Realtime Detection and Tracking of Moving Objects

C.Senthamizharasi[1], Nivedha[2], Revathi[3] and Manoranjani[4]

[1]Assistant Professor, Department of Electronics and Communication Engineering, Sri Krishna College of Engineering and Technology, Coimbatore, India.
[2]UG Scholar, Department of Electronics and Communication Engineering, Sri Krishna College of Engineering and Technology, Coimbatore, India.
[3]UG Scholar, Department of Electronics and Communication Engineering, Sri Krishna College of Engineering and Technology, Coimbatore, India.
[4]UG Scholar, Department of Electronics and Communication Engineering, Sri Krishna College of Engineering and Technology, Coimbatore, India.

Article Received: 22 March 2017 *Article Accepted: 31 March 2017* *Article Published: 03 April 2017*

ABSTRACT

This project focuses on the detection of foreground and tracking of moving objects in video. Since tracking of moving object is very important in many video processing applications, several research has been made on automated video surveillance to monitor illegal or accidental activities in urban traffic. In early works, single Gaussian model was used to model the value variations of each pixel and the parameters of Gaussian model were updated recursively with an adopted filter. It is robust in modeling the static background but it is sensitive to the dynamic background variations. To rise above works, we go for hybrid model. It is the combination of course fine detection theory algorithm and SILTP color and texture information. To represent observed image regions we apply a new binary descriptor called galaxy because it is very efficient for computation compared to state-of-the-art descriptors. Along with tracking and detection we can also count the number of vehicles in input video By using Blob analysis. We can elucidate this idea using MATLAB software through simulation which involves detection of an object in black and white format, color format and also tracking and counting the objects in a given input video.

Keywords: Video, Tracking, Moving objects and Texture.

1. Introduction

The use of video is becoming prevalent in many applications such as monitoring of traffic, detection of pedestrians, identification of anomalous behavior, etc., while a single image provides a portrait of a scene, the different frames of a video taken over time registers making it possible to capture motion in the sequence. Motion is very important in making objects easy to recognize as soon as they move, even if they are inconspicuous when still. Motion also carries information about the spatio-temporal relationships between objects. This allows us to model their interactions, enabling us to identify normal patterns and detect unusual events. The detection of moving objects must be accurate and robust to minimize false alarms and missed positives, and in real-time to enable corrective action. Moving object detection plays a major role in various video processing applications such as object categorization, re-identification, tracking and video condensation. It often serves as pre-processing for higher-level video analyses that directly affects the functioning of the subsequent applications.

2. ALGORITHMS USED

In this work, we propose a new integration framework of color and texture information which can inherit the earlier used Gaussian mixer model's various disadvantages while inhibiting their advantages. Since background modeling is usually a pre-treatment for higher-level video analyses, it should be computationally efficient. Initially, we give the input video and perform

(a) Coarse-to-fine detection theory algorithm based on background and foreground models represented by binary descriptors is used to extract foregrounds. The foreground extraction problem is formulated as coarse-to-fine binary hypothesis-testing problems in the coarse (Region) and fine (Pixel) levels, respectively. By estimating the maximum a posterior (MAP) probability of the testing regions and pixels with respect to the hypotheses, each hypothesis requires to build its own models.

(b) Then, we use the SILTP Algorithm.

(c) Active Contour Algorithm and then construct one model for each block, which is different from the pixel based strategy that has one model for each pixel so that, A lot of computational resources can be saved.
Traditional block based methods construct a model of several histograms for each block to deal with dynamic background and multimodal problems and make foreground detection decision for new frames by histogram matching, which are also time-consuming. Instead, we use

(d) BLOB ANALYSIS-Block wise background modeling to construct the background model of just one histogram with its bins indicating the probabilities of their corresponding patterns in the block. Since all the frequently appearing patterns can be dominant in the model histogram, we are able to deal with dynamic background and multimodal problems.

3. SOFTWARE EXPLANATION

We have used MATLAB Software for our project as it has several advantages in making us obtain our desired output in desired format.

MATLAB, the high-performance language integrates computation, visualization, and programming which provide an easy platform where problems and solutions are expressed in mathematical notation.

3.1 Toolboxes in MATLAB

MATLAB constitutes a family of application-specific solutions called toolboxes which are comprehensive collection of MATLAB functions (M-files). Toolboxes plays a prominent role to most users of MATLAB, thus it allows you to learn and apply specialized technology.

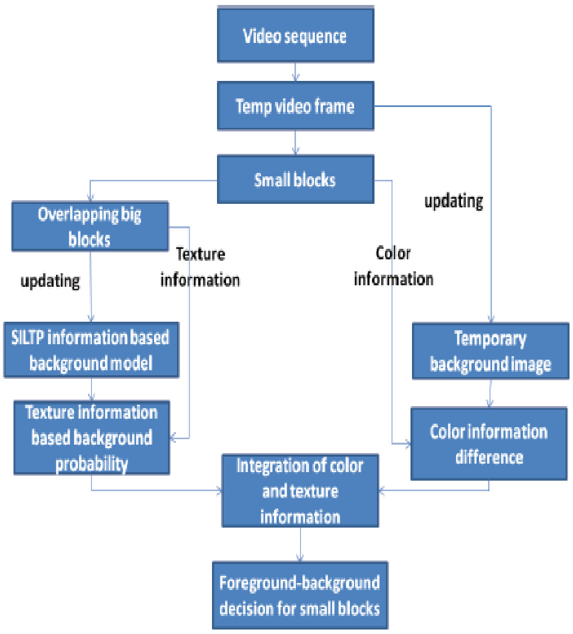
3.2 Image Processing Toolbox

Image Processing Toolbox serves a set of reference-standard for image processing, visualization, analysis and algorithm enrichment. We can perform image segmentation, image analysis, enhancement, image registration, noise reduction and geometric transformations and it lets us explore videos, adjust colour and contrast, examine a region of pixels, create contours or histograms and manipulate ROI (Region of interest).

3.3 Programming in MATLAB

MATLAB supports some basic programming structures that allow looping and conditioning commands along with relational and logical operators. The syntax and use of some of these structures are very similar to those found in C, Basic, and FORTRAN.

4. FLOWCHART



5. WORKING

5.1 Coarse Fine Detection Theory

Initially, we take an instance *Ip*(*t*') belongs to either a background instance or a foreground instance. Binary hypothesis-testing problem may arise while identifying the label of *Ip*(*t*') . Here, we consider two hypotheses. Given an instance *Ip*(*t*'), the hypothesis *H*0 is defined as *Ip*(*t*') is a background instance. The hypothesis *H*1 is defined as *Ip*(*t*') is a foreground instance. These hypotheses of the coarse level can be represented as follows:

H_0 : Ip(t') is a background instance
H_1 : Ip(t) is a foreground instance

In the coarse level, Bayesian maximum a posterior (MAP) is used to decide the label of each instance *Ip*(*t*'). The instance *Ip*(*t*') is classified as foreground instance or a background instance by the MAP detection theory as follows:

$$P(H_0 \mid I_p(t')) \underset{H_1}{\overset{H_0}{\gtrless}} P(H_1 \mid I_p(t')).$$

By applying Bayes' rule, the equation can be rewritten as follows:

$$\frac{P(I_p(t')|H_0)P(H_0)}{P(I_p(t'))} \underset{H_1}{\overset{H_0}{\gtrless}} \frac{P(I_p(t')|H_1)P(H_1)}{P(I_p(t'))}.$$

Using the above equation, *Ip*(*t*') decided as a foreground instance or a background instance.

Image region is represented by instances and the pixels in the region are at the same label. As a result, it will retrieve only the rough shapes of foreground object. Further, we can identify whether the pixel is foreground pixel or not by employing the fine level detection theory to decide the label of the pixel. Given a pixel *p*(*t*') in a foreground instance *IpF*(*t*'), we formulate the fine level binary hypothesis-testing problem to identify the label of *p*(*t*'). Similar to the coarse level detection theory, we also consider two hypotheses in the fine level. The hypothesis *G*0 is defined as *p*(*t*') is a background pixel. The hypothesis *G*1 is defined as *p*(*t*') is a foreground pixel.

The hypotheses can be represented as follows:
Then MAP detection theory is applied to decide the label of *p*(*t*') hinge on labels of instances containing *p*(*t*'). The instances containing *p*(*t*') form an union *Up*(*t*'), which is defined as {*Ip*'(*t*') | *p*'(*t*') = *p*(*t*') + (*x*', *y*'), *x*' = [-*q q*], *y*' = [-*q q*]}. Based on results of the detection theory, the label of *Ip*'(*t*') centered at *p*'(*t*') can be identified. The fine level detection theory is defined to identify the label of *p*(*t*') as follows:

$$P(G_0 \mid p(t')) \underset{G_1}{\overset{G_0}{\gtrless}} P(G_1 \mid p(t')).$$

By applying Bayes rule, Equation can be rewritten as follows:

$$\Lambda_G = \frac{P(p(t')|G_0)}{P(p(t')|G_1)} \underset{G_1}{\overset{G_0}{\gtrless}} \frac{P(G_1)}{P(G_0)}.$$

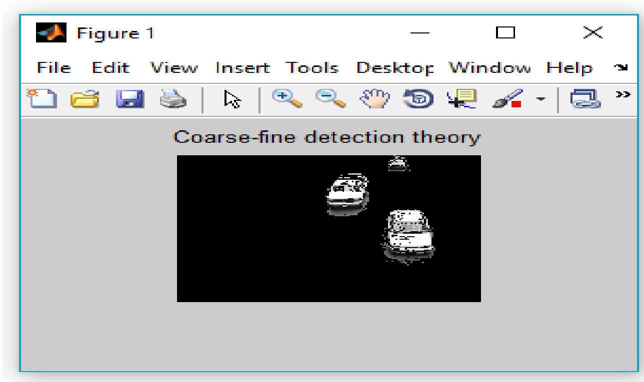
The likelihood functions *P*(*p*(*t*')|*G*0) and *P*(*p*(*t*')|*G*1) of *G*0 and *G*1 are defined as follows:

$$P(p(t')|G_0) = \frac{\sum\{1 | I_p(t') \in background \;\&\; I_p(t') \in U_p(t')\}}{(2q+1)^2};$$

and

$$P(p(t')|G_1) = \frac{\sum\{1 | I_p(t') \in foreground \;\&\; I_p(t') \in U_p(t')\}}{(2q+1)^2}.$$

Output of Coarse and Fine Detection Theory



5.2 SILTP Algorithm

Block wise background model based on SILTP information

We use SILTP representation for background modelling as it withstands illumination changes.

$$SILTP^{\tau}_{N,R}(x_c, y_c) = \bigoplus_{k=0}^{N-1} s_\tau(I_c, I_k)$$

Here, Ic and Ik are gray intensity values of centre pixel and neighbourhood pixels respectively. A circle of Radius R is taken and a piecewise function is specified as below

$$s_\tau(I_c, I_k) = \begin{cases} 01, & \text{if } I_k > (1+\tau)I_c, \\ 10, & \text{if } I_k < (1-\tau)I_c, \\ 00, & \text{otherwise.} \end{cases}$$

We divide our video into frames and then to equally sized blocks. On performing foreground detection decision, coarse boundaries and connection of adjacent moving objects were the outcomes pertaining to larger block size. The overlapped area size is lesser than the original size of the block. So we use a strategy known as 2 level block size.

Next we divide the image into tiny blocks. The 4 tiny blocks join to form a big block, which overlaps partially. Foreground detection decision and background model are done for tiny blocks and large block respectively.

For every big block, a histogram is calculated. We can encode any pixel from its neighbourhood pixels in 3 patterns. For every pixel i, Hs, the histogram of the large block is calculated as,

$$H_s(M_i) = H_s(M_i) + \frac{1}{S_b \cdot S_b}$$

Then we calculate the probability of a block belonging to the background as,

$$P_b^b = \sum_{i=1}^{N_b} H_s(i) T(B_s(i), \frac{\eta}{N_b})$$

Where

$$T(B_s(i), \frac{\eta}{N_b}) = \begin{cases} 1, & \text{if } B_s(i) \geq \frac{\eta}{N_b} \\ 0, & \text{if } B_s(i) < \frac{\eta}{N_b} \end{cases}$$

Siltp has some advantages as it is stable, refined and robust to illumination changes.

Blockwise background model based on color information

We have a disadvantage that it is hard to detect smooth foreground from smooth background. We update a new temporary background image so as to get the color information. On the arrival of a new video Frame, the Color difference is calculated between each of the tiny blocks and then the respective tiny block in the updated temporary background image is calculated.

Then with the differences of the colour channel of entire pixels in every tiny block, averages are computed and they are again combined in total as 3 channels and final difference is then calculated. We do this because it has several advantages as it is more stable and time saving. For example say Ns pixels are present in each tiny block, then the difference can be calculated as

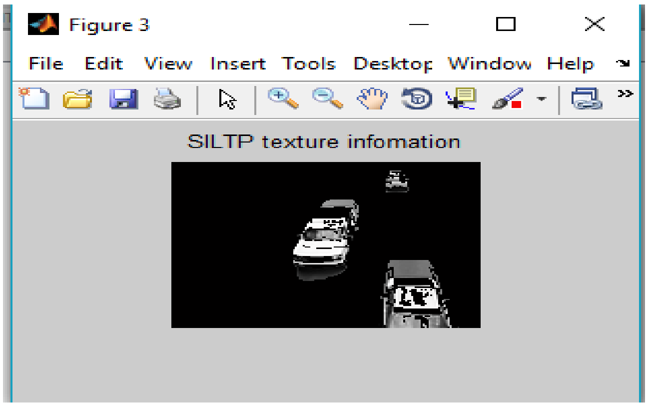
$$D^r = \sum_{i=1}^{N_s} (C_b^r(i) - C_n^r(i))$$

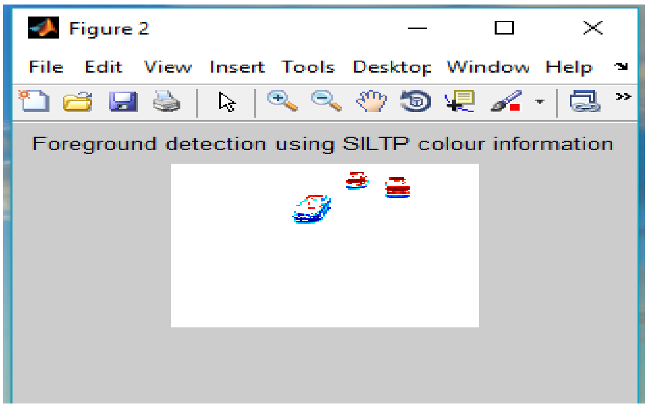
$$D^g = \sum_{i=1}^{N_s} (C_b^g(i) - C_n^g(i))$$

$$D^b = \sum_{i=1}^{N_s} (C_b^b(i) - C_n^b(i))$$

$$D = ((\frac{D^r}{N_s})^2 + (\frac{D^g}{N_s})^2 + (\frac{D^b}{N_s})^2)\frac{1}{255 \cdot 255 \cdot 3}$$

So, with these colour and texture information, foreground-background decision is finally obtained for every small block.





5.3 Active Contour Algorithm and Blob Analysis

Contour-based detection is used for detection and tracking of moving objects. It gives accurate information of moving objects in this algorithm. It requires 0.089s/frames on a 900mhz processor which is used for real time application.

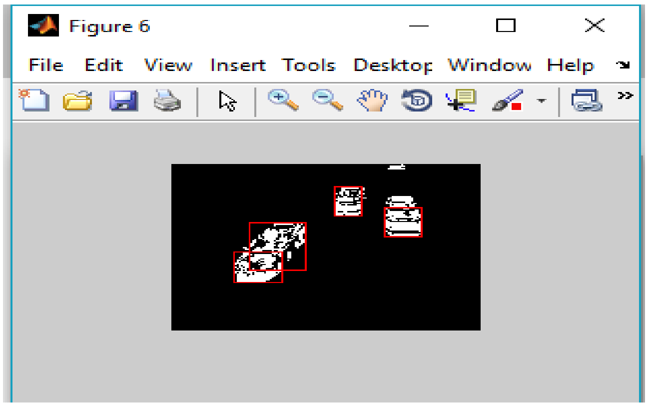
This method can be applied on various sequences such as cars, pedestrians, interfering objects, approaching or occluded cars, etc. This removes noise from the obtained frames and so we can detect a most accurate information from detected contours.

In Blob analysis block, Blob is detected for each vehicle and the area of blob can be used for traffic surveillance. The number of Blobs counted gives the information about total number of cars in the video.

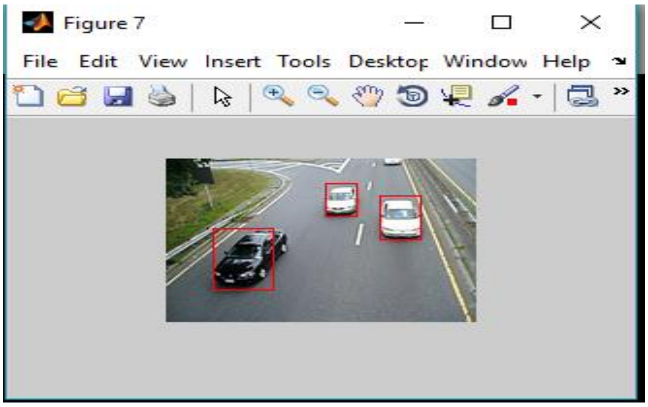
In this process, a rectangle is formed around detected blob i.e., around each vehicle. Between, connected component algorithm is used to eliminate the cracks and gaps in the rectangle because each line detected is counted as one.

Using this algorithm finally gives a regular closed rectangle blob.

Sometimes, the blob processing does not pay good at travel surveillance. Hence, an additional MATLAB function is included to count the number of passing cars.



FINAL OUTPUT



6. CONCLUSION

In this project, using hybrid model we evolve a binary descriptor based background modeling technique to extract foreground objects. Galaxy descriptors are used to represent background and foreground instances. A new quality measure is proposed for evaluating the performance of our method on various challenging videos, and the result is quite understanding compared with earlier methods. The memory consumption and speed can be reduced. This is well suitable for indoor and outdoor techniques, crowded environment, filtering, etc. Further analysis shows that our method is vigorous to illumination variations moving cast shadow problems and dynamic backgrounds.

ACKNOWLEDGEMENT

We would like to acknowledge the contribution of all the people who have helped in reviewing this paper. We would like to give sincere thanks to Miss. C.Senthamizharasi for her guidance and support throughout this paper. We would also like to thank our families and friends who supported us in the course of writing this paper.

REFERENCES

[1] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, “Pfinder: Real-time tracking of the human body,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 780–785, 1997.

[2] C. Stauffer and W. E. L. Grimson, “Adaptive background mixture models for real- time tracking,” *in IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2. IEEE, 1999.

[3] M. Harville, “A framework for high-level feedback to adaptive, per-pixel, mixture-of-gaussian background models,” *in Computer Vision łECCV 2002. Springer, 2002*, pp. 543–560.

[4] D.-S. Lee, J. J. Hull, and B. Erol, “A bayesian framework for gaussian mixture background modeling,” *in Proceedings. 2003 International Conference on Image Processing. ICIP 2003.*, vol. 3. IEEE, 2003, pp. III–973.

[5] A. Elgammal, R. Duraiswami, D. Harwood, and L. S. Davis, “Background and foreground modeling using nonparametric kernel density estimation for visual surveillance,” *Proceedings of the IEEE*, vol. 90, no. 7, pp. 1151–1163, 2002.

[6] L. Li, W. Huang, I. Y. Gu, and Q. Tian, “Foreground object detection from videos containing complex background,” *in Proceedings of the eleventh ACM international conference on Multimedia.* ACM, 2003, pp. 2–10.

[7] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. Davis, “Background modeling and subtraction by codebook construction,” *in International Conference on Image Processing, 2004.*, vol. 5. IEEE, 2004, pp. 3061–3064.

[8] “Real-time foreground-background segmentation using codebook model,” Real-time imaging, vol. 11, no. 3, pp. 172–185, 2005.

[9] S. Liao, G. Zhao, V. Kellokumpu, M. Pietikainen, and S. Z.Li, “Modeling pixel process with scale invariant local patterns for background subtraction in complex scenes,” *in 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE*, 2010, pp. 1301–1306.

[10] J. Yao and J.-M.Odobez, “Multi-layer background subtraction based on color and texture,” *in IEEE Conference on Computer Vision and Pattern Recognition*, 2007. CVPR’07. IEEE, 2007, pp. 1–8.

[11] Z. Zhang, C. Wang, B. Xiao, S. Liu, and W. Zhou, “Multi-scale fusion of texture and color for background modeling,” *in IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance (AVSS)*, 2012. IEEE, 2012, pp. 154–159.

[12] E. G. Learned-Miller, M. Narayana, and A. Hanson, “Background modeling using adaptive pixel wise kernel variances in a hybrid feature space,” *in IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[13] T. Matsuyama, T. Ohya, and H. Habe, “Background subtraction for non-stationary scenes,” *in Proceedings of Asian Conference on Computer Vision*, 2000, pp. 662–667.

