Non-liner Learning for Mixture of Gaussians

Chih-Yang Lin[1*], Pin-Hsian Liu[2], Tatenda Muindisi[1], Chia-Hung Yeh[2], and Po-Chyi Su[3]
[1] Dept. of Computer Science & Information Engineering, Asia University, Taichung, Taiwan, R.O.C.
E-mail: andrewlin@asia.edu.tw Tel: +886-4-23323456
[2] Department of Electrical Engineering, National Sun Yat-sen University, Kaohsiung, Taiwan, R.O.C.
E-mail: M003010144@student.nsysu.edu.tw Tel: +886-7-5252000
E-mail: yeh@mail.ee.nsysu.edu.tw Tel: +886-7-5252000
[3]Department of Computer Science & Information Engineering, National Central University, Taiwan, R.O.C.
Email: pochyisu@csie.ncu.edu.tw Tel:+886-3-422-7151

Abstract— **Background modeling plays a key role of event detection in intelligent surveillance systems. Gaussian Mixture Model (GMM) is the wide-used background modeling method in latest surveillance systems. However, the model has some disadvantageous when the object moves slowly. In this paper, we propose a mechanism which takes the advantage of Gaussian error function (ERF) to adjust the growths of each Gaussian's weights and variances, to solve the problem that traditional GMM misjudged the slow moving object as background. The mechanism improves the GMM model to detect the slow moving object accurately and enhance the robustness of surveillance systems.**

I. Introduction

In concern with environment security, most people have accepted surveillance systems installed near their surroundings. One of the critical tasks of a surveillance system is to detection moving objects. The most effective way to detect moving objects under a fixed camera is to build up a background model.

Several methods have been proposed to construct background models. For example, Wixson al. [1] proposed a statistical background model based on color information. Moreover, Wren et al. [2] proposed a single Gaussian distributed background. Nevertheless, single Gaussian background has defects for periodically changed backgrounds. Therefore, Stauffer and Grimson improved the single Gaussian model and then proposed the Gaussian mixture model (GMM) method [3]. The GMM method not only used multiple Gaussian distributions to model background [3][4][5], but it also proposed a mechanism to decide the number of Gaussians adaptively.

Although GMM has made a prominent progress on background modeling, its model still encounters several drawbacks. For example, when GMM detects slowly moving objects, these objects are often fragmented since some parts of the moving object will be regarded as background. Another example is when the moving object stops, this object will become background immediately. These unreasonable situations are mainly caused by the linear learning behavior in the traditional GMM.

In this paper, we propose a method which applies a non-linear learning function, called the Gaussian error function (ERF) for GMM. The non-linear learning function is used for both of the variance and the weight of each Gaussian. The results show that the proposed method can successfully solve the problem that traditional GMM may fail.

The following parts of this paper are organized as follows: First of all, take a brief review on the Gaussian mixture model in Section 2. Next, the proposed method is presented in Section 3. Then the experimental results are discussed in Section 4. Finally, conclusions are presented in Section 5.

II. Previous Work

In this section, the theory of a Gaussian mixture model, which was proposed by Stauffer and Grimson [3][4], will be explained. The background model is a pixel-based method and with K-Gaussian distributions, where K is usually set between 3 and 5. Each distribution has a weight to present partial data of the model.

As previously mentioned, each pixel has a mixture of K Gaussian distributions for the model, and the probability of the observed current pixel is presented by

$$P(X_t) = \sum_{j=1}^{K} \omega_{j,t} * \eta(X_t, \mu_{j,t}, \textstyle\sum_{j,t}), \qquad (1)$$

where X_t is the pixel value at time t, K is the number of Gaussian distributions, $\omega_{j,t}$ is a weight estimation of the jth distribution at time t, $\mu_{j,t}$ and $\sum_{j,t}$ are the mean value and covariance matrix, respectively, of the jth Gaussian in the mixture at time t. And η is a Gaussian probability density function (pdf).

After the model has been initialized, the incoming pixel in the following frames will be compared with the model. The input pixel will be classified as matching to one of the weighted Gaussian distributions if the difference between pixel value and the mean of the distribution is less than the 2.5 times of standard deviations. Once a matched pixel is found, the model will trigger an adaption process to update the relevant model; otherwise, the distribution with the lowest weight will be replaced with a new distribution using the incoming pixel as the mean value. In addition, the new distribution will be initialized with a high variance, and a low priority weight.

To choose the most suitable combination of distributions for each pixel, the K distributions are sorted based on the values ω/σ in a decreasing order. Then the first B distributions are selected to represent as background by following

$$B = \arg\min_b \left(\sum_{k=1}^{b} \omega_k > T_B\right), \tag{2}$$

where T_B is a predefined threshold and usually set to 90%, ω_k is the weight parameter of the kth distribution and b is the number of distributions selected for the background.

In the end of the update process, the weights of K Gaussian distributions are changed by

$$\omega_{k,t} = (1-\alpha)\omega_{k,t-1} + \alpha(M_{k,t}), \tag{3}$$

where α is the learning rate and $M_{k,t}$ is set to be 1 for the matched distribution and 0 for the unmatched ones. After the learning procedure, weights of distributions need to be renormalized. If the new pixel matches a Gaussian distribution, the mean μ_t and variance σ^2 of the distribution are updated as follows

$$\mu_t = (1-\rho)\mu_{t-1} + \rho X_t, \tag{4}$$

$$\sigma_t^2 = (1-\rho)\sigma_{t-1}^2 + \rho (X_t - \mu_t)^T (X_t - \mu_t), \tag{5}$$

where

$$\rho = \alpha \eta (X_t \mid \mu_k, \sigma_k). \tag{6}$$

III. Proposed Method

In the section, how the proposed method utilizes the Gaussian error function [6, 7] will be described.

A. Gaussian Error Function

The error function (ERF) is related to the cumulative distribution, the integral of the normal distribution which is a special form of the Gaussian function. The function is defined in (7) and Fig. 1 shows the curve of ERF.

$$ERF(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt \tag{7}$$

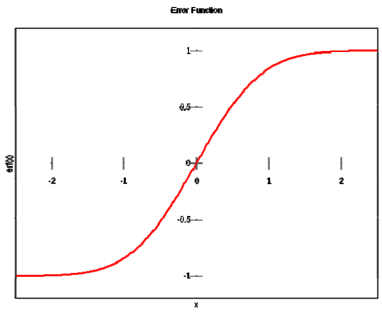

Fig. 1 The sigmoid shape of the ERF function.

B. Conventional Weight Updating

Fig. 2 is an experiment that we put a static object into the background for a period, and then remove the static object. In this experiment we use 3 Gaussians in the mixture model. Fig. 1 shows that the object is put at 53th frame, where the purple vertical line called "ObjIn" stands for, the object becomes background at 62th frame by the dotted line labeled "become_bg", and the object is removed at 467th frame, represented by the blue vertical line labeled "ObjOut".

In Fig. 2, the "First" curve presents the weight inclination of the Gaussian accounted for the original background; the "Second" curve accounts for the Gaussian of the static object; and the "Third" curve is never used in this case since two Gaussians are enough for this simple scenario as shown in Fig. 3, where the background is pure sky blue and the object is black with white characters. In this experiment, the learning rate α of (3) is set to 0.005, which is small enough to slow down the updating process. However, as Fig. 2 shows, weight variations in the original GMM are still so fast that transform the foreground object into background immediately (i.e. just passing 9 frames). This is not reasonable. Another example is slowly moving objects, which result in fragmented because of the same reason, linear learning and fast variations.

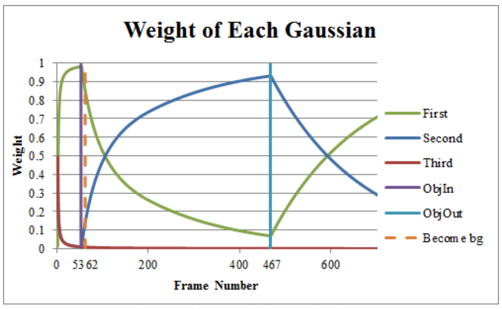

Fig. 2 The weight variations.

Fig. 3 The testing case.

Due to the drawbacks of the conventional GMM using linear learning, the non-linear learning of the ERF function is an effective way to improve these problems.

C. Observation of Variance Updating

Since the GMM uses the weight and variance to maintain the model, not only the weight but also the variance of each Gaussian can use ERF to obtain the more robust model and get better detection results. The following describes the variance variations and the variance and weight will be involved in ERF discussed in Section III.D.

(a)

(b)

(c)

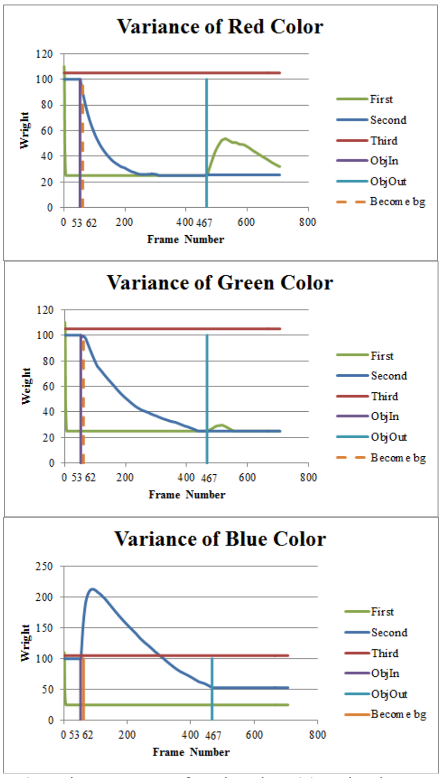

Fig. 4 Variance curve of each color: (a)Red color, (b)Green color, (c)Blue color.

Fig. 4 shows the variance variations of each channel under the same scenario in Fig. 3. In Fig. 4, the "First" curve presents the variance variations of the Gaussian for the original background; the "Second" curve represents the variance variations of the Gaussian for the static object; and the "Third" curve is never used and never updated. In this experiment, the "First" curve is almost unchanged since this existed Gaussian is never replaced with a new distribution and the variance will be updated only when the incoming pixel matches to the distribution. The "Second" curve shows that when the static object appears, the corresponding variance becomes smaller and smaller, and finally stable. Although the variance variation is independent of the static object, it is still an important factor for ERF to control the growth of the weight described in the following subsection.

D. New Weight Updating

Here the new weight updating process is introduced by applying ERF. First, the weight $w_{j,t}$ in the original GMM will be normalized to a new range of [-10, 10] because the new range help us get the better result of ERF. The normalization function is defined in (8).

$$n_{j,t} = (\omega_{j,t} \times 20) - 10\,. \tag{8}$$

Another parameter we used in ERF is V in (9), the variance factor combining variances of V_r, V_g, and V_b for three channels. In (9), the threshold TH is used to control the variation speed of ERF.

$$V = \frac{V_r + V_g + V_b}{TH} \tag{9}$$

$$\omega_{j,t}' = \frac{(ERF(\frac{n_{j,t}}{V}) + 1)}{2} \tag{10}$$

The new weight based on ERF is shown in (10), which has involved (8) and (9). Since the ERF value is between [-1, 1], Eq. (10) is designed to restrict the value within [0, 1]. Finally, these new weights for a mixture of Gaussians are renormalized by (11) to have a sum equal to 1.

$$\omega_{k,t}' = \frac{\omega_{k,t}'}{\sum_{k=1}^{b} \omega_{k,t}} \tag{11}$$

Fig.5 shows the results of the sigmoid curves for weight variations using ERF. These curves show that the slow moving objects or suddenly stopping objects will not become background immediately (i.e., requiring up to 187−53=134 frames) and solve the conventional problems in GMM.

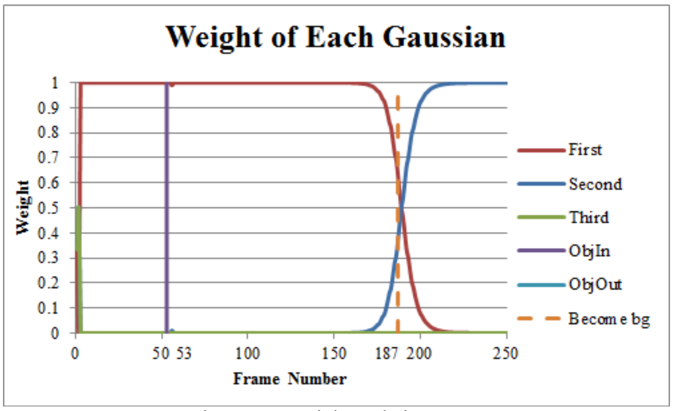

Fig. 5 New weight updating curve

IV. EXPERIMENTAL RESULTS

The experiments adopt three cases. The first case is a slow moving object moving towards the camera. The second case is that a static object is added on the table. The final case is that the static object on the table is removed.

The proposed scheme is compared with the original GMM and the results are shown in Figs. 6 to 8. Fig. 6 shows that the GMM method produces many cavities in the slow moving object, but the proposed method can improve it a lot. Fig. 7 shows the results of adding a static object. If ERF is not applied, the static object will immediately become background within 68−52=16 frames (about 0.5 sec); however, the proposed scheme can validate detection up to 218 −52=166 frames (about 5.5 sec), which is more reasonable for surveillance applications. Similar to Fig. 7, Fig. 8 shows that when the static object has been removed, the proposed scheme can detect this event longer and thus enhance the robustness of the moving object detection.

V. CONCLUSIONS

In this paper, we proposed a new weight updating mechanism for conventional GMM based on ERF. The proposed method has two main advantages: 1) resistance to cavity producing when objects move slowly or move towards cameras; 2) detecting a moving object, either stopped or removed, in a more reasonable period. The experimental results show that the proposed method improves the traditional GMM when ERF is applied. Our future work will focus on how to apply the improved GMM to other surveillance applications.

REFERENCES

[1] Wixson, L., "Detecting Salient Motion by Accumulating Directionally-Consistent Flow," *IEEE Transactions on Pattern Analysis and Machine Intelligence* vol. 22 No.8 pp. 774-780, 2000.

[2] Wren, C.R., Azarbayejani A., Darrell T., Pentland A.P., "Pfinder: Real-Time Tracking of the Human Body," *IEEE Transactions on Pattern Analysis and Machine Intelligence* vol.19 No.7 pp.780-785, 1997.

[3] Stauffer C., Grimson W.E.L., "Adaptive Background Mixture Models for Real-Time Tracking," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* pp. 246-252, 1999.

[4] Stauffer C., Grimson W.E.L., "Learning Patterns of Activity Using Real-Time Tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence* vol.22 No.8 pp.747-757, 2000.

[5] Z., Zivkovic., "Improved Adaptive Gaussian Mixture Model for Background Subtraction." *Proceedings of the 17th International Conference on Pattern Recognition* pp. 28-31, 2004.

[6] Greene, William H., "Econometric Analysis (Fifth Edition)," *Prentice-Hall* pp. 926.

[7] Van Zeghbroeck, Bart., "Principles of Semiconductor Devices," *University of Colorado*, 2011.

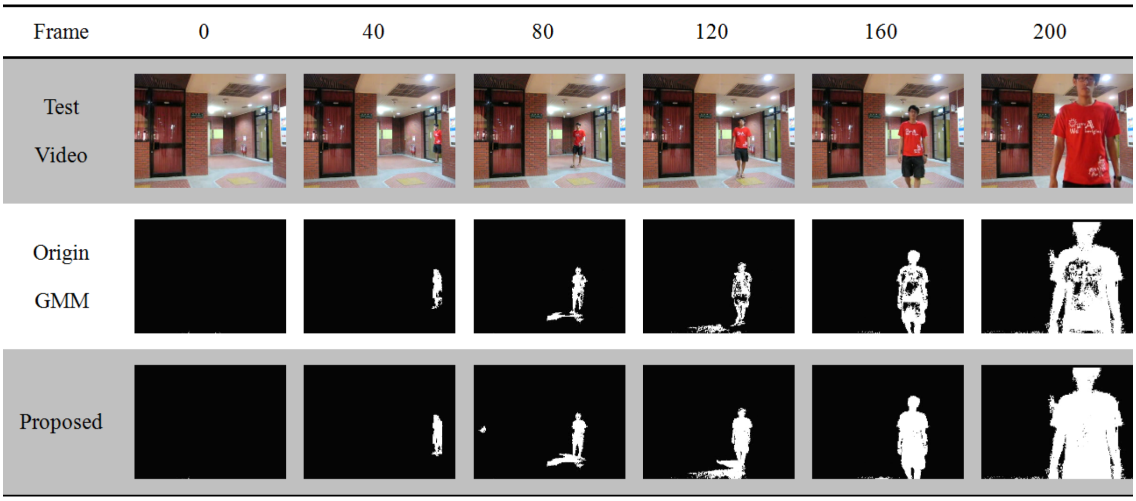

Fig. 6 Experimental results of the slow moving object

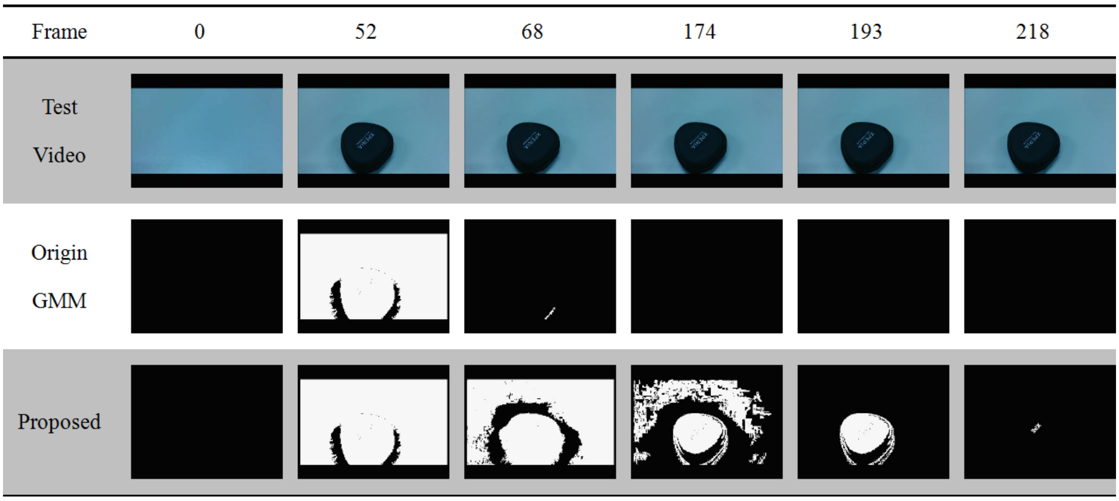

Fig. 7 Experimental results of adding a static object

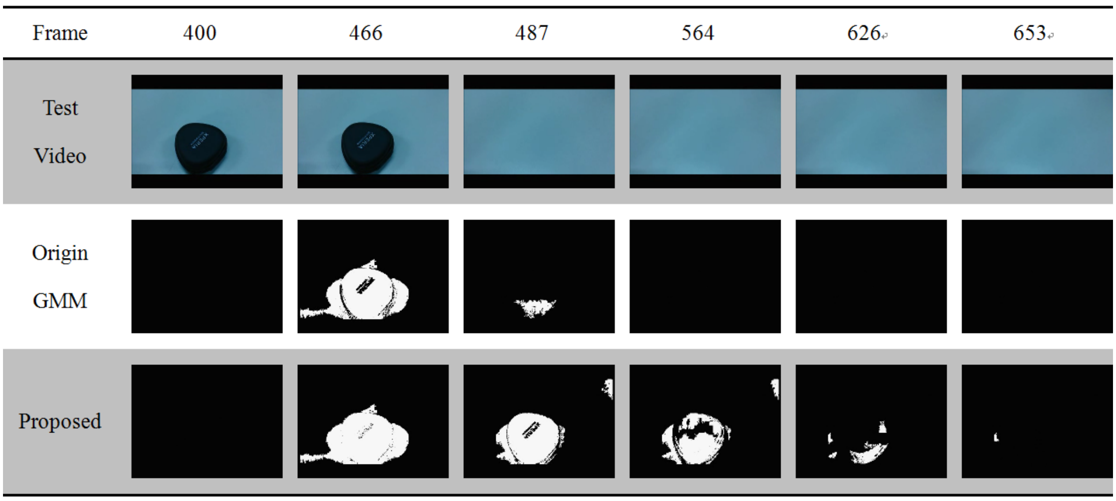

Fig. 8 Experimental results of removing the static object