A Novel Approach to Automatic Road-Accident Detection using Machine Vision Techniques

Vaishnavi Ravindran
Department of Computer Science
R.V. College of Engineering
Bangalore, India

Lavanya Viswanathan
Department of Computer Science
R.V. College of Engineering
Bangalore, India

Dr. Shanta Rangaswamy
Department of Computer Science
R.V. College of Engineering
Bangalore, India

Abstract—**In this paper, a novel approach for automatic road accident detection is proposed. The approach is based on detecting damaged vehicles from footage received from surveillance cameras installed in roads and highways which would indicate the occurrence of a road accident. Detection of damaged cars falls under the category of object detection in the field of machine vision and has not been achieved so far. In this paper, a new supervised learning method comprising of three different stages which are combined into a single framework in a serial manner which successfully detects damaged cars from static images is proposed. The three stages use five support vector machines trained with Histogram of gradients (HOG) and Gray level co-occurrence matrix (GLCM) features. Since damaged car detection has not been attempted, two datasets of damaged cars - Damaged Cars Dataset-1 (DCD-1) and Damaged Cars Dataset-2 (DCD-2) – was compiled for public release. Experiments were conducted on DCD-1 and DCD-2 which differ based on the distance at which the image is captured and the quality of the images. The accuracy of the system is 81.83% for DCD-1 captured at approximately 2 meters with good quality and 64.37% for DCD-2 captured at approximately 20 meters with poor quality.**

Keywords—Feature extraction; Image denoising; Machine vision; object detection; Supervised learning; Support vector machines

I. Introduction

A novel approach using image processing and machine learning tools to detect damaged cars from static images, which can be used to detect a road accident automatically is proposed. Detection or recognition of damaged cars falls under the category of object detection. Object detection or recognition using Machine Vision is achieved in two stages [1]. The first stage is feature extraction in which features common to instances from the object category are extracted from the corresponding images. The second stage includes training of a learning model like Support Vector Machines [2], Neural Networks [3] and AdaBoost [4] with the extracted features [1]. Principles used in most object detections do not work for detecting damaged instances of an object category since the damaged instances do not have the commonly extracted characteristics like shape, edges, Histogram of Gradients in common.

In this paper, a supervised learning method that detects damaged cars by making using of two facts – the state-of-the-art vehicle detection classifiers will not detect a damaged car and most damaged cars still have one or more car parts intact, is proposed.

The experimental results obtained show that the proposed approach gives promising results when tested on two different datasets of damaged cars which differ based on the quality, distance of the camera from the object and number of objects in an image. The two datasets were compiled for the sake of the project from various sources. The proposed method can be extended to other vehicles as a part of future work.

The work done includes three contributions. The first contribution includes proposing a novel approach to automatic road accident detection. The second contribution includes a supervised learning method that detects damaged cars from static images, a class of object that has not been detected so far using the techniques of machine vision. The third contribution includes the release of two public datasets of damaged cars- Damaged Cars Dataset-1 (DCD-1) and Damaged Cars Dataset-2 (DCD-2).

This paper is organized as follows. Section II describes related work. The proposed method is explained in Section III. The experimental results and analysis for two different datasets are presented in Section IV. Section V presents Conclusions.

II. Related Work

The Global status report on road safety 2015 [5] shows that the total number of deaths caused due to road accidents is at 1.25 million a year. One of the main reasons for fatalities from these accidents is a delay in reporting the accidents to near-by emergency health centres and delay in an ambulance reaching the accident location. Such a delay can be can be reduced if there is automatic detection and reporting of the accidents to emergency help centres. Most of the prevalent state-of the-art methods use sensor technology to detect road accidents. In [6],[7] the use of sensors present inside the vehicle including accelerometers, GSM and GPS modules is made to detect unusual movements and angles of the car to indicate an accident. In [8] the use of sensors like magneto resistive sensors is made outside the vehicle, installed on the roads. In [9], MMA621010EG is a proven special car accident sensor which is integrated XY-axis accelerometer and built-in serial peripheral interface SPI bus. The variations from this sensor is detected and through the GPS software fitted in the vehicle communication is made with the satellite. The latitude and longitude values are sent to the centralized server for contacting the emergency service. The drawback with using sensor technology is that the sensors can get damaged in the accident. One way to overcome this drawback is by making use of the surveillance cameras installed in traffic junctions and highways. The static images or video footage from these cameras can be used in detecting the presence of a damaged vehicle which in turn indicates the occurrence of an accident. A new approach based on vision based object detection is presented in this paper which detects the presence of a damaged car from static images.

As seen in [10][11] detection of damaged buildings and roads involves working with images of the object prior and post damage. This however cannot be applied to damaged car detection for the purpose of accident detection.

III. Proposed Method

The proposed method is a supervised learning one which works as a binary classifier distinguishing between images containing a damaged car as class 1 and images not containing it as class 0. Instances of damaged cars do not have anything in common due to loss of shape, edges and intensity gradients. Hence using the usual vision-based object detection methods where HOG, Haar, Gabor and SURF features are used to train SVM, AdaBoost and neural networks [12] will not achieve detection. However, a feature that most damaged cars do have in common is the presence of at least one car part. Our classifier is based on this fact. But training a classifier that detects car parts alone will not achieve successful detection of damaged cars since cars without any damage also show the presence of car parts. Hence there is a need to differentiate between cars that are damaged and cars that are not in addition to the step involving detection of car parts. When the state-of-the-art classifiers developed so far [12] for detection of vehicles and cars in specific were tested, results showed that they failed to detect most damaged cars. This is the second fact that forms the basis of the method developed in this project.

The input images can be divided into three types - images of damaged cars (type 1), images of undamaged cars (type 2) and images of all other objects and scenes (type 3). The classifier built should now work as a binary classifier which distinguishes between images containing a damaged car (type 1) as class 1 and images not containing it (type 2 and type 3) as class 0.

The realization of the system is done in three different stages which are combined into a single framework in a serial manner as shown in Fig. 1. Prior to the first stage is the pre-processing of images. The first stage is the vision based detection of undamaged cars using a SVM trained with HOG [26]. The SVM works as a binary classifier detecting the presence of a car without any damage and thus separates type 2 (class 1) from type 1 and type 3 (class 0). Since damaged cars are very close in appearance to undamaged cars results showed stage 1 misclassifies a considerable number of damaged cars as class 1, that is, as undamaged cars. Hence in order to improve the performance of stage 1, a binary classifier that detects the presence of damaged texture from all images classified as undamaged cars in the previous stage was introduced, this is stage 2. It reduces the number of false positives from stage 1. The detection of damaged texture at this stage is done by training a SVM with GLCM features. The third stage is now used to separate type 1 from type 3 by using a car parts detector which consists of three separate binary classifiers, each detecting the presence of one car part. Each of the binary classifiers at this stage is a SVM trained with HOG features of the corresponding car part. The three car parts considered are wheel, headlight and hood. The method does not work in the cases where the car is damaged to an extant where it has none of the car parts considered, this serves as the limitation of the proposed method. It can be overcome in the future by increasing the number of car parts detected.

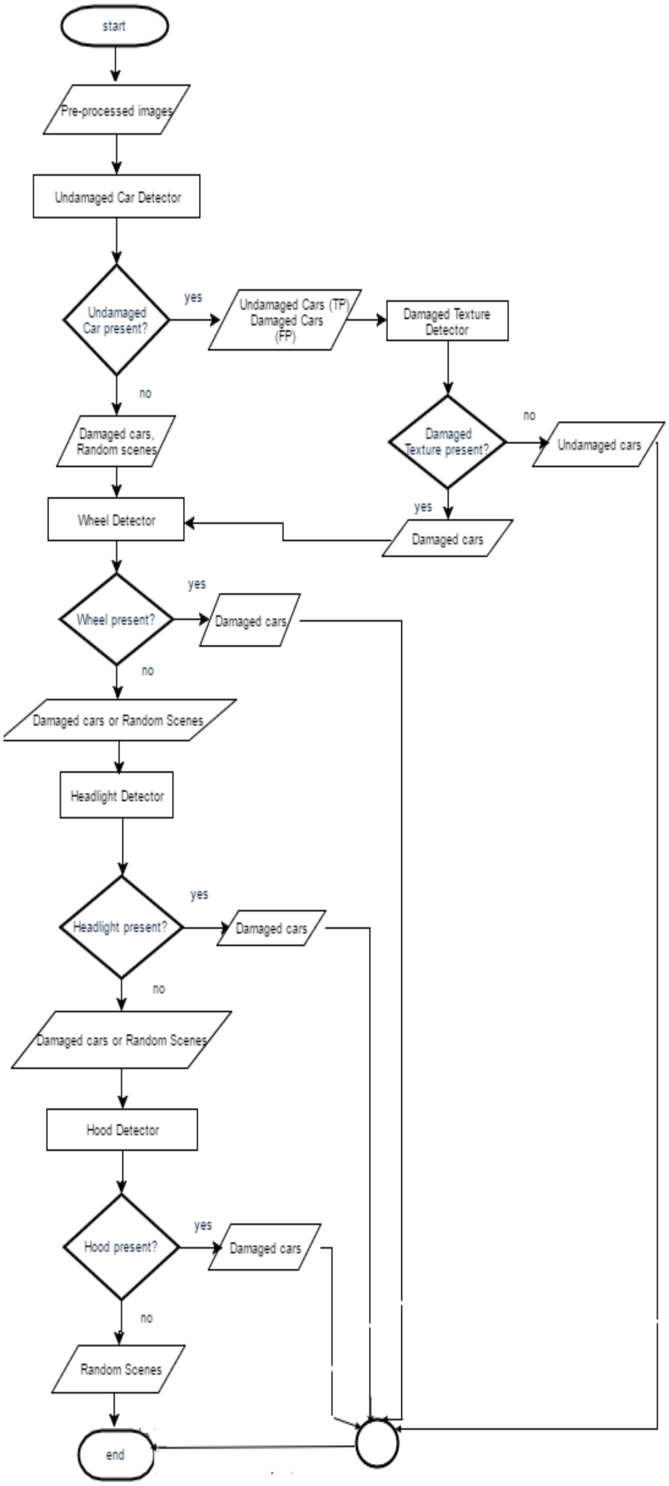


Fig. 1. Flow chart of the proposed method

A. Pre-processing

This is the first step that is carried in order to enhance the image and remove distortions like noise from the image. De-noising was achieved by using a Median filter [13]. The images are then resized to 256×256 for the next stage and converted to .JPEG format.

B. Undamaged Car Detector

Histogram of (HOG) features extracted from the training dataset is used to train a Support Vector Machine (SVM) which is a binary classifier based on supervised learning is employed in the first stage.

HOG is a feature descriptor introduced by Dalal et al.[14]. It is global feature and not a collection of many local features, that is, the object is described by one feature vector and not many feature vectors representing different parts of the object. The training images are resized to 256×256 and converted to gray scale. The HOG descriptor is then extracted from these 256×256 images, where 4×4, 8×8 or 16×16 pixels per cell are considered known as the cell size and from each cell the gradient vector at each pixel is calculated and put into a histogram with 8 bins. The histogram has 20 degrees in each bin and ranges from 0 to 180 degrees in total. Each gradient vector's magnitude is put into the histogram of bins and each value is split between the two closest bins. The reason behind the histogram is quantization and to reduce the number of values. Apart from this, it also generalizes the values in a cell.

The next step is normalizing the histograms to make them invariant to changes in illumination. When any vector is divided by its magnitude, it is said to be normalized. A gradient vector of a pixel in invariant to addition, subtractions and multiplications and hence the histograms can be normalized. This is taken one step further and instead of normalizing the histograms individually, the cells are grouped into blocks of 1× 1, 2×2 or 4×4 cells known as the block size and all the histograms in the block are normalized together. The reason behind this type of block normalization is that since changes in contrast occur in smaller regions of the image than compared to the entire image, normalizing in smaller blocks is more effective.

After extracting the HOG feature vectors, they are used to train an SVM model [15]. SVM is a binary linear classifier used for supervised learning. The SVM divides the training data into two classes by constructing a maxim margin hyperplane such that this plane or surface has the maximum distance from the closest points in each training set called support vectors.

In order to get the optimal parameters for the SVM used to classify undamaged cars, a Grid Search [16] is performed which takes all the combinations of the parameters and finds the optimal set using cross-validation [16]. The parameters that are involved in Grid Search include the type of kernel (linear or RBF), value of C and Gamma.

C. Damaged Texture Detector

This is the second stage which improves the performance of the first stage by reducing the number of false positives from the first stage since the undamaged car detector detects a few damaged cars as undamaged cars due to their similarity. This module is implemented by training a SVM with texture features. Two types of texture features, Local Binary Patterns (LBP) and Grey Level Co-occurrence Matrix (GLCM) were extracted and it was seen that GLCM had a better performance.

Local Binary Patterns (LBP) [17][18] is a texture descriptor in which the image is divided into cells with each cell containing a fixed number of pixels. Each pixel in a cell has its value compared to its 8 neighboring pixels (north, south, east, west, north-west, north-east, south-west and south-east). If the neighbor's value is greater, then a 0 is assigned else a 1. Each pixel thus generates an 8-digit value and all such values are used to compute a histogram. Each cell's histogram is then normalized and concatenated to form the feature vector.

Gray-level co-occurrence matrix (GLCM) is method of extracting texture features based on the spatial relationship between pixels. In 1973, the method was proposed by Haralick et al., [19]. In this method pairs of pixels are considered in specific spatial position and values and a matrix is constructed, from this matrix various statistical texture characteristics are extracted. The texture characteristics that were considered for this work were dissimilarity, energy, contrast, homogeneity, correlation and Angular Second Moment (ASM). These texture characteristics were combined together to form a feature vector.

D. Car Parts Detector

This is the final stage and consists of three SVM classifiers each trained to detect one car part. The three car parts considered are wheel, headlight and hood. The steps to train and test the SVM model with Histogram of Gradient Descent features are similar to the methods employed in undamaged car detector. The three classifiers are cascaded with each other to segregate images of type 1 (damaged cars) and type 3 (random scenes) where the detection of one of the parts leads to classification of the image as a damaged car (type 1). An image is passed to the next classifier in this stage only if it is classified as negative by the previous classifier. An image that passes through and classified as negative by all three classifiers are classified as type 3 images.

E. Working of the system

A sliding window scans the image at each of the three stages and it is these scanned parts from one stage that are sent to the next stage. If at least one scanned part from an image results in the presence of a damaged car after passing through all three stages then the image to which it belongs is classified as containing a "damaged car". However, if all the scanned parts of an image are classified as a negative case of a damaged car, then the image is classified as negative case of containing a "damaged car".

The input image is first pre-processed by being converted to grayscale and resized to 256×256. It is then passed through the undamaged car detector and a sliding window of size 128×128 slides across the image at various scales. The 128×128 scanned parts that are classified as positive by the undamaged car detector are sent to the damaged texture detector and the ones classified as negative to the car parts detector. The damaged texture detector uses a sliding window of size 50×50 across the input images it receives. If a 50×50 scanned part shows the presence of damaged texture the corresponding 128×128 image is sent to the car parts detector. Similarly each classifier in the car parts detector uses a sliding window of size 50×50 and detection of a car part results in classifying the corresponding 128×128 input image as damaged. Finally, the original sized image to which this 128×128 scanned part belongs, is said to contain a damaged car.

If a damaged car is detected after these three stages an alert is sent to the nearest emergency help centre.

IV. Experimental Results and Analysis

In this section, the experimental datasets used, the setting of various parameters, experimental results and analysis are discussed.

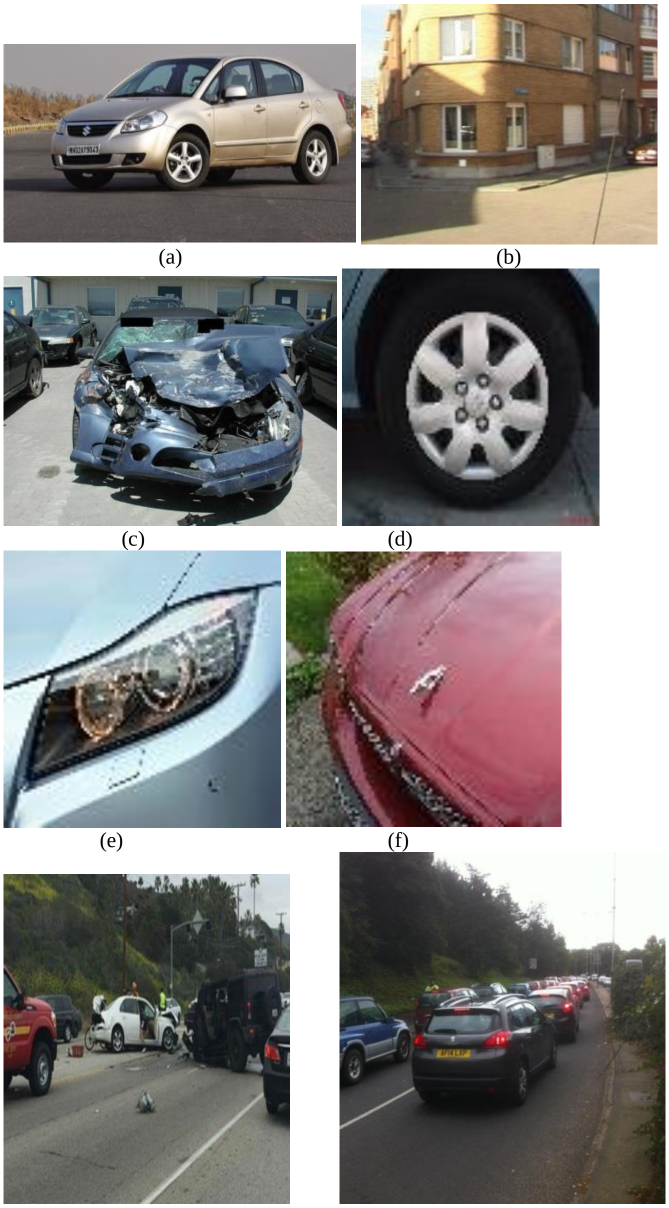
A. Experimental Datasets

The dataset used for each of the three stages is discussed in this section. Two different data sets have been used for testing of the overall system.

The undamaged car detector uses 250 images of undamaged cars from [20] as the positive training data set, an example is shown in Fig. 2(a). From this data set only the portion of the images containing cars has been extracted using an object marker utility to extract the region of interest. 250 images of random scenes, that is, an image without the presence of a car, was taken from [21][22] as the negative training data set, an example is shown in Fig. 2(b). It was noticed that the performance at this step in regard to classifying damaged cars as negative cars was not satisfactory and hence 125 images of random scenes along with 125 images containing damaged cars only was added to the negative training data set from [23], an example is shown in Fig. 2(c) and available at [27] This improved the detection accuracy and the ability of the system to classify damaged cars as negative cars. Images of damaged cars were taken from the CIREN database [23]. The positive and the negative training data was resized to 256×256 and converted to grey scale.

Damaged texture detector uses 250 images of damaged cars from [23] made available at [28], as positive training data and 250 images consisting of undamaged cars from [20] and random scenes from [21][22] (that have not been used in the previous stage) as negative training data. The positive images are cropped to contain only the damaged parts using an object marker utility.

The car parts detector has three different classifiers for three car parts -wheel, headlight, hood- and each classifier uses 250 images from [20] of that part as training data as shown in Fig. 2(d), 2(e) and 2(f) respectively. 250 images of undamaged cars from [20] and random scenes from [21][22] are used as negative training data (that have not been used in the previous stages).



(a) (b) (c) (d) (e) (f) (g) (h)

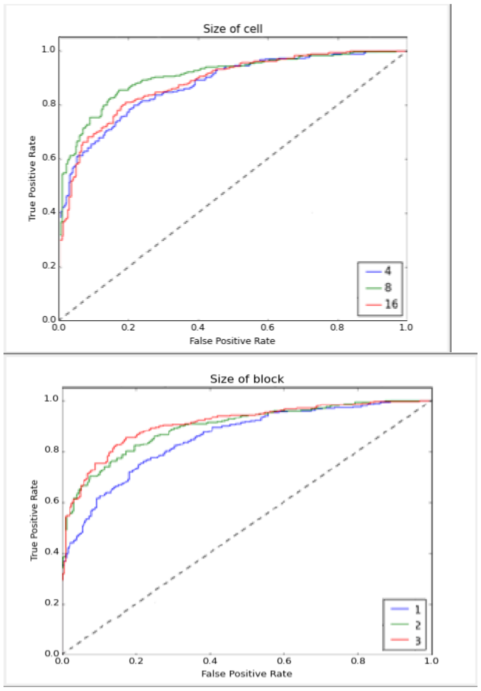
Fig. 2. Example images from the training and test datasets used. (a) and (h) are examples from the dataset used for undamaged cars, (b) from the dataset used for random empty scenes, (c) and (g) from DCD-1 and DCD-2, (d),(e) and (f) from datasets of the wheel, headlight and hood respectively

The test data for the overall system and each classifier was compiled and is called DCD-1 and DCD-2 as follows.

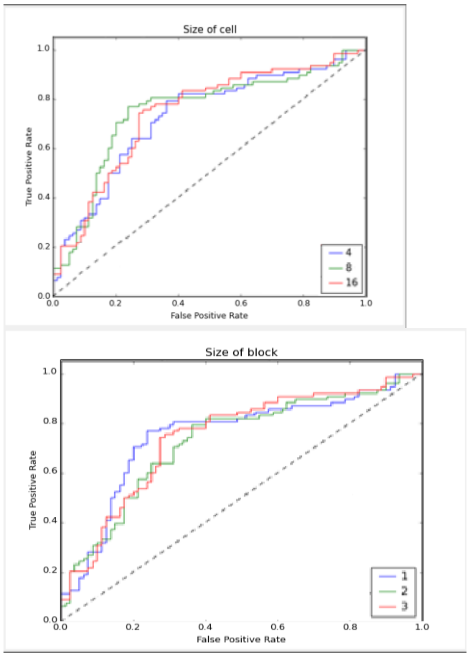
DCD-1

The first set- DCD-1- contains 300 images of damaged cars taken from [23] as the positive test data and 150 images of undamaged cars and 150 images of random scenes from [20][21][22] as negative test data (images that have not been used to train the model). The positive dataset contains images of individual damaged cars captured from different views (at approximately 2m from the damaged car) as shown in Fig. 2(c), available at [29] and the negative dataset contains images of individual undamaged cars captured from different views (at approximately 2m from the undamaged car) and images of empty scenes as shown in Fig. 2(a) and 2(b) respectively.

DCD-2- Since a part of the logic the system used is based on the difference between damaged and undamaged cars, images containing both types of cars were also considered, this forms the second set- DCD-2 - which contains images taken in real-time from surveillance cameras positioned on the sides of a road/highway with 80 images from [24] containing multiple damaged cars along with undamaged cars in a traffic scene as the positive test data set (at approximately 20m from the scene) as shown in Fig. 2(g) and available at [30]. 80 images from [24] containing multiple undamaged cars are used as the negative dataset (at approximately 20m from the scene) as shown in Fig. 2(h) and available at [31]. The images from DCD-2 were used for testing in an attempt to validate the working of the system in a realistic scenario since these images are of poorer quality, have higher changes in illumination and contrast and have more than one object present in the images.



(a)



(b)

Fig. 3. Results for various HOG parameter settings. (a) and (b) are the cell and block setting for DCD-1 and (3) and (4) are cell and block setting for DCD-2

B. Parameter setting

For the undamaged car detector, the optimal parameter for HOG feature extraction are evaluated for both datasets separately. The parameters are tested and optimized for HOG descriptor using ROC curves [25]. The parameters tested are size of the cell and size of the block where size of cell is the number of pixels contained in a cell and size of block is the number of cells contained in a block. The range of values considered for size of cell were 4×4, 8×8 or 16×16 pixels and the range considered for size of block were 1×1, 2×2, 3×3. The default parameters for HOG which are set while changing only one of the parameters are as follows:

- Cell size = 8,
- Block size=3
- number of bins = 9
- Step size= 10×10
- Minimum window size= 128

Fig. 3 shows that for DCD-1 when the different parameters for HOG were tested cell size 8 ×8 gave the best performance and for block size 3×3 gave the best performance and for DCD-2 cell size 8 ×8 and 16×16 and block size 1×1 and 3×3 gave the best performance. 8×8 was chosen as cell size and 3×3 as block size.

TABLE I. SVM PARAMETER RESULTS FOR DCD-1

Classifier	Kernel	C	Gamma
Undamaged Car	RBF	10	0.01
Damaged Texture	RBF	1	0.001
Wheel	RBF	10	10
Headlight	RBF	10	1
Hood	RBF	1	1

SVM: Grid search along with K-fold cross validation [16] is used for choosing the optimal parameters for SVM for each of the five classifiers used and for both Datasets. A Classification report is generated for the best parameter setting. The parameters that are tested for SVM are the type of kernel, C and Gamma values. Linear and RBF were the two type of kernels considered and for the value of C a range from 10^k for k € { -7,…..,7} and Gamma ranges from 10^k for k € { -6,…..,1}. The default parameters for SVM which are set while changing only one of the parameters are

TABLE II. SVM PARAMETER RESULTS FOR DCD-2

Classifier	KERNEL	C	Gamma
Undamaged Car	Linear	1	-
Damaged Texture	RBF	1000000	1e-06
Wheel	RBF	10	10
Headlight	RBF	10	1
Hood	RBF	1	1

- kernel = linear
- C= 1

After tuning the parameters of SVM for best f1 score for each of the classifiers, the best parameters found are given in Table I for DCD-1 and in Table II for DCD-2. For all 5 classifiers, hard negative mining [26] was employed in order to improve the performance.

C. Experimental Results And Analysis

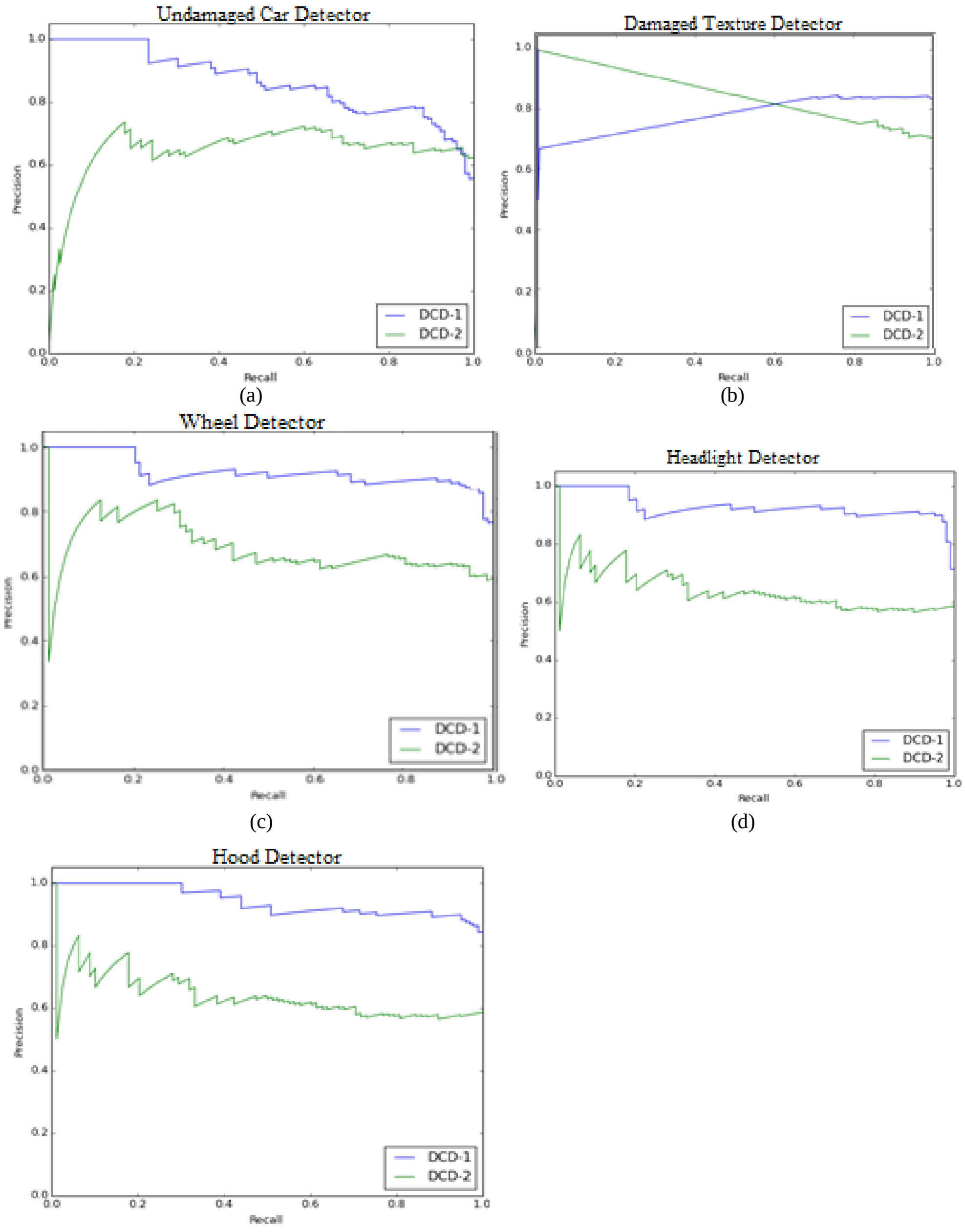
The precision, recall and accuracy for the five classifiers used and the overall system is given in Table III for DCD-1 and in Table IV for DCD-2. The five classifiers were tested on the images that are received from the preceding classifier. The precision-recall curves comparing the performance for the 5 classifiers when tested on both Datasets is given in Fig. 4. It can be seen that the performance of the five classifiers trained and tested for DCD-1 performs better than the five classifiers trained and tested for DCD-2. The reason for this is that DCD-1 consists of images captured at less than 2m from the objects and is of better quality whereas DCD-2 consists of images captured at less than 20m from the objects and is of poorer quality. For damaged texture detection in the second stage both LPB and GLCM were tried and since LBP had a precision of 40%, recall of 38.89% and accuracy of 37%, GLCM with a higher performance as seen in Table III and Table IV was used as the feature extracted in this stage for both Datasets. The overall system accuracy for DCD-1 was 81.83% which is greater than the 64.37% accuracy that was achieved for DCD-2.

TABLE III. PERFORMANCE RESULTS OF THE FIVE CLASSIFIERS AND THE OVERALL SYSTEM TESTED FOR DCD-1

Classifier	PRECISION	Recall	Accuracy
Undamaged Car	67.39	79.49	70.89
Damaged Texture	75.34	83.33	70.70
Wheel	64.03	74.21	71.56
Headlight	64.67	70.45	71.55
Hood	64.99	72.56	70.45
Overall System	60.36	82	64.37

TABLE IV. PERFORMANCE RESULTS OF THE FIVE CLASSIFIERS AND THE OVERALL SYSTEM TESTED FOR DCD-2

Classifier	PRECISION	Recall	Accuracy
Undamaged Car	83.67	83.67	83.67
Damaged Texture	88.03	99.60	88.29
Wheel	80.60	87.87	85.56
Headlight	80.51	87.88	82.6
Hood	79.2	86.2	81.5
Overall System	80	83.75	81.83



(a)

(b)

(c)

(d)

(e)

Fig. 4. Precision-recall curves obtained by testing each of the five classifiers' on datasets DCD-1 and DCD-2. (a), (b), (c), (d) and (e) show the precision-recall curves of the undamaged car detector, damaged texture detector, wheel detector, headlight detector and hood detector respectively

V. CONCLUSION AND FUTURE WORK

A new approach to accident detection by detecting damaged cars from footage captured from surveillance cameras has been presented in this paper. Detection of damaged cars using the techniques of Machine Vision was achieved successfully. The detection was done based on the fact that an undamaged car detector will not detect damaged cars and that most damaged cars have car parts in tact. The method developed used a total of five SVM classifiers trained with HOG and GLCM features.

Since damaged car detection has not been attempted before two datasets of damaged cars were compiled for the sake of the project and released for public use. The system implemented was tested on these two different datasets DCD-1 and DCD-2, which differ based on the distance of the camera from the damaged car, the quality of the images and the number of objects in the images. The accuracy of the system is 81.83% for DCD-1 and 64.37% for DCD-2. However, the system does not detect damaged cars that are damaged to an extant where none of the car parts being considered are present. This is the major limitation of the project.

The future work includes extending the working of the system to detect all types of damaged vehicles as it currently successfully detects only damaged cars. The working of the system can also be extended to detection in nighttime conditions.

REFERENCES

[1] X. Wen, L. Shao, W. Fang, and Y. Xue, “Efficient feature selection and classification for vehicle detection,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 3, pp. 508-517, Mar. 2015.

[2] Yichuan Tang. Deep learning using linear support vector machines. In *Workshop on Representation Learning, ICML*, 2013, Atlanta, USA, 2013.

[3] M. Egmont-Petersen, D. de Ridder, and H. Handels, “Image processing with neural networks- A review,” *Pattern recognition*, vol. *35, no.* 10, pp. 2279-2301, Oct. 2002.

[4] P.A. Viola and M.J. Jones, “Rapid object detection using a boosted cascade of simple features,” in *Proc. CVPR*, 2001, no. 1, pp. 511–518.

[5] "Global status report on road safety 2015", *World Health Organization*, 2016. [Online]. Available: http://www.who.int/violence_injury_prevention/road_safety_status/2015/en/. Accessed: 22- Mar- 2016

[6] V. Goud, "Vehicle accident automatic detection and remote alarm device," *International Journal of Reconfigurable and Embedded Systems (IJRES)*, vol. 1, no. 2, Jul. 2012.

[7] B. Prachi, D. Kasturi and C. Priyanka, “Intelligent accident-detection and ambulance-rescue system,” *PULSE*, vol. 450, no. 16, pp. 2, Jun. 2014.

[8] G. Liang, "Automatic traffic accident detection based on the Internet of Things and Support Vector Machine", *International Journal of Smart Home (IJSH)*, vol. 9, no. 4, pp. 97-106, Apr. 2015.

[9] C.Vidya Lakshmi, J.R.Balakrishnan, “Automatic Accident Detection via Embedded GSM message interface with Sensor Technology,” *International Journal of Scientific and Research Publications*, vol. 2, no. 4, Apr 2012.

[10] X. Tong, Z. Hong, S. Liu, X. Zhang, H. Xie, Z. Li, S. Yang, W. Wang and F. Bao, “Building-damage detection using pre-and post-seismic high-resolution satellite stereo imagery: a case study of the May 2008 Wenchuan earthquake,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 68, pp.13-27, Mar. 2012.

[11] H. Ma, N. Lu, L. Ge, Q. Li, Z. You, X. Li, “Automatic road damage detection using high-resolution satellite images and road maps,” in *Proceedings of the 2013 IEEE International conference on Geoscience and Remote Sensing Symposium,* Melbourne, Australia, Jul. 2013.

[12] S. Sivaraman and M.M Trivedi, “A review of recent developments in vision-based vehicle detection,” presented at the Intelligent Vehicles Symposium, 2013

[13] M. C. Motwani, M. C. Gadiya, R. C. Motwani, and F. C. Harris, “Survey of image denoising techniques,” in *Proceedings of Global Signal Processing Expo and Conference (GSPx ’04),* Santa Clara, Calif, USA, Sept. 2004.

[14] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2005, vol. 1, pp. 886-893.

[15] C. Cortes and V. Vapnik, “Support-vector network,” *Machine Learning*, vol. 20, no.3, pp. 273–297, Sept. 1995.

[16] C. Hsu, C. C. Chang, and C. J. Lin, “A practical guide to support vector classification,” Dep. Comp. Sci., National Taiwan Univ., Taiwan, Tech. Rep., 2005.

[17] T. Ahonen, A. Hadid, and M.Pietik¨ainen, “Face Description with Local Binary Patterns: Application to Face Recognition,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037-2041, 2006.

[18] T. M¨aenp¨a¨a. The Local Binary Pattern Approach to Texture Analysis — Extensions and Applications. PhD thesis, University of Oulu, 2003.

[19] R. M. Haralick, K. Shanmugam, and I. Dinstein, “Texture features for image classification,” in *IEEE Trans. Systems Man Cybemet*, 1973, pp. 610-621.

[20] “ImageNet,” [Online]. Available: http://imagenet.stanford.edu. Accessed: 20- Jun- 2016.

[21] “Caltech101,” [Online]. Available: http://www.vision.caltech.edu/Image_Datasets/Caltech101/. Accessed: 20- Jun- 2016.

[22] R. Fisher, “CVonline: The Evolving, Distributed, Non-Proprietary, On-Line Compendium of Computer Vision,” [Online]. Available: http://homepages.inf.ed.ac.uk/rbf/CVonline/. Accessed: 20- Jun- 2016.

[23] “CIREN The Nation's Largest Learning Laboratory,” [Online]. Available: http://www-nass.nhtsa.dot.gov/nass/ciren/SearchForm.aspx. Accessed: 20- Jun- 2016.

[24] “Google-Images,” [Online]. Available: https://images.google.com/. Accessed: 18- Apr- 2016.

[25] F. Suard, A. Rakotomamonjy, A. Bensrhair and A. Broggi, “Pedestrian detection using infrared images and histograms of oriented gradients,” presented at the IEEE Intelligent Vehicles Symposium, 2006.

[26] T. Malisiewicz, A. Gupta, and A. Efros, “Ensemble of Exemplar-SVMs for Object Detection and Beyond,” in *International Conference on Computer Vision*, 2011, pp. 89-96.

[27] "vaishnavi29/DCD", *GitHub*, 2016. [Online]. Available: https://github.com/vaishnavi29/DCD/tree/master/DCD-1/01. Accessed: 10- Jul- 2016.

[28] "vaishnavi29/DCD", *GitHub*, 2016. [Online]. Available: https://github.com/vaishnavi29/DCD/tree/master/DCD-1/02. Accessed: 10- Jul- 2016.

[29] "vaishnavi29/DCD", *GitHub*, 2016. [Online]. Available: https://github.com/vaishnavi29/DCD/tree/master/DCD-1/03. Accessed: 10- Jul- 2016.

[30] "vaishnavi29/DCD", *GitHub*, 2016. [Online]. Available: https://github.com/vaishnavi29/DCD/tree/master/DCD-2/01. Accessed: 10- Jul- 2016.

[31] "vaishnavi29/DCD", *GitHub*, 2016. [Online]. Available: https://github.com/vaishnavi29/DCD/tree/master/DCD-2/02. Accessed: 10- Jul- 2016.