# Pedestrian Detection and Tracking with Night Vision

Fengliang Xu[1]

Ohio State University
Columbus, OH 43210 USA

Kikuo Fujimura[2]

Honda R&D Americas
Mountain View, CA 94041, USA

**Abstract**

*This paper presents a method for pedestrian detection and tracking using a night vision video camera installed on the vehicle. To deal with the non-rigid nature of human appearance on the road, a two-step detection/tracking method is proposed. The detection phase is performed by a Support Vector Machine (SVM) with size-normalized pedestrian candidates and the tracking phase is a combination of Kalman Filter prediction and Mean Shift Tracking. The detection phase is further strengthened by information obtained by a road detection module that provides key information for pedestrian validation. Experimental comparisons have been carried out on gray-scale SVM recognition v.s. binary SVM recognition and entire body detection v.s. upper body detection.*

**Index Items:** *Pedestrian detection, tracking, infrared video, Support Vector Machine, Kalman Filter, Mean Shift Tracking*

## 1. Introduction

In traditional video surveillance with fixed cameras and stationary backgrounds, pedestrian detection can be performed by focusing on moving objects with motion-based and feature-based approaches including background elimination and analysis, periodic motion, symmetry, and silhouette shape analysis of foreground [10,11], spatial-temporal silhouette analysis of human parts with a 3D model [13], periodicity and self-similarity analysis [6], annealed particle filtering for articulated pedestrian [8], probabilistic modelling of pose and motion as well as maximum a posteriori detection [23], plane and parallax decomposition [22], wavelet template representation and Support Vector Machine (SVM) classifier [16].

However, in vehicular applications in which the camera is installed on a fast moving vehicle, new difficulties arise: the relative movement between pedestrian and background is insignificant and pedestrians vary enormously in scales. Pedestrian detection techniques can still be motion-based approaches: skeleton-based motion analysis and Kalman filter prediction [20], spatial-temporal periodic motion analysis [18]; or feature-based approaches: recognition by vertical linear features and symmetry [2], Haar wavelet representation and Support Vector Machine (SVM) classifier [15,14,16], stereo-based disparity segmentation and neural network-based recognition [26], or principal-component analysis and time-delay neural network tracking [9]. The detection of human parts is more effective than direct detection of a entire body [14].

Recently, the Support Vector Machine (SVM) [24] has been a focus of much attention. It provides a training/classification approach for object recognition: faces [12,17,21], face components [12], and pedestrians [14,15]. Pedestrian tracking techniques proposed recently include matching of middle line of candidates by Bayesian classification [3], spatial-temporal trajectory patterns and Kalman filtering prediction [19], combination of stereo parallax, colour consistency and face pattern [7], mean-shift method [5], and tracking of cubic B-spline human silhouettes [1].

In this paper, we apply SVM for pedestrian detection from night infrared videos. During the night, the

[1] Email: xu.101@osu.edu
[2] Email: kfujimura@hra.com

exposed parts of human body appear as hotspots in the infrared video. Our program starts with the detection of hotspots, estimates possible pedestrian size, clips corresponding image regions as pedestrian candidates, recognizes the candidates as pedestrians or non-pedestrians using SVM, and finally begins the tracking of recognized pedestrian using Kalman filtering prediction [25] and mean-shift [4] in tracking pedestrian's heads or bodies.

We present a number of contrasting approaches and compare their performance. (i) For the estimation of pedestrian size, we can use either hotspot directly (called hotspot candidate) or the region between the hotspot and the road (called body-ground candidate). (ii) For clipping a pedestrian from the image, we investigated the effectiveness of greyscale clips vs. binary clips. (iii) For pedestrian types, we try to recognize three types of pedestrians (along-street, across-street, and bicyclist) separately vs. all of them at one time. In (iii), for the first method, two groups of candidates (hotspot/body-ground) each has three classifiers. That is, a total of six classifiers are applied over the candidates one by one until a positive pedestrian is recognized. Secondly, a single classifier is built (to handle all types of candidates at one examination) and compared against the first method.

The training of these classifiers are undertaken on 10 night scene videos, and their performance of detection and tracking are applied on other 6 videos.

## 2. Algorithm

Our algorithm consists of two stages: human detection and human tracking. The detection stage includes candidate selection and pedestrian verification using SVM. The tracking stage uses Kalman filter to predict the approximate position of pedestrians followed by the mean-shift method to locate pedestrians precisely.

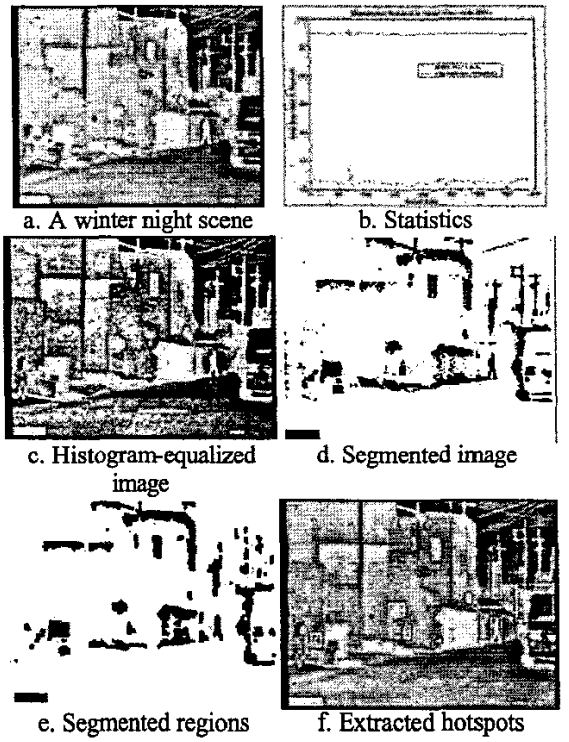
### 2.1 Candidate selection

Night-scene infrared video appears to have constant contrast, regardless of whether it is in summer or winter, and a human body often appears as a hotspot due to its high temperature compared with the environment, especially for heads and hands (and legs during winter). Our algorithm begins with the detection of hotspots with a dynamic threshold of each frame.

For night-scene infrared video, the threshold is set as a balance between the image mean intensity (Fig. 1b) and the white, and expressed as:

$$threshold = 0.2mean + 0.8white \qquad (1)$$

where *mean* denotes the mean intensity of image at current frame and *white* denotes the highest intensity (255 in 8-bit images).



a. A winter night scene b. Statistics c. Histogram-equalized image d. Segmented image e. Segmented regions f. Extracted hotspots

Figure 1. Segmentation threshold and hotspot (in b: top curve is mean density and bottom curve is segmentation threshold; in f: hotspots are in boxes)

This threshold is used for the segmentation of histogram-equalized images (Fig. 1c). The

segmentation has much noise (Fig. 1d). After noise suppression with morphological "close" and "open" operations, segmented regions (hotspots, Fig. 1e) are labelled and selected according to certain criteria: size criteria (area not too small or too large, width/height ratio is to be within 0.2 to 0.5), and positional criteria (pedestrian candidate centre should stay within the middle half of the frame). This eliminates most false pedestrian hotspots coming from buildings and vehicles. Hotspots can be either pedestrian-related hotspots: head, hands, legs (which are normally not as well-insulated as the upper-body), or the entire body; or other stuff that are not used in our system and are treated as noise.

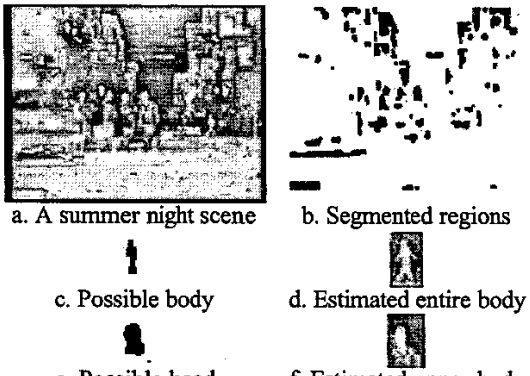


a. Normalized candidates b. Verified candidates

Figure 2. Normalization and verification of hotspot candidates from Figure 1. Image clips with even numbers are hotspots, while Image clips with odd numbers are body-ground candidates

We try to find hotspots that contains heads. There are two types of hotspot-based candidate-selection methods: hotspot candidate (pedestrian candidate with size estimated by the size of hotspot itself, which is normally the upper-body region if containing head) and body-ground candidate (pedestrian candidate with size estimated by the distance between the ground and the top of hotspot, which is exclusively the entire-body region).

If the entire body of pedestrian appears as an hotspot, then the estimates of pedestrian size from both methods should be the same and there will be no difference between hotspot candidate and body-ground candidate. However, people normally wear well-insulated clothing that reduces the infrared signature of entire body, thus the entire body seldom appears as a hotspot, except when the pedestrian is far and small or clothing is not well-insulated. As a result, these two types of candidates are different in most cases and the comparison between hotspot candidates and body-ground candidates are actually the comparison between the effectiveness of pedestrian detection using upper-body or entire-body.



a. A summer night scene b. Segmented regions

c. Possible body d. Estimated entire body

e. Possible head f. Estimated upper-body

Figure 3. Estimation of pedestrian size for hotspot candidates

a. Head candidates b. Range of candidates

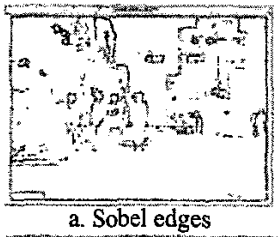
c. Normalized hotspot candidates d. Detected hopspot by SVM

Figure 4. Normalization and verification for hotspot candidate: upperbody case

The entire body's shape changes a lot due to the movement of legs and complexity of entire clothing, while upper-body has less movement and is more stable and simpler. Upper-body is relatively easier for detection and tracking, its drawback is relatively

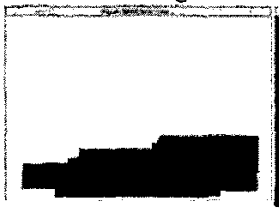
higher false alarms due to its simplicity: round-shape head is apt to be confused with other light sources such as lamps. Compared with it, entire-body is more precise in detection.

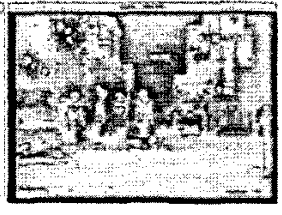
For a hotspot candidate, if the hotspot corresponds to the entire body (Figure 3.c), a region a little larger than the hotspot is extracted as a pedestrian candidate (Figure 3.d); if the hotspot is related to the head (Figure 3.e), the size of pedestrian's upperbody is estimated by doubling the head-size and the upperbody candidate is extracted with its centre overlapping with the head-centre (Figure 3.f). One extraction example is in Figure 4. Since we do not know in advance which part of pedestrian the hotspot corresponds to, we extract both regions as hotspot candidates.



a. Sobel edges

b. Morphological kernel





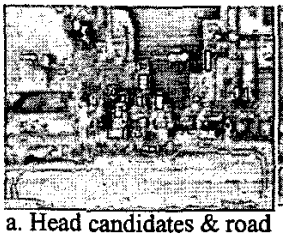
c. Ground detected d. Road boundary

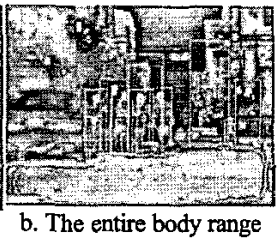
Figure 5. Ground extraction by morphological operations over edge map

For body-ground candidate, we assume pedestrians always walk along the roadside, thus their size could be estimated by the distance between the top of hotspot (regardless of whether it is a head or something else) and the ground, and this region of image could be extracted as a pedestrian candidate. Even when the pedestrian is actually walking in the road centre, the size estimate will still be correct because he still appears to be walking on the side of "detected" road. This requires a robust ground-detection method.

In infrared video, a road normally has a constant temperature and thus has no rapid density change. It can be extracted by morphological operations over the Sobel edge image as in Figure 5a. Then, erosion and dilation operations with a rectangular-shaped kernel (Fig.5b) are applied to separate the ground region from other non-ground regions. This kernel has a height of 20, and a variable width ranging from 100 to 200, which may increase if result shows that the ground is connected to the upper part of the image. The lowest large homogenous region is extracted as ground region and its contour is extracted as road boundary (Figure 5).

Once the road boundary is obtained, the distance between possible pedestrian head-top and the ground can be measured and thus the size of the pedestrian can be estimated (e.g., width can be selected as 1/3 of body height, in general) and the corresponding area in the image will be extracted as a pedestrian candidate. As seen in (Fig. 6b), pedestrian candidates may overlap.





a. Head candidates & road b. The entire body range

c. Normalized pedestrian candidates d. Detected pedestrians using SVM

Figure 6. Body-ground candidates and classification

Extracted ground also helps to further eliminate the non-pedestrian candidates. Hotspots will not be

considered if their sizes are not proportional to their height (too small yet too high, or too large yet too low).

Both types of pedestrian candidates have various sizes, and need to be normalized to a standard size (40x20 patch in our case, it has a similar height/width ratio for normal pedestrians) for comparison and classification. The normalized candidates are shown as in Fig. 2a, Fig. 4c. and Fig. 6c. They will be classified as a pedestrian or non-pedestrian later by using SVM. Some pedestrians recognized by SVM are shown in Fig. 2b, Fig. 4d. and Fig. 6d.

In the night scene, pedestrians walking along the street normally appear far, small, and blurred at the beginning, then get larger and clearer. It is usually hard to extract pedestrians when they are too blurred. Thus, we define the clarity of pedestrian to denote the greyscale difference between the blurred candidate and the most-clear candidate as:

$$clarity_1 = \exp(-\frac{1}{n}\sum_{i=1:n}|G_{ci} - G_{ri}|) \qquad (2)$$

where $G_{ci}$ is the grey-value of current pixel in the current blurred image and $G_{ri}$ is the grey-value of corresponding pixel in the most-clear image used as reference. $clarity_1$ ranges from zero to 1, with 1 being the clearest and a value near 0 means blurring.

If the clearest reference image is not available, then clarity can be defined in the following way: given an optimal threshold to classify the pixels in the candidate region as white pixels and dark pixels, with greyscale distribution m1 and m2 as mean value, and v1 and v2 as standard deviation, respectively, then the clarity is:

$$clarity_2 = \tanh(\frac{m1-m2}{m2} + \frac{1}{3}\frac{v1-v2}{v2}) \qquad (3)$$

Thus, high clarity means high contrast and more details, however, the contrast should be put more weight than the variance. Its value also ranges from zero to 1 (0 means blurring and 1 means clearest). One example of clarity of a bicyclist extracted from the night scene video is shown in figure 7.

0.83 0.87 0.94 1
0.61 0.71 0.82 0.84

Figure 7. Clarity of pedestrian ($clarity_1$ on the first row and $clarity_2$ on the second row)

## 2.2 Human Classification with SVM

The Support Vector Machine is a classification method which calculates the boundary (support vectors) between two sets of high dimensional vectors (training stage) and uses these support vectors to classify vectors from a similar source (classification stage). The number of support vectors generated and the detection rate can be used to evaluate the efficiency of different training data sets, e.g., gray-scale v.s. binary, or hotspot vs. body-ground candidates, for pedestrian detection.

The training of SVM is undertaken by the manual classification of extracted pedestrian candidates. Pedestrian candidates of some frames in a video are not used for training because of large redundancy. Instead, we select training frames as the first 5 frames of each 25 frames, thus ensuring both a locally high resample ratio and a globally low redundancy. Then, sets of support vectors are generated from the training set and are used for the recognition of pedestrian candidates from other videos or from other frames of the same video.

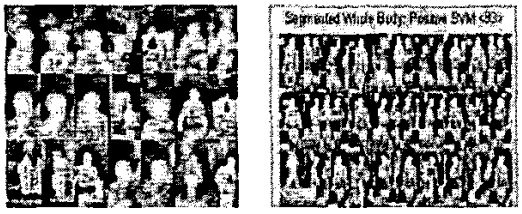
Comparisons are made to examine the effectiveness of the SVM for classification of greyscale pedestrian candidates vs. binary pedestrian candidates. The greyscale vector may suffer from the difference of clothing between human candidates and cannot catch the most important information: the shape of candidates. The binary vector may be more general. Binary vectors are obtained by thresholding the greyscale vectors. Fig. 8 shows examples of positive samples from both cases.

a. Samples of greyscale positive data  b. Samples of binary positive data

Figure 8. Comparison on greyscale vs. binary data

Experiments on self-trained SVM (training frames and testing frames are non-overlapping subsets of the same video) show that the pedestrian recognition using greyscale candidates is quite successful: it can always recognize the right candidates and works fine, as long as the difference is minor between the training frames and the testing frames. However, the binary candidates are too shape sensitive and hard to achieve a high detection ratio. Thus the greyscale method would be superior as long as there is no ideal segmentation method.

Another comparison is made between the performance of hotspot (mainly upper-body) and body-ground candidates (entire-body) as in Fig. 9.



a. Positive samples for hotspot candidate  b. Positive samples for body-ground candidate

Figure 9. Comparison between hotspot and body-ground candidates

The two types of candidates can achieve nearly the same detection ratio. However, the number of support vectors from the hotspot candidates is far less than that from body-ground candidates, indicating that the hotspot is more efficient, robust, and faster.

We further classify each training set into three types of pedestrians as in Fig. 10: along-street pedestrian, across-street pedestrian, and bicyclist (although not real pedestrians, they are equally vulnerable as pedestrians). The pedestrians in the training set are manually classified into three classes, then trained to generate SVM classifiers. Then, an attempt is made to detect each type of pedestrians independently. We have a total of six classifiers (three each for hotspot and body-ground candidates).

a. Along-street  b. Across-street  c. Bicyclist

Figure 10. Three types of pedestrians

There are two approaches for SVM classifiers: the first is a single classifier for all types of pedestrians, which is simple, yet has very large number of support vectors, long training time, and slow classification speed due to the large variety of positive candidates. Another approach is to build multiple classifiers, each for a specific type of pedestrians, resulting in compact positive candidates and reduction in both the time on training and classification as well as the size of necessary support vectors. (However, too many classifiers will also slow down the system because one candidate may have to be applied with every classifier until it is verified as non-pedestrian). Section 4 contains our experimental results based on 1-classifier and 6-classifier.

## 3. Pedestrian Tracking

Once pedestrians are detected, the tracking stage follows. Tracking of a human head is relatively easy because it is a hotspot and its shape does not change much between frames. The movement of head is predicted by a Kalman filter [25] using the following equations:

Time Update Equations:

Priori positions: $S_k^- = \Phi S_{k-1}$

Priori measurements: $P_k^- = \Phi P_{k-1} \Phi^T + Q$

Measurement Update Equations:
Kalman gain: $K_k = P_k^- H_k (H_k P_k^- H_k^T + R_k)^{-1}$
Posteriori positions: $S_k = S_k^- + K_k(Z_k - H_k S_k^-)$
Posteriori measurements: $P_k = (I - K_k H_k) P_k^-$

where $S_{k-1}$, $S_k$, and $S_k^-$ are estimated positions at time $k-1$, estimated position at time $k$, and estimated position at time $k$ before updating with the error between $S_k$ and $Z_k$, respectively. $P_{k-1}$, $P_k$, and $P_k^-$ are error covariance for current parameters at time $k-1$, current parameters at time $k$, and estimated parameters at time $k$; $\Phi$ is the transform matrix from $S_{k-1}$ to $S_k^-$; $Q$ represents the model error; $Z_k$ is the measurement at time $k$; $H_k$ is the noiseless connection between the measurement $Z_k$ and position $S_k$ at time $k$, $K_k$ is the Kalman gain or blending factor that minimizes the $P_k$. Time related parameters are updated each frame.

Based on the information from previous frames, the head position in a new frame $S_k$ (a posteriori position) can be estimated. Since the movement of pedestrian is not linear, the estimation may not be precise. We used the mean-shift method [4] to find the precise position around the posteriori position:

$$m(x) = \frac{\sum_{s \in S} K(s-x) w(s) s}{\sum_{s \in S} K(s-x) w(s)} \quad (4)$$

Where $x$ is the current position, $w(s)$ is a weight function (the ratio of histogram value between original greyscale level and the current greyscale level at location $s$), and $m(x)$ is the new position, $K$ is the kernel:

$$K(x) = \frac{1}{4\pi} e^{-\frac{\|x\|}{2}} \quad (5)$$

with shape:

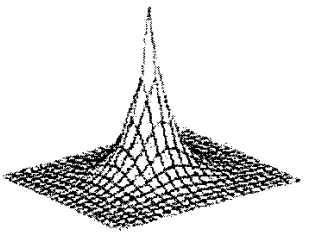


Figure 11. Kernel used in mean shift tracking

The tracking result is shown in Figure 12, where circles represent detected heads, and squares represent the heads being tracked. The method can track multiple pedestrian bodies simultaneously in real-time.

a b

Figure 12: Detection stage (a) and tracking stage (b) Circles and squares represent the pedestrian heads from detection and tracking stage, respectively

For the joint use of detection and tracking, since detection (0.4s per frame, on a P-III 500 computer with 256M memory) is more time-consuming than tracking only (0.2s per frame). Tracking is much more robust than detection because in the tracking stage, the target is hardly lost; however, in detection stage, a target may not be detected at all times. As a result, tracking is preferred. The drawback of tracking is that it only tracks the already-found pedestrians and does not look for new pedestrians. So we interleave the detection and tracking stages by applying detection every 5 frames or anytime the tracking of a pedestrian is lost. For the rest of time the system remains in tracking. Thus, our system makes good use of the balance between the robustness and the system's ability to discover new pedestrians.

## 4. Experiment Results

We have 16 night scene videos, 8 for summer and 8 for winter, including a total of 39 pedestrians (and bicyclists). We selected 10 typical videos as training video (Figure 13) and the other 6 videos containing a total of 13 pedestrians (Figure 14) for verification.

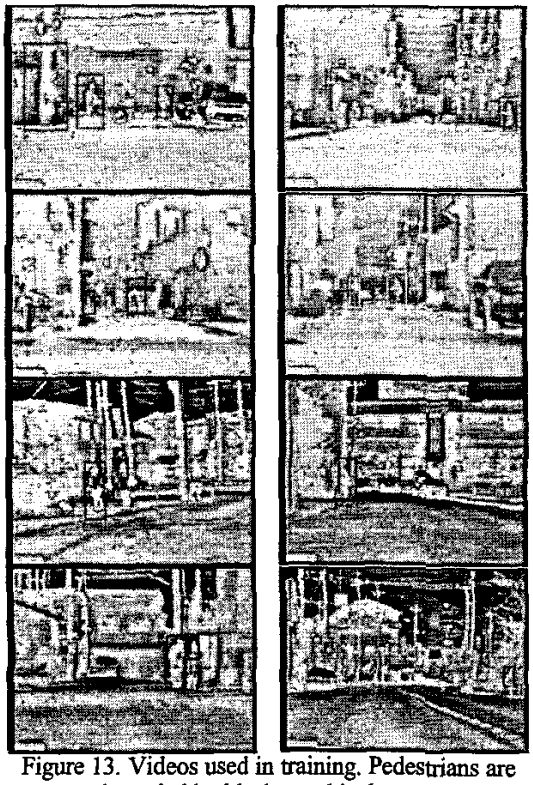


Figure 13. Videos used in training. Pedestrians are bounded by black or white boxes.

In Fig. 13, pedestrians are re-detected automatically using the support vectors, which are trained using manually classified pedestrian samples that are selected from some frames (5 out of each 25) of the training videos. Training videos consist of 8 videos as in Fig. 13, one video as in Fig. 1, and one video as in Fig. 3.

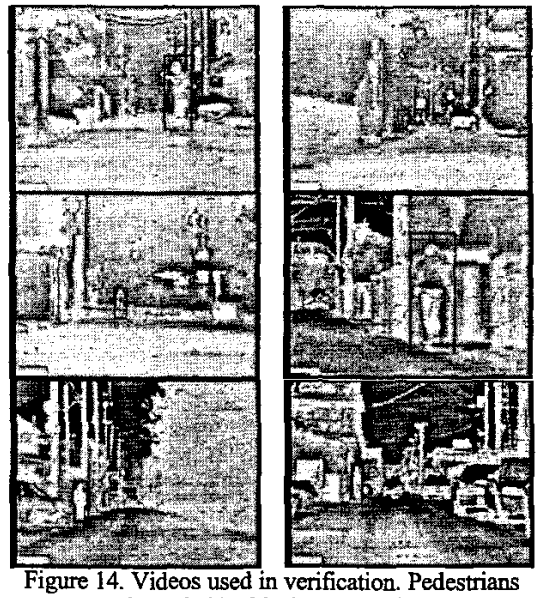


Figure 14. Videos used in verification. Pedestrians are bounded by black or white boxes

Table 1. The number of SVs in classifiers

| Type-specific classifiers | | Positive | Negative |
|---|---|---|---|
| body-centre | along-street | 339 | 829 |
| | across-street | 361 | 765 |
| | bicyclist | 34 | 89 |
| body-ground | along-street | 304 | 896 |
| | across-street | 446 | 1156 |
| | bicyclist | 27 | 68 |
| | | | |
| single classifier | | 1368 | 2437 |

The overall detection/tracking result of pedestrians for 6 verification videos is as in Table 2 (in number of pedestrians, F/A means False Alarm):

Table 2. Detection/Lost/False-alarm number

| | detected | missed | F/A |
|---|---|---|---|
| single classifier | 12 | 1 | 19 |
| multiple classifier | 12 | 1 | 9 |

"detected" in Table 2 means the pedestrian is detected during the video, yet not ensures that it could be detected in every frame.

For the performance regarding detection and tracking with respect to every frame in the video, we compare the detect ratio (only use detection without tracking) with the detect/track ratio (combination of detection and tracking) a typical result is as follows. For a pedestrian be covered by 200 continuous frames, for multiple-classifiers, it is detected in 51 frames (25% detection ratio), detected/tracked in 52 frames (26% detect/track ratio), with 6 false alarms; for single-classifier, the pedestrian is detected in 70 frames (35% detection ratio), detected/tracked in 189 frames (94% detect/track ratio), with 5 false alarms.

The results show that the single-classifier works better than the multiple-classifier. Although the detection ratio is not very high, the detected frames are evenly distributed in time, which means that the pedestrian could always be detected in a short period of time, then enter the tracking stage. In pedestrian detection, we argue that how soon pedestrian can be detected is important, while the detection rate (e.g., the number of frames they can be detected) may not matter much because once a pedestrian is detected, it will be enough for the driver.

For the issue of false alarms, although for every video there are several false alarms, relative to the number of candidates in one frame (30 to 50 per frame, depending on the complexity of the scene), it is still small, because over 90% false candidates are successfully detected by SVM.

The false alarm ratio is related to the training sample selection method we use. All the positive pedestrian samples are used for training, yet the amount of negative samples is too huge to use all of them. We have compared various size of negative examples. When a large amount of negative samples (30 times more than that of positive samples) is used, we had several false alarms. When the size is reduced (two times more than that of positive samples by random selection), the missing ratio remains the same with a slight increase in false alarms. So in practical training, we use a small size of negative samples.

If pedestrian type is to be reported, multiple classifier may be applied over a pedestrian after it is detected by the single classifier.

For pedestrians walking or riding bicycle along the road, it is usually hard to detect them when they are far and have low clarity. However, after they get closer and the clarity is higher ($clarity_2$>0.7), they will have more chance to be detected.

## 5. Summary and Conclusions

We have presented a new method for pedestrian detection and tracking based on night vision. The difficulty of night time vision has been that the appearance of pedestrians is not clear compared to that of day-time images and that pedestrians might appear differently due to various obstructions, overlaps, and distance. Our real-time technique makes use of a combination of appearance-based detection and tracking methods to benefit from the strengths of different techniques and overcome their respective limitations. Experimental results have shown the feasibility of this approach. Our further study includes performance improvement by an optimum combination of detection and tracking, the representation of pedestrian candidates with contour instead of region, and pedestrian detection based on legs movement.

## References


[1] A. Baumberg and D. Hogg. Learning flexible models from image sequences. In *European Conference on Computer Vision*, vol. 1, pp299-308, 1994

[2] A. Broggi et. al. Shape-based Pedestrian Detection, *Proceedings of the IEEE Intelligent Vehicles Symposium*, pp215-220, Dearbon (MI), USA, Oct, 2000

[3] Q. Cai and J.k. Aggarwal. Tracking Human Motion in Structured Environments Using a Distributed-Camera System. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(11): 1241-1247, Nov 1999

[4] Y. Cheng. Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern*

*Analysis and Machine Intelligence*, 17(8):790-799, 1995

[5] D. Comaniciu et. al. Real-time tracking of non-rigid objects using mean shift. In Proc. *IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 2, pp142-149, Hilton Head Island, SC, June 13-15, 2000

[6] R. Cutler and L. Davis. Robust real-time periodic motion detection, analysis, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):781-797, Aug. 2000

[7] T. Darrell et. al. Integrated person tracking using stereo, colour, and pattern detection. *IEEE Conference on Computer Vision and Pattern Recognition*, pp601-608, 1998

[8] J. Deutscher et. al. Articulated body motion capture by annealed particle filtering. In Proc. *IEEE Conf. on Computer Vision and Pattern Recognition*, volume II, pp126-133, Hilton Head Island, South Carolina, 2000

[9] U. Franke et. al. Autonomous driving goes downtown. *IEEE Intelligent Systems*, pp32-40, Nov./Dec. 1998

[10] I. Haritaoglu et. al. Backpack: Detection of people carrying objects using silhouettes. In *International Conference on Computer Vision*, Corfu, Greece, Sept. 1999

[11] I. Haritaoglu et. al. W4: Real-time surveillance of people and their activities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):809-831, Aug. 2000

[12] B. Heisele et. al. Face recognition with support vector machines: global versus component-based approach. *IEEE International Conference on Computer Vision* 2001, Vancouver, Canada, 2001

[13] I. Kakadiaris and D. Metaxas. Model-based estimation of 3D human motion, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12): 1453-1460, Dec. 2000

[14] A. Mohan and T. Poggio. Example-based object detection in images by components, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(4): 349-361, April 2001

[15] C. Papageorgiou et. al. A trainable pedestrian detection system. In *Proc. of Intelligent Vehicles*, pp241-246, Stuttgart, Germany, October 1998

[16] M. Oren et. al. Pedestrian detection using wavelet templates. *Proc. on Computer Vision and Pattern Recognition*, pp193-199. IEEE 1997

[17] E. Osuna et. al. Training support vector machines: an application to face detection. *Computer Vision and Pattern Recognition*, pp130–136, 1997.

[18] R. Polana and R. Nelson. Low level recognition of human motion. In *Workshop on Motion of Non-Rigid and Articulated Objects*, Austin, Tx, USA, Oct. 1994

[19] Y. Ricquebourg and P. Bouthemy. Real-time tracking of moving persons by exploiting spatio-temporal image slices. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8), pp797-808, Aug. 2000

[20] K. Rohr. Incremental recognition of pedestrians from image sequences. In *Proc. Computer Vision and Pattern Recognition*, pp8-13, 1993

[21] S. Romdhani et. al. Computationally efficient face detection. *IEEE International Conference on Computer Vision 2001*, Vancouver, Canada, 2001

[22] H. Sawhney et. al. Independent motion detection in 3D scenes, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10): 1191-1199, Oct. 2000

[23] Y. Song et. al. Towards detection of human motion. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pp810-817, 2000

[24] V. Vapnik. *The nature of statistical learning theory*. Springer, 2000

[25] G. Welch and G. Bishop. *An Introduction to the Kalman Filter*. University of North Carolina at Chapel Hill, Department of Computer Science, Chapel Hill, NC, USA. TR95-041

[26] L. Zhao and C. Thorpe. Stereo and neural network-based pedestrian detection. Proc. *Int'l Conf. on Intelligent Transportation Systems*, Tokyo, Japan, Oct. 1999

[27] P. Philonom, *Real-time generic object detection and tracking for "Smart" vehicles*. PhD Thesis, Univ. of Maryland, College Park, Dept. Computer Science, 2000.