ARPN Journal of Engineering and Applied Sciences

MEASURING OF BACKGROUND MODELING AND SUBTRACTION ALGORITHMS ON MOVING OBJECT DETECTION IN VIDEO SEQUENCES IN CHIANGMAI

Suepphong Chernbumroong Kitti Puritat and Pradorn Sureephong
Knowledge and Innovation Center College of Arts, Media and Technology, Chiang Mai University, Huaykaew Rd, Suthep, Muang, Chiang Mai
E-Mail: kitti@kic.camt.info

ABTRACT

The research analyst the traffic video. For the first step of analysis the traffic data in Thailand, real time segmentation algorithms of moving regions in image sequences is an important step in counting systems including automated video surveillance. Background subtraction of video sequences is mainly regards as a solved problem. In this paper not only helps better understand to which type of videos each method suits best for video surveillance of Thailand but also compared of basic background subtraction methods.

Keywords: image processing, video surveillance, traffic analysis, statistics.

1. INTRODUCTION

A usually applicable assumption is that the images of the scene without the intruding objects exhibit some regular behavior that can be represented by a statistical model. An intruding object can be detected by spotting the parts of the image that don't fit the model. It can called "background subtraction" Background subtraction involves calculating a reference image, subtracting each new frame from current and previous image and thresholding the result. In case gradual illumination changes, the problems lead to the requirement that solution must constantly re-estimate the background model. Many approaches have been proposed to adaptive the background modeling. An appropriate background model has to solve the issue with all the above mentioned issues. In particular, the model has to provide an approximation for a multi-modal probability distribution that can address the problem of modeling an inherently dynamic and fast changing background. Solutions based on a predefined distribution (e.g., Gaussian) for creating the background model can result ineffective, due to the need of modeling non-regular patter.

2. RELATED WORK

A common bottom-up approach is applied and the scene model has a probability density function for each pixel separately. What results is a binary segmentation of the image which highlights regions of non-stationary objects. The simplest form of the reference image is a time-averaged background image. This method suffers from many problems and requires a training period absent of foreground objects. The motion of background objects after the training period and foreground objects motionless during the training period would be considered as permanent foreground objects. In addition, the approach cannot cope with gradual illumination changes in the scene. These problems lead to the requirement that any solution must constantly re-estimate the background model. Many adaptive background-modeling methods have been proposed to deal with these slowly-changing stationary signals. Friedman and Russell modeled each pixel in a camera scene by an adaptive parametric mixture model of three Gaussian distributions [4]. They also provide some brief discussion on the online update equations based on sufficient statistics. Koller et al used a Kalman filter to track the changes in background illumination for every pixel [5]. They applied a selective update scheme to include only the probable background values into the estimate of the background. The methods can cope well with the illumination changes; however, cannot handle the problem of objects being introduced or removed from the scene. One solution is to use a multiple-color background model per pixel. Grimson *et al* employed an adaptive nonparametric Gaussian mixture model to solve these problems [1, 2, 3]. Their model can also lessen the effect of small repetitive motions; for example, moving vegetation like trees and bushes as well as small camera displacement. Elgammal *et al* used a kernel estimator for each pixel [6]. Kernel exemplars were taken from a moving window. They also introduced a method to reduce the result of small motions by employing a spatial coherence. This was done by comparing simply connected components to the background model of its circular neighbourhood. Although the authors presented a number of speed-up routines, the approach was still of high computational complexity. Other techniques using high level processing to assist the background modeling have been proposed; for instance, the Wallflower tracker [7] which circumvents some of these problems using high level processing rather than tackling the inadequacies of the background model. Our method is based on Grimson et al's framework [1, 2, 3], the differences lie in the update equations, initialization method and the introduction of a shadow detection algorithm.

3. BACKGROUND MODELING

3.1 Codebook

The codebook algorithm by Sigari and Fathy [10] is inspired by codebook by Kim *et al*. [8]. But in contrary to simple codebook, which contains an unique codebook per pixel, this method uses 2 codebooks. Each codebook contains some codeword to model a cluster of samples that constructs a part of background and each codeword contains these informations: 1) v_i: value of mean pixel (R, G, B), 2) I_{max}: high intensity bound of codeword, 3) I_{min}: low intensity bound of codeword, 4) f: frequency of codeword, 5) λ: MNRL (maximum negative run length), represents the longest number of image where the codeword doesn't occur in the sequence, 6) p: first occurrence of the codeword, and 7) q: last occurrence of the codeword. The principle is the same than simple codebook, but we have 2 codebook per pixel: a main codebook called M, and an hidden codebook called H. For each new pixel x_t = (R, G, B), its intensity It is calculated by

$$I_t = \sqrt{R^2 + G^2 + B^2} \quad (1)$$

The color distortion δ between this pixel x_t = (R, G, B) and a codeword c_i where v_i = (R, G, B) can be calculated by

$$\langle x_t, v_i\rangle^2 = (R_i, R + G_i, G + B_i, B)^2 \quad (2)$$

$$\|V_i\|^2 = \bar{R}_i^2 + \bar{G}_i^2 + \bar{B}_i^2 \quad (3)$$

$$\|x_t\|^2 = \bar{R}_i^2 + \bar{G}_i^2 + \bar{B}_i^2 \quad (4)$$

$$colordist\ (x_t, v_i) = \delta = \sqrt{\|x_t\|^2 - \frac{\langle x_t, v_i\rangle^2}{\|V_i\|^2}} \quad (5)$$

3.2 Gaussian mixture model

One of the most popular methods based on a parametric probabilistic background model proposed by Stauffer and Grimson [11], and improved by Hayman and Eklundh [6]. In this algorithm, a distribution of each pixel color is represented by a sum of weighted Gaussian distributions defined in a given colorspace: the Gaussian Mixture Model (or GMM). These distributions are generally updated using an online expectation-minimization algorithm. Even if this method is able to handle with low illumination variations, rapid variations of illumination and shadows are still problematic. Furthermore, the learning stage can be inefficient if it is realized with noisy video frames. To tackle these problems, many authors have extended the GMM. For example, Kaewtrakulpong and Bowden [7] propose to modify the updated equations in this model to improve the adaptation of the system to illumination variations.

Each pixel has a parametric distribution model given by a mixture of N Gaussians, 2 ≤ N ≤5 [11], [6]. For n = 1,…,N, an element of the GMM is represented with a mean μ_n, a standard deviation σ_n, and a weight σ_n ($\sum_n \alpha_n = 1$). We can notice that $\sigma_n$is reduced as a scalar, as discussed in [11]. As a new image is processed, the GMM parameters (for all pixels) are updated to explain the colors variations. In fact, at time t, we consider that the model M_t generated for each pixel from the measures $\{Z_0, Z_1, \ldots, Z_{t-1}\}$ is correct. The likelihood that a pixel is a background pixel is:

$$P(Z_t|M_t) = \sum_{n=1}^{n=N} \alpha_n N(\mu_n, \textstyle\sum_n) \quad (6)$$

$$N(Z_t, \textstyle\sum_n) = \frac{1}{2\pi^{d/2}|\sum_n|^{1/2}} e^{-\frac{1}{2}(Z_t,\mu_n)^T \Sigma^{-1}(Z_t - \mu_n)} \quad (7)$$

where d is the dimension of color space of the measures Z_t.

3.3 VU Meter

The VuMeter method proposed by Goyat *et al.* [4] is a non parametric model, based on a discrete estimation of the probability distribution. It is a probabilistic approach to define the image background model. I_t is an image at time t, and $y_t(u)$ gives the color vector Red Green Blue of pixel u. A pixel can take two states, (ω_1) if the pixel is background, (ω_2) if the pixel is foreground. This method tries to estimate $p(\omega_1|y_t(u))$. With 3 color component i (R, G, B), the probability density function can be approximated by:

$$p(\omega_1|y_t(u)) = \prod_{li-1}^{3} p(\omega_1|y_t{}^i(u)) \quad (8)$$

$$\prod_{li-1}^{3} p(\omega_1|y_t{}^i(u)) \approx K_i \sum_{j=1}^{N} \pi_t^{ij} \delta(b_t^i(u) - j) \quad (9)$$

3.4 Hierachical

Chen *et al.* [2] proposed a hierarchical method inspired by Stauffer and Grimson [11]. Here, we will focus only on the bloc-level approach. Using the algorithm of [11], Chen et al. [2] replace the RGB pixel descriptor by a 8×8 bloc texture one called contrast histogram. After dividing an image into blocks, a descriptor is built for each block B_c. Since the center pixel P_c in B_c does not exist, its value is estimated by averaging the four center pixels of B_c. Each block is separated into four quadrant bins, until positive and negative contrast-value histograms for each quadrant bin q_i are computed.

Let $j \in R, G, B$ and $k \in R, G, B$ stand for the color channels of p and P_c, respectively. The positive contrast histogram $CH_{q_i}^{j,k;+}P_c$ and negative $CH_{q_i}^{j,k;-}P_c$ one of q_i with respect to P_c are defined as follows:

$$CH_{q_i}^{j,k;+}P_c = \frac{\sum\{C^{(j,k)}(p,p_c)|p \in q_i \wedge C^{(j,k)}(p,p_c) > 0\}}{\omega_{q_i}^{+}} \quad (10)$$

$$CH_{q_i}^{j,k;-}P_c = \frac{\sum\{C^{(j,k)}(p,p_c)|p \in q_i \wedge C^{(j,k)}(p,p_c) > 0\}}{\omega_{q_i}^{-}} \quad (11)$$

3.5 Bayesian background model

Tuzel *et al.* [12] introduced a method for modeling background using recursive Bayesian learning approach. Each pixel is modeled with layers of Gaussian distributions. Using recursive Bayesian learning, they estimate the probability distribution of the mean and covariance of each Gaussian. Here, we will consider that the update phase is called for each frame, and the system is speed up by making indepence assumption on color channels. To update the layers, the following equations are used $(m_k(Z_t)$.

$$v_n' \leftarrow v_n' + m_k(Z_t) \tag{12}$$

$$k_n' \leftarrow k_n' + (2m_k(Z_t) - 1) \tag{13}$$

$$u_n' \leftarrow (1 - \frac{mk\,(Z_t)}{k_n' + mk\,(Z_t)}) u_n' + \frac{mk\,(Z_t)}{k_n' + mk\,(Z_t)} Z_t \tag{14}$$

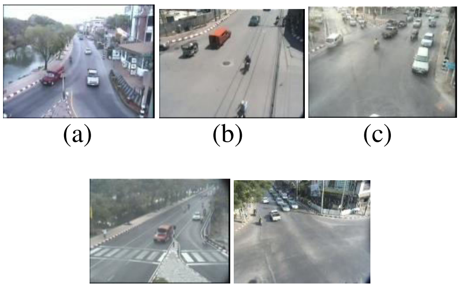
$$\theta_n' \leftarrow \theta_n' + \frac{k_n'}{k_n' + mk\,(Z_t)} (Z_t - u_n')^T (Z_t - u_n') \tag{15}$$

$$\textstyle\sum_n' \leftarrow (v_n' - 4)^{(-1)}\, \theta_n' \tag{16}$$

4. EXPERIMENTAL

4.1 Evaluation dataset

To measure background modeling algorithm and to compare, we implemented the algorithm in OpenCV, we used five video from ChiangMai Municipality containing all traffic around ChiangMai province category “Dynamic Background”. The results obtained by using two selectively chosen ground truth images for each sequence with ourselves.



(a) (b) (c)

(d) (e)

Figure-2. Videos of Chiangmai.

4.2 Shadow removal

For each algorithm, we applied the $C_1C_2C_3$ invariant color model for shadow removal. The $C_1C_2C_3$ invariant color models are proposed by Gevers *et al.* [13] in 1999, which is defined as follows:

$$c_1 = \arctan\frac{R}{\max(G,B)}$$
$$c_2 = \arctan\frac{R}{\max(R,B)}$$
$$c_3 = \arctan\frac{R}{\max(R,G)} \tag{17}$$

Where R, G and B representing the red, green, and blue color components of a pixel in the image. The pixel becomes a candidate shadow if its intensity is smaller than that of the reference pixel for all three channels. For each pixel in the coin image, the pixel (x , y) can be considered, as a shadow pixel when it meets the condition in equation follow by

$$({c_1}^{B(x,y)} - {c_1}^{I(x,y)})({c_2}^{B(x,y)} - {c_2}^{I(x,y)})({c_3}^{B(x,y)} - {c_3}^{I(x,y)}) < \mathrm{T} \tag{18}$$

Where $\boldsymbol{c_1}^{\boldsymbol{I(x,y)}}$ is the value of $\boldsymbol{c_1}$ at the pixel (x, y) in the background reference which is values given location by use $\boldsymbol{c_1}^{\boldsymbol{B(x,y)}}$ is the value of $\boldsymbol{c_1}$ at the pixel(x, y) in the current image, $\boldsymbol{c_2}^{\boldsymbol{B(x,y)}}$, $\boldsymbol{c_2}^{\boldsymbol{I(x,y)}}$, $\boldsymbol{c_3}^{\boldsymbol{B(x,y)}}$ and $\boldsymbol{c_3}^{\boldsymbol{I(x,y)}}$ are similarly defined for $\boldsymbol{c_1}$component.T is a Threshold value.

4.4 Evaluation measure

There are many different ways of evaluating the performance of algorithms, starting from analyzing individual pixels at the lowest level, to higher levels which consider the overall effectiveness of the application that the thresholding is embedded within. Our initial approach is to measure the correctness of the algorithms at the pixel level which is independent of a specific application. At a goal directed level we continued by evaluating the effectiveness of the results for change detection. The results of the low level pixel based comparison between the ground truth and the thresholded image for each frame of the sequence were based on the following values:

$$PR = \frac{TP}{TP + FP} \tag{19}$$

$$RE = \frac{TP}{TP + FN} \tag{20}$$

$$SP = \frac{TN}{TN + FP} \tag{21}$$

where four quantities the following measures were used:

- **True positives (TP):** number of change pixels correctly detected.
- **False positives (FP):** number of no-change pixels incorrectly flagged as change by the algorithm.
- **True negatives (TN):** number of no-change pixels correct detected.
- **False negatives (FN):** number of change pixels incorrectly flagged as no-change by the algorithm.

We consider the segmentation of images divided into two classes: foreground and background. For a given image in a video sequence, we compare the results of a binary segmentation S with the binary image of the ground truth T. A pixel is represented in white if it is part of a moving object (foreground), and black when it belongs to the background. A white pixel in S is called a positive. If it is also white in T, then it is a true positive (TP), whereas if it is black in T, it is a false positive (FP). Symmetrically, a

ARPN Journal of Engineering and Applied Sciences

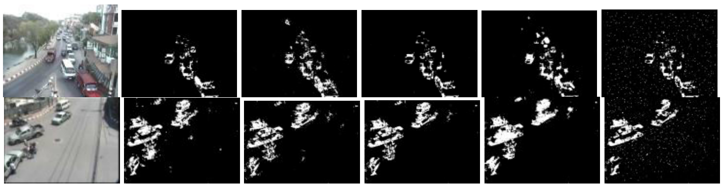
black pixel in S is a negative. If it is also black in T, it is a true negative (TN), while if it is white in T, it is a false negative (FN). We can then define the Precision (PR), Recall (RE) and Specificity (SP) for each image A perfect segmentation algorithm calculates an image S identical to the ground truth T. Such an algorithm will give values of Precision, Recall and Specificity.

In order to improve segmentation, we use quality measure to find good values for input parameters of image segmentation algorithms. The results of any segmentation algorithms vary as a function of the values of different parameters. D_{prs} (K. Intawong, 2013) which to measure the quality of segmentation as an Euclidean distance called D_{prs} in the space of the indicators, between the point (PR, RE, SP).

$$Dprs = \sqrt{(1-PR)^2 + (1-RE)^2 + (1-SP)^2} \qquad (22)$$

4.5 Result

Object detection by background modeling algorithms, used without postprocessing, very often let appear isolated pixels in the background. They are considered as foreground objects. In our experiments, we compare and analyse the best possible values for each of the segmentation quality measures. We calculate the following measures: recall, precision, specificity and D_{prs} for all images in a sequence on 5 videos composed of 1500 frames each around Chiangmai city. The best algorithms of D_{prs} = 1.09 is Codebook but the best result is still not good enough due to poor video for example color of car mirror and road is the same color. Another reason is crowd traffic that background modeling cannot well extract for each object.



Original a b c d e

Figure-3. Segementation results: (a) Gaussian Mixture Model (b) Bayesian (c) Hierachical (d) Codebook (e) VuMeter.

Table-2. Comparison average values of segementation methods in 5 videos.

Method	Recall	Precision	Specificity	D_{prs}
Gaussian Mixture Model	0.90	0.81	0.85	0.158
Bayesian	0.86	0.79	0.64	0.313
Hierachical	0.74	0.64	0.73	0.462
Codebook	0.93	0.85	0.87	**0.109**
VuMeter	0.72	0.68	0.52	0.612

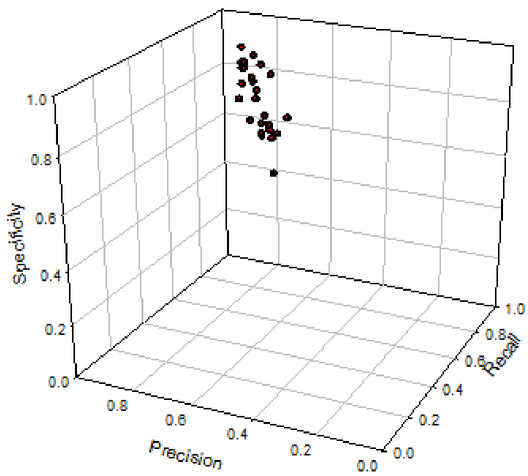


Figure-5. Graph 3D: Recall, precision and specificity.

CONCLUSIONS

This paper presents a comparison of segmentation methods in video analysis system for video surveillance of ChiangMai Thailand which has one of most cars on road in the world. Experimental results demonstrate that codebook algorithm was the best suitable of algorithm for vehicle counting systems for complexes environments of video for Thailand which has many object and poor video during peak time. The whole method of background modeling has been quantitatively compared with other BS methods implemented in the OpenCV 2.45 library with c++.

REFERENCES

[1] N. Buch, S.A. Velastin, J. Orwell. 2011. A review of computer vision techniques for the analysis of urban

traffic. IEEE Transact. Intell. Transp. Syst. 12(3): 920-939.

[2] B. Tian, Q. Yao, Y. Gu, K. Wang, Y. Li. 2011. Video processing techniques for traffic flow monitoring: a survey. In: Proceedings of International Conference on Intelligent Transportation Systems. pp. 1103-1108.

[3] S. Karaman, J. Benois-Pineau, V. Dovgalecs, R. Mégret, J. Pionquier, R. André-Obrecht, Y. Gaestel and J.F. Dartigues. 2011. Hierarchical Hidden Markov Model in Detecting Activities of Daily Living in Wearable Videos for studies of Dementia. Arxiv preprint ⟨arXiv:1111.1817⟩.

[4] H. Trinh, Q. Fan, P. Jiyan, P. Gabbur, S. Miyawaza, S. Pankanti. 989. Detecting human activities in retail surveillance using hierarchical finite state machine. in: Proceedings of ICASSP, IEEE, 2011, pp. 1337-1340. Writer's Handbook. Mill Valley, CA: University Science.

[5] Chen Y.T., Chen C.S., Huang C.R., Hung Y.P. 2007. Efficient hierarchical method for background subtraction. Pattern Recognition. pp. 2706-2715.

[6] Dhome Y., Tronson N., Vacavant A. 2010. A Benchmark for Background Subtraction Algorithms in Monocular Vision: a Comparative Study. Image Processing Theory Tools and Applications (IPTA).

[7] MIT. Traffic dataset. URL: http://www.ee.cuhk.edu.hk/xgwang/MITtraffic.html, 2009.

[8] R. Zhao and X. Wang. 2013. Counting vehicles from semantic regions. In Intelligent Transportation Systems. pp. 1016-1022.

[9] J.Davis and M. Goadrich. 2006. The relationship between precision-recall and roc curves. In International conference on Machine learning. pp. 233-240.

[10] K. Intawong, M. Scuturici, S. Miguet. 2013. A New Pixel-Based Quality Measure for Segmentation Algorithms Integrating Precision, Recall and Specificity, International, Conference on Computer Analysis of Images and Patterns. pp. 188-195.

[11] CRS, Computer Recognition Systems. [Online]. Available: http://www.crs-traffic.co.uk/.

[12] Gevers Theo and Arnold WM Smeulders. 1999. Color-based object recognition. Pattern recognition. 32.3: 453-464.