

RESEARCH

Open Access

# Vehicle color classification using manifold learning methods from urban surveillance videos

Yu-Chen Wang<sup>1\*</sup>, Chin-Chuan Han<sup>2</sup>, Chen-Ta Hsieh<sup>1</sup> and Kuo-Chin Fan<sup>1</sup>

## Abstract

Color identification of vehicles plays a significant role in crime detection. In this study, a novel scheme for the color identification of vehicles is proposed using the locating algorithm of regions of interest (ROIs) as well as the color histogram features from still images. A coarse-to-fine strategy was adopted to efficiently locate the ROIs for various vehicle types. Red patch labeling, geometrical-rule filtering, and a texture-based classifier were cascaded to locate the valid ROIs. A color space fusion together with a dimension reduction scheme was designed for color classification. Color histograms in ROIs were extracted and classified by a trained classifier. Seven different classes of color were identified in this work. Experiments were conducted to show the performance of the proposed method. The average rates of ROI location and color classification were 98.45% and 88.18%, respectively. Moreover, the classification efficiency of the proposed method was up to 18 frames per second.

**Keywords:** Color classification; Manifold learning; Dimension reduction; Coarse-to-fine strategy; Nearest feature line

## 1. Introduction

Recently, vehicle color identification has been widely in demand for video surveillance on urban roads. When an accident occurs, license plate (LP) numbers are an intuitive and direct cue for the escaping vehicle. However, these cues are ineffective because small LPs can be missed due to view angle or distance. Witnesses only tend to remember the escape vehicle's color or type. Moreover, the government has installed cameras on roads for traffic monitoring or crime prevention. The color identification from video data can assist police both in crime prevention and later investigation.

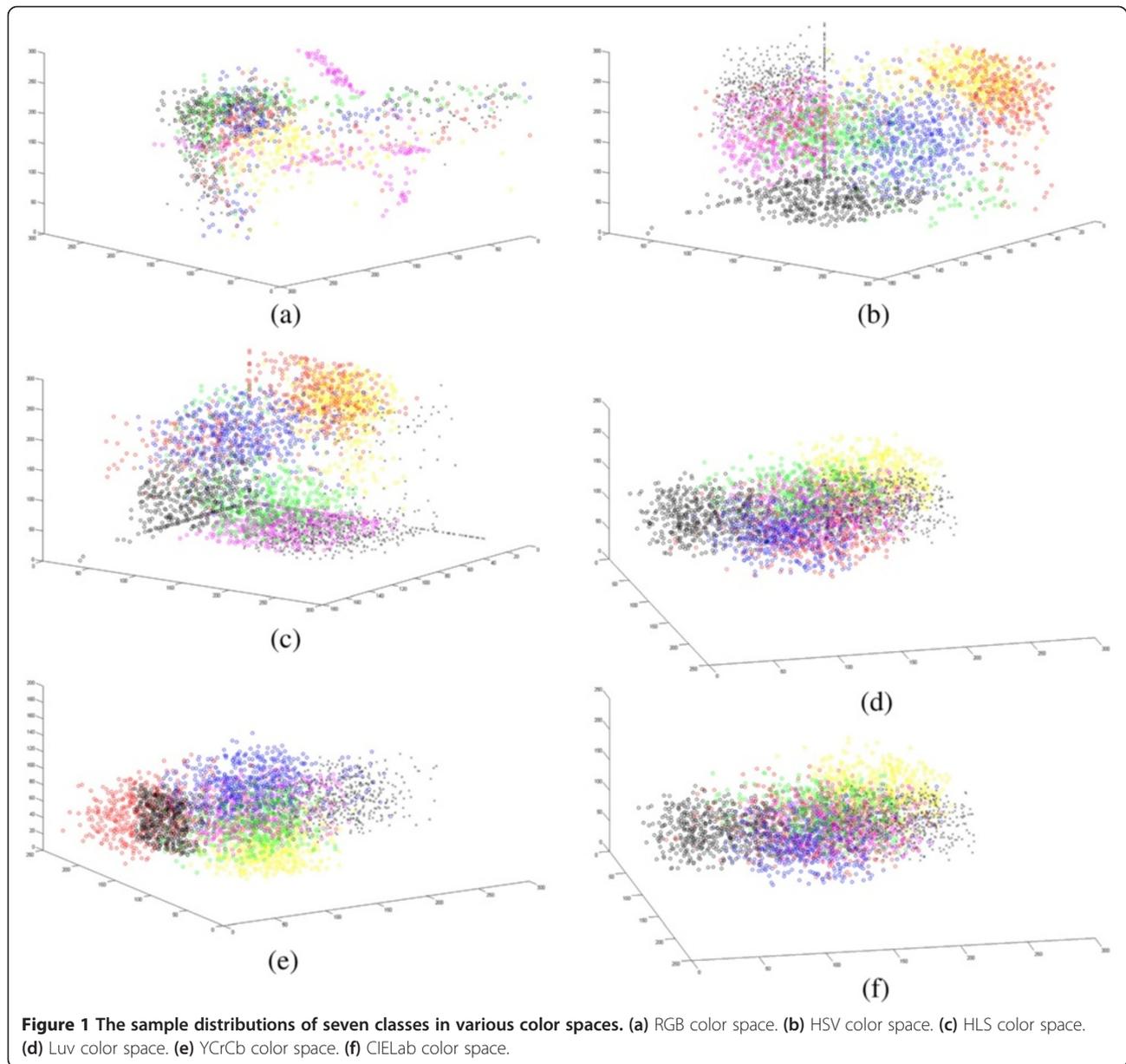
Earlier, color feature descriptors have been widely used in content-based image retrieval (CBIR) [1,2]. High quality images with less illumination impact were required in their studies. However, the color identification of vehicles from outdoor video clips is sensitive to camera installation and environmental factors. First, the sequential images captured from outdoor cameras are distorted by chromatic polarization and white balance functions. Second, the performance is influenced by illumination and weather conditions. As illustrated in Figure 1, classification errors

always occur because the mixed sample distributions of colors cannot be clearly separated in various color spaces. Many researchers try to reduce illumination impacts in classification by two approaches: *discriminative feature extraction* and *image-based color calibration*. Basically, discriminative features are extracted from the color histogram of an individual color space or the fused spaces. Histogram is the widely used object representation in color classification or image retrieval. In MPEG-7 standard, color channels are encoded and quantized to generate the color histogram, e.g., scalable color descriptor (SCD). Baek et al. [3] extracted the two-dimensional histogram features on the hue ( $H$ ) and saturation ( $S$ ) plane. Kim et al. [4] quantized color features on channels  $H$ ,  $S$ , and intensity ( $I$ ) and found the best combination of features. In [5], the color histograms on channels  $H$  and  $S$  are classified to determine the red, yellow, green, and blue colors, while the normalized features on channels  $RGB$  are classified to determine the black, gray, and white colors. The choice of a color space is a critical issue in identifying color objects. Tsai et al. [6] and Chen et al. [7] used the principal component analysis (PCA)-based technique to calculate the eigenvectors and the corresponding eigenvalues from the training samples in color space  $RGB$ . After the transformations of color spaces, the classifiers, e.g., multi-class support vector machine (SVM) [3],  $k$ -nearest neighbor (KNN) [4], Bayesian

\* Correspondence: m09502062@chu.edu.tw

<sup>1</sup>Department of Computer Science and Information Engineering, National Central University, Taoyuan, Taiwan

Full list of author information is available at the end of the article



classifier [6], or the multiple instance learning (MIL) [7], were trained to identify vehicles' colors. Furthermore, Brown [8] evaluates four color feature descriptors: *standard color histogram*, *weighted color histogram*, *variable bin size color histogram*, and *color correlogram*.

On the other hand, image-based color calibration is an enhancement method on image quality to reduce illumination variation. Li et al. [9] propose a classification method using the template-matching strategy. Before the identification process, a color calibration procedure is executed to adjust the image colors [10]. The relative error distances in color space *HSI* are calculated to identify vehicle colors. Similarly, color compensation is performed for

color calibration in [11]. Shen et al. [12] present an image correction algorithm which combines synthesized texture information to recover color information in overexposure regions. Guo et al. [13] separately recover the color and lightness of overexposure regions from a single image. The lightness in channel *L* is recovered using the exposure likelihood, and the colors in channels *a* and *b* are corrected by the weighted summation of neighborhoods. However, their methods [12,13] cannot be used in a real-time surveillance system because of the high computational loading.

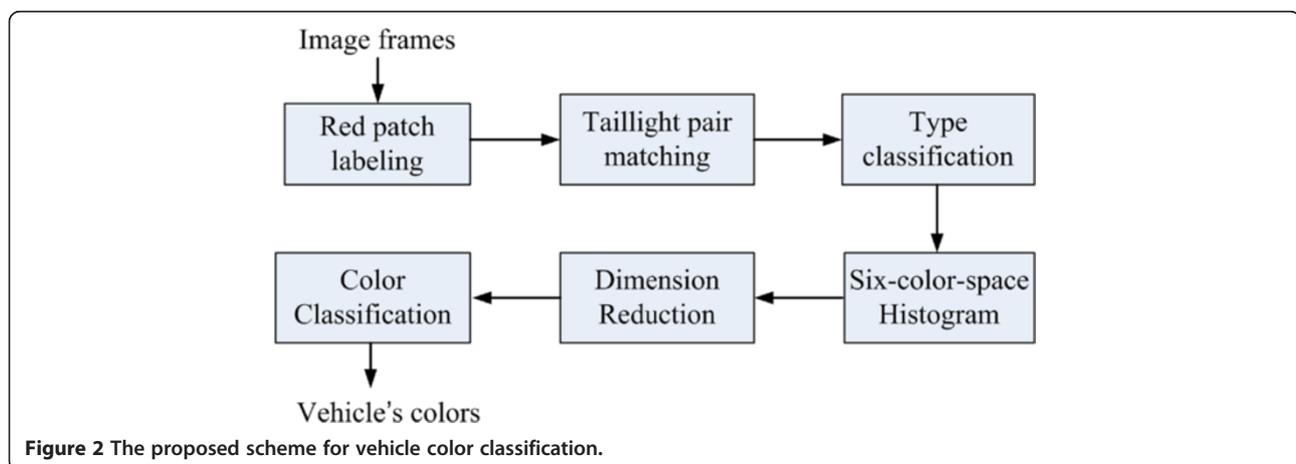
In color histogram-based classification, images with similar colors are represented in the same color histogram

because similar colors fall into the same bins due to the quantization. The original color components should be represented by more bits to keep more details, e.g., 8 bits of 256 levels. In addition, various color spaces represent different meanings. It is a hard work to accurately choose the color space for the classification on various applications. Stokman et al. [14] propose a fusion scheme to integrate several color spaces. Twelve color components are weighted and summarized by a linear programming method [15]. In this study, six color spaces of 18 components, *RGB*, *CIE Lab*, *YCrCb*, *HSV*, *Luv*, and *HLS*, were concatenated to generate a color histogram. However, the dimensionality of this histogram-based descriptor is very high, and samples of the same color are distributed in a manifold structure. Recently, many manifold learning and dimensional reduction (DR) approaches are proposed for face recognition, image classification, image retrieval, etc. Local linear embedding (LLE) [16], locality preservation projection (LPP) [17], nearest feature space embedding (NFSE) [18], ISOMAP [19], and Laplacian eigenmap (LE) [20] try to preserve the structural locality of samples which are distributed in a manifold structure. Even though the objective functions in these methods are different, the goal, preserving the manifold structure, is the same.

In traffic surveillance systems, cameras are frequently set up on islands or shoulders of roads. They capture vehicle images from the front or back view. The safety cameras commonly set on highways or urban roads face the lanes and capture the vehicle images of back view. In this study, vehicle color in multi-lanes was identified outdoors from the back. The identification procedure consisted of two modules as shown in Figure 2: The location of a valid ROI and the color classification. Unlike the traditional background subtraction methods, which need video sequences and are very sensitive to illumination changes, the taillight detection algorithm has been designed to obtain the valid ROI from video frames. The valid regions of interest (ROIs)

were determined by the detected red patches (e.g., vehicle taillights) and their corresponding pairs. The unfeasible taillight pairs were pre-filtered out and eliminated using geometric rules. It is necessary to identify the vehicle types because the locations of ROI are different for different vehicle types. Before the second module, non-panel regions, e.g., vehicle windows or other reflecting area, were eliminated. Second, the color histograms in an ROI were classified using a trained classifier. A manifold learning algorithm, called nearest feature line embedding (NFLE) [18], reduces the dimensionality of color features for reducing the illumination impacts. NFLE discovers the intrinsic manifold structure from the data by considering the relationship among samples. Not only the dimensions are reduced, but also the illumination impacts are reduced. Finally, the vehicle colors were determined by the dominant colors in the ROI. Seven colors, e.g., red, yellow, blue, green, black, white, and gray, were identified in this study.

The contributions of this work are twofold and briefly summarized in the following: The taillight detection and pairing algorithms fast locate ROI candidates from video frames. Foreground objects are found without the construction of background model in various urban roads. Furthermore, vehicle types are identified to correctly locate ROI regions. Second, the proposed manifold learning method, NFLE, preserves the local structures among samples in manifold distributions. The uncollected color prototypes are linearly approximated from collected prototypes by the NFL strategy in feature spaces. Both the high dimensionality and illumination impacts are reduced under various weather conditions. Comparing with our previous work in [21], four problems have been solved in this work. First, the proposed method locates multiple valid ROIs from video frames in multi-lane. Second, the vehicle types are identified by an SVM classifier instead of a heuristic rule for accurately locating ROI. The type identification also reduces the taillight pairing errors.



**Figure 2** The proposed scheme for vehicle color classification.

Third, the fused color descriptors in multiple color spaces are adopted rather than a single descriptor in a specified color space. Illumination impacts are reduced by generating the virtual prototypes from the linear combination of collected prototypes. Fourth, manifold learning algorithm reduces the high feature dimensions to find the discriminative features. In addition, the NFL strategy is embedded into the transformation. Instead of the template matching strategy in [21], an SVM classifier has been trained for classification. The proposed method is executed on the large-scale surveillance videos in various weather conditions and the color classification performance is significantly enhanced.

The rest of this paper is organized as follows: The location of valid ROIs is presented in Section 2. Vehicle color identification using a trained classifier is given in Section 3. Some experimental results to show the validity of the proposed method are presented in Section 4. Some conclusions are given in Section 5.

## 2. Location of region of interest (ROI)

Object detection is widely used in many computer vision applications. A detector is trained from the collected training samples, and a window slides an image from left to right and top to bottom in multi-scales to identify the objects. Homogenous objects, e.g., license plates and faces, are frequently detected by a trained detector. It is a challenge to locate the ROI from various vehicle shapes using the window-based detection approach. In addition, much time is needed to check a large number of sliding windows using brute force searching. Dule et al. [22] manually assigned the ROIs on vehicle hoods from the detected foreground objects. However, automatic ROI finding is a critical issue in surveillance systems. Wu et al. [23] find the ROIs by integrating the results of background subtraction and color segmentation. An SVM-based classifier in a two-layer structure is then applied to classify the pixels in ROI. Yang et al. [5] find the ROIs by using the geometric rule-based scheme. In this study, a coarse-to-fine strategy was adopted for efficiently identifying the ROIs from various types of vehicles. Three steps, *red patch labeling*, *taillight pair matching*, and *shape feature verification*, were performed to locate the ROIs from still images. First, red patches were detected for finding possible taillight candidates using several simple thresholding rules. Then, possible ROIs were generated from the taillight pairs, and geometric rules filtered out improbable taillight pairs to determine the valid ROI areas. Compared with the positions of taillight pairs, three vehicle types, e.g., sedan type (sedan, coupe, or Hatchback), SUV type (sport utility vehicles or recreational vehicles), and truck type (caravan, pickup truck, or autotruck), were defined for color classification. After that, shape features in a specified mask were verified to identify the exact ROI.

### 2.1 Red patch labeling

Since the sunlight shines vertically on the vehicle hoods, these are prone to reflect white in image frames. These overexposure regions generated incorrect results in color classification. Therefore, the valid color regions were obtained from the back view of vehicles. The ROI identification reduced overexposure during color classification.

In our previous work [24], the red pixels of taillights were detected using color features. The  $Cr$  component of pixels possesses more discriminant power than the other components in red pixel labeling. Image pixels in color space  $RGB$  were transformed to space  $YCbCr$  using Eq. (1). Each pixel  $I(x, y)$  in space  $YCbCr$  was next classified as a red pixel if satisfying the conditions in Eq. (2):

$$\begin{bmatrix} Y \\ Cr \\ Cb \end{bmatrix} = \begin{bmatrix} 0.257 & 0.504 & 0.098 \\ -0.148 & -0.291 & 0.439 \\ 0.439 & -0.368 & -0.071 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} \quad (1)$$

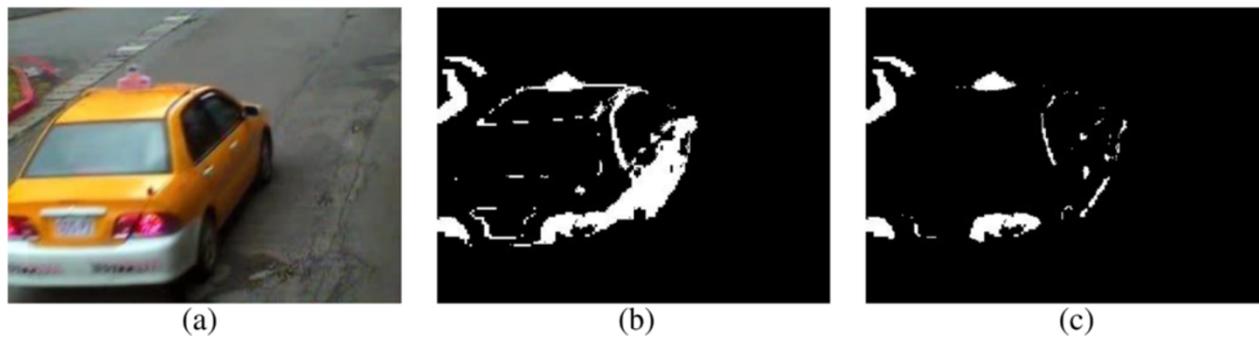
$$I(x, y) = \begin{cases} 255 & \text{if } \theta_{Cr} \leq Cr(x, y) \text{ and,} \\ & \theta_{Cb1} \leq Cb(x, y) \leq \theta_{Cb2}; \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

Here, threshold values  $\theta_{Cr}$  and  $[\theta_{Cb1}, \theta_{Cb2}]$  were manually assigned for channels  $Cr$  and  $Cb$ , respectively. Since yellow and red pixels in space  $YCbCr$  possess similar features, yellow pixels can frequently be misclassified as red pixels using the simple rules, i.e., the loose criteria in Eq. (2) and the pre-defined thresholds. Since yellow and red pixels were mixed on the  $CbCr$  plan, false alarms were frequently generated (see Figure 3b). When a yellow taxi was checked, a lot of yellow pixels were misclassified as red pixels. An SVM-based classifier was further trained to eliminate the false alarms as shown in Figure 3c.

The coarse-to-fine strategy was adopted in red patch labeling for efficiency. First, image pixels in space  $RGB$  were converted to space  $YCbCr$ . Pixels with features ( $Cr$  and  $Cb$ ) were checked using the simple criteria. If they satisfied the simple criteria, eight neighbors around them were further extracted and encoded to generate the 18-dimensional vectors in the SVM-based classification [25]. Based on the coarse-to-fine strategy, a large number of non-red pixels were filtered out using the simple criteria, and few patches of 3 by 3 were verified with the complex SVM-classifier.

### 2.2 Taillight pair matching

A binary image was obtained from the previous step, and red patches were labeled from the binary image. Three geometrical rules were tested for selecting the candidate pairs. The first one was to filter out too small or too large regions from the labeled results. Two thresholds  $\theta_{s1}$  and  $\theta_{s2}$  were manually set to reserve the red patches whose sizes fell within the range  $[\theta_{s1}, \theta_{s2}]$ . Furthermore, if a labeled patch was larger than  $\theta_{s2}$  and its compactness was high, a red vehicle was identified. The density of labeled



**Figure 3** The red patch labeling for a taxi. (a) A taxi image. (b) The misdetected red pixels using simple rules. (c) The labeled results verified by an SVM classifier.

points within a bounded box was calculated for a specified patch  $P$  in the second rule. If the density of a bounded box was greater than  $\theta_d$ , patch  $P$  was reserved; otherwise, it was eliminated. The last rule considered the angle of taillight pairs.  $C_2^N$  possible pairs were checked, where  $N$  was the number of red patches surviving from the first two rules. The impossible taillight pairs were filtered out by the line angle and length criteria. The line angle of two patch centers  $C_i(x_i, y_i)$  and  $C_j(x_j, y_j)$  was calculated, i.e.,  $\theta = \tan^{-1}(y_j - y_i / x_j - x_i)$ . A taillight pair was chosen; if its length was within a range  $[l_1, l_2] = [50, 150]$  and its line angle was close to zero, e.g., within  $-5^\circ$  to  $+5^\circ$ . As abovementioned, the candidate red patches are detected by Section 2.1; then, each candidate red path can be paired as taillight pair by using geometric rules which is described in this section. Thus, the valid ROI region in the image frame is set with a fix region according to taillight pair.

### 2.3 Type classification using shape features

Using the geometrical rules in the previous section, the remaining pairs were regarded as the taillight pairs. However, two problems occurred especially in the multi-lane cases. First, the left taillight of a vehicle matched the right taillight of another vehicle as shown in Figure 4a,b. The second problem was that a fixed ROI locating rule was not suitable for all vehicle types. Different vehicle types need different ROIs for color classification. Figure 4c,d illustrates the improper ROI problems for trucks and SUVs using the generation rule for sedans. In Figure 4c, the invalid ROI, the chassis shadow and the partial ground, was generated using the same rule of sedans. On the other hand, the windshield of SUV was located for the ROI as shown in Figure 4d. Thereafter, vehicle type classification is needed for locating the valid ROIs. Thus, vehicle type identification was needed to generate the desired ROIs using complex features, e.g., shape features. Three issues were considered for vehicle type identification: *checked region determination*, *feature representation*, and *classifier design*.

The checked window for type classification was determined by the taillight pair. The reference length of one taillight pair  $d$  determined the size of a verified region  $d \times d$  as shown in Figure 5. This region was normalized to a fixed size of 64 by 64. On average, fewer than ten pairs were checked in an image frame. This region was next represented by the histogram of oriented gradients (HOG) [26]. The HOG features of 1,764 in a verified ROI were extracted for classification. An SVM classifier of multi-class was trained using HOG features for vehicle type classification. One thousand two hundred image samples of four classes, i.e., sedan type, SUV type, truck type, and non-vehicle type, were collected for training. Figure 6a shows several illustrations for training the multi-class SVM classifier. In the training set, sedans, coupes, or Hatchbacks are classified as the 'sedan' type. Similarly, sport-utility vehicles and recreational vehicles are classified as the 'SUV' type; and caravans, pickup trucks, and autotrucks are classified as the 'truck' type. Moreover, the samples of 'non-vehicle' type are also collected for training. After the vehicle type classification, the ROI for a specified vehicle type was determined by its corresponding rule. For example, the ROIs of SUVs were located at the bottom of taillight pairs, while the ROIs of trucks were located at the top of taillight pairs. The ROIs of sedan type, SUV type, and truck type were determined, drawn by the blue rectangles as shown in Figure 6b-d. The color features in these ROIs are classified in the color classification below.

### 3. Eigenspace-based color classification

As is generally known, the classification process is composed of three steps in pattern recognition (PR): feature representation, feature discriminant analysis, and classifier design. In this study, a color space fusion plus dimension reduction scheme was designed for color classification. Color histograms in ROIs were extracted, reduced, and classified by a multi-class SVM classifier.

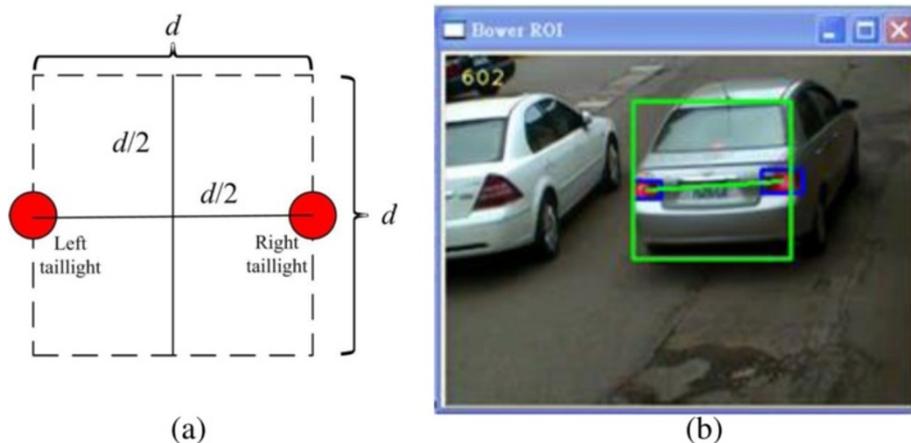


**Figure 4** Mismatched taillight pairs and invalid ROIs. (a) and (b) is the mismatched taillight pairs. (c) and (d) is the invalid ROIs using the rules for sedans.

### 3.1 Feature representation: linear color feature combination

Many color spaces [27], e.g., *RGB*, *HSV*, *HLS*, *CIELab*, *YCrCb*, ..., etc., were explored in color classification. The choice of a color space was critical in identifying vehicle colors. Though color spaces were interpreted in many different models, no color space could be regarded

as a universal space. A selection and fusion scheme proposed by Stokman et al. [14] combines many color spaces, and the better results are achieved. Twelve color components were weighted and summarized with a linear programming method in [15]. In this study, a color histogram was extracted from the color pixels in an ROI



**Figure 5** The verified window drawn by a green rectangle. (a) The rule of checked window. (b) The green rectangle is used to identify vehicle type.



**Figure 6 Vehicle type training samples and ROIs location.** (a) The training samples of four vehicle type classes. (b-d) The ROIs of sedan type, SUV type, and truck type drawn by the blue rectangles.

for feature representation. A window of  $w$  by  $w$  slides the ROI from left to right and top to bottom; here,  $w = 20$ . The color component for each pixel was quantized into 256 levels. The statistical histogram of length 256 was obtained. Eighteen histograms of six color spaces, i. e., *RGB*, *CIELab*, *YCrCb*, *HSV*, *Luv*, and *HLS* were combined to represent the ROI's color. The feature vector of length  $4,608 = 256 \times 6 \times 3$  in this window was extracted as shown in Figure 7. This descriptor with high dimensions was reduced to a lower dimensional space for reducing illumination impacts.

A toy example is given in the following. In order to show the reconstruction of intrinsic manifold structure using eigenspace approaches, 1,400 patches of size 20 by 20 of seven color classes are collected for training. These patches are drawn by their corresponding colors in six color spaces *RGB*, *YCrCb*, *HSV*, *HLS*, *Luv*, and *CIELab* as shown in Figure 1a-f, respectively. In these figures, cross marks represent the samples of white color due to the white papers, and circle marks of pink color represent the

samples of gray color for the clear representation. These samples of seven classes are heavily mixed due to the illumination factors. The features in six color spaces are catenated to generate a new vector of length 4,608. The same class patches under various illumination conditions are represented in different colors, e.g., the ROI templates of dark yellow, general yellow, and bright yellow colors as shown in Figure 8. They are classified as 'class yellow'. According to the consequences in Li's approach for face recognition [28], they claimed that 'the feature line approximates variants of the two prototypes under variations in pose, illumination, and expression'. A linear model virtually interpolates an infinite number of prototypes of the class in feature spaces. Similarly, the features of general yellow color are obtained from the linear combination of features of dark yellow and bright yellow features.

**3.2 Feature discriminant analysis: dimension reduction (DR)**  
 LPP [17] and LLE [16] are two popular manifold learning algorithms which are applied to keep the manifold

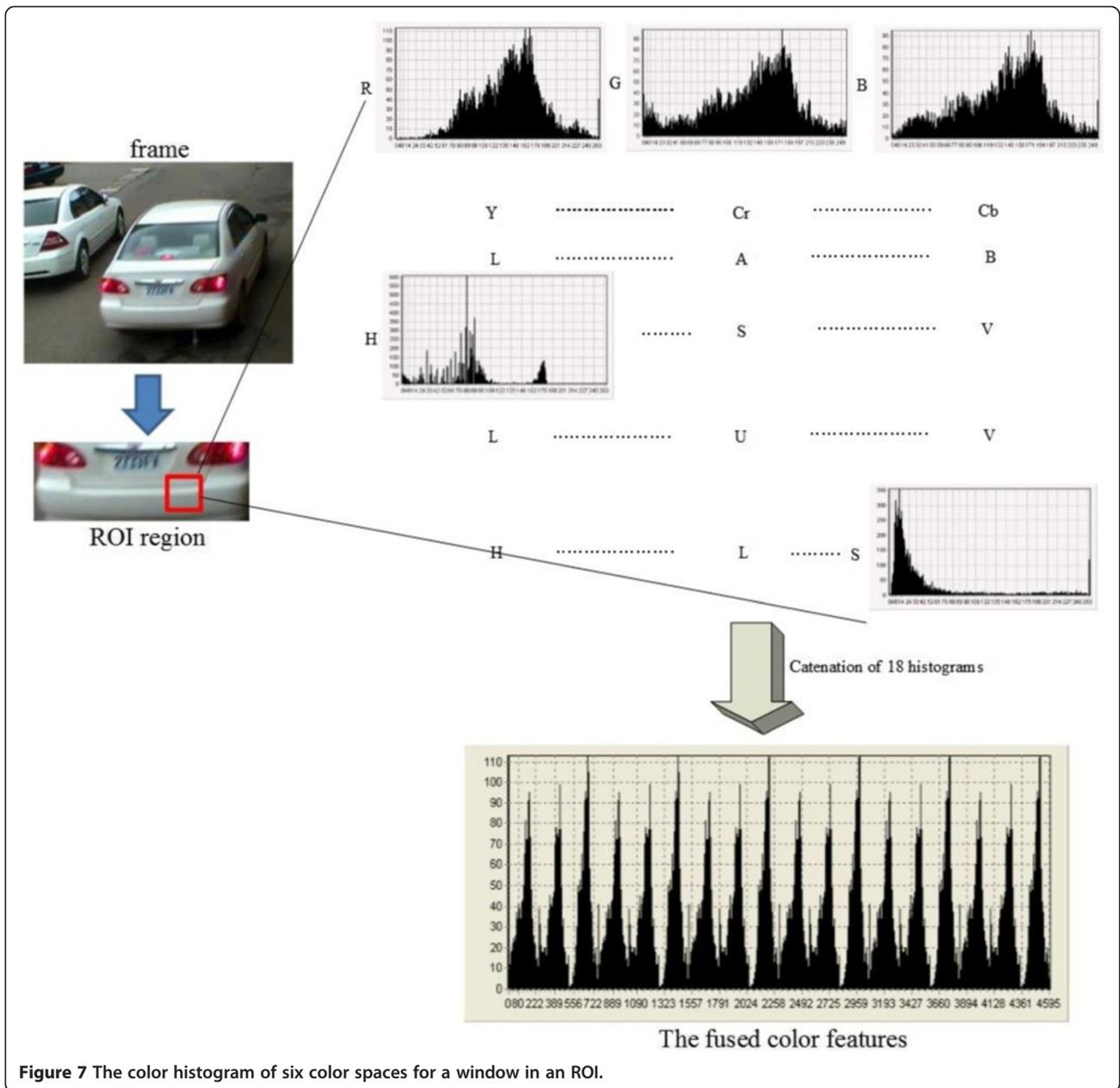


Figure 7 The color histogram of six color spaces for a window in an ROI.

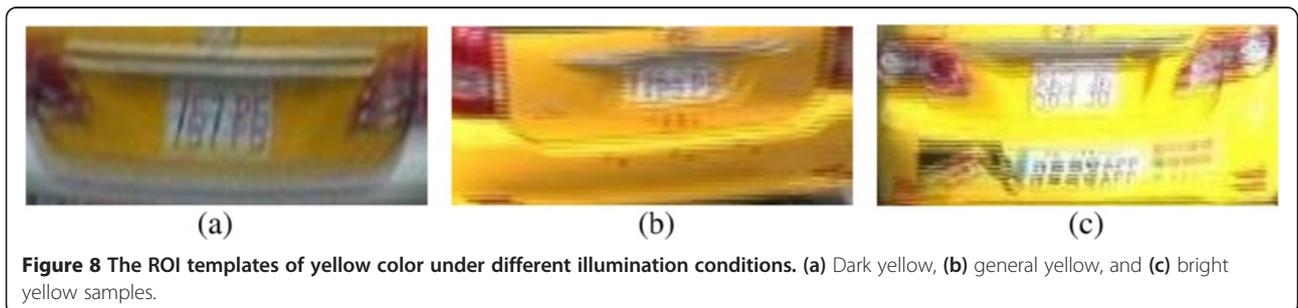


Figure 8 The ROI templates of yellow color under different illumination conditions. (a) Dark yellow, (b) general yellow, and (c) bright yellow samples.

structure of samples. They try to minimize the objective functions to obtain the best transformation for DR. Though their objective functions are represented in different forms, their goals are the same. Moreover, the two objective functions have been proved to be equivalent in [29]. Both of them were represented in a Laplacian matrix form. The best transformation matrix was composed of the eigenvectors with the smallest corresponding eigenvalues by solving the general eigenvalue decomposition problem. Two neighboring samples in a high-dimensional space were neighbors in a low-dimensional space in the LPP-based minimization.

The NFLE transformation [18] was a new manifold learning method based on the point-to-line (p-2-l) distance measurement which was originated from the nearest linear combination (NLC) approach [28]. The NFLE method tried to find a discriminative subspace for reducing the histogram dimensions of six color spaces for feature extraction. The reduced subspace had more discriminative power than any one specified color space. Before the classifier training, the PCA and NFL processes were used to reduce the feature dimensions. The NFLE transformation is briefly described below.

Given  $N$  training samples  $x_1, x_2 \dots x_N \in R^D$  constituting  $C$  classes, new samples  $y_1, y_2 \dots y_N \in R^d$ ,  $m < M$ , were obtained in a low-dimensional space with a linear projection  $y_i = W^T x_i$ . Consider a specified point  $y_i$  in the transformed space; the distance from point  $y_i$  to a feature line was defined as  $\|y_i - f_{m,n}(y_i)\|$ , in which  $f_{m,n}(y_i)$  was the projected point of line  $L_{m,n}$ . Point  $f_{m,n}(y_i)$  is a virtually constructed point which is generated by points  $y_m$  and  $y_n$ . In the training phase, it is a hard task to collect all possible prototypes in various outdoor illuminations. The NFL strategy creates more virtual points to efficiently represent the vehicle colors. For example,  $y_m$  and  $y_n$  can be regarded as the samples of bright yellow and dark yellow, respectively. Point  $f_{m,n}(y_i)$  is considered as the general yellow sample by linearly weighting samples  $y_m$  and  $y_n$ .

The scatter computation of feature points to feature lines were calculated and embedded during the discriminant analysis phase.  $C_2^{N-1}$  possible generated lines for point  $y_i$  were more than  $N - 1$  points in the conventional point-to-point (p-2-p) methods, e.g., LPP and LLE. Thus, the p-2-l method retained much more scatter information than the conventional p-2-p-based methods. In addition, the NFL metric was embedded into the transformation through the discriminant analysis phase instead of in the matching phase [22]. The objective function of NFLE is defined as follows:

$$W^* = \arg \min_W \sum_i \sum_{m \neq n} \left\| y_i - f_{m,n}(y_i) \right\|^2 w_{m,n}(y_i) \quad (3)$$

Here, weight  $w_{m,n}(y_i)$  represents the connectivity strength for point  $y_i$  and line  $L_{m,n}$ . Since the objective function in

Eq. (3) was represented as a Laplacian matrix, the topology of samples could be preserved. Furthermore, the within-class scatter matrix  $S_w$  was calculated as follows:

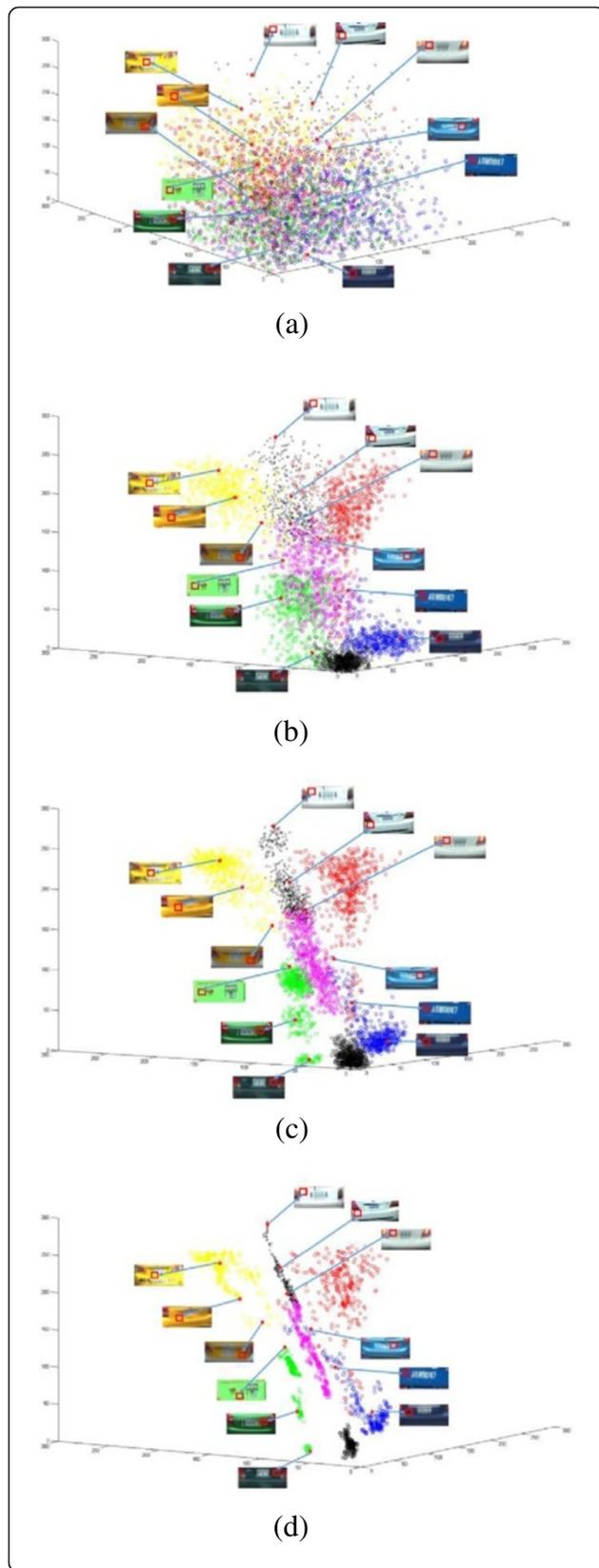
$$S_w = \sum_{p=1}^C \left( \sum_{\substack{x_i \in C_p \\ L_{m,n} \in F_{K_1}(x_i, C_p)}} (x_i - f_{m,n}(x_i))(x_i - f_{m,n}(x_i))^T \right), \text{ and} \quad (4)$$

where  $F_{K_1}(x_i, C_p)$  represents the  $K_1$  nearest feature lines within the same class  $C_p$  of point  $x_i$ . The within-class scatter matrix  $S_w$  was minimized to obtain the projection matrix  $W^*$ , which consisted of the eigenvectors with the corresponding smallest eigenvalues. In general, since the NFL metric generalized the representative capacity of prototypes during the discriminant analysis phase, the NFLE preserved much more information than the conventional p-2-p-based methods. More details are given in [18].

One thousand four hundred feature points are reduced to a new space of dimension 3 after the projecting transformations, e.g., PCA, PCA plus LDA, PCA plus LPP, and PCA plus NFLE. Similar to Figure 1, 1,400 transformed samples of seven classes are drawn in Figure 9 by their corresponding colors. In this figure, red stars denote the patch windows (red rectangles) of an ROI which are projected onto the new spaces by the transformations. In Figure 9c,d, a specified sample could be represented by the linear combination of other samples of the same class in the transformed space. For example, the sample of general yellow color is represented by the linear combination of samples with the dark and bright yellow colors. Points of the same class are as close as possible, while samples of different classes should be separated as far as possible. In summary, manifold learning methods, LPP or NFLE, discover the more intrinsic manifold structure than the global eigenspace methods, PCA and LDA. Besides, manifold learning algorithms not only reduce the feature dimensions but also preserve the sample relationship of the same classes under various illumination conditions.

### 3.3 Classifier design: 1-NN, SVM, and SRC

When discriminant features in an ROI were extracted by the DR process, the classifier was trained to classify the ROI's colors. In this study, the nearest neighbor classifier (1-NN), one-vs-all SVM classifier, and sparse representation classifier (SRC) [30] are adopted for color classification. The seven most used different classes of color in commercial vehicles were chosen for classification. They included red, yellow, blue, green, black, white, and gray. Classified with the trained classifiers, the vehicle colors were determined in the following steps. A valid ROI was obtained for color classification in Section 2. A window of 20 by 20



**Figure 9** The sample distributions of seven color classes after various eigenspace transformations. (a) PCA transformation, (b) PCA + LDA transformation, (c) PCA + LPP transformation, and (d) PCA + NFLE transformation.

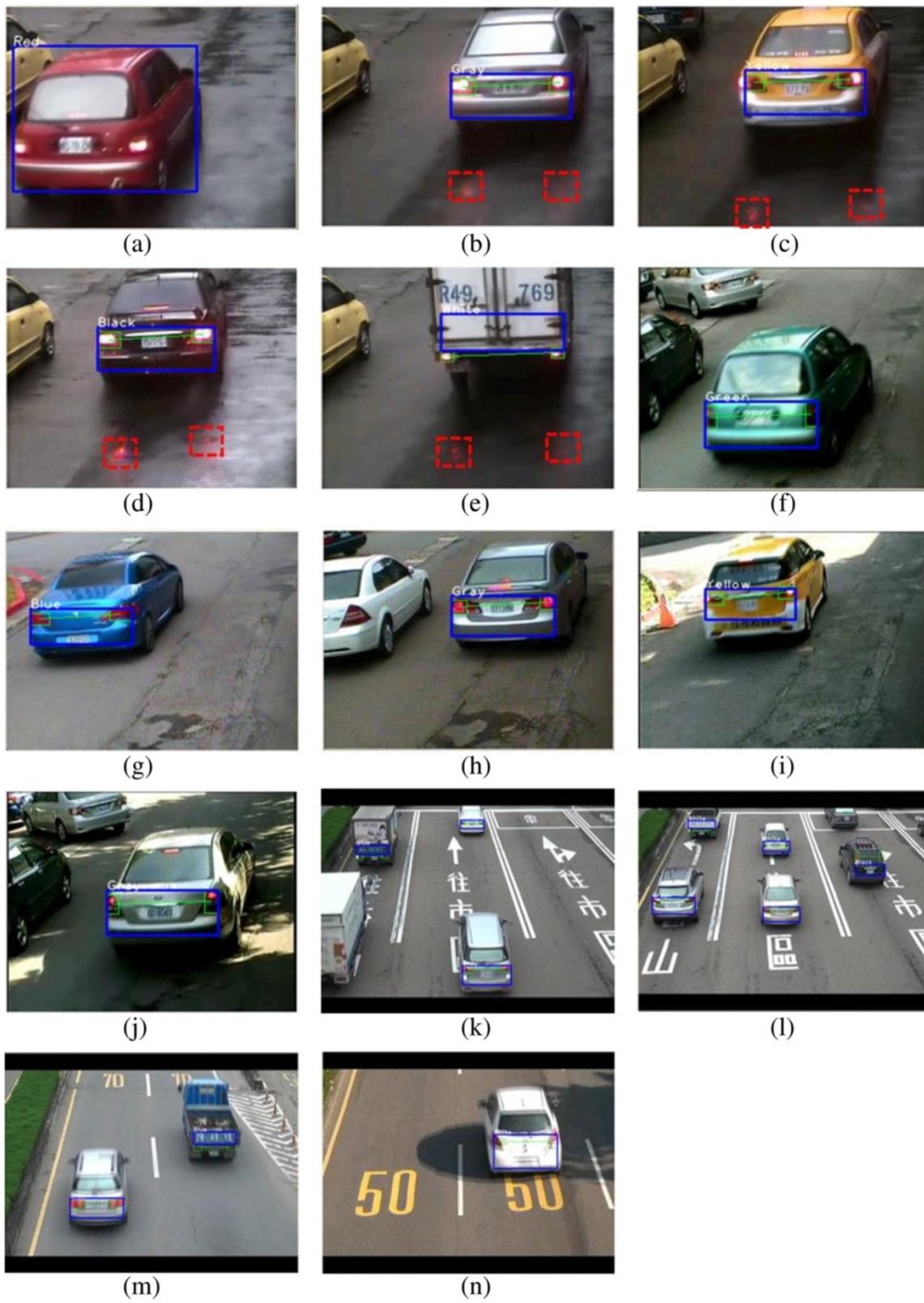
slides the ROI from left to right and top to bottom. The color histogram of six color spaces in this window was generated and reduced to a lower dimensional vector in feature discriminant analysis. This reduced vector was classified with the trained SVM classifier or the 1-NN classifier to determine the color class. Finally, the classification results of sliding windows in an ROI were counted, and the ROI color was determined using the voting strategy.

SRC is a discriminative nature of sparse representation for classification. The designed SRC classifier in this study is briefly described as following: (1) Similar to the SVM classifier, a window of 20 by 20 slides the ROI from left to right and top to bottom. The color histograms of each sliding window in six color spaces are generated. The reduced features in the low-dimensional space are obtained by the eigenspace transformation. (2) After feature extraction, the  $N$  training samples  $A = [A_1, A_2, \dots, A_7]$  are collected for seven color classes, in which  $A_i = [a_{i,1}, a_{i,2}, \dots, a_{i,n_i}] \in \mathbb{R}^{d \times n_i}$  is the sample of class  $i$  and  $N = \sum_{i=1}^7 n_i$ . The columns of  $A$  are normalized to be unit  $\ell^2$  norm. (3) When a test sample  $y \in \mathbb{R}^d$ , a reduced vector of fusing color histogram on a sliding window, is verified, solve the  $\ell^1$ -minimization problem via a primal-dual algorithm for linear programming based on [31,32]:  $\hat{x}_1 = \arg \min_x \|x\|_1$ , subject to  $Ax = y$ . The residual errors for each class are calculated:  $r_i(y) = \|y - A\delta_i(\hat{x}_1)\|_2$ ,  $i = 1, 2, \dots, 7$ , where  $\delta_i$  is the characteristic function which selects the coefficient associated with the  $i$ -th class. The test sample  $y$  is classified as the  $i$ -th class, if the residual error  $r_i(y)$  is the smallest. Finally, the color classification results of sliding windows in an ROI were counted, and the ROI color was determined using the voting strategy.

#### 4. Experimental results

In this section, the experiments conducted to show the performance of the proposed method are discussed. A stationary CCD camera was set up on the shoulder of roads. Eighteen video clips were captured in various weather conditions, e.g., in sunny, cloudy, or rainy. Due to the varied outdoor illumination from different weather conditions, the captured images are illustrated as shown in Figure 10.

Fifteen video clips of 320 by 240 were grabbed from the scenes in a single lane as shown in Figure 10a-j. On the other hand, three clips of an image of 720 by 480 were also grabbed from the senses in multi-lanes as shown in Figure 10k-n. The ROIs of the vehicles were incomplete when a vehicle moved in or out of the image frame. Two lines were set to obtain the complete rear view of vehicles.



**Figure 10** The vehicle color classification results in single and multiple lanes. (a-j) The classification results in a single lane. (k-n) The classification results in multiple lanes.

The data set consisted of 18 clips for evaluation. More than 42,000 vehicles were segmented from the clips. The ground truths (GTs) of taillight locations, vehicle types, and color classes were manually labeled in the data set. In this study, the locations of ROIs in a still image were first identified. The color histograms of blocks in an ROI were next classified for color classification. For a specified block in a valid ROI, a histogram-based feature vector of length 4,608 was extracted from the 20 by 20 window in six color spaces. This vector with high dimensionality was reduced to the lower dimensional subspace by PCA plus NFLE. The block color was classified by an SVM classifier. The ROI's color was determined using voting of the classification results. Two experiments were conducted to evaluate the performance of ROI location and color classification. The proposed method was implemented in a PC-based machine with a CPU model i7-920 in 2.67 GHz using the Microsoft Visual C++ 2008 and OpenCV 2.1 tool kits.

#### 4.1 ROI location

Before color classification, the ROI for each vehicle had to be accurately located. To achieve this goal, a coarse-to-fine strategy was adopted to locate the valid ROIs from still images. In the first experiment, three results were reported showing the performance of ROI location.

**Table 2 The confusion matrix of classification using texture features**

	Sedan	SUV	Truck	Non-vehicle	Accuracy rates (%)
Sedan	31,509	712	0	0	97.8
SUV	245	5,796	0	0	95.9
Truck	0	0	4,179	204	95.3
Non-vehicle	863	587	2,547	257,657	98.5

First, simple rules in Eq. (2) were employed to label the red patches. An SVM classifier was further trained to classify the confused pixels: yellow and red. Initially, the GT regions were manually labeled. The accuracy rates of red patch labeling were calculated by comparing the detected regions with the GT ones. When the overlapping region was larger than 1/10 of the corresponding GT region, the detected red patch successfully hit the taillight. The average hit rate of red patch labeling was more than 98% for 18 evaluation clips, as shown in Table 1. Using the simple rules with loose thresholds, taillight patches were labeled with high accuracy rates. In video clip 14, a low labeling rate was achieved because images were captured in the gradually dimming light of dusk. A lot of noise was generated in image frames due to the white balance function of cameras.

**Table 1 The correct rates of taillight pairs for the evaluation data set (%)**

Video clips	Times	Weather <sup>a</sup>	Number of red patches (ground truth)	Number of hit patches	Hit rates (%)	The average numbers of taillight pairs
1	8:00–10:30	Sun: through cloud/haze	424	424	100.00	3.91
2	13:30–15:00	Overcast sky	476	476	100.00	4.68
3	7:30–9:00	Overcast sky	468	468	100.00	3.05
4	9:30–11:00	Overcast sky	182	182	100.00	4.82
5	15:30–17:00	Partly cloudy sky	1,144	1,144	100.00	4.42
6	14:30–16:30	Sun: through cloud/haze	1,104	1,104	100.00	4.53
7	9:30–14:30	Outdoor shade areas	1,762	1,762	100.00	2.79
8	10:00–15:00	Outdoor shade areas	3,222	3,094	96.02	2.67
9	13:30–16:00	Outdoor shade areas	2,790	2,772	99.35	2.46
10	15:00–17:00	Sun: through cloud/haze	894	894	100.00	2.60
11	16:00–17:00	Partly cloudy sky	188	188	100.00	4.17
12	13:00–15:30	Sun: through cloud/haze	2,318	2,300	99.27	2.74
13	15:00–17:00	Overcast sky	2,038	2,038	100.00	2.54
14	15:30–18:00	Sun: through cloud/haze	1,810	1,620	89.5	3.54
15	9:00–16:00	Rainy day	10,600	10,452	98.60	3.86
16 (multi-lanes)	8:00–15:00	Daylight(sun + sky) and partly cloudy sky	17,094	16,924	99	6.89
17 (multi-lanes)	8:00–15:00	Daylight(sun + sky) and partly cloudy sky	18,850	18,556	98.45	8.03
18 (multi-lanes)	10:00–17:00	Daylight(sun + sky) and partly cloudy sky	21,160	20,892	98.73	11.45
Total			86,524	85,290	98.57	—

<sup>a</sup><http://www.3drender.com/glossary/colortemp.htm>.



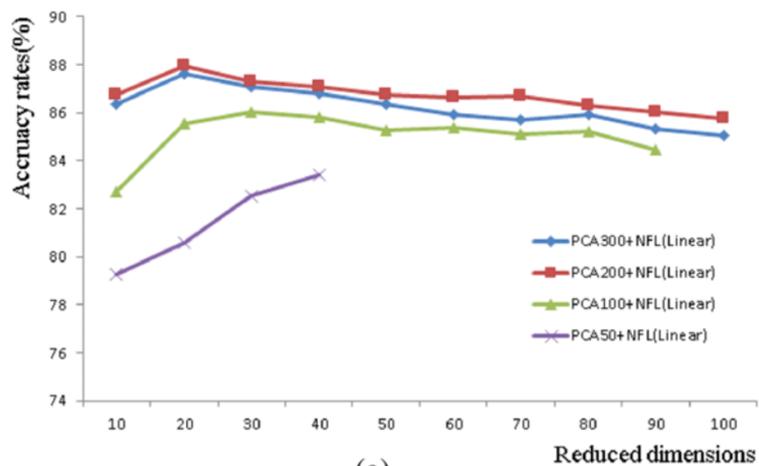
**Figure 11** The vehicle type misclassified examples. (a) Sedans were misclassified as type 'SUV'. (b) SUVs were misclassified as type 'sedan'. (c) Trucks were misclassified as the 'non-vehicle' regions. (d) The non-vehicle regions were misclassified as type 'sedan'. (e) The non-vehicle regions were misclassified as type 'SUV'. (f) The non-vehicle regions were misclassified as type 'truck'.

The geometric rule-based filter was then employed to determine the taillight pair candidates. All taillight pairs were reserved in the second results. On average, less than five pairs in a single lane and less than 12 pairs in multi-lanes were needed for the further process as listed in the last column of Table 1.

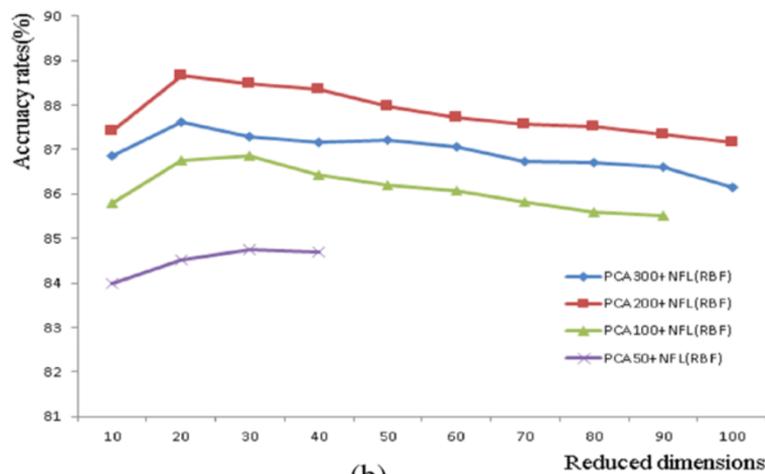
The third results were the ROI verification using HOG features. HOG is an efficient feature descriptor for object representation because it is robust to illumination and geometric distortion. Not only were both the vehicle and non-vehicle regions verified, but the vehicle types were also classified. Four classes, sedan, SUV, truck, and non-vehicle, were identified for vehicle type verification and classification. Three hundred images for each class were collected for training. A multi-class SVM classifier was trained using HOG features for vehicle type classification. In testing, more than 300,000 ROIs, including 32,221 sedans, 6,041 SUVs, 4,383 trucks, and 261,654 non-vehicle regions, were classified to determine the vehicle types.

The accuracy rates and confusion matrix are tabulated in Table 2 for 18 video clips. The correct rates for classes, sedan, SUV, truck, and non-vehicle region, were 97.8%, 95.9%, 95.3%, and 98.5%, respectively. The proposed method could effectively identify the valid vehicle ROIs in different types and multi-lanes. In addition, the proposed method is robust to the weather conditions. For an example, the reflected regions of taillights, the dark red regions in Figure 10b-e, are efficiently filtered out by the proposed method in rainy days.

In addition, some misclassification results are also given in Figure 11. In Table 2, the regions of sedan and SUV were misclassified due to the similar shapes as shown in Figure 11a,b. The regions of trucks were misclassified as the non-vehicle regions because the planar plates of the truck were similar to the ground regions as shown in Figure 11c. Similarly, the non-vehicle regions were frequently misclassified as the truck class, as shown in Figure 11d. On the other hand, the non-vehicle regions, the



(a)



(b)

**Figure 12** The accuracy rates of the NFLE-based feature reduction and a trained SVM classifier. (a) The linear kernel SVM classifier. (b) The RBF kernel SVM classifier.

false alarms, were misclassified as the regions of sedan and SUV were generated due to cluttered backgrounds as shown in Figure 11e,f. From the experimental results, more than 98.3% accuracy rate was achieved using HOG features and SVM classifiers. Therefore, 42,645 vehicles' ROIs survived from 43,262 vehicles after the ROI location step.

#### 4.2 Color classification

After the ROIs of vehicles in the rear view were located, seven most used color classes were classified, including red, yellow, blue, green, black, white, and gray. The color histograms of blocks of 20 by 20 were extracted from an ROI. Since the dimensionality of the color histogram was very high, i.e., 4,608, PCA was performed to find the best representation for avoiding the small-sample-size problem. The best discriminative projections were next found by the NFLE method. The color histograms of 4,608 were reduced to the vectors of dimensions  $D_{PCA}$  and  $D_{NFLE}$  by PCA and NFLE, respectively. Two thousand one hundred samples of seven classes were collected to train the multi-class SVM classifiers, and the linear and RBF kernel functions are used during the training. Each block was classified to determine its color. All classification results of blocks in an ROI were counted and the ROI's color was determined by a voting strategy. To evaluate the proposed method, more than 42,000 vehicles from 18 video clips were identified according to color features. Figure 10 shows the classification results of the testing video clips in a single-lane and in the multi-lane cases, respectively. The red patches and ROIs were drawn by the green and blue rectangular boxes. They also show that the proposed method was effectively performed on urban roads in various weather conditions. Moreover, the classification rates for various reduced dimensions are shown in Figure 12. Three curves represent the classification results in which  $D_{PCA}$  is the reduced dimensions of 300, 200, and 100 by PCA, and  $D_{NFLE}$  is the reduced dimension from 10 to 100 by NFLE. From the results in Figure 12, the classification results were very similar for these three curves. After DR, the SVM classifiers with a linear kernel function and an RBF kernel function were trained for color classification. The best classification rates are 87.93% and 88.67% for the linear SVM classifier and the RBF kernel SVM classifier, respectively. The RBF kernel SVM classifier with 1,594 support vectors obtains the best classification rate which  $D_{PCA}$  is 200 and  $D_{NFLE}$  is 20. Two parameters ( $c$  and  $\gamma$ ) are 2.0 and 0.0078125 which were obtained from the LIBSVM tool kit [25]. The other parameters were initialized as the default values for training the classifier. The best accuracy rates for each video clip are tabulated in Table 3, and the average classification rate for the 18 clips was 88.67% by the SVM classifier with a RBF kernel function.

To show the performance of color space fusion plus the DR scheme, two experiments were implemented for comparison. First, the original histograms of color spaces RGB, LAB, HSV, and fused space were fed to the SVM classifier for training and testing. According to the results in Table 4, the fused space outperformed the other color spaces. In the second experiment, the original histograms were reduced to new vectors of dimensions 200 and 20 by PCA and NFLE before the SVM training. The reduced vectors were classified by the trained classifier. The accuracy for all color spaces was improved. These implied the discriminative features had been extracted from feature discriminant analysis.

Similarly, the confusion matrix of color classification is tabulated in Table 5 for the testing video clips. The correct rates for color classes, red, yellow, blue, green, black, white, and gray, are 91.34%, 93.73%, 90.34%, 91.62%, 90.17%, 85.22%, and 87.8%, respectively. Illumination impacted the classification performance is given in Table 5, especially with classes 'black', 'white', and 'gray'. The worst results occurred at class 'white' in Table 5. The samples in class 'white' were misclassified as classes 'gray' and 'black' at nightfall. Similarly, the samples in class 'black' were misclassified as classes 'gray' and 'white' due to the sunlight. Several misclassification cases are given in Figure 13. The vehicle in Figure 13a was misclassified as 'black' due to the dark red pixels. The misclassification for Figure 13b,c occurred

**Table 3 The classification rates using the SVM classifier**

Video clips	Correct classification	Number of vehicles	Accuracy rates (%)
1	191	212	90.01
2	209	238	87.82
3	196	234	83.76
4	78	91	85.71
5	501	572	87.59
6	498	552	90.22
7	779	881	88.42
8	1,355	1,547	87.59
9	1,155	1,386	83.33
10	391	447	87.47
11	94	94	1
12	956	1,150	83.13
13	835	1,019	81.94
14	749	810	92.47
15	4,551	5,226	87.08
16 (multi-lane)	7,583	8,462	89.61
17 (multi-lane)	8,266	9,278	89.09
18 (multi-lane)	9,428	10,446	90.25
Average accuracy rate		88.67	

**Table 4 The comparison of accuracy rates for various color spaces**

Experiments	Color space	Histogram dimensions	Reduced dimensions (PCA/NFL)	Accuracy rates (%)
I	RGB	768	Nil	73.88
	LAB	768	Nil	77.79
	HSV	768	Nil	79.31
	Six color spaces	4,608	Nil	81.98
II	RGB	768	200/20	75.66
	LAB	768	200/20	82.15
	HSV	768	200/20	81.63
	Six color spaces	4,608	200/20	88.67

because of the bumper color. The misclassification results as given in Figure 13g,h. These were generated from the illumination impacts. Most misclassification occurred at classes ‘white’ and ‘gray’. In addition, the performance of this system was 18 frames per second.

In order to show the effectiveness of the proposed method, several state-of-the-art algorithms [3-5,22,23] are implemented for the comparison. Color histogram-based features are widely used in color classification. Bin quantization is the simplest skill for DR in many papers. Kim et al. [3,4] quantize the color bins in space *HSI*. The color histograms of lengths 360 and 128 are next classified by an SVM classifier and the 1-NN classifier. Dule et al. [22] list ten possible histograms for classification. These ten histograms are evaluated and randomly fused to find the best combination, i.e., *HS-SV-ab-La-Lb-gray*. The combined histogram of length 328 is classified by a neural-network classifier. Yang et al. [5] designed a two-layer classifier: *HS* color histogram for color classification in layer one and normalized *RGB* features for the block-gray-white classification in layer two. A two-stage classifier is proposed for color classification in [11,23]. Color (i.e., red, yellow, blue, and green) and monochrome (i.e., black, gray, and white) classes are first classified in the first stage. In stage two, different features are classified by two SVM-based classifiers for color and monochrome classes, respectively. Wu et al. [23] use color features on channels *HS* in stage 1. The features on channels *HV* and *SV* are respectively classified for the four color and

the three monochrome classes in stage two. On the other hand, Hsieh et al. [11] construct a Gaussian mixed model (GMM) for color/monochrome classification in stage one. Four color classes and three monochrome classes are identified by two trained SVM classifiers. Features in color space *Lab* plus features in normalized space *RGB* are classified for four color classes, and features in normalized space *RGB* are classified for three monochrome classes. The configurations for the compared algorithms are tabulated in Table 6. The quantized bin numbers are written in the parentheses. Two thousand one hundred samples of seven classes, 300 samples per class, were collected to train the classifier, and 42,645 vehicle ROIs from 18 video clips were collected for performance evaluation in this comparison. The training and testing sets are two disjoint datasets which were independently collected<sup>a</sup>. The training samples are also collected from video clips which are captured in different locations and time of testing ones. In order to show the effectiveness, the same evaluation process has been run five times, where  $2,100 \times 5$  patches were randomly selected for training and 42,645 ROIs were evaluated by five trained classifiers. The average accuracy rates and the standard derivations are listed in Table 6. From the compared results, the proposed method outperforms the other methods.

On the other hand, several eigenspace methods for DR have been implemented for comparison. After DR, three classifiers are trained for evaluation, e.g., *k*-NN classifier,

**Table 5 The confusion matrix of color classification**

	Red	Yellow	Blue	Green	Black	White	Gray	Accuracy rates (%)
red	1,677	38	0	0	121	0	0	91.34
yellow	81	3,168	0	0	0	69	62	93.73
blue	0	0	1,357	21	57	27	40	90.34
green	0	22	13	1,312	51	0	34	91.62
black	0	0	0	0	10,232	483	632	90.17
white	0	0	0	0	68	8,399	1,389	85.22
gray	0	0	0	0	691	931	11,670	87.8



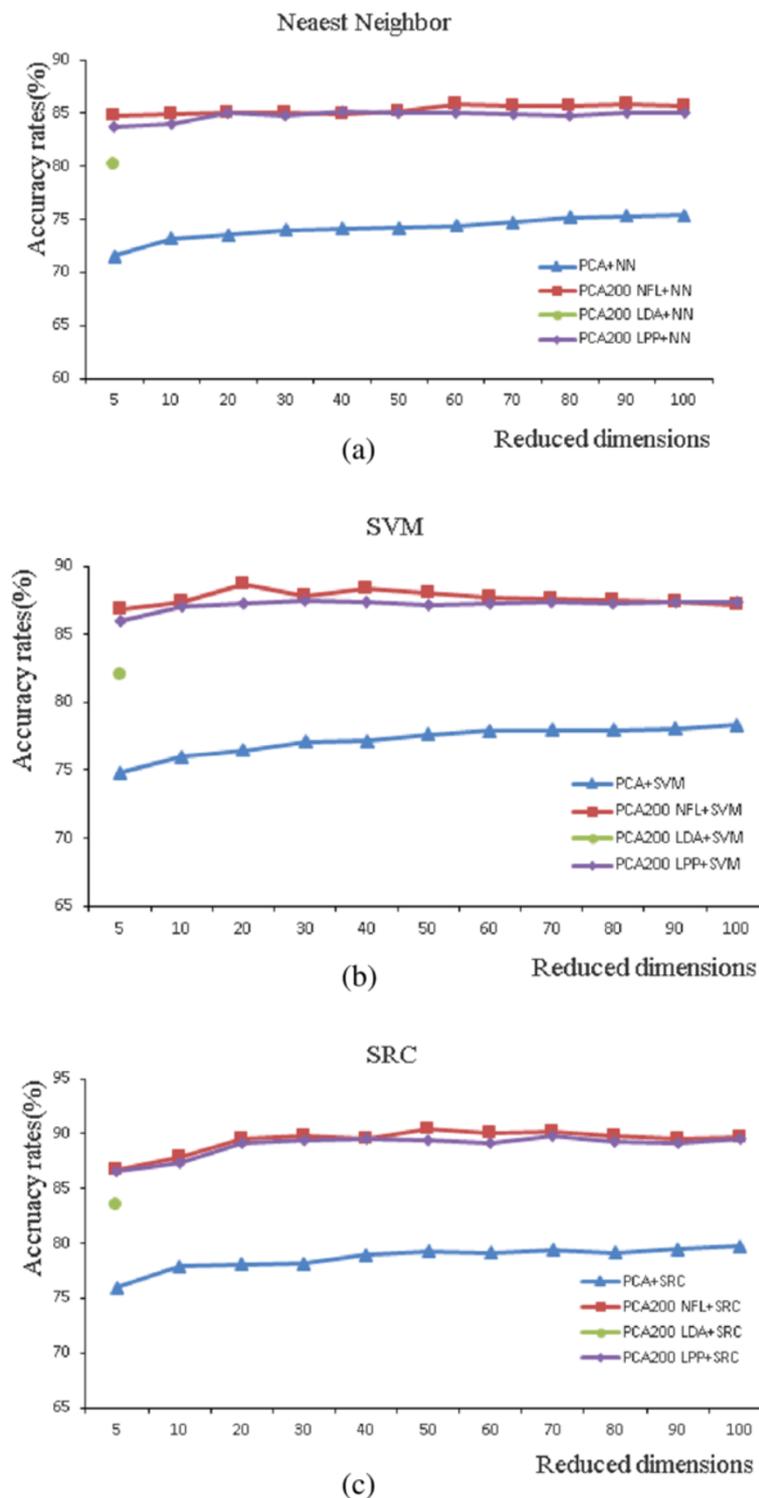
**Figure 13** The color misclassified examples. (a) Red cars were classified as black ones. (b) Yellow taxis were classified as white ones. (c) Blue cars were classified as gray ones. (d) Green cars were classified as white ones. (e) Black cars were classified as white ones. (f) Black cars were classified as gray ones. (g) White cars were classified as gray ones. (h) Gray cars were classified as white ones.

SRC, and SVM classifier with an RBF kernel function. The recognition results of three classifiers are compared as shown in Figure 14. In this experiment, the parameters for each classifier are set in the following: Value  $k$  is set as 1 in classifier  $k$ -NN, and the RBF kernel is applied

in the SVM classifier. The reduced feature dimensions are set from 10 to 100 for the DR methods PCA, PCA + LDA, PCA + LPP, and PCA + NFLE. The reduced dimension of PCA is set to be 200 for preserving more than 99% information of training samples. Since the reduced

**Table 6** The accuracy rates for the proposed method and the state-of-the-art algorithms (%)

Methods	Features	Classifiers	Average accuracy rates	Computational time (ms)
Baek [3]	$H (36)*S (10)$	SVM	73.88 ( $\pm 1.0$ )	18
Kim [4]	$H (8)*S (4)*I (4)$	1-NN	71.04 ( $\pm 1.12$ )	824
Yang [5]	Layer 1: $H (16) + S (8)$ Layer 2: normalized <i>RGB</i>	Two-layer rule-based classifier	64.03 ( $\pm 1.3$ )	34
Hsieh [11]	<i>Lab</i> + transformed <i>RGB</i>	GMM + two-stage SVM	84.77 ( $\pm 0.83$ )	58
Dule [21]	$HS (64) + SV (64) + ab (64)$ $+La (64) + Lb (64) + Gray(8)$	Neural network	76.12 ( $\pm 1.41$ )	1,210
Wu [22]	$HS (256) + HV (256) + SV (256)$	Two-stage SVM	80.66 ( $\pm 1.5$ )	33
The proposed method	Six color spaces (4,608) PCA reduction (200)	NFL (20) + SVM (RBF-kernel function)	88.18 ( $\pm 0.89$ )	18



**Figure 14** The recognition rates on 18 video clips for various feature extraction and classifiers. (a) 1-NN, (b) SVM (RBF kernel), and (c) SRC.

dimensions of LDA depend on the class number, the recognition results of LDA method only show the results of dimension 5 in the experiments. The best recognition rates and average processing time for three classifier and

four DR methods are tabulated in Table 7. The numbers in the parentheses are the reduced dimensions of the best recognition rates. The best recognition rates of classifier 1-NN for DR methods PCA, PCA + LDA, PCA +

LPP, and PCA + NFLE are 75.36%, 80.23%, 85.07%, and 85.84%, respectively. Similarly, the best recognition rates for classifiers SVM and SRC are tabulated in Table 7. From this figure, manifold learning-based DRs (LPP or NFLE) outperform the global learning-based methods (PCA and LDA). Though the recognition rates of SRC are higher than those of SVM classifier a little bit, the classification time of SRC is more expensive than SVM classifier. Practically, classifier SVM is adopted instead of SRC in designing a real-time surveillance system.

## 5. Conclusions

In this paper, a novel method is proposed for real-time vehicle color classification. Two modules: ROI location and color classification constituted the classification process. Unlike the traditional background subtraction methods which are sensitive to illumination change and the background models, the ROIs of vehicles taken from the back were located/determined using still images. To meet real-time requirements, the coarse-to-fine strategy was used in classifying from the simple pixel level to the complex region level. Six color spaces were fused to generate a histogram-based feature vector for the representation of ROI color. High-dimensional feature vectors were reduced to the lower ones in feature discriminant analysis. The best recognition rate in Table 7 is 90.51% by using PCA + NFL for DR and SRC for classification. Though the best performance is achieved by the SRC, it needs much computational time. Practically, the SVM-based method, PCA + NFL for DR and SVM for classification, is recommended for color classification. A multi-class SVM classifier was trained for color classification in a real-time surveillance system. Experimental results have shown that the vehicles' colors were effectively identified using the proposed method.

**Table 7 The best recognition rates and average time for three classifiers and four DR methods**

Classifiers	DR methods	Rates (%)	Time (seconds)
1-NN	PCA	75.36 (100)	0.8141
	PCA + LDA	80.23 (5)	
	PCA + LPP	85.07 (60)	
	PCA + NFLE	85.84 (60)	
SVM	PCA	78.34 (100)	0.0176
	PCA + LDA	82.01 (5)	
	PCA + LPP	87.45 (30)	
	PCA + NFLE	88.67 (20)	
SRC	PCA	79.71 (100)	4.0561
	PCA + LDA	83.43 (5)	
	PCA + LPP	89.78 (70)	
	PCA + NFLE	90.51 (50)	

## Endnote

<sup>a</sup>The color features are available in a website <http://www.csie.nuu.edu.tw/#/personal/labadd/lab404>.

## Competing interests

The authors declare that they have no competing interests.

## Acknowledgements

The work was supported by the National Science Council under grant nos. NSC 101-2221-E-239-034 and 102-2221-E-239 -023.

## Author details

<sup>1</sup>Department of Computer Science and Information Engineering, National Central University, Taoyuan, Taiwan. <sup>2</sup>Department of Computer Science and Information Engineering, National United University, Miaoli, Taiwan.

Received: 2 July 2014 Accepted: 1 October 2014

Published: 16 October 2014

## References

- G. Qiu, Embedded colour image coding for content-based retrieval. *J. Visual Commun. Image Repres.* **15**, 507–521 (2004)
- L.V. Tran, R. Lenz, Compact colour descriptors for colour-based image retrieval. *Signal Process.* **85**, 233–246 (2005)
- N. Baek, S.M. Park, K.J. Kim, S.B. Park, Vehicle color classification based on the support vector machine method. *Proc. Commun. Comput. Inf. Sci.* **2**, 1133–1139 (2007)
- K.J. Kim, S.M. Park, Y.J. Choi, Deciding the number of color histogram bins for vehicle color recognition. *Proc. IEEE Asia-Pacific Services Computing Conference*, 134–138 (2007)
- M.J. Yang, G. Han, X.F. Li, X.C. Zhu, L. Li, Vehicle color recognition using monocular camera. *Proc. IEEE Int. Conf. on Wireless Communications and Signal Processing*, 1–5 (2011)
- L.W. Tsai, J.W. Hsieh, K.C. Fan, Vehicle detection using normalized color and edge map. *IEEE Trans. Image Process.* **16**, 850–864 (2007)
- S.Y. Chen, J.W. Hsieh, J.C. Wu, Y.S. Chen, Vehicle retrieval using eigen color and multiple instance learning. *Proc. Int. Information Hiding and Multimedia Signal Process*, 657–660 (2009)
- L.M. Brown, Example-based color vehicle retrieval for surveillance. *Proc. IEEE Int. Conf. Advanced Video and Signal Based Surveillance*, 91–96 (2010)
- X. Li, G. Zhang, J. Fang, J. Wu, Z. Cui, Vehicle color recognition using vector matching of template. *Proc. Int. Symp. Electronic Commerce and Security*, 189–193 (2010)
- G.D. Finlayson, B. Schiele, J.L. Crowley, Comprehensive color image normalization. *Proc. 5th European Conference on Computer Vision* **1**, 475–490 (1998)
- J.W. Hsieh, L.C. Chen, S.Y. Chen, S.C. Lin, D.Y. Chen, Vehicle color classification under different lighting conditions through color correction. *Proc. IEEE Int. Symp. Circuits and Systems*, 1859–1862 (2012)
- Y. Shen, R. Mo, Y. Zhu, L. Wei, W. Gao, Z. Peng, Over-exposure image correction with automatic texture synthesis. *Proc. Int. Congress on Image and Signal Process*, 794–797 (2011)
- D. Guo, Y. Cheng, S. Zhuo, T. Sim, Correcting over-exposure in photographs. *Proc. IEEE Int. Conference on Computer Vision and Pattern Recognition*, 515–521 (2010)
- H. Stokman, T. Gevers, Selection and fusion of color models for image feature detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(3), 371–381 (2007)
- P. Wolfe, The simplex method for quadratic programming. *Econometric* **27**(3), 382–398 (1959)
- S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding. *Science* **290**(22), 2323–2326 (2000)
- X. He, S. Yan, Y. Ho, P. Niyogi, H.J. Zhang, Face recognition using Laplacianfaces. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(3), 328–340 (2005)
- Y.N. Chen, C.C. Han, C.T. Wang, K.C. Fan, Face recognition using nearest feature space embedding. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**(6), 1073–1086 (2011)
- J. Tenenbaum, V. de Silva, J. Langford, A global geometric framework for nonlinear dimensionality reduction. *Science* **290**(5500), 2319–2323 (2000)

20. M. Belkin, P. Niyogi, Laplacian eigenmaps and spectral techniques for embedding and clustering. *Proc Advances in Neural Information Processing Systems* (MIT Press) **14**, 585–591 (2001)
21. Y.C. Wang, C.T. Hsieh, C.C. Han, K.C. Fan, The color identification of automobiles for video surveillance. *Proc. IEEE Int. Carnahan Conference on Security Technology (ICCST)*, 1–5 (2011)
22. E. Dule, M. Gokmen, M.S. Beratoglu, A convenient feature vector construction for vehicle color recognition. *Proc 11th WSEAS International Conference on Neural Networks, Evolutionary Computing and Fuzzy systems*, 250–255 (2010)
23. Y.T. Wu, J.H. Kao, M.Y. Shih, A vehicle color classification method for video surveillance system concerning model-based background subtraction. *Proc. Pacific-Rim Conference on Multimedia* **6297**, 369–380 (2010)
24. Y.Y. Lu, C.C. Han, M.C. Lu, K.C. Fan, A vision-based system for the prevention of car collisions at night. *Mach. Vision Appl.* **22**, 117–127 (2011)
25. R.E. Fan, P.H. Chen, C.J. Lin, Working set selection using second order information for training SVM. *J. Mach. Learn. Res.* **6**, 1889–1918 (2005)
26. N. Dalal, B. Triggs, Histograms of oriented gradients for human detection. *Proc. IEEE Int. Conf. Computer Vision and pattern Recognition* **1**, 886–893 (2005)
27. G. Wyszecki, W.S. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae*, 2nd edn. (Wiley, New York, 1982)
28. S.Z. Li, Face recognition based on nearest linear combinations. *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, 839–844 (1998)
29. S. Yan, D. Xu, B. Zhang, H.J. Zhang, Q. Yang, S. Lin, Graph embedding and extensions: general framework for dimensionality reduction. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(1), 40–51 (2007)
30. J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, Y. Ma, Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(2), 210–227 (2009)
31. S. Boyd, L. Vandenberghe, *Convex Optimization* (Cambridge University Press, Cambridge, 2004)
32. E. Candes, J. Romberg,  $\ell^1$ -magic: Recovery of sparse signals via convex programming. (2005). <http://www.acm.caltech.edu/l1magic/>

doi:10.1186/1687-5281-2014-48

**Cite this article as:** Wang et al.: Vehicle color classification using manifold learning methods from urban surveillance videos. *EURASIP Journal on Image and Video Processing* 2014 **2014**:48.

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Immediate publication on acceptance
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

---

Submit your next manuscript at ▶ [springeropen.com](http://springeropen.com)

---