6: Miss rate vs false positives per window of both normalisation techniques, L1 and L2-Hys

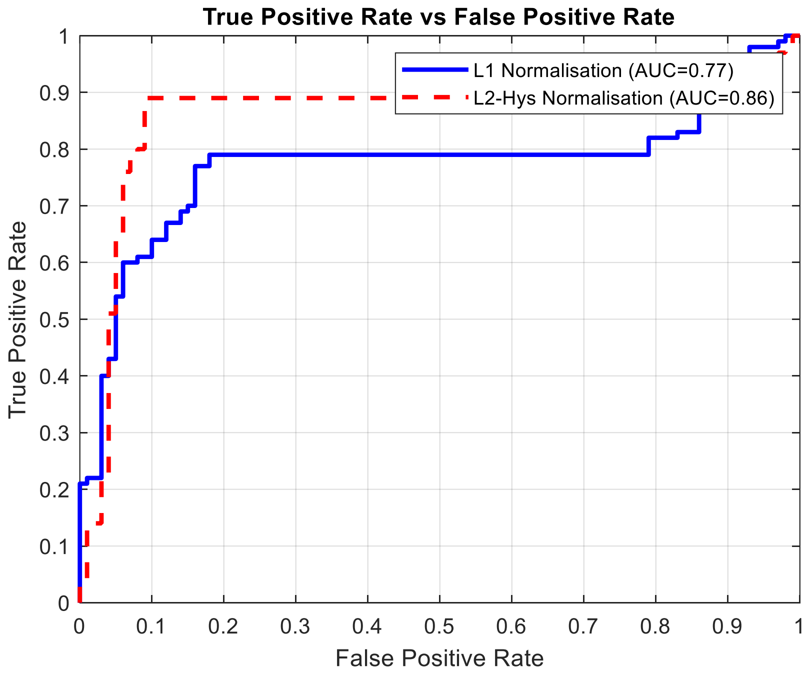

Figure 7: ROC curves comparing L1 and L2-Hys normalisation

Comparing the two techniques by metrics in Table 3 finds that the L1 normalisation is slightly worse than L2-Hys. The precision for L1 and L2-Hys is 90% and 91.38% respectively and the FPPW are both 0.025. Using these metrics, you would find that L1 normalisation is comparative to L2-

Hys. Observing Figures 5, 6 and 7, we see that L2-Hys performs significantly better than L1 normalisation. In Figures 5 and 6, the curve that is closer to the bottom left has a better model as it has the lowest miss rate, false positive rate and false negative rate. The L2-Hys curve is closer to the bottom left in both Figure 5 and 6, although the L1 curve does seem to perform better at low false positive rates. In Figure7, the greater area under the curve signifies a better model. The L2-Hys curve has a greater area under the curve of 0.86 compared to the L1 curve which has an area under the curve of 0.77. Although, again, L1 seems to perform better at low false positive rates. Overall L2-Hys performs significantly better over wider ranges than its counterpart.