Asian Journal of Pharmaceutical and Clinical Research

Special Issue (April)

Online - 2455-3891
Print - 0974-2441

Full Proceeding Paper

HYPERPROPERTIES-BASED OPTICAL FLOW-BASED AUTONOMOUS DRIVING SYSTEM

ANKUSH RAI, JAGADEESH KANNAN R

School of Computer Science & Engineering, VIT University, Chennai, Tamil Nadu, India. Email: ankushressci@gmail.com

Received: 13 December 2016, Revised and Accepted: 03 April 2017

ABSTRACT

Objective: This study presents an autonomous driving system based on the principles of trace vectors derived from hyperproperty of a modified optical flow algorithm.

Methods: A novel multiple object tracking algorithm based on a modified morphology of optical flow field information with trace vectors derivable from hyper properties of pixels is presented.

Results and Conclusion: This technique allows keeping track of the past motion vectors by tracking the constraint sets to overcome the non-linear attributes of the deformable feature points and motion vectors. The results presented in this work exhibits stable tracking and multi-step prediction in a limited number of steps with less training vectors.

Keywords: Object detection, Multiple object tracking, Optical flow, Object classification.

INTRODUCTION

Tracking multiple objects in a video sequence is a bit challenging though it has been an important problem in computer vision research area. As this involves detection and tracking of multiple objects; which requires prior training and classification algorithms for each timed step in the video sequence of each frame. The problem becomes more challenging when the objects in the scenes are unknown, and the counts of objects are variable. Mahler proposed a finite set statistics approach for multiple object tracking; which is based on systematic treatment of the object count [1]. To reduce the complexity of the method, Mahler put forth an approximation technique for the Bayes multi-target filter which is known as probability hypothesis density filter (PHD). There were other studies which points toward the outperform performance of PHD filter in comparison with Kalman filter, standard particle filter, and multiple hypothesis tracking [2,3]. In latter studies, the joint probabilistic data association filter is introduced to merge the results of tracking through closely spaced objects [4].

In other studies, PHD filter was used alone [5,6]. This has the disadvantage of working with only information in regard to the object's position which was gained from detector and latter being used to estimate velocity vectors. It reduces the robustness of the filter [7-9]. Other studies used PHD filters with optical flow method for extracting imagery data and velocity vectors [10-17]. Since the velocity vectors cannot be extracted from every pixel. Thus, the previous methods are unable to provide the position information of the object in the scene. There is a need for a fast strategy for computing the position and velocity vectors for multiple objects which we have accomplished in the following sections.

METHODS

Experimental setup

The proposed model is implemented using MATLAB R2012a under Windows platform. The experiments are conducted on the machine with hardware configurations of Intel's third generation 8-core microprocessor with Nvidia 630 graphic card, 2 GB RAM giving a fine clocking speed of 2.7 GHz. The consolidated database used in the study is available online at [18]. The images used in the study consist of frontal and side pose with different types of compression schemes. The properties of the used images are represented in Table 1.

The model

Let I_0, I_1, and I_2 be the three input images where I_0, I_1, and I_2: $\Omega \epsilon R^2 \rightarrow [0,1]$. Therefore, we can rewrite the optical flow model as follows:

$$\min\left\{\int_{\Omega}\sum_{d=1}^{2}\left|\nabla a_d\right| + \lambda\left|\delta(a(x))\right|dx\right\} \quad (1)$$

With $a(x)=(a_1(x), a_2(x))^T$, a_d: $\Omega \rightarrow R$, where λ is the parameter to balance the regularization of the data for relative weight between the pair of subsequent frames, that is, I_0 and I_1, I_1 and I_2 and the pair of subsequent frames from the given video sequences. Furthermore, $\delta(a(x)) = a(x)^T \nabla I_1(x) + I_1(x) - I_1(x)$ is the constrained equation of the optical flow model. Here, we have used Huber anisotropic regularization approach based on the following reasons [19-21]:

A. It ensures the optimal solutions

B. The parallelization of it allows the minimization of the computational workload. Thereby bestowing the advantage of processing massive operations.

This allows to compute every pixel sets x ε Ω in each step evolution represented by k for the velocity of the pixel determined $a_k(x)=(a_1, a_2)^T$.

Now, we are required to model an object detector with the flow field derived from the previous equations. Therefore, here we group the pixel sets based on the behavioral pattern of the flow fields as:

If the detected motion of the optical flow fields is represented by μ_k belongs to the moving object at any time, evolution step k is computed with the probability function for the motion is given by:

$$P_{motion}(x) = 1.0 - exp\left(-\frac{1}{2}\frac{\left(\sum_{i=1}^{m}\sum_{j=1}^{n}\sigma a_i - b_{jn}^{m} - \mu\right)^2}{\sigma^2}\right)$$

Table 1: Enlisted database used in the experiment

Database and compression type	Annotations per scenario	Distance covered (km)	Total objects detected in a trip
H.264 compression with different bitrates (H.264)	4	2	16
HEVC/H.265 compression with different QP values (HEVC)	7	6	49
Motion JPEG compression (MJPEG)	12	20	88
Wavelet-based compression using Snow codec (SNOW)	19	40	194

Where, μ and σ are the corresponding distributions of the pixel from that of stationary camera and relative to that of the stationary object in the background of the pixel sets (b_j) in the frame. Therefore, the actual velocity of the moving object can be determined from the computed center of gravity for every region corresponding to the detected regions for covariant vectors at time k. $X(x_k|y^k)$ as the density of states for the segmented motion vectors is given by:

$$X\left(x_k|y^{k-1}\right)=b\left(x_k\right)+\int\left[P_e\left(x_{k-1}\right)P_{motion}\left(x_k|x_{k-1}\right)+b\left(x_k|x_{k-1}\right)\right] D(x_{k-1}|y^{k-1})dx_{k-1}$$

Where, $b(x_k)$ describes the intensity function of spontaneous appearance or birth of the new objects, $b(x_k|x_{k-1})$ is the traced intensity function of the vector sets spawned in previous steps, that is, x_{k-1}. $P_e(x_{k-1})$ is the probability of existence of the object in the frame and $P_{motion}(x_k|x_{k-1})$ is the transition probability of index objects. Thus, the updation of multiple objects is given by:

$$x_k=\begin{pmatrix} I_{n+1}b\left(x_k|x_{k-1}\right) & TI_{n+1} \\ I_{n+1}P_e\left(x_{k-1}\right) & I_{n+1}P_s \end{pmatrix} x_{k-1}+X(x_{k-1}|y^{k-1})$$

Here, ΔT is the time difference between consecutive time evolution steps k−1 and k. The workflow process of the algorithm is given in Fig. 1, whereas the algorithm for the multiple object tracking process is given as:

Algorithm: Hyperproperties-based optical flow algorithm for motion estimation

Input: I_0, I_1, I_2...I_n be the input frames where $I_n: \Omega \in R^2 \rightarrow [0,1]$

Output: The density of states for the segmented motion vectors $X\left(x_k|y^{k-1}\right)$.

Step 1: Evaluate optical flow field between the consecutive pair of frames by evaluating:

$$\min\left\{\int_{\Omega}\sum_{d=1}^{2}\left|\nabla a_d\right|+\lambda\left|\delta(a(x))\right|dx\right\}$$

Step 2: Compute the probability function for the motion of the moving object at time evolution step k:

$$P_{motion}\left(x\right)=1.0-\exp\left(-\frac{1}{2}\frac{\left(\sum_{i=1}^{m}\sum_{j=1}^{n}\sigma a_i-b_{jn}^{m}-\mu\right)^2}{\sigma^2}\right)$$

Step 3*:* Determine the trace and covariant vectors of the actual velocity of an object in motion in the form of grouped pixels with its center of gravity which corresponds to the detected regions at time k:

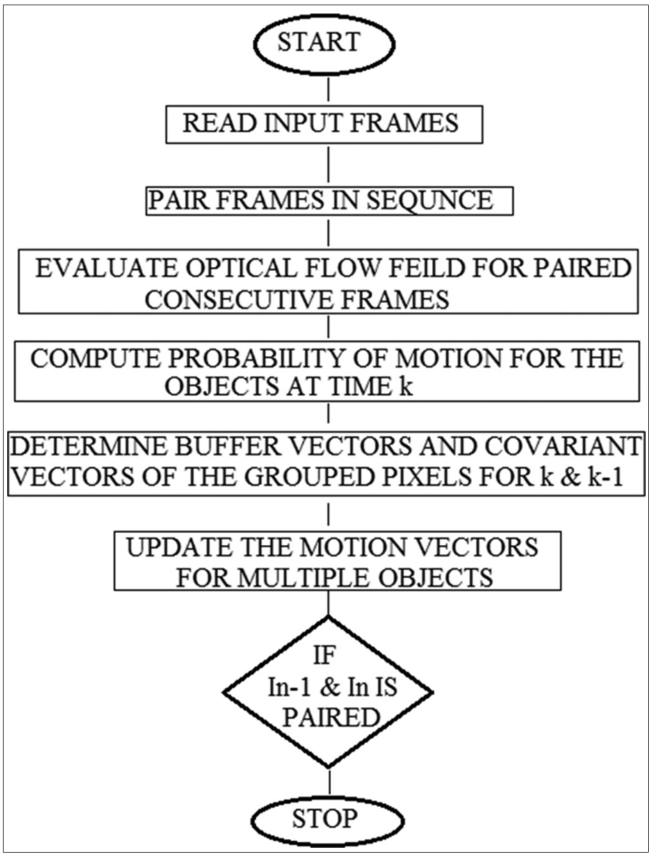
START
READ INPUT FRAMES
PAIR FRAMES IN SEQUNCE
EVALUATE OPTICAL FLOW FEILD FOR PAIRED CONSECUTIVE FRAMES
COMPUTE PROBABILITY OF MOTION FOR THE OBJECTS AT TIME k
DETERMINE BUFFER VECTORS AND COVARIANT VECTORS OF THE GROUPED PIXELS FOR k & k-1
UPDATE THE MOTION VECTORS FOR MULTIPLE OBJECTS
IF In-1 & In IS PAIRED
STOP

Fig. 1: Flowchart of the multiple object tracking algorithm

$$X\left(x_k|y^{k-1}\right)=b\left(x_k\right)+\int\left[P_e\left(x_{k-1}\right)P_{motion}\left(x_k|x_{k-1}\right)+b\left(x_k|x_{k-1}\right)\right] D(x_{k-1}|y^{k-1})dx_{k-1}$$

Step 4: Update the motion vectors for multiple objects in consecutive frames by:

$$x_k=\begin{pmatrix} I_{n+1}b\left(x_k|x_{k-1}\right) & \Delta TI_{n+1} \\ I_{n+1}P_e\left(x_{k-1}\right) & I_{n+1}P_s \end{pmatrix} x_{k-1}+X(x_{k-1}|y^{k-1})$$

Step 5*:* Repeat steps 1-4; until I_{n-1}, I_n is paired.

Step 6: End process.

RESULTS AND DISCUSSION

The image sequence in Fig. 2a represents the initialized optical flow field for the zero object in the frame of the given car trip. This represents the segmented flow field for the motion of the camera mounted on the

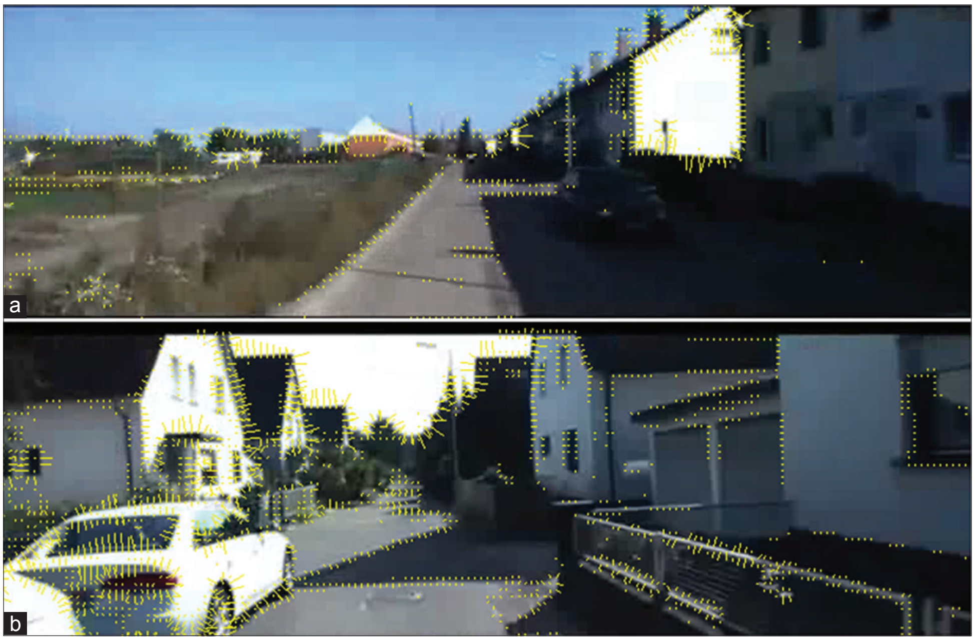

Fig. 2: (a) Result of deformable multiple object tracking, (b) Result of the position and structural feature extraction for the deformable objects from the scene through the proposed algorithm

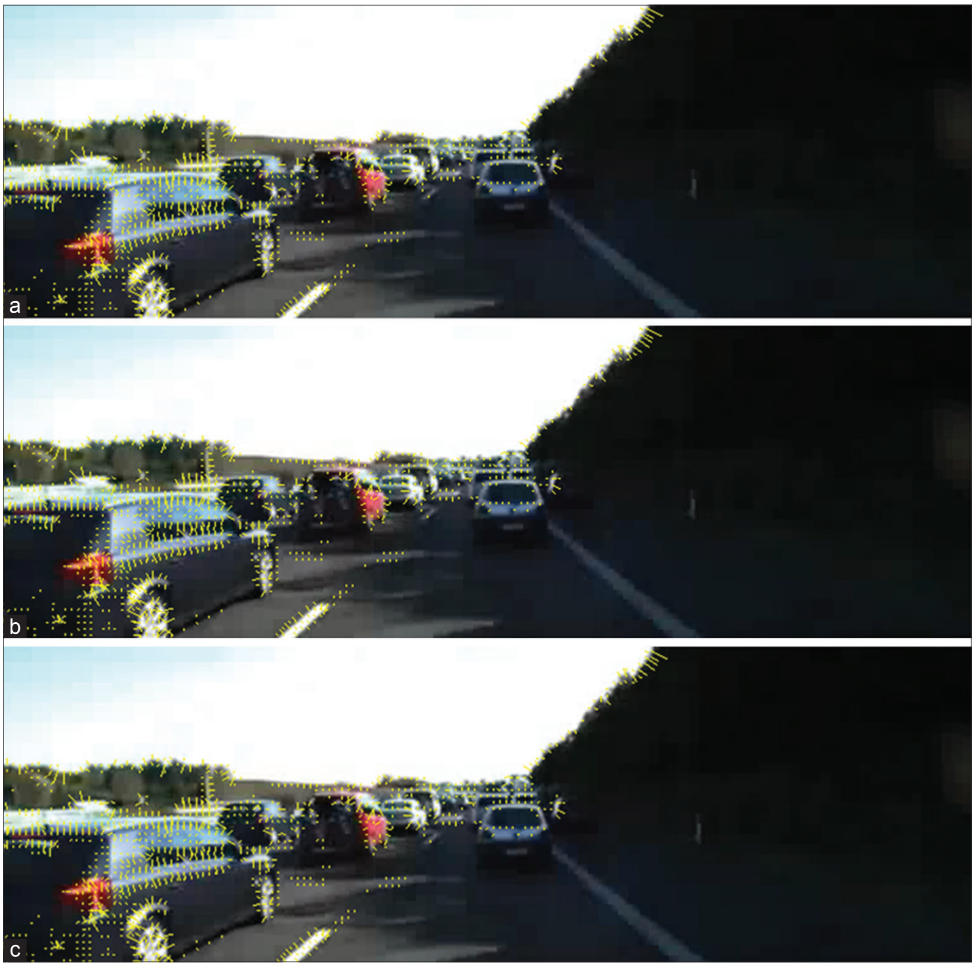

Fig. 3: Trace vectors carried over three consecutive frames (a-c), respectively

moving car. As shown in the image, the optical flow field is effectively segmented the vectors of the structure and position information of the objects. Whereas, Fig. 2b represents the extracted structural features in the form of quivers for the deformable objects. Here, the deformation

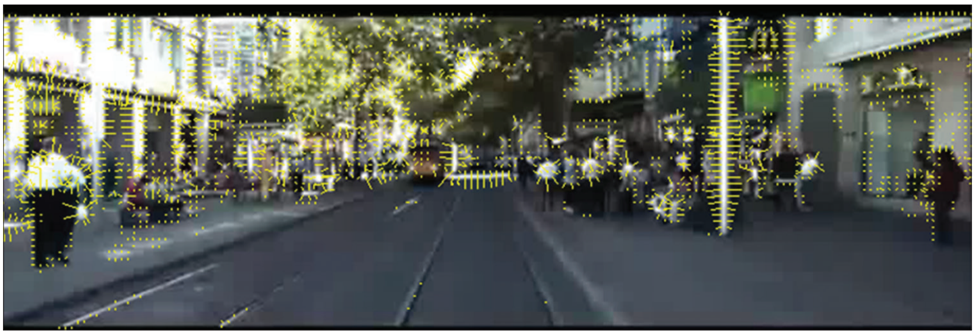

Figure 4: Occlusion handling for the flow field of multiple objects in deformable objects

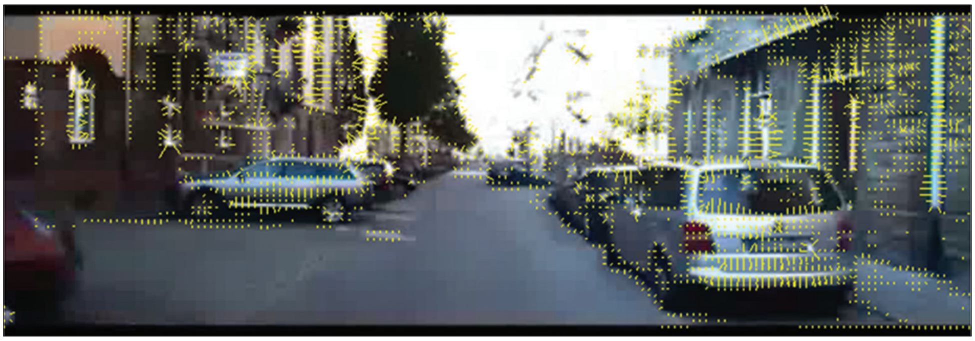

Fig. 5: Illustration of tracker vectors for tracking rigid object in the speeding car scenario

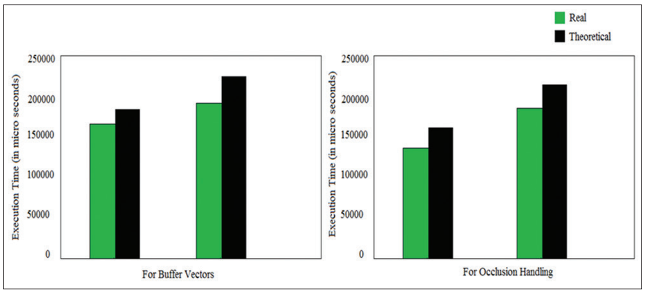

Fig. 6: Comparison between the theoretical and the actual execution times for multiple object tracking

is considered in the form of relative orientation of the object's position. Thus, changes in the visualization of perceptual view for object is overcome with compensation of non-prior training of the algorithm required for object's classification. This handles the real-time scenario of object's variation in geometrical representation from any perceptual or orthogonal perspective. Furthermore, Fig. 3a-c illustrates the notion of trace vectors in the given consecutive frames with a speed of 25 frames per second, involving 2500 particles. As shown in the figure, the motion vectors from previous frames are carried over to the next frames. Thereby adding the depth to the predictability of positional and motion vectors in next frame. Fig. 4 illustrates a case scenario of sticking the feature points in the form of segmented optical flow fields to handle the occlusion problem of multiple deformable objects. We have also studied the result of the algorithm for the worst case scenario for non-deformable objects from a camera view of speeding car to determine the motion vectors when the forward motion is essential to avoid motion blur in order to increase the influence of the flow field during object tracking the flow field tracking. Thus, as shown in the Fig. 5, the modified morphology of optical flow field in the proposed study effectively negates the effects in such worst case scenarios with a standard deviation of 2.56 through 2.5 pixels bands with 1.87% of error in the localization of the motion vectors.

CONCLUSION

The study presents a novel multiple object tracking algorithm based on a modified morphology of optical flow field information with trace vectors. In addition, we demonstrated the improved results in worst case scenario of the tracking in computer vision for car driving applications; not only it handles the occlusion problem but also helped picking out the merged missing detections. The Fig. 6 represents the plot of the execution time of the algorithm for multiple objects in the given car trip for trace vectors and that required for occlusion handling. Here, the bar chart shows the real and theoretical differences between the execution time of the algorithms. The differences arise due to the forward motion blurring and the camera calibration issues adhering to adverse the frame rates per second for the speeding car trip. It is able to deal with the clutter. The algorithm greatly reduces the execution time for multiple objects as it offers a great deal of parallelization and option of threading the algorithm. The future study of the research shall aim to compute the algorithm with a graphics processing unit computing scheme over multiple processing cores.

REFERENCES

1. Mahler R. Multitarget bayes filtering via first-order multitarget moments. IEEE Trans Aerosp Electron Syst 2003;4(39):1152-78.
2. Panta K, Vo BN, Singh S, Doucet A. Probability hypothesis density filter versus multiple hypothesis tracking. Proc SPIE 2004;5429:284-95.
3. Juang R, Burlina P. Comparative Performance Evaluation of GM-PHD Filters in Clutter. In: FUSION, July; 2009.
4. Bar-Shalom Y, Fortmann TE, Scheffe M. Joint probabilistic data association for multiple targets in clutter. In: Conference on Information Sciences and Systems; 1980.
5. Wang Y, Wu J, Kassim A, Huang W. Tracking a variable number of human groups in video using probability hypothesis density. In: International Conference on Pattern Recognition; 2006.
6. Maggio E, Taj M, Cavallaro A. Efficient multi-target visual tracking using random finite sets. IEEE Trans Circuits Syst Video Technol 2008;18(8):1016-27.
7. Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition; 2005.
8. Schikora M. Global optimal multiple object detection using the fusion of shape and color information. In: Energy Minimization Methods in

Computer Vision and Pattern Recognition; 2009.

9. Weickert J, Bruhn A, Brox T, Papenberg N. A survey on variational optic flow methods for small displacements. Math Models Registration and Appl Med Images 2006;10:103-36.
10. Baker S, Schastein D, Lewis JP, Roth S, Black MJ, Szeliski R. A database and evaluation methodology for optical flow. In: International Conference on Computer Vision; 2007.
11. Werlberger M, Trobin W, Pock T, Wedel A, Cremers D, Bischof H. Anisotropic Huber-L1 optical flow. In: British Machine Vision Conference. London, UK, September; 2009.
12. Vo BN, Singh S, Doucet A. Sequential Monte Carlo methods for multi-target filtering with random finite sets. IEEE Trans Aerosp Electron Syst 2005;41(4):1224-45.
13. Carpenter GA, Grossberg S. ART 2: Self-organizing stable category recognition codes for analog input patterns. Appl Opt 1987;26(23):4919-30.
14. Nummiaro K, Koller-Meier E, Van Gool L. An adaptive color-based particle filter. Image Vis Comput 2002;21(1):99-110.
15. Davis J, Sharma V. Background-subtraction using contour-based fusion of thermal and visible imagery. Comput Vis Underst 2007;106(2-3):162-82.
16. Pellegrini S, Ess A, Schindler K, van Gool L. You'll never walk alone: Modelling social behaviour for multi-target tracking. In: International Conference on Computer Vision; 2009.
17. Sahu AK, Choubey A. Motion detection surveillance system using background subtraction algorithm. Int J Adv Res Comput Sci Manag Stud 2013;1(6):58-65.
18. The KITTI Vision Benchmark Suite. Available from: http://www.cvlibs.net/datasets/kitti/.
19. Rai A. Attribute based level adaptive thresholding algorithm for object extraction. J Adv Robot 2015;12:64-8.
20. Rai A. An introduction of smart self-learning shell programming interface. J Adv Shell Program 2015;12:3-6.
21. Rai A. Dynamic data flow based spatial sorting method for GPUs: Software based autonomous parallelization. Recent Trends in Parallel Comput 2014;11:15-8.