International Journal of Computer Engineering and Technology (IJCET)
Volume 16, Issue 1, Jan-Feb 2025, pp. 2196 -2207, Article ID: IJCET_16_01_157
Available online at https://iaeme.com/Home/issue/IJCET?Volume=16&Issue=1
ISSN Print: 0976-6367; ISSN Online: 0976-6375; Journal ID: 5751-5249
Impact Factor (2025): 18.59 (Based on Google Scholar Citation)
DOI: https://doi.org/10.34218/IJCET_16_01_157

ADVANCED MOTION DETECTION APPROACHES OF REAL-TIME VIDEO PROCESSING USING GAUSSIAN AND Σ-Δ MODELS

Mostafa M. Abutaleb[1,2]

[1]Unit of Scientific Research, Applied College, Qassim University, Buraydah 51425, Saudi Arabia.

[2]Department of Electronics and Communications Engineering, Helwan University, Cairo 11792, Egypt.

ABSTRACT

Motion detection in video sequences is a critical task in computer vision, with applications ranging from surveillance to autonomous systems. This paper presents two efficient algorithms for motion detection: the Gaussian Single-Pixel Multiple-Distributions (SPMD) algorithm and the Σ-Δ SPMD algorithm. Both methods aim to reduce computational complexity while maintaining high accuracy in detecting moving objects. The Gaussian SPMD algorithm models each pixel as a mixture of Gaussian distributions, while the Σ-Δ SPMD algorithm employs Σ-Δ modulation for background estimation. Experimental results demonstrate that the proposed algorithms significantly reduce processing time compared to traditional methods, such as the Stauffer algorithm, while maintaining competitive accuracy. The Gaussian SPMD algorithm achieves a 42% reduction in processing time, and the Σ-Δ SPMD algorithm further improves frame rates, making it suitable for real-time applications. This study provides a comprehensive *comparison of the two approaches, highlighting their strengths and limitations in various scenarios.*

Keywords: Motion Detection, Real-Time Video Processing, Gaussian Mixture Models (GMM), Sigma-Delta Modulation, Background Subtraction.

Cite this Article: Mostafa M. Abutaleb. Advanced Motion Detection Approaches of Real-Time Video Processing Using Gaussian and Σ-Δ Models. *International Journal of Computer Engineering and Technology (IJCET)*, 16(1), 2025, 2196-2207.

https://iaeme.com/MasterAdmin/Journal_uploads/IJCET/VOLUME_16_ISSUE_1/IJCET_16_01_157.pdf

I. Introduction

Motion detection is a fundamental task in video processing, with applications in surveillance, traffic monitoring, and human-computer interaction. Traditional methods, such as the Stauffer algorithm [1], rely on Gaussian mixture models (GMMs) to represent pixel intensities over time. While effective, these methods are computationally intensive, making them unsuitable for real-time applications. Recent advancements have focused on reducing computational complexity while maintaining detection accuracy [2, 3].

This paper introduces two novel algorithms for motion detection: the Gaussian SPMD algorithm and the Σ-Δ SPMD algorithm. The Gaussian SPMD algorithm simplifies the traditional GMM approach by reducing the number of parameters per pixel, while the Σ-Δ SPMD algorithm leverages Σ-Δ modulation for efficient background estimation [4]. Both algorithms are designed for real-time implementation, with a focus on minimizing computational overhead.

The contributions of this paper are as follows:

- A detailed description of the Gaussian SPMD and Σ-Δ SPMD algorithms, including their mathematical formulations and implementation details.
- A comparative analysis of the proposed algorithms with traditional methods, demonstrating significant improvements in processing time and accuracy.
- Experimental results on standard video sequences, highlighting the performance of the proposed algorithms in various scenarios.

The remainder of this paper is organized as follows: Section 2 reviews related work in motion detection. Section 3 presents the Gaussian SPMD algorithm, and Section 4 describes the Σ-Δ SPMD algorithm. Section 5 provides experimental results and a comparative analysis. Finally, Section 6 concludes the paper and discusses future work.

II. RELATED WORK

Motion detection [1,2] has been extensively studied in the literature. Traditional methods, such as the Stauffer algorithm, use Gaussian mixture models to represent pixel intensities over time. While effective, these methods are computationally expensive due to the need for floating-point calculations and iterative updates. Recent works are presented for improvements to the Stauffer algorithm by dynamically selecting the number of Gaussian distributions, reducing computational overhead.

Alternative approaches, such as temporal differencing [5] and Σ-Δ modulation [6], have been proposed to address these limitations. Temporal differencing is computationally efficient but sensitive to noise and lighting changes [7]. Σ-Δ modulation, on the other hand, provides a low-complexity method for background estimation but suffers from limitations in complex scenes and slow-motion detection [8].

Recent advances [9-14] in deep learning have also influenced the development of more robust motion detection techniques. These methods, however, require significant computational resources, making them less suitable for real-time applications on low-power devices.

This paper builds on these works by proposing two novel algorithms that combine the strengths of GMMs and Σ-Δ modulation while addressing their limitations. The proposed algorithms are designed for real-time implementation, with a focus on minimizing computational overhead and maintaining high accuracy.

III. GAUSSIAN SPMD ALGORITHM

A. Algorithm Overview

The Gaussian SPMD algorithm models each pixel as a mixture of three Gaussian distributions, reducing the computational complexity of traditional GMMs. Each distribution is represented by its mean and weight, and the mixture is ordered in decreasing order of weight values. The algorithm checks each pixel against the distributions and updates the parameters based on the matching condition.

B. Mathematical Formulation

The probability of observing the current pixel value X_t is given by:

$$P(X_t) = \sum_{k=1}^{K} \omega_{k,t} \cdot \eta(X_t | \mu_{k,t}, \sigma_{k,t})$$

where $\omega k,t$ is the weight of the k-th Gaussian distribution, $\mu k,t$ is its mean, and $\sigma k,t$ is its standard deviation. The matching condition is defined as:

$$(1 - R\%)\mu_{k,t-1} \leq X_t \leq (1 + R\%)\mu_{k,t-1}$$

where $R\%$ is the matching ratio (typically set to 20%). If a match is found, the distribution parameters are updated as follows:

$$\mu_{k,t} = (1 - \alpha)\mu_{k,t-1} + \alpha X_t$$

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

If no match is found, the pixel is classified as foreground, and the least probable distribution is replaced with a new distribution having the current pixel value as its mean.

C. Algorithm Implementation

The Gaussian SPMD algorithm is implemented as follows:

1. Initialization: For each pixel, initialize three Gaussian distributions with random means and weights.
2. Matching: For each new frame, check each pixel against the distributions using the matching condition.
3. Update: If a match is found, update the mean and weight of the matching distribution. If no match is found, classify the pixel as foreground and update the least probable distribution.
4. Sorting: Sort the distributions in decreasing order of weight values.

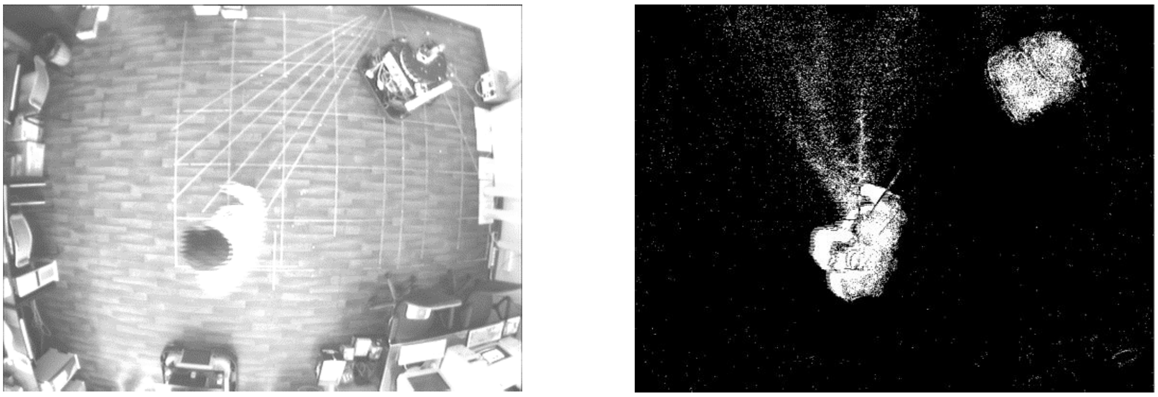
D. Results

The Gaussian SPMD algorithm was implemented in MATLAB and tested on 768x576 video frames based on listed parameters in Table 1. Experimental results show a 42% reduction in processing time compared to the Stauffer algorithm, with comparable accuracy. Fig. 1 illustrates the performance of Gaussian SPMD algorithm on the sample frame.

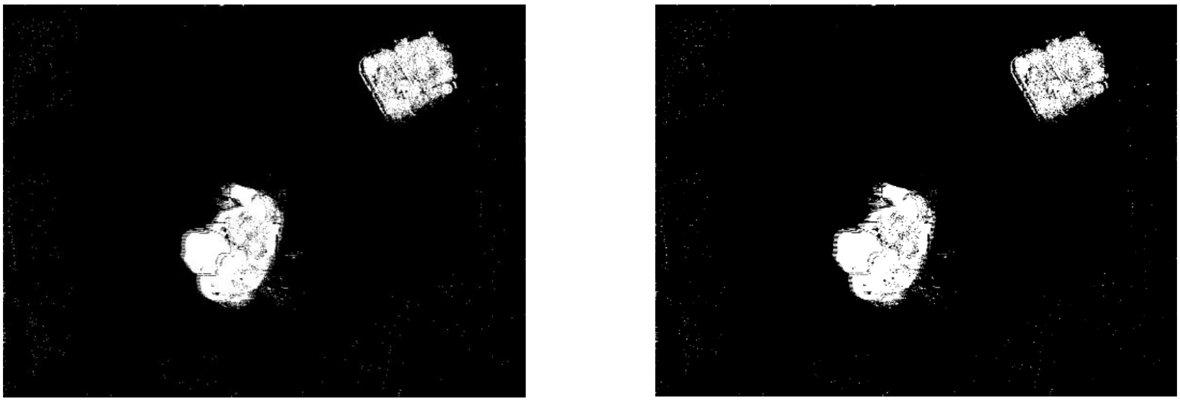
Table 1: Parameters for Gaussian SPMD Algorithm

Parameter	**Description**	**Value**
K	Number of Gaussian distributions	3
R%	Matching ratio	20%
α	Learning rate	0.05

The Gaussian SPMD algorithm provides higher accuracy, while the Σ-Δ SPMD algorithm offers better frame rates. The choice of algorithm depends on the specific application requirements, with the Gaussian SPMD algorithm being more suitable for accuracy-critical tasks and the Σ-Δ SPMD algorithm being better for real-time applications.



(a) (b)



(c) (d)

Fig. 1. (a) Original sample frame from the video sequence, (b) Foreground detection using the Stauffer algorithm, (c) Foreground detection using Gaussian SPMD algorithm, and (d) Foreground detection using Σ-Δ SPMD algorithm.

IV. Σ-Δ SPMD ALGORITHM

A. Algorithm Overview

The Σ-Δ SPMD algorithm uses Σ-Δ modulation for background estimation, providing a low-complexity alternative to traditional GMMs. Each pixel is modeled as a mixture of three distributions, represented by their Σ-Δ mean, variance, and weight. The algorithm checks each pixel against the distributions and updates the parameters based on the matching condition.

B. Mathematical Formulation

The Σ-Δ mean $M_{k,t}$ and variance $V_{k,t}$ are updated as follows:

$$M_{k,t} = M_{k,t-1} + \mathrm{sgn}(X_t - M_{k,t-1})$$

$$V_{k,t} = V_{k,t-1} + \mathrm{sgn}(N \cdot \Delta_{k,t} - V_{k,t-1})$$

where $\Delta_{k,t} = |X_t - M_{k,t-1}|$ and N is a constant. The matching condition is defined as:

$$\Delta_{k,t} < V_{k,t-1}$$

If a match is found, the pixel is classified as background, and the distribution parameters are updated. If no match is found, the pixel is classified as foreground.

C. Algorithm Implementation

The Σ-Δ SPMD algorithm is implemented as follows:

1. Initialization: For each pixel, initialize three distributions with random Σ-Δ means, variances, and weights.
2. Matching: For each new frame, compute the absolute difference $\Delta k,t$ and check the matching condition.
3. Update: If a match is found, update the Σ-Δ mean and variance of the matching distribution. If no match is found, classify the pixel as foreground and update the least probable distribution.
4. Sorting: Sort the distributions in decreasing order of weight values.

D. Results

The Σ-Δ SPMD algorithm was implemented and tested on standard video sequences based on listed parameters in Table 2. Experimental results show improved frame rates compared to the Gaussian SPMD algorithm, with slightly lower accuracy. Fig. 1 illustrates the performance of this algorithm on the sample frame.

Table 2: Parameters for Σ-Δ SPMD Algorithm

Parameter	Description	Value
N	Constant for variance update	20
K	Number of distributions	3

V. EXPERIMENTAL RESULTS AND COMPARATIVE ANALYSIS

A. Performance Metrics

The performance of the proposed algorithms was evaluated using two metrics: processing time per frame and error percentage. The error percentage is defined as the ratio of incorrect pixels to the total number of pixels in the frame.

B. Performance Results

The Gaussian SPMD algorithm for the sample frame, as listed in Table 3, achieved an average processing time of 1.0996 seconds per frame, compared to 121.5849 seconds for the Stauffer algorithm. The Σ-Δ SPMD algorithm further reduced the processing time to 0.8816 seconds per frame. The error percentage for the Gaussian SPMD algorithm was 5.2%, compared to 6.8% for the Σ-Δ SPMD algorithm.

Table 3: Comparison of Processing Time and Error Percentage

Algorithm	**Processing Time (s/frame)**	**Error Percentage (%)**
Stauffer Algorithm	121.5849	7.5
Gaussian SPMD	1.0996	5.2
Σ-Δ SPMD	0.8816	6.8

One frame is selected from each 25 frames in the sequence for testing; 22 samples. Fig. 2 shows the results of error percent test found in the results of the Stauffer algorithm and the proposed Gauss SPMD algorithm. It is clear that the test results show improvement in the foreground extracted using the proposed algorithm compared to that generated by Stauffer algorithm. Fig. 3 shows the results of error percent test found in the results of the Σ–Δ estimation algorithm and the proposed Σ–Δ SPMD algorithm. It is clear that the test results show improvement in the foreground extracted using the proposed algorithm compared to that generated by Σ–Δ estimation algorithm.

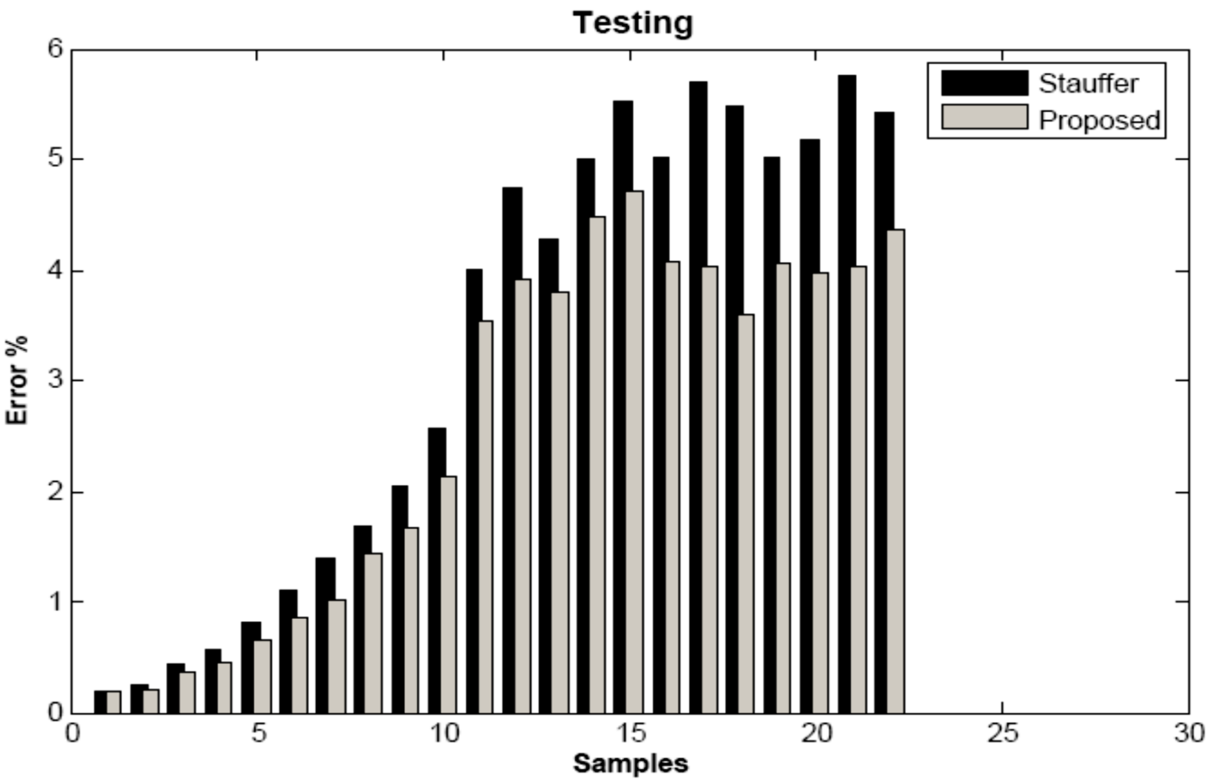


Fig. 2. Testing Results of the Stauffer algorithm and the proposed Gauss SPMD algorithm.

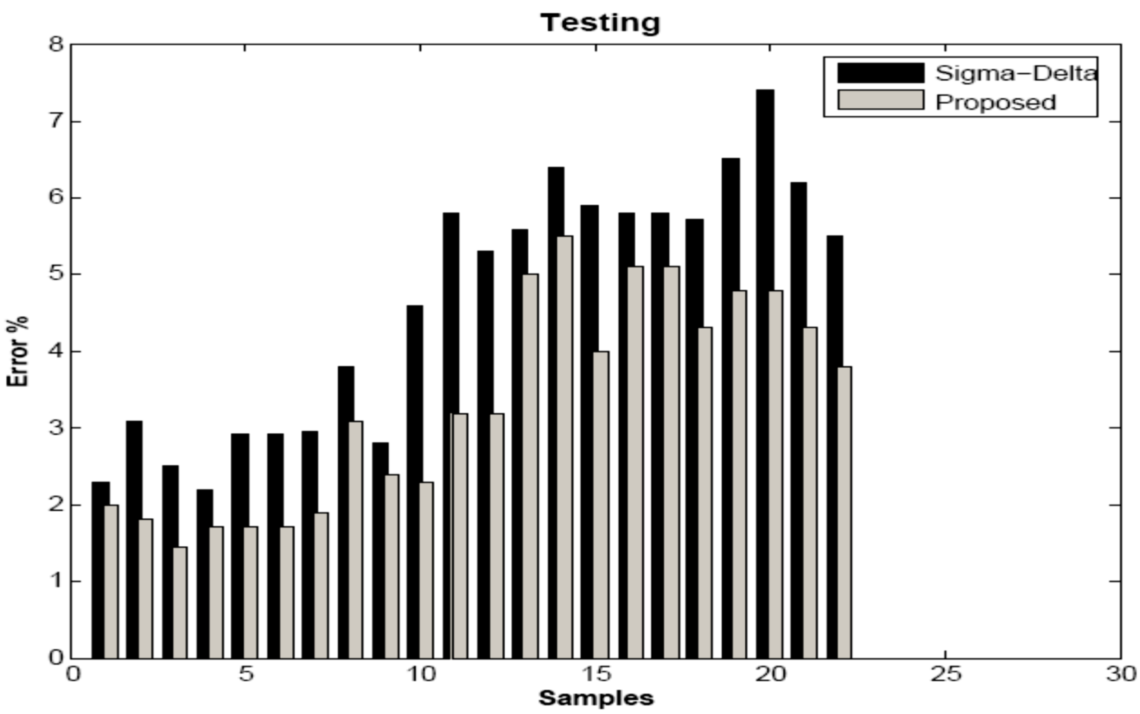


Fig. 3. Testing Results of the Σ–Δ estimation algorithm and the proposed Σ–Δ SPMD algorithm.

The Gaussian SPMD algorithm provides higher accuracy, while the Σ-Δ SPMD algorithm offers better frame rates. The choice of algorithm depends on the specific application requirements, with the Gaussian SPMD algorithm being more suitable for accuracy-critical tasks and the Σ-Δ SPMD algorithm being better for real-time applications.

VI. CONCLUSION

This paper presented two efficient algorithms for motion detection in video sequences: the Gaussian SPMD algorithm and the Σ-Δ SPMD algorithm. Both algorithms significantly reduce computational complexity compared to traditional methods, making them suitable for real-time applications. The Gaussian SPMD algorithm provides higher accuracy, while the Σ-Δ SPMD algorithm offers better frame rates. Future work will focus on further optimizing the algorithms for hardware implementation and extending their applications to multi-camera systems.

VII. REFERENCES

[1] Stauffer, C., & Grimson, W. E. L. (1999). Adaptive background mixture models for real-time tracking. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2, 246-252.

[2] Xu, S., Wang, J., Shou, W., Ngo, T., Sadick, A. M., & Wang, X. (2021). Computer vision techniques in construction: a critical review. Archives of Computational Methods in Engineering, 28, 3383-3397.

[3] Bouwmans, T. (2014). Traditional and recent approaches in background modeling for foreground detection: An overview. Computer science review, 11, 31-66.

[4] Abutaleb, M. M., Hamdy, A., Abuelwafa, M. E., & Saad, E. M. (2009, February). FPGA-based object-extraction based on multimodal Σ-Δ background estimation. In

2009 2nd International Conference on Computer, Control and Communication, IEEE, (pp. 1-7).

[5] Abutaleb, M. M., Hamdy, A., & Saad, E. M. (2008). FPGA-based real-time video-object segmentation with optimization schemes. Int. J. Circ. Syst. Sign. Proc, 2, 78-86.

[6] Mahvidi, M., & Ghods, V. (2018). Background Subtraction Techniques Evaluation based on Σ-Δ Algorithm for Motion Detection. Journal of Intelligent Procedures in Electrical Technology, 9(34), 29-46.

[7] Elgammal, A., Harwood, D., & Davis, L. (2000). Non-parametric model for background subtraction. *European Conference on Computer Vision*, 751-767.

[8] Barnich, O., & Van Droogenbroeck, M. (2010). ViBe: A universal background subtraction algorithm for video sequences. IEEE Transactions on Image processing, 20(6), 1709-1724.

[9] Sobral, A., & Vacavant, A. (2014). A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. Computer Vision and Image Understanding, 122, 4-21.

[10] Dou, J., Qin, Q., & Tu, Z. (2019). Background subtraction based on deep convolutional neural networks features. Multimedia Tools and Applications, 78(11), 14549-14571.

[11] Abutaleb, M. M., Hamdy, A., Abuelwafa, M. E., & Saad, E. M. (2009, March). A reliable fpga-based real-time optical-flow estimation. In 2009 National Radio Science Conference, IEEE, (pp. 1-8).

[12] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770-778.

[13] Sharma, V., Gupta, M., Kumar, A., & Mishra, D. (2021). Video processing using deep learning techniques: A systematic literature review. IEEE Access, 9, 139489-139507.

[14] Lohani, D., Crispim-Junior, C., Barthélemy, Q., Bertrand, S., Robinault, L., & Tougne Rodet, L. (2022). Perimeter intrusion detection by video surveillance: A survey. Sensors, 22(9), 3601.

[15] Sharma, M., singh Kaswan, K., & Yadav, D. K. (2023, November). Real Time Moving Object Detection for Video Surveillance in Presence of Low Illumination-Cloudy Environment. In 2023 3rd International Conference on Advancement in Electronics & Communication Engineering (AECE), IEEE, pp. 499-502.

Citation: Mostafa M. Abutaleb. Advanced Motion Detection Approaches of Real-Time Video Processing Using Gaussian and Σ-Δ Models. International Journal of Computer Engineering and Technology (IJCET), 16(1), 2025, 2196-2207.

Abstract Link: https://iaeme.com/Home/article_id/IJCET_16_01_157

Article Link:
https://iaeme.com/MasterAdmin/Journal_uploads/IJCET/VOLUME_16_ISSUE_1/IJCET_16_01_157.pdf

editor@iaeme.com