

An Improved Approach for Detecting Car in Video using Neural Network Model

¹T. Senthil Kumar and ²S.N. Sivanandam

¹Department of Computer Science,
Amrita School of Engineering Amrita Vishwa Vidyapeetham, Coimbatore, India

²Department of Computer Science and Engineering,
Karpagam College of Engineering, Coimbatore-641 032, India

Abstract: The study represents a novel approach taken towards car detection, feature extraction and classification in a video. Though many methods have been proposed to deal with individual features of a vehicle, like edge, license plate, corners, no system has been implemented to combine features. Combination of four unique features, namely, color, shape, number plate and logo gives the application a stronghold on various applications like surveillance recording to detect accident percentage (for every make of a company), authentication of a car in the Parliament (for high security), learning system (readily available knowledge for automobile tyro enthusiasts) with increased accuracy of matching. Video surveillance is a security solution for government buildings, facilities and operations. Installing this system can enhance existing security systems or help start a comprehensive security solution that can keep the building, employees and records safe. The system uses a Haar cascaded classifier to detect a car in a video and implements an efficient algorithm to extract the color of it along with the confidence rating. An AdaBoost trained classifier is used to detect the logo (Suzuki/Toyota/Hyundai) of the car whose accuracy is enhanced by implementing SURF matching. A combination of blobs and contour tracing is applied for shape detection and model classification while number plate detection is performed in a smart and efficient algorithm which uses morphological operations and contour tracing. Finally, a trained, single perceptron neural network model is integrated with the system for identifying the make of the car. A thorough work on the system has proved it to be efficient and accurate, under different illumination conditions, when tested with a huge dataset which has been collected over a period of six months.

Key words: Different classes, analysis, bias function, neural network, dimensionality, feature, SURF

INTRODUCTION

In real time traffic there are different objects of interest. The objects have characteristic and have a high probabilistic behavior in interaction with other objects in real time. The object characteristics can be analyzed as features of interest and can be modeled in real time video.

The features extracted can be analyzed using several approaches. The primary objective of feature extraction aims at achieving dimensionality reduction. The features analyzed vary from domain to domain.

There are lots of classification mechanisms in research. A proper classification methodology can be applied for classifying the objects belonging to different classes.

Previous research: The algorithm which is used for feature extraction is K nearest neighbor local margin maximization. The comparison is done using with Linear Discriminant Analysis (LDA), Principal Component Analysis (PCA) and Maximum Margin Criterion (MMC). Many feature extraction algorithms

have been proposed in the last decades. Principle Component Analysis (PCA), as a well-known unsupervised feature extraction method, aims to extract a subspace in which the reconstruction error can be minimized, or equivalently the variance of the total papered data points can be maximized (Pan *et al.*, 2009). This method performs better in high dimensionality. It reduces error rate and complexity by selecting only limited K nearest neighbors for each point. The approach will not work with high dimensional data.

Improve Harris corner detection and descriptor of SIFT method are applied to extract features from images. First, the method of Harris corner detection is used for finding the interest points from the two images to be matched, in which an extreme tactics is adopted for exactly determining the interest points (Zhang, 2004; Zhang *et al.*, 2009). This is followed by adopting a sub-block checking method for eliminating the cluster and reduce the 15 number of interest points. Sobel operator shows good results in edge extraction and inhabits the noisy better than the other two methods.

Corresponding Author: T. Senthil Kumar, Department of Computer Science, Amrita School of Engineering Amrita Vishwa Vidyapeetham, Coimbatore, India

General optical flow algorithm detects noisy and irrelevant features. The extended version of this algorithm extracting low level features. To overcome the problem due to complex scenes it introduces a frame jump technique along with thresholding of unwanted features. The demerits of this method is integrated with other feature tracking algorithm will increase the possibility of classifying variety of human actions (Choi, 2010).

The polygonal line algorithm applied to extract nonlinear feature extraction, in which the nonlinearities among the multivariable data can be described by a set of local linear models. The polygonal line algorithm can produce robust and accurate nonlinear curve estimation for different multivariate data types and it is helpful in reducing the computation complexity for existing principal curve approaches when the sample size is large (Zhang, 2004). This approach decrease noise effect in estimation accuracy.

The Table 1 below shows the comparative study between SIFT and SURF approaches (Juan and Gwun, 2008; Lowe, 2004; CED, 2012).

The Table 2 below shows the comparison of different classifier mechanisms (Smach *et al.*, 2005; Morra *et al.*, 2010; Zhu *et al.*, 2009).

Proposed approach: The proposed approach combines the approach of combining surf features with Neural network model is shown in Fig. 1. The extracted features include color, shape, logo and number plate region (Shi and Tomasi, 1994; Sebe *et al.*, 2003). The real time detection analysis of features forms the basis in the proposed model (Stojmenovic, 2005; Jang and Turk, 2011; Jazayeri *et al.*, 2011).

Object detection using haar like features and SURF features: The surf features is combined with haar like features for detecting objects (AT, 2011).

Step 1: Object detection using haar wavelet and confining roi.

The object (car) detection is done by Haar wavelet method. Haar wavelet technique uses a haarcascade car XML file which detects all rigid bodies. The detected car region is set as our ROI.

Step 2: Getting rgb and hsv value for color.

The central pixels (area = 5x5) of ROI is considered for identifying the color.

The RGB value of the pixel is converted to a HSV (Hue-Saturation-Value) as HSV is better at handling lighting differences and it gives an easy to use color value.

Step 3: Determining the color.

The approximate color is found based on HSV components.

User defined function used-getcolortype (int H, int S, int V).

Step 4: Finding confidence percentage of color.

Confidence percentage=initial confidence * (tally max count * 100/pixels) Tally max count=number of pixels found using each color type.

Number plate detection:

Step 1: Smoothing.

Apply Gaussian technique for smoothing.

Step 2: Morphological operations: Tophat

Top hat detects the License plate which will be lighter than its neighbors.

It exaggerates those portions which are lighter than the surroundings.

Assumption-The license plate mostly contains light portions.

Step 3: Thresholding.

The result is to be thresholded so that the light regions are more prominent for contour extraction. Thresholding has been done for all pixels above 128-255.

Step 4: Contour extraction and applying restrictions to roi.

The area between. 600 and 5000 and ratio between. 3.0 and 8.0 is the threshold set for contour extraction.

Logo detection and SURF matching: The logo recognition has been completed for a three class of cars using adaboost classifier (Burkhard *et al.*, 2011).

Step 1: Creating descriptive and vec files.

Creating the descriptive file of positive and negative samples (collected manually) and a vec file is created. A vec file is created using the create samples utility provided with opencv package (NB, 2010).

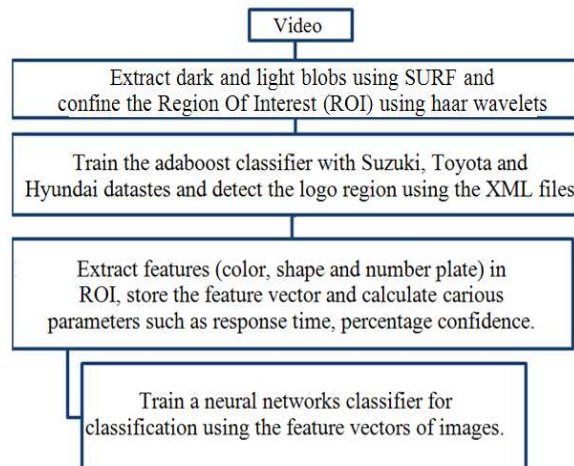


Fig. 1: Proposed Model for car recognition

Table 1: Comparative study between SURF and SIFT

Parameters	SIFT	SURF
Time	Very slow (response time/frame=2.1537e+007ms) but finds more matches (271)	Faster than SIFT (response time/frame =3362.86ms), lesser matches (186)
Repeatability	Good repeatability (21%)	Has the largest repeatability (31%)
Affine transformations	47% Efficiency	54% Efficiency

Table 2: Comparison of different classifiers

Classifier	Advantages	Disadvantages	Applications
Adaboost (Type: Boosting)	<ol style="list-style-type: none"> Does a good job of managing overfit training data (learns unknown patterns and hence has a good generalization power). Easier to implement Yields smaller test error (test error of AdaBoost actually starts to increase after a few iterations, then levels of around 0.53.) 	<ol style="list-style-type: none"> Suboptimal solution For noisy data the results are not better. 	<ol style="list-style-type: none"> Text classification OCR Image retrieval Medical diagnosis Fraud Detection Drug Discovery Electrical Power Monitoring
Support Vector Machines (SVM) (Type: Unsupervised learning)	<ol style="list-style-type: none"> SVM has a high generalization power, especially for high dimensional features(less over fitting). More stable than AdaBoost 	<ol style="list-style-type: none"> High algorithmic complexity and extensive memory requirements in large scale tasks. Forces a limitation in speed. Choice of the kernel is difficult (Gaussian/polynomial/more elaborate ones) Choice of kernel parameters (like sigma in Gaussian in) is a challenge. 	<ol style="list-style-type: none"> Geographical application. content based image retrieval Texture Classification E-learning Text classification
Neural approach (Type: Supervised learning)	<ol style="list-style-type: none"> Highly robust Since normalization is incorporated directly in the weights of the input network preprocessing is not required. It can identify unknown patterns if trained properly. 	<ol style="list-style-type: none"> Difficulty in identifying an appropriate neural model. Misclassification treatment is not effective enough. More research is to be done on feature variable selection. 	<ol style="list-style-type: none"> Finger print identification Voice recognition Face recognition Pattern matching Character recognition Signature recognition Classification of objects in scientific /research areas

Table 3: Templates for shape detection and classification

	Template 1	Template 2	Template 3
Suzuki	Swift	Dezire	SX4
Hyundai	Verna	Santro	I10
Toyota	Innova	Corolla	Avensis

Step 2: Train the adaboost classifier to obtain logo xml files.

The vec file obtained is converted to an XML file for processing.

Step 3: Logo detection using xml files.

XML files (which comprises of coordinate values.) of Suzuki, Hyundai and Toyota obtained as a result of classification is used to draw the ROI around the respective logos.

Content of XML-image name, the number of objects to be detected in the image, x-y coordinates of the location of the object in the image.

Step 4: SURF matching of detected logo with logo templates.

SURF detects ipoints (interest points) in the test data and the template and performs matching between these two. It records the number of matches found if any.

The logo detection results are better than the existing approaches (Sulehria and Zhang, 2007; Psyllos *et al.*, 2010a).

Shape extraction:

Step 1: Detection of the largest blob in the templates

Step 2: Draw contour around the largest blob detected

Step 3: Repeat Step 1 and 2 for a test image.

Step 4: Perform contour matching with test and template data.

The following templates as shown in Table 3 are chosen for shape detection and classification.

MATERIALS AND METHODS

The neural network has been designed for training and testing for recognizing the car in the video (Smach *et al.*, 2005).

Step 1: Parameters calculated

Detection time per frame, average detection time (for all features), confidence percentage (color) per frame, Average confidence percentage (color), error percentage (shape), No. of SURF matches with each logo template, percentage of frames with number plate detected

Step 2: Obtained the feature vector

< Color, number plate status, logo, shape template, % shape error >

Step 3: Collected training samples for NN

3 images of 3 models (Ex: Swift, Dezire, SX4 for Suzuki) for each logo were chosen for training. Their corresponding feature vectors were obtained.

Step 4: Train a single layer perceptron neural network model

The initial weights is 0. The inputs are 6 <feature vector > including a constant bias of 1. The classes used were -1, 0 and 1 to classify cars as Suzuki, Hyundai or Toyota. The error for each training sample to change the weights were found using the classes and the dot product of the weights and the feature vector including bias.

RESULTS

The proposed approach has been tested on datasets that have been shot using a PowerShot SX110 IS Canon

camera with 9.0 Mega Pixels and 10X Optical Zoom lens (10X IS). The image analysis characteristics is also considered for feature analysis Image Analysis Labs, 2010. The implementation has been completed in opencv with dotnet framework Structural Analysis and Shape Descriptors, 2010 (AB, 2010). The results for the different brand of cars is presented belowin Fig. 2-4.

The results obtained have been tested on videos containing yellow number plate on yellow black ground and similarly white number plate on white background and for front and back view of the cars as in Fig. 5 and 6.

The number plate detection has been tested for the top view and back view of cars and detection has been good as in other cases (Pysillos *et al.*, 2010b) as shown in Fig. 7 and 8. The comparisons are tabulated in Table 4 and 5.

Training of the single perceptron neural network: A sample of 27 images (3 images of 3 models for each logo) was collected for training and their corresponding feature vectors were obtained.

Input: Training data

Processing: The initial weights is 0. The inputs are 6 <feature vector > including a constant bias of 1. The classes used were -1, 0 and 1 to classify cars as Suzuki, Hyundai or Toyota. The error for each training sample to change the weights was found using the classes and the dot product of the weights and the feature vector including bias. Table 6 shows the different classes for classifier.

Table 4: Tabulated results of 3 different data sets

No.	Parameter	Dataset1 (Suzuki)	Dataset2 (Hyundai)	Dataset3 (Toyota)
1.	No. of frames:	30	30	30
Color				
2.	Approx. color of car:	White	Blue	Blue
3.	Avg. % confidence:	82.2667%	92.667%	79.4%
4.	Avg detection time:	712.965 ms	14.1667 ms	844.263 ms
Number plate				
5.	Detection status:	Detected	Detected	Detected
6.	No. of frames with no. plate detection:	30	30	11
7.	% of frames with no. plate detection:	100%	100%	34%
8.	Avg. Detection time:	1431.44 ms	655.946 ms	343.939 ms
Logo				
9.	Logo found to be:	Suzuki	Hyundai	Toyota
10.	Matching Template No.:	3	4	9
11.	Total No. of SURF matches with all templates:	54	425	5
12.	Avg. no. of SURF Matches with the approx. template:	1.80000	14.1667	0.2500
13.	Avg. detection time:	639.116 ms	644.405 ms	434.365 ms
Shape details				
14.	Template match with:	3	4	9
15.	Least Error % with template:	2.28465%	6.35295%	11.589%
16.	Approx. Shape Model:	Sx4	Verna	Avensis

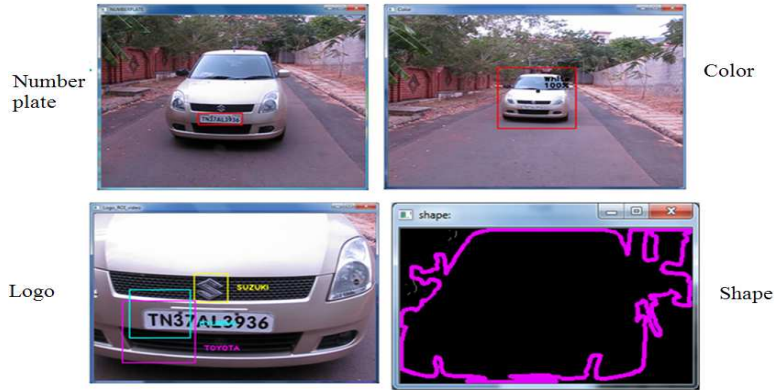


Fig. 2: Results for extracted features using the proposed model for Suzuki

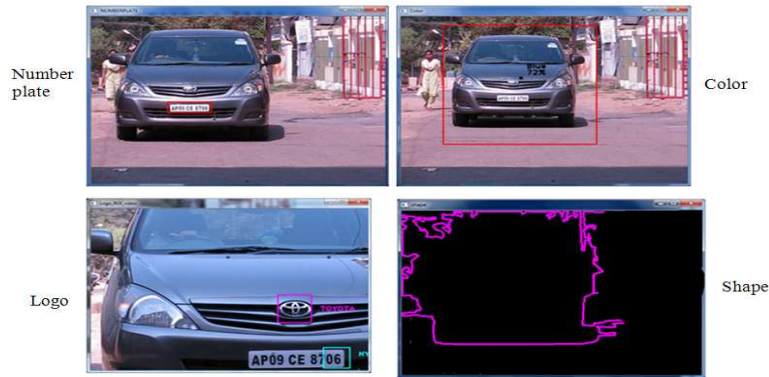


Fig. 3: Results for extracted features using the proposed model for Toyota

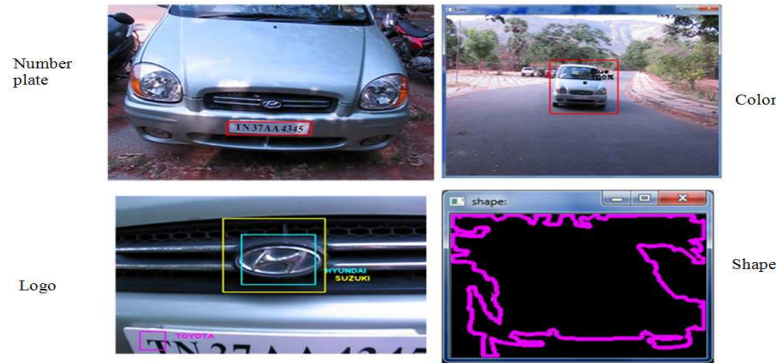


Fig. 4: Results for extracted features using the proposed model for Hyundai

Table 5: Inference of tested data sets for zoom and different views

	Zoomed in	Zoomed out	Normal
Color	No	Yes	Yes
Logo	Yes	No	Partial
Number plate	Yes	Yes	Yes
Shape	No	Partial	Partial
	Front view	Back view	Top view
Color	Yes	Yes	No
Logo	Yes	Yes	Yes
Number plate	Yes	Yes	Yes
Shape	Yes	Partial	No

Table 6: Classes for NN classifier

Class	Value
Suzuki	-1
Hyundai	0
Toyota	1

Table 7: Final weights after training

Weight	W[0]	W[1]	W[2]	W[3]	W[4]	W[5]
Value	-1.5	-0.5	-1.5	4.875	0	0.067



Fig. 5: Yellow number plate on yellow background



Fig. 6: White number plate on white

Table 8: The values of weights after each epoch

Epoch	No of errors per epoch	RMS Error	W[0]	W[1]	W[2]	W[3]	W[4]	W[5]
1	10	0.6085	0.0	0.0	0.0	0.125	0	0.00092
2	11	0.7934	0.5	-0.5	0.5	0.875	0	0.01162
3	11	0.7934	1.0	0.0	1.0	1.625	0	0.22325
4	12	0.8819	0.5	0.5	0.5	2.125	0	0.02861
5	10	0.6938	0.0	0.0	0.0	2.125	0	0.02732
6	10	0.6938	-0.5	-0.5	-0.5	2.125	0	0.02604
7	10	0.6938	0.0	0.0	0.0	2.625	0	0.03489
8	10	0.6938	-0.5	-0.5	-0.5	2.625	0	0.03361
9	10	0.6938	0.0	0.0	0.0	3.125	0	0.04246
10	10	0.6938	-0.5	-0.5	-0.5	3.125	0	0.04118
11	10	0.6938	0.0	0.0	0.0	3.625	0	0.05003
12	10	0.6938	-0.5	-0.5	-0.5	3.625	0	0.05760
13	10	0.6938	0.0	0.0	0.0	4.125	0	0.05632
14	10	0.6938	-0.5	-0.5	-0.5	4.125	0	0.05632
15	10	0.6938	-1.0	0.0	-1.0	4.125	0	0.05504
16	10	0.6938	-0.5	-0.5	-0.5	4.625	0	0.06389
17	10	0.6938	-1.0	0.0	-1.0	4.625	0	0.06261
18	10	0.6938	-0.5	-0.5	-0.5	5.125	0	0.07146
19	10	0.6938	-1.0	0.0	-1.0	5.125	0	0.07018
20	9	0.5773	-1.5	-1.5	-1.5	4.875	0	0.06704



Fig. 7: Back view of the car



Fig. 8: Front view of the car

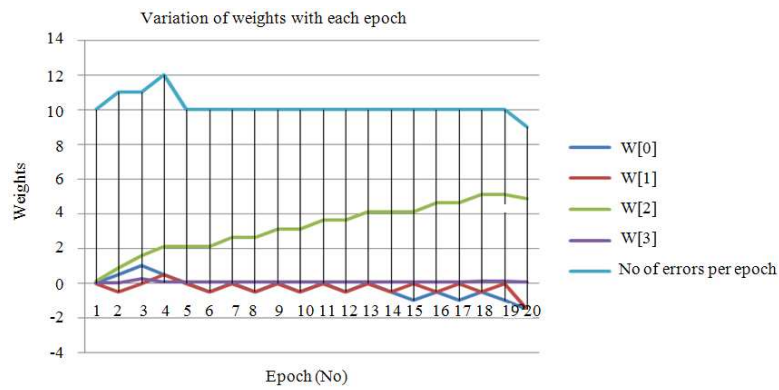


Fig. 9: Graph showing variation of weights with each epoch

Table 9: Feature vector samples for 3 different data sets

Dataset	Color	Number plate	Logo	Shape template	Shape error
Hyundai	1	1	0.25	0.4	0.06352
Suzuki	1	1	0.50	0.7	0.10435
Toyota	1	1	0.75	0.9	0.04270

Output: Weights corresponding to each entity in the feature vector is tabulated in Table 7-9. The performance is represented in graph in Fig. 9.

Initial Weights: $w[0] = 1$ (bias) $w[1] = w[2] = w[3] = w[4] = w[5] = 0$.

```

c:\Users\Ranjan\Documents\Visual Studio 2008\Projects\integrated_image\Debug\integrated_ima...
LMS ALGORITHM RUNNING
-----
1 10 0.608581 0 0 0 0.125 0 0.000923976
2 11 0.793492 0.5 -0.5 0.5 0.875 0 0.0116246
3 11 0.793492 1 0 1 1.625 0 0.0223252
4 12 0.881917 0.5 0.5 0.5 2.125 0 0.0286117
5 10 0.693889 0 0 0 2.125 0 0.0273286
6 10 0.693889 -0.5 -0.5 -0.5 2.125 0 0.0260455
7 10 0.693889 0 0 0 2.625 0 0.0348982
8 10 0.693889 -0.5 -0.5 -0.5 2.625 0 0.0336151
9 10 0.693889 0 0 0 3.125 0 0.0424678
10 10 0.693889 -0.5 -0.5 -0.5 3.125 0 0.0411847
11 10 0.693889 0 0 0 3.625 0 0.0500374
12 10 0.693889 -0.5 -0.5 -0.5 3.625 0 0.0487543
13 10 0.693889 0 0 0 4.125 0 0.057607
14 10 0.693889 -0.5 -0.5 -0.5 4.125 0 0.0563239
15 10 0.693889 -1 0 -1 4.125 0 0.0550408
16 10 0.693889 -0.5 -0.5 -0.5 4.625 0 0.0638935
17 10 0.693889 -1 0 -1 4.625 0 0.0626104
18 10 0.693889 -0.5 0.5 -0.5 5.125 0 0.0714631
19 10 0.693889 -1 0 -1 5.125 0 0.07018
20 9 0.57735 -1.5 -0.5 -1.5 4.875 0 0.0670489
-----

-----FEATURE VECTOR-----
COLOR      : 1 for black/blue/white and 0 otherwise
NUMBER PLATE : 1 if detected and 0 otherwise
LOGO       : 0.25 for Suzuki, 0.5 for Hyundai, 0.75 for Toyota and 0 for none
TEMPLATE   : Gives the template number of the image with which shape matches the most
SHAPE_ERROR : Gives the minimum error encountered when compared with all templates

COLOR  NP    LOGO    TEMPLATE SHAPE_ERROR
1      1     0.25   3         6.80601

-----CLASSIFICATION OUTPUT-----
THE TEST DATA BELONGS TO CLASS SUZUKI WITH SOP VALUE : -1.82491
    
```

Fig. 10: Results of the car recognition using proposed model

Final weights after training: The feature vector designed is as given below:

- Color: 1 if color=black/blue/white and 0 otherwise
- Number plate: 1 if it is detected and 0 otherwise
- Logo: 0.25 for Suzuki, 0.5 for Hyundai and 0.75 for Toyota
- Shape template: Template number with which the shape matches

Shape error: error percentage obtained from shape match.

DISCUSSION

After training with 27 images of different models, the final weights were obtained as -1.5, -0.5, -1.5, 4.875, 0 and 0.067. The test feature vectors of videos were given as input and they were classified into their appropriate classes with the help of the final weights obtained after training. Figure 10 shows the results after recognition stage.

In the generated results upto 0.5% of Root mean square error is allowed for classification.

CONCLUSION

For a live streaming video, the color (approximate) of the car can be displayed along with its confidence percentage, provided the video is not too zoomed in (few 100 cms near the car). If the car is really close to the camera then the color detection code does not work properly as the Haar cascade file fails to identify the car. The program also handles multiple objects and it can detect upto 4 cars along with their colors and confidence percentages. Color values are restricted to eleven.

The proposed system is designed to detect the number plate of an Indian car. For detecting vehicles of other nationalities, blob filtering values have to be altered. Scope can be extended to detect license plates of other vehicles too by changing the restrictions defined during contour extraction. The algorithm works well for number plates with lighter shade backgrounds other than white also.

Logo detection has been done for Hyundai, Suzuki and Toyota cars. SURF is combined with Haar to overcome the drawbacks of Haar and produce results with better classification accuracy (Pavania *et al.*, 2010). This module handles videos which have a well-zoomed version of a moving car better than a zoomed-out version.

The proposed work a novel method to extract the shape of the object by combining blobs and contours. It also helps to decide if it is the extracted shape is that of a car or not, by comparing it with template of car images. The feature vector is built and verified with the Single Perceptron Neural Network Classifier that has been built.

The application developed can be used for surveillance cameras. It is useful for detecting essential features of a car with which a lost or stolen vehicle can be identified. The feature vector generated is useful in recognizing and distinguishing cars from one other. The feature vector could take fuzzy values that can also be tested in future (Lin and Wang, 2007).

ACKNOWLEDGMENT

The researcher would like to thank his students Manjuluri Anusha, Ranjani P for their contribution in this study.

REFERENCES

- AB, 2010. OSS and multi-touch experimentation. Amit's Blog.
- AT, 2011. Electronics, robotics, image processing, embedded, hacking, RF electronics. Achu's TechBlog.
- Burkhard, T., A.J. Minich and C. Li, 2011. Vehicle logo recognition and classification: Feature descriptors vs. shape descriptors.
- CED, 2012. Computer vision and image processing. Christopher Evans.
- Choi, J., 2010. Realtime On-Road Vehicle Detection with Optical Flows and Haar-like feature detector. University of Illinois at Urbana-Champaign.
- Jang, D.M. and M. Turk, 2011. Car-Rec: A real time car recognition system. Proceedings of the IEEE Workshop on Applications of Computer Vision, Jan. 5-7, IEEE Xplore Press, Kona, pp: 599-605. DOI: 10.1109/WACV.2011.5711559
- Jazayeri, A., H. Cai, J.Y. Zheng and M. Tuceryan, 2011. Vehicle detection and tracking in car video based on motion model. IEEE Trans. Intell. Transportation Syst., 12: 583-595. DOI: 10.1109/TITS.2011.2113340
- Juan, L. and O. Gwun, 2008. A Comparison of SIFT, PCA-SIFT and SURF. Int. J. Image Process., 3: 143-152.
- Lin, C.F. and S.D. Wang, 2007. Fuzzy support vector machines. Department of Electrical Engineering.
- Lowe, D., 2004. Distinctive Image Features from Scale-Invariant Keypoints. Int. J. Comput. Vis., 60: 91-110. DOI: 10.1023/B:VISI.0000029664.99615.94
- Morra, J.H., Z. Tu, L.G. Apostolova, A.E. Green and A.W. Toga *et al.*, 2010. Comparison of AdaBoost and support vector machines for detecting Alzheimer's disease through automated hippocampal segmentation. IEEE Trans. Med. Imag., 29: 30-43. PMID: 19457748
- NB, 2010. Image processing and computer vision with open CV. Nashruddin's Blog.
- Pan, F., J. Wang and X. Lin, 2009. Feature extraction algorithm based on k nearest neighbor local margin. Proceedings of the Chinese Conference on Pattern Recognition, Nov. 4-6, IEEE Xplore Press, Nanjing, pp: 1-5. DOI: 10.1109/CCPR.2009.5344145
- Pavania, S.K., D. Delgado and A.F. Frangia, 2010. Haar-like features with optimally weighted rectangles for rapid object detection. Patt. Recogn., 43: 160-172. DOI: 10.1016/j.patcog.2009.05.011
- Psyllos, A., C.N. Anagnostopoulos, E. Kayafas, 2010a. Vehicle model recognition from frontal view image measurements. Comput. Standards Interf., 33: 143-151. DOI: 10.1016/j.csi.2010.06.005
- Psyllos, A.P., C.N.E. Anagnostopoulos and E. Kayafas, 2010b. Vehicle logo recognition using a SIFT-Based enhanced matching scheme. IEEE Trans. Intell. Transportation Syst., 11: 323-328. DOI: 10.1109/TITS.2010.2042714
- Sebe, N., Q. Tian, E. Loupias, M. Lew and T.S. Huang, 2003. Evaluation of salient point techniques. Image Vis. Comput., 21: 1087-1095. DOI: 10.1016/j.imavis.2003.08.012
- Shi, J. and C. Tomasi, 1994. Good features to track. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Jun. 21-23, IEEE Xplore Press, Seattle, WA., pp: 593-600. DOI: 10.1109/CVPR.1994.323794
- Smach, F., M. Atri, J. Miteran and M. Abid, 2005. Design of a neural networks classifier for face detection. World Acad. Sci. Eng. Technol.
- Stojmenovic, M., 2005. Real time object detection in images based on an AdaBoost machine learning approach and a small training set. M.Sc. Thesis, Carleton University, Ottawa.

- Sulehria, H.K. and Y. Zhang, 2007. Vehicle logo recognition using mathematical morphology. Proceedings of the 6th WSEAS International Conference on Telecommunications and Informatics, (TI' 07), ACM Press, USA., pp: 95-98.
- Zhang, F., 2004. A polygonal line algorithm based nonlinear feature extraction method. Proceedings of the 4th IEEE International Conference on Data Mining, Nov. 1-4, IEEE Xplore Press, pp: 281-288. DOI: 10.1109/ICDM.2004.10113
- Zhang, X., G. He and J. Yuan, 2009. A rotation invariance image matching method based on harris corner detection. Proceedings of the 2nd International Congress on Image and Signal Processing, Oct. 17-19, IEEE Xplore Press, Tianjin, pp: 1-5. DOI: 10.1109/CISP.2009.5304497
- Zhu, J., H. Zou, S. Rosset and T. Hastie, 2009. Multi-class AdaBoost. Stat. Interf., 2: 349-360.