Vehicle Recognition using extensions of Pattern Descriptors

Vehicle Recognition using extensions of Pattern Descriptors

V Keerthi Kiran[1,3*], Sonali Dash[1,4], Priyadarsan Parida[2,5]

[1]Department of Electronics and Communication Engineering, Raghu Institution of Technology, Visakhapatnam, India-531162
[2]Department of Electronics and Communication Engineering, GIET University, Gunupur, India -765022
[3]*v.keerthikiran@giet.edu*
[4]*sonali.isan@gmail.com*
[5]*priyadarsanparida@giet.edu*

Abstract. Vehicle identification and classification for still images are incredibly useful and can be extended to a range of traffic surveillance operations. Reliable and accurate recognition of vehicles is however a challenging issue due to changes in vehicle appearance and illumination difference in real time scene. In this paper, we present a simple and effective way of vehicle recognition technique based on vehicle's local texture features extraction and classification. The local features are extracted individually using the Local Binary Pattern (LBP), Median Binary Pattern (MBP), Gradient directional pattern (GDP), and Local Arc Pattern (LAP) descriptors and feed into Support Vector Machine (SVM) for classification. We also focus on vehicle classification using various color spaces like RGB, HSV, YCbCr for the texture descriptors extraction. The primary focus is to observe the effect of colour information on vehicle classification efficiency across different colour spaces. Initially, experiments are conducted for the classification of gray-level vehicle images of five different classes from the CompCars dataset. Then experiments are extended to different color spaces for the same dataset for color texture classification. The integration of different colour details increases the efficiency of vehicle classification, according to the experimental results.
Keywords: Support Vector Machine (SVM), Local Binary Pattern (LBP), Median Binary Pattern (MBP), Gradient directional pattern (GDP), and Local Arc Pattern (LAP)

1. Introduction

In a wide variety of computer vision applications, such as real-time traffic monitoring, transportation, traffic management, and the ability to distinguish vehicles in surveillance images is essential [1]. For example, video surveillance systems provide fast and accurate data for improved protection and traffic flow such as lane crossings, parked vehicles, traffic congestion, vehicle count as well as identification of the number plate, kind of vehicle, speed, and direction of the vehicles. The main objective of vehicle detection is usually to recognize potential vehicle positions in a given image for further processing tasks, marking them as a region of interest (ROI). In contrast, accurate detection of automobiles is a difficult and complicated task [2]. The challenge stems from the wide range of vehicle appearances (e.g., size, shape, and orientation) as well as reduced visibility due to camera noise or adverse weather conditions (e.g., rain, fog). Besides, the disparities in lighting, occlusion, and an unconstrained background intensify the complexity in comparison with general object detection.

Despite of these challenges, the research community and the transport sector have led to the advancement of various kinds of vehicle monitoring and detection approaches to increase the reliability of the traffic control systems and applications mentioned above. As a result, many new approaches have been suggested for the identification of vehicles in the literature that mostly focused on handcrafted or learning features [3]. Su et al. [4] suggested a method that uses rotation-invariant HOG-like features, which is also one of these methods. In [5] also most popular handcrafted features like Gabor filters, SIFT, and direction gradient have been used. Although these methods are resistant to intra-class dissimilarities, they appear to be susceptible to inter-class distinctions.

Some researchers are exploring learning feature-based approaches in vehicle detection tasks due to their excellent learning capacity. The majority of learning features based methods transform vehicle recognition into a classification problem by constructing vehicle proposals and then categorizing each one separately using classification algorithms such as support vector machines (SVM) or convolutional neural networks (CNN). The majority of current research focuses on categorizing vehicles into major groups, such as motorcycles, cars, buses, and trucks, however, but this lacks enough capabilities to meet user's needs. Many researchers have used frontal images to investigate the identification and recognition of vehicle logos to provide access to details concerning the vehicle manufacturer[6].

Yong Tang et al., [7] used Haar-like features and AdaBoost algorithms to extract features and create classifiers for vehicle recognition for on-road images. They have used histogram intersections to calculate similarities between various Local Gabor Binary Pattern (LGBP) and were used to determine the Euclidean distance in the closest neighborhood. When their model is validated with seven classes and 227 samples, a recognition rate of 92% is published. M. Hassaballah et al.,[8] have suggested an efficient vehicle identification system using clustering forests. Whereas visually identifiable codebooks are constructed using clustering forests based on the Chi-square for the features generated by the Local Binary Pattern (LBP). On the UIUC car dataset, they achieved detection rates of 99.6% for single-scale parts and 98.9% for multiscale parts, respectively. Wenjin Chu et al., [9] have developed a facial expression recognition algorithm that uses Gradient Direction Pattern (GDP), LBP, and Sparse Representation Classification (SRC). They have achieved recognition rates of 50% and 67.14% for GDP + SRC and LBP + SRC respectively and suggested that combining these pattern descriptors improves the recognition rate. To improve the performance of the face recognition system, Niaraki et al., [10] suggested a feature extraction algorithm built on the Co-occurrence Matrix of Local Median Binary Pattern (CMLMBP). The co-occurrence matrix of the processed image is determined after the Local Median Binary Pattern of the input image has been calculated. They have reported a recognition rate of 96.25% and 100%, with the ORL and Yale databases respectively. The continuation of the article is set out as follows: Literature review is discussed in section 2. Section 3 describes the proposed methodology. Experimental results are discussed in section 4. Finally, in Section 5, the conclusions are addressed.

2. Literature review

Two competing goals are expected to achieve a good texture feature, a key component of the texture classification. Local Binary Pattern (LBP) methods have been identified as one of the most prominent and commonly studied texture patterns, with a wide range of LBP variants being suggested to a variety of different applications.

2.1 Local Binary Patterns (LBP)

LBP is a type of texture descriptor that is both simple and powerful [11]. The 3 x 3 window covers each pixel of a grayscale image. The neighboring pixels are modified according to the center pixel. Where the value of neighboring pixels greater than the center pixel is given a value of 1, while the values of pixels with the value below the center pixel are 0. The thresholded neighboring pixels are then combined to form a binary code that defines the texture of the concerned pixel. As the image is divided into a grid,

a histogram of binary patterns for each pixel inside a patch is determined. As a result, a histogram is generated for each grid cell, which is then used to allocate a terrain class to the cell. As an 8-bit binary sequence can have 256 values, we can classify it using a histogram of 256 dimensions. Figure 1. shows a 3x3 pixel pattern of an image.

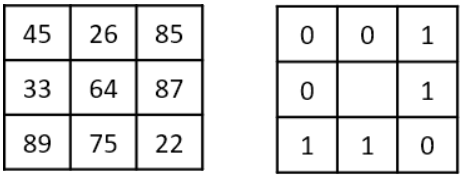


Figure 1. LBP code generation for a 3 × 3 pixel neighborhood

Binary Pattern = 00110110

2.2. Median Binary Patterns (MBP)pp0

Instead of using the central pixel always, the median binary pattern (MBP) operator maps a localized binary pattern to a threshold of pixels against the median value within the neighborhood to increase the sensitivity to microstructure and noise [12]. The median value of the patch is 120 for the example shown in Figure 2. The following equation is used to measure the MBP at pixel (i, j):

$$\mathrm{MBP(i,j)} = \sum_{k=0}^{L-1} 2^k H(b_k - \tau) \quad (1)$$

where L is the patch size (i.e., L = 9 for a patch of 3×3), and τ is the median of the local patch. When the threshold is set to the patch's median, the resulting binary pattern must contain at least five one bits, and only 256 binary patterns are possible. In this case, the MBP only produces a 256 binary pattern subset rather than the entire 9-bit spectrum of binary patterns (i.e. [0, 511]). By measuring the distribution of these patterns over the image using the MBP histogram, the texture descriptor is generated. The complete image is then modified to a 256 x 1 vector, which represents the MBP histogram that uses the median inside the local image patch.

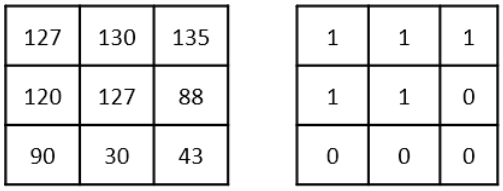


Figure 2. MBP code generation for a 3 × 3 patch

Here L=9; τ = 120

Output binary pattern MBP = 1000111102 ≡ 286.

2.3. Gradient directional pattern (GDP)

The LBP operator determines the threshold based on the center pixel's value. As a result, the LBP codes are vulnerable to noise and lighting fluctuations, as even small adjustments will cause their value to move concerning the center pixel. The direction of each pixel's gradient vector in an image is calculated first with the following equation:

$$\alpha(x,y) = tan^{-1}\left(\frac{\mathrm{G_x}}{\mathrm{G_y}}\right) \quad (2)$$

The gradient path angle (x, y) for pixels is represented here by $\alpha(x, y)$, and G_x and $G_y$are the two gradient vector elements that can be achieved with the implementation of the Sobel operator on the source image[13]. To obtain the values of G_x and G_y, the Sobel operator convolves the image with a horizontal and vertical kernel, as seen in Figure 3.

-1	-2	-1
0	0	0
1	2	1

a

-1	0	1
-2	0	2
-1	0	1

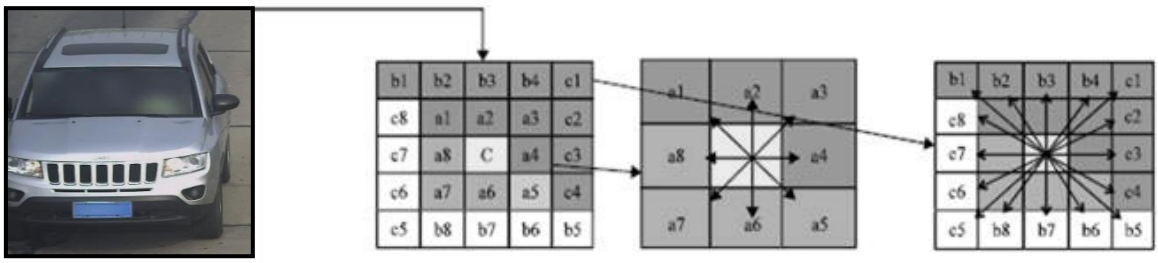
b

Figure.3. Sobel kernels (a) Horizontal kernel (b) Vertical kernel

The GDP operator selects a 3x3 neighborhood around each pixel of the image after measurement of gradient directions and quantizes the adjacent gradient directional angles to the central pixel direction angle through a threshold τ. In this case, Neighboring pixels with a gradient angle around the center angle are quantized to 1, while the rest are quantized to 0. The center pixel is then given the binary pattern value that has occurred.

2.4. *Local Arc Pattern (LAP)*

Local Arc Pattern (LAP) is a pattern that addresses nearly all of the cost concerns and shortcomings associated with LBP[14]. It computes two different patterns from the local 5x5 pixel region, that represent the local pattern at the center pixel. The LAP technique is a variant of the GDP. In all possible directions, the local pattern at a pixel describes variations in the grey color intensities of its neighbors.



(a) (b) (c) (d)

Figure 4. Local pixels notation (a) Vehicle image, (b) Local 5x5 pixels region, (c) Pixels used for pattern-1, and (d) Pixels used for pattern-2

The LAP pattern of the center pixel of the region, "C" is shown in Figure. 4, can be calculated with a 5 x 5 pixel local region. The LAP Binary patterns are based on grey color-intensity values of a(1-4), b(1-4), and c(1-4). One 4-bit binary pattern (i.e., Pattern-1 (P1)) and one 8-bit binary pattern (i.e., Pattern-2 (P2)) and make up a LAP pattern. P1 is determined from the grey color intensity values of a(1-8), while P2 is calculated from the grey color intensity values of b(1-8), and c(1-8).

P1 is limited to 24 = 16-bit combinations, while P2 is limited to 28 = 256-bit combinations. A bin is generated for each combination to count the number of times the combination occurs inside a given block. The LAP histogram for a block is combined into 16 bins for P1 and 256 bins for P2, respectively. As a result, each block's feature vector length is 16+256 (272) for the proposed method. Figure 4(c) illustrates how a(1-8) in a 4-bit binary pattern reflects the grey color attribute of the pixels. As shown in figure 4(d), the 8-bit binary pattern, b(1-8)and c(1-8) represent the corresponding grey

color value of the pixels. Figure 5. illustrates how to acquire LAP patterns from a 5x5 pixel region. The final feature vector of an image is formed by concatenating the histograms of all blocks in the image.

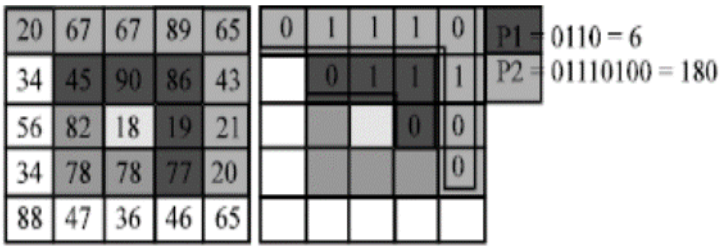


Figure 5. Illustration of LAP code generation (a) 5x5 pixels local region, (b) LAP representation of pixel 18

3. Proposed model

The aim of this work is to compare various texture descriptors in extracting the features of vehicles which in turn makes an effect on the recognition rate. Investigations are also carried how the various color spaces make out an impact on the texture descriptors for improving the classification accuracy. The proposed model is shown in Figure 6.

- Initially, in the pre-processing stage, two data sets are created out of which one dataset contains 70% training images and 30% of images are used for validation and another dataset consists of 50% training images, and 50% of images are used for validation. Then the images are resized to an RGB scale of 256x256x3 before they are applied to various descriptors.
- In the next step, for the study of various color spaces, the dataset is converted into the gray level, RGB, YCbCr, and HSV color spaces.
- The LBP, MBP, GDP, and LAP descriptors process the different color space images and extract the texture features.
- The features considered here are absolute mean, mean square error, standard deviation, and entropy.
- The extracted features are feed to SVM with different kernel functions like Radial Basis Function (RBF) and polynomial function for classification.

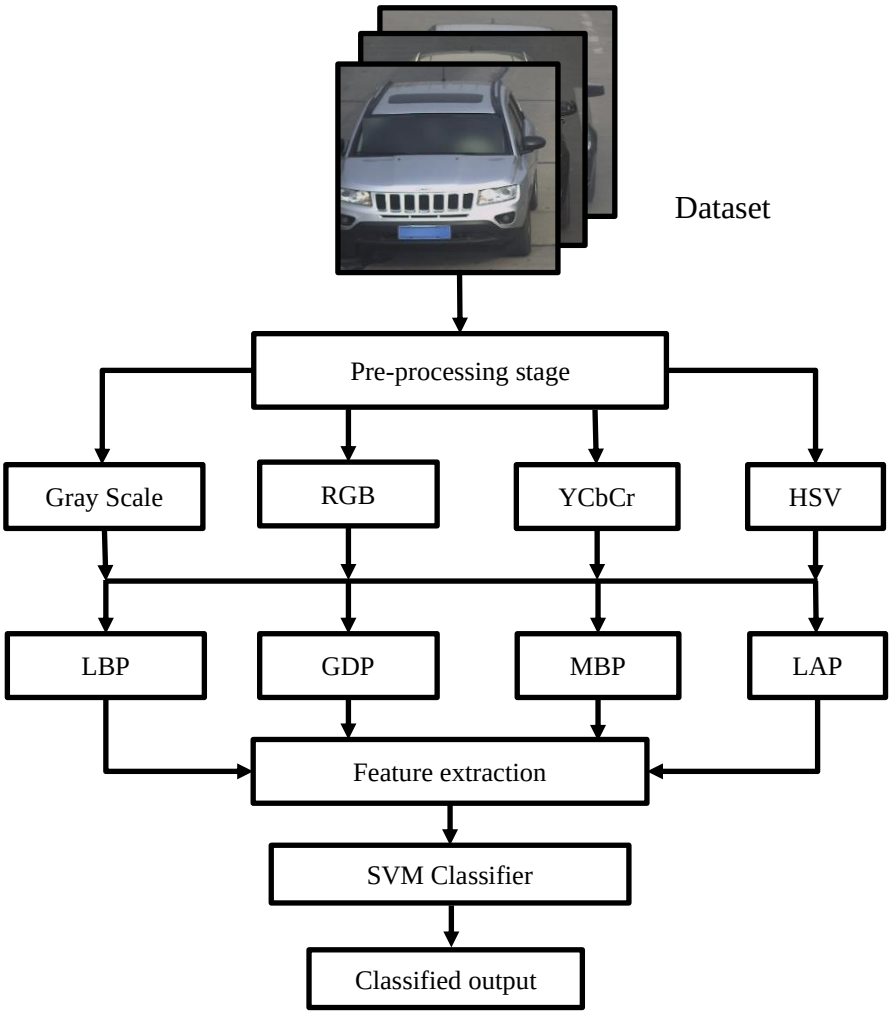
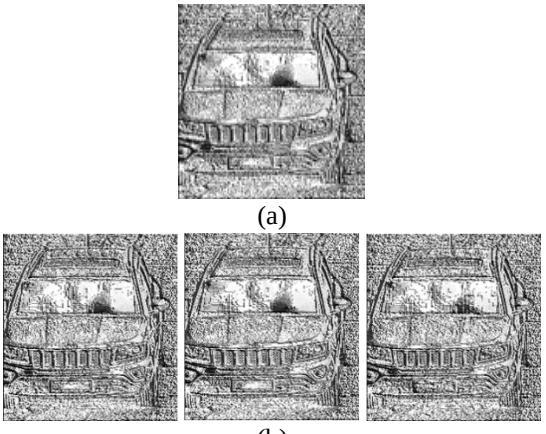
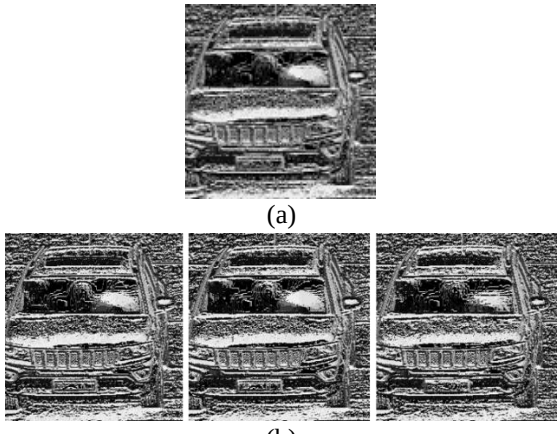


Figure 6. The block diagram of the proposed model

Figures 7, 8, 9, and 10 shows the generated images for a class of CompCars dataset [15] by the four pattern descriptors (i.e., LBP, MBP, GDP, and LAP) for various color spaces.





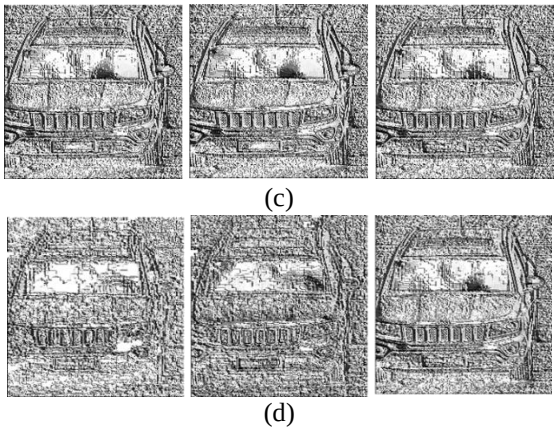


Figure 7. LBP descriptor generated images of class 'Jeep' (a) Gray level image (b) RGB (c) YCbCr (d) HSV

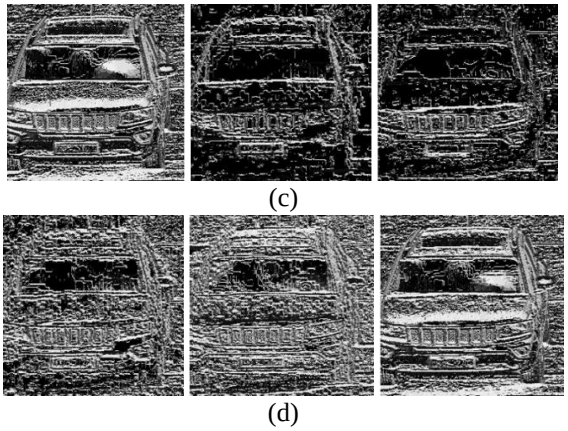


Figure 8. MBP descriptor generated images of class 'Jeep' (a) Gray level image (b) RGB (c) YCbCr (d) HSV

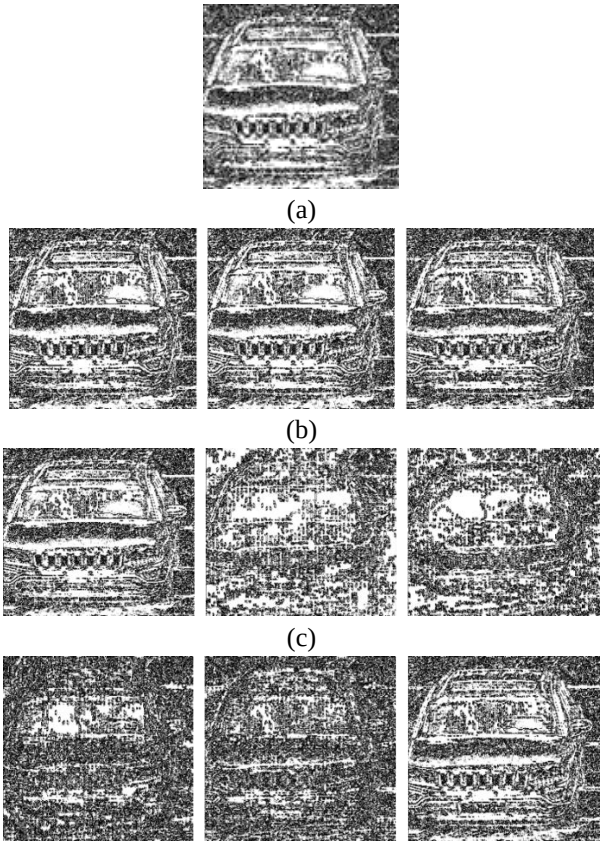


Figure 9. GDP descriptor generated images of class 'Jeep' (a) Gray level image (b) RGB (c) YCbCr (d) HSV

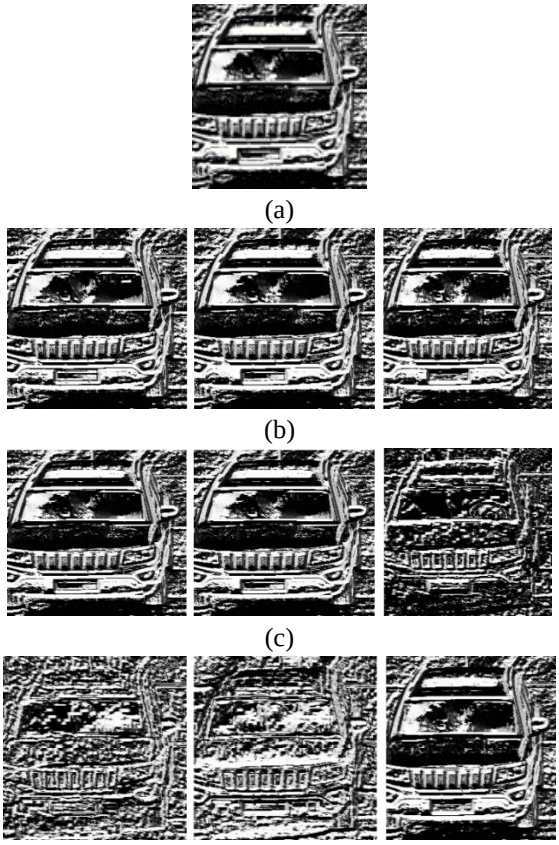


Figure 10. LAP descriptor generated images of class 'Jeep' (a) Gray level image (b) RGB (c) YCbCr (d) HSV

4. Experimental results

The proposed model is tested on the CompCars dataset which is categorized into Group-I and Group - II datasets. The Group-I dataset consists of 70% images for training and 30% for validation. Whereas

Group -II dataset contains 50% images for training and 50% for validation. In each dataset, 50 images of five different classes are considered. The experiments are conducted on MATLAB software.

Table 1 shows the classification results for various descriptors of Gray level color space images. For the Group-I dataset, the LBP and LAP descriptors achieved the highest accuracy of 37.60% with SVM using RBF kernel function with a box constraint of 1 and polynomial function of order 3 respectively. The LAP descriptors achieved the highest accuracy of 44.0% with SVM using the polynomial function of order 3 in the Group -II dataset.

Table 1. Classification accuracy of LBP, MBP, GDP, and LAP for Gray level color space images

Kernel	Group -I dataset				Group -II dataset			
	LBP	MBP	GDP	LAP	LBP	MBP	GDP	LAP
	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy
POLY 2	33.60%	34.40%	36.80%	32.00%	36.00%	28.00%	36.00%	38.67%
POLY 3	35.20%	32.00%	33.60%	37.60%	30.67%	34.67%	30.67%	44.00%
RBF 1	37.60%	20.00%	28.00%	33.60%	34.67%	17.33%	36.00%	36.00%
RBF 2	36.80%	20.00%	28.00%	34.40%	30.67%	17.33%	36.00%	32.00%

Table 2 shows the classification results for various descriptors of Gray level color space images. For the Group-I dataset, the LAP descriptor achieved the highest accuracy of 43.2% with SVM using the polynomial function of order 3. The LAP descriptors achieved the highest accuracy of 48.0% with SVM using the polynomial function of order 2.

Table 2. Classification accuracy of LBP, MBP, GDP, and LAP for R G B color space images

Kernel	Group -I dataset				Group -II dataset			
	LBP	MBP	GDP	LAP	LBP	MBP	GDP	LAP
	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy
POLY 2	40.00%	35.20%	40.00%	38.40%	38.67%	37.33%	36.00%	48.00%
POLY 3	39.20%	40.80%	28.00%	43.20%	34.67%	36.00%	32.00%	46.67%
RBF 1	30.40%	29.60%	28.00%	35.20%	30.67%	36.00%	30.67%	33.33%
RBF 2	28.80%	27.20%	30.40%	35.20%	32.00%	32.00%	30.67%	32.00%

Table 3 shows the classification results for various descriptors of Gray level color space images. For the Group-I dataset, the LAP descriptor achieved the highest accuracy of 52.0% with SVM RBF kernel function with a box constraint of 1 and 2. The LAP descriptors achieved the highest accuracy of 48.0% with SVM using a box constraint of 2.

Table 3. Classification accuracy of LBP, MBP, GDP, and LAP for HSV color space images

Kernel	Group -I dataset				Group -II dataset			
	LBP	MBP	GDP	LAP	LBP	MBP	GDP	LAP
	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy
POLY 2	45.00%	42.40%	44.80%	48.80%	38.67%	37.33%	34.67%	42.67%
POLY3	31.00%	42.40%	39.20%	42.40%	38.67%	40.00%	38.67%	42.67%
RBF 1	38.40%	47.20%	36.00%	52.00%	42.67%	44.00%	36.00%	45.33%
RBF 2	37.60%	46.40%	35.20%	52.00%	44.00%	44.00%	36.00%	48.00%

Table 4 shows the classification results for various descriptors of Gray level color space images. For the Group-I dataset, the LAP descriptor achieved the highest accuracy of 46.4% with SVM using the

polynomial function of order 2. The MBP and LAP descriptors achieved the highest accuracy of 40.0% with SVM using the polynomial function of order 2.

Table 4. Classification accuracy of LBP, MBP, GDP, and LAP for YCbCr color space images

Kernel	Group -I dataset				Group -II dataset			
	LBP	MBP	GDP	LAP	LBP	MBP	GDP	LAP
	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy	Classification Accuracy
POLY 2	42.40%	39.20%	37.60%	46.40%	38.67%	40.00%	33.33%	40.00%
POLY 3	39.20%	35.20%	36.00%	36.00%	36.00%	37.33%	36.00%	33.33%
RBF 1	38.40%	39.20%	37.60%	36.80%	38.67%	38.67%	32.00%	30.67%
RBF 2	36.80%	39.20%	36.80%	36.00%	33.33%	37.33%	33.33%	32.00%

5. Conclusion

In the field of intelligent transportation systems, accurate and robust vehicle identification and recognition remain a difficult challenge. In this paper, vehicle recognition is performed using the features generated by the pattern descriptors. The experimental results show that the LAP descriptor outperforms the other descriptors and the classification accuracy is improved by considering RGB, HSV, and YCbCr color spaces. The improvement of the classification results using deep learning is the future work to be considered.

References:

[1] Keerthi Kiran V, Parida P, and Dash S 2021 *Adv. in Intell. Sys. and Comput., Springer, Cham* **1180**
[2] Liu W, Liao S, and Hu W 2019 *Neuro. Comput.* **347**, pp 24-33
[3] Wang X, 2013 *Pattern Recog Letters* **34(1)**, pp.3-19
[4] Su A, Sun X, Zhang Y, and Yu Q 2016 *IET Comput Vision*, **10(7)**, pp.634-40
[5] Awad A I and Hassaballah M 2016 *Stud. in Comput. Intell. Springer* Int. P., Cham
[6] Yu Y, Li H, Wang J, Min H, Jia W, Yu J, and Chen C 2020 *IEEE Trans on Intell Trans Sys*
[7] Tang Y, Zhang C, Gu R, Li P, and Yang B 2017 *Multimedia Tools and Applic* **76,** 5817–32
[8] Hassaballah M, Kenk M A, and El-Henawy I M 2020 *Pattern Anal and Applic*, **23,** 1505–21
[9] Chu W, Ying Z, and Xia X 2013 *IEEE Int. Conf. on Green Comput. and Commun. and IEEE Internet of Things and IEEE Cy., Phys. and Soc. Comput.* pp. 1458-62
[10] Niaraki R J and Shahbahrami A 2019 *4th Int. Conf. on Pattern Recognition and Image Analysis (IPRIA)*, pp. 141-4
[11] Ojala T, Pietikäinen M, and Mäenpää T 2002 *IEEE Trans on Pattern Analysis and Machine Intelligence*, **24**, pp. 971-87
[12] Hafiane A, Seetharaman G, and Zavidovique B 2007 *Kamel M.,Campilho A. (eds) Image Analysis and Recognition. Lecture Notes in Comp. Science,* Springer **4633**
[13] Ahmed F, 2012 *Elect. Letters* P. 1203 – 4
[14] Islam M S and Auwatanamongkol S 2013 *Asian J. of Info. Tech., Trends Appl. Sci. Res*, **9**, pp.113-20
[15] Yang L, Luo P, Loy C C, and Tang X 2015 *Proc. of the IEEE conf. on Comp. Vision and Pattern Recognition* pp. 3973-81