International Journal of Advanced Computer Research (ISSN (Print): 2249-7277 ISSN (Online): 2277-7970)
Volume-4 Number-4 Issue-17 December-2014

Vehicle Classification by Lane Allowance

Vishakha Gaikwad[1*] and B.A.Sonkamble[2]

Received: 11-June-2014; Revised: 13-November-2014; Accepted: 10-December-2014

Abstract

Classification of vehicles from video is used for analysis of traffic, self-driving systems or security systems. This analysis is based on shape, size, velocity and track of vehicles. These features characterize vehicle in background subtraction and feature extraction methods. Extraction is done by active contours and morphological operations. Extracted vehicles are classified by applying various classification techniques. The combination of features and classification techniques varies with the application. Proposed system, Uses combination of K Nearest Neighbor (KNN) and Decision Tree techniques to overcome constraints. These constraints are instances of an object, overlapping of objects, and scaling factor. KNN is utilized to classify vehicle by size and lane. Decision tree manipulates the combination of these two features to classify accurately which results increased performance. This system classifies objects into three classes. These classes are four wheeler, bikers and heavy duty vehicle extracted from video.

Keywords

Classification, KNN, Decision tree, Active Contours and Pattern recognition.

1. Introduction

Classification of moving objects in video is widely used in many applications. There are different types of classification techniques used differently in different applications. Classification requires pre-processing of data which includes background subtraction and feature extraction.

Extraction techniques are required for more accurate classification described by Fabrizia et al [1]. It is a method for vehicle counting by category using background subtraction, PCA and Adaptive-KNN in three stages: normalization, training and classification. The training set is generated by number of edge points and principal components of image sequenced blocks which passed to Adaptive-KNN which calculates adaptive distance of the neighbors in sphere for classification.

Instead of PCA, a combination of simple principle and self-organization used by Jie et al[2].The self-organizing feature map (SOM) is an unsupervised learning algorithm combined with K-means to detect moving objects in traffic video. It constructs a system to obtain initial background when using the subtraction method to do motion detection. A tracking method is based on bidirectional comparison of centroid to track moving objects. Alternative of principle component, Active contours is suggested in proposed system which results exact edge etection of vehicle by comparing a deformable model to an image by energy minimization.

Apart from KNN, Vehicle detection and vehicle classification using neural network (NN), is achieved by Daigavane et al [3]. Width and length of the blob is calculated to result area of the vehicle which is passed to NN. Human and vehicle classification is manipulated by Longbin Chane [4]. Object classified by considering motion vector for the speed comparison and shape. Histogram of gradient models self-variance of objects in vector which is independent of each object location.

Previous methods considered size and velocity for classification. Consideration of lane of vehicle is also a principle factor to classify. On highways lanes are defined to each class of vehicle. For example, Heavy vehicle should be in third or fourth lane which is found sometimes in first or second lane and presence of two wheelers on highway. These kinds of rules are neglected by drivers which causes an accident. Proposed system manipulates size, velocity and lane

*Author for correspondence

Vishakha Gaikwad, Department of Computer, Pune Institute of Computer Technology, University of Pune, Pune, India.
Balwant Sonkamble, Department of Computer, Pune Institute of Computer Technology, University of Pune, Pune, India.

of vehicle to classify and examine. Region of interest counts instances of an object in each frame which counts vehicle many times. But virtual lines optimize the count of vehicle by considering vehicle online only. Line on road neglects the other moving objects, for example, trees or shadows. A novel detection and classification method are proposed by Niluthpol et al [5] using multiple time-spatial images (TSIs), each obtained from a virtual detection line on the frames of a video. Multiple TSIs produces time image to detect overlapping of the vehicles and to manipulate difference between the still and moving objects to increase the accuracy of detection.

Each instance is passed to classifier to classify object which decreases performance by calculating class each time. Effectual way of classifying image globally and instances locally is presented by Xin zao et al[6]. Organization of data requires accurate decision support for vehicle analysis and classification. Decision tree constructs possible consequences to find out cost of solution set which gives maximum outcome. Hang yang [7] gives optimized solution to reduce time complexity of deciding levels of decision tree by an adaptive tie evaluation and extra pruning conditions. Therefore for exact decision of vehicle class, Decision tree is used in proposed system.

This paper is organized as follows. Section 2 phases of the system. Section 3 states mathematical model and algorithm. Results are shown in Section 4. Conclusion is in Section 5, followed by References.

2. Vehicle Classification

Fig.1. shows phases of the system. Input video is accessed from the buffer of camera which decides the size of file. Assumption of this system is video file is taken from buffer with 1 minute video file.
Each frame is extracted from video for analysis of detection of key frame analyzed by Guozhu Liu et al [8]. Let *pi* is any frame and *pi+k* is next frame, *l* is length of vehicle and *m* is maximum speed of vehicle. Difference between the two consecutive frames gives vehicle counted and its speed. Selection of *k* and time required to pass the vehicle itself with *m* is calculated by following equation,

$$tpi+k-pi < l/m \qquad (1)$$

Let, *t* is the time required to pass vehicle itself. It is greater than time selected between frames showed in eq. (1). Motion estimation techniques for object extraction include the inter frame difference method. Objects are detected from a video by frames and background difference [9]. Features are identified for detected object. Preprocessing of input video has stages as shown in fig.2. Then classification of moving objects is done after an object is extracted from a frame. Classification of extracted objects is done by characteristics which defines a class. Attributes which categorizes object in to classes are size, shape, velocity, and lane. With these attributes views (top, side, front etc.) and rotation of an object are also important. To store attributes of vehicle data training is required called as trained data. From samples of trained data average is calculated to find class of an object. Global training considers all objects in one frame and local training is concerned with instances of an object. Extracted object classified by K-Nearest Neighbor (KNN) and Decision Tree (DT) by comparing with trained data.

2.1 Preprocessing

Selection of segmentation, edge detection, morphological operations and feature extraction algorithms is also important to increase performance. Fig.2 shows preprocessing steps.

Step1.Segmentation of video is done by the histogram difference and motion compensation ratio analysis to get key frames from video [8]. Analysis of camera width, length of screen, maximum speed and length of vehicle is required for accurate analysis. Selected frame is passed to canny edge detector to find out edges. Let, (x, y) be pixel variable, *thresh* is threshold, *b* is background frame, *p* is current frame and *subtract* be the segmented output in eq.(2),

$$Subtract=p(x,y,t,thresh)-b(thresh)$$

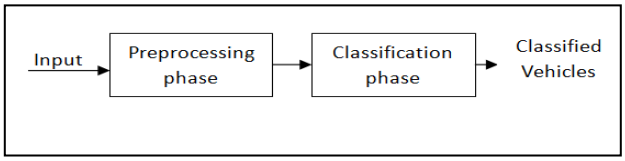

Fig.1. Overview of phases of the system.

Step2: Canny edge: Edge detection is done by convolving raw image with Gaussian filter and intensity gradients. The sobel operator results first derivative of raw image gives intensity difference accurately. Fig.3. (a) shows edges detected by canny edge detection in frame. This frame is given to closing operation to create blob of vehicle. Let, c be the

output frame of canny function with thresh1 and thresh2 to take required data in eq.(3),

$$c=canny(s, thresh1, thresh2) \quad (3)$$

***Step3*:** Closing Operation: Morphological operations are a set of operations that process images which apply a structuring element to an input image and generate an output image. Dilation is result of image convolved with kernel. Where erosion is output of minimal pixel value set overlapped by kernel [5]. Closing operation is difference of dilation and erosion which forms the blob by selecting kernel coefficient values. Fig.3. (b) shows car which is detected by these operations. Let, *clos* be the result of closing operation in eq.(4)

$$clos=erosion(dilation(c)) \quad (4)$$

***Step4*:** Contour detection: Output came from closing operation is processed by internal and external energy forces to find out shape. Detection is operated on virtual lines only. Inter frame difference gives the speed of vehicle here. Fig.3 (c) and (d) gives idea of contour look. Features of these contour blobs are identified and stored in sample set. Let, *E* is the energy function and vector *V* be the collection of contours found in one frame. (x,y) is position of contour represented by *s* and **ptr* is pointer to next contour.

$$E^* = \int E\big(V(s,* ptr)\big)\, ds \quad (5)$$

2.2 Classification

In this stage feature vector manipulated with trained set to identify class.
KNN and Decision tree is used to classify vehicles in to the class.

2.2.1 Feature extraction

Features which are invariant with the environment and less sensitive are selected for classification. Size invariant, size based and texture based geometric features is considered to construct the feature vector. Each object O features represented as *O {width, area, b, ratio, position}.*

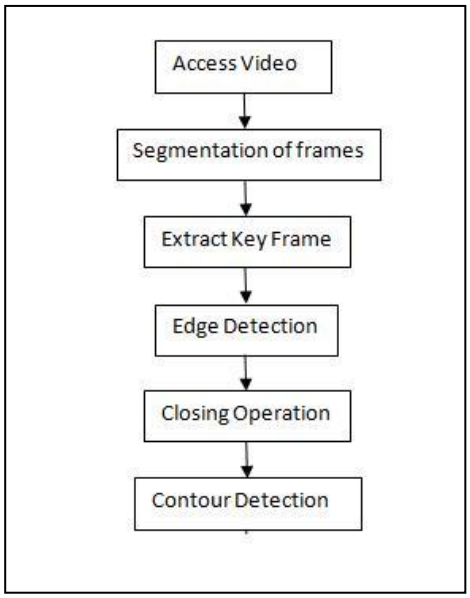

Fig.2. Preprocessing steps of the system.

1) Width: Rectangle is drawn around the detected vehicle. Width of rectangle is taken by rectangle around the vehicle.
2) Length: Same as to width length of vehicle is also calculated.
3) Area: Area of contours is found from the vector of linked contours. Area of vehicle is very important feature to identify the class of vehicle. Classes are done on the basis of small, medium and large.
4) Rectangular boundary (b): Rectangular boundary is bounded around the vehicle when it touches the virtual line drawn on the road. That time position of vertical axis is also identified to decide the lane of vehicle.
5) Ratio of width and length: Ratio of width and length creates importance in presence of occlusion of vehicles.
6) Position: This attribute is measured to find lane of the vehicle.
7) Lane: From position of vehicle, lane of the vehicle is decided by setting range on vertical axis as shown in table1. Three lanes are shown in fig.3 (c).

Table 2 shows lane number by position of vehicle. Let, W be presence of vehicle in wrong lane. Bike is not allowed in any lane. Car or class1 can be in any lane and heavy vehicle should be in lane 2 or lane 3. Presence of heavy vehicle in lane 1 can be observed by controller easily by the system.

2.2.2 K-nearest neighbor (KNN)

Compared with other machine learning methods, KNN is one of the simplest classification methods. It is based on an assumption that samples belong to the same class if they are close in the instance space. KNN classifies the extracted objects globally. This classifier requires trained data of known samples.

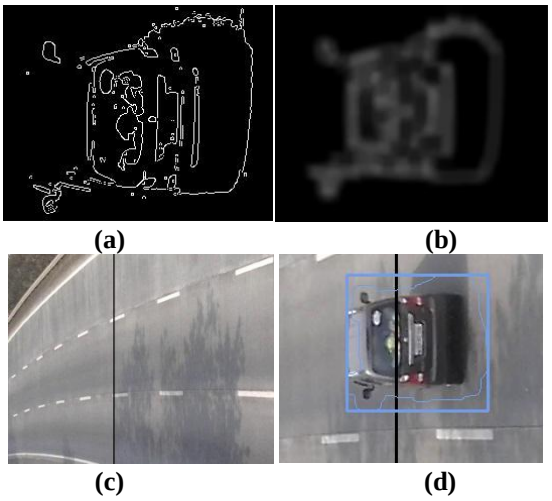

Fig.3. Output of different operations (a) Canny edge detection. (b) Closing operation. (c)Virtual line on road and three lanes on highway.(d) Vehicle detected on virtual line by active contour.

Table 1. Lane number of vehicle and average range of position.

Class of vehicle	Lane1	Lane 2	Lane 3
Two Wheeler	W	W	W
Car	A	A	A
Heavy Vehicles	W	A	A

Samples: Known data with known features and known classes stored in big numbers to identify class accurately. It is represented as *I {O, C}*

Trained data: Known data of known features. Feature and its values are stored in this vector. It is represented as *V {O}*.

Trained class: Known data of known classes. It is 1D vector of class of sample as sample number is equal to index of vector. Where, trained class is represented as *C {class}*.

Unknown data: Input of unknown objects is provided to the classifier to detect the class of that object. Let, U be the class of unknown object. *U{O}*

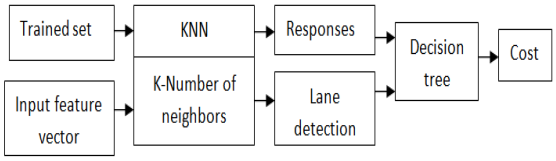

Fig.4. Classification and decision tree stages.

Let, *classifier* is the function of KNN. *K* be the number of neighbors and *r* b the response of the classifier.

$$response = classifier\ (O,V,C,U) \qquad (6)$$

2.2.3. Decision tree:

Decision Tree Classifier works on possible set of questions which leads optimized solution set [8]. The set of question is utilized with trained set to derive the class of vehicle. Let, *tree* is the function of tree structure in eq. (7). Let, *lane* is Lane calculated from the *position* of the vehicle which gives output DT {*A, Class*}. Where, *A= {allowed, not allowed}* shows that vehicle is allowed or not on lane of highway and *Class= {class1, class2, class3}*.

$$DT = tree\ (response, lane) \qquad (7)$$

Fig.5 shows the decision tree of the class and lane manipulation. Entropy coding method is used here to calculate cost.

3. Mathematical Model

Let S be the System,
S = *{Start, End, Input, Output, DD, NDD, ϕ}*

Where, *Start* is accessing of video. *End* of the system is classified object. V= *{b, r, s, F}* Where, V is video with bit rate b, frame rate r and s be the size of frame in pixel dimension format. P is set of frames in video $p \in P$ as explained above. Function of set of pixels in one frame is *p(x, y)*. Output is DT which gives class and lane of vehicle. It derives allowance of vehicle on particular lane.
Constraints (Φ):

Intersection of moving objects, Scaling factor should be considered for accurate shape detection, Speed Comparison of two objects can be destructed in some cases. Instances of an object can be seen in many frames of video. Detection of instances becomes challenge as object can look alike. Light intensity change detection is also a challenge.
DD =Deterministic data is Object should have less than 10000 area. Frame rate should be 26 f/s.
NDD =Nondeterministic data are light intensity, rain, fog.

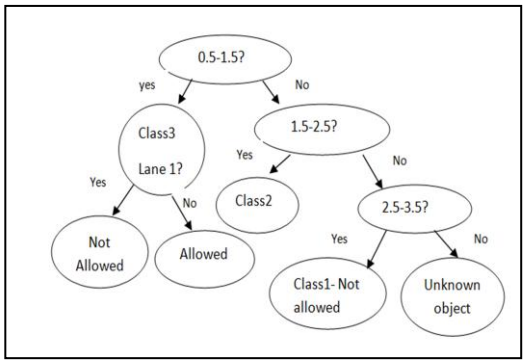

Fig. 5. Decision tree analysis for vehicle class and lane manipulation.

Morphism Analysis: $P \rightarrow O$, if O is moving object
And $O \rightarrow C$. The set range $R(O, C) = \{c \mid \langle o, c \rangle$ in R for some $o\}$ is called the range of the relation R.

4. Experimental setup and Results

The System is designed in QT 4.7 using Opencv 2.4.0 libraries. Opencv is well known library set for image processing. QT is cross platform framework and UI development framework. C++ object oriented language is experimented for system setup. MPEG 4 video file format of resolution 1280*720.

Training and classification is carried out by the system and shows results in Table 2 and Table 3. Distance in KNN classifier calculated by hamming distance of brute force matcher. Trained data is generated by feature vector of objects detected in frame. Average is taken to calculate response of the classifier. If any sample gives major difference gets replaced by another sample. 100 samples are taken to enumerate class accurately. Table 2 Shows result of probability of class and number of nearest K. As number of K increases confusion increases and error rate increases. The area and position of vehicle is calculated on vertical axis to find out lane and size of vehicle as shown in table 2. From the analysis of area of vehicles range of area for each class is identified.

Table 2. Error variation in response by number of neighbors.

Class of vehicle	K=10	K=30	K=50
Two Wheeler	0.82	0.85	0.84
Car	0.87	0.89	0.86
Heavy vehicle	0.92	0.93	0.91

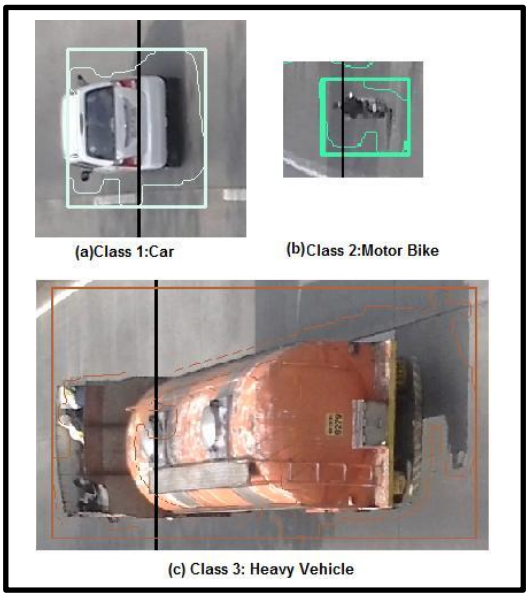

Fig.6. Outcome of System shows (a) class1 is of car (b) class2 is of bikes and (c) is heavy vehicle class3.

Result in fig. 6 shows vehicles are classified in three different classes. Use of vertical line optimizes solution as limited part is concerned. When two vehicles overlap, confusion is created in the result which can be avoided by use of decision tree. Class, position and lane manipulation question set of various combinations is generated. It checks for size as well as ratio of width to length of vehicle.

Error rate increases in presence of shadow of trees or heavy vehicles. Because of this reason error rate of class 1 and class 2 is increased. Second reason is number of class 1 on highways is maximum as compare to heavy class 2 and class 3 and has maximum speed which increases possibility of overlapping.

Table3. shows result of average error rate is calculated for each class. These errors are calculated by analysis of input combination of size and position of vehicle.

Table 3. Area and average error rate of classes.

Class of vehicle	Area of Vehicle(mm)	Average Error Rate	Average error rate (Decision tree)
Two wheeler	1500-3000	9.27%	6.54%
Car	4000-7000	7.54%	5.32%
Heavy	8000-10000	5.30%	3.43%

5. Conclusion and Future Work

Classes and lanes of vehicles identified accurately by the system. Result shows increased performance by the decision tree combined with KNN. Earlier detection of position of classes avoids accidents and useful in secured transportation system. Presence of illegal vehicles can be avoided.

Analysis of lane and speed comparison by using optical flow will result avoidance of maximum speed on highway in future. Maximum speed cannot be analyzed by simple inter frame difference methods.
Relative motion analysis is required for prevention of accidents. Three lanes and vehicle speed manipulation is possible by relative motion analysis. Time required to reach destination is related to relative motion. In future speed comparison and relative motion will create new level of self-driving system.

References

[1] Fabrzia Medeiros de Sousa Matos," An Image Vehicle Classification Method Based on Edge and PCA Applied to Blocks", 2012 IEEE International Conference on Systems, Man, and Cybernetics October 14-17, 2012, COEX, Seoul, Korea,pp. 1-7.

[2] Jie Xia, Jian Wu, Yan-yan Cao, Zhi-ming Cui, "SOM based Classification Method for Moving Object in Traffic Video", 2010 IEEE Third International Symposium on Intelligent Information Technology and Security Informatics,pp.1-5.

[3] P.M. Daigavane, Preeti R. Bajaj and M.B. Daigavane, "Vehicle Detection and Neural Network Application for Vehicle Classification" ,2011 IEEE International Conference on Computational Intelligence and Communication Systems, pp. 1-5.

[4] Longbin Chen, Rogerio Feris, Yun Zhai," An Integrated System for Moving Object Classification in Surveillance Videos", IEEE Fifth International Conference on Advanced Video and Signal Based Surveillance 2008, pp. 2-6.

[5] Niluthpol Chowdhury Mithun, Nafi Ur Rashid," Detection and Classification of Vehicles from Video Using Multiple Time-Spatial Images", IEEE Transaction on intelligent transportation system, VOL. 13, NO. 3, September 2012, pp. 1-9.

[6] Xin Zhao, "Global and Local Training for Moving Object Classification in Surveillance-Oriented Scene", IEEE International Conference on image processing November 2011, pp. 1-4.

[7] Hang Yang, Simon Fong," Optimized Very Fast Decision Tree with Balanced Classification Accuracy and Compact Tree Size, IEEE International conference, Sept, 2012, pp. 1-7.

[8] Guozhu Liu, and Junming Zhao, Key Frame Extraction from MPEG Video Stream, IEEE Proceedings of the Second Symposium International Computer Science and Computational Technology(ISCSCT 09) Huangshan, P. R. China, Dec. 2009, pp.7-11.

[9] Qian Yu and Gerard,"Motion Pattern Interpretation and Detection for Tracking Moving Vehicles in Airborne Video"', In Computer Vision and Pattern Recognition, 2009. CVPR 2009,pp. 2671-2678. IEEE, 2009.

Vishakha Gaikwad received B.E. degree in Computer Science and Engineering in 2009 from Cummins College of Engineering for Women, Pune .She is pursuing M.E. degree in Computer Science and Engineering from Pune Institute of Computer Technology, Pune under University of Pune. Her research interest includes image processing, KNN and pattern recognition. Email: Vishkha.gaikwad@gmail.com.

Balwant Sonkamble received B.E. degree in Computer Science and Engineering in 1994 and M.E. in 2004. He is pursued Ph. D. in Computer Science and Engineering from Shri Ramanand Teerth Marathwada University, Nanded, M.S. India. He is distinguished Assistant Professor in Computer Engineering Department at Pune Institute of Computer Technology, Pune, Mahrashtra state, India. His research interest includes speech processing, Support vector machines, computer vision, image processing, neural networks and pattern recognition.