

## NHÓM 2

Lê Phạm Công 20KTMT1  
Phan Công Danh 20KTMT1  
Đoàn Thế Lên 20KTMT1  
Lê Minh Nhật 20KTMT1

## BÁO CÁO

### Ứng dụng GMM phát hiện chuyển động trong đoạn video

Ngày: 22 – 04 - 2024

-----

---

#### **1. Giới thiệu**

Ngày nay, an ninh trở nên vô cùng quan trọng và nhiều thiết bị điện tử được sử dụng trong các ứng dụng an ninh để liên tục theo dõi các chuyển động của con người hoặc phương tiện và báo cáo khi một sự kiện không được định trước xảy ra. Một hệ thống dựa trên quan sát của con người để thực hiện điều tương tự có nhiều nhược điểm, nơi mà một người phải quan sát đầu ra của camera trên một TV và phát hiện khi bất kỳ sự kiện bất ngờ nào xảy ra. Báo cáo này đề cập đến việc phát hiện vật thể di chuyển trong một video được quay bởi một camera cố định trong các tình huống động, nơi mà nền có thể thay đổi thường xuyên, thuật toán được đề xuất cần phải thích ứng nhanh chóng với các thay đổi trong nền và không nên có nhận dạng sai nền là tiền cảnh.

Báo cáo này giới thiệu một mô hình tham số cho việc trừ nền sử dụng Gaussian Mixture Model (GMM). Điều này có thể được sử dụng để phát hiện bất kỳ vật thể di chuyển nào trong một video. Một tệp AVI hoặc mp4 được đọc và chuyển đổi thành các khung hình, những khung hình này được chuyển đổi thành ảnh xám, sau đó mô hình Gaussian được áp dụng lên các khung hình, phân tách nền và tiền cảnh, các thao tác hình thái được áp dụng để cải thiện kết quả của tiền cảnh. Cuối cùng, các khung hình sau đó được sử dụng để tạo thành một video chỉ hiển thị các vật thể di chuyển trong video.

#### **2. Phương pháp**

Trong mục này giới thiệu 2 phần, phần 2.1 trình bày về Gaussian Mixture Model (GMM) thông qua ứng dụng với bài toán clustering, phần 2.2 sẽ trình bày việc áp dụng trong việc phát hiện đối tượng chuyển động trong video.

##### **2.1. Gaussian Mixture Model[1]**

Trong lĩnh vực của các thuật toán unsupervised learning, Gaussian Mixture Model (GMM) là những thành phần đặc biệt. GMM dựa trên giả định rằng tất cả các điểm dữ liệu đến từ một hỗn hợp tinh tế của phân phối Gaussian với các tham số không biết. Chúng là các mô hình sinh tham số cố gắng học được phân phối dữ liệu thực sự.

GMM như là sự tổng quát của thuật toán phân cụm K-Means. Giống như K-Means, GMM cũng đòi hỏi số lượng cụm K là đầu vào của thuật toán học. Tuy nhiên, có một sự khác biệt chính giữa hai thuật toán này. K-Means chỉ có thể học được các cụm có hình tròn. Ngược lại, GMM có thể học được các cụm có bất kỳ hình dạng elip nào.

Ngoài ra, K-Means chỉ cho phép một quan sát thuộc về một và chỉ một cụm. Khác biệt, GMM cung cấp các xác suất liên quan đến mỗi ví dụ với một cụm cụ thể. Đối với mỗi quan

sát, GMM học các xác suất của ví dụ đó thuộc về mỗi cụm  $k$ . Nói chung, GMM cố gắng học mỗi cụm như là một phân phối Gaussian khác nhau. Nó giả định dữ liệu được tạo ra từ một hỗn hợp giới hạn của Gaussians.

Giả định dữ liệu một chiều và số lượng cụm  $K$  bằng 3, GMM cố gắng học 9 tham số: 3 tham số cho các giá trị trung bình (mean), 3 tham số cho phương sai (variance), 3 tham số tỷ lệ.

Ở đây, mỗi cụm được biểu diễn bằng một phân phối Gaussian riêng biệt. Đối với mỗi Gaussian, nó học một tham số trung bình và một tham số phương sai từ dữ liệu. 3 tham số tỷ lệ, 1 cho mỗi Gaussian, chỉ được sử dụng cho ước lượng mật độ.

Để học các tham số như vậy, GMM sử dụng thuật toán kỳ vọng tối đa (EM: expectation-maximization) để tối ưu hóa ước lượng hợp lý tối đa. Trong quá trình này, GMM sử dụng Định lý Bayes để tính toán xác suất của một quan sát nhất định  $x_i$  thuộc về mỗi cụm  $k$ , cho  $k = 1, 2, \dots, K$ .

Xem xét một dữ liệu một chiều được tổng hợp. Xây dựng một tập dữ liệu mô phỏng bằng cách lấy mẫu các điểm từ  $K$  phân phối Gaussian khác nhau. Mỗi phân phối (với trung bình và phương sai riêng) đại diện cho một cụm khác nhau trong dữ liệu được tổng hợp. Để làm cho mọi thứ rõ ràng hơn, sử dụng  $K=3$ .

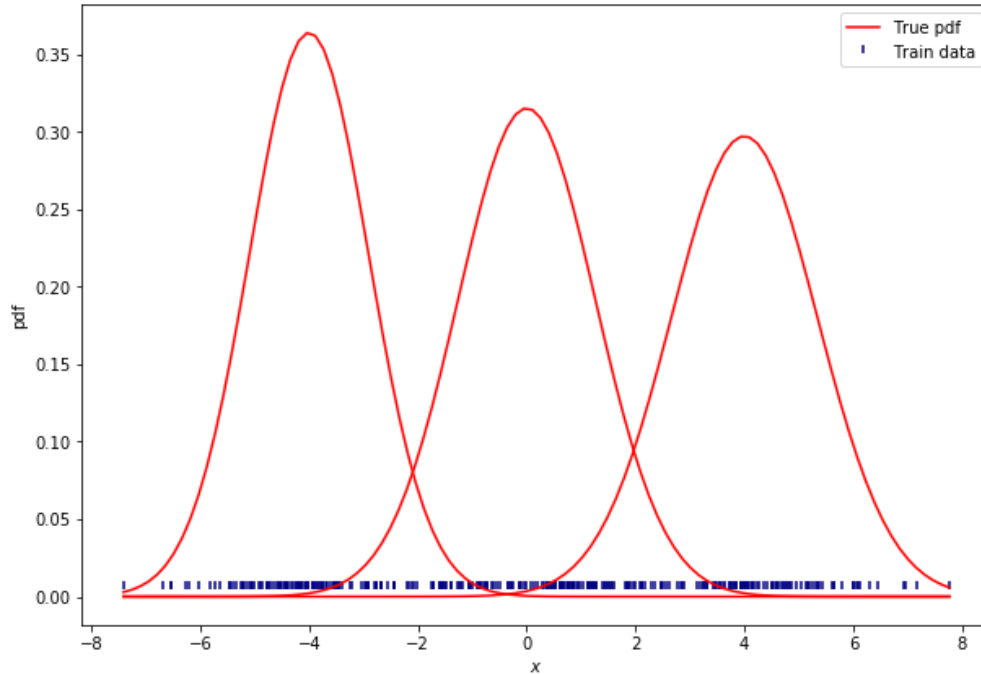
$$C_k \sim N(\mu, \sigma^2) \quad (2.1a)$$

Một biến ngẫu nhiên  $C_k$  tuân theo phân phối Gaussian được kí hiệu nhưng công thức (2.1a), trong đó  $\mu$  và  $\sigma$  là hai tham số đặc trưng của phân phối Gaussian. Phân phối Gaussian có hình dạng là một quả chuông mà giá trị xác suất lớn nhất tại  $C_k = \mu$ , hình dạng của phân phối đối xứng qua  $\mu$ .

Hàm mật độ xác suất (probability density function - pdf) được định nghĩa tại công thức (2.1b) như sau:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right) \quad (2.1b)$$

Hình 2.1 Mô tả dữ liệu được sử dụng để học các cụm với GMM. Một số điểm dữ liệu có thể bị chồng chéo với nhau



Hình 2.1. Mô tả dữ liệu sẽ sử dụng để học các cụm với GMM.

Có thể thấy GMM như là một tổng có trọng số của các phân phối Gaussian. Số lượng cụm K xác định số lượng phân phối Gaussian muốn điều chỉnh. Số lượng cụm cần được xác định trước. Số lượng cụm K đã được giả định là 3. Trong tình huống này, GMM sẽ cố gắng học 3 phân phối Gaussian. Đối với dữ liệu một chiều, cần học một tham số trung bình và một tham số phương sai cho mỗi Gaussian.

Trước khi bắt đầu chạy EM, cần đưa ra các giá trị ban đầu cho các tham số có thể học được. Có thể đoán các giá trị cho các trung bình và phương sai, và khởi tạo các tham số trọng số là  $1/k$ . Sau đó, có thể bắt đầu tối ưu hóa hợp lý tối đa bằng cách sử dụng thuật toán EM. EM có thể được đơn giản hóa trong 2 giai đoạn: các bước E (expectation: mong đợi) và M (maximization: tối đa hóa).

Trong bước E, toán xác suất của mỗi quan sát  $x_i$  bằng cách sử dụng các tham số ước lượng.

$$f(x|\mu_k, \sigma_k^2) = \frac{1}{\sqrt{2\pi\sigma_k^2}} \exp\left(-\frac{(x - \mu_k)^2}{2\sigma_k^2}\right) \quad (2.1c)$$

Công thức (2.1c) là công thức phân phối Gaussian một chiều (1-D Gaussian distribution) với các tham số:

$x$ : là biến ngẫu nhiên theo phân phối Gaussian

$\mu_k$ : trung bình của phân phối

$\sigma_k^2$ : phương sai của phân phối

Đối với mỗi cụm  $k = 1, 2, 3, \dots, K$ , tính toán mật độ xác suất (pdf) của dữ liệu của chúng ta bằng cách sử dụng các giá trị ước lượng cho trung bình và phương sai. Tại điểm này, những giá trị này chỉ là các dự đoán ngẫu nhiên.

Sau đó, có thể tính toán xác suất của một ví dụ cụ thể  $x_i$  thuộc về cụm thứ  $k$  như công thức (2.1d).

$$b_k = \frac{f(x|\mu_k, \sigma_k^2)\phi_k}{\sum_{k=1}^K f(x|\mu_k, \sigma_k^2)\phi_k} \quad (2.1d)$$

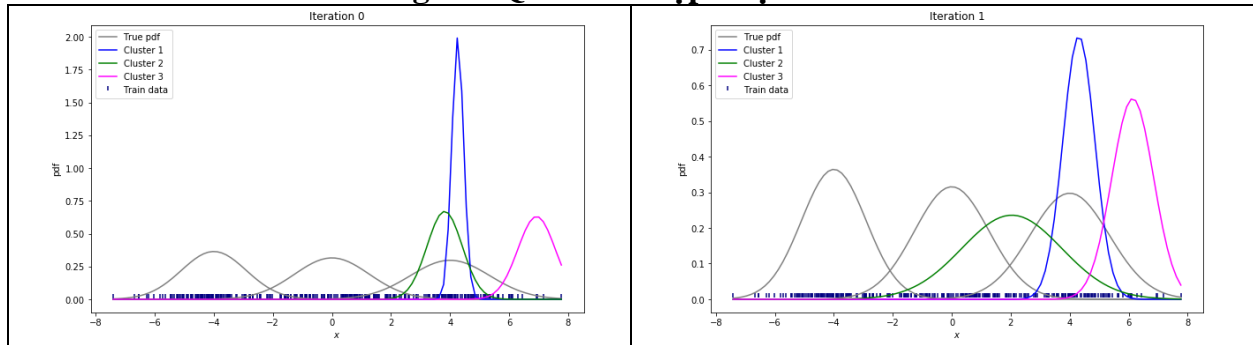
Sử dụng Định lý Bayes thu được xác suất hậu nghiệm của phân phối Gaussian thứ  $k$  để giải thích dữ liệu. Đó là xác suất mà quan sát  $x_i$  được tạo ra bởi phân phối Gaussian thứ  $k$ . Lưu ý rằng các tham số  $\Phi$  đóng vai trò như niềm tin tiên nghiệm của rằng một ví dụ được rút ra từ một trong các phân phối Gaussian đang được mô hình hóa. Vì không có bất kỳ thông tin bổ sung nào để ủng hộ một phân phối Gaussian hơn các phân phối khác nên sẽ bắt đầu bằng cách đoán rằng xác suất là bằng nhau để một ví dụ có thể đến từ mỗi phân phối Gaussian. Tuy nhiên, ở mỗi vòng lặp, cần cải thiện các tiên nghiệm cho đến khi hội tụ.

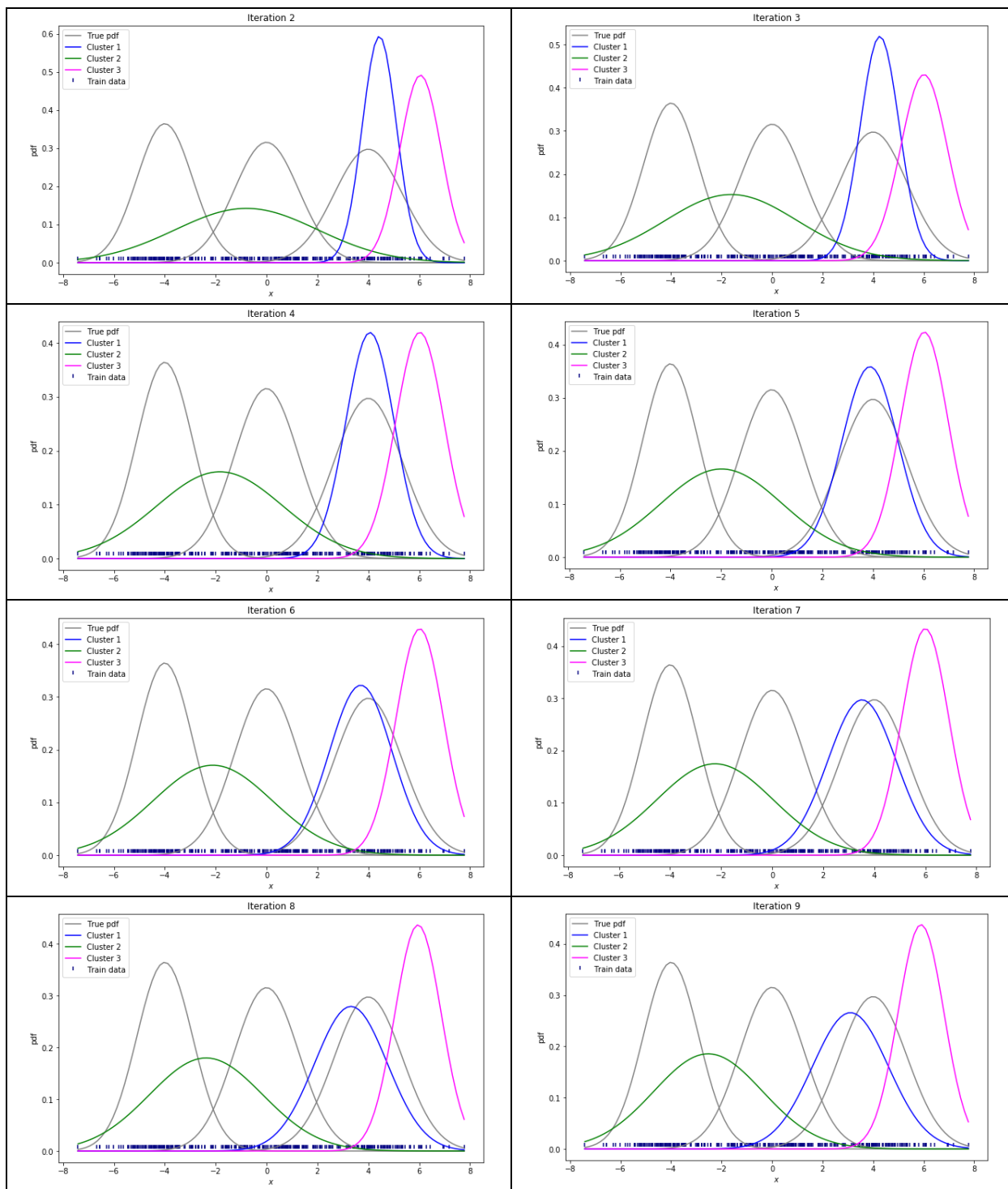
Sau đó, trong bước tối đa hóa, hoặc bước M, sẽ ước lượng lại các tham số học như công thức (2.1e):

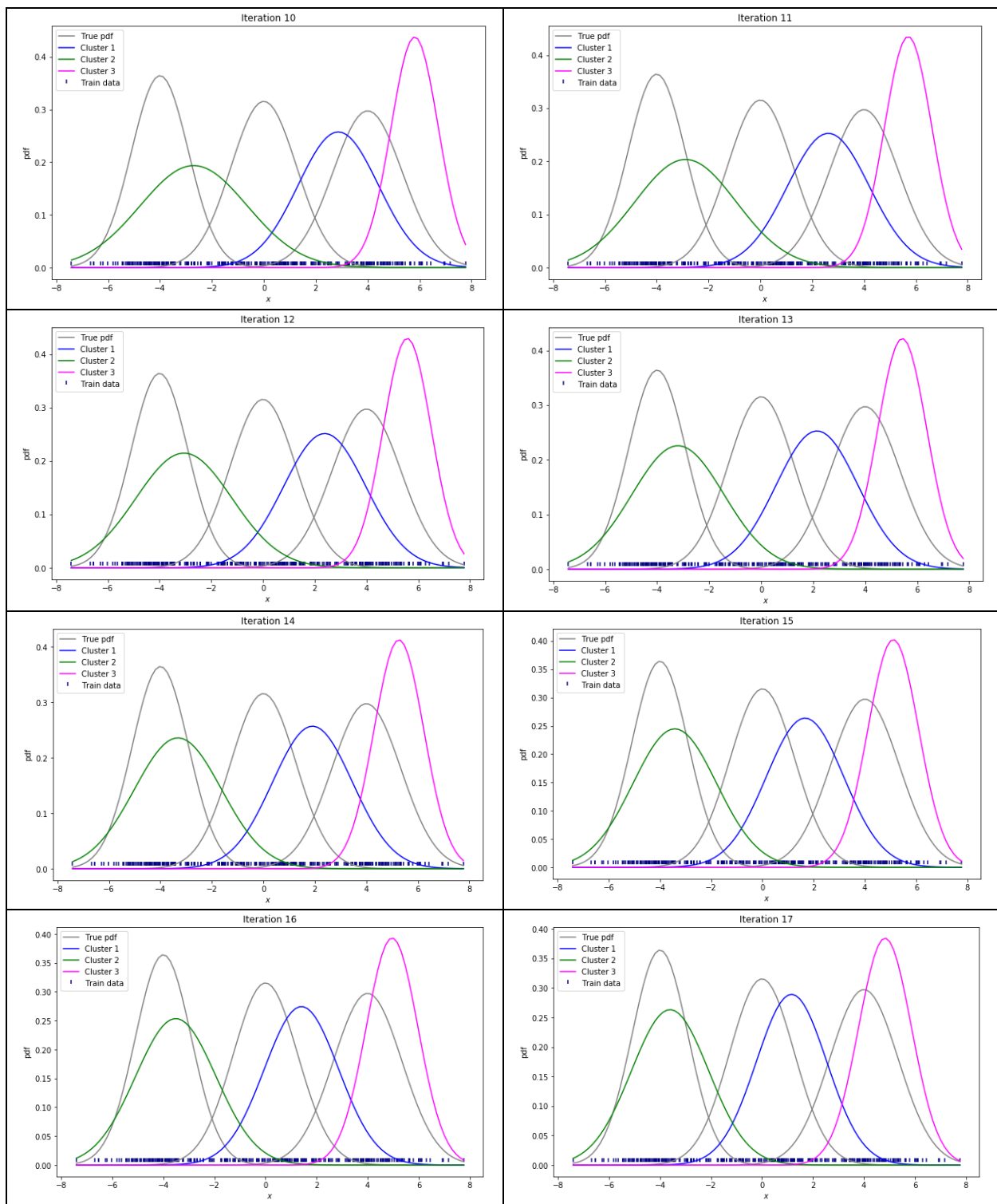
$$\mu_k = \frac{\sum b_k x}{\sum b_k}; \sigma_k^2 = \frac{\sum b_k (x - \mu_k)^2}{\sum b_k}; \phi_k = \frac{1}{N} \sum b_k \quad (2.1e)$$

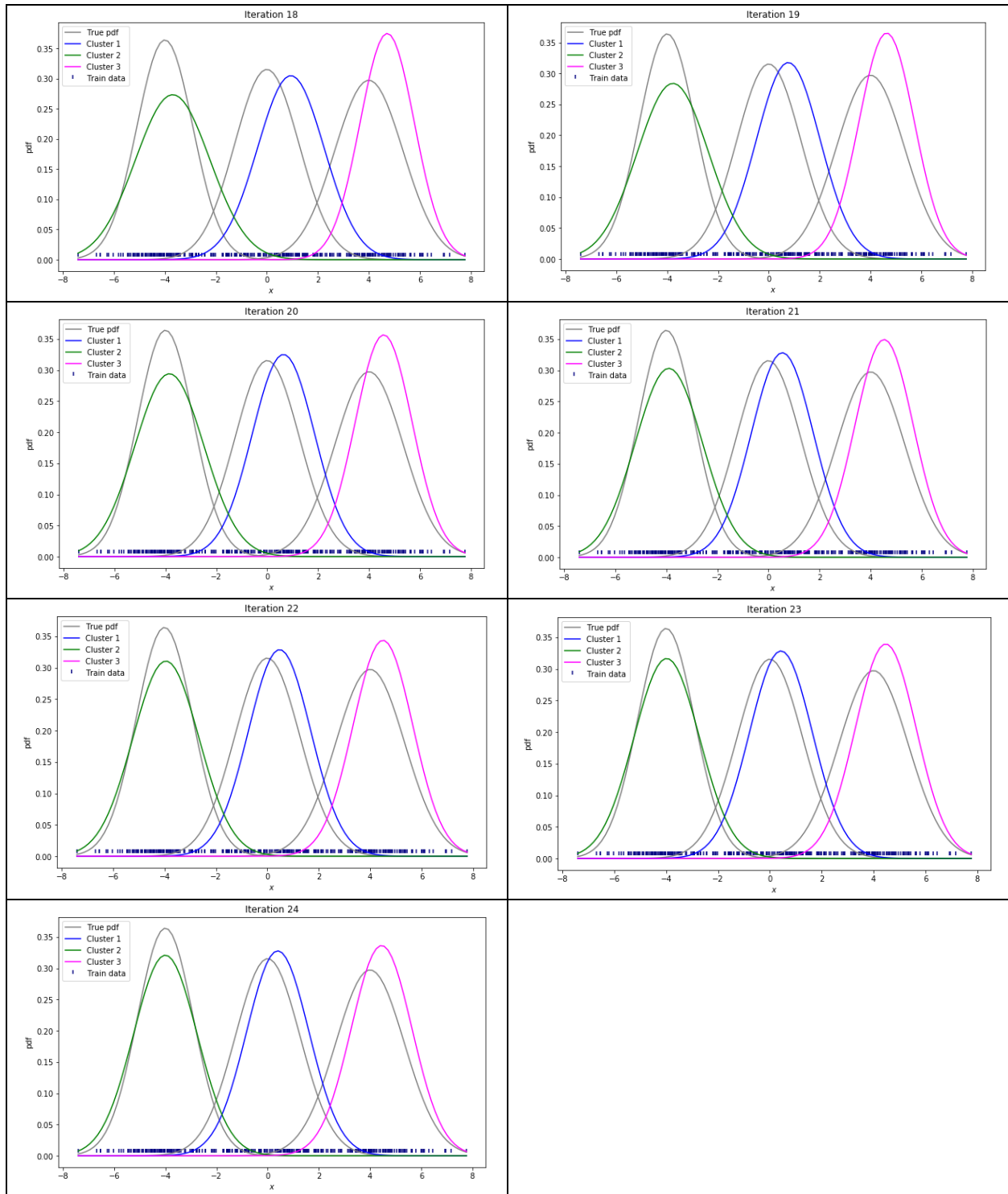
Ở đây, cho mỗi cụm, cập nhật trung bình ( $\mu_k$ ), phương sai ( $\sigma_k^2$ ), và các tham số tỷ lệ  $\Phi_k$ . Để cập nhật trung bình, cân nhắc mỗi quan sát bằng cách sử dụng xác suất có điều kiện  $b_k$ . Cần lặp lại các bước này cho đến khi hội tụ. Điều này có thể đến một điểm nơi các cập nhật tham số nhỏ hơn một ngưỡng dung sai cho trước. Ở mỗi vòng lặp, cập nhật các tham số để giống với phân phối dữ liệu thực sự.

**Bảng 2.1. Quá trình cập nhật tham số**









Bảng 2.1 mô tả kết quả quá trình thực hiện cập nhật tham số để thu được kết quả phân cụm giống với phân phối ban đầu như hình Iteration 24.

Mở rộng đối với dữ liệu nhiều chiều ( $D > 1$ ), chỉ có một số thay đổi nhỏ. Thay vì ước lượng trung bình và phương sai cho mỗi phân phối Gaussian thì ước lượng trung bình và ma trận

hiệp phương sai. Ma trận hiệp phương sai là một ma trận vuông có hình dạng  $(D, D)$  trong đó  $D$  đại diện cho số chiều của dữ liệu. Tham khảo ví dụ một tập dữ liệu 2-D được sử dụng để điều chỉnh một số lượng hỗn hợp của các phân phối Gaussian khác nhau tại [1].

## **2.2. Ứng dụng Gaussian Mixture Model phát hiện đối tượng chuyển động trong video**

Phương pháp sử dụng mô hình hỗn hợp Gaussian (Gaussian Mixture Model - GMM) để phát hiện vật thể chuyển động trong video được thực hiện thông qua các quá trình sau:

- 1) Video đầu vào được xử lý bằng thư viện OpenCV để trích xuất các khung hình.
- 2) Đối với mỗi vị trí điểm ảnh, giá trị pixel tại vị trí đó được lưu trữ qua tất cả các khung hình trong một mảng và được đưa vào hàm Gaussian\_Mixture để biểu diễn dưới dạng hỗn hợp của hai phân phối Gaussian.
- 3) Một phân phối Gaussian sẽ biểu diễn cho nền (background), và phân phối Gaussian còn lại sẽ biểu diễn cho vật thể nổi bật (foreground) tại vị trí điểm ảnh đó qua tất cả các khung hình.
- 4) Vì nền thường giữ nguyên trong hầu hết các khung hình, nên giá trị trung bình của phân phối Gaussian có trọng số lớn hơn và độ lệch chuẩn nhỏ hơn sẽ được sử dụng để xây dựng mô hình nền.
- 5) Để theo dõi vật thể, ta trừ mô hình nền này khỏi mỗi khung hình.

Cách tiếp cận này sử dụng GMM để mô hình hóa sự biến đổi giá trị pixel tại mỗi vị trí điểm ảnh qua các khung hình trong video. Bằng cách sử dụng hai thành phần Gaussian, nó phân tách được nền tĩnh và vật thể chuyển động. Nền được xây dựng từ thành phần Gaussian có trọng số cao nhất và độ lệch chuẩn thấp nhất. Sau đó, vật thể chuyển động có thể được phát hiện bằng cách trừ nền khỏi mỗi khung hình.

Phương pháp này hiệu quả trong việc phát hiện vật thể chuyển động ngay cả khi có sự thay đổi nhẹ về ánh sáng hoặc nhiễu trong video, vì GMM có khả năng mô hình hóa sự biến đổi này thông qua các phân phối Gaussian.

Thuật toán thực hiện việc phát hiện đối tượng chuyển động trong video dựa trên Gaussian Mixture Model như sau:

- Bước 1. Cho dữ liệu đầu vào là giá trị pixel của từng vị trí trên các khung hình trong video.
- Bước 2. Giả sử giá trị pixel tại mỗi vị trí được biểu diễn bằng hỗn hợp của hai phân phối Gaussian (một cho nền và một cho vật thể nổi bật).
- Bước 3. Khởi tạo các giá trị trung bình và phương sai ban đầu cho hai phân phối Gaussian.
- Bước 4. Tính mật độ xác suất (PDF - Probability Density Function) của từng quan sát (giá trị pixel) theo công thức phân phối Gaussian (2.2a).

$$f(x|\mu_k, \sigma_k^2) = \frac{1}{\sqrt{2\pi\sigma_k^2}} \exp\left(-\frac{(x - \mu_k)^2}{2\sigma_k^2}\right) \quad (2.2a)$$

- Bước 5. Tính xác suất để một quan sát thuộc về mỗi phân phối Gaussian theo công thức (2.2b):

$$b_k = \frac{f(x|\mu_k, \sigma_k^2)\phi_k}{\sum_{k=1}^K f(x|\mu_k, \sigma_k^2)\phi_k} \quad (2.2b)$$



Bước 6. Sử dụng định lý Bayes để tính xác suất hậu nghiệm (posterior probability) của mỗi Gaussian, cho biết xác suất mà quan sát được sinh ra từ phân phối đó.

Bước 7. Cập nhật các tham số (trung bình, phương sai, trọng số) cho các phân phối Gaussian dựa trên xác suất hậu nghiệm.

$$\mu_k = \frac{\sum b_k x}{\sum b_k}; \sigma_k^2 = \frac{\sum b_k (x - \mu_k)^2}{\sum b_k}; \phi_k = \frac{1}{N} \sum b_k \quad (2.2c)$$

Bước 8. Lặp lại quá trình cho đến khi các giá trị trung bình, phương sai và trọng số hội tụ.

### **3. Thử nghiệm**

Phần này sẽ mô tả quá trình kiểm tra thực hiện, từ đó rút ra kết quả và nhận xét.

Toàn bộ mã nguồn, cơ sở dữ liệu, kết quả, đánh giá được lưu trữ tại repository:

[https://github.com/lephamcong/Group2\\_GMM.git](https://github.com/lephamcong/Group2_GMM.git)

#### **3.1. Cơ sở dữ liệu**

Dữ liệu được sử dụng là video được quay bởi camera giám sát giao thông, thông số video như sau: thời lượng 30s, kích thước khung hình 1920x1080pixels, tốc độ khung hình 25fps.

Đường dẫn đến dữ liệu: <https://drive.google.com/file/d/1BXUZeLSU-4lmNtu7ZnffGqG2RwKMkNBU/view?usp=sharing>

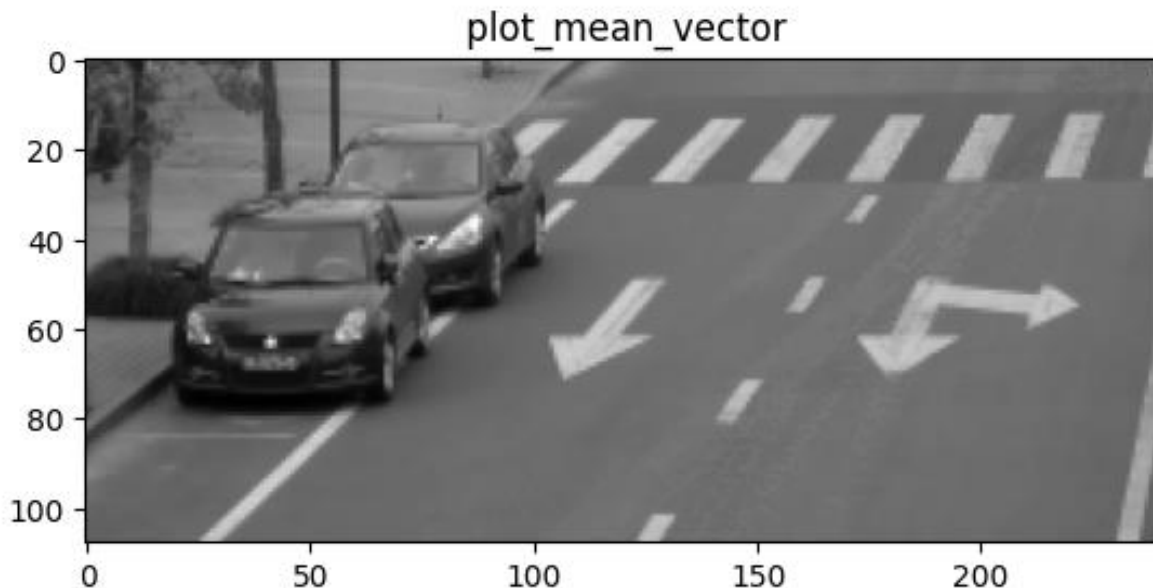
#### **3.2. Kết quả - nhận xét**

Kết quả khi thử nghiệm được mô tả trong mục 3.2.1 như bên dưới, và các nhận xét kết quả đạt được trong mục 3.2.2.

##### **3.2.1. Kết quả**

Sau khi thử nghiệm với Video 2 (đường dẫn ở trên), thì thu được một số kết quả sau:







1) Ảnh nền:



Hình : Ảnh nền

2) Một số frame sau khi thực hiện phép trừ nền (tất cả frame được lưu trữ lại repository ở trên):

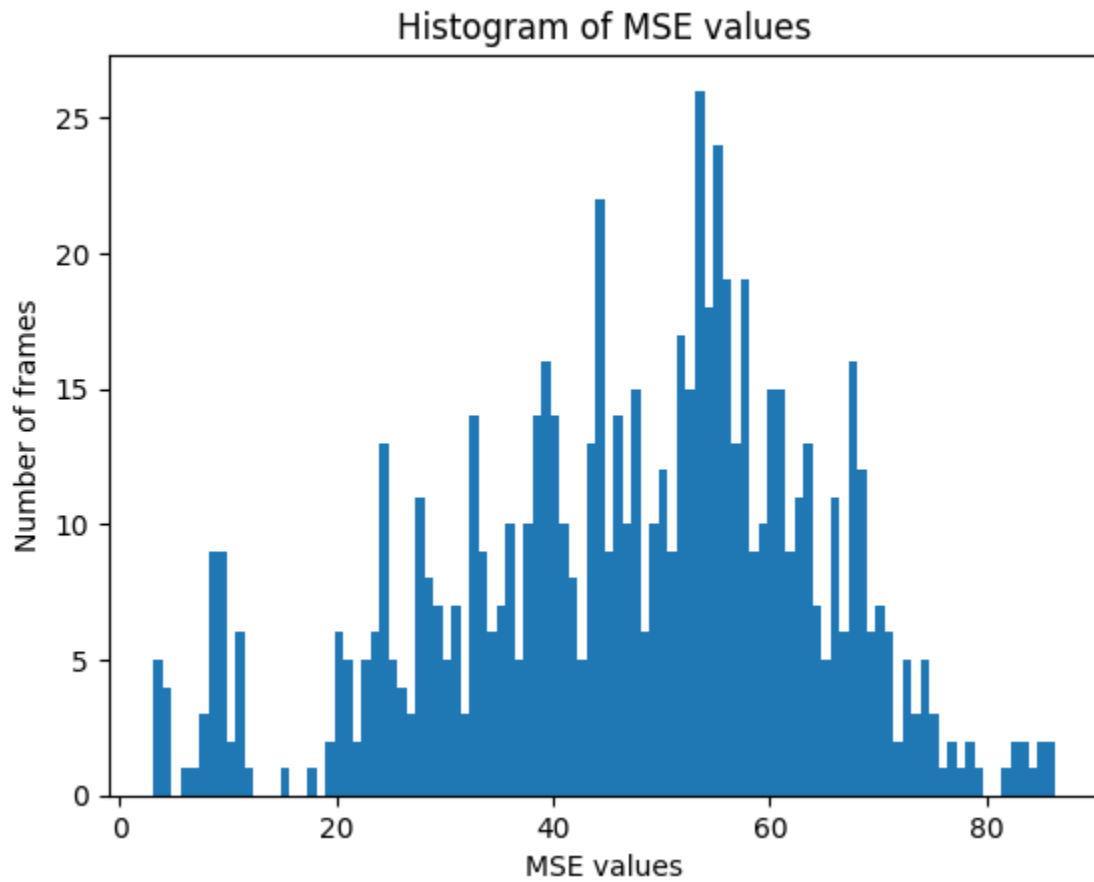
**Bảng 3.2a. Kết quả của phép trừ nền trên một số frame**

Frame	Kết quả của phép trừ nền
	
	
	

3) Video kết quả: [https://youtu.be/2Ktw\\_daaETQ](https://youtu.be/2Ktw_daaETQ)

4) Độ chính xác: Các khung hình đầu ra được tính toán bằng thuật toán được so sánh với các khung hình đầu ra thu được sau khi sử dụng hàm BackgroundSubtractorMOG2 dựng sẵn trong OpenCV2. Số liệu được sử dụng cho so sánh là MSE[2]. Các khung hình từ thuật toán được xây dựng chỉ được chuyển đổi sang màu đen và trắng cho so sánh công bằng vì hàm dựng sẵn chỉ cung cấp các khung đầu ra có màu đen và trắng. Giá trị lớn nhất RMSE thu được là 86,37 và RMSE tối thiểu thu được là 3.17

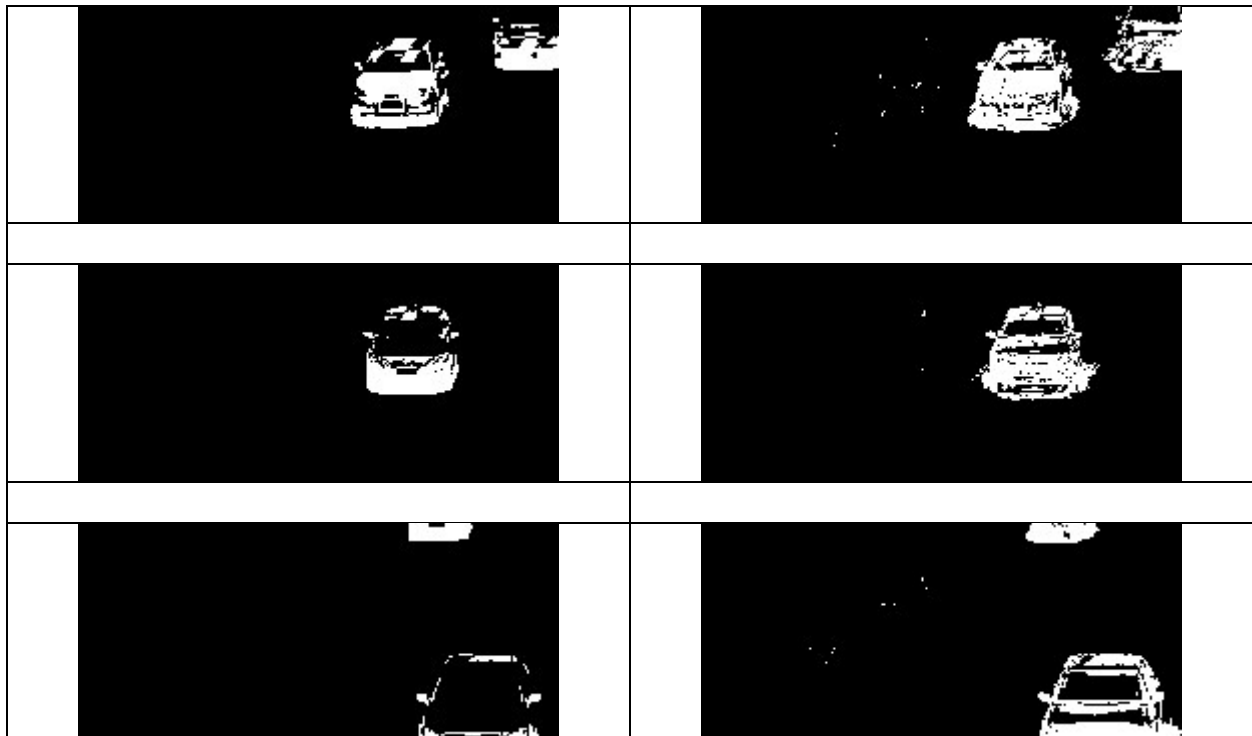
5) Histogram của MSE:



6) So sánh với thuật toán được xây dựng và sử dụng thư viện Opencv:

**Bảng 3.2b. So sánh kết quả phép trừ nền giữa thuật toán được xây dựng và hàm có sẵn trong thư viện**

Đầu ra của thuật toán		Đầu ra với hàm có sẵn trong thư viện	



### **3.2.2. Nhận xét**

Đầu tiên, thuật toán đã xác định đúng ảnh nền, đây là bước quan trọng đảm bảo phép trừ nền thực hiện chính xác hay không, nếu không xác định đúng nền, có thể dẫn đến kết quả cuối cùng không chính xác.

Thứ hai, phép trừ nền mang lại kết quả tương đối, một số chi tiết bị mất, nguyên nhân là do một số chi tiết có màu trùng với màu nền hoặc chất lượng video đầu vào không tốt vẫn còn nhiều nên các chi tiết đó bị trùng với nền.

Thứ ba, kết quả trong video đầu ra là các đối tượng được cho là đang chuyển động trong video, dựa vào video kết quả có thể nhận thấy, thuật toán đã detect thành công các đối tượng chuyển động trong video với sự hỗ trợ của Gaussian Mixture Model

Thứ tư, giá trị lớn nhất RMSE thu được là 86,37 và RMSE tối thiểu thu được là 3.17. Giá trị MSE cao chỉ ra rằng có một sự chênh lệch đáng kể giữa hình ảnh dự đoán và hình ảnh thực tế trong một số trường hợp cụ thể, ở đây là giữa việc sử dụng thuật toán tự xây dựng và thư viện hỗ trợ. Giá trị MSE thấp cho thấy thuật toán hoạt động tốt trong một số khung hình, với sự chênh lệch ít giữa hình ảnh dự đoán và hình ảnh thực tế. Điều này có thể xảy ra trong các điều kiện lý tưởng nơi nền tương đối đơn giản và ít nhiễu, hoặc khi các đối tượng chuyển động có đường nét rõ ràng. Thông qua việc xem xét chi tiết các trường hợp khi RMSE đạt giá trị cao nhất và thấp nhất để tìm ra cách để tối ưu hóa thuật toán và cải thiện độ chính xác của nó trong các tình huống khác nhau.

#### **4. Kết luận**

Việc sử dụng mô hình Gaussian Mixture (GMM) để phát hiện đối tượng chuyển động trong video là một phương pháp phổ biến và có hiệu quả, nhất là trong các ứng dụng giám sát video và an ninh. Dưới đây là một số điểm mạnh và hạn chế của phương pháp này, cùng với một số kết luận về hiệu quả của nó:

**Điểm Mạnh:**

- 1) Phân biệt Nền và Đối tượng: GMM giỏi trong việc phân biệt giữa nền tĩnh và đối tượng chuyển động bằng cách mô hình hóa mỗi điểm ảnh như một hỗn hợp của nhiều phân phối Gaussian. Điều này cho phép nó phát hiện chuyển động ngay cả trong các điều kiện ánh sáng phức tạp.
- 2) Cập nhật Động: Mô hình có thể được cập nhật theo thời gian để thích ứng với các thay đổi trong cảnh, như sự thay đổi của điều kiện ánh sáng hoặc biến động nền mới.
- 3) Hiệu quả với Nhiều Đối tượng: GMM có khả năng phát hiện nhiều đối tượng chuyển động cùng một lúc, điều này là hữu ích cho các ứng dụng giám sát đông đúc.

**Hạn Chế:**

- 1) Nhạy cảm với Nhiễu: Mặc dù GMM có thể thích ứng với các thay đổi trong video, nó vẫn có thể nhạy cảm với nhiễu, nhất là trong điều kiện ánh sáng yếu hoặc khi có sự chuyển động của bụi hoặc lá cây.
- 2) Tính toán Tốn kém: Việc ước tính và cập nhật các tham số Gaussian cho mỗi điểm ảnh có thể rất tốn kém về mặt tính toán, đặc biệt là với video độ phân giải cao và trong thời gian thực.
- 3) Thiếu Ổn định trong Môi trường Động: Trong các cảnh có sự thay đổi nền liên tục, GMM có thể mất nhiều thời gian để thích ứng, dẫn đến các phát hiện sai hoặc bỏ sót.

Tóm lại, mô hình Gaussian Mixture là một công cụ mạnh mẽ để phát hiện đối tượng chuyển động trong video, nhờ khả năng mô hình hóa chính xác và thích ứng với sự thay đổi của nền. Tuy nhiên, tính hiệu quả của nó có thể bị ảnh hưởng trong các điều kiện nhiễu hoặc động. Để tối đa hóa hiệu quả, các nhà phát triển và nhà nghiên cứu nên cân nhắc kết hợp GMM với các kỹ thuật khác như bộ lọc nhiễu, phân tích điểm khóa, hoặc sử dụng nó trong một khuôn khổ học sâu nâng cao để giải quyết các hạn chế về nhiễu và tốn kém tính toán. Các cải tiến này có thể giúp GMM trở thành một giải pháp hiệu quả hơn cho các ứng dụng giám sát video và an ninh hiện đại.

#### **5. Tài liệu tham khảo**

- [1] <https://towardsdatascience.com/how-to-code-gaussian-mixture-models-from-scratch-in-python-9e7975df5252>
- [2] <https://statisticsbyjim.com/regression/mean-squared-error-mse/>
- [3] <https://medium.com/@prantiksen4/background-extraction-from-videos-using-gaussian-mixture-models-6e11d743f932>