Change Detection using GMMs

***Abstract*—Change detection is one of the most commonly used methods for detecting moving objects in a computer vision system. It detects moving objects in an image sequence by comparing the background model with the current frame. Traditional moving target detection algorithms have problems such as dynamic background changes caused by illumination changes, noise and shadow sensitivity. In this report, a method based on Gaussian Mixture Model (GMM) is introduced for the challenge of object detection, which modeling each pixel of each frame as a mixture of Gaussians and using an on-line approximation to update the model. Based on whether the Gaussian distribution that represents it most effectively is considered part of the background model, each pixel is classified. This report is targeted to utilizes the python to implement the GMM algorithm to segment background for several videos.**

Keywords—Change detection, GMM, Python

I. Introduction

In these few years, computational abbilities of computers has hugly improved the complexity of real-time video processing application and it also improves the artificial neural network, which are commonly used in background subtracting and moving objects dectation. However, training a artificial nerual network with high performance, whatever in lighting variation scene , in multiple moving objects scene, etc, requires huge training sets and data, which not only can yield data collecction problems but also can increase the cost of development in practice. [1] proposed an approach to subtract the background, which is to describe the histroy of the last n pixel values by a single Gaussian probability distribution. However, modeling by a single Gaussian is sensitive to fast pixel variations. Practially, a single Gaussian can not memorize the old states of the pixel. This is the reason requiring migration to a robuts and multimodal approach. To adress above problems, Stauffer and Grimson proposed a method that using Gaussian mixture model, rather using a single Gaussian distribution[1]. This is a robust and adaptive tracking system that is flexible enough to handle variations in lighting, moving scene clutter, multiple moving objects and other arbitrary changes to the observed scene. Besides that, it is also a lower computational complexity method comparing CNN, deep neural network and thus it definitely will improve the real-time segmentation performance. Work presented in[2] have demonstrated that GMM provides a good compromise between quality and execution time compared to other methods. GMM is a statistical model that desribes pixel variations by several Gaussian distributions[1]. This report firstly introudce related researches regarding background subtraction in these few years. In this part, several methods are introduced and analysed. The second part is mainly to analyse the concrete methodology of Gaussian Mixture Model in segmenting background and foreground. Practically only using a single adaptive Gaussian model is not enough if there are many conditions change[1]. Thus, this section introduces a Gaussian mixture model and its principle based on Stauffer and Grimson [1]. And the next part is to demonstrate the acutal performance of the Gaussian mixture model with some visualized figures. This part is based on the algorithms in part two and implemented by using python. The final part is to analyse the results form the third part and to propose some ideas to improve the Gaussian mixture model. Finally, this report will do the conclusions for this project.

II. Relative researches

Recently various approaches, systems and methods have been proposed and developed to inspect dynamic regions and static regions. One of the most intuitive approaches is to compute the absolute difference either between two successive frames[3], or between a reference image I_R , without any moving object and the current image. In order to determine the objects in motion, a binary mask is applied according to a predefined threshold on the pixels of the resulting image[4].

Another approach to subtract the background is to describe the history of the last n pixel values by a Gaussian probability distribution[5]. However, modeling by a single Gaussian is sensitive to fast pixel variations. Indeed, a single Gaussian cannot memorize the old states of the pixel. This requires migration

Authors in[6] have proposed the first model which describes the variance of the recent values of each pixel by a mixture of the Gaussians. In this model, the Expectation Maximization (EM) algorithm is used to initialize and estimate the parameters of each Gaussian. In [7] authors have estimated the probability density function of the recent N values of each pixel by a kernel estimator (KDE). Furthermore, authors in[8] have provided a nonparametric estimation of the background pattern. They used the concept of a visual dictionary words to model the pixels of the background. Indeed, each pixel of the image is represented by a set of three values (visual word) which describes its current state. These values are initially estimated during the learning phase and are updated regularly over time to build a robust modeling.

Indeed, each pixel of the image is represented by a set of three values (visual word) which describes its current state. These values are initially estimated during the learning phase and are updated regularly over time to build a robust modeling.

In addition, several works have taken spatial information into consideration. Oliver et al. have proposed a sub-spatial learning based on PCA (SL-PCA) [9]. The aim is to make a learning of the N background images using PCA. Moving objects are identified according to the input image and the reconstructed image from its projection in the reduced dimension space. While, Tsai and Lai have provided in [10] a quick schema (SL-ICA) for background subtraction with Independent Component Analysis (ICA). Also, Bucak and Gunsel [11] have presented a decomposition of video content by an incremental non-negative matrix factorization (NMF).

Other methods in [12], [13], [14], [15] and [16] have focused on the selection and combination of good characteristics (color, texture, outlines) to improve the result quality.

Recently, some research works have introduced the fuzzy concept to develop more efficient and robust methods for modeling the background, such as [17], [18], [19], [20], [21] and [22].

Moreover, work presented in [2], has shown that GMM offers a good compromise between quality and execution time compared to other methods. The first GMM model was proposed by[6]. However, Stauffer and Grimson[1] have presented a standard GMM with efficient update equations. Several works and contributions have been proposed to improve the quality of GMM. Among these methods, some of them focus on improving the model adaptation speed, such as[23] and [24]. While other studies have been interested in hybrid models, such as GMM and K-means[25], GMM and fuzzy logic[21], GMM and adaptive background [26], Markov Random Fields[27], GMM and Block matching[28], GMM with PSO[41] to overcome GMM problems. There are also several works that have invested in the characteristics type[29], [30] or in the acquisition material[31]. In addition to spatio-temporal methods[32], some researchers have used local contextual information around a pixel, such as the region[33], [34], the block [35] and the cluster[36], [37]. In the last years, there have been several methods that used deep learning for subtracting the background, among them : FgSegNet S (FPM) [38], Cascade CNN [39], DeepBS[40], Deep background subtraction with scene-specific convolutional neural networks[41]. However, deep leaning methods require a large number of simples and needs more time for training.

III. METHODOLOGY

In this section, concrete algorithm of Gaussian Mixture Models is mathematically introduced. According to Stauffer and Grimson [1], it is sufficient to model the pixel value using a single, adaptive Gaussian distribution if only lighting changed over time. Practically as there are many conditions change, multiple and adaptive Gaussian are demanded to model this process. The following descriptions are mostly original from [1] since it is more accurate to describe the hidden mathematical principles, but the whole algorithm graph is shown .

A. *Online mixture model*

Considered that at any time, t, what can be known about a particular pixel, $\{x_0, y_0\}$, is its history

$$\{X_1, \dots, X_t\} = \{I(x_0, y_0, i) : 1 \leq i \leq t\}$$

where I is the image sequence. The value of each pixel demonstrates a measurement of the radiance in the direction of the sensor of the first object intersected by the optical ray of pixel.

That value would be relatively constant if background and lighting are static. It is required for the Gaussian model to track those changes, such as the illumination changes in a stationary scene. If a fixed object was added to the scene and was not recognized as the background until it had been there longer than the previous object, then the corresponding pixels are deemed as foreground for arbitrarily long time. This would result in accumulated errors in the foreground estimation, finally leading to poor tracking behavior. Therefore, more recent measurements may be more significant in deciding the Gaussian parameter estimates. An additional aspect of variation occurs if moving objects are present in the scene. Even a relatively consistently colored moving object is commonly expected to produce more variance than a fixed object. Also, generally, since they are repeated, whereas pixel values for different objects are often not the same color, there should be more data supporting the background distributions. There are guiding considerations when choosing model and updating process. The recently historical value of each pixel, $\{X_1, \dots, X_t\}$, is modeled by a mixture of K Gaussian distribution. The probability of observing the current pixel value is

$$P(X_t) = \sum_{i=1}^{K} w_{i,t} * \eta(X_t, \mu_{i,t}, \Sigma_{i,t})$$

where K is the number of distributions. $w_{i,t}$ is an estimate of the weight (what portion of the data is accounted for by this Gaussian) of the i^{th} Gaussian in the mixture at time t, $\mu_{i,t}$ is the mean value of the i^{th} Gaussian in the mixture at time t, $\Sigma_{i,t}$ is the covariance matrix of the i^{th} Gaussian in the mixture at time t, and where η is a Gaussian probability density function

$$\eta(X_t, \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(X_t-\mu_t)^T \Sigma^{-1}(X_t-\mu_t)}$$

K is determined by the available memory and computational power. Currently, 3 are used in this report. Also, for computational reasons, the covariance matrix is assumed to be of the form:

$$\Sigma_{k,t} = \sigma_k^2 I$$

This assumes that the R, G, B pixel values are independent and have the same variances. While this is certainly not the case, the assumption is to avoid a costly matrix inversion at the expense of some accuracy. Hence, the distribution of recently observed values of each pixel in the scene is represented by a mixture of Gaussians. A new pixel value will generally be characterized by one of the principal elements of the mixture model and used to update the model. If the pixel process could be considered a stationary process, a standard method for maximizing the likelihood of the observed data is expectation maximization[2]. Unfortunately, each pixel process changes over time because the state of the world changes, hence the approximate approach which fundamentally considers each new observation as a sample set of size 1 and uses standard learning rules to integrate the new data are used.

It is costly to implement an exact EM algorithm on a window of recent data since there is a mixture model for every pixel in the image. Alternatively, we perform an on-line K-means approximation. Every magnitude of new pixel, X_t, is checked against the existing K Gaussian distributions, until a match defined as a pixel value within 2.5 standard deviations of a distribution, is found.

$$|X_t - \mu_{i,t-1}| < 2.5\sigma_{i,t-1}$$

This threshold can be perturbed with effects on performance that introduced later. This is effectively a per pixel/ per distribution threshold. This is highly effective in different place with different illumination, as objects presenting in shaded regions do not commonly generate as much noise as objects in lighted regions.

If none of the K distributions match the current pixel value, the tail probable distribution is replaced with a distribution with the current value as its mean value, an initially high variance, and low prior weight.

The prior weights of the K distributions at time t, $w_{k,t}$, are adjusted as follows

$$w_{k,t} = (1-\alpha)w_{k,t-1} + \alpha(M_{k,t})$$

where α is the learning rate and $M_{k,t}$ is 1 for the model which matched and 0 for the remaining models. After this approximation, the weights are renormalized. $1/\alpha$ defines the time constant which determines the speed at which the parameters of distribution change. $w_{k,t}$ is effectively a causal low-pass filters average of the (thresholded) posterior probability that pixel values have matched model *k* given observations from time 1 through *t*. This is equivalent to the expectation of this value with an exponential window on the past values.

The μ and σ parameters for unmatched distributions remain the same. The parameters of the distribution which matches the new observation are updated as follows:

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

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

where the second learning rate, ρ, is

$$\rho = \alpha\eta(X_t|\mu_k, \sigma_k)$$

This is effectively the same type of causal low-pass filter as mentioned above, except that only the data which matches the model is included in the estimation.

One of the significant advantages of this method is that when something is allowed to become part of the background, it does not destroy the existing model of the background. The original background color remains in the mixture until it becomes the K^{th} most probable and a new color is observed. Therefore, if an object is stationary just long enough to become part of the background and then it moves, the distribution describing the previous background still exists with the same μ and σ^2, but a lower w and will be quickly re-incorporated into the background.

B. Background model estimation

As the parameters of the mixture model of each pixel change, we would like to determine which of the Gaussians of the mixture are most likely produced by background processes. Heuristically, we are interested in the Gaussian distributions which have the most, supporting evidence and the least variance.

To understand this choice, consider the accumulation of supporting evidence and the relatively low variance for the "background" distributions when a static, persistent object is visible. In contrast, when a new object occludes the background object, it will not, in general, match one of the existing distributions which will result in either the creation of a new distribution or the increase in the variance of an existing distribution. Also, the variance of the moving object is expected to remain larger than a background pixel until the moving object stops. To model this, we need a method for deciding what portion of the mixture model best represents background processes.

First, the Gaussian are ordered by the value of w/σ. This value increases both as a distribution gains more evidence and as the variance decreases. After re-estimating the parameters of the mixture, it is sufficient to sort from the matched distribution towards the most probable background distribution, because only the matched models relative value will have changed. This ordering of the model is effectively an ordered, open-ended list, where the most likely background distributions remain on top and the less probable transient background distributions gravitate towards the bottom and are eventually replaced by new distributions.

Then the first B distributions are chosen as background model, where

$$B = \operatorname{argmin}_b\left(\sum_{k=1}^{b} w_k > T\right)$$

where T is a measure of the minimum portion of the data that should be accounted for by the background. This takes the "best" distributions until a certain portion, T, of the recent data has been accounted for. If a small value for T is chosen, the background model is usually unimodal. If this is the case, using only the most probable distribution will save processing.

If T is higher, a multi-modal distribution caused by a repetitive background motion (e.g. leaves on a tree, a flag in the wind, a construction flasher, etc.) could result in more than one color being included in the background model. This results in a transparency effect which allows the background to accept two or more separate colors.

The following figure is to visually show the whole process of Gaussian Mixture Model

$$P(X_t) = \sum_{i=1}^{K} w_{i,t} * \eta(X_t, \mu_{i,t}, \Sigma_{i,t})$$

$$P(X_t) = \sum_{i=1}^{K} w_{i,t} * \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(X_t-\mu_t)^T \Sigma^{-1}(X_t-\mu_t)}$$

$$P(X_t) = \sum_{i=1}^{K} w_{i,t} * \frac{1}{(2\pi)^{\frac{n}{2}}|\sigma_k^2 I|^{\frac{1}{2}}} e^{-\frac{1}{2}(X_t-\mu_t)^T \Sigma^{-1}(X_t-\mu_t)}$$

each pixel is constructed by K mixture Gaussian distribution

Determining whether matching each pixel value of current input frame to K Gaussian distribution according to $|X_t - \mu_{i,t-1}| < 2.5\sigma_{i,t-1}$

if matching successfully

if matching unsuccessfully

Based on following formulas:

$$\mu_t = (1-\rho)\mu_{t-1} + \rho X_t$$
$$\sigma_t^2 = (1-\rho)\sigma_{t-1}^2 + \rho(X_t - \mu_t)^T(X_t - \mu_t)$$
$$w_{k,t} = (1-\alpha)w_{k,t-1} + \alpha$$

Increasing the weight, mean and variance of matching Gaussian distribution (α is learning rate, deciding the speed of background updating, if it is small then the adaptive ability is lower than when it is high that easily causing the noise)

Based on following formulas:

$$w_{k,t} = (1-\alpha)w_{k,t-1}$$

Decreasing the weight, but the mean and variance do not change and continue to match remaining Gaussian distribution

If all the Gaussian distribution do not match its value, then according to w/σ, sorting distribution from top to bottom. and replacing the least distribution with current mean and large variance and then normalize each weight

after updating the background, the correlation between Gaussian model and background is higher as the σ decreases and w increases.

sorting the background distribution from the top to bottom according to w/σ and the most probable is the top T distribution, satisfying the following equation:

$$B = \mathrm{argmin}_b\left(\sum_{k=1}^{b} w_k > T\right)$$

where T is a measure of the minimum portion of the data that should be accounted for by the background. This takes the "best" distributions until a certain portion, T, of the recent data has been accounted for.

Figure 1: the whole process of Gaussian mixture model

IV. Result

This part is to demonstrate the acutal performance of the Gaussian mixture model with some visualized figures, based on the algorithms in part two and implemented by using python. Several videos are tested in this part. The camera is static in these videos and objects randomly appear in these static scene. Based on experiments, the processing rate is related to the used processor and the frame size.

This section firstly show the whole performance of GMM algorithm is segmenting foreground and background. And then different learning rate and threshold T and K will be test to discuss different performance since it is known that the learning rate and threshold T and the number of distributions K have impact on performance.

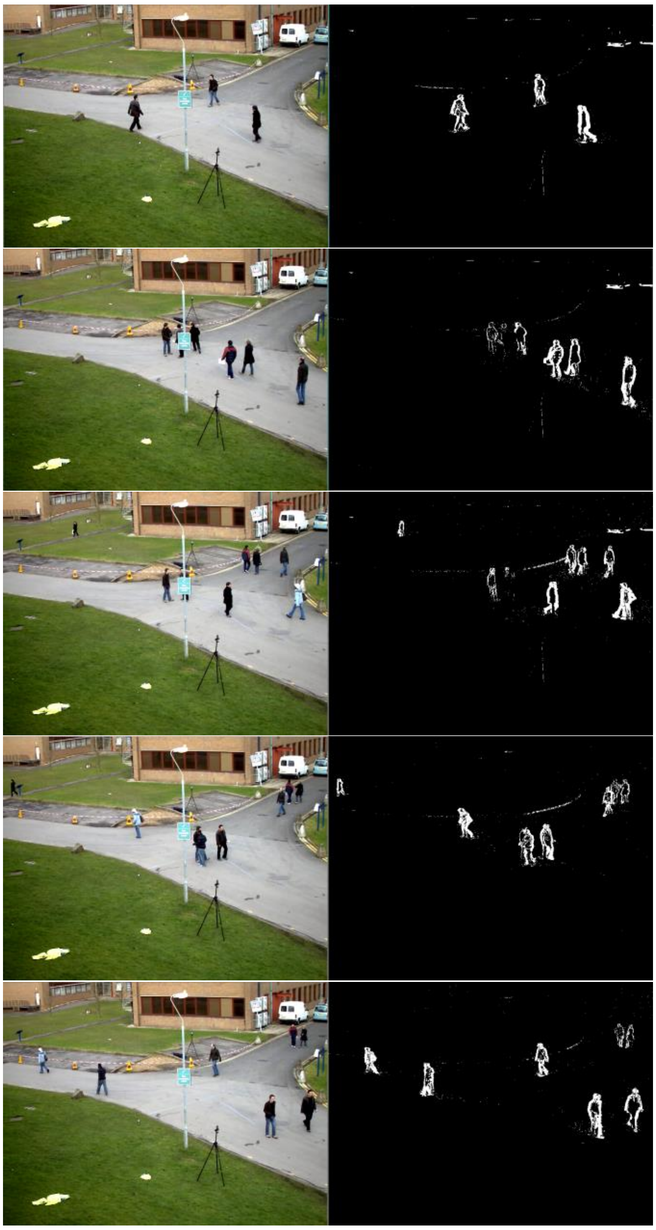

Figure 2: the performance of GMM algorithm in background and foreground segmentation. The left column is original frame and the right column is the processed framed.

The used videos are from PETS 2009 database (ftp://pets.rdg.ac.uk)[42]. This test video have mutiple moving objects crossing each other. Figure 2 shows segmentation in one scene. The learning rate, the threshold T and the number of Gaussian distributions here are set to 0.2 and 0.7 and 3, respectively. Multiple moving objects can also be segmented by GMM algorithm.

A. *Testing in different number of distributions*

It is evident from the Figure 2 that the contours of moving object is not quite clear, resulting from the unchanging number of Gaussian distributions in this experiment. In order to visulize the different performance between the unchanging and changing number of Gaussian distributions, the package from OPENCV, namely MOG2, is used, which is also a Gaussian mixture-based background/foreground segmentation algorithm. It is proposed by Zivkovic in[43]. The important feature of this algorithm is that it selects the appropriate number of Gaussian distribution for each pixel, instead of invariably tooking a K Gaussian distributions throughout the algorithm.

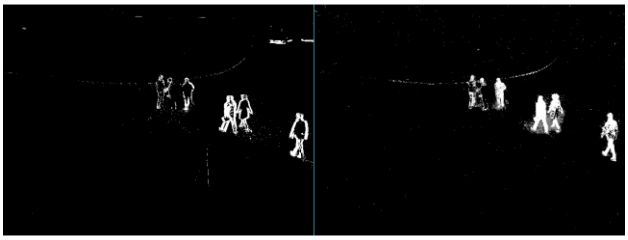

Figure 3: testing the different number of Gaussian distributions. The left one is arbitrarily choosing K =3 and the right one is processed frame using MOG2 library.

What we can see from the Figure 3 is that the MOG2 performance is better than the basic Gaussian mixture model in moving object contours and denoising.

B. *Testing in different learning rate*

It also can be seen in Figure 2 that there is noise in the processed image. As mentioned before, the learning rate has impacts on change detection as it increases, resulting the noise will easily appear.

In order to test what the difference between different learning rate, the single moving objected is observed, rather than multiple moving objects. In the following experiments, the different learning rate are set. Figure 4 shows that the performance of GMM algorithm with different learning rate with only single moving objects.

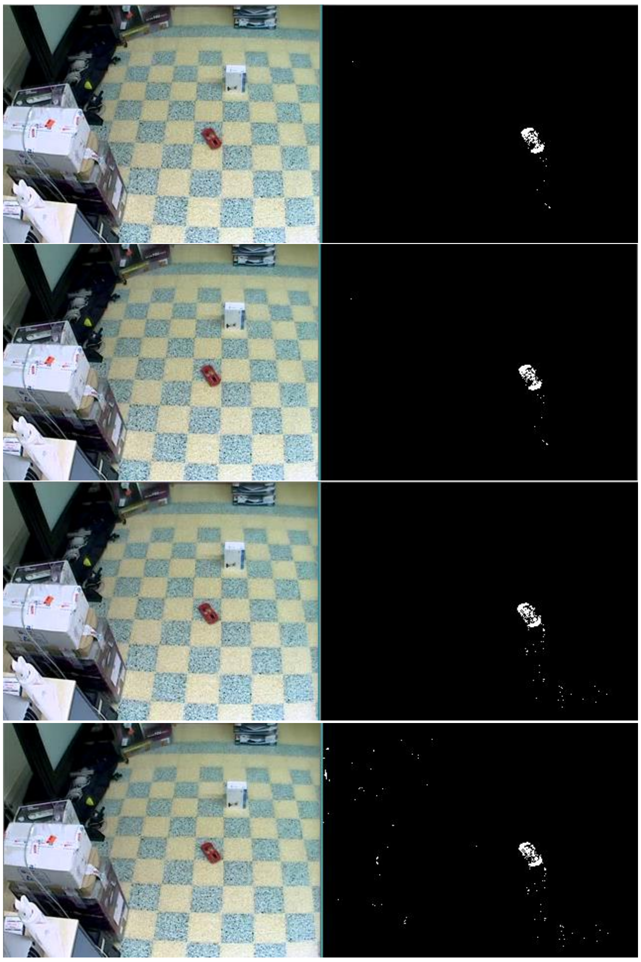

Figure 4: testing the different performance with different learning rate. The image from the top to the bottom are corresponding the learning rate 0.00001, 0.005, 0.2 and 0.5, respectively.

It can be seen obviously that as the magnitude of learning rate increases, the noise becomes more evident and apparent. However, arbitrarily choosing the magnitude of learning rate can not perform robustly in different conditions. It is necessary to adapt the learning rate to handle diverse scenes.

C. *Testing in different threshold*

It is mention before that the magnitude of threshold T is also an influence factors on GMM performance. In this sub-section, the different value of T are set and the performance will be observed.

Figure 5: testing the different performance with different threshold. The magnitude of threshold of the left one is 0.3 and the right one is 0.7.

It can be seen that the lower threshold will occur more noise in the image, even cosidering the unmoving objects as the moving object and then show them as foreground. Hence what is evident is that as the magnitude of threshold of T, the result could be better.

V. FUTURE WORKS

In this project, Gaussian mixture models is modeled mathemtically and is also basically implemented practically to observed the performance. However, the running rate is not quite satisfactory in real-time processing tasks. In this experiments, the processor is based on the Interl I5-7300 CPU. We also demonstrate that using a high quality of CPU can run the algorithm faster and can use a large number of Gaussians distributions in this model to improve the segmentation performance. Besides that, the performance of segmentation can further developed by using a full covariance matrix. Meanwhile, Stauffer and Grimson also present that making a system more robust in tracking of lighting changes can be done by adding prediction to each Gaussian distribution. We also mention some impacted factors on performance in previous section. Additionally, the inter-dependecies of the pixel processes is a direction to improve the result.

Although the segmentation is done in this project, the improved performance can be done in tracking and counting the moving object.

VI. CONCLUSION

This project is target to probabilistically model a segmentation algrothim that adaptive to different scenes based on the Gaussian mixture model. It models each pixel with numerous Gaussian distributions. The parameters α and T is changeable to adapt to different scenarios or working conditions. This method is capabale to handle mutiple moving object detection of the real world, which means that background and foreground can be seperated. It is a simple implementation that do not require huge data set to train and it saving time to development such requirements like tracking, segmentation.

This system has been test in several videos, which work effectively in different scens, whatever indoor or outdoor scenes. Meanwhile, this report also elaborates the hidden mathematical principles of Gaussian mixture models. Based these algorthms, using python to implement, the outcomes of segmentation are visualized in this report. Besides that this report considers that both increasing the learning rate and decreasing the magnitude of threshold will cause more noise while raising the number of the Gaussian distribution for each pixel will result in better contours of moving objects. The improved performance can be done in prediction, processing speed direction. This report achieves the goals of change detection.

FERENCES

[1] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in 1999, . DOI: 10.1109/CVPR.1999.784637.

[2] J. Yu, X. Zhou and F. Qian, "Object kinematic model: A novel approach of adaptive background mixture models for video segmentation," in 2010, . DOI: 10.1109/WCICA.2010.5554402.

[3] R. T Collins, A. J. Lipton, T.Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, ... & L. Wixson (2000). A system for video surveillance and monitoring. *VSAM final report*, 1-68.

[4] B. Farou *et al*, "Efficient local monitoring approach for the task of background subtraction,"*Engineering Applications of Artificial Intelligence,* vol. 64, pp. 1-12, 2017.

[5] C. R. Wren *et al*, "Pfinder: real-time tracking of the human body," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 19, *(7),* pp. 780-785, 1997.

[6] N. Friedman and S. Russell, "Image segmentation in video sequences: a probabilistic approach," presented at the Proceedings of the Thirteenth conference on Uncertainty in artificial intelligence, Providence, Rhode Island, 1997.

[7] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric model for background subtraction," in *European conference on computer vision*, 2000, pp. 751-767: Springer.

[8] H. Smail, H. David, and S. D. Larry, "W4: Real-Time surveillance of People and their Activities," *IEEE transactions on pattern analysis and machine intelligence*, vol. 22, no. 8, 2000.

[9] N. M. Oliver, B. Rosario, and A. P. Pentland, "A Bayesian computer vision system for modeling human interactions," *IEEE transactions on pattern analysis and machine intelligence*, vol. 22, no. 8, pp. 831-843, 2000.

[10] D.-M. Tsai and S.-C. Lai, "Independent component analysis-based background subtraction for indoor surveillance," *IEEE Transactions on image processing*, vol. 18, no. 1, pp. 158-167, 2008.

[11] S. S. Bucak and B. Gunsel, "Video content representation by incremental non-negative matrix factorization," in *2007 IEEE International Conference on Image Processing*, 2007, vol. 2, pp. II-113-II-116: IEEE.

[12] M. M. Azab, H. A. Shedeed, and A. S. Hussein, "A new technique for background modeling and subtraction for motion detection in real-time videos," in *2010 IEEE International Conference on Image Processing*, 2010, pp. 3453-3456: IEEE.

[13] X. Jian, D. Xiao-qing, W. Sheng-jin, and W. You-shou, "Background subtraction based on a combination of texture, color and intensity," in *2008 9th International Conference on Signal Processing*, 2008, pp.

1400-1405: IEEE.
[14] V. Jain, B. B. Kimia, and J. L. Mundy, "Background modeling based on subpixel edges," in *2007 IEEE International Conference on Image Processing*, 2007, vol. 6, pp. VI-321-VI-324: IEEE.
[15] F. Kristensen, P. Nilsson, and V. Öwall, "Background segmentation beyond RGB," in *Asian COnference on Computer Vision*, 2006, pp. 602-612: Springer.
[16] H. Zhang and D. Xu, "Fusing color and texture features for background model," in *Fuzzy Systems and Knowledge Discovery: Third International Conference, FSKD 2006, Xi'an, China, September 24-28, 2006. Proceedings 3*, 2006, pp. 887-893: Springer.
[17] F. El Baf, T. Bouwmans, and B. Vachon, "Fuzzy integral for moving object detection," in *2008 IEEE International Conference on Fuzzy Systems (IEEE World Congress on Computational Intelligence)*, 2008, pp. 1729-1736: IEEE.
[18] M. H. Sigari, N. Mozayani, and H. Pourreza, "Fuzzy running average and fuzzy background subtraction: concepts and application," *International Journal of Computer Science and Network Security,* vol. 8, no. 2, pp. 138-143, 2008.
[19] T. Bouwmans and F. El Baf, "Modeling of dynamic backgrounds by type-2 fuzzy Gaussians mixture models," *MASAUM Journal of of Basic and Applied Sciences,* vol. 1, no. 2, pp. 265-276, 2010.
[20] F. El Baf, T. Bouwmans, and B. Vachon, "Type-2 fuzzy mixture of Gaussians model: application to background modeling," in *International Symposium on Visual Computing*, 2008, pp. 772-781: Springer.
[21] F. El Baf, T. Bouwmans, and B. Vachon, "Fuzzy statistical modeling of dynamic backgrounds for moving object detection in infrared videos," in *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2009, pp. 60-65: IEEE.
[22] Z. Zhao, T. Bouwmans, X. Zhang, and Y. Fang, "A fuzzy background modeling approach for motion detection in dynamic backgrounds," in *International Conference on Multimedia and Signal Processing*, 2012, pp. 177-185: Springer.
[23] P. W. Power and J. A. Schoonees, "Understanding background mixture models for foreground segmentation," in *Proceedings image and vision computing New Zealand*, 2002, vol. 2002.
[24] P. KaewTraKulPong and R. Bowden, "An improved adaptive background mixture model for real-time tracking with shadow detection," in *Video-based surveillance systems*: Springer, 2002, pp. 135-144.
[25] T. Charoenpong, A. Supasuteekul, and C. Nuthong, "Adaptive background modeling from an image sequence by using K-Means clustering," in *ECTI-CON2010: The 2010 ECTI International Confernce on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, 2010, pp. 880-883: IEEE.
[26] A. Doulamis, I. Kalisperakis, C. Stentoumis, and N. Matsatsinis, "Self Adaptive background modeling for identifying persons' falls," in *2010 Fifth International Workshop Semantic Media Adaptation and Personalization*, 2010, pp. 57-63: IEEE.
[27] K. Schindler and H. Wang, "Smooth foreground-background segmentation for video processing," in *Asian Conference on Computer Vision*, 2006, pp. 581-590: Springer.
[28] B. Farou, M. N. Kouahla, H. Seridi, and H. Akdag, "Efficient local monitoring approach for the task of background subtraction," *Engineering Applications of Artificial Intelligence,* vol. 64, pp. 1-12, 2017.
[29] R. Caseiro, J. F. Henriques, and J. Batista, "Foreground segmentation via background modeling on Riemannian manifolds," in *2010 20th International Conference on Pattern Recognition*, 2010, pp. 3570-3574: IEEE.
[30] N. A. Setiawan, H. Seok-Ju, K. Jang-Woon, and L. Chil-Woo, "Gaussian mixture model in improved hls color space for human silhouette extraction," in *International Conference on Artificial Reality and Telexistence*, 2006, pp. 732-741: Springer.
[31] M. Seki, H. Okuda, M. Hashimoto, and N. Hirata, "Object modeling using gaussian mixture model for infrared image and its application to vehicle detection," *Journal of Robotics and Mechatronics,* vol. 18, no. 6, p. 738, 2006.
[32] A. Shimada, Y. Nonaka, H. Nagahara, and R.-i. Taniguchi, "Case-based background modeling: associative background database towards low-cost and high-performance change detection," *Machine vision and applications,* vol. 25, no. 5, pp. 1121-1131, 2014.
[33] X. Fang, W. Xiong, B. Hu, and L. Wang, "A moving object detection algorithm based on color information," in *Journal of Physics: Conference Series*, 2006, vol. 48, no. 1, p. 384: IOP Publishing.
[34] D. Pokrajac and L. J. Latecki, "Spatiotemporal blocks-based moving objects identification and tracking," *IEEE Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS),* pp. 70-77, 2003.
[35] S. Varadarajan, P. Miller, and H. Zhou, "Spatial mixture of Gaussians for dynamic background modelling," in *2013 10th IEEE International Conference on Advanced Video and Signal Based Surveillance*, 2013, pp. 63-68: IEEE.
[36] H. Bhaskar, L. Mihaylova, and S. Maskell, "Automatic target detection based on background modeling using adaptive cluster density estimation," 2007.
[37] B. Valentine, S. Apewokin, L. Wills, and S. Wills, "An efficient, chromatic clustering-based background model for embedded vision platforms," *Computer Vision and Image Understanding,* vol. 114, no. 11, pp. 1152-1163, 2010.
[38] L. A. Lim and H. Y. Keles, "Foreground segmentation using a triplet convolutional neural network for multiscale feature encoding," *arXiv preprint arXiv:1801.02225,* 2018.
[39] Y. Wang, Z. Luo, and P.-M. Jodoin, "Interactive deep learning method for segmenting moving objects," *Pattern Recognition Letters,* vol. 96, pp. 66-75, 2017.
[40] M. Babaee, D. T. Dinh, and G. Rigoll, "A deep convolutional neural network for video sequence background subtraction," *Pattern Recognition,* vol. 76, pp. 635-649, 2018.
[41] M. Braham and M. Van Droogenbroeck, "Deep background subtraction with scene-specific convolutional neural networks," in *2016 international conference on systems, signals and image processing (IWSSIP)*, 2016, pp. 1-4: IEEE.
[42] Cvg.reading.ac.uk. (2019). *PETS 2009*. [online] Available at: http://www.cvg.reading.ac.uk/PETS2009/a.html [Accessed 8 Jun. 2019].
[43] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction," in *ICPR (2)*, 2004, pp. 28-31: Citeseer.