IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 18, 2018, accepted January 5, 2019, date of publication January 18, 2019, date of current version February 27, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2893320

A Novel Feature Selection in Vehicle Detection Through the Selection of Dominant Patterns of Histograms of Oriented Gradients (DPHOG)

NATTHARIYA LAOPRACHA, KHAMRON SUNAT, AND SIRAPAT CHIEWCHANWATTANA
Department of Computer Science, Faculty of Science, Khon Kaen University, Khon Kaen 40002, Thailand

Corresponding author: Khamron Sunat (khamron_sunat@yahoo.com)

This work was supported in part by the Thailand Research Fund under Grant number RTA6080013, and in part by the Mahasarakham University, Thailand.

ABSTRACT This paper proposes a novel method that addresses the selection of the dominant patterns of the histograms of oriented gradients (DPHOGs) in vehicle detection. HOG features lead to an expensive classification with high misclassification rates since HOG generates a long vector containing both redundant and ambiguous features (similarities between the vehicle and non-vehicle images). Several modifications of HOG were proposed to resolve these issues such as the vertical histograms of oriented gradient and one that includes position and intensity with HOG; however, these methods still contain some ambiguous features. A feature selection method can exclude these ambiguous features, allowing for better classification rates and a reduction in classification times. The proposed method uses the ideal vectors of the vehicle and non-vehicles images for selecting features in dominant patterns. The segments indicating the differences between the vehicle and non-vehicle classes are the dominant patterns, in which the length of the feature vector is shortened. We performed DPHOG on three standard datasets, in which the kernel extreme learning machine, the support vector machine, K-nearest neighbor, random forest, and deep neural network were used as classifiers. We then compared the performance of the DPHOG with eight well-known feature selection methods and three existing feature extraction methods for vehicle detection. In evaluations with each comparative method concerning the accuracy, true positive, false positive, and F1-score, the DPHOG presented the highest performances with less running time in each dataset.

INDEX TERMS Vehicle detection, feature selection, dominant patterns, histograms of oriented gradients.

I. INTRODUCTION

Intelligent vehicle detection systems have been developed to assist and be of benefit to vehicle drivers and passengers [1], [2]. The systems have been applied in many applications, such as automatic-car systems [3], traffic surveillance [4], vehicle counting [5], and an application to find a car park automatically [6]. Computer vision-based [7] and sensor-based systems [8] are the most commonly used methods. The computer vision-based methods make the systems more robust than sensor systems in understanding and perceiving information [8]. Also, the data (as images) may be collected for further use, such as in providing evidence for intelligent security systems. The vision-based method comprises two steps [9], [10]. The first step is hypothesis generation (HG), which detects objects, and outputs either an image of a vehicle or non-vehicle. In general, the HG is classified into three basic categories; (1) knowledge-based, (2) stereo-vision-based, and (3) motion-based methods. The second step is the hypothesis verification (HV). In this step, the objects within the image are classified into either vehicles or non-vehicles. HV can be achieved through either template-based or appearance-based methods. The template-based methods compare the test image with the images in a database and output the detected objects. The appearance-based methods use machine learning for classifying objects as vehicles or non-vehicles. In general, machine learning methods use two methods for processing: (1) feature extraction/ representation modules and (2) classification. In vehicle detection, the HV step plays an important role, as it is used to verify objects as vehicles or non-vehicles. In particular, the appearance-based methods consist of the feature extraction and classification, which produce more optimal time and

are more robust in complex backgrounds than the template-based methods. However, the appearance-based methods are more challenging and develop fundamental problems concerning processing time and accuracy [10].

In the appearance-based method of the HV step, the feature vectors used to describe vehicle images are an important factor affecting both processing time and accuracy. A long feature vector can consume more time in the classification stage. Similarities between the characteristics of vehicles and non-vehicles can lead to misclassification when separating vehicles from non-vehicles. Feature extraction is also an important factor regarding both processing time and accuracy in the appearance-based method of the HV step. In vehicle detection, there are a number of feature extraction methods including Scale Invariant Feature Transform (SIFT) [11], Haar-like [12], Gabor filter [13], log-Gabor [14], and HOG [15], to name but a few. However, HOG remains a popular feature extraction method used in the HV step because it is robust in various conditions [16]; for example, low-light, low-quality images, blurred images, color variation, multi-scales of an image, etc. [15]. HOG was successfully applied in various intelligent systems with high accuracy; including traffic surveillance systems [17], unmanned aerial vehicles (UAV) [18], driver assistance systems [19], parking space management systems [20], and deep learning systems for vehicle detection [21].

HOG has often been applied to vehicle detection systems as a major feature extraction method. The technique, however, suffers from similar disadvantages in most applications that use HOG. Namely, HOG has two drawbacks where vehicle detection systems are concerned; (1) HOG produces redundant features and computational expense [15], and (2) HOG does not provide discriminative information about the vehicles and non-vehicles leading to high misclassification rates [22]. In 2013, a technique called the vertical histograms of oriented gradients (V-HOG) was proposed by Arrospide *et al*. [15] to reduce the redundant features and the computational times. Within the generation of features, V-HOG only uses the vertical direction of the gradients, as opposed to HOG, which uses both vertical and horizontal directions; as a result, V-HOG uses fewer features and has faster computational times. The V-HOG, however, yields a lower accuracy than HOG. In 2015, the πHOG was developed by Kim *et al.* [22] to improve the non-dominant features of HOG. This technique uses the positions of the orientation bins and intensity information to generate features, in addition to the features produced by the original HOG. As a result, it yielded a higher accuracy than HOG in the KITTI dataset [23], but consequently generated a more extended feature vector and longer computational time than HOG. Therefore, while feature extraction methods may produce better features describing the object, they may not always lead to better classification performance, i.e., time and accuracy. Therefore, some research studies have focused on feature selection that aims to select prominent features, as well as improve classification time and accuracy [24].

This paper aims to improve the accuracy and to reduce the computational time of HOG within the hypothesis generation (HV) step. We propose a new feature selection method, which we refer to as the *selection dominant patterns of the HOG algorithm* (DPHOG). The DPHOG eliminates some of the common features that appear in both vehicle and non-vehicle classes. The DPHOG produces a promising presentation of features between vehicles and non-vehicles, yielding very short computational times compared to the other well-known feature selection methods.

We compared the performances of two techniques; (1) the feature extraction method, and (2) the feature selection method. Feature extraction methods include the HOG, V-HOG, and πHOG. The feature selection methods compared in our work were selected from well-known feature selection methods in pattern recognition and real-world datasets, which include conditional infomax feature extraction (CIFE) [25], infinite feature selection (Inf-FS) [26], the fast correlation-based filter solution method (FCBF) [27], mutual information quotient (MIQ) [28], global MI-based feature selection via spectral relaxation (SPEC_{CMI}) [29], eigenvector centrality feature selection (EC-FS) [30], adaptive neuro-fuzzy classifier of linguistic hedge (ANFC-LH) [31], and the whale optimization algorithm with simulated annealing (WOA-SA) [32]. Both the CIFE and Inf-FS are widely used in face recognition and object detection applications. Based on the comparison results conducted on three datasets (CompCars, GTI, and KITTI), the proposed method was more successful in selecting the dominant patterns of HOG than the well-known feature selection methods, and with shorter computational times. Accordingly, the proposed technique also produced smaller sizes of feature subsets compared with the HOG, V-HOG, πHOG, WOA-SA, MIQ, Inf-FS, SPEC_{CMI}, Inf-FS, and CIFE.

The rest of the paper is organized as follows: in Section II, we review the related works involving vehicle detection and in Section III, we present the computational explanation of the proposed DPHOG and its application in vehicle detection. The experimental data and evaluation procedure, the experimental results, and the discussions are presented in Section IV, V, and VII, respectively. Our conclusions and intentions for future work are shown in Section VII.

II. RELATED WORKS

In general, hypothesis verification (HV) in appearance-based methods use the feature extraction method and classification method for object verification [10]. The current feature extraction and classification methods used in the HV step are described in the first sub-section. The feature selection methods are then described in the second sub-section.

A. FEATURE EXTRACTION AND CLASSIFICATION

In vehicle detection systems, feature extraction and classification are important for verifying vehicles and non-vehicles. The features extraction method gives a simple description of data representation [33], which facilitates training in the

classification stage. The classification method is a machine learning-based method for classifying images into the vehicle and non-vehicle classes. Examples include the Support Vector Machine (SVM) [34], Adaboost [12], extreme learning machine (ELM) [35], K-nearest neighbor (KNN) [22], random forest (RF) [36], and deep learning machine [37].

More specifically, there are many existing feature extraction methods in the literature. The SIFT [11] is a method that is capable of describing rotate and scale invariant images and is robust in changing illumination. SIFT has therefore been applied to several vehicle detection applications; however, the accuracy is not sufficient in blued or affine images. Haar-like [12] is a popular technique often employed to many computer vision applications, including vehicle detection. The drawback of Haar-like is its sensitivity under changes in illumination, thereby producing a high degree of redundant features. In the same manner, the Gabor filter [13] is also a popular technique within vehicle detection systems, yet produces a massive number of features, which present a significant drawback. The log-Gabor filter was introduced to alleviate this problem [14], but each version of the Gabor filter produced less accuracy than the HOG [14], [38]. Ammour *et al*. [39] developed a deep convolutional neural network (CNN) for extracting features of vehicles in UAV images using the SVM classifier. However, the results of the CNN with SVM produced a high rate of false positives.

Out of all of these techniques, the HOG, proposed by Dalal and Triggs [40], remains one of the most widely used algorithms for vehicle detection. Zhu *et al*. [35] applied both HOG features and gray value features for vehicle detection using the SVM, ELM, or backpropagation (BP) network to classify vehicles. They found that the features produced by HOG yielded higher accuracies than those using the gray value features. ELM coupled with HOG spent less time in classification, yet shared the same similarity values of true positive and false positive rates with SVM with HOG. Huy and Lee [41] used HOG with SVM for automobile verification in which front-view car detection counted the number of cars within a dense traffic flow in a real-time environment. Their tests, involving detecting, tracking, and counting multiple vehicles, yielded very favorable experimental results. Xu *et al*. [18] proposed a method for detecting vehicles using images from an unmanned aerial vehicle (UAV), using both Viola-Jones and HOG features to classify objects with SVM. Although their evaluations in vehicle detection produced satisfactory results, they were unable to detect turning vehicles effectively. Huang *et al.* [19] presented a driver assistance system for vehicle detection and inter-vehicle distance estimation, in which the characteristics of vehicles and non-vehicles were extracted by HOG before being classified by the SVM. The distances between the host and the front vehicle were estimated based on the locations of detected vehicles and vanishing points. The detection rates achieved accuracies up to 98.08%. More recently, the deep learning technique has been applied to vehicle detection [4]. However, the pooling layer of the deep learning system cannot filter and cope with the system's varied rotations and scales [21] and it was found to be strongly dependent on the user's choice of learning dataset [37]. With the integration of HOG, the feature extraction techniques are applied before feeding the results to deep learning, which improves the variation of an image; thereby increasing the accuracy of deep learning [21]. Vitek and Melnicuk [20] employed a wireless camera system for managing parking spaces, in which classification through HOG and SVM separated objects into vehicles and non-vehicles. Their technique conducted on a PKLot database was compared with other methods including the Hue histogram and SVM, CNN, histograms of textural features with SVM, and HOG with AdaBoost. The results performed by HOG with SVM achieved roughly 96% accuracy; superior to the Hue histogram with SVM, histograms of textural features with SVM, and HOG with AdaBoost. Additionally, HOG with SVM proved more comfortable and flexible than the CNN in preparing training images within the training stage.

In conclusion, HOG maintains its popularity in vehicle detection systems due to its ability to describe images in various lighting conditions, background complexities, and image qualities. Furthermore, HOG can improve the performance of deep learning in vehicle detection. Notably, HOG may produce some redundant or ambiguous features (features similar in both vehicle and non-vehicle images) that affect computational times and may result in a high misclassification rate within the classification stage [22]. It was for these reasons that HOG was chosen to improve the performance of vehicle detection in our proposed method.

B. FEATURE SELECTION

Feature selection plays a vital role in the classification stage of pattern recognition [42]. While feature extraction provides a significant description of an image, it may not always lead to better classification performance. Therefore, feature selection was employed, which selects prominent features, to improve classification accuracy, and reduce computation times [24]. Feature selection consists of the wrapper models and filter models. Wrapper models use classifiers for scoring subsets; whereas filter models ignore classifiers, and evaluate the prominent subsets depending on the data itself. Filters generally perform faster than wrappers but wrappers are more robust and provide better accuracy [32].

Other studies have presented well-known feature selection methods that have been applied in pattern recognition datasets, including the CIFE [25] and Inf-FS [26]. In face recognition systems, the CIFE reduces dimensions by deriving a new information decomposition model [25]. The CIFE produces lower error rates than principal component analysis (PCA) [43], linear discriminant analysis (LDA) [44], unified subspace analysis (UniSA) [45], and maximum mutual information (MMI) [46]. In object recognition, the Inf-FS generates features using graph techniques in considering the reduction of some features. The Inf-FS selects features from the CNN feature extraction method and uses linear SVM as a classifier, which was conducted on two public datasets.

The Inf-FS significantly improved the accuracy of the CNN for object detection.

Other well-known feature selection methods that have produced high accuracy in real-world datasets include the FCBF [27], MIQ [28], SPEC_{CMI} [29], EC-FS [30], ANFC-LH [31], and WOA-SA [32]. However, the MIQ, SPEC_{CMI}, EC-FS, ANFC-LH, and WOA-SA are notorious for consuming large amounts of time when selecting prominent features. While the FCBF has a fast feature selection method [27], it may not be optimal regarding its sub-features, which produce low accuracy rates. The FCBF does not have a stopping criterion and therefore, search termination occurs when the entire feature space has been explored [47].

In summary, the feature selection methods can improve the performance of the classification stage in pattern recognition and real-world datasets; however, the running time for selecting features remains a computational burden. Where the FCBF offers fast feature selection, albeit with lower accuracies, the proposed DPHOG selects the dominant patterns of HOG while retaining optimal speed in selecting HOG features and maintaining high accuracy.

III. THE PROPOSED DPHOG

A. DPHOG COMPUTATION

This paper proposes a new method to select the dominant patterns of HOG for vehicle detection, which optimally performs computations resulting in a faster feature selection process. The basic premise of the proposed method is that object images belonging to different classes should have different dominant pattern subsets. Accordingly, the non-dominant patterns must be removed from each image to identify object images belonging to different classes with less ambiguity. Therefore, we took out the intersection set of non-dominant pattern sets across all training images belonging to each class, resulting in our proposed method, called the dominant patterns of HOG (DPHOG). DPHOG excludes the non-dominant patterns that are common features that appear in both vehicle and non-vehicle classes. The remaining features are the dominant features representing the vehicle or non-vehicle classes. Figure 1 depicts the diagram of the three computing steps of the DPHOG, and are described as follows.

1) TRAINING SET PREPARATION

each image contains either a vehicle or non-vehicle object (Fig. 1 (a)) produced by the HG step. HOG extracts features of the object image, where FV and FN represent the matrices of features belonging to the vehicle and non-vehicle classes, respectively (Fig. 1(b)). Equation (1) shows the structure of the two matrices, where v and n are the total number of vehicle and non-vehicle samples (images), respectively; m is the dimension or length of the HOG feature vector, a_{11} is the HOG feature of the 1st vehicle or non-vehicle at the 1st dimension, a_{22} is the HOG feature of the 2nd vehicle or non-vehicle at the 2nd dimension, and a_{vm} and a_{nm} are the HOG features of the v^{th} vehicle and n^{th} non-vehicle, respectively, at the m^{th} dimension.

$$FV = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ a_{v1} & a_{v2} & a_{v3} & a_{vm} \end{bmatrix}_{v\times m}, \quad FN = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & a_{nm} \end{bmatrix}_{n\times m} \tag{1}$$

2) IDEAL VECTOR COMPUTATION

all objects can be represented by an ideal vector $\overline{FV}$ [48], computed by (2), and vector $\overline{FN}$, computed by (3); representing the vehicle and non-vehicle ideal vectors, respectively. The dimension of the ideal vector is $1\times m$.

$$\overline{FV} = \left[\frac{1}{v}\sum_{i=1}^{v} FV_{i,1}, \frac{1}{v}\sum_{i=1}^{v} FV_{i,2}, \ldots, \frac{1}{v}\sum_{i=1}^{v} FV_{i,m}\right]_{1\times m} \tag{2}$$

$$\overline{FN} = \left[\frac{1}{n}\sum_{i=1}^{n} FN_{i,1}, \frac{1}{n}\sum_{i=1}^{n} FN_{i,2}, \ldots, \frac{1}{n}\sum_{i=1}^{n} FN_{i,m}\right]_{1\times m} \tag{3}$$

3) DOMINANT PATTERNS DETECTION

the ideal vectors are segmented as l-consecutive dimensions, where the segmentation number of the ideal vector is expressed as $K = m/l$. The average of the differences of vehicles and non-vehicles of the k^{th} segment is denoted by $\overline{d}_k$ (4). The average of the differences between vehicles and non-vehicles of the ideal vector is indicated by D(5). The pattern having the average of the dissimilarities of $\overline{FV}$ and $\overline{FN}$ features in the segment greater than or equal to a predetermined threshold value is called the *dominant pattern*. The mathematical expression for this is $\overline{d}_k \leq (D+\alpha)$, where α is a shifting threshold value. The k^{th} segment is a dominant pattern if, and only if, the features contained in the segment form into a dominant pattern. Otherwise, the k^{th} segment is a non-dominant pattern. The index of dominant patterns is denoted as I(6). We varied the values of l and α to find the optimal l and α, where l was set to 5, 10, 15, 20, 25, and 30; and α was set to -0.006, -0.003, 0.0, and 0.001. From studies, these values have produced high accuracy rates.

$$\overline{d_k} = \frac{\sum_{i=(k-1)l+1}^{kl} \left|\overline{FV}_{(1,i)} - \overline{FN}_{(1,i)}\right|}{l} \tag{4}$$

$$D = \frac{\overline{d}_1 + \overline{d}_2 + \overline{d}_3 + \ldots + \overline{d}_K}{K}, \quad K = \frac{m}{l} \tag{5}$$

$$I = \left\{k | \overline{d_k} \geq (D+\alpha)\right\} \tag{6}$$

Figure 1(a) displays 4,136 vehicle and 5,797 non-vehicle images from the CompCars dataset. FV and FN are shown in Fig. 1(b). Figure 1(c) visualizes $\overline{FV}$ (dashed red line) and $\overline{FN}$ (blue line). Supposing the number of features is 20, and the segment length is five ($l = 5$), there are four segments ($K = 4$) generated from the feature vectors. Figure 1(d)

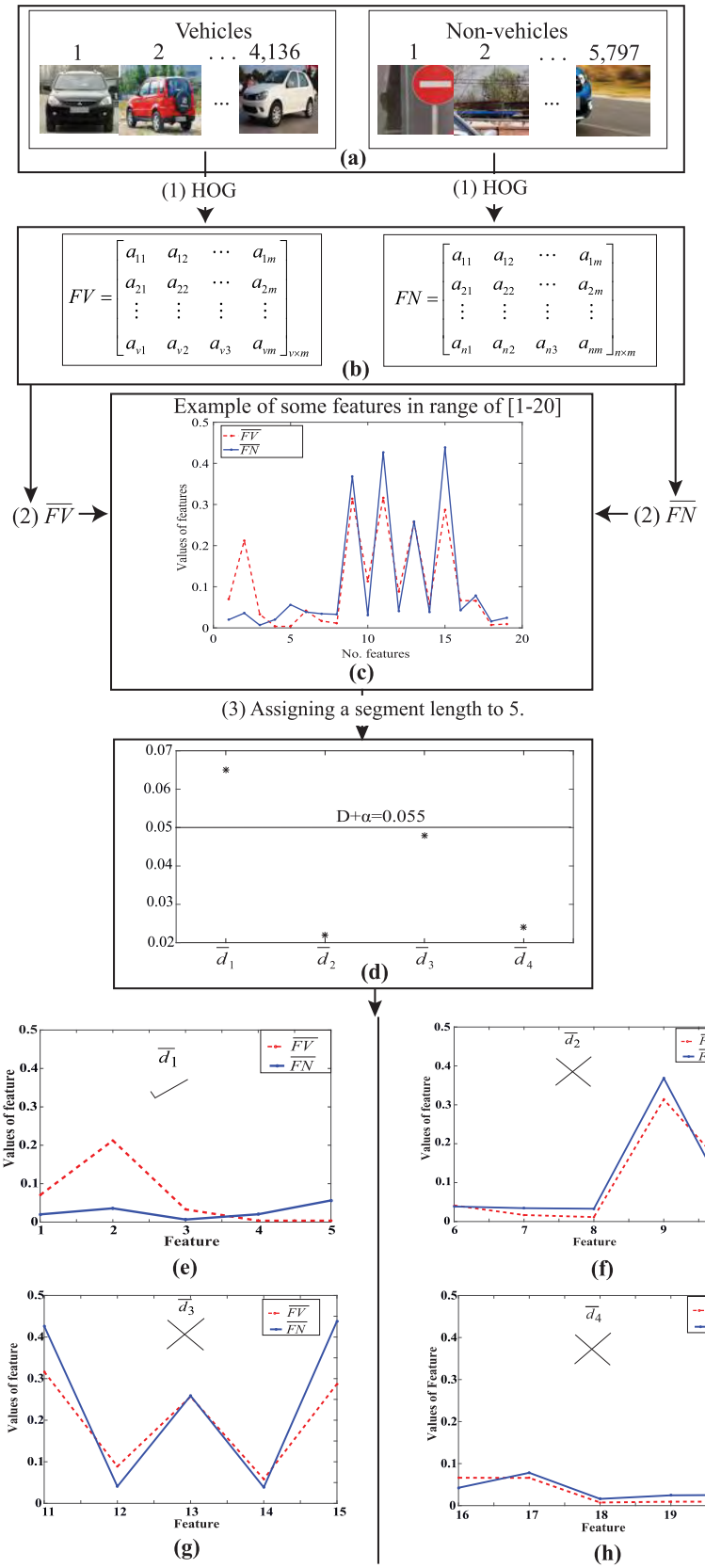


FIGURE 1. Demonstrative computation of DPHOG.

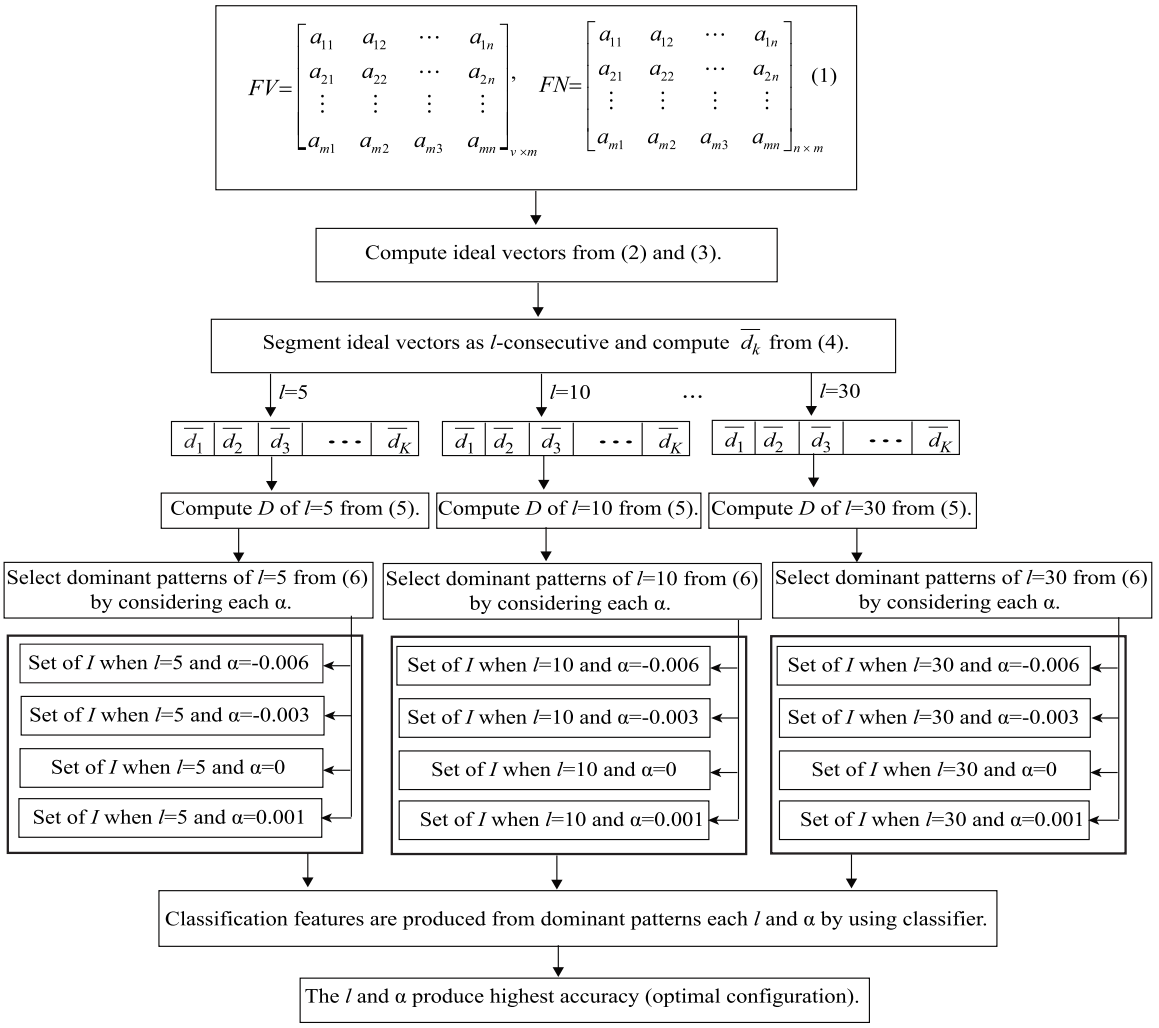


FIGURE 2. The application of DPHOG in vehicle detection.

contains a horizontal line, $D = 0.055 + 0.0$, i.e., $\alpha = 0.0$, and four points, which represent $\overline{d}_1, \overline{d}_2, \overline{d}_3$, and $\overline{d}_4$. Figures 1 (e), (f), (g), and (h) visualize four segments of the ideal vectors $\overline{FV}$ and $\overline{FN}$. Each segment has five values of $\overline{FV}$and five values of $\overline{FN}$, generated from the corresponding segments. Equation (6) determines the dominant pattern, $\overline{d}_1 \geq 0.055$,shown in Fig.1 (e). Whereas Figs.1 (f), (g), and (h) are non-dominant patterns; represented as $\overline{d}_2 < 0.055$, $\overline{d}_3 < 0.005$, and $\overline{d}_4 < 0.005$; respectively.

B. APPLYING DPHOG IN VEHICLE DETECTION

This section describes in detail the application of DPHOG in vehicle detection. DPHOG contains l and α parameters that affect both accuracy and time computation in vehicle detection in the classification stage, which are important to the safety of drivers and passengers in intelligent vehicle detection systems. However, applying DPHOG in vehicle detection requires the optimal configuration of both parameters.

Figure 2 presents the concept of DPHOG in finding the optimal configurations of l and α, and its application in vehicle detection. Vehicle and non-vehicle images are represented by HOG and are represented as FV and FN, respectively. The DPHOG selects the dominant patterns from HOG features through the use of ideal vectors of the vehicle and non-vehicle images. The ideal vectors are computed by (2) and (3) and are then segmented to K chunks, in which l is the number of the dimension of segmentation ($l = 5$, 10, 15, 20, 25, and 30). The dominant patterns of each l were computed as follows; first, the average of the differences of vehicles and non-vehicles of the k^{th} segment is computed by (4). Next, the averages of the differences of vehicles and non-vehicles of the ideal vector are computed by (5). The dominant patterns are then selected by considering each α, computed by (6). The α assigns values in the set of $\{-0.006, -0.003, 0, 0.001\}$, and each l parameter uses four α values in finding the optimal configuration. For example, $l = 5$ creates combinations with α; that are $l = 5$, with $\alpha = -0.006$; $l = 5$, with $\alpha = -0.003$; $l = 5$, with $\alpha = 0$; and $l = 5$, with $\alpha = 0.001$.

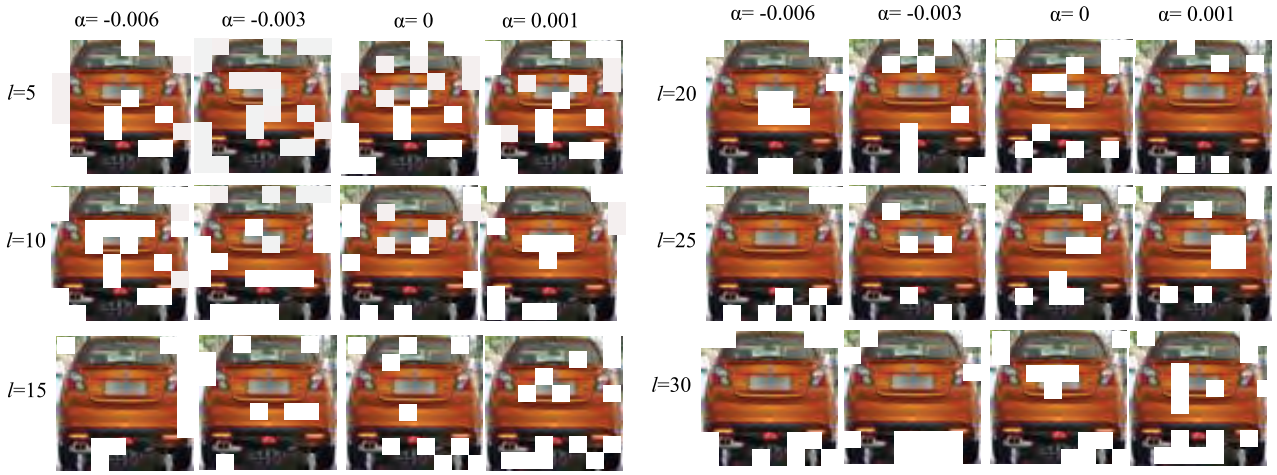


FIGURE 3. Result images after applying DPHOG in varied l and α values.

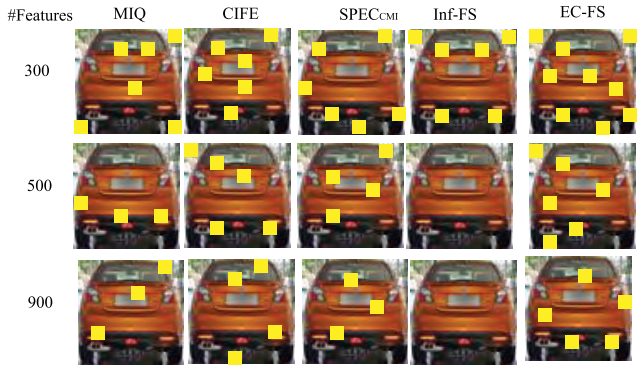


FIGURE 4. Result images after applying MIQ, CIFE, $SPEC_{CMI}$, Inf-FS, and EC-FS with the number of selected features of 300, 500 and 900.

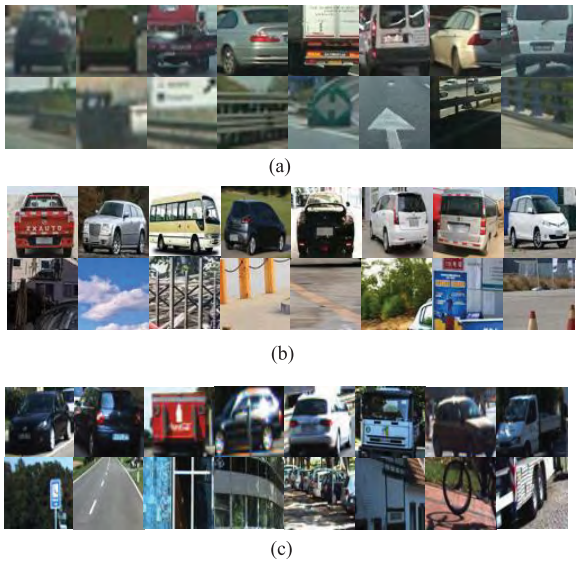


FIGURE 5. Three public datasets: (a) vehicles and non-vehicles of GTI dataset; (b) original vehicles and non-vehicles of CompCars captured manually from original images; and (c) vehicles and non-vehicles of the KITTI dataset, captured manually from original images.

Combinations are similarly repeated through $l = 10$, 15, 20, 25, and 30. The final step involves the selection of the dominant patterns by DPHOG using various combinations of l and α, which are then classified by the classifier. The l and α within the optimal configuration provided the highest of all accuracy rates.

Figure 3 presents result images after applying DPHOG with varied l and α values, given a cell size of 8 ($s = 8$). DPHOG ignores the white cells, and the remaining cells comprise more than 60% of the total number of features within a cell of HOG [40]. Within the varied l and α values, DPHOG selects and discards cells around the boundaries of vehicles that are similar to non-vehicles. For example, cells found within the margin of a vehicle image may belong to the image background and are therefore ignored by DPHOG. Subsequently, DPHOG produced the highest accuracies in all values of l and α.

Many feature selection methods have found optimal values similar to DPHOG; for example, MIQ, CIFE, $SPEC_{CMI}$, Inf-FS, and EC-FS. These methods rank the indices of the prominent features in descending order and then determine the optimal number of features. They then use classifiers in the performance evaluations of the number of features. Again, the highest accuracies were obtained from classifiers that produced the optimal number of features. We compare the performance of the DPHOG with these methods in the experiment results section. Figure 4 presents result images after applying MIQ, CIFE, $SPEC_{CMI}$, Inf-FS, and EC-FS with the number of selected features (#Features) of 300, 500 and 900. These methods select all cells of HOG and differentiate the cells that contain less than 40% of all the number of features within a cell of HOG (yellow cells) from those greater than 40% of the total number of features within a cell of HOG [40]. This process, however, may result in the selection of features that are similar to both vehicle and non-vehicle images. The DPHOG, on the other hand, discards cells of similarity between the vehicle and non-vehicle

images, resulting in greater performance accuracy than the MIQ, CIFE, $SPEC_{CMI}$, Inf-FS, and EC-FS feature selection methods.

IV. EXPERIMENTAL EVALUATION

A. DATASET

There are three customized datasets in the experiments used in this study: the (1) GTI [49], (2) CompCars [50], and (3) KITTI [23] datasets.

In the GTI dataset [49], images are collected from videos of road sequences. There are a total of 7,325 images; of which 3,425 are vehicle images, and 3,900 are non-vehicles images. The vehicle images captured in rear view were divided into four different regions, according to their positions i.e., far range, middle/close range in front of the camera, middle/close rage at left of the camera, and close/middle range at right of the camera (Fig. 5 (a)).

The original CompCars dataset [50] collected the images from surveillance-nature and web-nature scenarios. The surveillance-nature images have background characteristics similar to that of the GIT dataset. We therefore selected the web-nature scenarios as our main dataset because they offer several views of vehicles at different speeds and degrees of background complexity. A total of 136,727 images of cars covering different models from the past ten years were collected. The steps for the building of the training and test images from this dataset are as follows; first, the images were selected; second, the cars and backgrounds were manually clipped; and last, the clipped images were resized to 64 $\times$ 64 pixels. The resulting dataset consisted of 11,593 images of cars and 8,272 background images (Fig. 5 (b)).

The original KITTI dataset [23] contained vehicle images with various backgrounds, which included road and traffic signs, buildings, the sky, etc. The customized training and testing images (7,481 and 7,518, respectively) were prepared in the same manner as the CompCars dataset. Our completed KITTI dataset ultimately consisted of 1,160 vehicle images and 3,468 non-vehicle images; some of which are shown in Fig. 5 (c).

We assigned the image size (64 $\times$ 64 pixels) to the images of all three datasets to facilitate their extraction. The characteristics of each dataset, CompCars, GTI, and KITTI, present uniquely different characteristics. First, the GTI captures only the rear view of the vehicles with different angles, while the CompCars and KITTI datasets consist of the side view, rear view, and front view of the vehicles at various angles. Second, the non-vehicle classes of the CompCars and KITTI datasets contain a variety of object types, such as roads, bridges, traffic signs, signboards, trees, buildings, etc. The non-vehicle class of the GTI dataset consists only of the objects relating to the roads, for example, street lines, traffic signs, and road barriers, etc. Finally, the image qualities of the GTI and KITTI images are low, whereas the image quality of the CompCars dataset is significantly higher.

B. PERFORMANCE CRITERIA

There were four performance evaluation matrices used to evaluate the performance of the proposed method. The equations (7), (8), (9), and (10) compute the accuracy, true positive rate (TPR), false positive rate (FPR), and F1-score, respectively.

In particular, the F1-score is a major criterion in considering the performance comparisons when TPR and FPR share similar values [51].

$$accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{7}$$

$$TPR = \frac{TP}{TP + FN} \tag{8}$$

$$FPR = \frac{FP}{FP + TN} \tag{9}$$

$$F1 - score = \frac{TP}{TP + (\frac{FP+FN}{2})} \tag{10}$$

where TP is the number of vehicles correctly classified in the vehicle samples, TN is the number of non-vehicles correctly classified in the non-vehicle samples, FP is the number of vehicles classified in the non-vehicle samples, and FN is the number of non-vehicles classified in the vehicle samples.

C. EXPERIMENT SETUP

This section explains the experiment settings and performance comparisons. The DPHOG was compared with the HOG, V-HOG, and πHOG; as well as several well-known feature selection methods, including the WOA-SA [32], ANFC-LH [31], FCBF [27], MIQ [28], CIFE [25], $SPEC_{CMI}$[29], In-FS [26], and EC-FS [30].

Two parameter settings were made for the HOG, V-HOG, and πHOG; (1) the size of cells (s), which was set at 4, 8, and 16; and (2) the number of the orientation of bins (β), which was set to 9. Both the V-HOG and πHOG set the values of s and β according to HOG, which varied greatly. The DPHOG contains two parameters of HOG, as well as two additional parameters, $l \in \{5, 10, 15, 20, 25, 30\}$, and $\alpha \in \{-0.006, -0.003, 0.0, 0.001\}$.

Within the feature selection methods, the ANFC-LH, WOA-SA, and FCBF can directly determine the proper number of prominent features, whereas the MIQ, CIFE, $SPEC_{CMI}$, Inf-FS, and EC-FS rank prominent indices of each feature. This is followed by the determination of a proper number of selected features. Consequently, we varied the number of features of the MIQ, CIFE, $SPEC_{CMI}$, Inf-FS, and EC-FS; as illustrated in Table 1. The parameter settings of the remaining feature selection methods were set based on their original works.

The efficiency of the DPHOG, HOG, V-HOG, πHOG, and the various feature selection methods were evaluated using two classifiers, namely, the Kernel Extreme Learning Machine (KELM) and Support Vector Machine (SVM). The KELM with an RBF kernel has been reported to achieve greater accuracy than the basic ELM [52]. More importantly, the number of hidden nodes does not need to be specified to

TABLE 1. The Number of features depending on the cell size of the MIQ, CIFE, $SPEC_{CMI}$, Inf-FS, and EC-FS; the maximum number of the feature of DPHOG; and the number of features of HOG.

s	The number of features depending on s of MIQ, CIFE, $SPEC_{CMI}$, Inf-FS, and EC-FS	The maximum number of features of DPHOG	The number of features of HOG
4	2,000, 3,000, ,4000, 5,000 and 6,000	5,790	8,100
8	300, 500, 900, 1200 and 1500	1,215	1,764
16	50, 100, 200 and 250	240	324

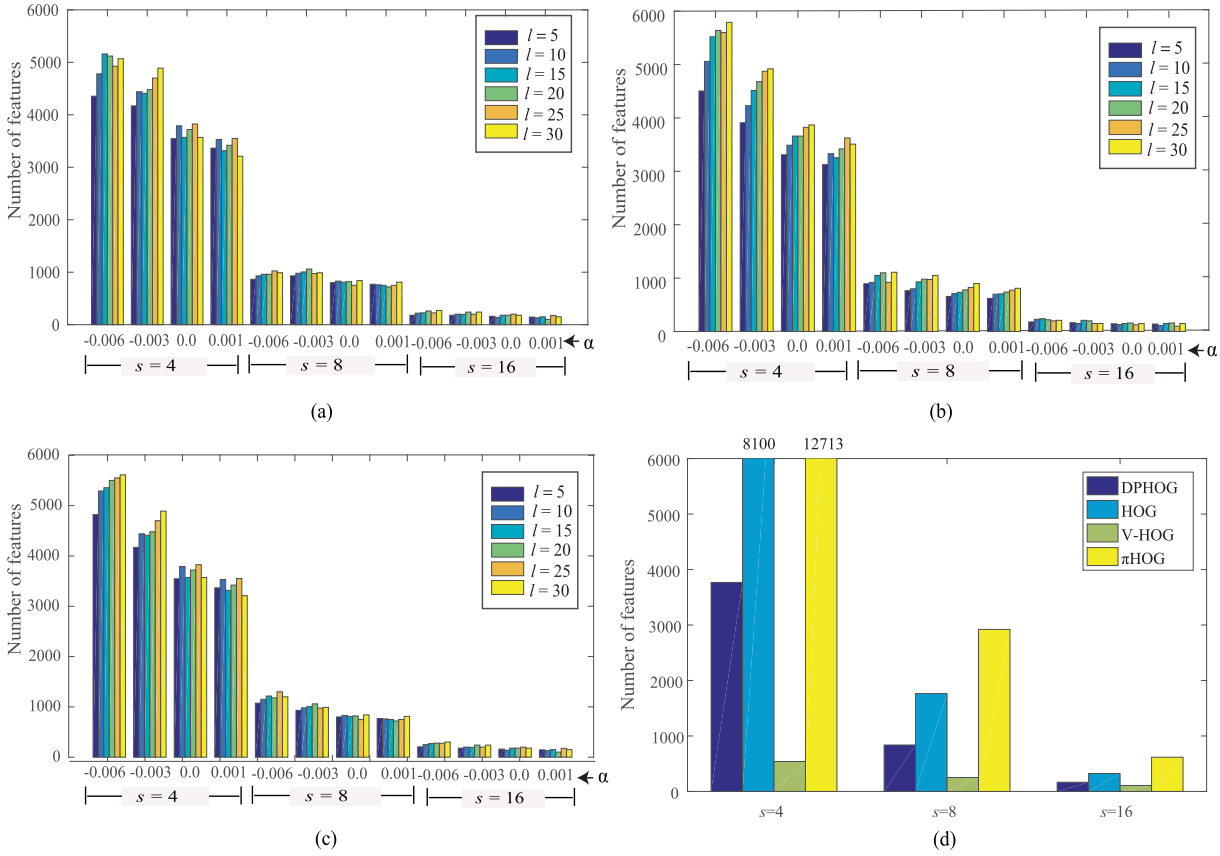


FIGURE 6. The number of features of DPHOG (a) number of features of CompCars, (b) number of features of GTI, (c) number of features of KITTI, and (d) comparison average of the number of features of DPHOG with HOG, V-HOG, and πHOG.

carry out the experiments. The SVM with a linear kernel was chosen, as it produces high-performance results in vehicle detection [20], [21].

In each dataset, 30 replications of training and testing sets with a ratio of 0.5:0.5 were randomly constructed. Each pair of the training-testing sets had no common data. The experimental environment was a PC with an Intel (R) Core(TM) i7 3.20 GHz processor and 64-bit Windows 7 operating systems. Each algorithm was coded in MATLAB R2012a.

V. EXPERIMENT RESULTS

A. EXPERIMENT RESULTS OF THE DPHOG

This section presents the experimental results of the proposed technique and the comparison results with several benchmark methods, in which we demonstrate the significance of the length of generated features. We also present the accuracies of DPHOG with KELM and SVM by varying s, l, and α. We determined the optimal configuration based on each dataset, and the optimal common configuration, obtained by the best accuracy of all datasets (CompCars, GTI, and KITTI) of the DPHOG. The comparative results of the DPHOG and the HOG, V-HOG, πHOG, and the selected feature selection methods will be presented in the next section.

In general, the number of features produced by the DPHOG depends upon s, l, α, and the dataset, shown in Figs. 6(a), (b), and (c). We noticed that a DPHOG with a small value of s generated longer feature vectors than a DPHOG with a larger value of s. This phenomenon is also confirmed with the value α. The maximum number of the extracted features from Figs. 6(a), (b), and (c), where $s$equaled 4, 8, and 16, were 5,790, 1,215, and 240, respectively. These numbers then guide the settings of the numbers

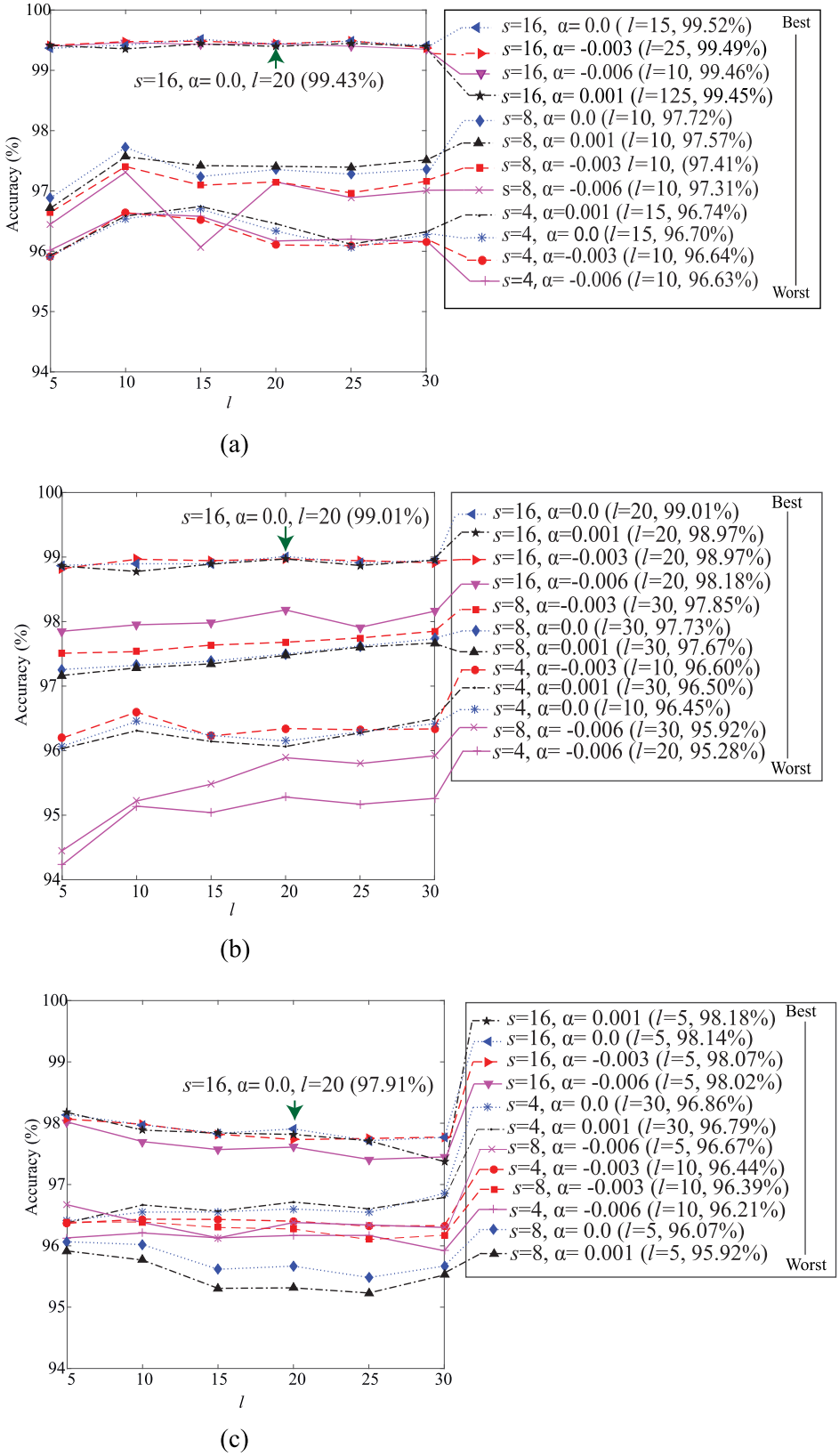


FIGURE 7. The accuracy of DPHOG with various *s*, *l*, and α values using KELM as a classifier; (a) CompCars, (b) GTI and, (c) KITTI datasets.

of extracted features for the CIFE, Inf-FS, $SPEC_{CMI}$, MIQ, and EC-FS feature selection algorithms, as shown in Table 1.

The number of features resulting from DPHOG with a set s value depends on the l and α parameters and the dataset. However, the numbers of features extracted through HOG, V-HOG, and πHOG were not affected by the s value of the dataset. Therefore, the number of features created by the DPHOG were determined by the average number of features of every l and α in each dataset. This average number of DPHOG features was compared with the number of features produced by HOG, V-HOG, and πHOG; shown in Fig. 6(d). From Fig. 6(d), the DPHOG produced a lower number of features than that of the πHOG and HOG. The V-HOG also generated the shortest feature vectors.

To find the optimal configuration of DPHOG in each dataset, the ranking of each DPHOG configuration was performed as follows: for a given dataset, there were 12 combinations of s and α values. We then found the 12 highest accuracies of the classification performed on DPHOG, with pre-defined segment lengths from {5, 10, 15, 20, 25, 30}. All 12 accuracy values were sorted in ascending order, and the rank of each combination of s and α values corresponded to the indices of the sorted accuracies (Figs. 7 and 8).

Each sub-figure in Figs. 7 and 8 show the accuracy and ranking of DPHOG with 72 combinations of s, l, and α values; where the x-axis is length of segment, $l \in \{5, 10, 15, 20, 25, 30\}$ and the y-axis is the accuracy produced from either the KELM or SVM classifier classifying the DPHOG features.

The results obtained from various feature extraction techniques using the KELM classifier on CompCars (Fig. 7(a)) were interpreted as follows; DPHOG with $s = 16$ and $\alpha = 0$ ranks first, as it gave the highest accuracy of 99.52% when l was 15; DPHOG with $s = 8$ and $\alpha = 0.0$ ranked fifth, as it produced the highest accuracy of 97.72% when l was 10; DPHOG with $s = 4$ and $\alpha = -0.006$ ranked last, as it gave the highest accuracy of 96.63% when l was 10. Therefore, in the KELM classifier, DPHOG with $s = 16$, $\alpha = 0$, and $l = 15$ is the optimal configuration, and DPHOG with $s = 4$, $\alpha = -0.006$, and $l = 10$ is the worst configuration. A similar interpretation of the results was found using KELM on GIT and KITTI (Figs. 7(b) and (c)), as well as the results obtained from the SVM on CompCars, GIT, and KITTI; Figs. 8(a), (b), and (c), respectively.

The optimal configuration is important for developing a system in only one environment, for example, a motorway, in urban traffic, or a local road. The highest accuracy within each dataset determines the optimal configuration. We found that each dataset had different optimal configuration values. However, some systems for vehicle detection are developed to work in more than one environment; for example, classifying vehicles and non-vehicles in motorways, urban traffic, and local roads. Therefore, we attempted to find the optimal common configuration using the same configuration (l and α values) that would produce the highest accuracy developed in several environments, or datasets. The following procedures can find the optimal common configuration: the accuracies produced by DPHOG with 72 combinations of s, l, and α values are ranked by competition ranking [53] for each dataset, which gives us the ranks for each configuration. The average ranking of all datasets is then computed and sorted in value from small to high. Table 2 shows the top 12 ranks of the DPHOG configurations, in which KELM paired with DPHOG ($s = 16$, $l = 20$ and $\alpha = 0.0$); and SVM paired with DPHOG ($s = 4$, $l = 10$ and $\alpha = -0.006$) produced the optimal results and the top ranks. The common optimal configurations are represented by green arrows (Figs. 7 and 8).

Summarizing Figs. 7 and 8, the DPHOG with $s = 16$, $\alpha = 0.0$, and $l = 15$ using KELM provided the optimal configuration in the CompCars dataset; the DPHOG with $s = 16$, $\alpha = 0.0$, and $l = 20$ using KELM was the optimal configuration in the GTI dataset; and the DPHOG with $s = 4$, $\alpha = -0.006$, and $l = 15$ using SVM was the optimal configuration in the KITTI dataset.

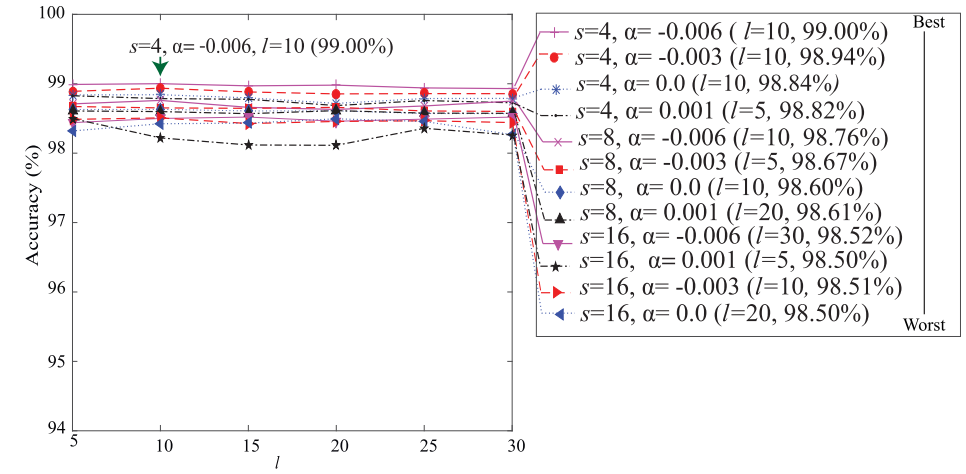
Two common optimal configurations were found; namely, the DPHOG with $s = 16$, $\alpha = 0.0$, and $l = 20$ with the KELM classifier, and the DPHOG with $s = 4$, $\alpha = -0.006$, and $l = 10$ with the SVM.

The DPHOG with $s = 16$ produced the highest accuracy when using the KELM classifier in every dataset. Also, the DPHOG with $s = 16$ paired with KELM, where l and α were free parameters, resulted in the same marginal performance in every dataset.

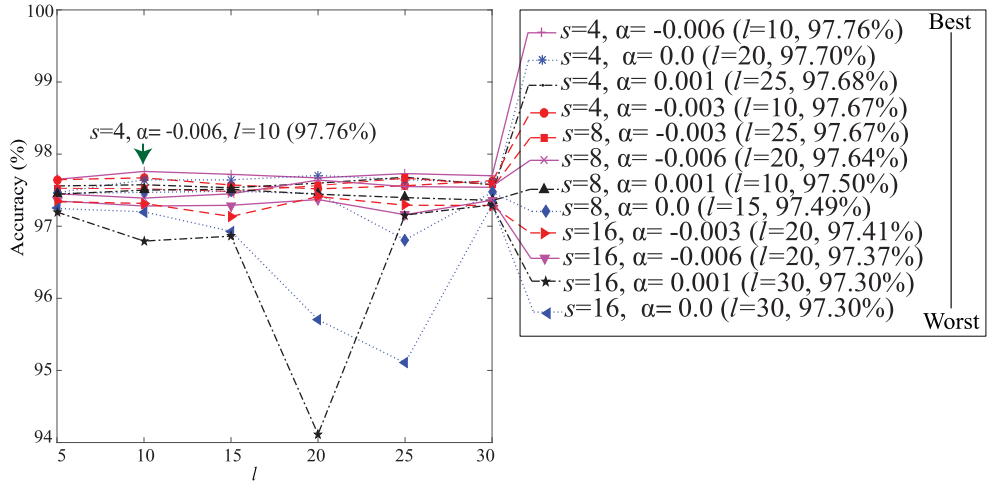
Finally, the DPHOG with $s = 4$ produced high accuracies with the SVM classifier. The l and α values were non-sensitive parameters that produced the same marginal results in the CompCars and GTI datasets. The DPHOG with $s = 4$ paired with SVM were shown to be sensitive to the l and α values in the KITTI dataset. However, using $\alpha = -0.006$, $l = \{5, 10, 15, 20, \text{and } 25\}$ in the DPHOG with SVM produced high accuracies in the KITTI dataset.

B. EXPERIMENT RESULTS OF HOG, V-HOG, πHOG, AND THE WELL-KNOWN FEATURE SELECTION METHODS

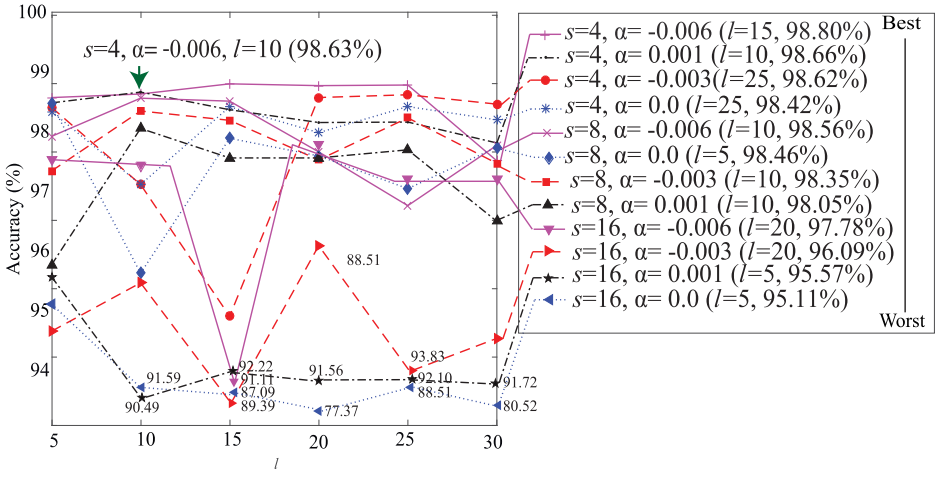
This section compares the accuracies of three feature extraction methods (HOG, V-HOG, πHOG) and eight selected well-known feature selection methods (ANFC-LH, WOA-SA, FCBF, MIQ, CIFE, $SPEC_{CMI}$, Inf-FS, and EC-FS). The highest accuracies of these methods were compared with those of the proposed DPHOG. Table 3 shows the accuracies of feature extraction methods classified by KELM and SVM on the CompCars, GTI, and KITTI datasets. V-HOG yielded the highest accuracy ($s = 8$) with the KELM classifier in all three datasets. HOG and πHOG ($s = 16$) classified by KELM produced the highest accuracies in both the CompCars and GTI datasets. In the KITTI dataset, consisting of low quality, multi-view images of the vehicle; HOG provided the highest accuracy when using $s = 4$, and classified with the SVM. πHOG gave the highest accuracy ($s = 8$) with the SVM. The highest accuracy of each method and each dataset, represented in bold font, is presented in Table 3.



(a)



(b)



(c)

FIGURE 8. The accuracy of DPHOG with various *s*, *l*, and α values using SVM as a classifier; (a) CompCars, (b) GTI and, (c) KITTI datasets.

TABLE 2. The computation ranks of DPHOG based on the top 12 l and α combination values.

s=16		KELM				s=4		SVM			
l	α	CompCars	GTI	KITTI	Average	l	α	CompCars	GTI	KITTI	Average
20	**0.000**	**12**	**1**	**7**	**6.67**	**10**	**-0.006**	**1**	**1**	**5**	**2.33**
15	0.000	1	12	10	7.67	**10**	**-0.006**	**1**	**1**	**5**	**2.33**
25	-0.003	2	8	15	8.33	15	-0.006	4	3	1	2.67
20	-0.003	10	3	16	9.67	25	-0.003	10	24	6	13.33
5	0.001	14	16	1	10.33	10	0.001	19	21	4	14.67
20	0.001	22	2	12	12.00	25	0.000	18	12	14	14.67
25	0.001	7	15	18	13.33	25	0.001	22	6	23	17.00
5	0.000	22	18	2	14.00	20	0.000	25	4	26	18.33
5	-0.003	16	17	13	15.33	10	-0.003	6	7	47	20.00
10	-0.006	6	22	19	15.67	10	-0.003	6	7	47	20.00
20	-0.006	10	19	20	16.33	10	0.000	13	14	47	24.67
5	-0.006	20	25	4	16.33	5	0.001	41	44	52	45.67

TABLE 3. Accuracies of the HOG, V-HOG and πHOG feature extraction methods using KELM and SVM as classifiers when the s of HOG is varied.

s	CompCars classified by KELM			CompCars classified by SVM		
	HOG	V-HOG	πHOG	HOG	V-HOG	πHOG
4	95.42	98.14	96.65	98.95	93.51	98.97
8	96.92	**98.80**	93.37	98.72	86.91	98.83
16	**99.36**	97.89	**99.27**	97.76	82.30	98.50
s	GTI classified by KELM			GTI classified by SVM		
	HOG	V-HOG	πHOG	HOG	V-HOG	πHOG
4	96.10	97.76	96.81	97.89	96.97	97.28
8	97.93	**98.27**	97.69	97.85	96.91	98.24
16	**98.75**	97.79	**98.31**	97.80	96.75	97.72
s	KITTI classified by KELM			KITTI classified by SVM		
	HOG	V-HOG	πHOG	HOG	V-HOG	πHOG
4	97.73	96.01	96.95	**98.05**	90.16	96.91
8	96.53	**97.48**	93.56	97.95	93.74	**98.20**
16	97.82	96.99	97.54	96.42	89.71	97.75

Within the performance accuracies of the HOG, V-HOG, and πHOG in each dataset, the highest accuracy of V-HOG is lower than the highest HOG in each dataset. The explanation for this is that the V-HOG uses only information in a vertical direction, which means that it is incapable of describing the difference between vehicles and non-vehicles in some situations (multi-view and multi-angles of vehicles, and low-quality images). The πHOG out produced HOG in the KITTI dataset, yet fell behind the HOG in terms of accuracy in both the CompCars and GTI datasets. The πHOG embedded the positions of orientation bins and intensity features into the HOG features, which increased the accuracies of HOG in low-quality images, multi-views of a vehicle, multi-angles of vehicles, and background complexities in the KITTI dataset. HOG represented vehicles and non-vehicles better than the πHOG, in which the vehicle and non-vehicle images were clearer in the CompCars and GTI datasets than those of the KITTI dataset. Therefore, we can conclude that adding the positions of orientation of bins and intensity features in πHOG will not properly represent vehicles and non-vehicles in some datasets.

Table 4 presents the accuracies of the WOA-SA, ANFC-LH, and FCBF feature selection methods, in which the highest accuracy for each method and each dataset is represented in bold font. These methods directly determined the prominent feature indices of the HOG features. The process of HOG with $s = 16$ selected by ANFC-LH and classified with KELM gave the highest accuracy in every dataset. FCBF provided the highest accuracy for each dataset when HOG ($s = 8$) was classified with the KELM. The highest accuracy result, within the WOA-SA feature selection method, was produced by HOG ($s = 16$) and the KELM classifier in the CompCars and GTI datasets. The WOA-SA produced the highest accuracy in the KITTI dataset ($s = 4$) with the SVM classifier. Features selected by ANFC-LH and FCBF retained less than 20% of their HOG features and were not sufficient enough for describing the differences between vehicles and non-vehicles, producing low accuracies. The WOA-SA retained more than 49% of its HOG features, containing more prominent features than both the ANFC-LH and FCBF, which resulted in it presenting a higher accuracy in every dataset.

Figures 9, 10, and 11 show the accuracies of the chosen feature selection methods, MIQ, CIFE, SPEC_{CMI}, Inf-FS, and EC-FS using the KELM or SVM classifiers in the CompCars, GTI, and KITTI datasets, respectively. The number of features used in each of these methods is dependent on s, as seen in Table 1. The objective of assigning various feature numbers is to determine the optimal number of features required by each method, in each dataset.

For each sub-figure of Figs. 9, 10 and 11, the x-axis is the number of features, and the y-axis is the accuracy produced from the KELM or SVM classifiers following the MIQ, CIFE, SPEC_{CMI}, Inf-FS, and EC-FS feature selection methods. Figures 9(a), (b) and (c) depict the accuracies of the MIQ, CIFE, SPEC_{CMI}, Inf-FS, and EC-FS in the CompCars dataset, with $s = 4$, $s = 8$, and $s = 16$, respectively. The optimal number of features is presented in bold font (Fig. 9(c)). The optimal number of features within the CompCars dataset, given the highest accuracies of the MIQ, CIFE, Inf-FS, and EC-FS using the KELM classifier ($s = 16$); was determined to be 250 (Figs. 9(a), (b), and (c)). The highest accuracy of SPEC_{CMI} with KELM ($s = 16$) produced the optimal number of features of 200.

Figures 10(a), (b), and (c) depict the accuracies of the MIQ, CIFE, SPEC_{CMI}, Inf-FS and EC-FS when HOG uses $s = 4$, $s = 8$, and $s = 16$ in the GTI dataset, respectively. Again, the optimal number of features is represented in bold

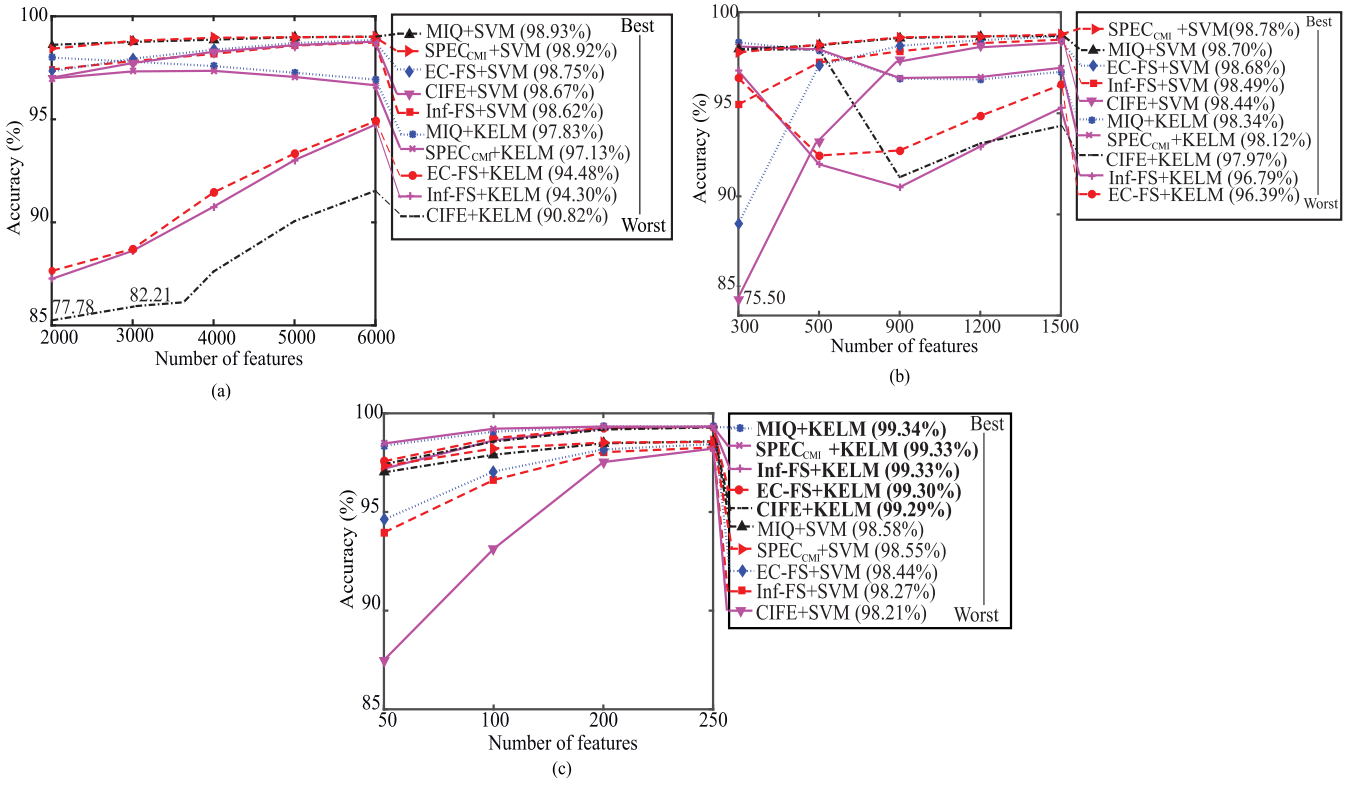


FIGURE 9. Accuracies of MIQ, CIFE, SPEC$_{CMI}$, Inf-FS, and EC-FS in the CompCars dataset, where (a) $s = 4$, (b) $s = 8$, and (c) $s = 16$.

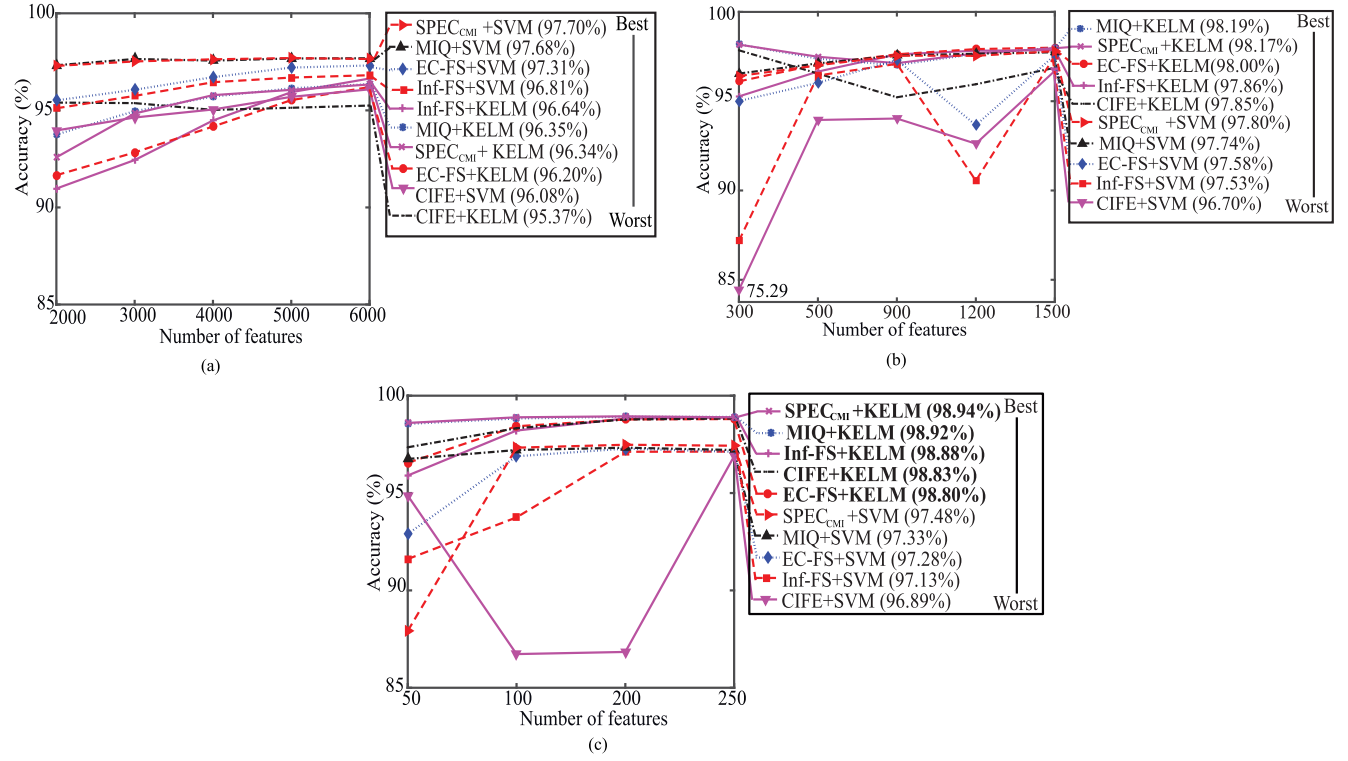


FIGURE 10. Accuracies of MIQ, CIFE, SPEC$_{CMI}$, Inf-FS, and EC-FS in the GTI dataset, where (a) $s = 4$, (b) $s = 8$, and (c) $s = 16$.

font (Fig. 10 (c)). MIQ, CIFE, SPEC$_{CMI}$, Inf-FS, and EC-FS with $s = 16$, classified with KELM produced the highest accuracies when the optimal number of features were 200, 250, 100, 250, and 200, respectively.

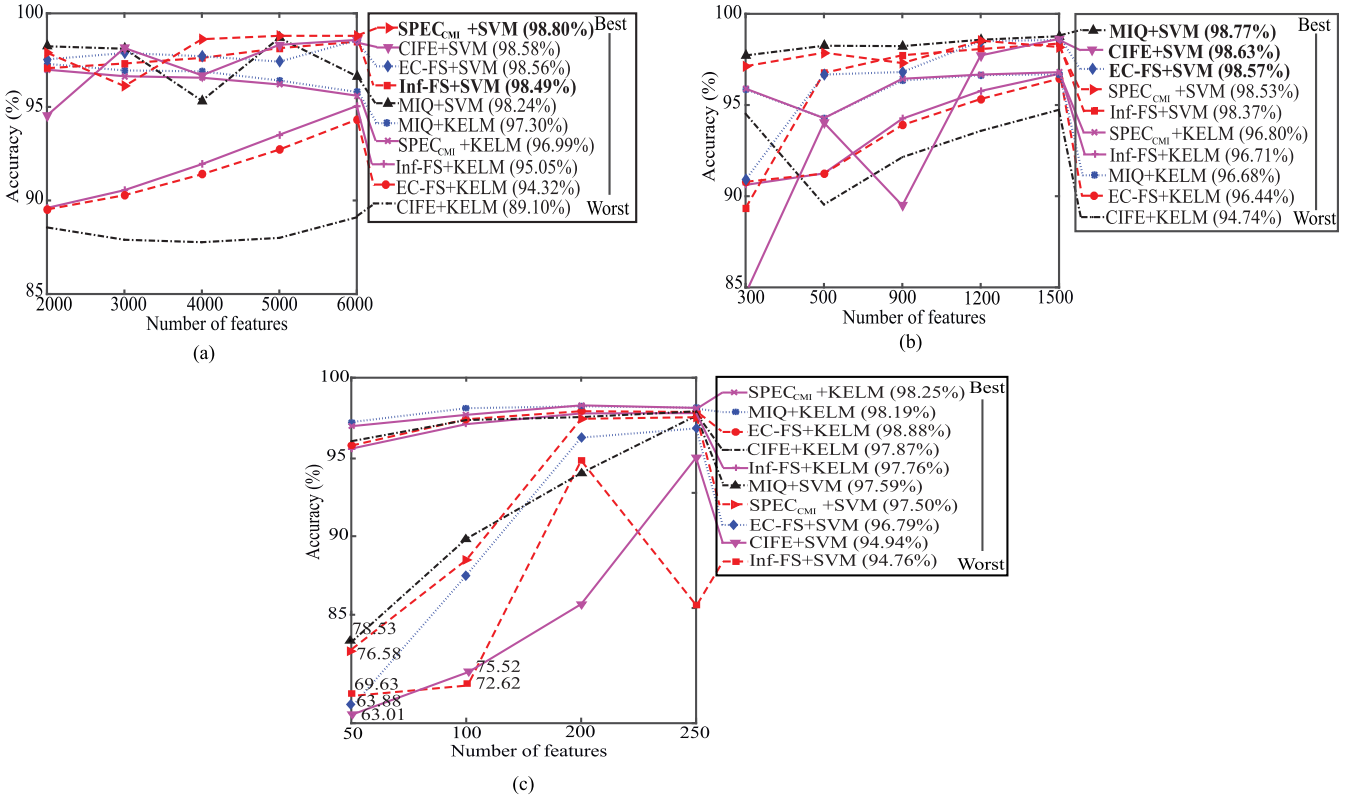


FIGURE 11. Accuracies of MIQ, CIFE, SPEC$_{CMI}$, Inf-FS, and EC-FS in the KITTI dataset, where (a) $s = 4$, (b) $s = 8$, and (c) $s = 16$.

TABLE 4. Accuracies of the WOA-SA, ANFC-LH AND FCBF feature selection methods using KELM and SVM as classifiers when s of HOG is varied.

s	CompCars using KELM			CompCars using SVM			The number of features of CompCars		
	WOA-SA	ANFC-LH	FCBF	WOA-SA	ANFC-LH	FCBF	WOA-SA	ANFC-LH	FCBF
4	95.62	95.84	95.97	98.83	90.15	95.51	3998	231	166
8	96.50	98.59	**98.62**	98.60	97.95	96.39	887	203	**45**
16	**99.46**	**99.11**	98.53	98.39	98.28	94.72	**173**	**90**	20
S	GTI using KELM			GTI using SVM			The number of features of GTI		
	WOA-SA	ANFC-LH	FCBF	WOA-SA	ANFC-LH	FCBF	WOA-SA	ANFC-LH	FCBF
4	96.16	96.12	97.77	97.52	95.10	95.10	4080	311	224
8	97.89	98.21	**98.26**	97.18	96.75	93.59	1058	188	**90**
16	**98.97**	**98.72**	96.92	97.35	96.84	94.40	**155**	**70**	31
S	KITTI using KELM			KITTI using SVM			The number of features of KITTI		
	WOA-SA	ANFC-LH	FCBF	WOA-SA	ANFC-LH	FCBF	WOA-SA	ANFC-LH	FCBF
4	94.76	95.36	91.71	**98.78**	81.00	84.23	**4512**	154	226
8	94.96	97.80	**97.78**	97.46	94.69	87.02	853	166	**71**
16	98.35	**98.13**	94.71	94.09	81.10	91.98	171	**95**	21

Figures 11 (a), (b), and (c) depict accuracies of the MIQ, CIFE, SPEC$_{CMI}$, Inf-FS, and EC-FS feature selection methods, with $s = 4$, $s = 8$, and $s = 16$ in the KITTI dataset, respectively. These methods produced the highest accuracies when using SVM as a classifier, but with different s values. SPEC$_{CMI}$and Inf-FS with $s = 4$ had the optimal number of features at 6,000. MIQ, CIFE, and EC-FS with $s = 8$ had the optimal number of features at 1,500.

In summary, within the CompCars dataset, the MIQ, CIFE, Inf-FS, and EC-FS had the optimal number of features of 250; whereas the SPEC$_{CMI}$ had the optimal number of features of 200. These methods can reduce the number of HOG features by about 30%, In the GTI dataset, MIQ, CIFE, SPEC$_{CMI}$, Inf-FS, and EC-FS produced different optimal numbers of features. The SPEC$_{CMI}$ method produced the highest accuracy of those compared and reduced the number

TABLE 5. comparison of the best accuracy of DPHOG with best accuracies of the group of feature extraction methods and the selected feature selection methods in the CompCars dataset.

#	Method	Acc. (%)	h	TRP	FPR	F1-score	TS. (second)	RT. (second)	Classifier	#Feature	s
1	DPHOG (α=0, l=15)	**99.52**	-	**0.9938**	**0.0039**	**0.9942**	0.027	**0.0235**	KELM	165	16
2	WOA-SA	99.46	1	0.9928	0.0042	0.9935	0.391	82,727	KELM	173	16
3	HOG	99.36	1	0.9972	0.0044	0.9923	0.053	-	KELM	324	16
4	MIQ	99.34	1	0.9905	0.0047	0.9921	0.036	6.393	KELM	250	16
5	Inf-FS	99.33	1	0.9901	0.0044	0.9920	0.034	6.356	KELM	250	16
6	$SPEC_{CMI}$	99.33	1	0.9907	0.0049	0.9919	0.031	6.593	KELM	200	16
7	EC-FS	99.30	1	0.9898	0.0048	0.9916	0.038	1.378	KELM	250	16
8	CIFE	99.29	1	0.9910	0.0058	0.9914	0.042	6.370	KELM	250	16
9	πHOG	99.27	1	0.9889	0.0045	0.9913	0.103	-	KELM	616	16
10	ANFC-LH	99.11	1	0.9878	0.0065	0.9893	0.018	1496.40	KELM	90	16
11	V-HOG	98.80	1	0.9840	0.0091	0.9859	0.044	-	KELM	252	8
12	FCBF	98.62	1	0.9755	0.0165	0.9738	**0.008**	1.234	KELM	45	8

of features of HOG by about 70%. The optimal numbers of features of the MIQ, CIFE, $SPEC_{CMI}$, Inf-FS, and EC-FS in the KITTI dataset were greater than those produced in both the CompCars and GTI datasets. This was due to the low-quality images, multi-views of vehicles, multi-angles of vehicles, and the complexity of non-vehicles within the KITTI dataset. However, the MIQ, CIFE, and EC-FS reduced the number of features of HOG by about 15% ($s = 8$). By comparison, the $SPEC_{CMI}$ and Inf-FS reduced the number of features of HOG by about 25% ($s = 4$).

C. COMPARISON PERFORMANCE OF THE DPHOG WITH HOG, V-HOG, πHOG, AND THE SELECTED WELL-KNOWN FEATURE SELECTION METHODS

This section summarizes the highest accuracies of the DPHOG, HOG, V-HOG, πHOG, and the selected well-known feature selection methods obtained through our experiments. The performance evaluation metrics of the techniques consisted of accuracy (Acc.), significance of the t-test (h), true positive rate (TPR), false positive rate (FPR), F1-score, testing time in the classification stage (TS), running time for selecting features (RT), and number of features (#Feature).

The highest accuracies attained in each method are presented in Tables 5, 6, and 7, as well as the optimal configurations of DPHOG in the CompCars, GTI, and KITTI datasets.

Analysis of the evaluations from the three comparative tables can be summarized as follows:

(1) DPHOG produced the highest accuracy and overall performance, which is statistically significant ($h = 1$), the highest TPR, the lowest FPR, and the highest F1-score in the CompCars and GTI datasets.

(2) DPHOG and $SPEC_{CMI}$ produced marginally high performances in accuracy in TPR, FPR, and F1-scores in the GTI and KITTI datasets. However, the DPHOG proved faster than $SPEC_{CMI}$ in both testing time and running time in selecting features.

(3) DPHOG demonstrated its flexibility with high performance (accuracy, TPR, FPR, and F1-scores) in all datasets, consisting of high and low quality images, multi-views of vehicles, multi-angles of vehicles, and complexity of non-vehicles images (due to the non-dominant patterns of HOG between vehicles and non-vehicles having been discarded by DPHOG).

(4) DPHOG reduced the testing time of HOG in each dataset, while increasing the accuracy of HOG.

(5) FCBF resulted in the fastest times in the testing stages, and obtained short running times but at the expense of lower accuracy than the DPHOG.

(6) Both the WOA-SA and DPHOG resulted in same similar TPR, FPR, and testing times in the CompCars, GTI, and KITTI datasets. However, DPHOG was faster in the feature selection process and produced greater accuracies and F1-scores in all three datasets.

(7) Inf-FS and CIFE, the feature selection methods used in pattern recognition increased HOG accuracies in the GTI and KITTI datasets, but not in the CompCars dataset. DPHOG, however, produced higher HOG accuracies in all datasets.

(8) πHOG, which embeds the positions of orientation bins and intensity features of the differences between vehicles and non-vehicles, increased the accuracies of HOG and the F1-score of the KITTI dataset. It also obtained the highest FPR in all datasets. However, πHOG resulted in lower accuracy, TPR, F1-score, and slower testing times when compared to the DPHOG.

(9) V-HOG proved faster than HOG but was unable to increase the accuracy of HOG. Comparatively, the DPHOG produced higher accuracy, TPR, and F1-score than V-HOG in

TABLE 6. comparison of the best accuracy of DPHOG with best accuracies of the group of feature extraction methods and the selected feature selection methods in the GTI dataset.

#	Method	Acc. (%)	h	TPR	FPR	F1-score	TS. (second)	RT. (second)	Classifier	#Feature	s
1	DPHOG (α=0, l=20)	**99.01**	-	**0.9861**	**0.0052**	**0.9890**	0.0091	**0.075**	KELM	160	16
2	WOA-SA	98.97	1	0.9852	0.0064	0.9889	0.0079	11,988	KELM	155	16
3	$SPEC_{CMI}$	98.94	1	0.9853	0.0078	0.9882	0.0064	2.579	KELM	100	16
4	MIQ	98.92	1	0.9845	0.0066	0.9885	0.0106	2.74	KELM	200	16
5	Inf-FS	98.88	1	0.9835	0.0065	0.9880	0.0139	2.189	KELM	250	16
6	CIFE	98.83	1	0.9815	0.0056	0.9875	0.0135	2.895	KELM	250	16
7	EC-FS	98.80	1	0.9830	0.0078	0.9870	0.0113	0.360	KELM	200	16
8	HOG	98.75	1	0.9795	0.0053	0.9867	0.0159	-	KELM	324	16
9	ANFC-LH	98.72	1	0.9852	0.0110	0.9863	0.0153	590.23	KELM	70	16
10	πHOG	98.31	1	0.9732	0.0079	0.9821	0.0347	-	KELM	616	16
11	V-HOG	98.27	1	0.9788	0.0146	0.9811	0.0145	-	KELM	252	8
12	FCBF	98.26	1	0.9802	0.0153	0.9814	**0.0055**	2.013	KELM	90	8

TABLE 7. comparison of the best accuracy of DPHOG with best accuracies of the group of feature extraction methods and the selected feature selection methods in the KITTI dataset.

#	Method	Acc. (%)	h	TPR	FPR	F1-score	TS. (second)	RT. (second)	Classifier	#Feature	s
1	DPHOG (α= -0.006, l=15)	**98.80**	-	0.9798	0.0095	**0.9747**	0.5721	**0.1126**	SVM	5,355	4
2	$SPEC_{CMI}$	**98.80**	0	0.9824	0.0103	**0.9747**	0.7807	1417.70	SVM	6,000	4
3	WOA-SA	98.78	1	0.9794	0.0095	0.9743	0.5629	1.41×10^5	SVM	4,512	4
4	MIQ	98.77	1	0.9805	0.0106	0.9743	0.1586	79.482	SVM	1,500	8
5	CIFE	98.63	1	0.9765	0.0105	0.9712	0.1856	3.830	SVM	1,500	8
6	EC-FS	98.57	1	0.9672	**0.0082**	0.9701	0.1997	1.946	SVM	1,500	8
7	Inf-FS	98.49	1	0.9689	0.0099	0.9683	0.7612	929.89	SVM	6,000	4
8	πHOG	98.20	1	0.9599	0.0097	0.9625	0.2901	-	SVM	2,920	8
9	ANFC-LH	98.13	1	0.9846	0.0195	0.9599	0.0050	430.60	KELM	95	16
10	HOG	98.05	1	0.9539	0.0087	0.9597	0.8603	-	SVM	8,100	4
11	FCBF	97.78	1	**0.9849**	0.0242	0.9522	**0.0023**	1.782	KELM	71	8
12	VHOG	97.48	1	0.9748	0.0251	0.9456	0.0096	-	KELM	252	8

every dataset and it was faster in both the CompCars and GTI datasets.

(10) The selected well-known feature selection methods (WOA-SA, MIQ, Inf-FS, EC-FS, CIFE, and $SPEC_{CMI}$) produced higher performances regarding accuracy and F1-score, and testing times than both the V-HOG and πHOG in every dataset. They did, however, require a long time in selecting features, resulting in their inability to increase the accuracy of HOG in the CompCars dataset. We can therefore conclude that these feature extraction methods alone are insufficient to represent the different characteristics of vehicles and non-vehicles.

The proposed DPHOG is a novel method for selecting features of vehicles and non-vehicles, which is optimal in both running and testing times, and produces the best performance concerning the accuracy, TPR, and F1-score in all datasets. DPHOG also provides an easy method of computation by using ideal vectors of vehicles and non-vehicles that is fastest in running time for selecting features, compared to all other methods.

This paper focuses on increasing the performance of HOG by selecting the dominant patterns of HOG. Tables 8 and 9 depict the highest accuracies of HOG, and the optimal configuration and optimal common configuration of DPHOG when using the KELM and SVM classifiers, referred to as $DPHOG_B$ and $DPHOG_C$, respectively. KELM with $DPHOG_B$ and $DPHOG_C$ produced better accuracies, TPR, FPR, F1-scores than KELM with HOG in every dataset. Within the GTI dataset, the $DPHOG_B$ and $DPHOG_C$ applied the same parameters (α and l). Each classification technique, KELM with $DPHOG_C$ and KELM with HOG, produced similar results regarding TPR and FPR in both the CompCars

TABLE 8. Comparison of the accuracies of DPHOG with HOG when using the KELM classifier.

Method	CompCars						
	Acc. (%)	TPR	FPR	F1-score	#Feature	TS	*s*
$DPHOG_B$ (α=0.0, *l*=15)	**99.52**	**0.9938**	**0.0039**	**0.9942**	**165**	**0.027**	16
$DPHOG_C$ (α=0.0, *l*=20)	99.43	0.9928	0.0047	0.9932	200	0.035	16
HOG	99.36	0.9972	0.0044	0.9923	324	0.053	16
Method	GTI						
	Acc. (%)	TPR	FPR	F1-score	#Feature	TS	*s*
$DPHOG_B$ (α=0.0, *l*=20) $DPHOG_C$ (α=0.0, *l*=20)	**99.01**	**0.9861**	**0.0052**	**0.9890**	**160**	**0.0091**	16
HOG	98.75	0.9795	0.0053	0.9867	324	0.0160	16
Method	KITTI						
	Acc. (%)	TPR	FPR	F1-score	#Feature	TS	*s*
$DPHOG_B$ (α=0.001, *l*=5)	**98.18**	**0.9860**	**0.0195**	**0.9608**	**145**	**0.0065**	16
$DPHOG_C$ (α=0.0, *l*=20)	97.91	0.9823	0.0220	0.9549	180	0.0074	16
HOG	97.82	0.9859	0.0240	0.9527	324	0.0950	16

TABLE 9. Comparison of the Accuracies of DPHOG with HOG when using the SVM classifier.

Method	CompCars						
	Acc. (%)	TPR	FPR	F1-score	#Feature	TS	*s*
$DPHOG_B$ (α=-0.006, *l*=10) $DPHOG_C$ (α=-0.006, *l*=10)	**99.00**	**0.9905**	0.0102	**0.9881**	**4,780**	**4.176**	4
HOG	98.95	0.9869	**0.0087**	0.9874	8,100	6.213	4
Method	GTI						
	Acc. (%)	TPR	FPR	F1-score	#Feature	TS	*s*
$DPHOG_B$ (α=-0.006, *l*=10) $DPHOG_C$ (α=-0.006, *l*=10)	97.76	0.9772	0.0219	0.9760	**5,060**	**1.347**	4
HOG	**97.89**	**0.9788**	**0.0209**	**0.9774**	8,100	2.390	4
Method	KITTI						
	Acc. (%)	TPR	FPR	F1-score	#Feature	TS	*s*
$DPHOG_B$ (α=-0.006, *l*=15)	**98.80**	**0.9798**	**0.0086**	**0.9747**	**5,355**	**0.572**	4
$DPHOG_C$ (α=-0.006, *l*=10)	98.63	0.9771	0.0109	0.9710	5,290	0.566	4
HOG	98.05	0.9539	0.0087	0.9529	8,100	0.860	4

and KITTI datasets. Additionally, $DPHOG_B$ and $DPHOG_C$ reduced the number of features of HOG by more than 39% in every dataset and performed faster than HOG in the testing times in all datasets.

According to Table 9, SVM with $DPHOG_B$ and $DPHOG_C$ produced higher accuracy, TPR, and F1-scores than HOG with SVM, and reduced the number of features by more than 33% in both the CompCars and KITTI datasets. SVM with $DPHOG_B$ and $DPHOG_C$ produced lower accuracies, TPR, and F1-score than HOG with SVM in the GTI dataset; given that both the $DPHOG_B$ and $DPHOG_C$ reduced the number of features of HOG by about 38% in the GTI dataset.

In summary, the optimal configuration and the optimal common configuration of DPHOG is capable of reducing the number of features and time of HOG for every dataset. In addition, the optimal configuration of DPHOG using KELM resulted in better performance than KELM with HOG for every dataset. The optimal configuration of DPHOG with KELM produced the best performances in both the CompCars and GTI datasets; whereas SVM with the optimal configuration of DPHOG produced the best accuracy in the KITTI dataset. The optimal common configuration of DPHOG with KELM outperformed both HOG with KELM and HOG with SVM in terms of TPR and F1-score in every dataset.

VI. DISCUSSION

This section discusses DPHOG in three sub-discussions. First, we present the different features within the dominant and non-dominant patterns of HOG, in which the dominant patterns can more capably separate vehicles and non-vehicles than the non-dominant patterns. Second, we demonstrate the performance of the DPHOG and its increase in HOG accuracy in various classifiers, such as the K-nearest neighbor algorithm (KNN) and random forest (RF). Finally, we compare the performance of DPHOG with the spare auto-encoder (SAE) based deep neural network (DNN) [54], as well as with the DPHOG, when combined with the spare auto-encoder based DNN.

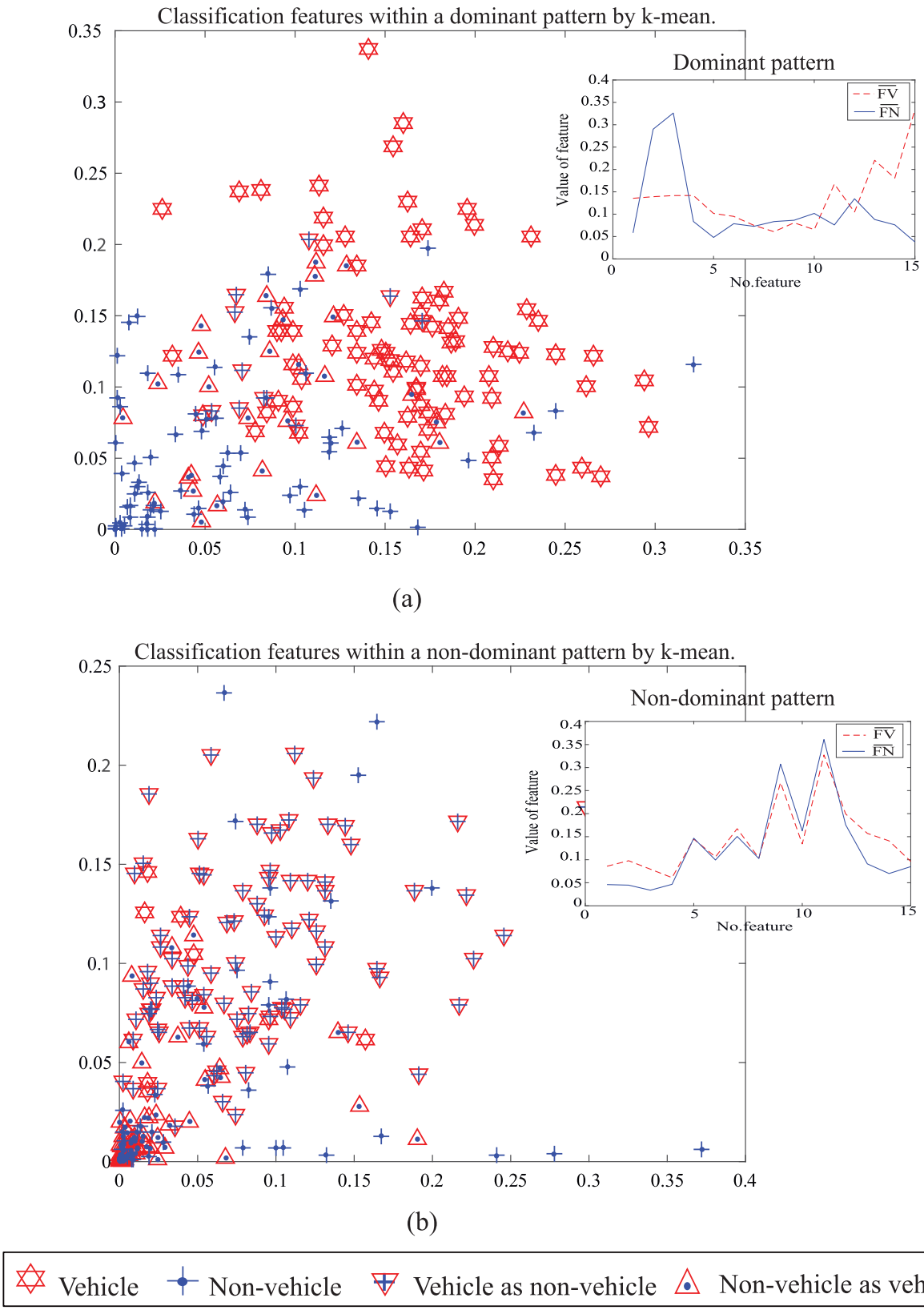


FIGURE 12. Classifying vehicles and non-vehicles features through the k-means method and using the optimal configuration of DPHOG within the (a) dominant pattern and (b) non-dominant pattern in the CompCars dataset.

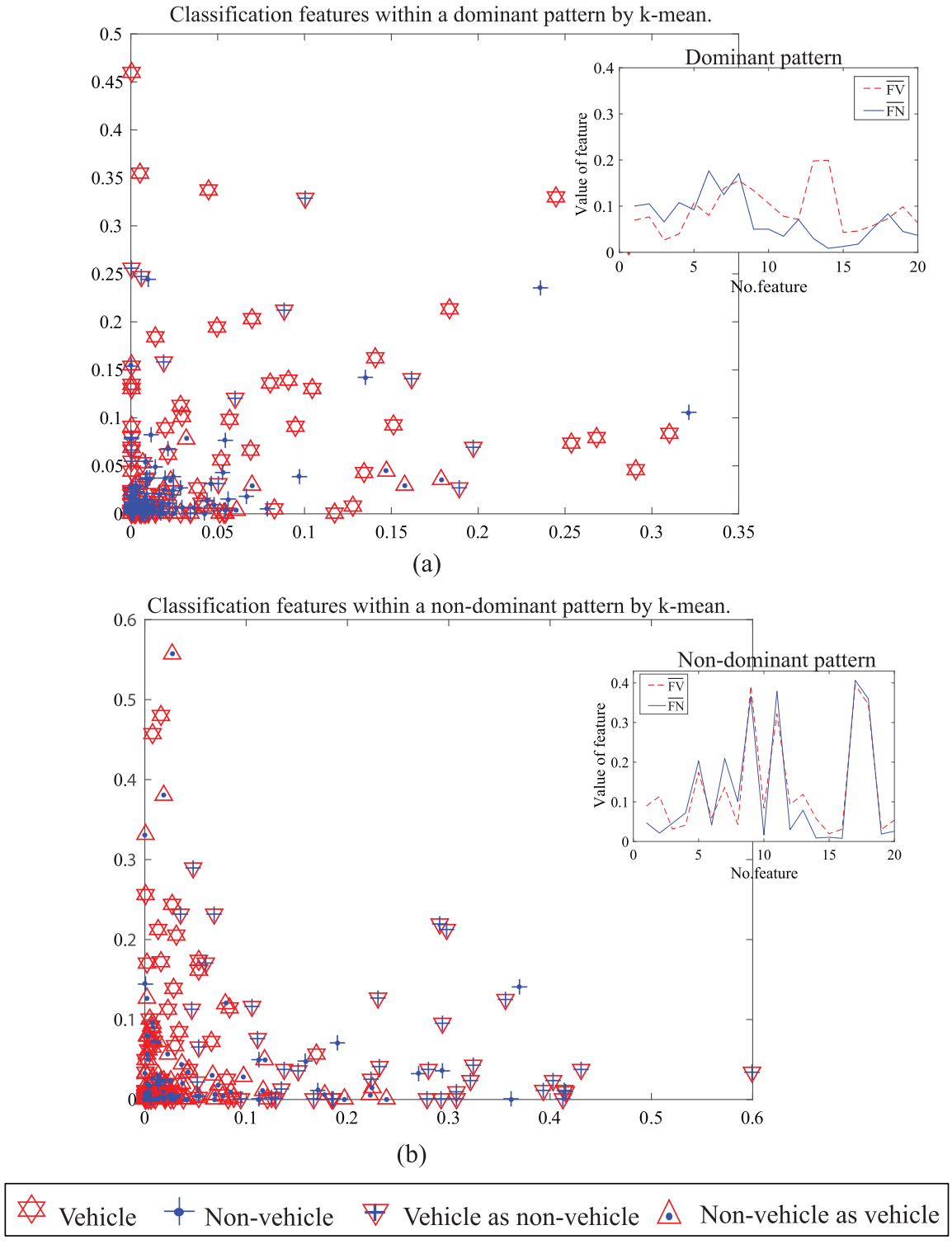


FIGURE 13. Classifying vehicles and non-vehicles features through the k-means method and using the optimal configuration of DPHOG within the (a) dominant pattern and (b) non-dominant pattern in the GTI dataset.

We first present the features that appeared within the dominant and non-dominant patterns of HOG. To easily visualize the separation of the vehicle from non-vehicle features within the dominant and non-dominant patterns, we used the k-means method [55] to present their classification features. The k-means method uses the default parameters of Matlab 2012a in each dataset. Figures 12, 13, and 14 illustrate both the dominant pattern and non-dominant pattern

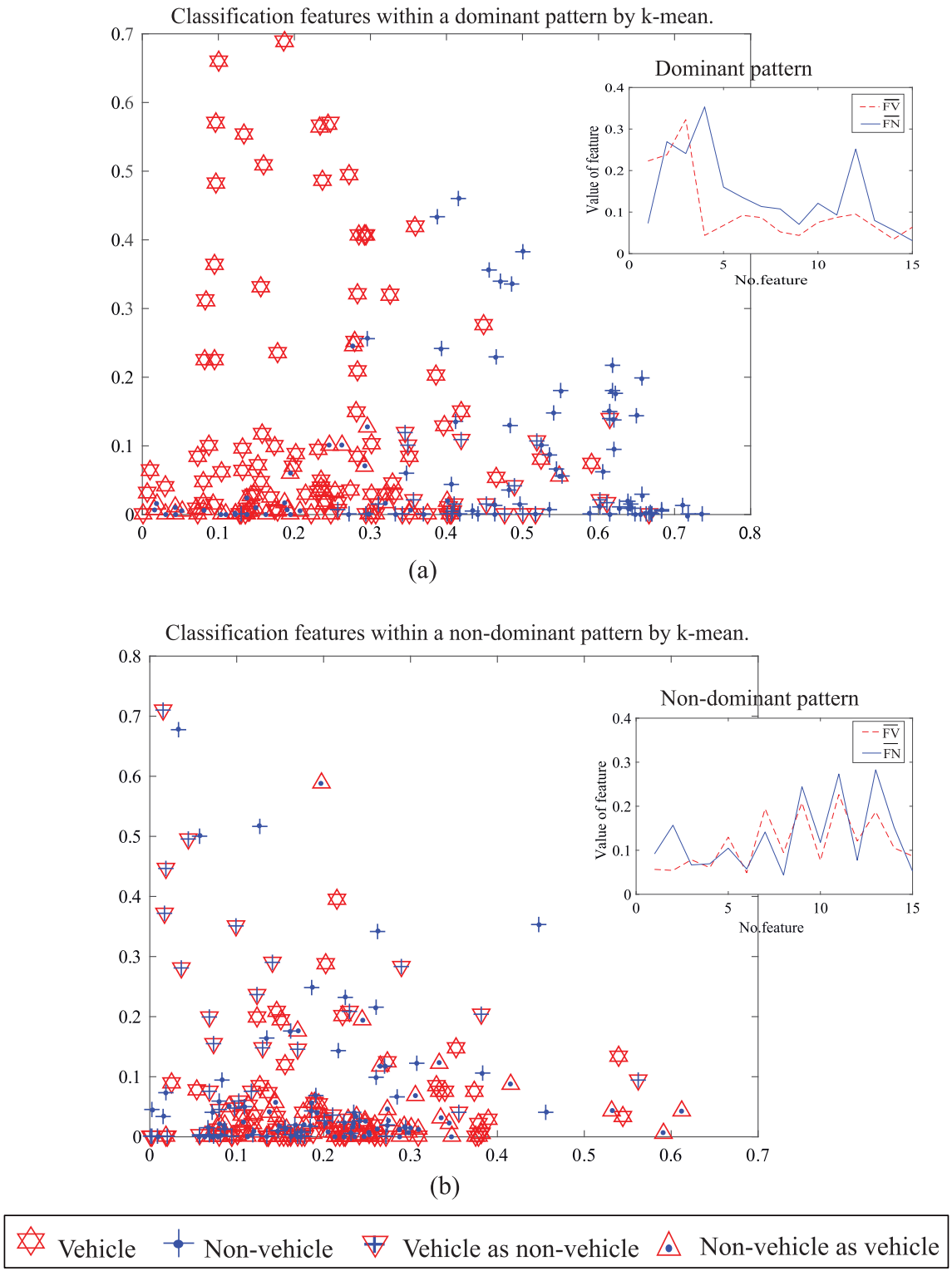


FIGURE 14. Classifying vehicles and non-vehicles features through the k-means method and using the optimal configuration of DPHOG within the (a) dominant pattern and (b) non-dominant pattern in the KITTI dataset.

characteristics in the CompCars, GTI, and KITTI datasets, respectively. The total number of images is comprised of 100 vehicle and 100 non-vehicle images, as shown in Figs. 12, 13, and 14.

Figure 12 illustrates the classification of features within the dominant and non-dominant patterns of the CompCars dataset. We used the optimal configuration for demonstrating the classification features of k-means. The optimal

TABLE 10. Comparison of the performance of DPHOG with HOG, $SPEC_{CMI,}$ and WOA-SA when using the KNN classifier and an assigned cell size of 16.

Method	CompCars				
	Acc. (%)	TPR	FPR	F1-score	TS (second)
HOG	98.08	0.97326	0.01371	0.97706	0.00025
$SPEC_{CMI}$	97.77	0.97305	0.01898	0.97323	0.00017
WOA-SA	98.16	0.97558	0.01411	0.97793	0.00014
DPHOG (α=-0.003, *l*=10)	98.05	0.97358	0.01447	0.97669	0.00015
Method	GTI				
	Acc. (%)	TPR	FPR	F1-score	TS (second)
HOG	98.10	0.97216	0.01091	0.97986	0.00052
$SPEC_{CMI}$	98.09	0.97872	0.01720	0.97957	0.00009
WOA-SA	97.79	0.96707	0.01227	0.97657	0.00016
DPHOG (α=-0.006, *l*=20)	98.39	0.97775	0.01060	0.98286	0.00014
Method	KITTI				
	Acc. (%)	TPR	FPR	F1-score	TS (second)
HOG	96.11	0.95407	0.03693	0.91505	0.00009
$SPEC_{CMI}$	98.01	0.97125	0.01195	0.97884	0.00008
WOA-SA	98.09	0.99182	0.04686	0.98674	0.00007
DPHOG (α=-0.006, *l*=10)	96.30	0.95871	0.03586	0.91905	0.00006

TABLE 11. Comparison of the performance of DPHOG with HOG, $SPEC_{CMI,}$ and WOA-SA; with the RF classifier and an assigned cell size of 16.

Method	CompCars					
	Acc. (%)	TPR	FPR	F1-score	TS (second)	#Root
HOG	93.75	0.94092	0.06483	0.92354	50.53	200
$SPEC_{CMI}$	95.31	0.92365	0.02400	0.94503	38.55	200
WOA-SA	93.65	0.94549	0.06932	0.92185	80.53	300
DPHOG(α=0.001, *l*=25)	94.53	0.94050	0.05116	0.93386	58.81	300
DPHOG(α=-0.003, *l*=25)	94.41	0.92903	0.04488	0.93322	46.17	200
Method	GTI					
	Acc. (%)	TPR	FPR	F1-score	TS (second)	#Root
HOG	96.44	0.96971	0.04018	0.96157	20.51	100
$SPEC_{CMI}$	95.88	0.02567	0.96489	0.96489	42.38	200
WOA-SA	95.78	0.95557	0.04019	0.95485	20.69	200
DPHOG(α=-0.006, *l*=20)	96.85	0.96369	0.02731	0.96636	21.23	300
DPHOG(α=-0.003, *l*=25)	96.56	0.96266	0.03180	0.96321	19.65	300
DPHOG(α=-0.006, *l*=25)	96.52	0.96183	0.03185	0.96280	16.03	200
Method	KITTI					
	Acc. (%)	TPR	FPR	F1-score	TS (second)	#Root
HOG	80.71	1.00000	0.2020	0.3182	0.68188	10
$SPEC_{CMI}$	85.28	0.99777	0.1618	0.5540	9.4284	200
WOA-SA	80.65	1.00000	0.2026	0.3158	0.67778	10
DPHOG(α=0.001, *l*=10)	84.60	0.99036	0.1675	0.5246	0.49469	10

configuration of the CompCars has $l = 15$, which indicates that the number of features within both the dominant and non-dominant patterns is 15; as seen in Figs. 12(a) and (b), respectively. While k-means capably classified vehicle and non-vehicle features within the dominant pattern, many misclassifications occurred (vehicles as non-vehicles, and non-vehicles as vehicles) within the non-dominant pattern.

Figures 13 and 14 present the GTI and KITTI datasets. The dominant pattern was capable of separating vehicles and non-vehicles more effectively than the non-dominant pattern, similar to the CompCars dataset. The DPHOG, therefore, produced high performances and increased the performances of HOG in all three datasets.

We then demonstrated the performance of DPHOG, which increased HOG performances with various classifiers. From the experiment results, the KELM and SVM classifiers produced high accuracy and fast computation times when compared with the selected well-known classifiers in vehicle

TABLE 12. Comparison of the performance of SAE based DNN, combination HOG with SAE based DNN, combination DPHOG with SAE based DNN, HOG using KELM, SVM, KNN and RF as classifiers and DPHOG using KELM, SVM, KNN and RF as classifiers.

Method	CompCars						
	Acc. (%)	TPR	FPR	F1-score	TS	Classifier	Parameter
SAE	96.67	0.9731	0.0377	0.9594	2.6965	DNN	H=200
HOG+SAE	99.23	0.9924	0.0077	0.9907	0.3348	DNN	H=100
DPHOG (α=-0.006, l=5) + SAE	99.26	0.9915	0.0067	0.9911	0.2507	DNN	H=100
HOG	99.36	0.9972	0.0044	0.9923	0.0530	KELM	-
DPHOG (α=0, l=15)	**99.52**	**0.9938**	**0.0039**	**0.9942**	0.0270	**KELM**	**-**
HOG	97.76	0.9592	0.0075	0.9736	0.1230	SVM	-
DPHOG (α=-0.006, l=30)	98.52	0.9815	0.0122	0.9822	0.1081	SVM	-
HOG	98.08	0.9733	0.0137	0.9771	0.0002	KNN	-
DPHOG (α=-0.003, l=10)	98.05	0.9736	0.0145	0.9767	0.0002	KNN	-
HOG	93.75	0.9409	0.0648	0.9235	58.5340	RF	-
DPHOG	94.53	0.9405	0.0512	0.9339	50.8060	RF	α=0.001, l=25
Method	GTI						
	Acc. (%)	TPR	FPR	F1-score	TS	Classifier	Parameter
SAE	95.06	0.9548	0.0531	0.9467	0.7933	DNN	H=200
HOG+SAE	98.99	0.9849	0.0057	0.9892	0.6777	DNN	H=200
DPHOG (α=-0.003, l=20)+SAE	98.94	0.9844	0.0062	0.9887	0.5602	DNN	H=200
HOG	98.75	0.9795	0.0053	0.9867	0.0159	KELM	-
DPHOG (α=0, l=20)	**99.01**	**0.9861**	**0.0052**	**0.9890**	0.0091	**KELM**	**-**
HOG	97.80	0.9751	0.0194	0.9765	0.0473	SVM	-
DPHOG (α=-0.003, l=20)	97.41	0.9723	0.0242	0.9723	0.0519	SVM	-
HOG	98.10	0.9722	0.0109	0.9799	0.0005	KNN	-
DPHOG (α=-0.006, l=20)	98.39	0.9778	0.0106	0.9829	0.0001	KNN	-
HOG	96.44	0.9697	0.0402	0.9616	21.5140	RF	-
DPHOG (α=-0.006, l=20)	96.85	0.9637	0.0273	0.9664	20.2290	RF	-
Method	KITTI						
	Acc. (%)	TPR	FPR	F1-score	TS	Classifier	Parameter
SAE	93.41	0.9066	0.0585	0.8535	0.1479	DNN	H=100
HOG+SAE	98.60	0.9523	0.0129	0.9703	0.0765	DNN	H= 100
DPHOG (α=-0.006, l=20)+SAE	**98.68**	0.9773	**0.0102**	**0.9722**	0.0412	**DNN**	H= 100
HOG	97.82	0.9859	0.0240	0.9527	0.0950	KELM	-
DPHOG (α=0.001, l=5)	98.18	0.9860	0.0195	0.9608	0.0065	KELM	-
HOG	96.42	0.9176	0.0143	0.9271	0.0298	SVM	-
DPHOG (α=-0.006, l=20)	97.78	0.9597	0.0167	0.9530	0.0126	SVM	-
HOG	96.11	0.9541	0.0369	0.9151	0.0001	KNN	-
DPHOG (α=-0.006, l=10)	96.30	0.9587	0.0359	0.9191	0.0001	KNN	-
HOG	80.71	1.0000	0.2020	0.3182	0.6819	RF	-
DPHOG (α=0.001, l=10)	84.60	0.9904	0.1675	0.5246	0.4947	RF	-

detection. Within this sub-discussion, we show how the proposed DPHOG increases performance and reduces computation times of HOG with various classifiers (KNN [21] and RF [35]) in vehicle detection. We further compared the DPHOG's performance with HOG with various well-known feature selection methods ($SPEC_{CMI}$ and WOA-SA), as they produce the highest performance among all well-known feature selection methods. The methods can increase the performance of HOG, represented in bold font in Tables 10 and 11. We assigned the cell size of ($s = 16$), which more capably supports real-time more accurately than with $s = 4$ and $s = 8$.

Table 10 compares performances of DPHOG with HOG, $SPEC_{CMI}$, and WOA-SA when using KNN as the classifier. The WOA-SA showed an increase in performance in terms of accuracy, TPR, F1-score, and testing time of HOG in twin CompCars and KITTI datasets. The DPHOG increased performance in terms of accuracy, TPR, FPR, F1-score, and testing time of HOG in the twin GTI and KITTI datasets. In contrast, $SPEC_{CMI}$ increased performance in terms of accuracy, TPR, FPR, F1-score, and testing time of HOG in only the KITTI datasets. While both the DPHOG and WOA-SA demonstrated superior flexibility in the multiple datasets, the DPHOG demonstrated greater vehicle detection capabilities over the WOA-SA, as it proved faster in selecting features and offered easier computation than the WOA-SA. The computation times required for selecting features of DPHOG and WOA-SA are shown in Tables 5, 6, and 7.

Table 11 compares the performance of DPHOG with HOG, $SPEC_{CMI}$, and WOA-SA when using RF as the classifier. We varied the number of roots of RF to find the optimal number of roots, which were 10, 100, 200, and 250. The highest accuracy of each method is compared in Table 11. $SPEC_{CMI}$ produced the highest accuracy when using the root number 200 in all three datasets, as well as increased performance in terms of accuracy and FPR of HOG in the twin CompCars and KITTI datasets. Notably, the DPHOG also

increased performance in terms of accuracy, F1-score, and testing time of HOG in all three datasets. As DPHOG selects only the dominant patterns of HOG, the features selected are more capable of separating vehicles from non-vehicles than HOG (Figs. 12, 13, and 14).

Finally, we compared the performance of DPHOG with SAE based DNN, as well as comparing the combination of DPHOG and HOG with SAE based DNN. SAE is a feature learning method capable of producing high performance in the complex dataset [56]. Within the combination of DPHOG and HOG, features are selected and fed to SAE for feature identification, using DNN as the classifier. In this subdiscussion, DPHOG and HOG utilized a cell size of 16, and an image size of 64×64 pixels. Within the SAE based DNN, images were assigned sizes of either 28×28 pixels or 32×32 pixels. A hidden layer of size (H) was assigned the varied values of 10, 100, and 200.

Table 12 compares the highest accuracies of the (1) SAE based DNN, (2) combination of HOG with SAE based DNN, (3) combination of DPHOG with SAE based DNN, (4) HOG using KELM, SVM, KNN, and RF classifiers, and (5) DPHOG using KELM, SVM, KNN, and RF classifiers. The results are summarized as follows:

(1) SAE based DNN produced the highest accuracy in all three datasets when using the image size of 28×28 pixels.

(2) The combination of DPHOG with SAE based DNN and the combination HOG with SAE based DNN produced the highest performances in terms of accuracy, TPR, FPR, F1-score, and testing times, making it superior to the SAE based DNN in all three datasets. We may conclude that learning features through selections from HOG and DPHOG before using SAE is more beneficial than through only pixels within images.

(3) The combination of DPHOG with SAE based DNN produced better performances in terms of accuracy, TPR, FPR, F1-score, and testing time than the combination of HOG with SAE based DNN in all three datasets.

(4) The SAE based DNN produced lower performance levels than DPHOG and HOG, using KELM, SVM, and KNN classifiers in all three datasets.

(5) DPHOG with KELM produced the best performances in terms of accuracy, TPR, FPR, and F1-score in the CompCars and GTI datasets. The combination of DPHOG with SAE base DNN produced the highest accuracy and F1-score in the KITTI dataset, yet scored the lowest FPR.

In summary, the DPHOG, through the use of the KELM, SVM, KNN, and RF classifiers, provided better performance than the SAE based DNN. Additionally, the combination of DPHOG with SAE based DNN yielded better performances than the combination of HOG with SAE based DNN and the SAE based DNN in multiple datasets. Therefore, selecting only the dominant patterns of DPHOG produced higher performances in the separation of vehicles from non-vehicles in various classifiers and datasets, and improved the performances of SAE based DNN in multiple datasets.

VII. CONCLUSION AND FUTURE WORKS

This paper proposes a new method, referred to as DPHOG, for selecting dominant patterns of HOG for vehicle detection within the hypothesis verification (HV) step. DPHOG uses the ideal vector of vehicles and the ideal vector of non-vehicles, which are segmented as l-consecutive features within the k^{th} segment of each group. Comparisons of their differences were made, in which similarities within the non-dominant pattern were discarded. Performance evaluations of the DPHOG were conducted on three standard datasets; CompCars, GTI, and KITTI, using both SVM and KELM as classifiers. The results were compared with the performances of the standard HOG, V-HOG, and πHOG feature extraction methods, as well as several well-known feature selection methods (ANFC-LH, WOA-SA, FCBF, MIQ, CIFE, SPEC_{CMI}, Inf-FS, and EC-FS). The value of s was varied in each method to seek the highest accuracy.

DPHOG proved optimal for solving problems concerning classification rates and computation times within the classification stage and running time for selecting features. Experiment results of the DPHOG found that the optimal configuration of DPHOG with KELM (using $s = 16$) outperformed the HOG, V-HOG, and πHOG; as well as the various well-known feature selection methods in both the CompCars and GTI datasets. In the KITTI dataset, the optimal configuration of DPHOG with SVM (using $s = 4$) produced the highest performance of all the comparative methods and provided the fastest running times in selecting features in all three datasets. Additionally, the DPHOG also displayed superiority in vehicle detection ability compared with the feature extraction methods (HOG, V-HOG, and πHOG), as well as all of the feature selection methods, in all three datasets. DPHOG encompassed easier implementation and less computation time in classifying objects into two different classes (positive and negative classes), for example, gender and pedestrian classifications.

While the well-known feature selection methods (WOA-SA, SPEC_{CMI}, MIQ, EC-FS, CIFE, and Inf-FS) used in our experiments outperformed HOG, V-HOG, and πHOG, we found that the corresponding feature extraction methods presented problems in the representation of vehicles and non-vehicles, as well as in the testing times within the classification stage.

DPHOG increased the performance of HOG with various classifiers (KELM, SVM, KNN, and RF), and produced better performance than the SAE based DNN. Additionally, the combination of DPHOG with the SAE based DNN outperformed the SAE based DNN, as well as the combination of HOG with the SAE based DNN. Therefore, selecting only the dominant patterns of DPHOG produced higher performances in the separation of vehicles from non-vehicles in various classifiers and datasets, and improved the performances of SAE based DNN in multiple datasets.

In future work, we shall further task the DPHOG to select features of gender in pedestrians [57], and more

accurately identify non-pedestrian features [58], [59]. Furthermore, we will apply DPHOG with other classifiers; for example, combining the extended Kalman filter with a cost-sensitive, dissimilar extreme learning machine (EKF-CS-D-ELM) [60]. The EKF-CS-D-ELM classifier produces fast and highly accurate real-time results that might support real-time vehicle detection.

ACKNOWLEDGMENT

N. Laopracha would like to thank Asst. Prof. Dr. Phattanapong Chomphuwiset, Asst. Prof. Dr. Jantima Polpinij, and Asst. Prof. Dr. Rapeeporn ChamChong for their generous and helpful suggestions.

REFERENCES

[1] Z. Yang and L. S. C. Pun-Cheng, "Vehicle detection in intelligent transportation systems and its applications under varying environments: A review," *Image Vis. Comput.*, vol. 69, pp. 143–154, Jan. 2018.
[2] W. Chu, Y. Liu, C. Shen, D. Cai, and X.-S. Hua, "Multi-task vehicle detection with region-of-interest voting," *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 432–441, Jan. 2018.
[3] V. Milanés *et al.*, "Intelligent automatic overtaking system using vision for vehicle detection," *Expert Syst. Appl.*, vol. 39, no. 3, pp. 3362–3373, Feb. 2012.
[4] Y. Zhou, L. Liu, L. Shao, and M. Mellor, "Fast automatic vehicle annotation for urban traffic surveillance," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 6, pp. 1973–1984, Jun. 2018.
[5] H. Tayara, K. G. Soo, and K. T. Chong, "Vehicle detection and counting in high-resolution aerial images using convolutional regression neural network," *IEEE Access*, vol. 6, pp. 2220–2230, Feb. 2018.
[6] P. R. L. de Almeida, L. S. Oliveira, A. S. Britto, Jr., E. J. Silva, Jr., and A. L. Koerich, "PKLot—A robust dataset for parking lot classification," *Expert Syst. Appl.*, vol. 42, no. 11, pp. 4937–4949, Feb. 2015.
[7] H. Schilling, D. Bulatov, and W. Middelmann, "Object-based detection of vehicles using combined optical and elevation data," *ISPRS J. Photogramm. Remote Sens.*, vol. 136, pp. 85–105, Feb. 2018.
[8] Z. Ouyang, J. Niu, and M. Guizani, "Improved vehicle steering pattern recognition by using selected sensor data," *IEEE Trans. Mobile Comput.*, vol. 17, no. 6, pp. 1383–1396, Jun. 2018.
[9] M. Cheon, W. Lee, C. Yoon, and M. Park, "Vision-based vehicle detection system with consideration of the detecting location," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 1243–1252, Sep. 2012.
[10] X. Wen, L. Shao, W. Fang, and Y. Xue, "Efficient feature selection and classification for vehicle detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 3, pp. 508–517, Mar. 2015.
[11] M. D. Bugdol, P. Badura, J. Juszczyk, W. Wieclawek, and M. J. Bienkowska, "System for detecting vehicle features from low quality data," *Promet-Traffic-Traffico*, vol. 30, no. 1, pp. 11–20, Feb. 2018.
[12] X. Wen, L. Shao, Y. Xue, and W. Fang, "A rapid learning algorithm for vehicle classification," *Inf. Sci.*, vol. 295, pp. 395–406, Feb. 2015.
[13] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection using evolutionary Gabor filter optimization," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 2, pp. 125–137, Jun. 2005.
[14] J. Arróspide and L. Salgado, "Log-Gabor filters for image-based vehicle verification," *IEEE Trans. Image Process.*, vol. 22, no. 6, pp. 2286–2295, Jun. 2013.
[15] J. Arróspide, L. Salgado, and M. Camplani, "Image-based on-road vehicle detection using cost-effective histograms of Oriented Gradients," *J. Vis. Commun. Image Represent.*, vol. 24, no. 7, pp. 1182–1190, Oct. 2013.
[16] Y. Wei, Q. Tian, J. Guo, W. Huang, and J. Cao, "Multi-vehicle detection algorithm through combining Harr and HOG features," *Math. Comput. Simul.*, vol. 155, pp. 130–145, Jan. 2018.
[17] J. Lan, Y. Jiang, G. Fan, D. Yu, and Q. Zhang, "Real-time automatic obstacle detection method for traffic surveillance in urban traffic," *J. Signal Process. Syst.*, vol. 82, no. 3, pp. 357–371, Mar. 2016.
[18] Y. Xu, G. Yu, Y. Wang, X. Wu, and Y. Ma, "A hybrid vehicle detection method based on viola-jones and HOG + SVM from UAV images," *Sensors*, vol. 16, no. 8, p. E1325, Aug. 2016.
[19] D.-Y. Huang, C.-H. Chen, T.-Y. Chen, W.-C. Hu, and K.-W. Feng, "Vehicle detection and inter-vehicle distance estimation using single-lens video camera on urban/suburb roads," *J. Vis. Commun. Image Represent.*, vol. 46, pp. 250–259, Jul. 2017.
[20] S. Vitek and P. Melničuk, "A distributed wireless camera system for the management of parking spaces," *Sensors*, vol. 18, no. 1, p. E69, Dec. 2018.
[21] Y. Gao and H. J. Lee, "Local tiled deep networks for recognition of vehicle make and model," *Sensors*, vol. 16, no. 2, p. 226, Feb. 2016.
[22] J. Kim, J. Baek, and E. Kim, "A novel on-road vehicle detection method using πHOG," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 6, pp. 3414–3429, Dec. 2015.
[23] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.
[24] C. De Stefano, F. Fontanella, C. Marrocco, and A. S. di Freca, "A GA-based feature selection approach with an application to handwritten character recognition," *Pattern Recognit. Lett.*, vol. 35, no. 1, pp. 130–141, Jan. 2014.
[25] D. Lin and X. Tang, "Conditional infomax learning: An integrated framework for feature extraction and fusion," in *Proc. Eur. Conf. Comput. Vis.*, May 2006, pp. 68–82.
[26] G. Roffo, S. Melzi, and M. Cristani, "Infinite feature selection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4202–4210.
[27] L. Yu and H. Liu, "Feature selection for high-dimensional data: A fast correlation-based filter solution," in *Proc. 20th Int. Conf. Mach. Learn.*, Aug. 2003, pp. 1–8.
[28] C. Ding and H. Peng, "Minimum redundancy feature selection from microarray gene expression data," in *Proc. IEEE 2nd Comput. Syst. Bioinf. Conf.*, Aug. 2003, pp. 523–528.
[29] X. V. Nguyen, J. Chan, S. Romano, and J. Bailey, "Effective global approaches for mutual information based feature selection," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2014, pp. 512–521.
[30] G. Roffo and S. Melzi, *Ranking to Learn: Feature Ranking and Selection Via Eigenvector Centrality*. Cham, Switzerland: Springer, Apr. 2017, pp. 19–35.
[31] B. Cetisli, "The effect of linguistic hedges on feature selection: Part 2," *Expert Syst. Appl.*, vol. 37, no. 8, pp. 6102–6108, Aug. 2010.
[32] M. M. Mafarja and S. Mirjalili, "Hybrid whale optimization algorithm with simulated annealing for feature selection," *Neurocomputing*, vol. 260, pp. 302–312, Oct. 2017.
[33] S. Khalid, T. Khalil, and S. Nasreen, "A survey of feature selection and feature extraction techniques in machine learning," in *Proc. Sci. Inf. Conf. (SAI)*, Aug. 2014, pp. 372–378.
[34] R.-H. Zhang, F. You, F. Chen, and W.-Q. He, "Vehicle detection method for intelligent vehicle at night time based on video and laser information," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 32, no. 4, pp. 1850009-1–1850009-20, 2018.
[35] W. Zhu, J. Miao, J. Hu, and L. Qing, "Vehicle detection in driving simulation using extreme learning machine," *Neurocomputing*, vol. 128, pp. 160–165, Mar. 2014.
[36] K. Liu and G. Mattyus, "Fast multiclass vehicle detection on aerial images," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 9, pp. 1938–1942, Sep. 2015.
[37] H. Zhang *et al.*, "A novel infrared video surveillance system using deep learning based techniques," vol. 77, no. 20, pp. 26657–26676, Oct. 2018.
[38] S. S. Teoh and T. Braunl, "Performance evaluation of HOG and Gabor features for vision-based vehicle detection," in *Proc. IEEE Int. Conf. Control Syst., Comput. Eng. (ICCSCE)*, Nov. 2015, pp. 66–71.
[39] N. Ammour, H. Alhichri, Y. Bazi, B. Benjdira, N. Alajlan, and M. Zuair, "Deep learning approach for car detection in UAV imagery," *Remote Sens.*, vol. 9, no. 4, p. 312, Mar. 2017.
[40] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, pp. 886–893.
[41] H. Van Pham and B.-R. Lee, "Front-view car detection and counting with occlusion in dense traffic flow," *Int. J. Control. Autom. Syst.*, vol. 13, no. 5, pp. 1150–1160, Oct. 2015.
[42] G. Herman, B. Zhang, Y. Wang, G. Ye, and F. Chen, "Mutual information-based method for selecting informative feature sets," *Pattern Recognit.*, vol. 46, no. 12, pp. 3315–3327, Dec. 2013.
[43] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cognit. Neurosci.*, vol. 3, no. 1, pp. 71–86, 1991.

[44] P. N. Belhumeur, J. P. Hespanha, and D. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.
[45] X. Wang and X. Tang, "A unified framework for subspace face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1222–1228, Sep. 2004.
[46] K. Torkkola and W. M. Campbell, "Mutual information in learning feature transformations," in *Proc. 7th Int. Conf. Mach. Learn.*, 2000, pp. 1015–1022.
[47] S. Egea, A. R. Mañez, B. Carro, A. Sánchez-Esguevillas, and J. Lloret, "Intelligent IoT traffic classification using novel search strategy for fast-based-correlation feature selection in industrial environments," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1616–1624, Jun. 2018.
[48] P. Luukka and T. Leppälampi, "Similarity classifier with generalized mean applied to medical data," *Comput. Biol. Med.*, vol. 36, no. 9, pp. 1026–1040, Sep. 2006.
[49] *The GTI-UPM Vehicle Image Database*. Accessed: Sep. 20, 2014. [Online]. Available: http://www.gti.ssr.upm.es/research/gti-data/databases
[50] L. Yang, P. Luo, C. C. Loy, and X. Tang, "A large-scale car dataset for fine-grained categorization and verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3973–3981.
[51] H. Ji, Y. Wang, H. Qin, Y. Wang, and H. Li, "Comparative performance evaluation of intrusion detection methods for in-vehicle networks," *IEEE Access*, vol. 6, pp. 37523–37532, 2018.
[52] H. Fu, C.-M. Vong, P.-K. Wong, and Z. Yang, "Fast detection of impact location using kernel extreme learning machine," *Neural Comput. Appl.*, vol. 27, no. 1, pp. 121–130, Jan. 2016.
[53] S. B. Plaisier, R. Taschereau, J. A. Wong, and T. G. Graeber, "Rank-rank hypergeometric overlap: Identification of statistically significant overlap between gene-expression signatures," *Nucl. Acids Res.*, vol. 38, no. 17, p. e169, Sep. 2010.
[54] W. Sun, S. Shaoa, R. Zhaob, R. Yana, X. Zhangc, and X. Chen, "A sparse auto-encoder-based deep neural network approach for induction motor faults classification," *Measurement*, vol. 89, pp. 171–178, Jul. 2016.
[55] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. 5th Berkeley Symp. Math. Statist. Probab.*, 1967, vol. 1, no 233, pp. 281–297.
[56] Y. Bengio, "Learning deep architectures for AI," *Found. Trends Mach. Learn.*, vol. 2, no. 1, pp. 1–127, 2009.
[57] L. Cai, J. Zhu, H. Zeng, J. Chen, C. Cai, and K.-K. Ma, "HOG-assisted deep feature learning for pedestrian gender recognition," *J. Franklin Inst.*, vol. 355, no. 4, pp. 1991–2008, Mar. 2018.
[58] K. Park, S. Kim, and K. Sohn, "Unified multi-spectral pedestrian detection based on probabilistic fusion networks," *Pattern Recognit.*, vol. 80, pp. 143–155, Aug. 2018.
[59] A. Brunetti, D. Buongiorno, G. F. Trotta, and V. Bevilacqua, "Computer vision and deep learning techniques for pedestrian detection and tracking: A survey," *Neurocomputing*, vol. 300, pp. 17–33, Jul. 2018.
[60] K. Yan, Z. Ji, H. Lu, J. Huang, W. Shen, and Y. Xue, "Fast and accurate classification of time series data using extended ELM: Application in fault diagnosis of air handling units," *IEEE Trans. Syst., Man, Cybern. Syst.*, to be published.

NATTHARIYA LAOPRACHA received the B.Sc. degree in computer science from Mahasarakham University and the M.S. degree in computer science from Khon Kaen University, Thailand, where she is currently pursuing the Ph.D. degree in computer science. Her current research interests include computer vision, pattern recognition, and intelligent system and machine learning.

KHAMRON SUNAT received the B.Sc. degree in chemical technology (chemical engineering) from Chulalongkorn University, Thailand, in 1989, and the M.Sc. degree in computational science and the Ph.D. degree in computer science from Chulalongkorn University, in 1998 and 2004, respectively. He is currently an Assistant Professor with the Department of Computer Science, Khon Kaen University, Thailand. His research interests include neural networks, pattern recognition, computer vision, soft computing, fuzzy systems, and nature-inspired evolutionary computing and learning.

SIRAPAT CHIEWCHANWATTANA graduated in statistics from Khon Kaen University, Thailand, and received the M.Sc. degree in computer science from the National Institute of Development Administration, Thailand, and the Ph.D. degree in computer science from Chulalongkorn University, in 2007. She is currently an Assistant Professor with the Department of Computer Science, Khon Kaen University, Thailand. She is active in the research groups in intelligence systems and machine learning. She is interested in neural networks, soft computing, and pattern recognition.