Research Journal of Engineering Sciences ISSN 2278 – 9472
Vol. **2(12),** 9-13, December **(2013)** *Res. J. Engineering Sci.*

A Synergistic Approach of Combining both Lane and Vehicle Detection, Tracking and Localization

K.S. Deepa
Department of Electronics and Instrumentation Engineering, Karunya University, Embedded Systems, Coimbatore, Tamil Nadu, INDIA

Available online at: www.isca.in, www.isca.me
Received 6th December 2013, revised 16th December 2013, accepted 26th December 2013

Abstract

In this paper, a synergistic approach to integrated lane and vehicle tracking for driver assistance is introduced and also the Improved on the performance of both lane tracking and vehicle Tracking modules. Further, the presented approach introduces an approach to localizing and tracking other vehicles on the road with respect to lane position, which provides an information contextual relevance that neither the lane tracker nor vehicle tracker can provide by itself

Keyword: Synergistic, lane, vehicle, detection, tracking, localization.

Introduction

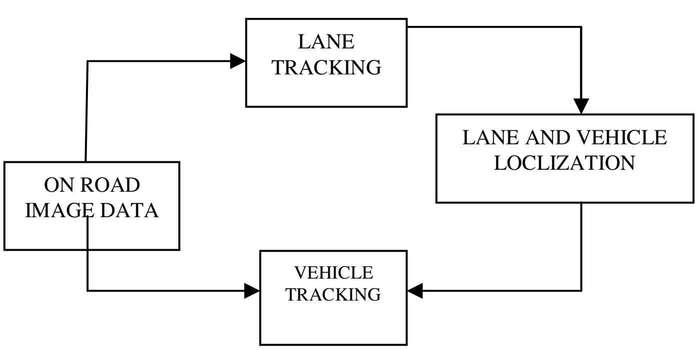
In this paper, a synergistic approach to integrated lane and vehicle tracking for driver assistance is introduced. Vehicle tracking performance has been improved by utilizing the lane tracking system to enforce geometric constraints based on the road model. This paper focus on monitoring the exterior of the vehicle. Monitoring the exterior can consist of estimating lanes[1], pedestrians, vehicles[2], or traffic signs and also full integration to benefit both vehicle tracking and lane tracking,

Basic Problem of the Statement

Each year, some 1.2 million people die worldwide as a result of traffic accidents. To avoid accidents focus on monitoring the vehicle exterior address one particular on-road concern. This paper adds the valuable safety functionality and provides a contextually relevant representation of the on road environment for driver assistance

Overview of the Project

Using the MATLAB code, mapping the image on road data with lane and tracking the vehicle.



Edge Detection

Edge detection[1] is the process of localizing pixel intensity transitions. The edge detection has been used by object 3 target tracking[8], segmentation, and etc. Therefore, the edge detection is one of the most important parts of image processing.

There mainly exist several edge detection[4] methods (Sobel, Prewitt, Roberts, and Canny). These methods have been proposed for detecting transitions in images. Early methods determined the best gradient operator to detect sharp intensity variations[5]. Commonly used method for detecting edges is to apply derivative operators on images. Derivative based approaches can be categorized into two groups, namely first and second order derivative methods. First order derivative based techniques depend on computing the gradient several directions and combining the result of each gradient. The value of the gradient magnitude and orientation is estimated using two differentiation masks[6].

In this work, Sobel which is an edge detection method is considered. Because of the simplicity and common uses, this method is preferred by the others methods in this work. The Sobel edge detector uses two masks, one vertical and one horizontal. These masks are generally used 3×3 matrices. Especially, the matrices which have 3×3 dimensions are used in MATLAB. The masks of the Sobel edge detection are extended to 5×5 dimensions[7], are constructed in this work. A MATLAB function, called as Sobel5×5 is developed by using these new matrices. Standard Sobel operators, for a 3×3 neighborhood, each simple central gradient estimate is vector sum of a pair of orthogonal vectors[8]. Each orthogonal vector is a directional derivative estimate multiplied by a unit vector specifying the derivative's direction. The vector sum of these simple gradient estimates amounts to a vector sum of the 8 directional derivative vectors.

a	b	c
d	e	f
g	h	i

The directional derivative estimate vector G was defined such as density difference /distance to neighbor. This vector is determined such that the direction of G will be given by the unit vector to the approximate neighbor.

Morphological Algorithm

Morph means shape. Morphological processing can solve an image processing problem, view the image processing toolbox
Dilation and Erosion: Morphology is a broad set of image processing operations. The process images based on shapes[5]. Morphological operations apply a structuring element to an input image[9], creating an output image on the same size. In a morphological operation, the value of each pixel in the output image is based on a comparison of the corresponding pixel in the input image with its neighbors. By choosing the size and shape of the neighborhood, can construct a morphological operation that is sensitive to specific shapes in the input image. The most basic morphological operations[10] are dilation and erosion. Dilation adds pixels to the boundaries of objects in an image, while erosion removes pixels on object boundaries. The number of pixels added or removed from the objects in an image. Depends on the size and shape of the structuring element used to process the image. In the morphological dilation and erosion operations, the state of any given pixel in the output image is determined by applying a rule to the corresponding pixel and its neighbors in the input image.

Structuring Elements: An essential part of the dilation and erosion[11] operations is the structuring element used to probe the input image. A structuring element[12] is a matrix consisting of only 0's and 1's that can have any arbitrary shape and size. The pixels with values of 1 define the neighborhood. Two dimensional or flat, structuring elements are typically much smaller than the image being processed. The center pixel of the structuring element, called the origin, identifies the pixel of interest-the pixel being processed. The pixel in the structuring element containing 1's defining the neighborhood of the structuring element. The pixels are also considered in dilation or erosion processing. Three dimensional or nonflat, structuring elements use 0's and 1's to define the extent of the structuring element in the x-and y-planes and height values to define the third dimension.

Morphological Operations for Holes Filling: The formulation of the two morphology filling operation can be formulated as follow:
Hole-Pixel Initial Algorithm (HPIA)
Consider O = {o0, o1... ok | oi represents the (x, y) pixels of object i, k represents the number of objects in image}, then the initial for this algorithm is a vector P where in (1) and (2)

$$P = \{I_i \mid i=1, 2, \ldots k\} \quad (1)$$

$$I_i = \{ x_j, y_j), j \in \text{number of holes in object } i \} \quad (2)$$

This required having starting points that belongs for each hole in each object of the image, which is very difficult since the real time application for such algorithm cannot go along with this since it requires the human interaction in a very essential step which is all the latter processing depends on. However, the equation for this algorithm is in (3) taken from [1] which closes the holes of the object after certain number of epochs which means until there is no change in Xk:

$$X_k = (X_{k-1} \oplus B) \cap A^c, \text{ for } k = 1, 2, 3 \ldots \quad (3)$$

Where X0 is the marker with initial points Ii, B is the 3x3 SE with zeros at the corners and ones elsewhere, and A is the original input binary image.

Optical Flow

Optic flow is the pattern[6] of apparent motion of objects, surfaces, and edges in a visual scene caused by the relative motion between an observer (an eye or a camera) and the scene. The optical flow system[8] object estimates object velocities from one image or video frame to another. It uses either the horn-schunck or the locas- kanade method.

Optical Flow Estimation for Video: Optical flow[9] is the distribution of the apparent velocities of objects in an image. By estimating optical flow between video frames, can measure the velocities of objects in the video. In general, moving objects that are closer to the camera will display more apparent motion than distant objects that are moving at the same speed.

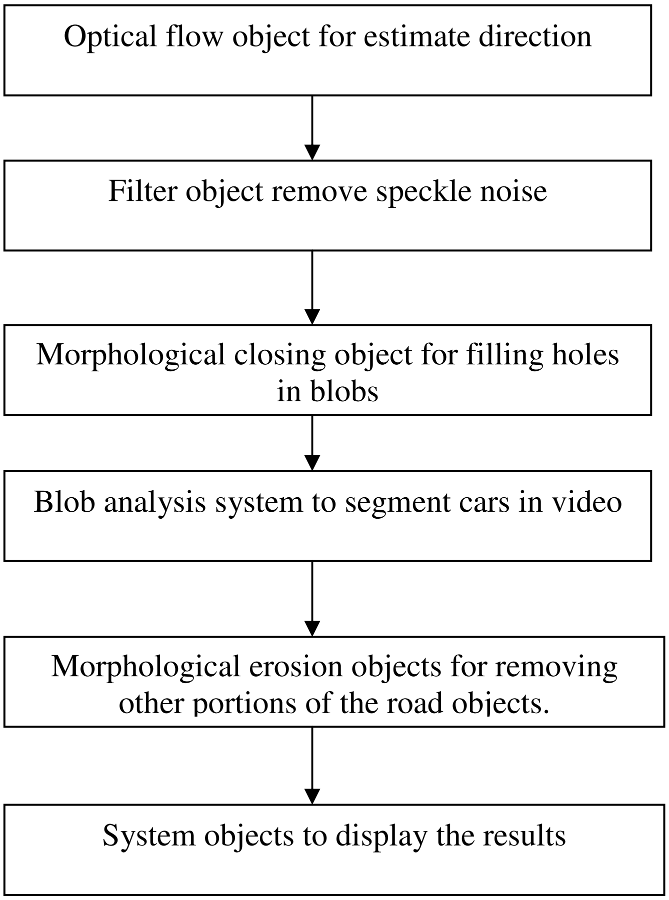
Optical flow estimation[14] is used in computer vision to characterize and quantify the motion of objects in a video stream, often for motion-based object detection and tracking systems.

The model uses an optical flow estimation technique to estimate the motion vectors in each frame of the video sequence .By thresholding[15] and performing morphological closing on the motion vectors, the model produces binary feature images[13].The model locates the cars in each binary feature image[8] using the block analysis block, then it uses the draw shapes block to draw a green rectangle around the cars.

Objects for Reading Video File: Optical flow object for estimating direction and speed of object motion. Create two objects for analyzing optical flow vectors. Filter object for removing speckle noise introduced during segmentation. Morphological closing object for filling holes in blobs.

Create a blob analysis System object to segment cars in the video. Morphological erosion objects for removing portions of the road and other unwanted objects. Create objects for drawing the bounding boxes and motion vectors.

This object will write the number of tracked cars in the output image. Create System objects to display the original video, motion vector video, the threshold video and the final result.



Firgue-1
Working Flow Chart

Simulation Results

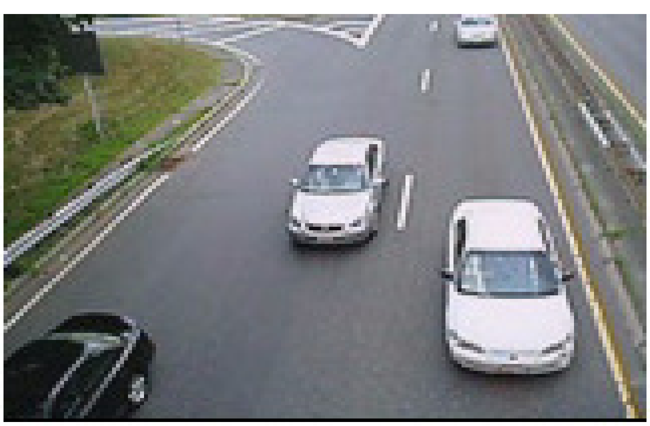


Figure-2
Original video

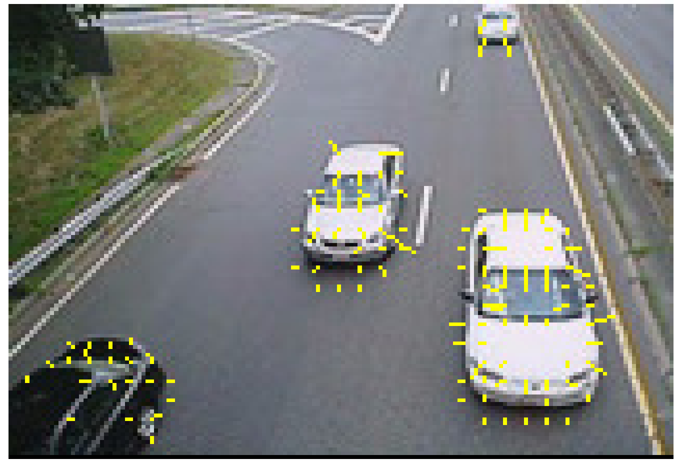


Figure-3
Motion vector

Figure-4
Threshold video

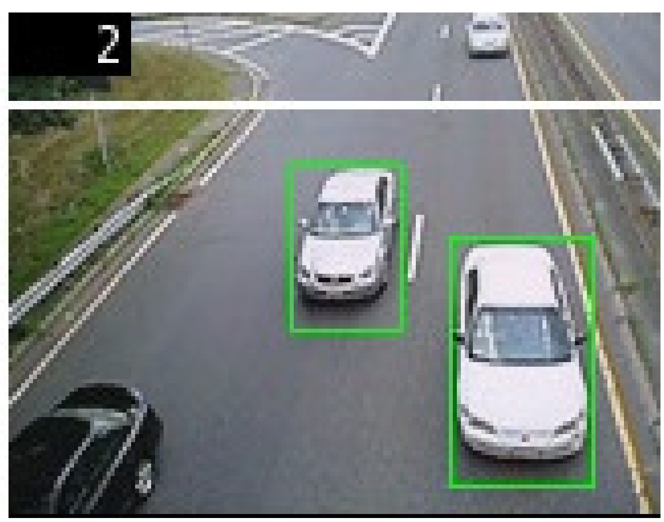


Figure-5
vehicle detection

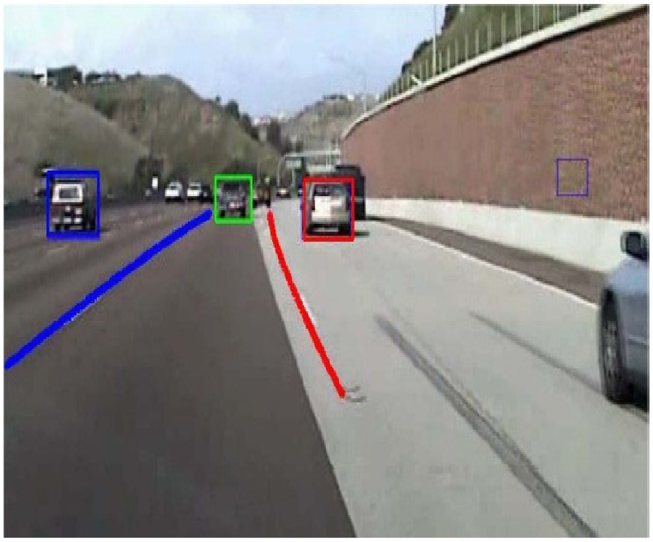


Figure-6
with respect to lane vehicle detection

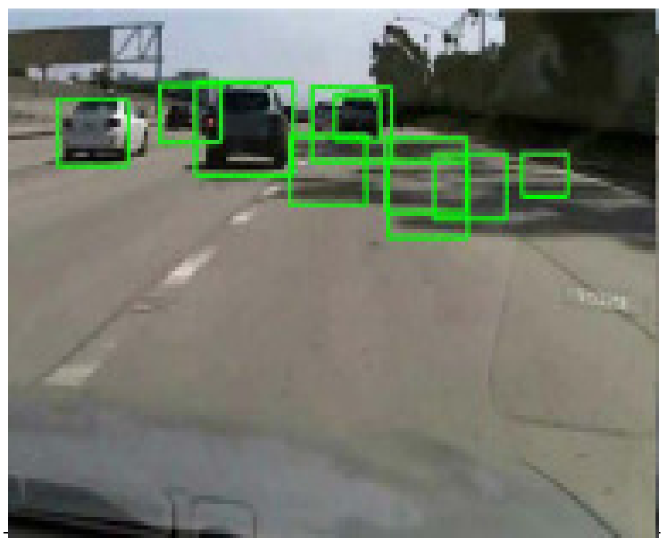


Figure-7
Localizing and tracking other vehicle on the load

Conclusion

First improved the performance of the lane tracking system, and extended its robustness to high-density traffic scenarios. Second, improved the precision of the vehicle tracking system, by enforcing geometric constraints on detected objects, derived from the estimated ground plane. Third introduced an approach to localizing and tracking other vehicles on the road with respect to the estimated lanes.

Future Work: The future work is improved version of both lane and vehicle tracking system. Next it has determined improving version of localizing and tracking other vehicle on the road with respect to lane position.

References

1. Doshi A. and Trivedi M., on the roles of eye gaze and head dynamics in predicting driver's intent to change lanes, *IEEE Trans. Intell. Transp. Syst.,* **10(3),** 453–462 **(2009)**
2. Takeuchi A., Mita S. and McAllester D., On-road vehicle tracking using deformable object model and particle filter with integrated likelihoods, *Proc. IEEE IV Symp.,* 1014–1021 **(2010)**
3. Murphy- Chutorian E. and Trivedi M., Head pose estimation and augmented reality tracking: An integrated system and evaluation for monitoring driver awareness, *IEEE Trans. Intel. Transp. Syst.,* **11(2),** 300–311 **(2010)**
4. Sivaraman S. and Trivedi M., Real-time vehicle detection using parts at intersections, *Proc. 15th Int. IEEE ITSC,* 1519–1524 **(2012)**
5. Oliveira L., Nunes U. and Peixoto P., on exploration of classifier ensemble synergism in pedestrian detection, IEEE Trans. Intell. Transp. Syst., **11(1),** 16–27 **(2006)**
6. H. Gomez- Moreno, S. Maldonado-Bascon, P. Gil-Jimenez and S. Lafuente-Arroyo, Goal evaluation of segmentation algorithms for traffic sign recognition, *IEEE Trans, Intell. Transp. Syst.,* **11(4),** 917–930 **(2010)**
7. Malley R.O., Jones E. and Glavin M., Rear-lamp vehicle detection and tracking in low-exposure color video for night conditions, *IEEE Trans. Intell. Transp. Syst.,* **11(2),** 453–462 **(2010)**
8. Doshi A., Cheng S.Y. and Trivedi M., A novel active heads-up display for driver assistance, *IEEE Trans. Syst., Man, Cybern. B, Cybern.,* **39(1),** 85–93 **(2009)**
9. Kim Z., Robust lane detection and tracking in challenging scenarios, *IEEE Trans. Intell. Transp. Syst.,* **9(1),** 16–26 **(2008)**
10. Enzweiler M. and Gavrila D., A mixed generative-discriminative framework for pedestrian classification, *Proc. IEEE Conf. CVPR,* 1–8 **(2008)**
11. Sivaraman S. and Trivedi M., A general active-learning framework foron-road vehicle recognition and tracking, *IEEE Trans. Intell. Transp.Syst.*, **11(2),** 267–276 **(2010)**
12. Danescu R. and Nedevschi S., Probabilistic lane tracking in difficult road scenarios using stereovision, *IEEE Trans. Intell. Transp. Syst.,* **10(2),** 272–282 **(2009)**
13. Sivaraman S. and Trivedi M., Active learning for on-road vehicle detection: A comparative study, *Mach. Vis. Appl.,* **16,** 1–13 **(2011)**
14. Morris A. and Trivedi M., Unsupervised learning of motion patterns of rear surrounding vehicles, *Proc. IEEE ICVES,* 80–85 **(2009)**
15. Jazayeri A., Cai H., Zheng J.Y. and Tuceryan M., Vehicle detection and tracking in car video based on motion model," *IEEE Trans. Intell. Transp.Syst.,* **12(2),** 583–595 **(2011)**
16. Kembhavi A., Harwood D. and Davis L., Vehicle detection using partial least squares, *IEEE Trans. Pattern Anal. Mach. Intell.,* **33(6),** 1250–1265 **(2011)**

17. Broggi A., Cerri P., Ghidoni S., Grisleri P. and Jung H.G., A new approach to urban pedestrian detection for automatic braking, *IEEE Trans. Intell. Transp. Syst.,* **10(4),** 594–605 **(2009)**
18. Loose H., Franke U. and Stiller C., Kalman particle filter for lane recognition on rural roads*, Proc. IEEE Intell. Veh. Symp,* 60–65 **(2009)**
19. Meuter M., Muller-Schneiders S., Mika A., Hold S., Nunn C. and Kummert A., A novel approach to lane detection and tracking, *Proc.12th Int. IEEE ITSC,* 1–6 **(2009)**
20. Sivaraman S. and Trivedi M., Improved vision-based lane tracker performance using vehicle localization, *Proc. IEEE IV Symp.,* 676–681 **(2010)**