



Technologies for Safe & Efficient Transportation

THE NATIONAL USDOT UNIVERSITY
TRANSPORTATION CENTER FOR SAFETY

Carnegie Mellon University

UNIVERSITY of PENNSYLVANIA

Pedestrian Friendly Traffic Signal Control

FINAL RESEARCH REPORT

Stephen F. Smith (PI), Gregory J. Barlow,
Hsu-Chieh Hu, Ju-Hsuan Hua

Contract No. DTRT12GUTG11

DISCLAIMER

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the U.S. Department of Transportation's University Transportation Centers Program, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof.

1. Problem

This project continues research aimed at real-time detection and use of pedestrian traffic flow information to enhance adaptive traffic signal control in urban areas where pedestrian traffic is substantial and must be given appropriate attention and priority. Our recent work with Surtrac [12], a real-time adaptive signal control system for urban grid networks, has resulted in an extended intersection scheduling procedure that integrates sensed pedestrians and vehicles into aggregate multi-modal traffic flows and allocates green time on this integrated basis [17]. In this project we consider the companion problem of providing the pedestrian sensing capability necessary for effective use of this extended intersection scheduling procedure.

Although some commercial pedestrian detection and counting capabilities do exist, they typically require the purchase and installation of additional higher resolution video camera technology, which can double the cost of detection per intersection. Our interest is in a solution that does not significantly increase infrastructure cost. The hypothesis investigated in this work is that lower resolution vehicle detection camera technology can be used to provide a relaxed form of pedestrian count data that is sufficient for incorporating pedestrian flow information into real-time intersection scheduling. Specifically, we study the possibility of extracting an approximate but usable measure of pedestrian “density” from the video stream of a commercial traffic camera. Our target functionality is the ability to qualitatively discriminate between “no”, “few” or “many” waiting pedestrians. Contemporary traffic camera technologies provide resolution as low as 320×240 gray scale images (see Figure 1), together with the ability to specify and monitor a set of occupancy zones within the image.

Pedestrian detection and counting is not a hard task for humans, but it is challenging for computers. The challenges include diverse shapes and occlusion among pedestrians, a dynamic background, and low video quality. First, pedestrians can have various appearances because of clothing, accessories, assistive devices, and change of pose while walking. This high intra-class variation, as well as occlusion, makes pedestrian detection a hard problem. An alternative to classification based pedestrian detection is to find foreground pixels in each frame of the video, and analyze those pixels to estimate the number of pedestrians. However, since the system is deployed in an outdoor environment, shadows caused by moving objects or sudden change in illumination can create noise that complicates foreground detection. Moreover, to get a broader view of the intersection, the camera is installed at a certain height. Thus, pedestrians are small in the images and have fewer details for computer vision processing.



Figure 1: Sample images taken by the CitiLog XCAM-ng Camera.

These challenges have made pedestrian detection and counting a topic of interest in computer vision research for many years, and several classes of techniques have been developed. At the same time, this field of research has invariably assumed the availability and use of state of the art detection hardware. The concept of “pedestrian detection on a budget”, as typified by our goal of exploiting existing vehicle detection camera technology to do double duty and detect waiting pedestrian volumes, has received little attention.

2. Approach

To address the problem of pedestrian density estimation, a two-stage approach was taken. In the first stage, we analyzed the potential of various detection techniques developed within the field of computer vision as a basis estimating pedestrian density from low-resolution traffic camera technology. For most existing techniques, effectiveness is tested on images captured with our XCAM-ng camera using an open source computer vision library of detection techniques. In the second stage, after finding no satisfactory approach to this problem with existing computer vision detection techniques, we developed and tested a more motion-based approach to detection. Under this approach the video image is segmented into a dense grid of occupancy zones, and detection of qualitative pedestrian density levels is formulated as a sequence classification problem. We consider each of these two stages of the approach in turn in the subsections below.

2.1 Phase 1: Examination of existing computer vision approaches

One popular approach to pedestrian detection within the computer vision literature locates a human body or body parts by first computing shape features over the image and then applying a classifier on image patches. Shaped-based detectors and classifiers have been shown to be effective on the task of human detection, yet they depend on a large amount of training images that are well classified. A second way to look at this task is based on the fact that the camera is static, and the only moving objects in the scene are pedestrians and vehicles. This fact leads to the idea of using background estimation and subtraction techniques. However, since the system is deployed outdoors, the scene often experiences sudden change in lighting and background subtraction generally performs badly in this case. A third approach

computes optical flow, which represents object movements in frames and detects humans from it.

In the subsections below we examine the suitability of each of these approaches in more detail. To support this task, we make use of the OpenCV software suite of computer vision techniques [1] and images collected by a Citilog XCAM-ng camera that has been provided by Citilog (now AXIS).

2.1.1 Shape-based Detection

Histogram of Oriented Gradients - Dalal et al propose a human detection framework in [4] that first computes the histogram of oriented gradients (HOG) descriptors of an image and classifies it with a support vector machine (SVM). The HOG descriptor consists of orientation and spatial binning of gradients within grids. In [4] it is shown that a detection window with 64×128 in size and 16 pixels of margin around the person gives the best result in pedestrian detection.

We tested the static HOG pedestrian detector using our videos with the `HOGDescriptor` class defined in OpenCV, but the detector failed to identify the pedestrians. This is because pedestrians in the captured scene are too small for the 64×128 detection window. OpenCV also provides a `people` detector with detection window size 48×96 , but people can still be smaller since we expect the camera to be mounted to overlook the entire intersection.

In [5], the use of HOG descriptors is further extended to detect pedestrians in videos. The detector computes oriented histograms not only on appearance but also on optical flow. The two proposed families of oriented histograms descriptor on motion are Motion Boundary Histograms (MBH) and Internal Motion Histograms (IMH). The experiment in [5] shows that combining static HOG descriptors with one of two variants of the IMH family performs the best. However, our assessment is that this enhancement will still not be sufficient in our low resolution camera setting.

Viola-Jones Detector - Another widely used object detection approach is the cascade classifier proposed in [13]. First, Haar-like features are computed at different scales of an image, and important features are selected using AdaBoost. Then, a cascade classifier detects and locates objects. Experiments were performed with OpenCV's implementation of the `CascadeClassifier` class using its predefined models for the cascade detector, (e.g., full human body, lower body, and upper body). However, the cascade classifier fails to detect pedestrians or their body parts in our videos.

The authors also propose in [14] a pedestrian detector using features of the static images used in [13] together with motion patterns. The motion patterns are pixel value differences between the frame at time t and the frame at time $t + 1$ shifted in four directions. The cascade pedestrian detector also uses AdaBoost in training. In the experiment in [14], the training set consisted of 2250 positive examples and 2250 negative examples, and each example is a pair of consecutive frames. This detector works well with a detection window of size 20×15 , which is similar to the size of pedestrians in our videos. We were not able to test this extended technique, but the approach that we pursue below similarly emphasizes the use of motion patterns.

Challenges - Most of the pedestrian detectors described above use examples of normal-looking pedestrians that are standing upright, yet this is not always true in real world settings. Instead, pedestrians may be sitting in a wheelchair or wearing animal

costumes, and these examples seldom appear in pedestrian datasets. In fact, it is impossible to anticipate all possible pedestrian appearances in the training set, and hence there is always the possibility of detection failure. Moreover, vision-based pedestrian detector cannot be directly applied to other types of sensors, such as radar or infrared traffic sensors.

2.1.2 Foreground Detection

Another perspective in thinking about our pedestrian density estimation problem is that since the camera is static, the background stays the same except for illumination. If it is possible to determine the foreground, then there is no need to scan through the whole image to look for pedestrians, making the computation efficient. Several vision-detection techniques are based on this viewpoint.

Background Estimation and Subtraction- OpenCV provides

BackgroundSubtractorMOG2 which implements the techniques first put forth in [20] and [21]. The method uses a Gaussian mixture model to estimate background and foreground, and updates its estimates at each frame. To reduce the effect of noise in the environment, Gaussian smoothing is performed on a frame before applying the background subtractor to it, and a morphological opening is applied to the foreground mask. (Morphological opening is image erosion followed by dilation.) This operation is useful in removing noise that occupies a few pixels without disrupting the original shape of objects. Overall, the background subtractor gives good results, but noise increases over time. In addition, this approach is prone to shadows and sudden change in illumination.

Candes et al [2] have shown that robust principal component analysis works well for background subtraction in the case of a sparse foreground. A video sequence can be represented as a large matrix M , whose columns (or rows) are pixel values of video frames. If the background is static, we can decompose M into a low-rank matrix L , which corresponds to the stationary background, and a sparse matrix S , which captures the moving objects in the foreground. Computation of L and M can be formulated as a convex optimization problem and solved by an augmented Lagrange multiplier method. However, since singular value decomposition must be performed each iteration, the computation is too expensive for our application. Hence we rely on simple BackgroundSubtractorMOG2 to provide a starting point for “blob” detection (see below).

Blob Detection - After the foreground mask is found for a scene, objects or groups of objects can be located in the scene by performing blob detection on the foreground mask with OpenCV's findContours function. Regions that do not have the appropriate shape can then be discarded, and the number of pedestrians within a blob of each frame can be estimated. To adjust for discontinuity in the result caused by noise or sudden change in illumination, a moving average or majority vote over a short sequence of frames can be computed to get a more robust answer. Sample results are illustrated in Figure 2.



Figure 2: Sample results from blob detection and filtering. Detected pedestrians are marked in red.

Regression on Foreground Pixels - Another approach to estimating the number of pedestrians in the foreground is to use regression. The methods in [3,9] compute features and area of the foreground masks of video frames as input to a regression function. Using a regression function is efficient, yet the answer is valid only when the scene consists solely of pedestrians and the shadows are light. Hence, it does not really provide a general solution and is probably more suitably applied as a heuristic when it is known that there are pedestrian groups in the foreground.

Pedestrian Detection in Occluded Scenes - Wu et al [16] formulate the pedestrian detection problem as a maximum a posteriori problem. They introduce a silhouette-oriented feature called an “edgelet” to detect body parts, and responses from edgelet part detectors are combined to form a joint-likelihood model for detection and location of possibly occluded people. Like Viola and Jones detection framework, the part detectors are trained with an enhanced boosting method.

Challenges - Background subtraction methods are sensitive to sudden changes in illumination because they classify foreground pixels by color values. Sudden changes in illumination include shadows and change in intensity of sunlight caused by clouds, which are very common in outdoor environments. Using only the foreground mask to count pedestrians would likely lead to failure in providing a close estimate. However, since precision in counting is not our concern, there may be ways to deal with the problem of illumination and shadows, such as keeping a short history in the background model, and performing morphological opening and closing to detach shadows from humans. A different challenge is that when the foreground is crowded with pedestrians and vehicles, it is difficult to separate them using only the foreground mask.

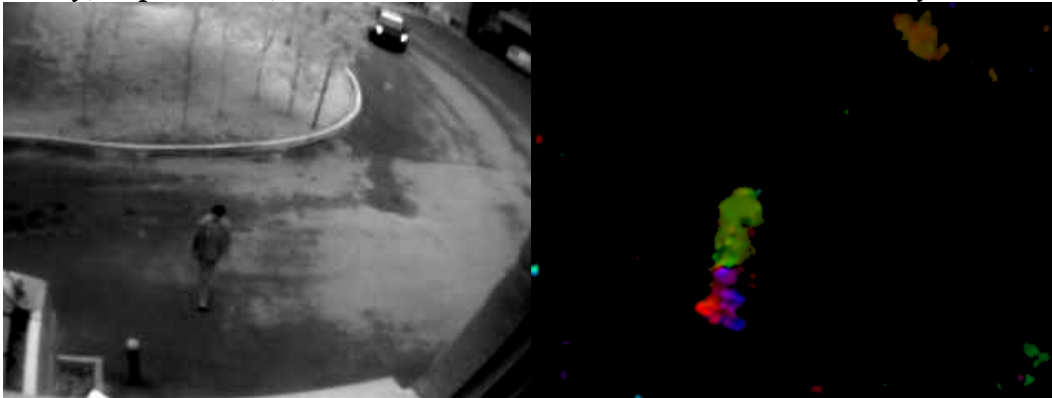
2.1.3 Motion Analysis

One drawback of background subtraction is that the foreground mask may include non-pedestrian objects that are not of interest. If we think more about what happens at the intersection, pedestrians move towards the intersection before crossing the road. Therefore, detecting movements near the intersection may help us count.

Dense Optical Flow - One way to detect and count pedestrians in video streams is to find moving objects in the scene by computing dense optical flow. OpenCV’s `calcOpticalFlowFarneback` implements Farneback’s method [8] for computing dense optical flow in which the displacement between two frames is estimated by

approximating the neighborhood of both frames with quadratic polynomials. The result is a vector field where each vector describes the magnitude and direction of motion at a pixel (Figure 3). The noise can be removed by setting a threshold on the magnitude of the vectors.

Challenges - We can count the number of awaiting pedestrians by detecting movements towards or away from the intersection. However, optical flow does not detect stationary pedestrians, which can be an issue to the robustness of our detection capability, in particular, when the movements occur outside of the analysis window.



(a) Original frame

(b) Optical flow vector field. Intensity implies magnitude in movement and color describes direction of movement

Figure 3: Optical flow

2.1.4 Summary

We have considered the suitability of three categories of pedestrian detection methods that have emerged from the computer vision community for the task of counting pedestrians at road intersections. Shape-based detectors identify humans by features on the shape of humans or body parts. Shadows and change in illumination do not affect shape-based detectors much if we assume the plane that pedestrians move on is known. However, such detectors rely on a large amount of training data that includes all possible appearances of human to be robust, which is implausible. Background subtraction is faster than detectors, but it is prone to sudden change in illumination and shadows. Unlike detectors and background subtraction, motion analysis methods like optical flow do not need a model on pedestrians or the background, but stationary pedestrians do not get detected. Overall, none of these techniques provide a satisfactory basis for estimating pedestrian density from vehicle cameras.

2.2 Phase 2: Pedestrian Density Estimation via Sequence Classification

Given the inability to find a satisfactory solution from prior research in computer vision, a more motion-based approach to pedestrian counting was defined and analyzed, based on the use of a dense grid of occupancy zones that are drawn over the video frame image. Figure 4 illustrates the idea with a 3 x 3 grid drawn over the corner of an intersection. As pedestrians pass through this region and wait for the crossing phase, those grid zones occupied by pedestrians will be activated. By capturing grid zone activation values periodically, sequential patterns of activation can be constructed and subsequently used to develop a qualitative pedestrian density classifier (e.g., empty, medium or high density). In operation then, the classifier is applied to the sequential pattern of activation zones accumulated over a given horizon (e.g., the time since the last green phase), and the timing of the traffic light is controlled according to the (qualitative) number of pedestrians detected. A possible pattern of two pedestrians is shown in Figure 5. At certain time steps, two or more pedestrians can trigger a single detection zone. It is also possible for a pedestrian at the boundary of two zones to trigger both. However, we expect that the sequence of activated zones will produce patterns that correct such ambiguous activations and enable extraction and detection of accurate density information.

2.2.1 Sequence Classification

We formulate the sequence classification problem as follows: A given sequence of activation patterns is associated with a single class label and the whole sequence is available to the classifier at classification time. More formally, the problem is to predict a single label l given the input sequence of activation zone patterns

$$(p_1, p_2, \dots, p_n).$$

Sequence classification methods can be divided into several categories [7,15]. Feature-based classification is a class of methods that maps an input sequence to a feature vector, and then applies conventional classification methods such as support vector machine (SVM) learning. Feature selection plays an important role in this type of method. For other kernel-based methods, it is not necessary to explicitly

conduct feature selection. Instead a kernel function is used to expand the feature vector to a high dimension feature space and it is possible to simply measure similarity between two time series of activated patterns.

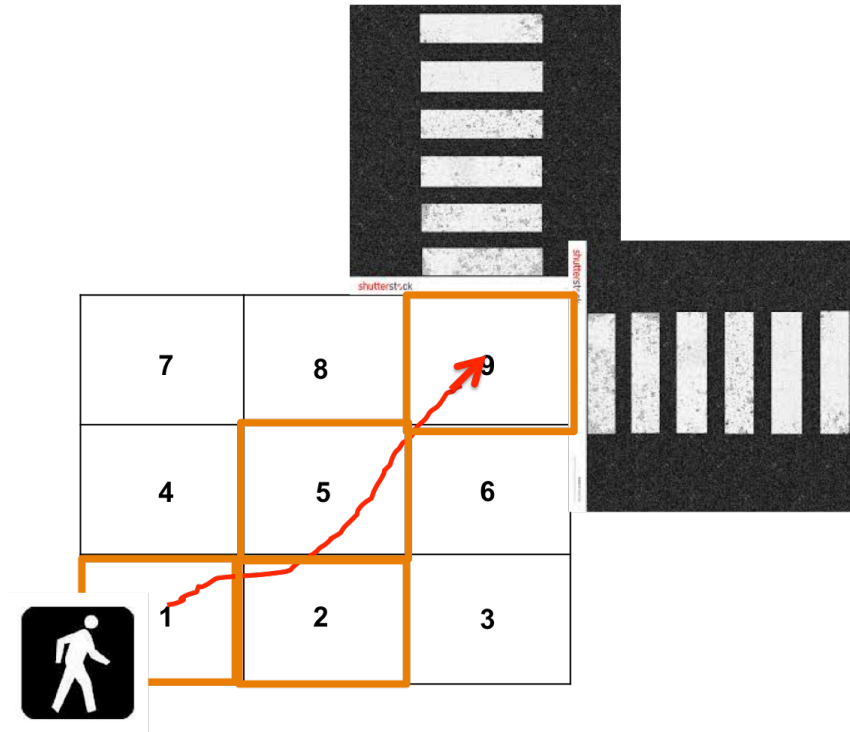


Figure 4: Detection zones for single Intersection.

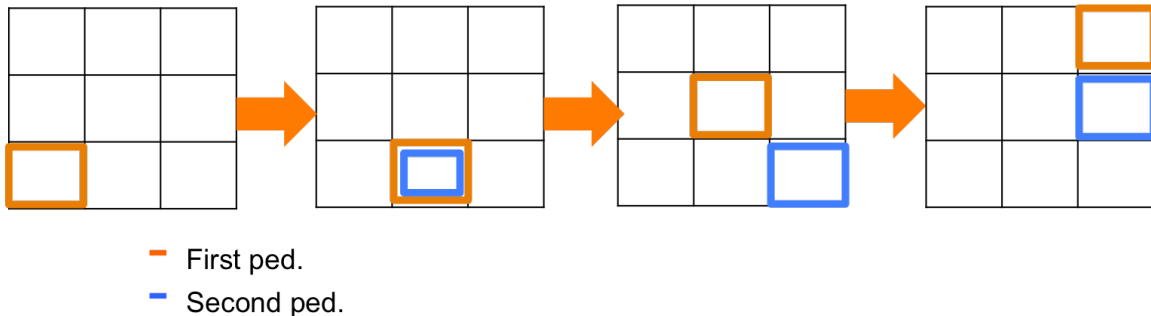


Figure 5: Example pattern for two pedestrians.

2.2.2 Learning from the Patterns

Within a given detection time period T , the movements of approaching pedestrians activate different patterns of activated zones. A supervised learning method is an efficient way to differentiate these patterns when we have ground truth information for a number of training examples. SVM in supervised learning has proven to be an effective method for sequence classification. The basic idea of applying SVM on sequence data is to map input sequences into a feature space and find the maximum-margin hyperplane that separates the two classes. For our purposes in pedestrian density estimation, we use a C-support vector classification SVM variant for multi-class classification with a radial basis kernel function. The kernel function is

$$\exp(-\gamma \|u - v\|^2),$$

where u and v are two feature vectors.

We define a SVM feature vector that includes multiple sampled snapshots of activated patterns. In our initial proposed implementation, a single feature, the number of activated zones, is used, i.e.,

$$x(N) = (x_1, x_2, \dots, x_t), \quad 1 \leq x_i \leq N,$$

where N is the number of pedestrians, t is the number of snapshots, and x_i is the number of activated zones in snapshot i . For example, the sequence depicted in Figure 5 would be described by the feature vector $(1, 1, 2, 2)$ with $N = 2$ and $t = 4$.

3. Methodology

To test the viability of the approach, we develop a simple simulation of pedestrian movement at an intersection corner and analyze the accuracy of the learned classifier. We simulate the movement of pedestrians by a predefined Markov model, whose transition probability is defined in Figure 6. When a pedestrian arrives at the edge of the intersection, s/he will not move until the pedestrian phase starts. To track pedestrians, a counter is associated with each detector zone and when a pedestrian enters/exits a zone its counter is incremented/decremented. To make the simulation more realistic, each pattern of single pedestrian arrival is offset by a random delay to create temporal separation of pedestrian traffic entering the detection zones. The movements are observed for some period T (representative of the time since the last green phase), and then pedestrian density is estimated at the end of T .

Using this simulator, samples are generated to examine performance of the approach under different assumptions about (1) the number of snapshots provided over time (or equivalently the length of the feature vector), (2) the use of composite metrics that summarize the feature vector instead of utilizing it directly, and (3) the use of density classes/categories of different size. For each experiment, 1000 samples are used; the first 500 samples are used to train the classifier and the second 500 to test its performance.

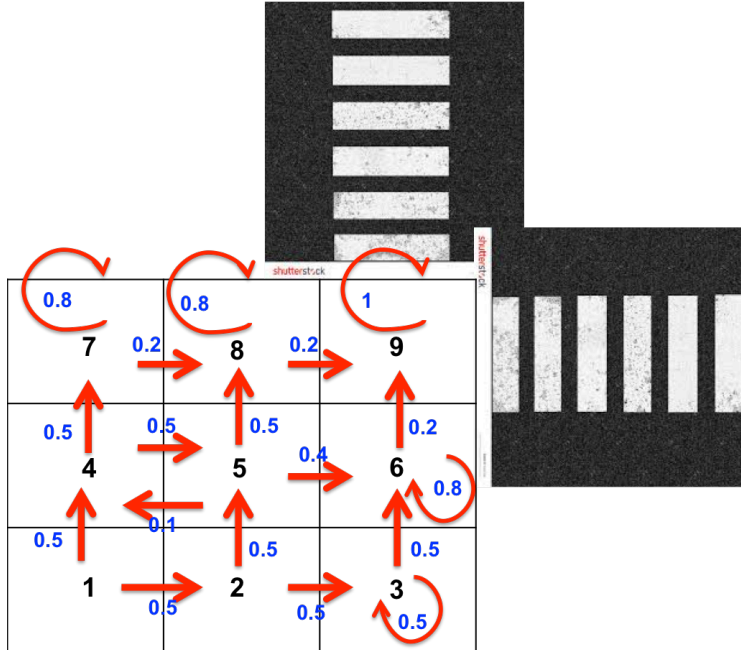


Figure 6: Transition probability of Markov chain modeling pedestrian movement.

4. Findings

We first examined the multi-class classification case with three classification intervals, defined as $[0]$, $[1-5]$, $[5-15]$ (signifying “none”, “few”, “many”). We assume a maximum of 15 pedestrians and 9 detections zones. Figure 7 shows that the misclassification rate decays as the number of snapshots increases, indicating that density estimation improves as the number of snapshots increases and pedestrian motion can be taken into account. At some point (12 in the case of Figure 7) the value of adding more snapshots becomes marginal. The reason is that the number of detection zones provides only limited resolution for tracking pedestrian motion and this constrains overall performance. The number of detection zones will also place a bound on the maximum number of pedestrians that can be effectively detected. However, as long as this upper bound is greater than the minimal value of the “many” classification interval, this constraint should not impact classification performance.

As a second experiment, we considered the utility of relying on aggregate statistics of the feature vector for estimating pedestrian density instead of directly using the feature vector itself. Specifically, we considered both the mean and maximum (max) number of activated zones across all snapshots, as a basis for estimating pedestrian density. In Figures 8 and 9 we illustrate how both the *mean* and *max* number of activated zones vary with respect to the actual number of pedestrians in the detection zones. The diminishing utility of increasing the number of snapshots after some point that was observed when using the feature vector directly is also observed for these aggregate measures. For example, the cases of 6 and 8 snapshots have similar curves. In addition, it is noted that dynamic range of *max* is larger than *mean*, which implies that *max* is inherently more informative as an indicator of density. If one were to

restrict detection to use only low dimensional features as a basis for estimating density, then max would be the preferred statistic.

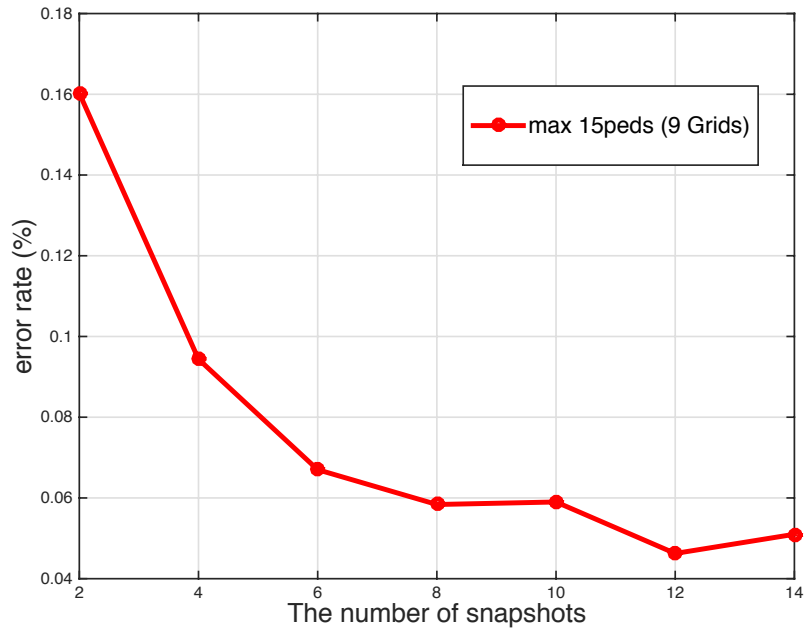


Figure 7: Three possible labels ([0], [1-5], [5-15]) under different sampled periods

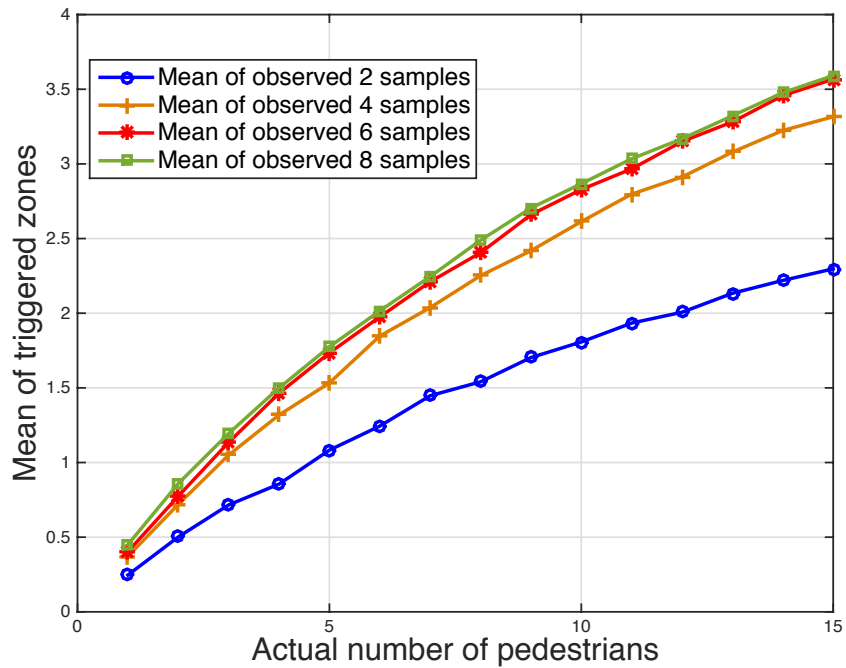


Figure 8: Accuracy of using mean number of triggered zones.

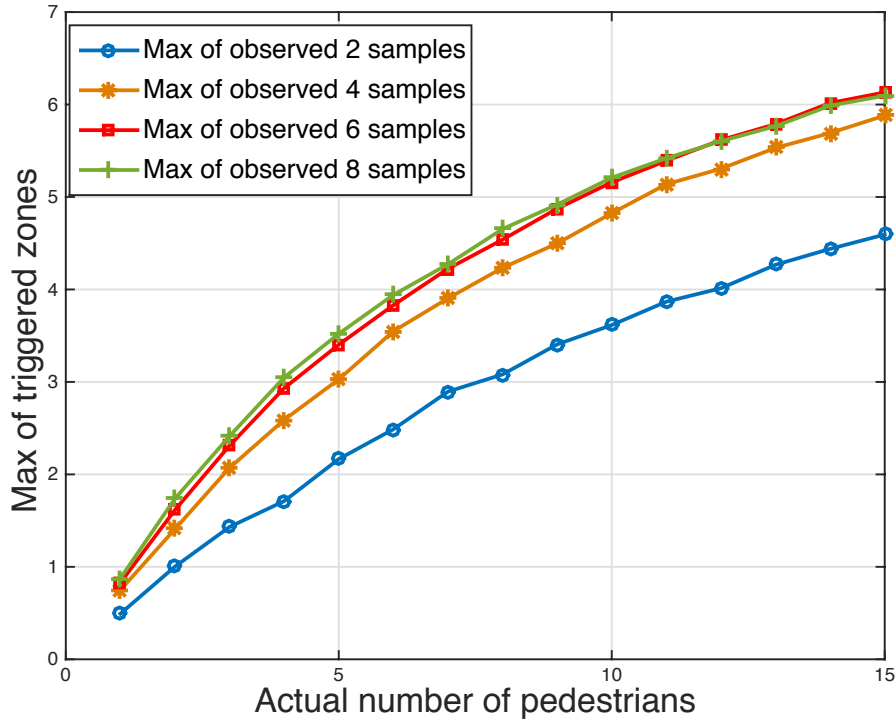


Figure 9: Accuracy of using maximum number of triggered zones.

Finally, multi-class classification performance is compared for increasing numbers of classification labels in Table 1 and Table 2. We make two basic observations. First, the error rate increases as the number of classification intervals is increased; the limited resolution provided by the fixed set of detection zones places an upper bound on the fidelity of pedestrian estimation that can be achieved. Second, we note that larger class membership intervals for the second “few pedestrians” category generally increase the error rate. This is due to the fact that it is easier to distinguish the patterns for small numbers of pedestrians with a fixed number of detection zones. When there are fewer pedestrians, there is less ambiguity in the detector zone states over time. Note however, that distinctions between different small numbers of pedestrians may in fact be the most important categories to detect for integration of pedestrian flows into adaptive signal control decisions.

Classification labels	[0], [1-3], [4-15]	[0], [1-5], [6-15]	[0], [1-7], [8-15]
Error Rate (%)	3.4	5.65	7.7

Table 1: Classification over 3 classes

Classification labels	[0], [1-3], [4-6], [6-15]	[0], [1-5], [6-10], [11-15]	[0], [1-7], [8-10], [11-15]
Error Rate (%)	10.225	16.65	18.87

Table 2: Classification over 4 classes

5. Conclusions and Recommendations

We have investigated the problem of detecting pedestrians with the same camera technology currently used for vehicle detection, as a means of minimizing the additional infrastructure cost required for real-time, multi-modal optimization of traffic flows. Given the low resolution of conventional vehicle detection technology (i.e., resolution on the order of 320×240), we focused on the development of approximate counting techniques that produce a pedestrian “density” estimate. We first examined the suitability of various approaches to detection developed within the field of computer vision. However our experiments with a range of techniques led to the conclusion that none provided a satisfactory solution.

Given this conclusion, we then considered the viability of estimating pedestrian density from a set of occupancy zones defined over the intersection image frame). We focused specifically on the concept of using a dense grid of occupancy zones that cover a corner of the intersection (i.e., the equivalent of an extremely small number of on/off “pixels”), and emphasizing classification based on the movement patterns that can be seen in multiple snapshots of the grid’s state over time. A simulation model using a 3×3 grid of zones and pre-specified Markov model of pedestrian movement was developed to provide a simple environment for testing this idea. An SVM learner was used to generate a multi-class classifier, and a series of experiments were performed to analyze the effectiveness of the approach when learning over feature vectors with different numbers of snapshots, when composite feature vector metrics substituted as the basis for classification, and when the number and size of the density categories is varied. Overall, the results indicate that it is possible to pretty reliably distinguish between such categories/classes as “no pedestrians”, “few pedestrians” and “many pedestrians”.

These results are encouraging but obviously preliminary. The simulation model used in our analysis simplifies the actual detection problem in significant ways. For example, it assumes that pedestrian traffic is moving in a single known direction through the intersection as opposed to moving through the intersection in either of two directions at each corner. Next steps should be to extend the simulation model and classification approach to overcome these simplifications and produce a technique that addresses all relevant constraints for operation in the field. At this point, an evaluation in the field can be attempted, and, if successful, pedestrian density information can be integrated into real-time traffic signal decisions.

Acknowledgements: Thanks to CitiLog, Inc. (now AXIS) for providing the project with a Citilog XCAM-ng traffic camera for use in experimentally testing various detection techniques. Thanks also to Zack Rubinstein and Hana Rudova for helpful comments on an earlier draft of this report.

References

- [1] Gary Bradski and Adrian Kaehler, *Learning OpenCV: Computer vision with the OpenCV library*. O'Reilly Media, Inc. (2008).
- [2] Emmanuel J Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011.
- [3] Antoni B Chan and Nuno Vasconcelos. Counting people with low-level features and Bayesian regression, *IEEE Transactions on Image Processing*, 21(4): 2160–2177, 2012.
- [4] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pages 886–893, 2005.
- [5] Navneet Dalal, Bill Triggs, and Cordelia Schmid. Human detection using oriented histograms of flow and appearance. In *Computer Vision—ECCV 2006*, pages 428–441. Springer, 2006.
- [6] Fernando De la Torre and Oriol Vinyals. Learning kernel expansions for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–7, 2007.
- [7] Thomas G. Dietterich, Machine learning for sequential data: a review, in *Structural, Syntactic and Statistical Pattern Recognition*, Springer, pp. 15-30, 2002.
- [8] Gunnar Farneback. Two-frame motion estimation based on polynomial expansion. In *Image Analysis*, pages 363–370. Springer, 2003.
- [9] Ya-Li Hou and Grantham KH Pang. People counting and human detection in a challenging situation. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 41(1):24–33, 2011.
- [10] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. arXiv preprint arXiv:1408.5093, 2014.
- [11] Stefan Munder and Dariu M Gavrilă. An experimental study on pedestrian classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11): 1863–1868, 2006.
- [12] Smith, S.F. G.J. Barlow, X-F Xie, and Z.B. Rubinstein, “SURTRAC: Scalable Urban Traffic Control”, Proceedings 92nd Transportation Research Board (TRB) Annual Meeting, Washington DC, January 2013
- [13] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features, *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages I–511, 2001.
- [14] Paul Viola, Michael J Jones, and Daniel Snow. Detecting pedestrians using patterns of motion and appearance. *International Journal of Computer Vision*, 63(2): 153–161, 2005.
- [15] Christian Wohler and Joachim K Anlauf. An adaptable time-delay neural-network algorithm for image sequence analysis, *IEEE Transactions on Neural Networks*, 10(6): 1531– 1536, 1999.
- [16] Bo Wu and Ramakant Nevatia. Detection of multiple, partially occluded humans in a single image by Bayesian combination of edgelet part detectors,

Proceedings Tenth IEEE International Conference on Computer Vision, volume 1, pages 90–97, 2005.

[17] Xie, X-F., S.F. Smith, T. Chen and G.J. Barlow, “Real-Time Traffic Control for Sustainable Urban Living”, *Proceedings 17th International IEEE Conference on Intelligent Transportation Systems*, Qingdao, China, October 2014.

[18] [Xhengshenz Xing, Jian Pei and Eamon Keogh, A brief survey on sequence classification, *ACM SIGKDD Explorations Newsletter*, 12(1), pp. 40-48, 2010.

[19] Tao Zhao and Ramakant Nevatia. Tracking multiple humans in crowded environment, *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages II–406, 2004.

[20] Zoran Zivkovic. Improved adaptive Gaussian mixture model for background subtraction. *Proceedings 17th International Conference on Pattern Recognition*, volume 2, pages 28–31. 2004.

[21] Zoran Zivkovic and Ferdinand van der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction, *Pattern Recognition Letters*, 27(7): 773–780, 2006.