

Moving Vehicle Segmentation Based on EM Algorithm and Fast Motion Estimation Algorithms

Ms. Priya Joy

CSE Department

VJCET, Vazhakulam, Muvattupuzha, Kerala, India

ABSTRACT

Moving vehicles segmentation is an interesting yet difficult problem in intelligent transportation system. Ordinary segmentation method is obstructed by some problems: the moving objects in dynamic scenes are the examples. Bayesian framework is utilized to classify the motion in the scenes to improve the robustness of the model and EM algorithm is used to estimate the parameters of the model. The direction and magnitude of the motion vectors are the inputs to the Bayesian rule and EM algorithm. In existing methods traditional Exhaustive search method is used for finding the motion vector. Computational cost of the Exhaustive search method is high. In this paper we are using some fast algorithms like Three Step Search and Four Step Search algorithms for reducing the computational cost. Experimental results on the scene, 'waving trees', shows that the proposed model can segment the moving vehicles correctly with less computational cost. Quantitative evaluations demonstrate that the proposed method outperform the existing methods.

Keywords:

Bayesian framework; EM algorithm; Motion vector; Exhaustive search algorithm; Three step search algorithm; Four step search algorithm

1. INTRODUCTION

Moving vehicles segmentation has been actively investigated in the past decade [5, 6]. In real world scenes, challenges remain in the moving vehicles segmentation,

such as unstableness of background scenes. Background subtraction is an effective technique for foreground objects detection in stationary background. [9, 5] However, there are usually instable objects in the background of scenes, such as waving trees and flags fluttering. Traditional methods are able to cope with static background scenes. However, their performance deteriorates when the scene is non-stationary over time.

Gupte et al. [5] proposed a method for detecting moving vehicles by using background subtraction. But Background subtraction will be an effective technique for foreground objects detection only in the case of stationary background. Kim et al. [7] applied the bayes decision rule with a Gaussian probability density function. to each

pixel in the current frame, in order to obtain a segment of the moving object. This paper also deals with static background.

Friedman and Russell [4] proposed a method to classify the pixel values into three separate, predetermined distributions corresponding to road color, shadow color and vehicle color. It can be seen that our method differs from Friedman's. Friedman's method has superiority over our method by its shadows detection, while our method gets an advantage over Friedman's method for its immunity from the unstableness of the scenes.

Wei Zhang et al [1] proposed Gaussian motion model for moving vehicles segmentation in the dynamic scenes. They investigated the distinction between the motion vectors of the dynamic background and those of the moving vehicles; they found that the motion vectors of the moving vehicles cluster in a small region while those of the dynamic background are dispersive. Consequently, Gaussian motion model is proposed to model the motion of the moving pixels in the scene, and Bayesian framework is employed to classify moving pixels into the moving vehicle or the dynamic background.

But in their work the computational cost is high. Most of the computational cost is occupied by the computation of the motion vector for every moving pixel. In their work the traditional exhaustive block matching method is used to compute the motion vector pixel-wised. It can be seen that our method differs from Wei Zhangs method. In our paper we are using some fast algorithms like Three Step Search and Four Step Search for finding the motion vector and reduce the computation cost.

2. MOVING VEHICLE SEGMENTATION

Moving vehicle segmentation method has 3 main steps.

1. Background subtraction using adaptive background Estimation
2. Motion vector calculation
3. Segmentation using EM algorithm.

2.1. Background subtraction (BS) using adaptive background Estimation

It is necessary to extract moving pixels before performing the moving vehicle segmentation. Background subtraction to segment the moving pixels. Let CB and Fn be current background image

and nth frame, respectively. Then the moving pixels of frame n, MPn, is computed as

$$MP_n(x, y) = \begin{cases} 1 & \text{if } |CB(x, y) - F_n(x, y)| > T_n \\ 0 & \text{else} \end{cases}$$

Where T_n is computed according to Otsu, [8].

To extract CB from the dynamic scenes, we use the adaptive background update method proposed in [5], which is briefly described as follows. Firstly the background subtraction and the frame difference are implemented to extract the mask:

$$Obj_mask(x, y) = \begin{cases} 1 & \text{if } |F_n(x, y) - CB(x, y)| > T_o \\ 0 & \text{else} \end{cases}$$

$$Mo_mask(x, y) = \begin{cases} 1 & \text{if } |F_n(x, y) - F_{n-1}(x, y)| > T_m \\ 0 & \text{else} \end{cases}$$

Mask = Mo_mask & Obj_mask;

The thresholds, T_o and T_m , are computed according to Otsu, [8]. Secondly, instantaneous background IB is defined by sampling the mask areas in CB and the none mask areas in F_n :

$$IB(x, y) = \begin{cases} F_n(x, y) & \text{if } mask(x, y) = 0 \\ CB(x, y) & \text{if } mask(x, y) = 1 \end{cases}$$

Finally the background is updated as the weight sum of IB and CB

$$CB = \alpha \times IB + (1 - \alpha) \times CB$$

The weight α is empirically determined to be 0.1 in Gupte et al., [5].

Up to now we have segmented the moving pixels MPn. Nevertheless, there may be ‘holes’ in the segmented moving objects (even in the vehicles to be detected), which is not consistent with the reality. Thus, we apply flood fill operation on MPn and get the resultant moving pixel MPFn. [1]

2.2 Motion vector calculation

Before going to the segmentation part we have to find out the motion vector of moving pixels. In this paper we tried 3 algorithms for finding the motion vector.

1. Exhaustive search algorithm
2. Three step search algorithm
3. Four step search algorithm

2.2.1. Exhaustive search algorithm

This algorithm, also known as Full Search, It is the most computationally expensive block matching algorithm [2]. This algorithm calculates the cost function at each possible location in the search window. The traditional MSE block matching method is used to compute the motion vector pixel-wised: For pixel(x, y) an

nxn image block with (x, y) being in the center of it is extracted and the motion vector of the block represents that of pixel(x, y).

2.2.2. Three step search algorithm

TSS starts with the search location at the center and sets the ‘step size’ $S = 4$, for a usual search parameter value of 7. It then searches at eight locations $\pm S$ pixels around location (0, 0). From these nine locations searched so far it picks the one giving least cost and makes it the new search origin. It then sets the new step size $S = S/2$, and repeats similar search for two more iterations until $S = 1$. At that point it finds the location with the least cost function and the macro block at that location is the best match. The calculated motion vector is then saved for transmission. It gives a flat reduction in computation by a factor of 9. So that for $p = 7$, ES will compute cost for 225 macro blocks whereas TSS computes cost for 25 macro blocks. The general idea is represented in Figure 1.

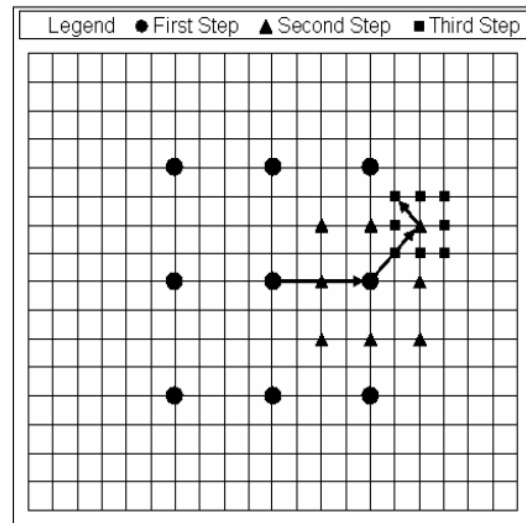
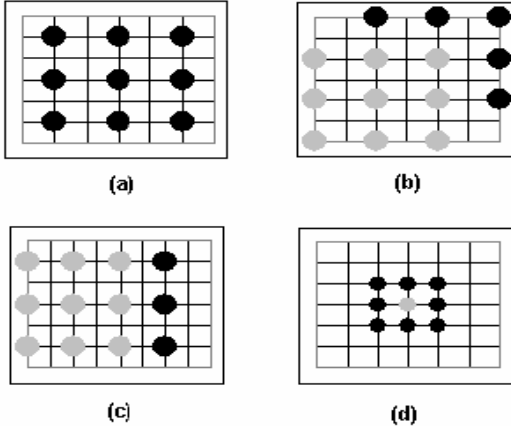


Fig1: three Step Search procedures. The motion vector is (5, -3). [2]

2.2.3. Four step search

4SS [2] sets a fixed pattern size of $S = 2$ for the first step, no matter what the search parameter p value is. Thus it looks at 9 locations in a 5x5 window. If the least weight is found at the center of search window the search jumps to fourth step. If the least weight is at one of the eight locations except the center, then we make it the search origin and move to the second step. The search window is still maintained as 5x5 pixels wide. Depending on where the least weight location was, we might end up checking weights at 3 locations or 5 locations. Once again if the least weight location is at the center of the 5x5 search window we jump to fourth step or else we move on to third step. The third is exactly the same as the second step. IN the fourth step the window size is dropped to 3x3, i.e. $S = 1$. The location with the least weight is the best matching macro block and the motion vector is set to point o that location. The general idea is represented in Figure 2.



Search patterns of the FSS. (a) First step (b) Second/Third step (c) Second/Third Step (d) Fourth Step [2]

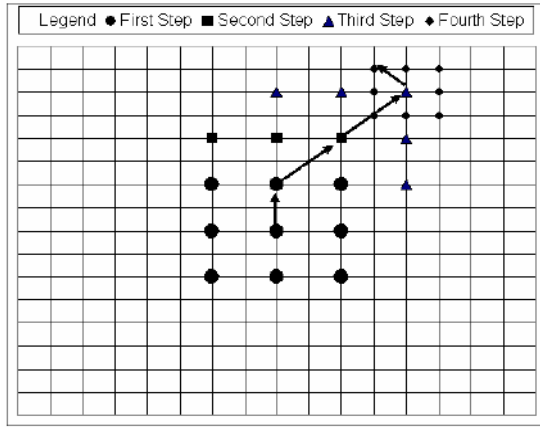


Fig2: four Step Search procedures. The motion vector is (3, -7). [2]

2.3 Segmentation using EM algorithm

We now propose our Gaussian motion model and combine it with Bayesian framework [1]. Suppose that the moving pixels in MPFn are partitioned into two classes: ω_1 (the moving pixels belong to the moving vehicles) and ω_2 (the moving pixels belong to the background). Let $M_{i,n} = (r_{i,n}, \theta_{i,n})$ be the motion vector of the i th moving pixel in polar coordinates in MPFn, and for $M_{i,n}$, the conditional probability that the event x_1 will occur is given by $P_n(x_1|M_{i,n})$:

$$P_n(\omega_1|M_{i,n}) = \frac{P_n(M_{i,n}|\omega_1)P_n(\omega_1)}{P_n(M_{i,n}|\omega_1)P_n(\omega_1) + P_n(M_{i,n}|\omega_2)P_n(\omega_2)}$$

The corresponding pixel is detected as the moving vehicles if the following decision rule is satisfied:

$$P_n(M_{i,n}|\omega_1)P_n(\omega_1) > P_n(M_{i,n}|\omega_2)P_n(\omega_2)$$

Assume that the direction and amplitude of the motion vectors are statistical independent, we have:

$$P_n(M_{i,n}|\omega_1)P_n(\omega_1) > P_n(M_{i,n}|\omega_2)P_n(\omega_2)$$

$j=1, 2$; The Gaussian motion model mathematically as follows.

$$P_n(r_{i,n}|\omega_j) = \frac{1}{\sqrt{2\pi} \sigma_{r,j}^2} \exp \left(-\frac{(r_{i,n} - \mu_{r,j})^2}{2\sigma_{r,j}^2} \right)$$

$$P_n(\theta_{i,n}|\omega_j) = \frac{1}{\sqrt{2\pi} \sigma_{\theta,j}^2} \exp \left(-\frac{(\theta_{i,n} - \mu_{\theta,j})^2}{2\sigma_{\theta,j}^2} \right)$$

To initialize the model parameters, we compute the motion vectors of the moving pixels of the particular frames. And mean ME_p , and variance VI_p , of r and θ are calculated.

Then the Gaussian motion model of the moving vehicle and that of the background are initialized as follows.

$$P(\omega_1) = 0.5 \quad P(\omega_2) = 0.5$$

$$\mu_{p,1} = ME_p, \quad \mu_{p,2} = 0$$

$$\sigma_{p,1}^2 = VI_p, \quad \sigma_{p,2}^2 = VI_p,$$

Where $p = r, \theta$.

2.3.1 EM algorithm based model update

If a pixel is classified to be the moving vehicles or the background, the corresponding Gaussian model is updated as follows.

$$\mu_{r,j} = (1 - \rho_{r,j})\mu_{r,j} + (\rho_{r,j})r_{i,n}$$

$$\mu_{\theta,j} = (1 - \rho_{\theta,j})\mu_{\theta,j} + (\rho_{\theta,j})\theta_{i,n}$$

$$\sigma_{\theta,j}^2 = (1 - \rho_{\theta,j})\sigma_{\theta,j}^2 + (\rho_{\theta,j})(\theta_{i,n} - \mu_{\theta,j})^2$$

$$\sigma_{r,j}^2 = (1 - \rho_{r,j})\sigma_{r,j}^2 + (\rho_{r,j})(r_{i,n} - \mu_{r,j})^2$$

$$\rho_{r,j} = \beta_{r,j} P(r_{i,n} | \omega_j)$$

$$\rho_{\theta,j} = \beta_{\theta,j} P(\theta_{i,n} | \omega_j)$$

$$\beta_{r,j} = \beta_{\theta,j} = 10^{-4}$$

Where $j=1,2$;

$\beta_{r,j}, \beta_{\theta,j}$ Control the sensitivity of the model to the change of the motion in the scenes.

The moving pixels in MPFn are used to update the parameters of the Gaussian models until the change of the segment results before and after updating is less than 1%

3. CONCLUSION

The ROC curve shows that the exhaustive search method performs as a good classifier. But time taken using exhaustive search algorithm is high. From Roc curve we can understand that the four step search method also gives good results. And the time taken using this method is less. From these analyses we can conclude that FSS method is good when the movement is small. Three step search takes more time, compared to FSS. Also result is not good compared to exhaustive search and FSS methods. In 'waving trees' sequence we can found that the movement of pixels from one

frame to next frame is small. So the exhaustive search method and FSS method give better results.

REFERENCES

- [1] Wei Zhang, Xiang Zhong Fang, Xiaokang Yang, "Moving vehicles segmentation based on Bayesian framework for Gaussian motion model ". *Pattern Recognition Letters* Volume 27, issue 9, 1 July 2006 pages 956-967
- [2] Aroh Barjatya, 2004. "Block Matching Algorithms for Motion Estimation". *DIP 6620 Spring Final Project Paper*
- [3] Pless, Larson, Siebers, Westover, "Evaluation of local models of dynamic backgrounds". *CVPR* 2, 2003 pp. 1/73 1/78.
- [4] Friedman, N., Russell, S., 1997. Image segmentation in video sequences: A probabilistic approach. In: *Proc. of 13th Conf. on Uncertainty in Artificial Intelligence*, pp. 175–181.
- [5] Gupte, S., Masoud, O., Martin, R.F.K., Papanikolopoulos, N.P., 2002. Detection and classification of vehicles. *IEEE Trans. Intell. Transport. Syst.* 3, 37–47.
- [6] Ha, D.M., Lee, J.-M., Kim, Y.-D., 2004. Neural-edge-based vehicle detection and traffic parameter extraction. *Image Vision Comput.* 22, 899–907.
- [7] Lee, K.-W., Kim, J., 1999. Moving object segmentation based on statistical motion model. In *Electronics Letters*. 35, pp. 1719–1720.
- [8] Otsu, N., 1979. A threshold selection method from Gray-level histograms. *IEEE Trans. Syst., Man, Cyber.* 9 (1), 62–69. 966 W. Zhang et al. / *Pattern Recognition Letters* 27 (2006) 956–967
- [9] Stauffer, C., Grimson, W.E.L., 1999. Adaptive background mixture models for real-time tracking. In *CVPR* 2, pp. 246–252.