Journal of Engg. Research Vol. 6 No. (3) September 2018 pp 34-53

Pedestrian traffic light detection in complex scene using AdaBoost with multi-layer features

Xue-Hua Wu*, Renjie Hu*** and Yu-Qing Bao***

* *School of Electrical Engineering, Southeast University, Nanjing 210096, China wuxuehuaseu@163.com.*

*** *School of Electrical and Automation Engineering, Nanjing Normal University, Nanjing 210023, China baoyuqing@njnu.edu.cn.*

** *Corresponding Author: hurenjie@seu.edu.cn.*

ABSTRACT

In order to improve the accuracy of the pedestrian traffic light detection in complex scene, an image detection method using AdaBoost with multi-layer features is proposed. In the proposed method, the multi-layer features are adopted to characterize the pedestrian traffic lights, and the AdaBoost algorithm is used to extract the discriminative multi-layer features automatically. The multi-layer features consist of luminance and chrominance components, in which the luminance-layer features are to grasp the shape information, and the chrominance-layer features are to acquire color information. Based on the numerous features, hundreds of efficient weak classifiers are selected by the AdaBoost algorithm to construct a strong classifier. With the strong classifier, images are scanned in test procedure for detection of pedestrian traffic light. Testing results show that the proposed multi-layer features in the CIELAB color space greatly improve the accuracy of the pedestrian traffic light detection, and the proposed methods result in much better performance than the state-of-the-art machine learning methods.

Keywords: pedestrian traffic light detection; AdaBoost; multi-layer features.

INTRODUCTION

Although long canes and guide dogs are widely used to help the visually impaired people with independent mobility, they are found unsatisfied in large environments where more information is needed (Hersh & Johnson, 2008). In the urban street, the visually impaired people need to distinguish the pedestrian traffic light and decide whether to go, pause, or caution. But it is so difficult that the pedestrian traffic light lies in the opposite site of the street beyond boundaries of the long cane, and it is difficult for the guide dog to learn since it could not distinguish colors.

To help the visually impaired people cross the roads according to the pedestrian traffic lights, several research works focus on this issue. Aranda & Mares (2004) propose a visual system that could learn, detect, identify, and track the state of the pedestrian traffic light. Roters et al. (2011) propose a mobile vision system to detect the pedestrian traffic light in live video streams. In addition, several similar works have been proposed in the traffic light detection. Omachi et al. (2009) propose a method for the traffic light detection with color and edge information. Charette et al. (2009) propose a real-time traffic light detection system for on-vehicle camera applications. Sung et al. (2013) propose a real-time traffic light detection system with a filtering scheme.

Generally, the existing methods for the pedestrian traffic light detection can be mainly classified into two categories: 1) color-based and 2) shape-based methods. The color-based methods use predefined modules from the positive data and perform correlation. The work by Aranda & Mares (2004) adopts the color histogram analysis to establish a fixed model, and the rest of works analyze color distributions and design parameterized filter rules for the color. With prior knowledge of the shape and size of objects, the shape-based methods can be used. In the work by Omachi et al. (2009), the edge of the traffic light is considered to be a circle, and the Hough transformation is applied to make a final decision. In the work by Roters et al. (2011), the parameters of the aspect ratio and possible corresponding size are utilized to filter out the regions that are too small or too huge to be a crucial pedestrian traffic light. In the work by Sung et al. (2013), the shape and size of blobs are used to filter out false detections.

In the above research works, the discriminative features used for detection are selected beforehand manually. These techniques need a lot of pre-analysis, which relies on the rich experience of experts. To reduce the burden of experts and enhance the stability of the detection, the machine learning procedures need to be adopted. In the machine learning procedures, the features are extracted and optimized according to the actual needs automatically.

The AdaBoost algorithm provides a good solution for the machine learning. Without prior knowledge of the shape and size information, it can automatically select hundreds of most effective weak classifiers based on numerous weak classifiers through the iteration calculation, and then construct a strong classifier from the weighted sum of these weak classifiers.

Usually the Haar-like features are chosen to develop the weak classifiers. The performances of the AdaBoost algorithm based on the Haar-like features are good in the visually based object detection (Viola & Jones, 2004). However, the Haar-like features only work in the gray-scale image and lack the ability to describe colors. This weakness may limit the accuracy of the detection. To better grasp the color information of the pedestrian traffic light, the multi-layer features are proposed in this paper. In the multi-layer features, the luminance-layer features are used to characterize the shape of the objects, and the chrominance-layer features are used to characterize the color of the objects. Because the multi-layer features can access both shape and color information, the accuracy of the detection can be therefore improved.

The remainder of this paper is organized as follows. In Section 2, the AdaBoost algorithm based on the Haar-like features is introduced. In Section 3, an improved detection method based on the multi-layer features is proposed. Testing results are provided in Section 4. Discussions are presented in Section 5. Finally, conclusions are summarized in Section 6.

ADABOOST-BASED DETECTION

In this section, the problems in the pedestrian traffic light detection are illustrated, as well as the potential solution, the AdaBoost algorithm based on the Haar-like features is introduced.

Necessities

It is challenging to detect the pedestrian traffic light in the real world, due to the difficulties as follows:

- The pedestrian traffic light is not attractive in a cluttered background (shown in Figure 1).
- The appearance of the pedestrian traffic light looks like the shape of a man, which is more difficult to describe than the circle and the arrow (shown in Figure 1(f)).
- The appearances of the red pedestrian traffic light and the green pedestrian traffic light are different (shown in Figure 1(a), (f)). The appearances of the pedestrian traffic light in different countries or even in different residential districts are different (shown in Figure 1(a), (b), (c)).
- More than one pedestrian traffic light is captured, but only one is crucial (shown in Figure 1(g), (h)).
- The captured colors of the pedestrian traffic lights are affected with different illumination conditions (shown in Figure 1(a), (b), (c), (d)).

Figure 1. Challenges of detecting pedestrian traffic lights in images (Roters et al., 2011).

Although there are a lot of difficulties, there are positive factors to detect the pedestrian traffic light:

- The color, structure, shape, circuitry, background, arrangement, and installation of the pedestrian traffic lights are similar.
- It is observed that when illumination conditions vary, the absolute intensity values of different regions change dramatically, but their mutual ordinal relationship remains largely unaffected (Oren et al., 1997).

The Haar-like features can capture the differences between average intensities of adjacent rectangular regions. However, the quantity of the Haar-like features is very large; it is time consuming to enumerate the whole features. In addition, not all but a few effective weak classifiers are distinguishable, so it is not essential to compute every feature to make the final decision (Wen et al., 2015). To cope with this problem, the AdaBoost algorithm can be adopted. The AdaBoost algorithm can automatically select the most effective weak classifiers to construct a strong classifier. The rest of this section will give some brief introduction of the Haar-like features and the AdaBoost algorithm.

Haar-like features

The Haar-like features are traditional descriptors used to encode the knowledge about the local image region. These features are reminiscent of Haar basis functions (Oren et al., 1997). A group of Haar-like features are shown in Figure 2. The Haar-like features are defined in the gray-scale image, and each feature consists of at least two vertically or horizontally adjacent rectangular regions. The value of each Haar-like feature is the result of the sum of pixel values within gray rectangular regions subtracted from the sum of pixel values within white rectangular regions. The procedure for enumerating all the Haar-like features in the feature pool is summarized in Algorithm 1.

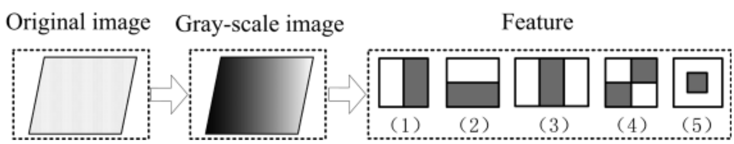

Figure 2. Examples of Haar-like features in the gray-scale image.

Algorithm 1. Enumerate all the Haar-like features in the feature pool.

Input

Original image in the RGB colour space

Process

- Crop the region of interest (ROI) as a sample
- Scale the sample to a base resolution of W by H pixels, and transform the sample into gray-scale image
- Compute the integral image
- Enumerate all the Haar-like features according to Figure 2
- Compute all the feature values with the integral image

Output

Feature pool

AdaBoost algorithm

The AdaBoost algorithm is a good approach to find out the most effective Haar-like features. It was first proposed by Freund & Schapire (1995). Later, Viola and Jones applied it in a face detection framework in 2001. The AdaBoost algorithm can construct either a monolithic classifier or a cascaded classifier, and there is little difference in terms of accuracy between the monolithic 200-feature classifier and cascaded classifier (Viola & Jones, 2004).

The AdaBoost algorithm for the strong classifier construction of T most effective features is summarized in Algorithm 2. During every cycle, each feature is used to develop a weak classifier. Then we choose the most effective weak classifier that best separates the positive and negative samples. When the loop is over, we construct a strong classifier based on the T best weak classifiers. The function of the weak classifier is shown in (1). It is a two-value equation consisting of three parameters. The parameter f indicates the value of the corresponding feature, the parameter θ indicates the threshold, and the parameter p indicates the direction of the inequality.

$$h_j = \begin{cases} 1 & \text{if } p_j f_j(x) < p_j \theta_j \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Algorithm 2. The AdaBoost algorithm for the strong classifier construction of T most effective features.

Input

Samples x_i with artificial classification, where classification y_i=0 for negative sample and y_i=1 for positive sample.

Process

- initialize the distribution $w_{1,i}=1/n$, where n is the number of samples
- for t=1,…,T

normalize the distribution

$$w_{t,i} = w_{t,i} / \sum_{i=1}^{n} w_{t,i}$$

train a weak classifier for each feature in the feature pool

$$h_j = \begin{cases} 1 & \text{if } p_j f_j(x) < p_j \theta_j \\ 0 & \text{otherwise} \end{cases}$$

computer the weighted error

$$\varepsilon_j = \sum_{i=1}^{n} w_{t,i}[y_i \neq h_j(x_i)]$$

select the weak classifier with the minimal weighted error

$$h_t = \arg\min_{h_j} \varepsilon_j$$

computer the weak classifier weight

$$\alpha_t = \frac{1}{2}\ln\frac{1-\varepsilon_t}{\varepsilon_t}$$

update the distribution

$$w_{t+1,i} = w_{t,i}\exp(-\alpha_t h_t(x_i) y_i)$$

• The final strong classifier is

$$H(x) = \begin{cases} 1 & \sum_{t=1}^{T} \alpha_t h_t(x) \geq \theta_s \\ 0 & \text{otherwise} \end{cases}$$

Output

Weak classifier set h_t

Strong classifier H(x)

THE PROPOSED METHOD

Although the Haar-like features selected by the AdaBoost algorithm can well distinguish different shapes, they work in the gray-scale image, which inevitably disregards color information. In order to improve the accuracy of the pedestrian traffic light detection in complex scene, an image detection method using AdaBoost with multi-layer features is proposed. In the multi-layer features, new types of chrominance-layer features are designed to characterize the colors of the pedestrian traffic lights. In this section, a brief introduction of the proposed method is presented.

Normally, color images are described in the familiar RGB, HSV, YCrCb, and CIELAB color spaces as follows:

- RGB: The RGB color space is often resorted to since most color images captured by digital cameras are represented by the RGB primaries. The RGB color space is the simplest color model using a Cartesian coordinate system, in which three axes represent the red, green, and blue components, respectively (Elias, 2014).
- HSV: The HSV color space is approximately uniform space derived from the RGB color space. The H (hue) represents the dominant color in a pure form; the S (saturation) represents the chroma of a color divided by its luminance; the V (value) represents the perceptual correlate of lightness (Smith, 1978; Manjunath et al., 2001; Zhao et al., 2012).
- YCrCb: The YCrCb color space converts the RGB primaries into an opponent luma-chroma color space. In the YCrCb color space, the Y is luminance component, which corresponds to the intensity, and Cr and Cb are chrominance components corresponding to color-difference (R-Y) and (B-Y) signals (Recommendation ITU-R BT.601–7, 2011; Ramanath & Drew, 2014).
- CIELAB: The CIELAB color space is proposed to satisfy human perceptual behavior. In the CIELAB color space, axes are denoted by L (lightness), a (redness-greenness), and b (yellowness-blueness) (Encyclopedia of Color Science and Technology, 2016).

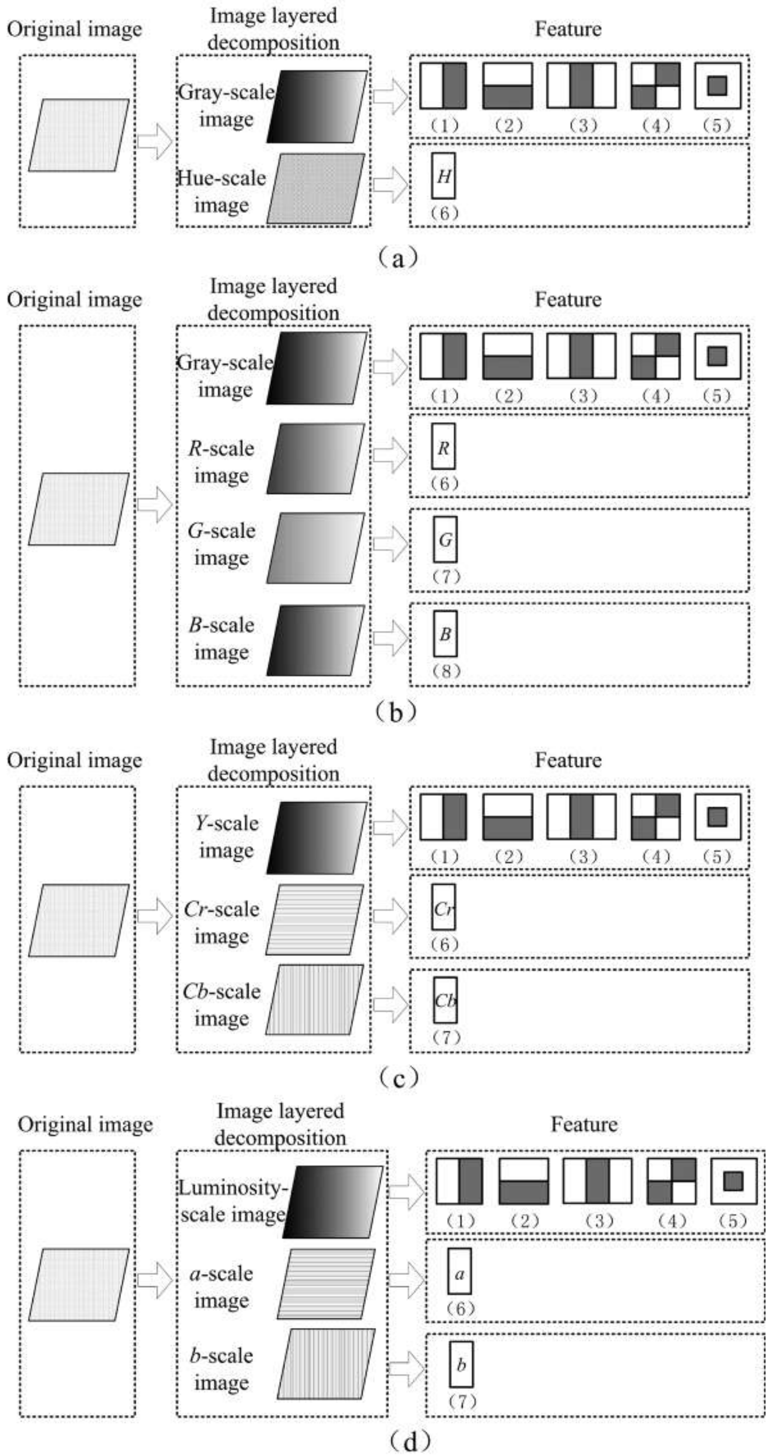

Figure 3. Examples of multi-layer features (a) in the gray-scale and hue-scale images; (b) in the gray-scale, R-scale, G-scale, and B-scale images; (c) in the Y-scale, Cr-scale, and Cb-scale images; (d) in the luminosity-scale, a-scale, and b-scale images.

According to the selected color space (e.g., the RGB, HSV, YCrCb, and CIELAB color spaces), the original image is decomposed into the luminance-layer and chrominance-layer components. The Haar-like features belong to the luminance-layer, and the new types of features are designed for chrominance-layer. The new types of features are one-rectangle features, which are shown in Figure 3. When the HSV color space is selected (shown in (a)), the value of each new feature is the sum of pixel values within white rectangular region in the hue-scale image. When the RGB color space is selected (shown in (b)), the value of each of feature (6), feature (7), and feature (8) is the sum of the pixel values within white rectangular regions in the R-scale image, the G-scale image, and the B-scale image, respectively. When the YCrCb color space is selected (shown in (c)), the Haar-like features act in the Y-scale image. The value of each of feature (6) and feature (7) is the sum of the pixel values within white rectangular regions in the Cr-scale image and Cb-scale image, respectively. When the CIELAB color space is selected (shown in (d)), the Haar-like features act in the luminosity-scale image. The value of each of feature (6) and feature (7) is the sum of the pixel values within white rectangular regions in the a-scale image and b-scale image, respectively. In order to describe all the features simply, we name them the multi-layer features. For instance, the multi-layer features in the gray-RGB color space for a W by H pixels image are summarized in Table 1.

Table 1. The number of the multi-layer features in the gray-RGB space for a W by H pixels image.

Feature type	**Feature number**	**Feature number (W=10,H=30)**
(1)	$num_1=\left(\left[\frac{W}{2}\right]+\left[\frac{W-1}{2}\right]+\ldots+\left[\frac{3}{2}\right]+1\right)\times\left(\left[\frac{H}{1}\right]+\left[\frac{H-1}{1}\right]+\ldots+\left[\frac{2}{1}\right]+1\right)$	11625
(2)	$num_2=\left(\left[\frac{W}{1}\right]+\left[\frac{W-1}{1}\right]+\ldots+\left[\frac{2}{1}\right]+1\right)\times\left(\left[\frac{H}{2}\right]+\left[\frac{H-1}{2}\right]+\ldots+\left[\frac{3}{2}\right]+1\right)$	12375
(3)	$num_3=\left(\left[\frac{W}{3}\right]+\left[\frac{W-1}{3}\right]+\ldots+\left[\frac{4}{3}\right]+1\right)\times\left(\left[\frac{H}{1}\right]+\left[\frac{H-1}{1}\right]+\ldots+\left[\frac{2}{1}\right]+1\right)$	6975
(4)	$num_4=\left(\left[\frac{W}{2}\right]+\left[\frac{W-1}{2}\right]+\ldots+\left[\frac{3}{2}\right]+1\right)\times\left(\left[\frac{H}{2}\right]+\left[\frac{H-1}{2}\right]+\ldots+\left[\frac{3}{2}\right]+1\right)$	5625
(5)	$num_5=\left(\left[\frac{W}{3}\right]+\left[\frac{W-1}{3}\right]+\ldots+\left[\frac{4}{3}\right]+1\right)\times\left(\left[\frac{H}{3}\right]+\left[\frac{H-1}{3}\right]+\ldots+\left[\frac{4}{3}\right]+1\right)$	2175
R (6)	$num_6=\left(\left[\frac{W}{1}\right]+\left[\frac{W-1}{1}\right]+\ldots+\left[\frac{2}{1}\right]+1\right)\times\left(\left[\frac{H}{1}\right]+\left[\frac{H-1}{1}\right]+\ldots+\left[\frac{2}{1}\right]+1\right)$	25575
G (7)	$num_7=\left(\left[\frac{W}{1}\right]+\left[\frac{W-1}{1}\right]+\ldots+\left[\frac{2}{1}\right]+1\right)\times\left(\left[\frac{H}{1}\right]+\left[\frac{H-1}{1}\right]+\ldots+\left[\frac{2}{1}\right]+1\right)$	25575
B (8)	$num_8=\left(\left[\frac{W}{1}\right]+\left[\frac{W-1}{1}\right]+\ldots+\left[\frac{2}{1}\right]+1\right)\times\left(\left[\frac{H}{1}\right]+\left[\frac{H-1}{1}\right]+\ldots+\left[\frac{2}{1}\right]+1\right)$	25575
Total	$num = num_1+num_2+num_3+num_4+num_5+num_6+num_7+num_8$	115500

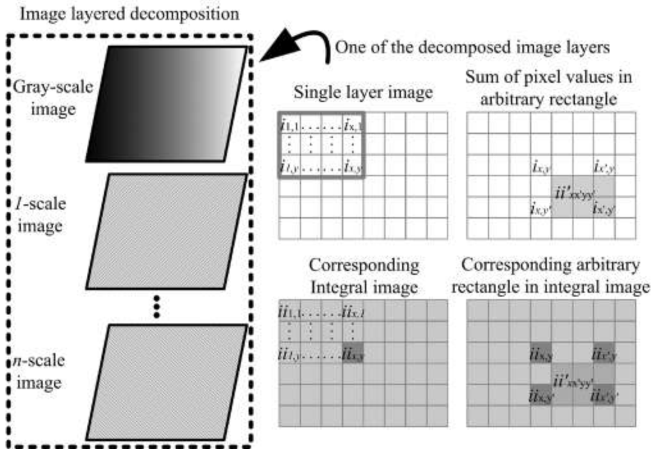

Figure 4. Integral image of a single layer.

It is proved that the rectangular features can be computed fast, and any rectangular sum can be computed in four array references using the integral image (Viola & Jones, 2004). Here, we adopt the same approach. In the proposed method, the integral image of each decomposed layer needs to be computed. In a single layer shown in Figure 4, $i_{x,y}$ represents the value of the pixel (*x,y*), which is at the location of width x and height y. In the corresponding integral image, $ii_{x,y}$ represents the sum of pixel values above and to the left of the pixel (*x,y*), that is,

$$ii_{x,y} = \sum_{a \le x, b \le y} i_{a,b} \tag{2}$$

The sum of pixel values in any rectangle enclosed by pixels (*x,y*), (*x',y*), (*x,y'*), (*x',y'*) can be calculated as follows:

$$ii'_{xx',yy'} = \sum_{x < a \le x', y < b \le y'} i_{a,b} \tag{3}$$

When the integral image is used herein, it can be easily obtained by (4). The integral image can reduce the computational burden in calculating the sum of pixel values.

$$ii'_{xx',yy'} = ii_{x',y'} - ii_{x,y'} - ii_{x',y} + ii_{x,y} \tag{4}$$

In order to explore the feasibility of the proposed method, a monolithic 200-feature classifier is trained and used for pedestrian traffic light detection. The proposed pedestrian traffic light detection system is illustrated in Figure 5. The original images are taken at intersections when pedestrians want to cross the road. The resolution of these images is 640×480. During the training process, the training samples cropped from the original images are normalized at size 30×10 and decomposed into multiple layers. The multi-layer features are enumerated in the feature pool and are used to develop numerous weak classifiers. The AdaBoost algorithm is adopted to train a strong classifier based on the most distinguishable weak classifiers. During the testing process, the decomposed layers of the original image are scanned using the strong classifier, in order to identify the candidate region of the corresponding pedestrian traffic light.

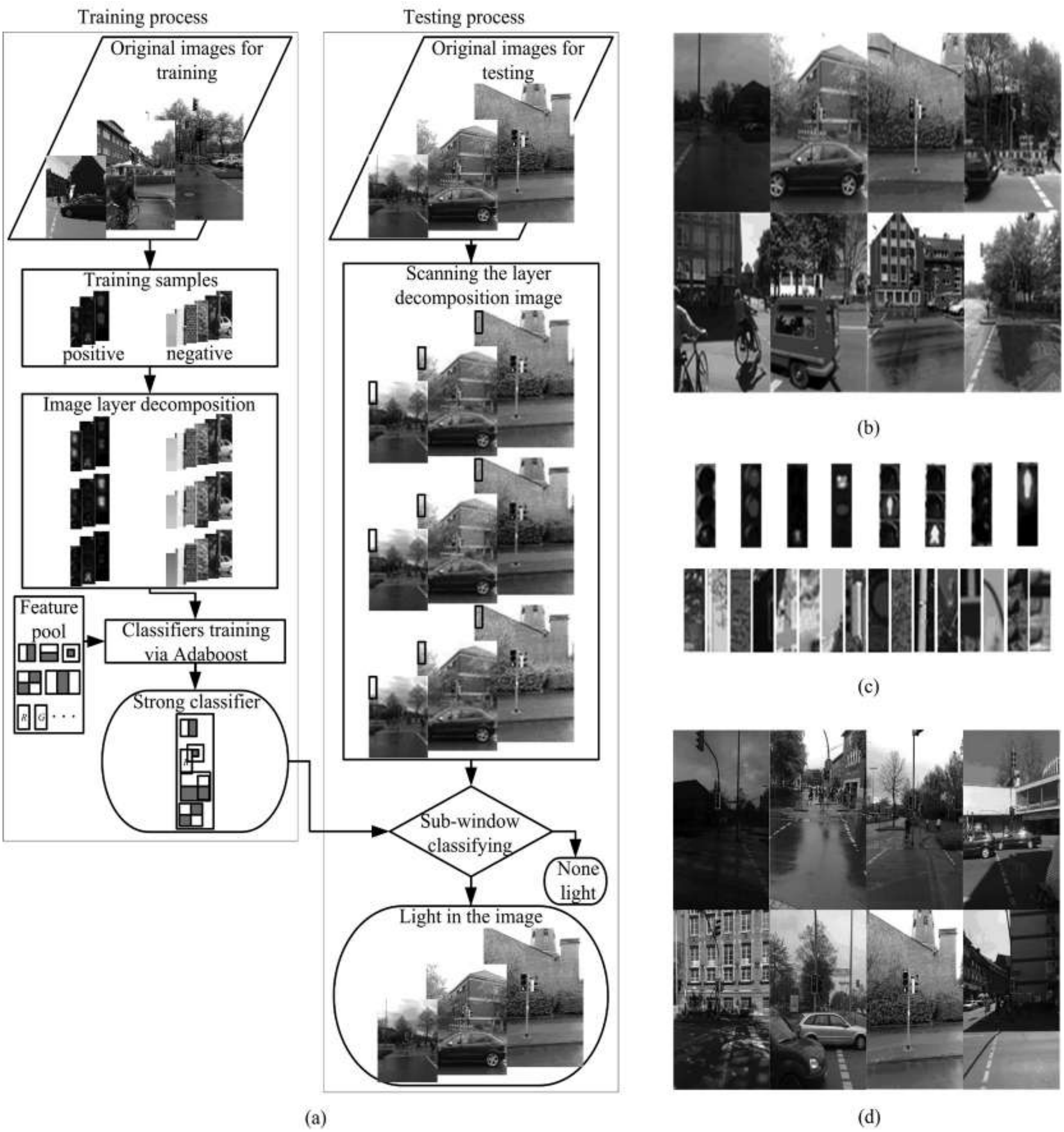

Figure 5. The illustration of the proposed pedestrian traffic light detection system. (a) The process of the proposed pedestrian traffic light detection system; (b) examples of original images taken at intersections when pedestrians want to cross the road; (c) examples of positive and negative samples for training; (d) examples of results for testing.

For the task of the pedestrian traffic light detection, the features selected by the AdaBoost algorithm are meaningful and easily interpreted. Figure 6 gives a few examples of the multi-layer features for the description of the pedestrian traffic light appearance.

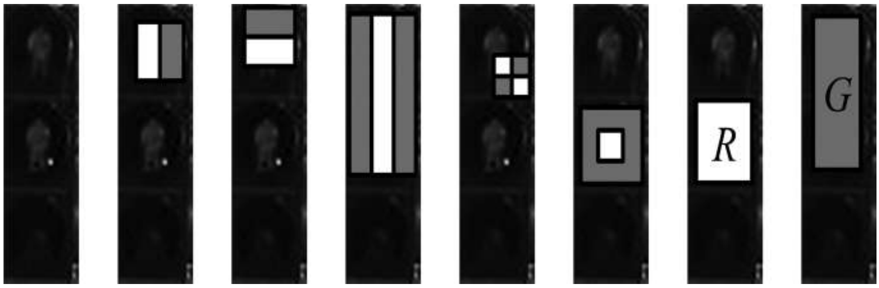

Figure 6. Examples of the multi-layer features for describing the pedestrian traffic light appearance.

EXPERIMENTS

In this section, several experiments are performed to verify the proposed method. All experiments are implemented on the MATLAB software. The process of the proposed pedestrian traffic light detection system is shown in Figure 5. The data used for the experiments includes 501 images with resolution up to 640×480 and is publicly available on the Internet (Roters et al., 2011). All the images are taken at intersections when pedestrians want to cross the road, and almost every image contains pedestrian traffic lights. Furthermore, the visible pedestrian traffic lights each are located within a bounding box, the crucial lights are marked, and phases of the lights (red or green) are stored. The positive samples of the pedestrian traffic lights are cropped by reference to the bounding boxes of the crucial lights and then scaled to 30×10 pixels. Totally, 496 positive samples are obtained including 312 positive samples of the red pedestrian traffic lights and 184 positive samples of the green pedestrian traffic lights.

For comparison, the following two methods are compared:

- In Method 1, the above AdaBoost-based testing procedure is used to identify the candidate region for the pedestrian traffic light, and then the color within the region is analyzed (e.g., analyzing the histogram within the region) to decide whether it is a red or green pedestrian traffic light.
- In Method 2, the above AdaBoost-based testing procedure is used to directly identify the red or green pedestrian traffic light.

The flow charts of Method 1 and Method 2 are shown in Figure 7. In each method, five experiments are considered:

E1: With Haar-like features in the gray-scale image;

E2: With multi-layer features for the gray-HSV color space;

E3: With multi-layer features for the gray-RGB color space;

E4: With multi-layer features for the YCrCb color space;

E5: With multi-layer features for the CIELAB color space.

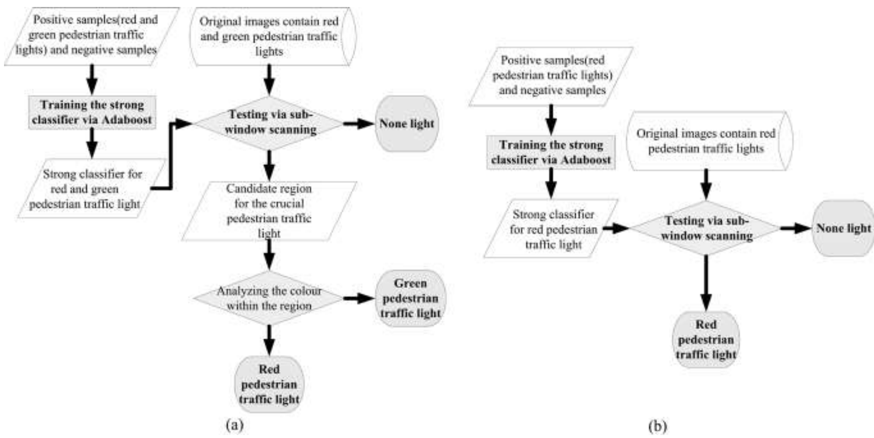

Figure 7. The flow charts of the proposed (a) Method 1; (b) Method 2.

Method 1:

To validate the performance of Method 1, the dataset is divided into two disjoint sets: the first set is used for training, and the second set is used for testing. The first set contains 373 images, and the second set contains 123 images. 30×10 samples for training and testing are cropped from the 640×480 training and testing images, respectively. On these 30×10 samples, we compare the effectiveness of Method 1 (with multi-layer features for the CIELAB color space) with that of some state-of-the-art machine learning methods. The ROCs (receiver operating characteristic curves) of these validation methods are shown in Figure 8. For the multi-layer features extraction of Method 1, 89925 features are enumerated in the feature pool. For the feature extraction of HOG (Dalal & Triggs, 2005), 2×2 cell blocks of 8×8 pixel cells are performed to generate a 756-dimensional vector for the grayscale image of each 30×10 sample. For the feature extraction of Gabor (Yan, Li & Gao, 2011; Xing & Luo, 2016), we select one scale and four orientations. Then in Method 1, 200 most effective multi-layer features are selected by the AdaBoost algorithm. While training RBF neural network, the HOG features are taken as input vectors to create a two-layer network. While training classifiers with the SVM algorithm (Chang & Lin, 2001; Mittal & Sharma, 2016), we use radial basis kernel function for the HOG features and linear kernel function for the Gabor features. While training classifiers with the AdaBoost algorithm, magnitudes of the HOG (or Gabor) features are normalized and stored as one-dimensional main features for selection according to the idea of paper by Kishino et al. (2013).

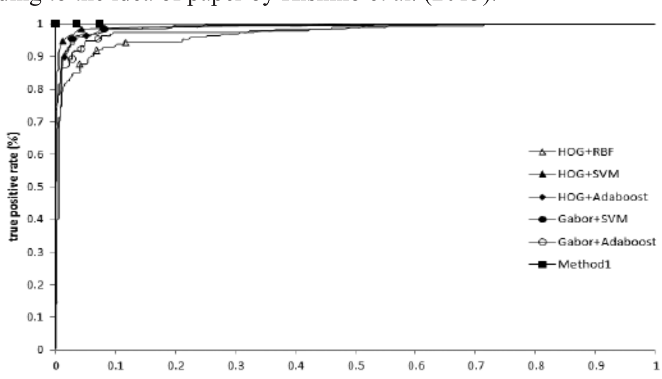

Figure 8. The ROCs of Method 1 and other validation methods.

In Figure 9, the first 10 features of the selected distinguishable weak classifiers under different color spaces are overlaid on the typical positive sample. These features are meaningful and easily interpreted. Then the trained strong classifier is used to scan the 123 640×480 testing images. Each image is gradually zoomed out from 100% to 20% and a moving window of size 30×10 scans it meanwhile. The detection results in testing images of Method 1 are shown in Table 2. To evaluate the performance of experiments E1~E5 in Method 1, some indicators are considered:

Correct detection rate (*CDR*): probability of the total number of predictions; those are correct.

$$CDR = \frac{TP}{total} \times 100\% \tag{5}$$

False alarm rate (*FAR*): probability of the total number of predictions; those are incorrect.

$$FAR = \frac{FP}{total} \times 100\% \tag{6}$$

Missed detection rate (*MDR*): probability of the total number of predictions; those are missed.

$$MDR = \frac{M}{total} \times 100\% \tag{7}$$

where *TP* is the number of correct detections that the crucial pedestrian traffic light in the image is detected; *FP* is the number of incorrect detections that the other region in the image is considered to be the crucial pedestrian traffic light; *M* is the number of missed detections that none of the pedestrian traffic lights is detected in the image; total is the number of the images for test.

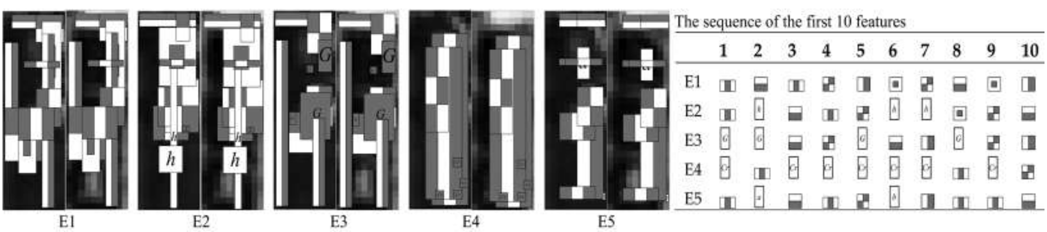

Figure 9. The first 10 features overlaid on the typical positive samples in Method 1.

Table 2. The detection results in testing images of Method 1.

	E1*	E2*	E3*	E4*	E5*
Number of correct detections	100	107	108	41	115
Number of false detections	22	12	7	82	3
Number of missed detections	1	4	8	0	5
Total number	123	123	123	123	123
Correct detection rate (%)	81.30	86.99	87.80	33.33	93.50
False alarm rate (%)	17.89	9.76	5.69	66.67	2.44
Missed detection rate (%)	0.81	3.25	6.50	0.00	4.07

* E1~E5 denote experiments with Haar-like features (E1, in the gray-scale image) and the multi-layer features (E2~E5, for the gray-HSV, gray-RGB, YCrCb, and CIELAB color spaces).

Method 2:

In Method 2, the 312 images that have crucial red pedestrian traffic lights are divided into two disjoint sets. 234 images are used for training, and the remaining images are used for testing. Samples for training and testing are cropped from the training and testing images, respectively.

In order to evaluate the performance of Method 2, we compare it with some state-of-the-art machine learning methods on 30×10 samples. In Method 2, 89925 multi-layer features for the CIELAB color space are enumerated, and 200 most effective features are selected by the AdaBoost algorithm. For the gray-scale images of 30×10 samples, the HOG features of 756 dimensional are generated to train a two-layer RBF neural network. Using the same HOG features, a RBF-SVM classifier is trained by the SVM algorithm. The SVM algorithm is also used to train a linear-SVM classifier based on Gabor features of one scale and four orientations. The ROCs of these validation methods under 30×10 testing samples are shown in Figure 10. In Figure 11, the first 10 features of the selected distinguishable weak classifiers under different color spaces are overlaid. Using the trained strong classifier to scan the 640×480 images, the detection results of experiments E1~E5 in Method 2 are listed in Table 3.

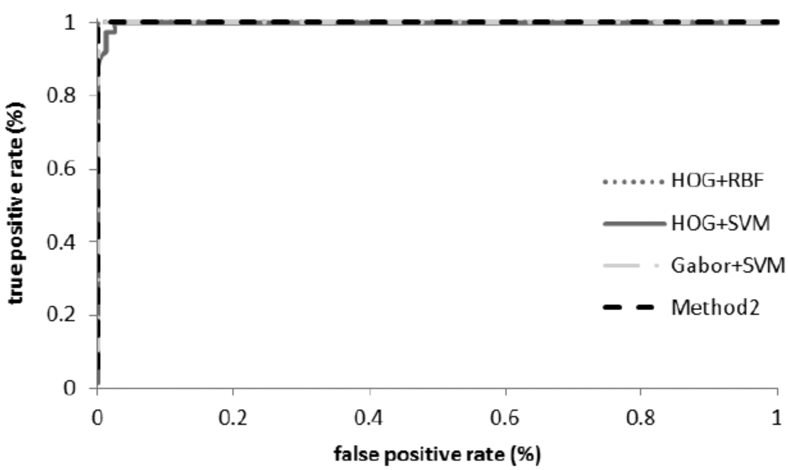

Figure 10. The ROCs of Method 2 and other validation methods.

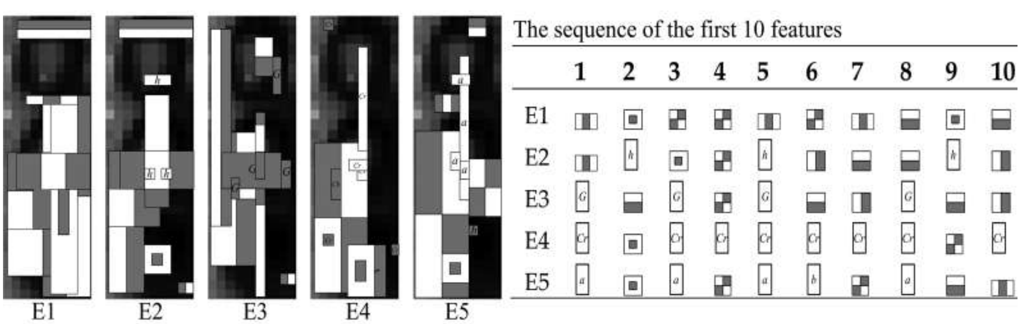

Figure 11. The first 10 features overlaid on the typical positive samples in Method 2.

Table 3. The detection results in testing images of Method 2.

	E1*	E2*	E3*	E4*	E5*
Number of correct detections	64	62	69	73	76
Number of false detections	12	0	1	0	0
Number of missed detections	2	16	8	5	2
Total number	78	78	78	78	78
Correct detection rate (%)	82.05	79.49	88.46	93.59	97.44
False alarm rate (%)	15.38	0.00	1.28	0.00	0.00
Missed detection rate (%)	2.56	20.51	10.26	6.41	2.56

* E1~E5 denote experiments with Haar-like features (E1, in the gray-scale image) and the multi-layer features (E2~E5, for the gray-HSV, gray-RGB, YCrCb, and CIELAB color spaces).

DISCUSSIONS

In this section, discussions of the results of all the experiments (introduced in Section 4) are provided.

Method 1:

From Figure 8, we can conclude that Method 1 produces better classification results than the other methods. By comparing the performances of E1~E5 (shown in Figure 9, Table 2), we have the following observations:

- The performance of E4 is not satisfactory. It seems that the multi-layer features for the YCrCb color space are not suitable for Method 1. In the YCrCb color space, the color-difference (R-Y) and (B-Y) were chosen as chrominance components in order to minimize encoding-decoding errors (Dios & Garcia, 2003; Dios & Garcia, 2004). The chrominance components Cr and Cb correspond to R and B of the RGB space normalized in intensity, respectively (Mathew, 2014). The lack of the color-difference (G-Y) caused it to be not appropriate for describing the green color (Chen et al., 2009; Chaves-gonzález et al., 2010; Qu & Ding, 2010; Tang et al., 2016).
- The correct detection rate of E2, E3, and E5 is higher than the rate of E1. The false alarm rate of E2, E3, and E5 is lower than the rate of E1, and the missed detection rate of E2, E3, and E5 is higher than the rate of E1. It makes sense that the multi-layer features for the gray-HSV, gray-RGB, and CIELAB color spaces are more suitable for Method 1. When the visually impaired come across intersections, it is better not to report than give a wrong report.
- The performance of E5 is better than E2~E4, since the CIELAB color space is perceptually uniform. In this space, uniform changes of components correspond to uniform changes of perceived colors, so it can achieve more accurate perception of colors (Hill et al., 1997; Sangwine, 2000; Kim et al., 2012; Zhang, 2014). In the CIELAB color space, the luminance-layer L is a measure of lightness, which approximates the related human sensation for a greater range of chroma than does the intensity; the chrominance-layer a is a measure of redness (or –a of greenness) and b of yellowness (or –b of blueness) (Kasson & Plouffe, 1992; Gil-Munoz et al., 1998). In the work by Yang & Sowmya (2015), it is proved that the sharpness of L, a and b

layer mainly determines the extent of observers' visual perception. Therefore, the multi-layer features in the CIELAB color space are efficient at representation of objects. They are the most suitable features for Method 1.

Method 2:

In Method 2, it is shown in Figure 10 that Method 2 produces a competitive classification result. From Figure 11 and Table 3, we can conclude that:

- The correct detection rate of E3~E5 is higher than the rate of E1. The false alarm rate of E2~E5 is lower than the rate of E1, and the missed detection rate of E2~E5 is higher than (or equal to) the rate of E1. These mean that the multi-layer features in the gray-RGB, YCrCb, and CIELAB color spaces are more suitable for Method 2. In particular, the false alarm rate of E2, E4, and E5 is zero, so it is safer to use the multi-layer features in the gray-RGB, YCrCb, and CIELAB color spaces for the red pedestrian traffic light detection.
- The performance of E5 is the best one. More concretely, the correct detection rate of E5 is the highest of all, the false alarm rate of E5 is zero, and the missed detection rate of E5 is the lowest of all. This is because the CIELAB color space is a widely accepted device independent perceptual color space derived from experiments. In this color space, the range of colors does not depend on the characteristics of a particular device, or the visual skills of a specific observer or the lightening condition (Sharifzadeh et al., 2014).

General:

In all, it is obvious that the multi-layer features in the CIELAB color space are most suitable for the pedestrian traffic light detection. In Method 2, especially, the correct detection rate is 97.44%, the false alarm rate is 0.00%, and the missed detection rate is 2.56%. To an extent, it is in accordance that the CIELAB color space is one of the optimal color spaces for segmentation (Hernández-hernández et al., 2016).

CONCLUSIONS

In this paper, the multi-layer-features-based AdaBoost method is proposed for the detection of the pedestrian traffic light. The contributions of this paper can be summarized as follows:

- The machine learning method AdaBoost algorithm is adopted for detecting the pedestrian traffic light. Compared to previous works, the AdaBoost algorithm extracts and optimizes features for the detection according to actual needs.
- The multi-layer features are specially designed for the AdaBoost-based pedestrian traffic light detection system. Not only the luminance-layer features, but also the chrominance-layer features are considered. The Haar-like luminance-layer features are used to describe the shape, whereas the chrominance-layer features are used to catch the color information. Experimental results show that it improves the pedestrian traffic light detection accuracy and reduces the false alarm rate.
- The effectiveness of the proposed method has been verified by numerous testing results. Although the proposed method is designed for the detection of pedestrian traffic lights, it can be applied to other fields, such as blind path detection, fabric defects detection, and traffic or safety signs detection.

REFERENCES

Aranda, J. & Mares, P. 2004. Visual system to help blind people to cross the street, International Conference on Computers Helping People with Special Needs, 2004:454–461.

Charette, R.D. & Nashashibi, F. 2009. Real time visual traffic lights recognition based on spot light detection and adaptive traffic lights templates, IEEE Intelligent Vehicles Symposium, 2009:358–363.

Chang, C. & Lin, C. 2001. LIBSVM: a library for support vector machines. Available Online: http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Chen, P., Yang, Y. & Chang, L. 2009. Automated bridge coating defect recognition using adaptive ellipse approach, Automation in Construction, 18:632–643.

Chaves-gonzález, J.M., Vega-rodríguez, M.A., Gómez-pulido, J.A. & Sánchez-pérez, J.M. 2010. Detecting skin in face recognition systems: A colour spaces study, Digital Signal Processing, 20:806–823.

Dalal, N. & Triggs, B. 2005. Histograms of oriented gradients for human detection, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CA, USA.

Dios, J.J.d. & Garcia, N. 2003. Face detection based on a new color space YCgCr, International Conference on Image Processing, Barcelona, Spain.

Dios, J.J.d. & Garcia, N. 2004. Fast face segmentation in component color space, International Conference on Image Processing, Singapore, Singapore.

Elias, R. 2014. Digital Media: A problem-solving approach for computer graphics, Springer International Publishing, Switzerland, doi:10.10076-05137-319-3-978/.

Encyclopedia of Color Science and Technology. 2016. CIELAB (Standards: CIE), Springer, New York, doi:10.1007521000_7-8071-4419-1-978/.

Freund, Y. & Schapire, R.E. 1995. A desicion-theoretic generalization of on-line learning and an application to boosting, European Conference on Computational Learning Theory, 1995:23–37.

Gil-Munoz, R., Gomez-Plaza, E. & Martinez, A. et al. 1998. Evolution of the CIELAB and other spectrophotometric parameters during wine fermentation Influence of some pre and postfermentative factors, Food Research International, 30(9):699–705.

Hersh, M.A. & Johnson, M.A. 2008. Assistive technology for visually impaired and blind people, Springer, London, doi:10.10078-867-84628-1-978/.

Hill, B., Roger, T. & Vorhagen, F.W. 1997. Comparative analysis of the quantization of color spaces on the basis of the CIELAB color-difference formula, ACM Transactions on Graphics, 16(2):109–154.

Hernández-hernández, J.L., García-mateos, G. & González-esquiva, J.M. 2016. Optimal color space selection method for plant/soil segmentation in agriculture, Computers and Electronics in Agriculture, 122:124–132.

Kishino, T., Zhe, S. & Micheletto, R. 2013. A fast and precise HOG-Adaboost based visual support system capable to recognize Pedestrian and estimate their distance, International Conference on Image Analysis and Processing, 8158:20–29.

Kim, K., Kim, S. & Cho, H.G. 2012. A compact photo browser for smartphone imaging system with content-sensitive overlapping layout, 6th International Conference on Ubiquitous Information Management and Communication, Kuala Lumpur, Malaysia.

Kasson, J.M. & Plouffe, W. 1992. An analysis of selected computer interchange color spaces, ACM Transactions on Graphics, 11(4):373–405.

Manjunath, B.S., Ohm, J. & Vasudevan, V.V. 2001. Color and texture descriptors, IEEE Transactions on Circuits and Systems for Video Technology, 11(6):703–715.

Mittal, T. & Sharma, R. K. 2016. An improved SVM using predator prey optimization and Hooke-Jeeves method for speech recognition, JOURNAL OF ENGINEERING RESEARCH, 4(1):2–20.

Mathew, S. 2014. Performance analysis of spatial color information for object detection using background subtraction, 2014 International Conference on Future Information for Object, 10:63–69.

Omachi, M. & Omachi, S. 2009. Traffic light detection with color and edge information, 2nd IEEE International Conference on Computer Science and Information Technology, 2009:284–287.

Oren, M., Osuna, E. & Poggio, T. 1997. Pedestrian detection using wavelet templates, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1997:193–199.

Papageorgiou, C.P. 1998. A general framework for object detection, 6th International Conference on Computer Vision, 1998:555–562.

Qu, X. & Ding, T. 2010. A fast feature extraction algorithm for detection of foreign fiber in lint cotton within a complex background, Acta Automatica Sinica, 36(6):785–790.

Roters, J., Jiang, X. & Rothaus, K. 2011. Recognition of traffic lights in live video streams on mobile devices, IEEE Transactions on Circuits and Systems for Video Technology, 21(10):1497–1511.

Roters, J., Jiang, X. & Rothaus, K. 2011. Identification of Traffic Lights for Pedestrians with Visual Impairment by the use of Mobile Devices. Available Online: http://cvpr.uni-muenster.de/research/pedestrianlights.

Recommendation ITU-R BT.601–7. 2011. Studio Encoding Parameters of Digital Television for Standard 4:3 and Wide-screen 16:9 Aspect Ratios. Available Online: http://www.itu.int/rec/R-REC-BT.601-7-201103-I/en.

Ramanath, R. & Drew, M.S. 2014. Color Spaces, Springer, US, doi:10.1007452_6-31439-387-0-978/.

Sung, T. & Tsai, H. 2013. Real-time traffic light recognition on mobile devices with geometry-based filtering, 7th International Conference on Distributed Smart Cameras, CA, USA.

Smith, A.R. 1978. Color gamut transform pairs, ACM SIGGRAPH Computer Graphics, 12(3):12–19.

Sangwine, S.J. 2000. Colour in image processing, Electronics and Communication Engineering Journal, 2000:211–219.

Sharifzadeh, S., Clemmensen, L.H. & Borggaard, C. 2014. Supervised feature selection for linear and non-linear regression of L*a*b* color from multispectral images of meat, Engineering Applications of Artificial Intelligence, 27:211–227.

Tang, J., Chen, X., Miao, R. & Wang, D. 2016. Weed detection using image processing under different illumination for site-specific areas spraying, Computers and Electronics in Agriculture, 122:103–111.

Viola, P. & Jones, M. 2004. Robust real-time face detection, International Journal of Computer Vision, 57(2):137–154.

Wen, X., Shao, L., Xue, Y. & Fang, W. 2015. A rapid learning algorithm for vehicle classification, Information Sciences, 295:395–406.

Xing, Y. & Luo, W. 2016. Facial expression recognition using local Gabor features and Adaboost classifiers *, IEEE International Conference on Progress in Informatics & Computing, 016:228–232.

Yan, J., Li, J. & Gao, X. 2011. Chinese text location under complex background using Gabor filter and SVM, Neurocomputing, 74(17):2998–3008.

Yang, M. & Sowmya, A. 2015. An underwater color image quality evaluation metric, IEEE Transactions on Image Processing, 24(12):6062–6071.

Zhao, Y., Xu, X., Chen, C. & Yang, D. 2012. Color image segmentation algorithm of rapid level sets based on HSV color space, International Conference on Information Engineering and Applications, 2012:483–489.

Zhang, H. 2014. A new shadow removal algorithm based on Susan and CIELAB Color Space, 6th International Conference on Internet Multimedia Computing and Service, 2014:222–225.

Submitted: 02/06/2017
Revised : 11/08/2017
Accepted : 27/09/2017

Journal of Engg. Research **Vol. 6 No. (3) September 2018 pp.** *53*

الكشف عن إشارة مرور المشاة في مشهد معقد باستخدام خوارزمية Adaboost ذات خصائص متعددة الطبقات

اكسو–هو وو*، رينجي هو * ويوسينج باو **
* كلية الهندسة الكهربائية، جامعة ساوث ايست، نانجينج، الصين
** كلية الهندسة الكهربائية والتشغيل الآلي، جامعة نانجينج نورمال، نانجينج، الصين

الخلاصة

من أجل تحسين دقة الكشف عن إشارة مرور المشاة في مشهد معقد، تم اقتراح طريقة الكشف عن الصور باستخدام Adaboost مع خصائص متعددة الطبقات. في الطريقة المقترحة، تم اعتماد الخصائص متعددة الطبقات لتمييز إشارات مرور المشاة، وتم استخدام خوارزمية Adaboost لاستخراج خصائص الطبقات المتعددة المميزة تلقائياً. وتتكون الخصائص متعددة الطبقات من عناصر الإنارة والتلوين، والتي تكون فيها خصائص طبقة الإنارة هي إدراك معلومات الشكل، وخصائص طبقة التلوين هي اكتساب معلومات ملونة. واستناداً إلى الخصائص المتعددة، تم اختيار المئات من المصنفات ضعيفة الكفاءة من خلال خوارزمية Adaboost لبناء مصنف قوي. وبواسطة هذا المصنف القوي، تم فحص الصور في تجربة للكشف عن إشارة مرور المشاة. أظهرت نتائج التجربة أن الخصائص متعددة الطبقات المقترحة في مساحة الألوان CIELAB تعمل على تحسين دقة الكشف عن حركة مرور المشاة بشكل كبير، وأن أداء الطرق المقترحة أفضل بكثير من الأساليب الاعتيادية.