OPTIMAL USE OF EXISTING FREEWAY MANGEMENT SURVEILLANCE INFRASTRUCTURE ON PEDESTRIAN BRIDGES WITH COMPUTER VISION TECHNIQUES.

H VAN DER MERWE*, M BOOYSEN* and S ANDERSEN**

*Department of E&E Engineering, Stellenbosch University, Private Bag X1, Matieland, 7602; Tel: 021 808-4013; Email: mjbooysen@sun.ac.za
**Department of Civil Engineering, , Stellenbosch University, Private Bag X1, Matieland, 7602; Tel: 021 808-2255; Email: jandersen@sun.ac.za

ABSTRACT

South Africa as a developing country has to make the most out of the infrastructure that are available. Given the high level of crash involving pedestrians, it is critical that all means available are utilised to characterise pedestrian movements on the highway and pedestrian bridges. This paper will focus on using the existing camera infrastructure, but will extend its use to automatically detect and count pedestrians that use the pedestrian bridges. The pedestrian movement data can be used to aid with the evaluation of pedestrian safety campaigns, or to recognise trends in pedestrian movement. The paper presents the impact of various parameter changes to the state of the art technique used, as well as orientation suggestions for future installations. This is done to make optimal use of existing infrastructure, and provides an alternative to existing high-end systems. The methodology includes training a computer vision-based algorithm to recognise and count pedestrians for specific scenes, for example pedestrian bridges. The paper evaluates different suppression techniques to reduce false positives. The results show that 72% of pedestrians can be detected (a hit rate of 72%), with the camera facing a pedestrian bridge squarely from the side, so that silhouettes are clearly visible. High end products not using existing infrastructure typically have a hit rate of 70%-90%. The solution in this paper competes with high-end products, and can be expanded for infrastructure security applications, e.g. monitoring copper cables or monitoring of high risk areas.

1. INTRODUCTION

A report released by (SANRAL, Western Cape Provincial Goverment, TCT, November 2014) shows that there are a total of 356 pedestrians involved in crashes along the highways from 2010 to 2014. The most accidents occur on a Friday with the peak times at 7:00am and 7:00pm, give or take an hour. The survey also shows that the number of people that use the pedestrian bridges exceeds by far the number of pedestrians that cross the freeway at grade in the same area. With 8 666 pedestrians crossing the freeway and 15 363 pedestrians crossing by means of pedestrian bridges on a typical weekday.

In South Africa, there is a need to better understand the nature and number of pedestrians that use the highways or pedestrian bridges daily. In Cape Town along the freeway system alone there are about 300 surveillance cameras streaming

ISBN Number: 978-1-920017-64-4

directly to the Transport Management Centre (TMC). These cameras form part of the ITS infrastructure for the Cape Town Freeway Management Systems Project jointly funded by SANRAL, the Western Cape Provincial Government and the City of Cape Town. This means that there is an opportunity for pedestrian data-mining. Currently pedestrian statistics are collected from the system's existing camera infrastructure, but requires employees to manually count every pedestrian from a historical video feed if the data is needed. The problem with the above-mentioned technique is that it is extremely time consuming and monotonous work, where resources could be allocated more effectively. Alternatively there are modern camera systems that could automate the counting process, but requires specialised hardware and software, such as high-end thermal cameras, specialized image processors, as well as the supporting infrastructure. The goal of this paper is to propose an alternative method that uses existing equipment and infrastructure, while making minimal infrastructure modifications.

This paper introduces a method with which to automate the counting process using state-of-the-art computer vision and supervised learning techniques. These techniques are then applied to SANRAL's existing surveillance infrastructure, specifically on cameras monitoring the pedestrian bridges, with the option of expanding to monitoring pedestrians along the highway. The system can also be expanded for safety and security applications such as cable monitoring, as it would be able to detect thieves. It is currently not possible to monitor the pedestrians on the highway due to too low camera resolution, as well as operational factors, such as operators controlling the cameras when road incidents occur. This paper looks at the effect that different camera configurations has on the observed detector performance. The paper also suggests parameter settings and reflects on broad requirements for the detector to obtain satisfactory performance in this case. This could possibly feed into enhanced specifications for future camera upgrades. Pedestrian detection using existing infrastructure is an especially challenging problem due to the low-resolution of the existing video feed. Figure 1a, and figure 1b shows a snapshot from dataset-1 and dataset-2 respectively.

Figure 1a: Dataset-1 screenshot from existing infrastructure on R300

Figure 1b: Dataset-2 screenshot from existing infrastructure on R300

2. RELATED WORK

The field of computer vision and non-rigid (pedestrian) detection is rapidly expanding, with new techniques and higher benchmarks being set in quick succession. Various pedestrian detection techniques have been proposed over the years, with the birth of modern pedestrian/face detection stemming from the use of Haar-wavelets (Viola & Jones, 2001). This technique uses key (interesting) features. These features are obtained by applying predefined filters to images. The key components are then normalized over the dataset and used to train a classifier or detector model.

In a paper (Dalal, et al., 2004) suggested HOG (Histogram of orientated gradients) for the application of pedestrian detection. Many today see this as the basis of most pedestrian detection algorithms, and is widely considered as the birth of practical multi-scale pedestrian detection. HOG computes the orientation and magnitude information of image edges to construct a histogram of orientated gradients, or edges. The image is densely scanned at multiple scales by up- and down-sampling the image to create a so-called image pyramid of image gradients. The training method proposed is a supervised SVM (simple vector machine) method. When annotating, a bounding box is drawn around the object to indicate an object. The data contained within the bounding box is then extracted from the image as a positive sample, and multi-scale image gradient, or edge, histograms are collected and added together to create a large vector. This vector is then normalized to give a basic framework of how the objects profile looks like.

Annotations are metadata that indicates that the region inside the bounding box is the object, and after enough different positive- (containing object, e.g. human) and negative (not containing objects e.g. mountain landscape) images an average gradient, or edge, orientation profile can be created. This is called training. A probabilistic model is used to calculate how likely the input image matches the object profile.

A detector or classifier's performance is measured with the term hit rate. This states the percentile of pedestrians that will be detected (hit) out of the total amount of pedestrians there are. Some graphs use miss rate, which is one minus hit rate. When the detector accurately detects an object it is called a Tp, or true positive. A true positive is when the detection is the same as the annotation. An Fp, or false positive, would be a false detection, say detecting an object that is not there. The detections are compared to the annotated, or ground truth data. The same goes for Tn, true negatives, and Fn, false negatives.

The paper by (Benenson, et al., 2014) shows popular detectors from the last decade that are compared experimentally. Figure 2 adapted from (Benenson, et al., 2014) below shows nine of the most used pedestrian detectors. The (I) and (C) next to the detector name shows that the detector is trained with the Inria and Caltech standard pedestrian datasets. The article compares the log-average miss rates of various detectors. *VJ*, (Viola & Jones, 2001), scores lowest with a miss rate of 94%, while *HOG*, (Dalal, et al., 2004), has a miss rate of 78%. The method used in this paper, *SquareChnFtrs* (Dollár, et al., 2009) scored 58% when trained on the Inria dataset, but scored a miss rate of 11% when trained with the Caltech dataset. Products on

the marked today typically have a miss rate of 10%-30% depending on the environment and camera. It is clear that the dataset, and detecting algorithm used play vital roles in the accuracy of the detector. The above mentioned paper also gives recommendations for parameter settings determined empirically. Some of these parameters are used in this paper, with slight adjustments made to obtain optimal performance.

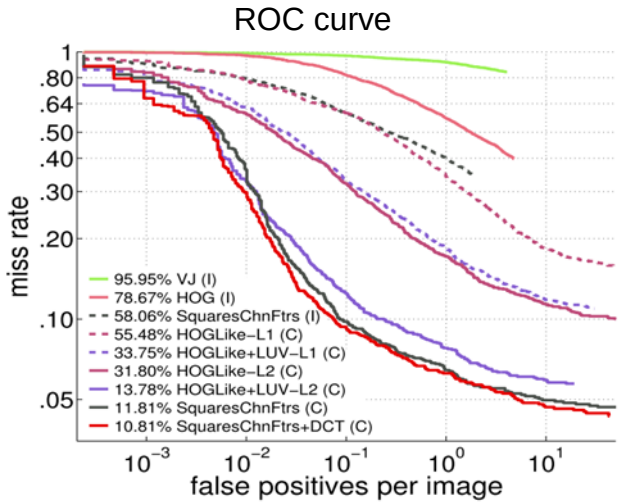

Figure 2: ROC curve of 9 modern pedestrian detectors as adapted from (Benenson, et al., 2014). Tending to lower-left corner is better

Integral channel features in its simplest form uses the same basics as the *HOG* detector but instead of only using gradient, or edge, orientation and magnitude information, it combines colour space (RGB/LUV/CMYK) channel features with *HOG* data to add more fields of information, and thus increases the robustness of the algorithm. The channels of the detector are shown in figure 3 as adapted from (Dollár, et al., 2010). One of the key features of subsequent detectors is that the image pyramid is only constructed for a few scales and the rest of the information can be extrapolated from that. This saves a lot of computational time as densely scanning the detector at different image scales contributes a large portion of the total runtime. A comparison of the runtime and accuracy can be seen in (Dollár, et al., 2010). The big upside to this method is that detection can be done in/close-to real-time if set up correctly.

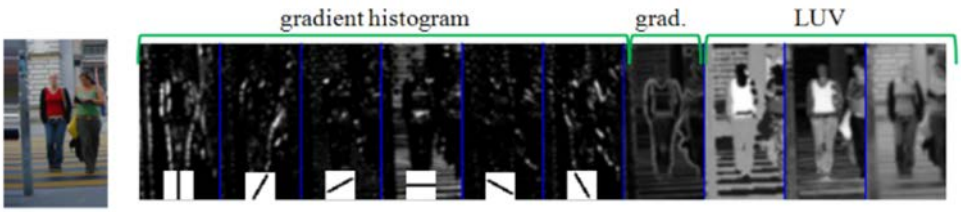

Figure 3: Representation of detector channels. Firstly the sample is shown, then the gradient orientation of the HOG bins. Next the total image gradient and the three LUV colour channels are shown as adapted from (Dollár, et al., 2010)

For training, an advanced AdaBoost (adaptive boosting) algorithm is used. This is derived and adapted to modern applications, from the early works of (Viola & Jones, 2001). AdaBoost is a method of training soft-cascades. It acts as a decision tree to

classify the detections as object or non-objects. At each cascade it detects all the objects it can with its set of parameters, and allows the rest through to the next stage. Each stage performs more dense detection scans, at the cost of computational time. This can be seen as muddy water, (complete dataset) falling through a multi-stage filter, cascade, with increasing density. The first coarse stage collects all the large and easy to filter material, (positive samples) quickly, while the last fine stage takes a long time to remove the dirt, from the water, ultimately leaving fresh water, (negative samples). It then becomes an optimization problem to assign weights to positive and negative samples and obtain a detector algorithm.

Another adaptation made by (Dollár, et al., 2014) to the original *Integral Channel Features* algorithm is that there are pre-training options available for applying smoothing filters such as Guassian, as well as applying jitter to positive samples amongst others. Jitter means that operational transforms are applied to the positive samples such that new positive samples can be generated from the original. This is done by e.g. mirroring the image, rotating it slightly, or applying slight distortion whereby unique positive samples are generated (Nam, et al., 2014).

To summarise, the first practical detector was introduced in (Viola & Jones, 2001). Then (Dalal, et al., 2004) introduced *HOG* which used pedestrian silhouette profiles as features. In the paper (Dollár, et al., 2009) *HOG* is used together with colour channels and the training method of *VJ* to form a detector called *Integral channel features*. Further adaptations where made to the detector of *VJ* to improve runtime and accuracy significantly, (Nam, et al., 2014), (Dollár, et al., 2009). Continuous adaptations and additions to the original algorithm of *VJ* and *HOG* makes the detector used in this paper one of the best performing detectors available today.

3. METHODOLOGY

In this paper an automated approach to pedestrian detection and counting is proposed using computer vision techniques. The footage used for training and testing is obtained from SANRAL's existing surveillance cameras monitoring the pedestrian bridges along the N2 and R300 in the Cape Town region. Figure 4 shows an overview of the methodology used, with three distinct stages, pre-processing the data, training the classifier, and performance testing.

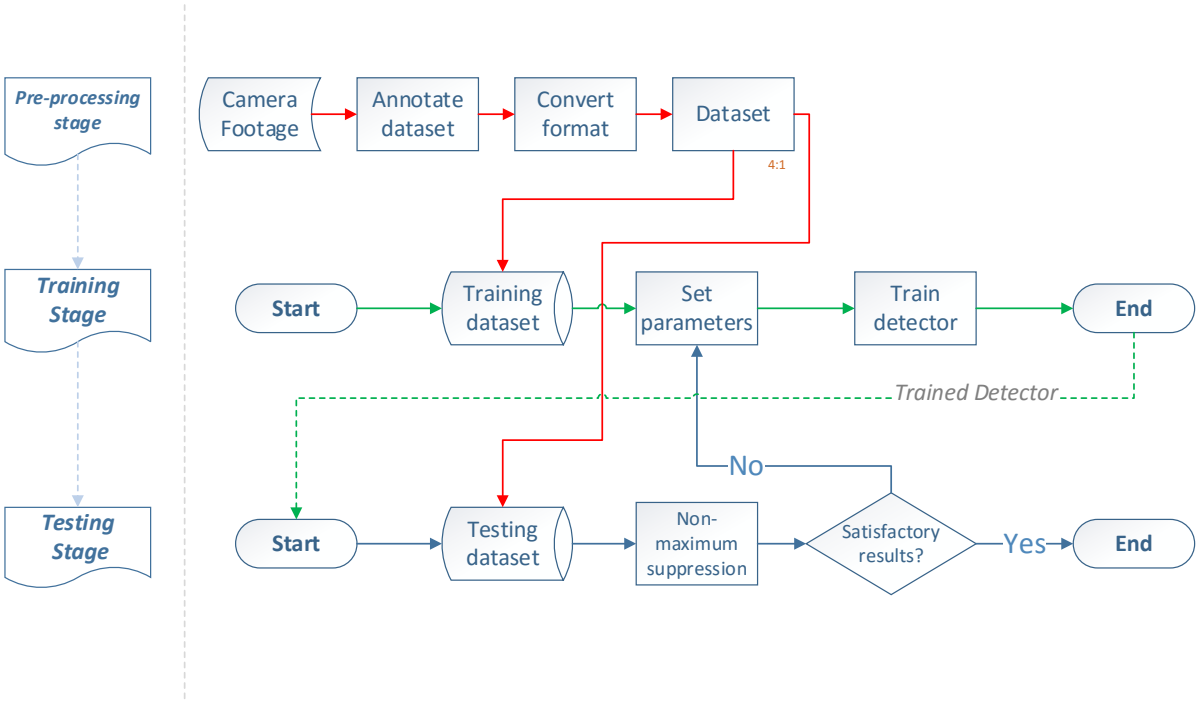

Figure 4: Flow diagram of respective stages

3.1 Pre-Processing Stage

Care had to be taken that the exported videos do not contain downtime or signs of packet loss, which introduces unwanted noise or losses in the dataset. These losses are introduced by the conversion of the real-time to the historical feed for archiving. Archiving greatly reduces the frame-rate and resolution to that of a specified amount contractually required by the project specifications. This was, at the time of dataset collection, around 320x240 at 12.5fps (now 480x272 at 5fps), with pedestrian sizes averaging at 36x28 pixels (now 30x15). Care had to be taken to ensure that the camera stays in focus throughout the required period. Care also had to be taken that the weather is of such condition that a human can easily be able to distinguish between background and pedestrian. This is referred to as fair/reasonable conditions in computer vision, and represents a best-case scenario.

The MATLAB toolbox for pedestrian detection by P. Dollár is used, and includes the integral channel feature algorithm (Dollár, et al., 2009), together with the adaptations made as discussed in the previous section (Nam, et al., 2014), (Dollár, et al., 2010), (Dollár, et al., 2014). This also includes an optimized variation of the AdaBoost algorithm for training, as well as a basic annotation tool. The toolbox was used as it provides all the tools needed for pedestrian detection, together with speed optimizations made over years of continuous improvement.

The video files need to be converted to .seq format for annotation. Changing the format converts the video file into a large sequence of images that can easily be skipped through without the need to decode each frame. The reasonable .seq files then need to be annotated by hand. Labelling is still very time consuming as only about 70 frames can be annotated at a time, with 12 hours of video accumulating to 600 000 frames. It must be noted however, that only a fraction of the frames contains

objects of interest i.e. pedestrians.

The annotation should be done with great care as this has a big impact on the performance of the detector. If frames are miss-labelled there is no ground truth data, which introduces false negatives in the training stage, as well as being classified as a false positive in the verification stage. The misclassification would occur because the detector would still detect the object but would be classified as incorrect compared to the ground-truth. After annotation the .seq file and corresponding ground-truth file are extracted to .txt and .jpg files respectively.

The annotated dataset is then divided into training- and testing-datasets, with a good starting ratio being 4:1 respectively. The training set should cover all scenarios being tested in the testing-set, e.g. pedestrian walking up stairs, to avoid misclassification. The optimal size of training- versus testing-dataset size can be determined empirically by comparing validation performance.

3.2 Training Stage

Training is to “train” a classifier comprised of a soft-cascade to recognize pedestrians by “showing” the algorithm a number of positive samples, images with pedestrians in this case. This is similar to training a child or pet. Before training can commence the parameters should be set up appropriately, with most parameters giving a trade-off between accuracy and speed. Only significant parameters will be discussed here as there are far too many for the scope of this paper. A detailed discussion and empirical results of generic default values are shown in (Dalal, 2006) for the *HOG* parameters and in (Dollár, et al., 2009), where different parameters and their impact on different kinds of detectors, and training methods are compared. Most of the parameters used are set to its default value as recommended by P. Dollár in the respective papers used as the basis of the MATLAB toolbox.

Firstly the minimum model size of expected pedestrians should be set, this helps to determine the starting scale used in creating the image-pyramid. For this paper the pedestrians are observed to be an average of 32x26/24x18 pixels while annotating respective datasets. Next the amount of padding is chosen. Padding adds a set number of blank pixels around the positive samples as to avoid boundary conditions while densely scanning the image. The model size with padding uses the same ratio as the recommended 1:1.25, which equates to total sample sizes of 40x32/30x22 respectively. The window stride length is how many pixels the sliding window densely scanning the image, moves to form each detection window. This has a big impact on performance, as this determines how much each consecutive detection window overlaps. In (Dalal, 2006) is shown that using overlapping windows reduces the detector miss rate by up to 5%. The stride length used was 4 pixels as this was seen to give the best performance in preliminary tests.

To improve performance the feature channels are convolved by a set of ‘filters’ in order to remove local correlations, and thus removing data that could skew the detector. This slows performance, as convolution of 9 large image channels with even a simple filter is computationally complex (Nam, et al., 2014). From the original paper this is seen to have significant improvements on accuracy. The default value of a [4 5] filter is used as discussed in detail in (Dalal, 2006).

Training commences in 4 stages, with training alternating between sampling and training an AdaBoost classifier. Each stage has an increasing number of trees, or weak classifiers. There are initially only 32 trees in the first stage, and 2048 in the last stage. Ultimately all the stages are cascaded, to form a soft-cascade classifier. It is shown in (Dollár, et al., 2009) that any increase in trees or stages will result in diminishing returns.

3.3 Testing Stage

In the testing stage the detector is given a set of images where the ground truth data is known, and the performance of different parameter changes can be evaluated based on the detector results.

Lastly the type of non-maximum suppression (NMS) should be chosen. This is the act of reducing multiple bounding boxes drawn at different scale, to one bounding box with the highest score/probability. This is critical as this brings together all the steps to ultimately give detections in the form of a definitive answer, a single bounding box for a single detection. The toolbox offers 4 methods, being mean-shift, absolute maximum, an optimized absolute maximum algorithm, and cover suppression. This comes down to an optimization and cost assignment problem which is outside of the scope of this paper.

The tracking and counting procedure is still to be implemented. The output of the non-maximum suppression method is used as absolute detections, and the position of the detected pedestrian being represented by a single bounding box. If a pedestrian move past a used defined rectangle, say in the middle of the bridge or video feed, a counter is incremented. This means that the relative number of crossings are counted instead of the absolute number of unique pedestrians that use the pedestrian bridges. Provision will be made to record the time and direction at which the counter is incremented, as to provide a relationship between the number of crossings in a direction and the time of day/ day of the week.

4. RESULTS

Results are collected in the verification stage and by inspection. In the verification stage the detector is applied to the test dataset, and the detection results are compared to the annotated ground truths. The verification results is represented in Receiver Operating Characteristics (ROC) - and in Precision/Recall, (PR) - graphs.

A ROC graph plots the miss rate, against the number of false positives per image, FPPI. FPPI can be set for a specific detector by adjusting the threshold in the NMS-parameters. This is useful in applications where little to no false detections may occur, or setting the FPPI high where false detections are acceptable. It is clear that the detector behaves differently depending on the threshold score, or strictness, required to be classified as a positive. A good ROC curve tends to the left bottom corner of the graph.

Miss rate: $$\frac{Fn}{Fn+Tp} \quad (1)$$

A precision-recall graph gives a different view of the data. The precision is compared against the recall rate. Here nothing can be explicitly set and evaluated, but a relationship between the total detections and a ratio of true/false positives can be seen. A good PR curve tends to the right-upper corner of the graph.

Recall: $$\frac{Tp}{Tp+Fn} \quad (2)$$

Precision: $$\frac{Tp}{Tp+Fp} \quad (3)$$

Firstly both detectors are trained and tested on their respective validation sets, with dataset-2 performing significantly better than dataset-1. The improvement is because the silhouettes in dataset-2 is more clearly visible against the background. It is clear that the camera orientation with relation to the bridge plays an important roll. Pedestrians in dataset-1 also tend to be more occluded by other pedestrians and the terrain. The best hit rate performance for both detectors evaluated in this paper was 61% and 72% for dataset-1 and dataset-2 respectively. Detector-1 was also tested on dataset-2 and visa-versa which results show that the detectors need to be trained for individual scenes, the proof has been omitted as the respective PR graphs are a straight line at 0, i.e. 100% miss rate. This is due to the nature of the low-resolution video feed, or the difference of camera orientation, as dataset-1 views pedestrians from the front, and dataset-2 views them from the side. Both detectors obtain miss rates in the high 90% region when validating on the opposing test-set. Specifically trained detectors outperform pre-trained detectors dramatically, with the best state-of-the-art pre-trained detector obtaining a miss rate of 85%.

The size of the dataset is then tested in conjunction with slight parameter changes in model-size and padding. Figure 5 shows a dramatic improvement in PR characteristics with the increase in dataset size, the blue line, and only a slight decrease in performance when using less optimal parameters, the red line. This shows that there is a correlation between the performance of a detector and the training set size up to a certain point. This is because an increase in dataset size means that there is a bigger feature pool with more positive and negative samples. Further increases in size results in diminishing returns as the classifier becomes saturated.

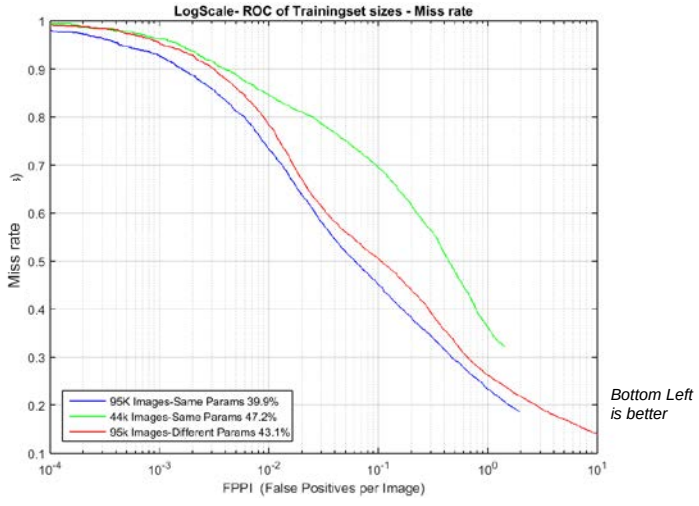

Figure 5 Log scale-ROC curve of detector performance with different parameters and training dataset-1 size.

The choice in non-maximum suppression method is important as this takes all detections at multiple scales in consideration to determine a single bounding-box output for a detection. Thus the Nms has a significant impact computational time and detector accuracy. The 4 methods available from the MATLAB toolbox by P. Dollár are compared using dataset-2, as dataset-2 gives the best case scenario. Figure 6a

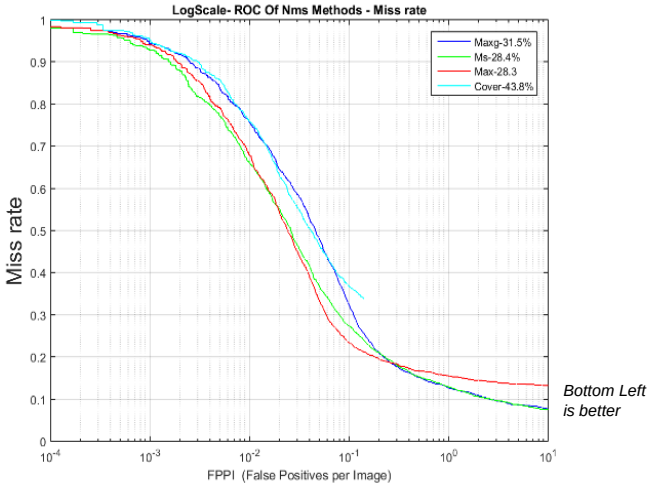

Figure 6a Log scale-ROC curve of non-maximum suppression methods applied to

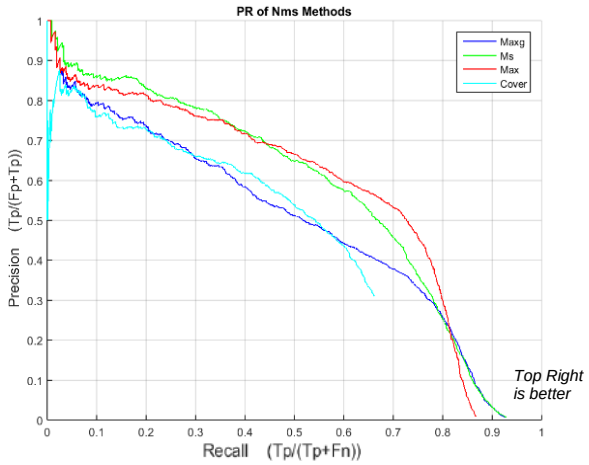

Figure 6b PR curve of non-maximum suppression methods applied to

shows the log-average ROC curve for the various cases. Figure 6b shows the PR curve. The method, *Max*, can be seen as the best Nms-method with a miss rate of 28.3%, but because *Ms*, mean-shift, is less computationally expensive and is thus chosen for this paper.

The hit rate performance of the detector degrades as the number of pedestrians being detected increases. This is due to occlusion that occurs between pedestrians when they walk past each other. Occlusion is when terrain or other objects partially block the view of the object that needs to be detected. Figure 7 shows the detection results when one, or two detections are made simultaneously.

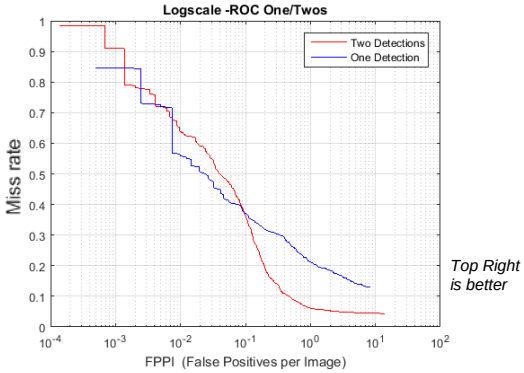

Figure 7 **Log scale-ROC curve of detector performance when only one, or two pedestrians are being detected at a time respectively.**

This would be when one or two pedestrians use the pedestrian bridge. The detector performance would degrade further with more pedestrians, but due to low-resolution problems distinguishing between pedestrians in larger groups this could not be tested. The detector performs similarly in both cases when the FPPI is low, but

quickly diverges above 1 FPPI, with one detection being significantly more accurate than the latter. Subsequent alterations has improved detection while occluded dramatically, by including more occluded pedestrians in the dataset.

5 RESEARCH FINDINGS

This paper discussed the background and methodology used to implement a pedestrian detector. Recommendations for parameters and other features, as well as some shortcomings of the system are also discussed. After reviewing the results it can clearly be seen that the detector is influenced by a number of important factors, from the dataset size to the number of detections at any given moment. The optimal parameters can be empirically determined as each scene is unique with the pedestrian model sizes and number of required positive samples varying from case to case.

The detection performance improved drastically as the training-dataset size was doubled as can be seen in figure 5. The optimal size has to be empirically determined as it comes at a cost of training time. A training dataset with 95 000 images takes up to 30 hours to train a soft-cascade classifier with the AdaBoost algorithm. It is also clear that each detector needs to be trained for each specific scene and camera orientation, which does mean that some time will have to be spent annotating video footage.

The verification results show that the camera with a visible pedestrian silhouettes, dataset-2, performs significantly better than dataset-1. Although this is not a complete comparison it shows that camera orientation plays a vital role. This is to be expected as a big part of the detection algorithm comprises of the *HOG* feature set. Thus the silhouettes of pedestrians contain important information. With a peak hit rate performance of 72% dataset-2 is competitive with high-end products, even though the detector is working with a video feed of half the resolution of that of high-end products.

Problems still arise when big groups of bundled pedestrians crosses the bridges. It is still difficult for a human to distinguish where one pedestrian ends and the other starts in such crowds. There are thus instances where the dataset can be seen as unreasonable. This is however only a problem during the peak pedestrian traffic. Other unreasonable conditions include rainy weather or out of focus cameras.
Although this application is far from a practical integrated solution for most of the existing infrastructure, it does serve as a proof of concept that, under the ideal conditions, it can be done. With this in mind future camera installations could be done in such a way as to enable the use of automatic pedestrian detection, which is a step in the right direction for pedestrian safety and monitoring.

REFERENCES

Benenson, R., Omran, M., Hosang, J. & Schiele, B., 2014. Ten Years of Pedestrian Detection , What Have We Learned ?. *European Conference on Computer Vision - ECCV.*

Dalal, N., 2006. Finding People in Images and Videos. *INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE.*

Dalal, Navneet, Triggs & William, 2004. Histograms of Oriented Gradients for Human Detection. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR05,* Volume 1, pp. 886-893.

Dollár, P., Appel, R., Belongie, S. & Perona, P., 2014. Fast Feature Pyramids for Object Detection. *PAMI.*

Dollár, P., Belongie, S. & Perona, P., 2010. The Fastest Pedestrian Detector in the West. *Procedings of the British Machine Vision Conference 2010,* pp. 68.1-68.11.

Dollár, P., Tu, Z., Perona, P. & Belongie, S., 2009. Integral Channel Features. *BMVC 2009 London England,* pp. 1-11.

Nam, W., Dollár, P. & Han, J. H., 2014. Local Decorrelation For Improved Detection. *Nips,* pp. 1-9.

SANRAL, Western Cape Goverment, TCT, November 2014. *Pedestrian survey and analyses for western cape freeways,* Cape Town: s.n.

Viola, P. & Jones, M., 2001. Rapid Object Detection using a Boosted Cascade of Simple Features. *CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION 2001.*