

Article

# Efficient Pedestrian Detection at Nighttime Using a Thermal Camera

Jeonghyun Baek, Sungjun Hong, Jisu Kim and Euntai Kim \*

The School of Electrical and Electronic Engineering, Yonsei University, Seoul 120-749, Korea; jhyun25@yonsei.ac.kr (J.B.); imjune@yonsei.ac.kr (S.H.); jisukim2000@yonsei.ac.kr (J.K.)

\* Correspondence: etkim@yonsei.ac.kr; Tel.: +82-2-2123-7729

Received: 19 May 2017; Accepted: 7 August 2017; Published: 10 August 2017

**Abstract:** Most of the commercial nighttime pedestrian detection (PD) methods reported previously utilized the histogram of oriented gradient (HOG) or the local binary pattern (LBP) as the feature and the support vector machine (SVM) as the classifier using thermal camera images. In this paper, we propose a new feature called the thermal-position-intensity-histogram of oriented gradient (TPIHOG or  $T\pi$ HOG) and developed a new combination of the  $T\pi$ HOG and the additive kernel SVM (AKSVM) for efficient nighttime pedestrian detection. The proposed  $T\pi$ HOG includes detailed information on gradient location; therefore, it has more distinctive power than the HOG. The AKSVM performs better than the linear SVM in terms of detection performance, while it is much faster than other kernel SVMs. The combined  $T\pi$ HOG-AKSVM showed effective nighttime PD performance with fast computational time. The proposed method was experimentally tested with the KAIST pedestrian dataset and showed better performance compared with other conventional methods.

**Keywords:** pedestrian detection; far-infrared sensor; thermal-position-intensity-histogram of oriented gradient

## 1. Introduction

For the commercialization of the advanced driver assistance system (ADAS), the most important factors are reliability and robustness, and pedestrian detection (PD) is certainly one of the ADAS functions that require high reliability and robustness. For a robust and reliable PD, reasonable performance even in the nighttime is important because more than half of pedestrian-related accidents occur in the nighttime, even though the volume of traffic is much less than in the daytime [1,2].

For effective nighttime PD, most studies used a thermal camera sensor because it visualizes objects using the infrared (IR) heat signature and it does not depend on lighting conditions. Among several types of thermal cameras, the far infrared (FIR) sensor is commonly used for PD in the nighttime because thermal radiation from pedestrian peaks in the FIR spectrum [3]. Compared with visible images, FIR images are robust against illumination variation but are significantly affected by weather because FIR sensors capture temperature changes in the output images. For example, pedestrians appear brighter than the background in cold days while they appear darker in hot days [1]. Furthermore, FIR images only contain a single channel of intensity information; thus, information on these images is not as detailed as that of visible images.

Similar to PD using visible images, PD using FIR images also consists of two steps: feature extraction and classification. In the feature extraction step, the features developed for daytime PD can also be used for nighttime PD. For example, local binary pattern (LBP) [4] and its variations, such as the HOG-LBP [1,5], center-symmetric LBP (CSLBP) [6], and oriented CSLBP (OCSLBP) [7] were also proposed as daytime PD features. However, the LBP-based features have only orientation information of pixel intensity; therefore, they are sensitive to lighting conditions. On the other hand,

there are some methods that use the shape of pedestrians as features. Dai et al. [8] utilized the joint shape and appearance cue to find the exact locations of pedestrians. Wang et al. [9] extracted the features using a shape descriptor and Zhao et al. [3] proposed the shape distribution histogram (SDH). These shape-based features simply used only pixel intensity information and employed background subtraction methods for fixed camera images. Therefore, shape-based features are not suitable for vehicle environment where complex background is not fixed.

As a robust feature for pedestrian detection, the histograms of oriented gradient (HOG) [10] is one of the most popular PD features and its variations have been proposed [11]. Co-occurrence HOG (CoHOG) is one of the extensions of the HOG and it utilizes pair of orientations for computing histogram feature [12]. N. Andavarapu et al. [13] proposed weighted CoHOG (W-CoHOG) that considers gradient magnitude factor to extracting CoHOG. Spatio-temporal HOG (SPHOG), which contains motion information, is proposed for image sequences with fixed camera [14]. Scattered difference of directional gradient (SDDG) that extracts local gradient information along the certain direction is also proposed for IR images [15]. Kim et al. proposed position-intensity HOG (PIHOG or  $\pi$ HOG) that includes not only HOG but also the detail position and intensity information for vehicle detection [16]. These HOG based features utilize only the gradient information based on color images or do not consider the thermal intensity information which is important cue for pedestrian detection in nighttime.

To address these problems of conventional features, we propose a thermal position intensity HOG (TPIHOG or  $T\pi$ HOG). The  $T\pi$ HOG is the extended version of  $\pi$ HOG and it is applied for pedestrian detection in nighttime. Unlike the  $\pi$ HOG, the proposed  $T\pi$ HOG has thermal intensity information and can be computed more simply than  $\pi$ HOG.

With respect to the classification, the linear support vector machine (linear SVM) is widely used as classifier in many studies, such as in [17–19] because it is fast and has reasonably good performance. The kernel SVM has better classification performance than the linear SVM but requires a longer computation time owing to kernel expansion [20–22]. However, the additive kernel SVM (AKSVM) has better performance than the linear SVM and also has a classification speed comparable with the linear SVM [20,23,24]. Recently, deep learning has also been applied to object detection system.

Kim et al. utilize convolutional neural network (CNN) for nighttime PD using visible images [25]. Liu et al. and Wagner et al applied fusion architectures to CNN which fuse the visible channel feature and thermal channel features for multispectral PD [26,27]. Cai et al. generates the candidates using saliency map and used deep belief network (DBN) as a classifier for vehicle detection in nighttime [28]. John et al. used Fuzzy C-means clustering for generating candidates and CNN for verification for PD in thermal images [29]. However, the deep learning based method requires dataset with large clean annotations and training procedure, which takes too much time to converge [30]. In addition, GPU is necessary for the training of deep learning, but it is not suitable for implement autonomous system because the system needs to be embedded system [31].

In this study, we propose a combination of the  $T\pi$ HOG and the AKSVM ( $T\pi$ HOG-AKSVM) to achieve improved performance for nighttime PD in terms of PD performance compared with conventional methods.

The remainder of this paper is organized as follows. In Section 2, some background about the  $\pi$ HOG is presented. Details of the proposed  $T\pi$ HOG-AKSVM are presented in Section 3. Section 4 presents experimental results and discussions, and conclusions are drawn in Section 5.

## 2. Preliminary Fundamentals

### *Position Intensity Histogram of Oriented Gradient ( $\pi$ HOG)*

The HOG is defined as the histogram of the magnitude sum for gradient orientations in a cell and it is widely used as an effective feature for PD or vehicle detection (VD). The HOG feature, however, has the limitation that information on gradient position in the cell is lost and the pixel intensity

information is not used. Recently, the  $\pi$ HOG has been proposed to address this problem and shows better detection performance than HOG for vehicle detection [16]. The  $\pi$ HOG contains not only the HOG but also additional information about gradient Position and pixel Intensity. The  $\pi$ HOG consists of three parts: the Position (**P**) part, the Intensity (**I**) part, and the conventional HOG. The **P** part of the  $\pi$ HOG is extracted by computing the average position of each orientation bin. That is, if  $\theta(x, y, c)$  denotes the orientation of the gradient at position  $(x, y)$  of the  $c$ -th cell and the orientation bin of the gradient  $B(x, y, c)$  is defined by

$$B(x, y, c) = \left\lceil \frac{T\theta(x, y, c)}{2\pi} \right\rceil, \quad (0 \leq \theta(x, y, c) < 2\pi) \quad (1)$$

where  $T$  is the number of bins and  $B(x, y, c) \in \{1, \dots, T\}$ , then the averages of  $x$  and  $y$  positions of the  $d$ -th bin ( $d \in \{1, \dots, T\}$ ) in the  $c$ -th cell are defined by

$$\begin{aligned} M_{x,d}^c &= \frac{\sum_{x=1}^{c_s} \sum_{y=1}^{c_s} x \mathbb{I}[B(x,y,c)=d]}{\sum_{x=1}^{c_s} \sum_{y=1}^{c_s} \mathbb{I}[B(x,y,c)=d]} \\ M_{y,d}^c &= \frac{\sum_{x=1}^{c_s} \sum_{y=1}^{c_s} y \mathbb{I}[B(x,y,c)=d]}{\sum_{x=1}^{c_s} \sum_{y=1}^{c_s} \mathbb{I}[B(x,y,c)=d]} \end{aligned} \quad (2)$$

where  $c_s$  denotes the cell size and  $\mathbb{I}(\cdot)$  is the binary function which returns 1 if the input argument is true and 0 if the argument is false. Then the **P** part of the  $c$ -th cell in  $\pi$ HOG is  $\mathbf{P}_c = [\mathbf{M}_x^c, \mathbf{M}_y^c]$ , where  $\mathbf{M}_x^c = [M_{x,1}^c, \dots, M_{x,T}^c]$  and  $\mathbf{M}_y^c = [M_{y,1}^c, \dots, M_{y,T}^c]$ .

The **I** part of the  $\pi$ HOG can be defined in terms of the pixel intensity of vehicle images. There are a variety of shapes and sizes of vehicles, but all types of vehicles have low intensity values in some common areas, such as tires and bottom of the vehicles. Using this knowledge, the intensity invariant region (IIR) was proposed in [16]. The IIR is defined as the region of pixels in which the corresponding standard deviation is relatively low. Then, the **I** part is defined as the summation of standard normal deviation values in the IIR. A detailed procedure for extracting the **I** part of the  $\pi$ HOG is explained as follows. When a set of positive vehicle images  $\mathcal{V}^+ = \{\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_{N_v}\}$  is given, where  $N_v$  is the number of the training images in  $\mathcal{V}^+$ ;  $\mathbf{s} = [s_1, s_2, \dots, s_N]^T \in \mathbb{R}^N$  is the vehicle image and all the images in  $\mathcal{V}^+$  are resized to images of same size and aligned;  $s_j$  is the intensity value of  $j$ th pixel value of  $\mathbf{s}$  and  $N$  is the image size, the mean and standard deviation of the vehicle images are defined as

$$\mathbf{M} = \frac{1}{N_v} \sum_{i=1}^{N_v} \mathbf{s}_i = [m_1, m_2, \dots, m_N]^T \quad (3)$$

$$\sigma = \sqrt{\frac{1}{N_v} \sum_{i=1}^{N_v} (\mathbf{s}_i - \mathbf{M}) \circ (\mathbf{s}_i - \mathbf{M})} = [\sigma_1, \sigma_2, \dots, \sigma_N]^T \quad (4)$$

where  $\circ$  denotes component-wise multiplication. In Equation (4), low  $\sigma$  means that the corresponding region has similar intensity values over all types of vehicles including sedans, trucks or sport utility vehicles (SUVs). Therefore, the region with low standard deviation  $\sigma$  can be used as a distinctive cue for classifying vehicles. This region was defined as IIR and a new feature was extracted from the IIR [19]. To determine the IIR, the values of  $\sigma$  are divided into  $M$  intervals and the binary mask  $\mathcal{U}_k$  ( $k = 1, 2, \dots, M$ ) is constructed as

$$\mathcal{U}_k = \{j | \xi_k \leq \sigma_j \leq \xi_{k+1}, j = 1, 2, \dots, N\} \quad (5)$$

where  $\xi_k$  is

$$\zeta_k = (k - 1) \times \left\lceil \frac{N}{M} \right\rceil \text{th smallest value of.} \tag{6}$$

Finally, the I part of the  $\pi$ HOG is the feature for the IIR region masked by  $\mathcal{U}_k$  of standard normal deviate image  $\mathbf{z}$ . That is, if the test image  $\mathbf{s} \in \mathfrak{R}^N$  is given, then  $\mathbf{z}$  is computed by

$$\mathbf{z} = \left( \frac{1}{\sigma} \right) \circ (\mathbf{s} - \mathbf{M}) \in \mathfrak{R}^N \tag{7}$$

and the I part is defined by

$$\mathbf{I} = [h_1, \dots, h_k] \tag{8}$$

where

$$h_k = \frac{1}{|\mathcal{U}_k|} \sum_{j \in \mathcal{U}_k} z_j \tag{9}$$

Figure 1 is an example of computing I part using 4 IIR masks from testing images.

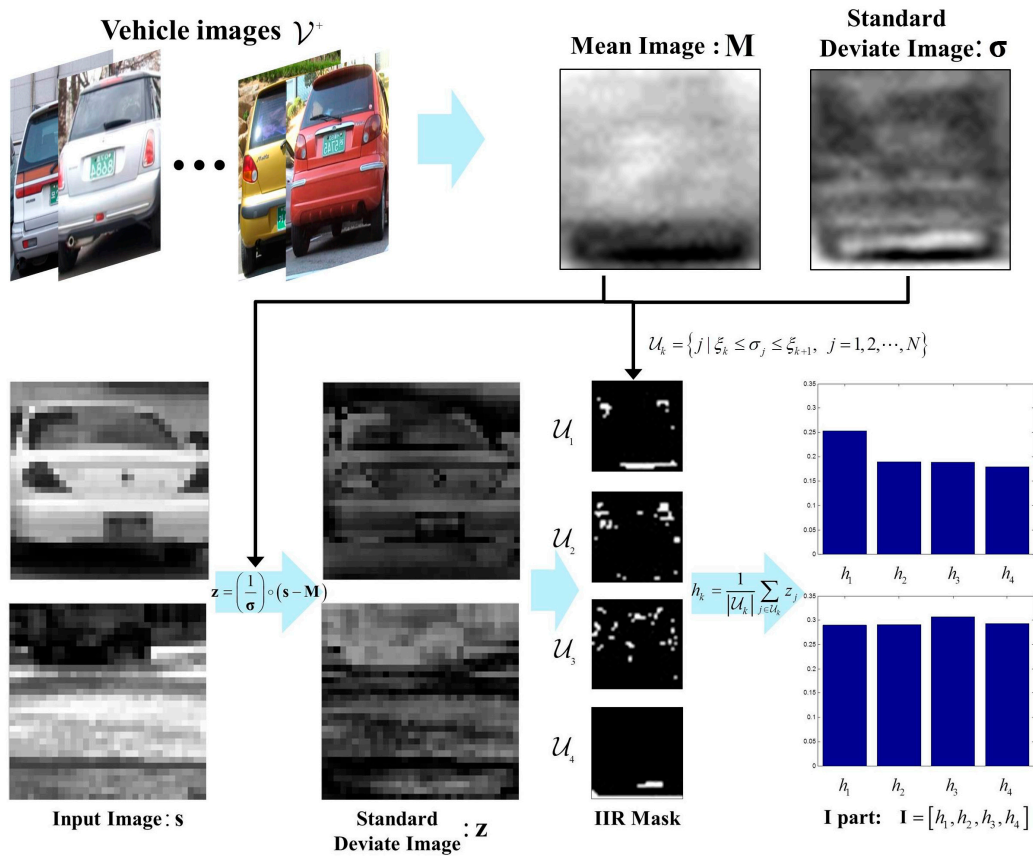


Figure 1. Examples of computing I part using 4 IIR masks [16].

Finally, the  $\pi$ HOG is defined as a concatenation of the three parts, the HOG, the P part and the I part.

### 3. Proposed Method

Figure 2 shows some examples of pedestrians in thermal images. As shown in the figure, pedestrian detection using a thermal sensor is quite different from PD using a visible sensor owing to the characteristics of the thermal images.

In thermal images, pedestrians appear brighter than the background and they do not include any color information, only silhouettes. In addition, their intensities vary according to changes in the

weather because the thermal sensors visualize temperature radiation from the objects in the images. Therefore, it is important to extract features that can reliably capture pedestrian silhouette under various weather conditions in thermal images.



**Figure 2.** Examples of thermal images in KAIST dataset.

In previous works [1,5], the HOG was popularly used for nighttime pedestrian detection because it captures the appearance of pedestrians by stacking gradient information. However, in this paper, a new feature called the  $T\pi$ HOG is proposed to improve nighttime PD performance of the HOG. The proposed  $T\pi$ HOG is based on the  $\pi$ HOG [16] and it is developed so that the  $T\pi$ HOG has more distinctiveness than the HOG when thermal sensors are used. The  $T\pi$ HOG includes not only thermal gradient information but also its locations and thermal intensities. The  $T\pi$ HOG is not a simple application of the  $\pi$ HOG to thermal images, but it is redesigned to handle PD problems in thermal images.

In addition, instead of the linear SVM, the additive kernel support vector machine (AKSVM) is used as a classifier to enhance the detection performance, as well as the detection time.

### 3.1. Thermal Position Intensity Histogram of Oriented Gradient ( $T\pi$ HOG)

The  $\pi$ HOG takes longer time and thus, computationally more expensive than the HOG since the  $\pi$ HOG requires additional pixel-wise computation to compute the mean of pixel locations. However, the pixel-wise computation in the  $\pi$ HOG is not suitable for commercial PD because it requires real time operation. Thus, in the proposed  $T\pi$ HOG, the cell-wise approach, and not the pixel-wise approach, was adopted to reduce the computational time. Since the cell values have already been computed in extracting the HOG, less computation is required to compute the  $T\pi$ HOG compared with the conventional HOG.

Furthermore, unlike the **I** part based on the IIR in [16], the **I** part in this paper was increased such that it has the same size as the orientation channel of the HOG. This is because the **I** part in the original work [16] used only 4 values from the 4 IIR masks as features and it had relatively small effects on the PD performance compared with the **P** or the HOG parts.

The  $T\pi$ HOG consists of four parts: the **T** part, the Position (**P**) part, the Intensity (**I**) part, and the conventional HOG. In the conventional HOG, we used the HOG of [32] which has 27 gradient channels (18 signed orientations, 9 unsigned orientations) and 4 gradient energy channels using different normalization methods. A detailed description of these four parts is presented in following subsections.

#### 3.1.1. T Channel Part

For the first part of the  $T\pi$ HOG, we used the **T** channel proposed in [33]. The **T** channel is defined as an aggregated version of a thermal image. For example, given  $64 \times 32$  IR images and  $4 \times 4$  cell size, the **T** channel has  $16 \times 8$  cells and the value in each cell is the sum of pixel intensities within the cell. Figure 3 shows an example of an IR image and its **T** channel.

Unlike the visible image, the pedestrians have higher pixel intensity values than backgrounds in IR images. Thus, **T** channel that consists of aggregations of IR intensities can play an important role in classifying pedestrians from other objects.

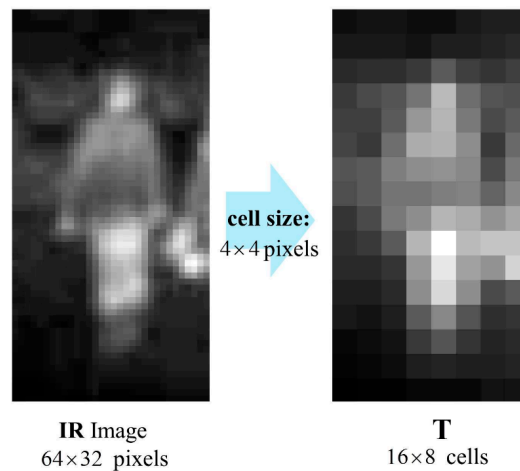


Figure 3. Example of IR image and its T channel.

### 3.1.2. Position Part

In the **P** part of the  $T\pi$ HOG, cell locations of the gradients and not pixel locations, are used unlike in the  $\pi$ HOG. In the feature implementation, the HOG consists of multiple orientation channels and each channel contains the bin value for the corresponding orientation of a cell histogram. In Figure 4, shown are the examples of HOG that has  $16 \times 8$  cells with 9 gradient orientations.

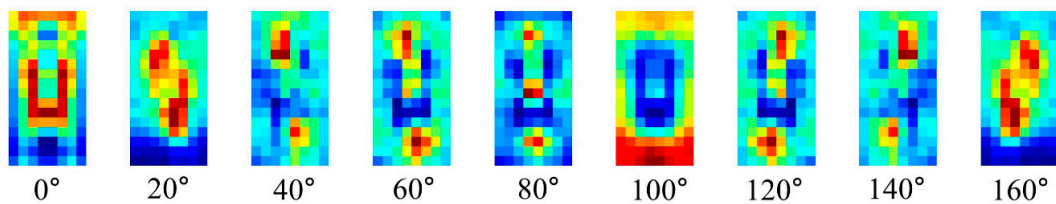


Figure 4. Example of HOG channels with 9 orientations.

In Figure 4, the values in each channel denote the bin values of cell histogram for the corresponding orientation. For example, the first channel of the HOG contains the first bin value of the cell histogram. In computing the **P** part of the  $T\pi$ HOG, we divide the HOG cells into several blocks as shown in Figure 5.

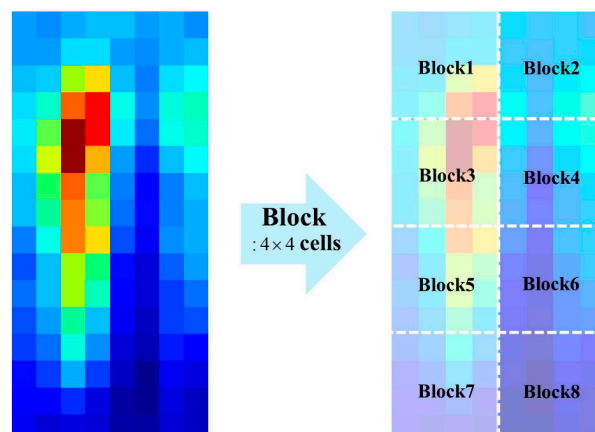


Figure 5. Example of dividing HOG into 8 blocks.

The  $\mathbf{P}$  part is defined as the  $(x, y)$  location in which each orientation component exists in a block. Assuming that  $HOG(B, x, y, d)$  is the value of a cell located at  $(x, y)$  of the  $B$ th block in the  $d$ th orientation channel, then the  $\mathbf{P}$  part is defined by

$$P_d^B = [M_{x,d}^B, M_{y,d}^B] \tag{10}$$

where

$$M_{x,d}^B = \frac{\sum_{x=1}^{B_x} \sum_{y=1}^{B_y} x \mathbb{I}[HOG(B, x, y, d) > \tau_d]}{\sum_{x=1}^{B_x} \sum_{y=1}^{B_y} \mathbb{I}[HOG(B, x, y, d) > \tau_d]} \tag{11}$$

$$M_{y,d}^B = \frac{\sum_{x=1}^{B_x} \sum_{y=1}^{B_y} y \mathbb{I}[HOG(B, x, y, d) > \tau_d]}{\sum_{x=1}^{B_x} \sum_{y=1}^{B_y} \mathbb{I}[HOG(B, x, y, d) > \tau_d]}$$

where  $\tau_d$  is the threshold for each orientation. Figure 6 shows an example of the computation of the two values for  $P_2^3$ , the 3rd block in the second orientation channel, when the block size is  $4 \times 4$  cells with 9 orientations.

In Figure 6, only the cells with values of that exceed the threshold  $\tau_2$  are used to compute the  $\mathbf{P}$  part  $P_2^3$  of the  $T\tau$ HOG. Similarly, the  $\mathbf{P}$  part contains location information of each orientation channel and is computed by

$$\mathbf{P} = [P_1, \dots, P_9] \tag{12}$$

where  $\mathbf{P}_d = [P_d^1, \dots, P_d^8]$ .

For the sake of better understanding of  $\mathbf{P}$  part, the HOG and  $\mathbf{P}$  part are visualized in Figures 7–9. Shown in Figure 7 are the examples of pedestrians IR images and they are adopted from KAIST pedestrian dataset [33].

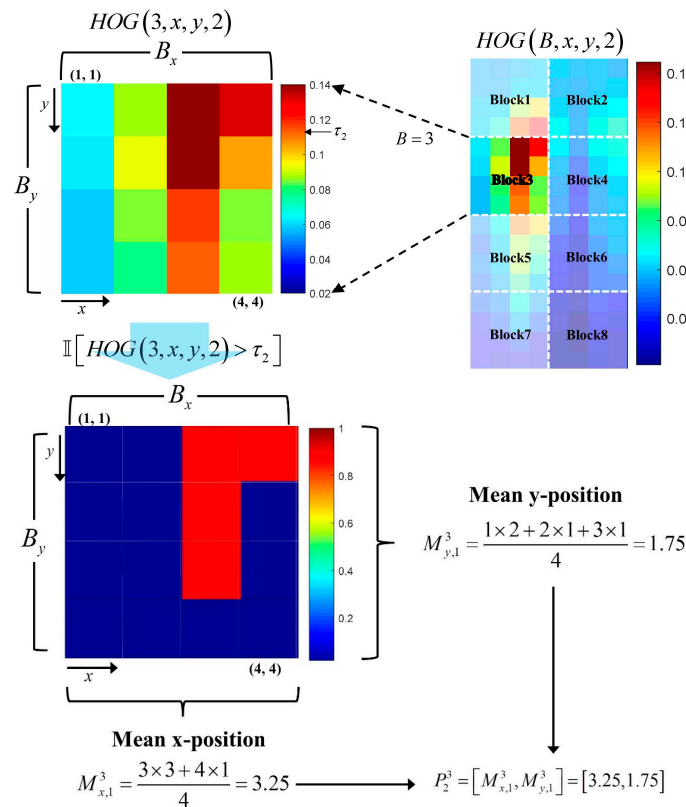


Figure 6. Example of  $\mathbf{P}$  part extraction.

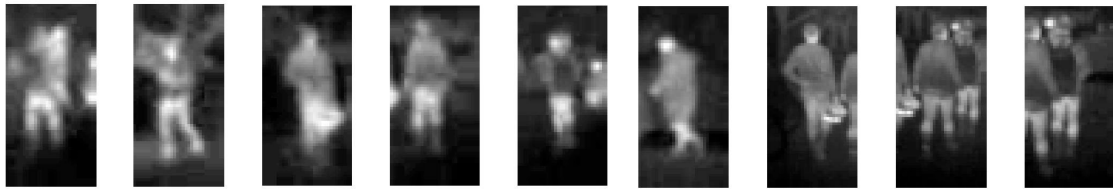


Figure 7. Examples of pedestrian IR images in KAIST pedestrian dataset [33].

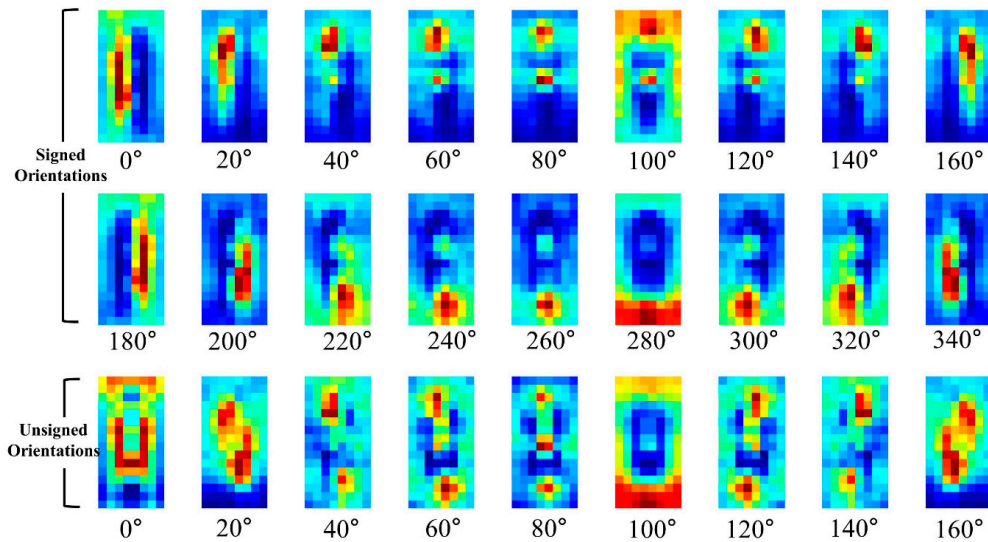


Figure 8. Average gradient channels of HOG for pedestrians.

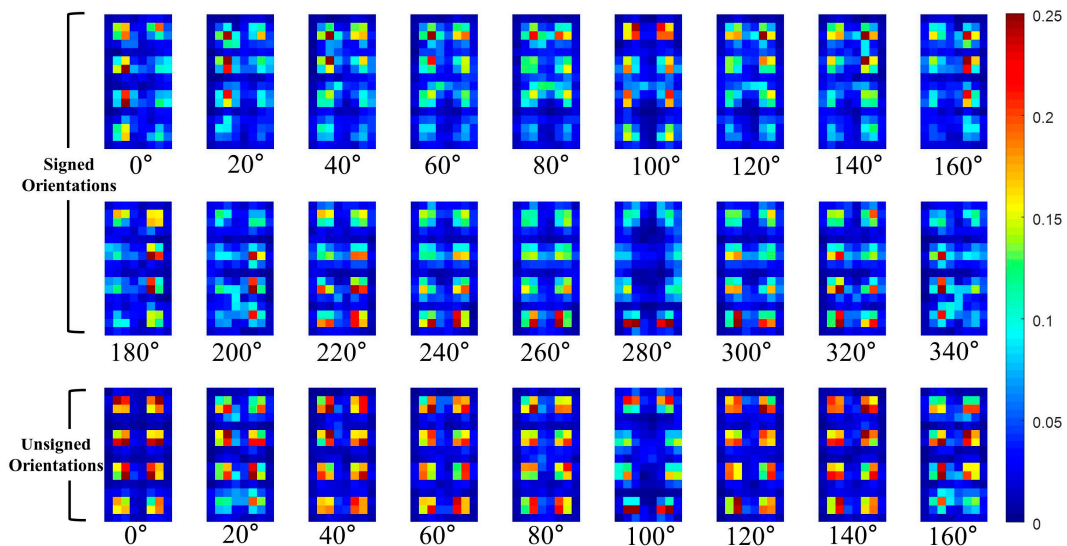


Figure 9. Average of P parts of  $T\pi$ HOG for pedestrians.

The average HOG channels for training pedestrian data are visualized in Figure 8. The average channels are computed using  $64 \times 32$  cropped IR images of 2244 pedestrians. In the figure, the first two rows indicate the HOGs for signed 18 orientations while the third row indicates the HOGs for unsigned 9 orientations.

In Figure 8, the closer the cell is to the red color, the higher the corresponding HOG. As shown in the figure, pedestrians have specific cell parts that have relatively high values for each orientation



channel. The conventional HOG uses only these values as the feature but the  $T\pi$ HOG also uses the cell locations of the orientations as well for nighttime PD.

In this paper, the  $\mathbf{P}$  part of the  $T\pi$ HOG is extracted from non-overlapped blocks that comprising of  $4 \times 4$  cells. For example, assuming the HOG has 31 channels (18 signed orientations, 9 unsigned orientations, 4 different normalizations), the size of the HOG is  $16 \times 8 \times 31$  cells and it has  $4 \times 2 \times 31 = 248$  blocks. The  $\mathbf{P}$  part  $P_o^B = [M_{x,o}^B, M_{y,o}^B]$  of the  $T\pi$ HOG is extracted from each block and the  $\mathbf{P}$  part has additional 496 values. Figures 9 and 10, show the visualization of the average  $\mathbf{P} = [\mathbf{P}_1, \dots, \mathbf{P}_9]$  of the  $T\pi$ HOG for pedestrians and non-pedestrians, respectively. In Figures 9 and 10, the average  $\mathbf{P}$  parts are computed for 2244 pedestrian images and 5000 non-pedestrian images, respectively. All the images are  $64 \times 32$  cropped IR images.

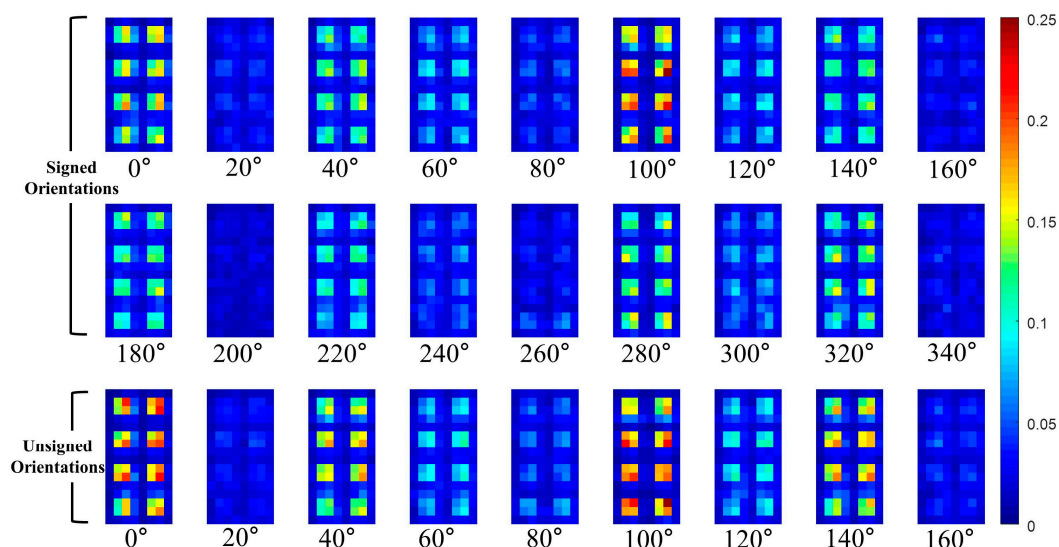


Figure 10. Average of  $\mathbf{P}$  parts of  $T\pi$ HOG for non-pedestrians.

As shown in Figures 9 and 10, the average  $\mathbf{P}$  parts of the pedestrians are focused on a couple of cell locations for each block. In particular, the average  $\mathbf{P}$  parts of the pedestrians are mostly larger than those of the non-pedestrians except for unsigned orientation of  $0^\circ$  and  $100^\circ$ . Instead, the average  $\mathbf{P}$  parts of the non-pedestrians usually have lower values than those of pedestrians. Further, in the orientations of  $20^\circ$  and  $160^\circ$ , the  $\mathbf{P}$  parts of non-pedestrians have the values close to 0 and they are not included in computation of  $\mathbf{P}$  parts. This difference between the two classes provides the  $T\pi$ HOG with discriminatory power for robust PD compared with the HOG.

### 3.1.3. Intensity Part

The conventional  $\mathbf{I}$  part of the  $\pi$ HOG is defined as a partial pixel-wise sum of the standard deviate image [16] within the IIR. Thus, the evaluation of the  $\mathbf{I}$  part requires pixel-wise computation; however, it is obviously computationally expensive for real-time application. Furthermore, the conventional  $\mathbf{I}$  part in [16] is 4 dimensions long and too short compared with 3968 ( $16 \times 8 \times 31$ ) dimensions of the HOG, thereby producing minimal effect on the PD performance. In this paper, a modified version of the  $\mathbf{I}$  part is developed for the PD in thermal images. Rather than using IIR masks, the new  $\mathbf{I}$  part is directly computed from a normal standard deviate image of the set of  $\mathbf{T}$  channels which are computed from training pedestrian data ( $64 \times 32$  cropped IR images of pedestrians). Therefore, the feature length of the  $\mathbf{I}$  part is the same as that of the  $\mathbf{T}$  channel. That is, given the  $\mathbf{T}$  channel set of pedestrian images  $\mathbf{T}^+ = \{\mathbf{T}_1, \mathbf{T}_2, \dots, \mathbf{T}_{N_p}\}$  where the superscript '+' means the positive pedestrian samples,  $N_p$  is the number of the  $\mathbf{T}$  channels in  $\mathbf{T}^+$  and  $\mathbf{T} \in \mathbb{R}^{16 \times 8}$ .  $\mathbf{M}_T$  and  $\sigma_T$  are the mean and standard deviation of  $\mathbf{T}^+$ , respectively, and the  $\mathbf{I}$  part for a testing  $\mathbf{T}$  channel  $\mathbf{s}_T$  is computed by

$$\mathbf{I}_T = \left| \left( \frac{1}{\sigma_T} \right) \circ (\mathbf{s}_T - \mathbf{M}_T) \right| \quad (13)$$

How to compute the  $\mathbf{I}$  part from both pedestrian and non-pedestrian testing images is summarized in Figure 11.

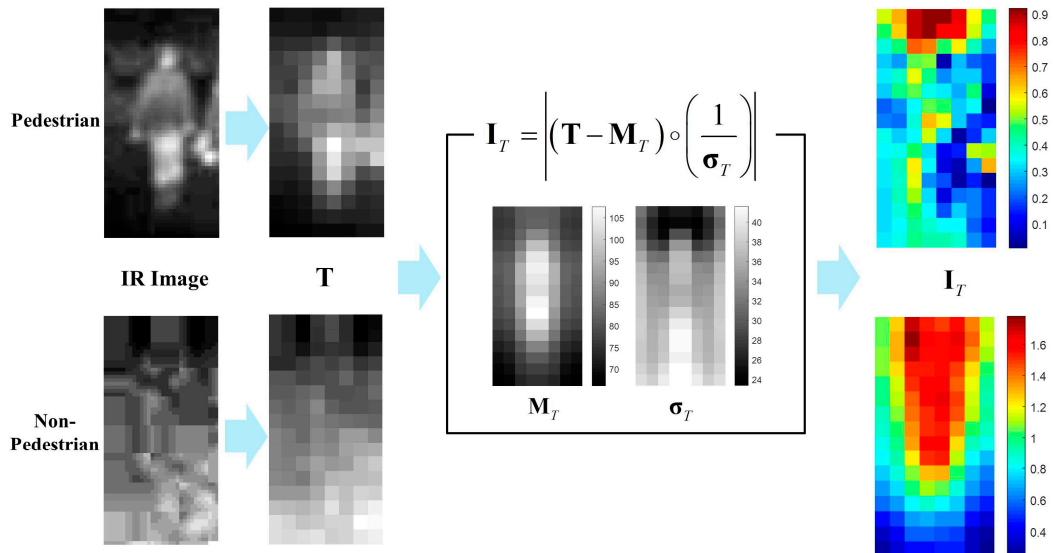


Figure 11. Example of computing  $\mathbf{I}$  part from testing images.

Shown in Figure 11 is the average of  $\mathbf{I}$  parts for pedestrians and non-pedestrians. The images are  $64 \times 32$  cropped IR images of testing dataset of KAIST pedestrians Dataset.

As shown in Figure 12, most of the average  $\mathbf{I}$  parts for pedestrian images are less than 1. In particular, the parts corresponding to the lower body are less than 0.7. On the other hand, the average  $\mathbf{I}$  parts for non-pedestrian generally are larger than 0.7 and the parts corresponding to the upper body have the values larger than 1. This difference provides the  $\mathbf{I}$  parts with strong discriminatory power between pedestrians and non-pedestrians. Further, the extraction of the  $\mathbf{I}$  part is a cell-wise computation and does not require the additional computation for developing the histogram; therefore, the proposed  $\mathbf{I}$  part is computationally more efficient than that of the conventional  $\pi$ HOG [16].

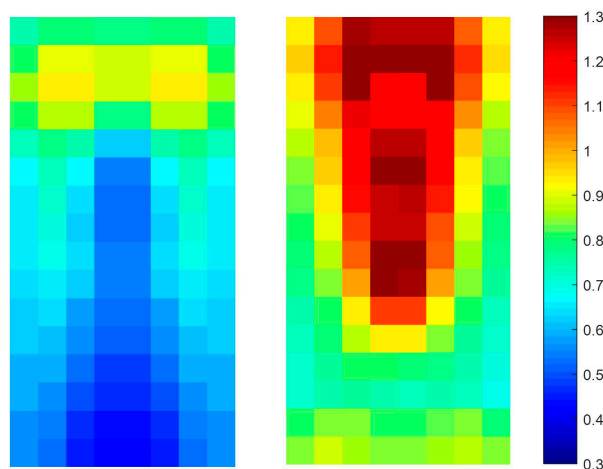


Figure 12. Average  $\mathbf{I}$  part for testing images (Left: pedestrian, Right: non-pedestrian).

### 3.2. Additive Kernel SVM (AKSVM)

The SVM is one of the popular binary classifiers used in object detection in computer vision. Given the training set  $\mathcal{S} = \left\{ \left( \mathbf{x}^{(i)}, y^{(i)} \right) \right\}_{i=1}^L$  with  $L$  samples, the SVM is trained to classify input data  $\mathbf{x}^{(i)}$  as positive ( $y^{(i)} = 1$ ) class or negative ( $y^{(i)} = -1$ ) class where  $\mathbf{x}^{(i)} = \left[ x_1^{(i)}, x_2^{(i)}, \dots, x_N^{(i)} \right]^T \in \mathbb{R}^N$  and  $y^{(i)} \in \{-1, +1\}$ . If input data  $\mathbf{x}^{(i)}$  is mapped to a higher dimensional feature space as  $\phi(\cdot)$  then the decision function of the SVM is defined by

$$f(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + b \tag{14}$$

where  $\phi(\mathbf{x}) \in \mathbb{R}^D, D \gg N, \mathbf{w} \in \mathbb{R}^D$  is the weight and  $b \in \mathbb{R}$  is the bias. The SVM is trained by finding optimal solutions of  $\mathbf{w}$  and  $b$  which maximizes the margin between the two classes. It can also be trained in dual space using the kernel trick with  $\kappa(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \phi(\mathbf{x}^{(i)}) \cdot \phi(\mathbf{x}^{(j)}) \in \mathbb{R}$ . After training the SVM in dual space, the decision function (11) can be evaluated by

$$\begin{aligned} f(\mathbf{x}) &= \sum_{i=1}^L \alpha^{(i)} y^{(i)} \kappa(\mathbf{x}^{(i)}, \mathbf{x}) + b \\ &= \sum_{i \in \mathcal{SV}} \alpha^{(i)} y^{(i)} \kappa(\mathbf{x}^{(i)}, \mathbf{x}) + b \end{aligned} \tag{15}$$

where  $\mathcal{SV} = \left\{ i \mid \alpha^{(i)} > 0 \right\}$  denotes a set of support vectors. If the  $\kappa(\cdot)$  of Equation (15) is nonlinear, the kernel SVM performs better than the linear SVM in classification. However, it requires the high computation and memory resource owing to the kernel computation with its support vectors for every testing.

However, the additive kernels (AK) enable fast computation of the decision function, while maintaining the robust performance of the kernel SVM [23,24]. The AK is defined as the kernel that can be decomposed into a summation of dimension-wise components and it is represented by  $\kappa(\mathbf{x}, \mathbf{z}) = \sum_{n=1}^N \kappa_n(x_n, z_n)$  where  $\mathbf{x} = \{x_1, x_2, \dots, x_N\} \in \mathbb{R}^N$  and  $\mathbf{z} = \{z_1, z_2, \dots, z_N\} \in \mathbb{R}^N$ . Various AKs have been reported and they include the linear kernel  $\kappa_{LIN}$ , intersection kernel  $\kappa_{IK}$ , the generalized intersection kernel  $\kappa_{GIK}$  and the  $\chi^2$  kernel  $\kappa_{\chi^2}$  defined as

$$\kappa_{IK}(\mathbf{x}, \mathbf{z}) = \sum_{n=1}^N \min(x_n, z_n) \tag{16}$$

$$\kappa_{GIK}(\mathbf{x}, \mathbf{z}) = \sum_{n=1}^N \min(|x_n|^2, |z_n|^2) \tag{17}$$

$$\kappa_{\chi^2}(\mathbf{x}, \mathbf{z}) = \sum_{n=1}^N \frac{2x_n z_n}{x_n + z_n} \tag{18}$$

The decision function of the SVM in Equation (15) with the AK can be represented by

$$\begin{aligned} f_{AK}(\mathbf{x}) &= \sum_{i=1}^L \alpha^{(i)} y^{(i)} \kappa(\mathbf{x}^{(i)}, \mathbf{x}) + b \\ &= \sum_{i=1}^L \alpha^{(i)} y^{(i)} \sum_{n=1}^N \kappa_n(x_n^{(i)}, x_n) + b \\ &= \sum_{n=1}^N \left\{ \sum_{i=1}^L \alpha^{(i)} y^{(i)} \kappa_n(x_n^{(i)}, x_n) \right\} + b \\ &= \sum_{n=1}^N h_n(x_n) + b \end{aligned} \tag{19}$$

where

$$\begin{aligned}
 h_n(x_n) &= \sum_{i=1}^L \alpha^{(i)} y^{(i)} \kappa_n(x_n^{(i)}, x_n) \\
 &= \sum_{i \in S_V} \alpha^{(i)} y^{(i)} \kappa_n(x_n^{(i)}, x_n)
 \end{aligned}
 \tag{20}$$

and  $h_n(x_n)$  is a one-dimensional function of  $x_n \in \mathfrak{R}$ . In Equation (20), the  $\alpha^{(i)}, y^{(i)}$  are given; therefore the output of  $h_n(\cdot)$  can be pre-computed for all possible input data  $x_n \in \mathfrak{R}$  and computed output values are stored in look-up-table (LUT) for each  $h_n(\cdot)$ . Assuming the  $LUT_n \in \mathfrak{R}^{N_L}$  is the LUT that consists of sampled  $N_L$  values from  $h_n(\cdot), h_n(x_n)$  of Equation (20) can be simply approximated as

$$h_n(x_n) \approx LUT_n(\lceil x_n / s \rceil)
 \tag{21}$$

where  $s = 1/N_L$  is the sampling interval on  $x_n$ . Figure 13 shows an example of retrieving value of  $h_n(x_n)$  from  $LUT_n$ . In Figure 13, the size of  $LUT_n$  is  $N_L = 25$  and the values of  $LUT_n$  are sampled from  $h_n(\cdot)$  with sample interval  $s = 0.04$ .

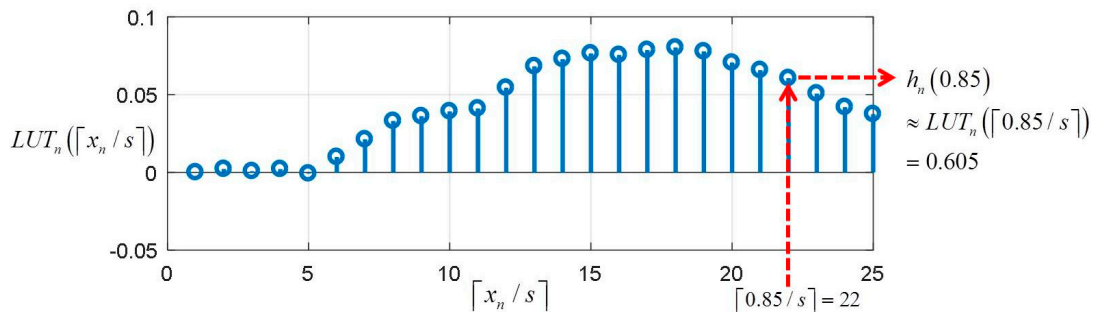


Figure 13. Example of retrieving value of  $h_n(x_n)$  from  $LUT_n$  for  $x_n = 0.85$ .

Therefore, using the LUTs of  $h_n(\cdot)$ , testing of Equation (20) can be simply carried out as the summation of values taken from the  $LUT_n$ s without kernel computation as

$$f_{AK}(\mathbf{x}) \approx \sum_{n=1}^N LUT_n(\lceil x_n / s \rceil) + b
 \tag{22}$$

### 3.3. $T\pi$ HOG-AKSVM for Nighttime PD

In this subsection, how to combine AKSVM with  $T\pi$ HOG for nighttime PD is explained. The test process of the combination of AKSVM with  $T\pi$ HOG is summarized in Figure 14.

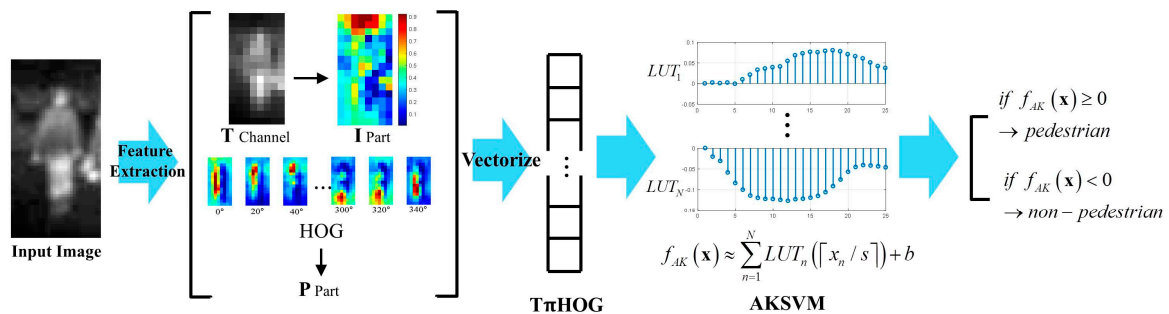


Figure 14. Test process of  $T\pi$ HOG-AKSVM for nighttime PD.

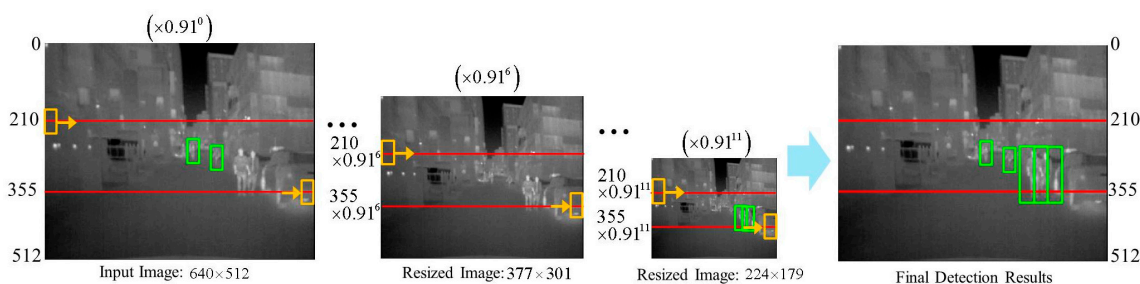
As shown in the figure, the HOG and T channel are extracted from input IR image first. Then **P** and **I** parts are extracted from HOG and T channel, respectively. All these features are vectorized and  $T\pi$ HOG is completed by concatenating these vectorized features (**T** channel, **P** part, **I** part, HOG). Then, the score of each component in  $T\pi$ HOG is read off from the LUT and the total score of  $T\pi$ HOG is computed by summing the scores of the component features. Finally, if the total score is larger than 0, the input image is classified as pedestrian. Otherwise, it is classified as non-pedestrian.

#### 4. Experimental Results

In this section, the proposed method is applied to the KAIST pedestrian dataset [14] and its performance is compared with other conventional methods. The KAIST pedestrian dataset consists of a number of pairs of visible-thermal images that are aligned in the image size of  $640 \times 512$ . The dataset images were taken by both visible and thermal sensors (FIR) in the day and nighttime at three locations (Campus, Road, Downtown). In this experiment, we use images in the nighttime of the KAIST dataset for training and testing. In the nighttime dataset, there are 838 training images with 1122 annotations and 797 test images.

In this experiment, we set the size of the ROI image to  $64 \times 32$  pixels, the cell size of the HOG to  $4 \times 4$  pixels, and the block size of the  $T\pi$ HOG to  $4 \times 4$  cells. The AKSVM classifiers are trained with  $T\pi$ HOGs of training set using the LibSVM MATLAB toolbox [24,34]. For LUTs of AKSVM, we used the LUT of  $N_L = 100$  and  $s = 0.01$ . Piotr's Computer Vision Toolbox [35] is also used for feature extraction and testing.

For testing, sliding window approach is employed to detect pedestrians with various scales. In the sliding window approach, the step size is fixed to the cell size and scale ratio is set to 1.09 ( $1/0.91$ ), which result in 8 scales per octave. As in [36,37], we narrow down the search area by restricting the  $y$ -coordinate of the center of search window to lie within 210th and 355th pixel in  $y$ -axis. Shown in Figure 15 is an example of our sliding window approach for pedestrian detection. In the figure, the yellow boxes denote the search window, green boxes are the detection results and the red lines denote the boundaries of search region within which the  $y$ -coordinate of the center of search window is restricted.



**Figure 15.** Example of detecting pedestrians of multiple scales using sliding window approach.

We compare the detection performance of the proposed method with other conventional methods [10,19] using ROC curve and the log-average miss rate. The ROC curve shows the detection performance by plotting miss rate against the false positive per image (FPPI). The lower is the ROC curve, the better the detection performance. The log-average miss rate (MR) is the average of miss rate for FPPI of  $[10^{-2}, 10^0]$  on the log scale. For comparison, we choose HOG-LinearSVM and the ACF-T-THOG [33] as a base line and the state-of-the-art, respectively. ACF-T-THOG utilizes pairs of visible-thermal images of nighttime and extracted ACF from visible images, T-THOG form thermal images. Except ACF-T-THOG, all SVM based classifier (LinearSVM, AKSVM) are trained with thermal images.

Then, the effect of the cell size on detection performance is analyzed. The ROC curves of  $T\pi$ HOG with AKSVM are plotted while varying the cell size in Figure 16. The intersection kernel  $\kappa_{IK}$  in Equation (16) is used. The subsequent results (Log average MR, time per frame) are summarized in Table 1. In this experiment, we set the number of blocks for  $P$  part as 8 blocks per orientation layer as Figure 5.

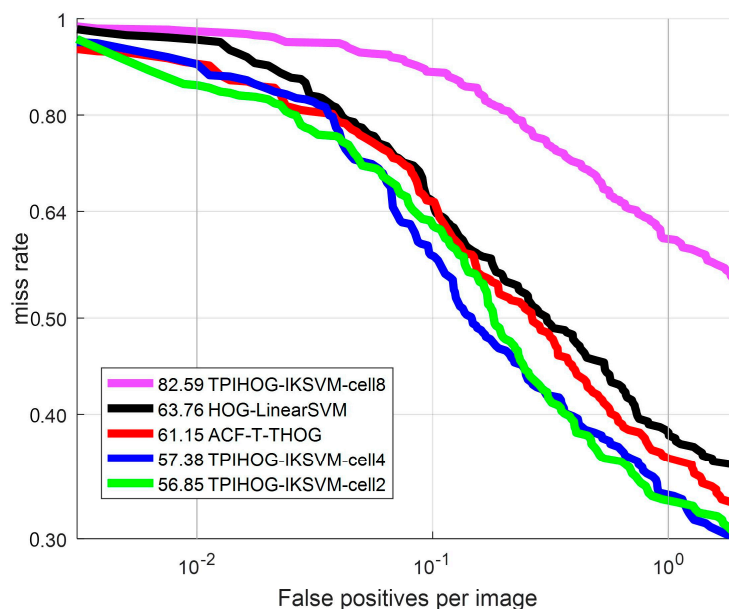


Figure 16. ROC curve: Comparison of classifiers with different cell size.

Table 1. Detailed information of feature-classifier.

	Feature	Cell Size	Log Average MR (%)	Time Per Frame (s)
ACF-T-HOG	ACF + T Channel + HOG	4	61.15	0.25
HOG-LinearSVM	HOG	4	63.76	1.78
$T\pi$ HOG-IKSVM-cell8	$T\pi$ HOG	8	82.59	0.29
$T\pi$ HOG-IKSVM-cell4	$T\pi$ HOG	4	57.38	2.84
$T\pi$ HOG-IKSVM-cell2	$T\pi$ HOG	2	56.85	39.28

As shown in Table 1 and Figure 16, the smaller the cell size of  $T\pi$ HOG is, the better detection performance is obtained.  $T\pi$ HOG-IKSVM-cell8 spends comparable detection time with ACF-T-THOG but it demonstrates about 30% higher Log average MR than ACF-T-THOG and HOG-LinearSVM. On the other hand,  $T\pi$ HOG-IKSVM-cell2 demonstrates the best performance among all the competing methods but its detection speed is about 20 times slower than  $T\pi$ HOG-IKSVM-cell4. Considering the trade-off between the detection performance (log average MR) and detection time, the recommended cell size is 4.

To compare the discriminating power of the feature, we plot the ROC curves for several feature combinations with IKSVM in Figure 17. Table 2 presents the detailed information of the feature-classifier combinations used in experiments.

As shown in Figure 17, the THOG-LinearSVM performs better than the HOG-LinearSVM by 0.08% in terms of the log average MR. and THOG-IKSVM shows better detection performance than HOG-IKSVM by 1.2%. From this result, it can be observed that adding the T channel to the HOG improves detection performance from the conventional HOG. To examine the effect of the additive kernel, we measure the detection performance of the THOG-IKSVM. As shown in Figure 17 and Table 2, THOG-IKSVM demonstrates improved performance from THOG-LinearSVM indicating that the additive kernel resulted in significant improved detection performance of the SVM by 4.98% from the linear kernel.

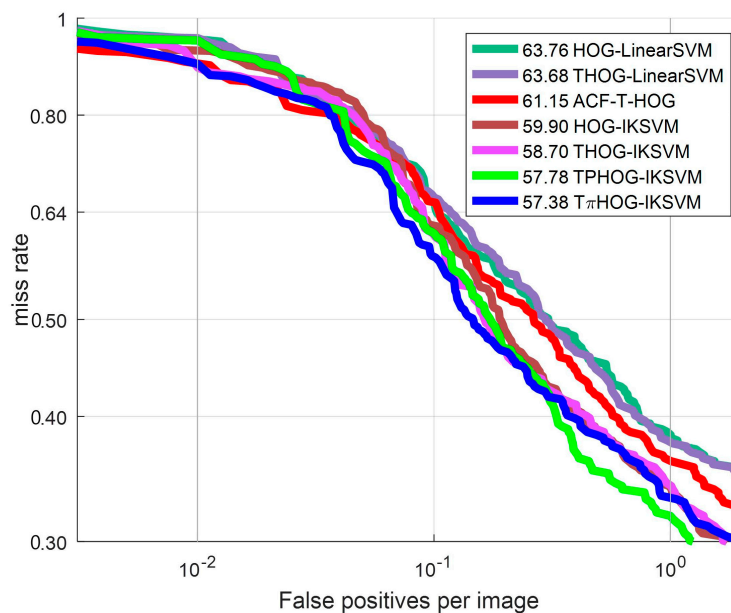


Figure 17. ROC curve: Comparison of feature.

Table 2. Detailed information of feature-classifier.

	Feature	Classifier	Log Average MR (%)	Time Per Frame (s)
ACF-T-HOG	ACF + T Channel + HOG	Decision Tree	61.15	0.25
HOG-LinearSVM	HOG	Linear SVM	63.76	1.78
HOG-IKSVM	HOG	AK SVM (Intersection Kernel)	59.90	1.78
THOG-LinearSVM	T Channel + HOG	Linear SVM	63.68	1.88
THOG-IKSVM	T Channel + HOG	AK SVM (Intersection Kernel)	58.70	1.88
TPHOG-IKSVM	THOG + P part	AK SVM (Intersection Kernel)	57.78	2.69
T $\pi$ HOG-IKSVM	TPHOG + I part	AK SVM (Intersection Kernel)	57.38	2.84

Finally, we measure the detection performance using the TPHOG and the T $\pi$ HOG in connection with the AKSVM. The TPHOG which adds the P part to THOG demonstrates 0.92% enhancement from THOG-IKSVM. The T $\pi$ HOG which adds I part to TPHOG demonstrates improved performance from TPHOG by 0.4% and outperforms all other methods. From these experimental results, it can be seen that the T $\pi$ HOG-AKSVM enhance the performance of conventional method in terms of both feature and classifier performance.

In terms of detection speed, the  $T\pi$ HOG-IKSVM takes more computational time than HOG-IKSVM by 1.06 s. To be specific, adding **P** part takes 0.8 s more time than HOG while **I** part and **T** channels requires additional 0.1 s compared with HOG.

Shown in Figure 18, shown are the ROC curves for the  $T\pi$ HOG with four different additive kernels: linear kernel, intersection kernel  $\kappa_{IK}$ , generalized intersection kernel  $\kappa_{GIK}$  and  $\chi^2$  kernel  $\kappa_{\chi^2}$ . The computational time of  $T\pi$ HOG-AKSVMs in Figure 16 are the same as one of  $T\pi$ HOG-IKSVM of Table 2 because they are performed with LookupTable of same size in this experiment.

As shown in Figure 18, all types of  $T\pi$ HOG-AKSVMs performs better than the HOG-LinearSVM and  $T\pi$ HOG-LinearSVM. Among the AKSVMs, the  $T\pi$ HOG-IKSVM and  $T\pi$ HOG-GIKSVM demonstrate the significant improvement from the  $T\pi$ HOG-LinearSVM, and they also perform better than the ACF-T-HOG. In Figure 19, the examples of the detection results for  $T\pi$ HOG-IKSVM and ACF-T-HOG are compared

As shown in Figure 19, the ACF-T-HOG generates a lot of false positives to the vertical objects such as headlight or buildings. On the other hand, the proposed method shows good detection results with no false positive.

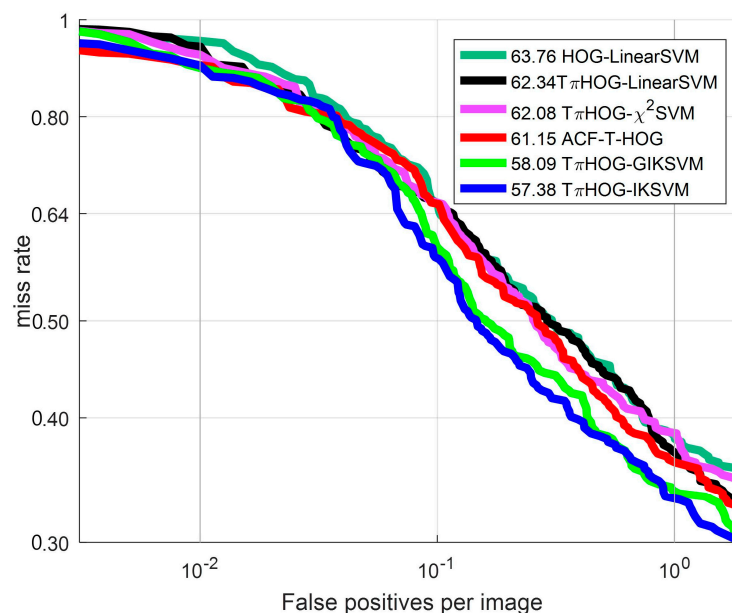


Figure 18. ROC curve: Comparison of additive kernel



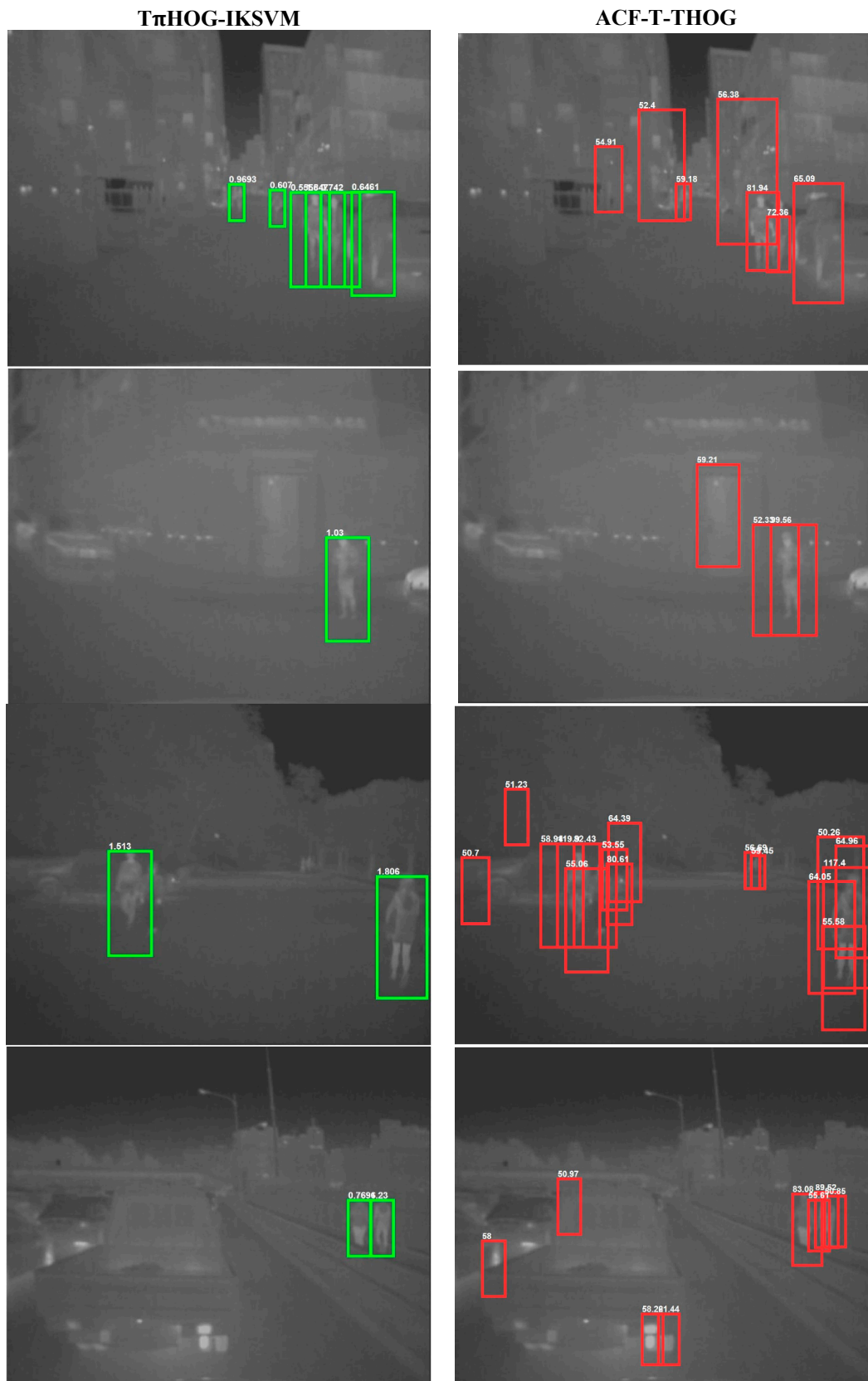


Figure 19. Detection results on test images of KAIST DB.

## 5. Conclusions

In this paper, a novel night-time pedestrian detection method using a thermal camera has been proposed. A new feature named  $T\pi$ HOG was developed and it was combined with AKSVM. The proposed  $T\pi$ HOG has more robust discriminative power than HOG because it uses not only the gradient information but also cell location of the gradient for each orientation channel. The proposed method was applied to KAIST pedestrian dataset and results show that its detection performance improved compared with other conventional methods for pedestrian detection in the nighttime. A comparison of experimental results with KAIST pedestrian dataset shows that the  $T\pi$ HOG performs better than the HOG in the distinctiveness of feature and the  $T\pi$ HOG-AKSVM shows better performance than other conventional methods.

**Acknowledgments:** This work was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology under Grant NRF-2016R1A2A2A05005301.

**Author Contributions:** Jeonghyun Baek and Euntai Kim designed the algorithm, and carried out the experiment, analyzed the result, and wrote the paper; Sungjun Hong and Jisu Kim analyzed the data and gave helpful suggestion on this research.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Hurney, P.; Waldron, P.; Morgan, F.; Jones, E.; Glavin, M. Night-time Pedestrian Classification with Histograms of Oriented Gradients-Local Binary Patterns Vectors. *IET Intell. Transp. Syst.* **2015**, *9*, 75–85. [[CrossRef](#)]
2. European Road Safety Observatory. Traffic Safety Basis Facts 2012—Pedestrian. Available online: [http://roderic.uv.es/bitstream/handle/10550/30206/BFS2012\\_DaCoTA\\_INTRAS\\_Pedestrians.pdf?sequence=1&isAllowed=y](http://roderic.uv.es/bitstream/handle/10550/30206/BFS2012_DaCoTA_INTRAS_Pedestrians.pdf?sequence=1&isAllowed=y) (accessed on 11 August 2017).
3. Zhao, X.Y.; He, Z.X.; Zhang, S.Y.; Liang, D. Robust pedestrian detection in thermal infrared imagery using a shape distribution histogram feature and modified sparse representation classification. *Pattern Recognit.* **2015**, *48*, 1947–1960. [[CrossRef](#)]
4. Ojala, T.; Pietikainen, M.; Maenpaa, T. Multiresolution Gray-scale and Rotation Invariant Texture Classification with Local Binary Patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 971–987. [[CrossRef](#)]
5. Wang, X.; Han, T.X.; Yan, S. An HOG-LBP Human Detector with Partial Occlusion Handling. In Proceedings of the 2009 IEEE 12th Conference on Computer Vision (ICCV), Kyoto, Japan, 29 September–4 October 2009.
6. Ko, B.; Kim, D.; Nam, J. Detecting Humans using Luminance Saliency in Thermal Images. *Opt. Lett.* **2012**, *37*, 4350–4352. [[CrossRef](#)] [[PubMed](#)]
7. Jeong, M.R.; Kwak, J.; Son, E.; Ko, B.; Nam, J. Fast Pedestrian Detection using a Night Vision System for Safety Driving. In Proceedings of the 2014 International Conference on Computer Graphics, Imaging and Visualization (CGIV), Singapore, 6–8 August 2014.
8. Dai, C.; Zheng, Y.; Li, X. Pedestrian Detection and Tracking in Infrared Imagery Using Shape and Appearance. *Comput. Vis. Image Underst.* **2007**, *106*, 288–299. [[CrossRef](#)]
9. Wang, J.; Chen, D.; Chen, H.; Yang, J. On Pedestrian Detection and Tracking in Infrared Videos. *Pattern Recognit. Lett.* **2012**, *33*, 775–785. [[CrossRef](#)]
10. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Diego, CA, USA, 20–26 June 2005; pp. 886–893.
11. Dollar, P.; Wojek, C.; Schiele, B.; Perona, P. Pedestrian Detection: An Evaluation of the State of the Art. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 743–761. [[CrossRef](#)] [[PubMed](#)]
12. Watanabe, T.; Ito, S.; Yokoi, K. Co-occurrence histograms of oriented gradients for pedestrian detection. In Proceedings of the 2009 Conference on Pacific-Rim Symposium on Image and Video Technology, Tokyo, Japan, 13–16 January 2009; pp. 37–47.

13. Andavarapu, N.; Vatsavayi, V.K. Weighted CoHOG (W-CoHOG) Feature Extraction for Human Detection. In *Proceedings of Fifth International Conference on Soft Computing for Problem Solving*; Springer: Singapore, 2016; Volume 437, pp. 273–283.
14. Hua, C.; Makihara, Y.; Yagi, Y. Pedestrian detection by using a spatio-temporal histogram of oriented gradients. *IEICE Trans. Inf. Syst.* **2013**, *96*, 1376–1386. [[CrossRef](#)]
15. Qi, B.; John, V.; Liu, Z.; Mita, S. Pedestrian detection from thermal images with a scattered difference of directional gradients feature descriptor. In *Proceedings of the 17th IEEE Conference on Intelligent Transportation System (ITSC)*, Qingdao, China, 8–11 October 2014; pp. 2168–2173.
16. Kim, J.; Baek, J.; Kim, E. A Novel On-road Vehicle Detection Method Using  $\pi$ HOG. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 3414–3429. [[CrossRef](#)]
17. Xu, F.; Liu, X.; Fujimura, K. Pedestrian Detection and Tracking with Night Vision. *IEEE Trans. Intell. Transp. Syst.* **2005**, *6*, 63–71. [[CrossRef](#)]
18. Bertozzi, M.; Broggi, A.; Del Rose, M.; Felisa, M.; Rakotomamonjy, A.; Suard, F. A Pedestrian Detector Using Histograms of Oriented Gradients and a Support Vector Machine Classifier. In *Proceedings of the IEEE Conference on Intelligent Transportation System Conference (ITSC)*, Seattle, WA, USA, 30 September–3 October 2007.
19. O'Malley, R.; Jones, E.; Glavin, N. Detection of Pedestrians in Far-Infrared Automotive Night Vision Using Region-Growing and Clothing Distortion Compensation. *Infrared Phys. Technol.* **2010**, *53*, 439–449. [[CrossRef](#)]
20. Baek, J.; Kim, J.; Kim, E. Fast and Efficient Pedestrian Detection via the Cascade Implementation of an Additive Kernel Support Vector Machine. *IEEE Trans. Intell. Transp. Syst.* **2017**, *16*, 3414–3429. [[CrossRef](#)]
21. O'Malley, R.; Glavin, M.; Jones, E. An Efficient Region of Interest Generation Technique for Far-Infrared Pedestrian Detection. In *Proceedings of the International Conference on Consumer Electronics (ICCE)*, Las Vegas, NV, USA, 9–13 January 2008; pp. 1–2.
22. Sun, H.; Wang, C.; Wang, B. Night Vision Pedestrian Detection Using a Forward-looking Infrared Camera. In *Proceedings of the 2011 International Workshop on Multi-Platform/Multi-Sensor Remote Sensing and Mapping (M2RSM)*, Xiamen, China, 10–12 January 2011; pp. 1–4.
23. Maji, S.; Berg, A.C.; Malik, J. Efficient Classification for Additive Kernel SVMs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 66–77. [[CrossRef](#)] [[PubMed](#)]
24. Vedaldi, A.; Zisserman, A. Efficient Additive Kernels via Explicit Feature Maps. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 480–492. [[CrossRef](#)] [[PubMed](#)]
25. Kim, J.H.; Hong, H.G.; Park, K.R. Convolutional Neural Network-Based Human Detection in Nighttime Images Using Visible Light Camera Sensors. *Sensors* **2017**, *17*, 1065.
26. Liu, J.; Zhang, S.; Wang, S.; Metaxas, D.N. Multispectral deep neural networks for pedestrian detection. *arXiv*, **2016**, arXiv:1611.02644.
27. Wagner, J.; Fischer, V.; Herman, M.; Behnke, S. Multispectral pedestrian detection using deep fusion convolutional neural networks. In *Proceedings of the 24th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)*, Bruges, Belgium, 27–29 April 2016; pp. 509–514.
28. Cai, Y.; Sun, X.; Wang, H.; Chen, L.; Jiang, H. Night-Time Vehicle Detection Algorithm Based on Visual Saliency and Deep Learning. *J. Sens.* **2016**, *2016*, 8046529. [[CrossRef](#)]
29. John, V.; Mita, S.; Liu, Z.; Qi, B. Pedestrian detection in thermal images using adaptive fuzzy C-means clustering and convolutional neural networks. In *Proceedings of the 14th IAPR International Conference on Machine Vision Applications (MVA)*, Tokyo, Japan, 18–22 May 2015; pp. 246–249.
30. Biswas, S.K.; Milanfar, P. Linear support tensor machine with LSK channels: Pedestrian detection in thermal infrared images. *IEEE Trans. Image Process.* **2017**, *22*, 4229–4242. [[CrossRef](#)] [[PubMed](#)]
31. Choi, E.; Lee, W.; Lee, K.; Kim, J.; Kim, J. Real-time pedestrian recognition at night based on far infrared image sensor. In *Proceedings of the 2nd International Conference on Communication and Information Processing*, Singapore, 26–29 November 2016; pp. 115–119.
32. Felzenszwalb, P.F.; Girshick, R.B.; McAllester, D.; Ramanan, D. Object Detection with Discriminatively Trained Part-based Models. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *32*, 1627–1645. [[CrossRef](#)] [[PubMed](#)]
33. Hwang, S.; Park, J.; Kim, N.; Choi, Y.; Kweon, I. Multispectral Pedestrian Detection: Benchmark Dataset and Baseline. In *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, 7–12 June 2015; pp. 1037–1045.

34. Chang, C.-C.; Linm, C.-J. LIBSVM: A Library for Support Vector Machines. *ACM Trans. Intell. Syst. Technol.* **2011**, *2*, 27. [[CrossRef](#)]
35. Doll'ar, P. Piotr's Computer Vision Matlab Toolbox (PMT). Available online: <http://vision.ucsd.edu/pdollar/toolbox/doc/index.html> (accessed on 2 December 2016).
36. Daniel Costea, A.; Nedeveschi, S. Semantic channels for fast pedestrian detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2360–2368.
37. Baek, J.; Hong, S.; Kim, J.; Kim, E. Bayesian learning of a search region for pedestrian detection. *Multimed. Tools Appl.* **2016**, *75*, 863–885. [[CrossRef](#)]



© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).