

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  
**KHOA CÔNG NGHỆ THÔNG TIN**

---



# **ĐỒ ÁN** **TỐT NGHIỆP ĐẠI HỌC**

## **NGHIÊN CỨU MỘT SỐ KỸ THUẬT** **PHÂN ĐOẠN CHUYỂN ĐỘNG VIDEO**

Sinh viên thực hiện : **Bùi Duy Hùng**  
Lớp : **ĐH KHMT 1 – K3**  
Giáo viên hướng dẫn : **Ths. Ngô Đức Vĩnh**  
Cán bộ phản biện : **Ths. Trần Thanh Hùng**

--- Hà Nội, tháng 6 – 2012 ---

[illegible]

**Giáo viên hướng dẫn**

## This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

### **Giáo viên phản biện**

## LỜI NÓI ĐẦU

Thị giác máy tính (Computer Vision) được khởi đầu từ năm 70 của thế kỷ trước, cùng với sự phát triển không ngừng của tốc độ xử lý máy tính là động lực quan trọng cho lĩnh vực này. Khi sức mạnh máy tính được cải thiện, công nghệ thị giác máy tính đã trở nên khả thi ứng dụng cho nhiều lĩnh vực.

Một trong những ứng dụng nổi bật của lĩnh vực thị giác máy tính là bài toán phân đoạn chuyển động video, đã có những thành công lớn góp phần vào sự phát triển của ngành công nghiệp nội dung, công nghiệp điện tử - tiêu dùng, cũng như trong quốc phòng và an ninh...

Xuất phát từ đó, tôi quyết định lựa chọn đề tài **“Nghiên cứu một số kỹ thuật phân đoạn chuyển động video”**. Mong muốn của tôi là được nghiên cứu những kỹ thuật tiên tiến trong lĩnh vực thị giác máy tính – ứng dụng trong bài toán phân đoạn chuyển động video và phát triển một số kỹ thuật khác nhằm đóng góp một phần làm cho lĩnh vực này trở nên phong phú và phát triển hiệu quả hơn nữa.

Lời đầu, tôi muốn gửi những lời cảm ơn chân thành đến Thạc sĩ Ngô Đức Vĩnh – trưởng khoa Công nghệ thông tin trường ĐH Công nghiệp Hà Nội đã giúp tôi định hướng đề tài, lựa chọn hướng nghiên cứu, thường xuyên đánh giá và động viên tôi trong suốt thời gian thực hiện đồ án. Tôi cũng biết ơn sự chỉ dẫn của các anh Nguyễn Duy Hoàng, Bùi Việt Hưng – Trung tâm Chỉ huy và điều khiển, Viện nghiên cứu và phát triển Viettel trong suốt thời gian thực tập, từ đó giúp tôi hiểu rõ và sâu sắc hơn về một số bộ lọc và bài toán phân tách, theo vết đa đối tượng chuyển động.

Và không thể không cảm ơn sự cộng tác hiệu quả của các bạn Phạm Việt, Lê Văn Hình, Nguyễn Hồng Gấm trong suốt quá trình thực hiện đề tài nghiên cứu khoa học – làm tiền đề cho quá trình hoàn tất đồ án trên.

Sau cùng, tôi muốn gửi tặng tất cả những điều tốt đẹp đến gia đình, đến những người thân yêu luôn đứng sau là chỗ dựa tinh thần, là niềm tin cũng như nguồn động lực lớn lao trong suốt quá trình thực hiện và hoàn tất đồ án cũng như suốt quãng thời gian mang trên mình niềm tự hào của sinh viên Đại học Công nghiệp Hà Nội.

## TÓM TẮT ĐỒ ÁN

Phát hiện và phân đoạn chuyển động của đối tượng là một trong những nhiệm vụ quan trọng và cần thiết của công nghệ thị giác máy tính [17]. Phân đoạn chuyển động video là một kỹ thuật cơ bản cho việc phân tích một chuỗi hình ảnh của hoạt cảnh thực tế được thu nhận lại [51]

Trong hệ thống giám sát, an ninh, các thiết bị ghi nhận hình ảnh thường được đặt cố định và hạn chế dịch chuyển. Do đó, các kỹ thuật phân đoạn chuyển động thực hiện quá trình phân tách các đối tượng chuyển động, bỏ qua việc lưu trữ thông tin về không gian không có sự thay đổi giữa nhiều khung hình liên tiếp.

Ngoài việc nghiên cứu tổng quan các kỹ thuật phân đoạn chuyển động video, tôi lựa chọn việc mở rộng nghiên cứu và ứng dụng vào các bài toán giám sát cơ bản, trong đó các đối tượng di chuyển với quỹ đạo không quá phức tạp. Trong bài toán cụ thể đó việc theo dõi đối tượng sử dụng kỹ thuật luồng quang học có thể được áp dụng triệt để.

Tuy nhiên, giả thuyết đặt ra của kỹ thuật luồng quang học là cường độ sáng của các điểm ảnh trong một đối tượng xác định trong nhiều khung hình liên tiếp là không thay đổi. Trên thực tế, cường độ sáng của một đối tượng bị ảnh hưởng bởi nhiều yếu tố và không đảm bảo điều kiện lý tưởng cho kỹ thuật luồng quang học thực thi một cách hiệu quả. Vì vậy, **cải tiến thứ nhất**, tôi đề xuất kỹ thuật lọc lỗi trung vị nhằm cải tiến kỹ thuật luồng quang học.

Đồng thời tôi cũng sử dụng kỹ thuật ước lượng nền dựa trên phân bố Gaussian, cụ thể hơn là mô hình hỗn hợp Gaussian áp dụng vào bài toán phân đoạn chuyển động với camera tĩnh. Sau đó, tôi đề xuất mô hình phân đoạn chuyển động cơ bản, với **cải tiến thứ hai** – tôi áp dụng bộ lọc Kalman cho quá trình theo vết các vùng chuyển động.

Kết quả của các đề xuất được đánh giá thông qua thực nghiệm vào bài toán giám sát thực tế (giám sát phương tiện giao thông, giám sát sân trường khu A – trường ĐH Công nghiệp Hà Nội), cho thấy việc đề xuất của tôi có những ưu điểm: đã loại bỏ lỗi của kỹ thuật luồng quang học trong một số môi trường nhất định, bộ lọc Kalman giúp theo dõi quỹ đạo của các vùng chuyển động chính xác hơn. Đồng thời cũng bộc lộ nhược điểm: Độ phức tạp cao vì phải thực hiện hai lần các phép tính toán của luồng quang học.

Kết quả thực nghiệm cho thấy, có thể mở rộng nghiên cứu theo các hướng đề xuất.

## ABSTRACT

Motion detection and segmentation is one of the most important and essential task of computer vision technology. Motion segmentation in the video is the basic technique for image sequences analysis obtained from actual scene.

In surveillance system, security, image capturing devices put fixedly and restrict motion. So, motion segmentation techniques segment moving objects, skip archiving information of the non-change regions between sequences frames.

Beside the study about general approaches of motion segmentation in the video, I extend in applying to surveillance which has objects moving with the non-complex orbit. Actually, in this case, optical flow can be applied effectively.

However, the hypothesis for optical flow can be applied is that the brightness of pixels of moving objects are not change between a sequence of frames. In fact, the brightness of pixels of an object obtained from camera is affected by many factors and does not guarantee ideal conditions for optical flow implemented effectively. So, my **first proposal** is median filtering process to reduce errors of optical flow when working at non-ideal conditions.

Simultaneously, I apply foreground estimation based on gaussian mixture models for motion segmentation with fixed camera. After that, I propose the model of basic motion segmentation process with my **second proposal** – applying Kalman filter in regions tracking.

Results of my proposals judged through experiments in actual surveillance problem (vehicle surveillance, surveillance in school yard – Hanoi university of industry), showed that proposals have advantages: reduced the error of optical flow in some conditions, Kalman filter make regions tracking process to be more exactly. Beside the disadvantage that the complexities of my proposal is increased because the optical flow calculation processed twice.

The experimental results show that the study can expanded under the direction proposed.

# MỤC LỤC

<b>LỜI NÓI ĐẦU .....</b>	<b>1</b>
<b>TÓM TẮT ĐỒ ÁN.....</b>	<b>1</b>
<b>ABSTRACT .....</b>	<b>1</b>
<b>Chương 1: MỞ ĐẦU .....</b>	<b>1</b>
1.1. Lý do chọn đề tài.....	1
1.2. Phương pháp và mục tiêu nghiên cứu .....	1
1.2.1. Mục tiêu nghiên cứu .....	1
1.2.2. Phương pháp nghiên cứu .....	1
1.2.3. Đối tượng nghiên cứu .....	2
1.2.4. Dự kiến kết quả nghiên cứu.....	2
1.3. Cấu trúc báo cáo.....	2
<b>Chương 2: NỘI DUNG NGHIÊN CỨU.....</b>	<b>3</b>
2.1. Tổng quan về lĩnh vực nghiên cứu.....	3
2.1.1. Một số nghiên cứu hiện tại .....	4
2.1.1.1. Một số đặc điểm chính của phân đoạn chuyển động.....	4
2.1.1.2. Một số hướng tiếp cận chính .....	5
2.1.2. Một số kỹ thuật phân đoạn đối tượng chuyển động nổi bật .....	7
2.1.3. Vai trò và ứng dụng của kỹ thuật phân đoạn chuyển động .....	11
2.2. Phương pháp tiếp cận thứ nhất – Dựa trên sự sai biệt ảnh.....	12
2.2.1. Tổng quan về kỹ thuật trừ nền.....	12
2.2.2. Trừ nền đơn giản.....	15
2.2.3. Trừ nền trung vị và nền trung bình.....	17
2.2.3.1. Trừ nền trung vị.....	17
2.2.3.2. Trừ nền trung bình.....	19
2.2.3.3. Cập nhật nền .....	20
2.2.3.4. Đánh giá.....	21
2.3. Phương pháp tiếp cận thứ hai – Dựa trên kỹ thuật luồng quang học .....	22
2.3.1. Tổng quan về luồng quang học.....	22
2.3.2. Trường chuyển động.....	22

2.3.3. Phương trình luồng quang học .....	23
2.3.4. Đánh giá.....	25
2.3.5. Áp dụng luồng quang học trong phân đoạn chuyển động video .....	26
2.4. Phương pháp tiếp cận thứ ba – Dựa trên mô hình thống kê.....	27
2.4.1. Mô hình hỗn hợp Gaussians .....	27
2.4.1.1. Mô hình Gaussian đơn.....	27
2.4.1.2. Mô hình hỗn hợp Gaussian.....	28
2.4.1.3. Áp dụng GMM cho bài toán ước lượng nền .....	29
2.4.2. Mô hình RANSAC .....	30
2.4.2.1. Tổng quan về mô hình RANSAC.....	30
2.4.2.2. RANSAC là gì? .....	31
2.4.2.3. Áp dụng RANSAC cho bài toán phân đoạn chuyển động .....	34
2.5. Một số kiến thức nền tảng hỗ trợ cho quá trình nghiên cứu .....	37
2.5.1. Một số bộ lọc trong xử lý tín hiệu số.....	37
2.5.2. Tổng quan về các kỹ thuật lọc .....	37
2.5.2.1. Mô hình không gian trạng thái rời rạc theo thời gian.....	38
2.5.2.2. Ước lượng trạng thái tuyến tính .....	38
2.5.2.3. Tiêu chí đánh giá các bộ lọc .....	39
2.5.3. Bộ lọc Kalman .....	39
2.5.3.1. Bản chất của bộ lọc Kalman .....	40
2.5.3.2. Phương trình Kalman rời rạc .....	40
2.5.3.3. Thuật toán bộ lọc Kalman rời rạc .....	42
2.5.3.4. Đánh giá lọc Kalman .....	43
2.5.4. Kỹ thuật tương quan dữ liệu .....	43
2.5.4.1. Vai trò của tương quan dữ liệu trong bài toán theo dõi đa đối tượng chuyển động .....	43
2.5.4.2. Kỹ thuật tương quan dữ liệu ứng dụng trong phân tách đa đối tượng chuyển động. ....	45
2.5.4.3. Thuật toán Hungary .....	46
<b>Chương 3: Một số đề xuất và các bước thực hiện .....</b>	<b>50</b>
3.1. Đề xuất thứ nhất – cải tiến kỹ thuật luồng quang học.....	50
3.1.1. Tổng quan kỹ thuật cải tiến luồng quang học.....	50



3.1.2. Tính lỗi và lọc .....	51
3.1.3. Tính luồng quang học tiến và tính ngược trở lại .....	51
3.1.4. So khớp mẫu .....	52
3.1.5. Lọc trung vị.....	53
3.1.6. Tính vị trí và kích thước thay đổi của đối tượng .....	53
3.2. Đề xuất thứ hai - Mô hình chung của bài toán thực hiện .....	56
3.2.1. Phát hiện đối tượng chuyển động .....	56
3.2.2. Sử dụng hỗn hợp Gaussian để ước lượng nền cho bài toán ước lượng chuyển động .....	57
3.2.3. Áp dụng bộ lọc kết hợp cùng thuật toán cải tiến luồng quang học cho bài toán theo vết các vùng chuyển động .....	58
3.1.3.1. Mô hình hệ thống.....	58
3.1.3.2. Mô hình của bộ lọc Kalman .....	58
3.2.4. Áp dụng thuật toán Hungary giải quyết bài toán tương quan dữ liệu ....	60
<b>Chương 4: Thực nghiệm và kết luận.....</b>	<b>61</b>
4.1. Điều kiện thực nghiệm .....	61
4.1.1. Môi trường .....	61
4.1.2. Thiết bị.....	61
4.1.3. Cấu hình máy .....	61
4.1.4. Khoảng cách thực nghiệm .....	61
4.1.5. Môi trường triển khai.....	61
4.2. Thực nghiệm kỹ thuật phân đoạn video .....	61
4.2.1. Phương pháp thực nghiệm.....	61
4.2.2. Kết quả thực nghiệm.....	62
4.3. Kết luận .....	64
4.4. Hướng nghiên cứu.....	64
<b>TÀI LIỆU THAM KHẢO.....</b>	<b>65</b>

# DANH MỤC HÌNH ẢNH

Figure 1: Quy trình chung của bài toán thị giác máy tính.....	3
Figure 2: Bảng tổng hợp một số kỹ thuật phân đoạn chuyển động nổi bật từ 2005-2008 ..	6
Figure 3: Mô hình của A. Colombari và cộng sự đề xuất .....	7
Figure 4: Mô hình của Aessandro Verri và cộng sự đề xuất .....	8
Figure 5: Áp dụng GPCA để giảm thiểu số chiều của không gian con.....	9
Figure 6: Áp dụng kỹ thuật học không giám sát cho bài toán phân đoạn chuyển động.....	9
Figure 7: Thuật toán đề xuất của Bing Han.....	10
Figure 8: Kỹ thuật đề xuất của A. Colombari .....	10
Figure 9: Kỹ thuật phân đoạn chuyển động video ứng dụng theo dõi phương tiện giao thông .....	11
Figure 10: Mô hình chung của phương pháp tiếp cận dựa trên sai biệt .....	13
Figure 11: Một ví dụ về kỹ thuật trừ ảnh.....	14
Figure 12: Ví dụ về làm trơn ảnh sử dụng bộ lọc ảnh .....	14
Figure 13: Ví dụ về kỹ thuật khoanh vùng đối tượng chuyển động.....	15
Figure 14: Ảnh mô phỏng thuật toán trừ nền đơn giản .....	16
Figure 15: Kỹ thuật trừ nền trước bộc lộ nhược điểm.....	16
Figure 16: Ví dụ về kết quả của kỹ thuật trừ nền đơn giản .....	17
Figure 17: Ví dụ về kết quả của kỹ thuật trừ nền trung vị.....	18
Figure 18: Ví dụ về kết quả của kỹ thuật trừ nền trung bình .....	20
Figure 19: Trường chuyển động trong kỹ thuật luồng quang học.....	22
Figure 20: Phương trình Optical flow biểu diễn trên hệ trục tọa độ .....	24
Figure 21: Ứng dụng luồng quang học vào phân đoạn chuyển động.....	26
Figure 22 Đồ thị hàm mật độ Gauss.....	27
Figure 23 Mô hình GMM.....	28
Figure 24: Ví dụ 1 về bài toán áp dụng RANSAC.....	31
Figure 25: Ví dụ 1 về bài toán lọc nhiễu mô hình bằng RANSAC.....	32
Figure 26: Kết quả ví dụ về bài toán lọc nhiễu mô hình bằng RANSAC .....	32
Figure 27: Kết quả của ví dụ về bài toán lọc nhiễu mô hình bằng RANSAC.....	34
Figure 28: Trường chuyển động khi ô-tô rẽ trái.....	34
Figure 29: Ví dụ véc-tơ chuyển động được trích ra khi ô-tô rẽ trái .....	36
Figure 30: Kết quả lọc sử dụng RANSAC cho ví dụ véc-tơ chuyển động được trích ra khi ô-tô rẽ trái .....	36
Figure 31: Ứng dụng lọc Kalman để lọc tín hiệu thu được từ tọa độ con trỏ của người dùng .....	39

Figure 32: Mô hình hoạt động của mạch lọc Kalman .....	40
Figure 33: Hình trên tóm tắt cơ bản của mạch lọc Kalman.....	43
Figure 34: Hình ảnh minh họa theo dõi 2 đối tượng chuyển động bằng kỹ thuật xử lý ảnh và bộ lọc.....	44
Figure 35: Hình ảnh minh họa theo dõi 2 đối tượng chuyển động bằng kỹ thuật tương quan dữ liệu. ....	45
Figure 36: Mô tả các khả năng ghép đối tượng .....	45
Figure 37: Trình tự thực hiện .....	50
Figure 38: Các bước lọc và tính lỗi .....	51
Figure 39: Ví dụ về tính luồng quang học xuôi - ngược .....	51
Figure 40: Lọc trung vị.....	53
Figure 41: Ví dụ áp dụng luồng quang học theo dõi chuyển động của rubik .....	54
Figure 42: Ví dụ tính kích thước mới của đối tượng dùng cải tiến của luồng quang học.	54
Figure 43: Mô hình đề xuất của kỹ thuật ước lượng chuyển động sử dụng GMM.....	57
Figure 44: Mô hình đề xuất sử dụng cải tiến optical flow và lọc Kalman cho bài toán phân đoạn chuyển động .....	58
Figure 45: Mô hình đề xuất sử dụng thuật toán Hungary cho bài toán tương quan.....	60
Figure 46: Thuật toán đề xuất hoạt động kém hiệu quả khi camera bị dịch chuyển.....	62
Figure 47: Ước lượng nền sử dụng GMM hoạt động kém hiệu quả khi camera bị dịch chuyển.....	63

## **DANH MỤC BẢNG BIỂU**

Bảng 1: Xác suất ghép cặp đối tượng	46
Bảng 2: Bảng kết quả ghép cặp	46
Bảng 3: Bảng phân công m công việc cho n công nhân	47
Bảng 4: Biểu diễn ma trận của thuật toán Hungary	47
Bảng 5: Kết quả phân công cho bài toán giao m công việc cho n công nhân	48

## DANH MỤC CÁC TỪ VIẾT TẮT

STT	TỪ VIẾT TẮT	ĐẦY ĐỦ
1.	KF	Kalman Filter – Bộ lọc Kalman
2.	EKF	Extended Kalman Filter – Bộ lọc Kalman mở rộng
3.	UKF	Unscented Kalman Filter – Bộ lọc Unscented Kalman
4.	PF	Particle Filter
5.	IMM	Interactive Multiple Model – Mô hình lọc tương tác
6.	OpenCV	Thư viện xử lý ảnh mã nguồn mở Open Computer Vision library
7.	EmguCV	Thư viện xử lý ảnh OpenCV được đóng gói để lập trình trên môi trường .NET
8.	Filter	Bộ lọc nhiễu trong bài toán xử lý tín hiệu số
9.	GMM	Gaussian Mixture Model – mô hình hỗn hợp Gaussians

# Chương 1: MỞ ĐẦU

## 1.1. Lý do chọn đề tài

Thị giác máy tính (Computer Vision) được khởi đầu từ năm 70 của thế kỷ trước, cùng với sự phát triển không ngừng của tốc độ xử lý máy tính là động lực quan trọng cho lĩnh vực này. Khi sức mạnh máy tính được cải thiện, công nghệ thị giác máy tính đã trở nên khả thi ứng dụng cho nhiều lĩnh vực.

Một trong những ứng dụng nổi bật của lĩnh vực thị giác máy tính là bài toán phân đoạn chuyển động video, đã có những thành công lớn góp phần vào sự phát triển của ngành công nghiệp nội dung, công nghiệp điện tử - tiêu dùng, cũng như trong quốc phòng và an ninh...

Xuất phát từ đó, tôi quyết định lựa chọn đề tài **“Nghiên cứu một số kỹ thuật phân đoạn chuyển động video”**. Mong muốn của tôi là được nghiên cứu những kỹ thuật tiên tiến trong lĩnh vực thị giác máy tính – ứng dụng trong bài toán phân đoạn chuyển động video và phát triển một số kỹ thuật khác nhằm đóng góp một phần làm cho lĩnh vực này trở nên phong phú và phát triển hiệu quả hơn nữa.

## 1.2. Phương pháp và mục tiêu nghiên cứu

### 1.2.1. Mục tiêu nghiên cứu

- Nghiên cứu và đánh giá các kỹ thuật phân đoạn chuyển động video và các xu hướng nghiên cứu trong lĩnh vực phân đoạn chuyển động
- Nghiên cứu, đánh giá kỹ thuật luồng quang học đồng thời đưa ra cải tiến
- Đề xuất một số mô hình phân đoạn chuyển động cho bài toán giám sát với camera tĩnh
- Thử nghiệm và đưa ra kết luận.

### 1.2.2. Phương pháp nghiên cứu

- Tìm hiểu kiến thức về xử lý ảnh thông qua các giáo trình Xử lý ảnh, các bài viết về xử lý ảnh trên các diễn đàn – song song với việc tham khảo các tài liệu do giáo viên hướng dẫn cung cấp.
- Tham khảo các mã nguồn mở về xử lý ảnh trên các website như [codeproject.com](http://codeproject.com), [codeplex.com](http://codeplex.com), [sourceforge.net](http://sourceforge.net), [github.com](http://github.com)... Nhằm mục đích tìm hiểu quá trình cài đặt các thuật toán xử lý ảnh cơ bản và nâng cao trên ngôn ngữ lập trình C/C++ và C#
- Nghiên cứu dựa trên các bài báo, tạp chí khoa học về thị giác máy tính từ đó xây dựng góc nhìn tổng quan về lĩnh vực đang nghiên cứu.

- Nghiên cứu về các bộ lọc ứng dụng dựa trên các thư viện của một số trường đại học, một số viện nghiên cứu và các phòng thí nghiệm.
- Từng bước nghiên cứu, đánh giá và đề xuất các kỹ thuật cải thiện cho nội dung nghiên cứu.
- Thực nghiệm dựa trên xây dựng ứng dụng (sử dụng thư viện xử lý ảnh nguồn mở OpenCV) và đánh giá các kết quả nghiên cứu.

### **1.2.3. Đối tượng nghiên cứu**

- Tổng hợp các nghiên cứu của kỹ thuật phát hiện đối tượng chuyển động trong video từ năm 1989 đến 2011
- Một số mô hình thống kê cho kỹ thuật phân đoạn chuyển động (RANSAC)
- Kỹ thuật trừ ảnh và ứng dụng
- Mô hình hỗn hợp Gaussian cho bài toán ước lượng nền
- Một số bộ lọc trong xử lý tín hiệu số (lọc Kalman, lọc Particle)
- Nghiên cứu kỹ thuật luồng quang học và đánh giá, đồng thời áp dụng tư tưởng tính luồng quang học theo chiều ngược lại để xác định lỗi.
- Nghiên cứu một số kỹ thuật tìm thành phần liên thông trên ảnh nhị phân, đồng thời đề xuất áp dụng giải thuật tìm thành phần liên thông với độ phức tạp  $O(N^2)$
- Nghiên cứu ứng dụng bộ lọc trong bài toán theo dõi đa đối tượng chuyển động
- Cải tiến và ứng dụng kỹ thuật học trực tiếp cho bài toán gán nhãn đối tượng.

### **1.2.4. Dự kiến kết quả nghiên cứu**

- Nghiên cứu một vài ứng dụng của bộ lọc trong kỹ thuật phân đoạn chuyển động.
- Đề xuất một kỹ thuật cải tiến: Áp dụng kỹ thuật lọc lỗi trung vị nhằm cải tiến kỹ thuật luồng quang học.
- Đề xuất áp dụng thuật toán tìm thành phần liên thông trên ảnh nhị phân thay cho phương pháp loang thông thường trong một số nghiên cứu trước đó.
- Thực nghiệm trong hệ thống theo dõi phương tiện giao thông và theo dõi sân trường ĐH Công nghiệp Hà Nội.
- Đánh giá mô hình đề xuất và đưa ra kết luận

## **1.3. Cấu trúc báo cáo**

**Chương 1:** Tổng quan về đề tài

**Chương 2:** Trình bày về cơ sở lý thuyết của đề tài, các kỹ thuật hiện tại và những tồn tại.

**Chương 3:** Một số đề xuất và các bước thực hiện

**Chương 4:** Thực nghiệm và kết luận

## Chương 2: NỘI DUNG NGHIÊN CỨU

Trong phần này, tôi đề cập đến những nền tảng của nghiên cứu, trong đó đề cập đến bốn nội dung nghiên cứu chính:

- **Khái quát chung về kỹ thuật phân đoạn chuyển động:** Đưa ra một số nghiên cứu hiện tại, tổng quan về các hướng nghiên cứu chính cũng như một số nghiên cứu nổi bật.
- **Phân đoạn chuyển động tiếp cận dựa trên sự sai biệt ảnh giữa nhiều thời điểm khác nhau:** Cụ thể hơn, tôi đưa ra nền tảng của kỹ thuật trừ ảnh trước, trừ ảnh trung vị, trừ ảnh trung bình. Sau cùng tôi trình bày về mô hình hỗn hợp Gauss.
- Phân đoạn chuyển động tiếp cận dựa trên luồng quang học
- Phân đoạn chuyển động dựa trên các mô hình thống kê: GMM, RANSAC
- Một số bộ lọc trong xử lý tín hiệu số: Bộ lọc trong xử lý tín hiệu số kết hợp cùng một số thuật toán theo vết đối tượng chuyển động như MeanShift, CAMShift, luồng quang học, ... Là công cụ đắc lực cho việc theo sát các vùng chuyển động.

### 2.1. Tổng quan về lĩnh vực nghiên cứu

Trong lĩnh vực thị giác máy tính, phân đoạn ảnh nhằm mục đích phân chia một bức ảnh số thành nhiều vùng (mỗi vùng bao gồm một tập các điểm ảnh) có ý nghĩa khác nhau cho từng ứng dụng cụ thể.

Quá trình xử lý trên video có đặc trưng khác đôi chút so với việc xử lý trên ảnh thông thường là chúng ta luôn phải quan tâm đến một chuỗi hình ảnh, thay vì một ảnh. Tuy nhiên, về mặt bản chất, xử lý video cũng tương tự với xử lý từng ảnh, phân đoạn ảnh cũng vậy.

Với phân đoạn ảnh, mục tiêu là đơn giản hoá việc biểu diễn một bức ảnh thành một số thành phần có ý nghĩa hơn và dễ dàng hơn cho việc phân tích, lưu trữ, truyền tải, mô tả bức ảnh... Thông thường, việc phân đoạn ảnh được coi là bước tiền xử lý cho các bài toán thị giác máy tính. Quy trình thường được mô tả như sau:

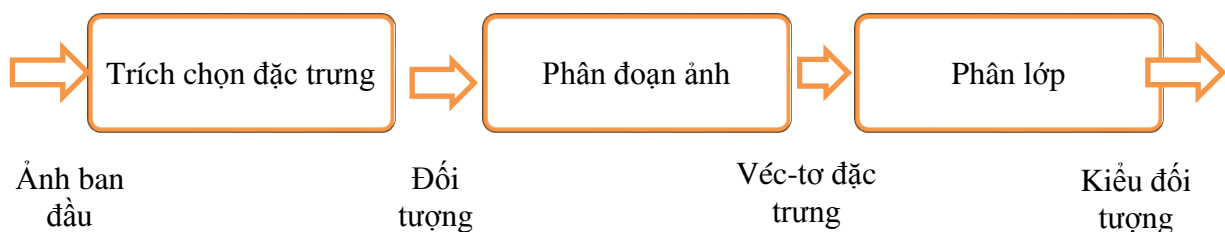


Figure 1: Quy trình chung của bài toán thị giác máy tính



Kết quả của quá trình phân đoạn ảnh là một tập các vùng có đủ khả năng biểu diễn lại thông tin của toàn bộ tấm ảnh, hoặc mặt khác kết quả của quá trình phân đoạn ảnh có thể là một tập các đường biên được trích xuất ra. Từng điểm ảnh trong cùng một vùng được gán nhãn giống nhau. Từng điểm ảnh trong cùng một vùng có chung các đặc tính có thể là màu sắc, kết cấu, độ sâu hoặc chuyển động.

Vì vậy, phân đoạn ảnh thực sự hữu ích và cần thiết nhằm, ví dụ, để tái cấu trúc đối tượng trong không gian ba chiều, tự động chuẩn đoán các sai sót trong sản xuất công nghiệp (phát hiện sự khiếm khuyết của vật liệu) hoặc giảm dung lượng phục vụ cho truyền dẫn và lưu trữ dữ liệu.

Chức năng quan trọng nhất của hệ thống thị giác máy tính là nhằm để nhận dạng các đối tượng chuyển động trong khung cảnh, đặc biệt trong những năm gần đây, các hệ thống lái xe tự động, hệ thống hỗ trợ lái xe, chủ đề này nhận được rất nhiều sự quan tâm khắp nơi trên thế giới. Quá trình xử lý này thực sự đơn giản với việc các camera thu nhận hình ảnh được đặt cố định, vì chỉ dựa trên kỹ thuật đơn giản là so sánh các khung hình thu nhận được tại nhiều thời điểm. Tuy nhiên, bài toán với camera chuyển động, không cố định thực sự phức tạp, bởi khi camera dịch chuyển thì sự chuyển động của camera độc lập với sự chuyển động của đối tượng. [11]

### **2.1.1. Một số nghiên cứu hiện tại**

#### **2.1.1.1. Một số đặc điểm chính của phân đoạn chuyển động**

Để một phương pháp – thuật toán phân đoạn chuyển động được áp dụng, ta thường đánh giá các khả năng thực thi của thuật toán đó trong các điều kiện:

- **Nhiều:** Một trong những vấn đề chính là xuất hiện nhiều trong ảnh thu nhận được. Trong một số ứng dụng nhất định, nhiều có thể ảnh hưởng trực tiếp đến hiệu quả thuật toán.
- **Dữ liệu không đầy đủ:** Sai sót trong quá trình thu nhận/truyền tải/tiền xử lý dữ liệu có thể gây ra sự mất mát dữ liệu trên ảnh. Từ đó, với dữ liệu không đầy đủ, hoặc không trọn vẹn (bị chèn thêm watermark, bị ngược sáng, ...), đòi hỏi thêm khắt khe các phương pháp phân đoạn.
- **Sự thiếu sót các tri thức tiên nghiệm:** Không phải lúc nào chúng ta cũng có trước tri thức tiên nghiệm về các đối tượng hoặc về số lượng các đối tượng đang hiện hữu. [9]

**Từ đó, ta có các đặc trưng chính của thuật toán phân đoạn chuyển động:**

- **Dựa trên đặc trưng hoặc dựa trên mật độ:** Trong phương pháp dựa trên đặc trưng, các đối tượng được biểu diễn bởi một số lượng giới hạn các điểm như các

góc, các điểm nổi bật, trong khi đó phương pháp dựa trên mật độ tính toán một tập các điểm ảnh có chuyển động [27]

- **Sự bí ẩn:** Xuất hiện sự đan xen của nhiều đối tượng trong một vùng
- **Đa đối tượng**
- **Không gian liên tục**
- **Tạm dừng thời gian:** Đối tượng ngừng chuyển động trong một khoảng thời gian
- **Mạnh mẽ:** Có thể xử lý được nhiễu
- **Tính tuần tự**
- **Xử lý dữ liệu không đầy đủ:** Đảm bảo xử lý được các dữ liệu bị mất mát
- **Các đối tượng dịch chuyển có khớp chuyển động**
- **Mô hình camera:** Thuật toán thực thi yêu cầu loại camera nào (trực giao, phối cảnh và bán phối cảnh)

#### **2.1.1.2. Một số hướng tiếp cận chính**

Hướng nghiên cứu về phân đoạn chuyển động rất rộng, vì vậy để khái quát các hướng tiếp cận dễ dàng cho chúng ta nghiên cứu, tôi sẽ phân loại các phương pháp dựa trên những lý thuyết chính của cách tiếp cận đó.

*Có thể chia thành 6 hướng tiếp cận chính:*

- Dựa trên sự sai biệt của ảnh thu nhận được giữa 2 hoặc nhiều thời điểm
- Dựa trên thống kê
- Dựa trên kỹ thuật luồng quang học
- Dựa trên biến đổi wavelets
- Dựa trên việc phân lớp, phân vùng
- Dựa trên phân tích phần tử
- Bảng dưới đây tổng hợp một số phương pháp nghiên cứu chính trong phân đoạn chuyển động video

<b>Sai biệt ảnh</b>	<i>Cavalaro et al. 2005 [34]</i>	F/D	√	√	√	√		-	-	√	-	
	<i>Cheng et al. 2006 [29]</i>	D	√	S	√		√	-	-	√	-	X
	<i>Li et al. 2007 [15]</i>	D		√	√	√		-	-	√	-	
	<i>Colombari et al. 2007 [19]</i>	D	√	√	√	√	√	-	-	√	-	
<b>Thống kê</b>	<b>MAP</b>											
	<i>Rasmussen et al. 2001 [52]</i>	D	√	√	√	√		-	-	√	-	X
	<i>Cremers et al. 2005 [33]</i>		√	√	√	√		-	-	√	-	X
	<i>Shen et al. 2007 [14]</i>		√	√	√	√	√	-	-	√	-	X
	<b>PF</b>		√	√	√	√		-	-		-	X
<b>Wavelets</b>	<i>Stolkin et al. 2008 [6]</i>		√	√	√	√	√	-	-		-	X
	<i>Wiskott 1997 [63]</i>	F		√	√			-	-		-	
<b>O.F.</b>	<i>Kong et al. 1998 [61]</i>	F	√	√	√		√	-	-	√	-	
	<i>Zhang et al. 2007 [16]</i>	F		√	√			-	-		-	X
<b>Phân tầng</b>	<i>Kumar et al. 2008 [10]</i>	F	√	S	√	√	√	-	-	√	-	T
<b>Phân tích nhân tử</b>	<i>Costeira et al. 1998 [59]</i>	F		√		√					O	
	<i>Ichimura et al. 2000 [57]</i>	F		√		√					O	
	<i>Kanatani et al. 2002 [49]</i>	F		√		√					O	
	<i>Brand 2002 [48]</i>	F	√	√		√	√		√	√	O	
	<i>Zelnik-Manor et al. 2003[45]</i>	F		√		√	√				O	
	<i>Zhou et al. 2003[46]</i>	F	√	√		√	√		√	√	P	
	<i>Sugaya et al. 2004[35]</i>	F		√		√	√				O	
	<i>Zelnik-Manor et al. 2004[40]</i>	F		√		√	√				O	
	<i>Gruber et al. 2004[37]</i>	F	√	√		√			√		O	
	<i>Vidal et al. 2004[39]</i>	F	√	√		√			√		O	
	<i>Yan et al. 2006[22]</i>	F		√		√				√	O	
	<i>Gruber et al. 2006[24]</i>	F	√	√	√	√	√		√		O	
	<i>Del Bue et al. 2007[23]</i>	F		√		√	√			√	P	
	<i>Goh et al. 2007[20]</i>	F		√		√			√	√	P	X
	<i>Julià et al. 2007[18]</i>	F	√	√		√	√		√		O	
Đặc trưng (F) / Mật độ (D) Sự bất ổn Đa đối tượng (S camera tĩnh) Không gian liên tục Thời gian Mạnh mẽ Tuần tự Dữ liệu không đầy đủ Đối tượng chuyển động có khớp chuyển động Kiểu camera (O Trực giao, p Bán phối cảnh., P Phối cảnh.) Tri thức tiên nghiệm (X)/Huấn luyện (T)												

Figure 2: Bảng tổng hợp một số kỹ thuật phân đoạn chuyển động nổi bật từ 2005-2008

### 2.1.2. Một số kỹ thuật phân đoạn đối tượng chuyển động nổi bật

- *Phân đoạn chuyển động dựa trên sự sai biệt của ảnh thu nhận được giữa hai hoặc nhiều thời điểm:*

A. Colombari [12] và cộng sự đề xuất kỹ thuật theo dõi đa đối tượng áp dụng cho phân đoạn các đối tượng chuyển động trong video. Giải pháp mà họ đưa ra gồm có: Mô hình hóa ảnh nền sử dụng phương pháp mô hình hóa hiệu quả, tìm và so khớp các vùng ảnh được phân tách sau quá trình trừ ảnh và phân cụm đối tượng. Thuật toán so khớp vùng được sử dụng dựa trên khoảng cách Mahalanobis giữa 2 chuỗi khung hình và sử dụng phân tách giá trị để tính toán một tập hợp các nguyên lý của khoảng cách và nguyên lý của loại trừ tương ứng.

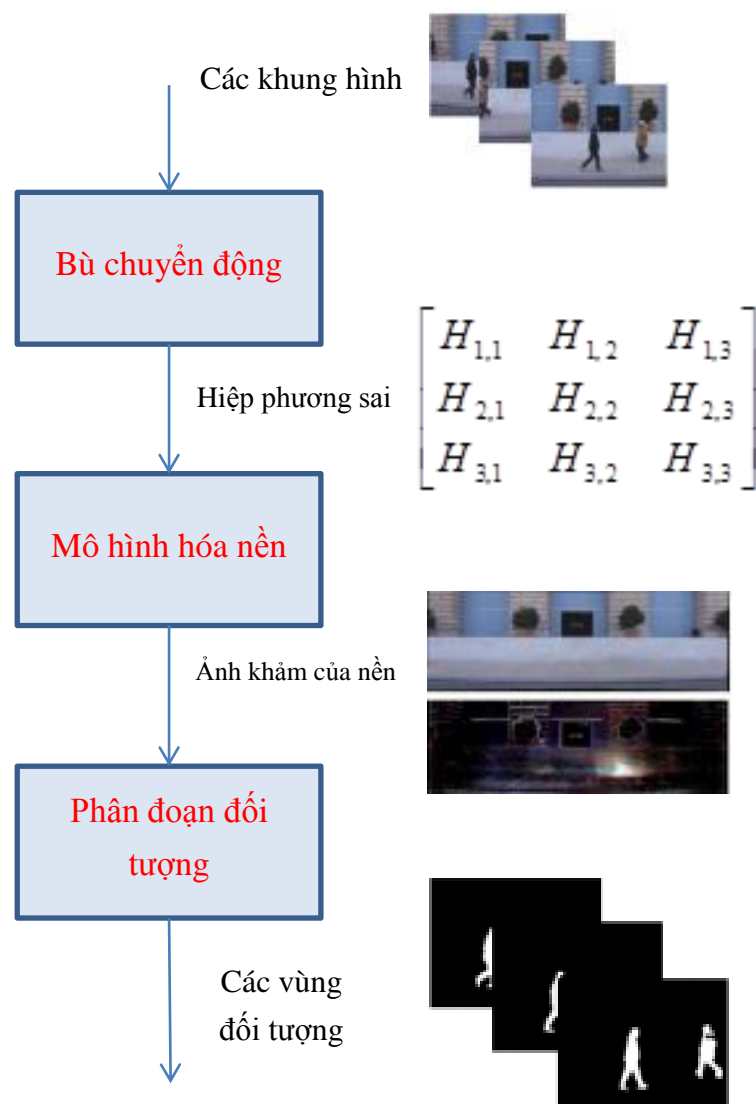


Figure 3: Mô hình của A. Colombari và cộng sự đề xuất

- ***Phân đoạn chuyển động dựa trên trường chuyển động của kỹ thuật luồng quang học***

Kỹ thuật trên được giới thiệu bởi Alessandro Verri và cộng sự năm 1989 [70]

Tư tưởng chính của kỹ thuật phân đoạn dựa trên 2 phương pháp:

- + Dựa trên ma trận Jacobi của đạo hàm riêng của các thành phần trong luồng quang học. Từ đó tính số lượng các vùng mở rộng, góc quay và các lát cắt
- + Dựa trên đặc tính cục bộ của luồng quang học, thứ bất biến với biến đổi tuyến tính được tính toán từ vết chuyển động và tính quyết định của ma trận đó.

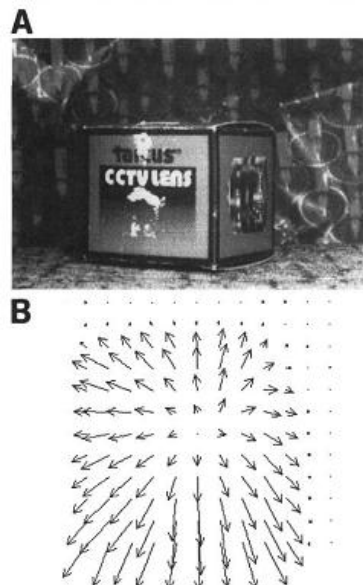
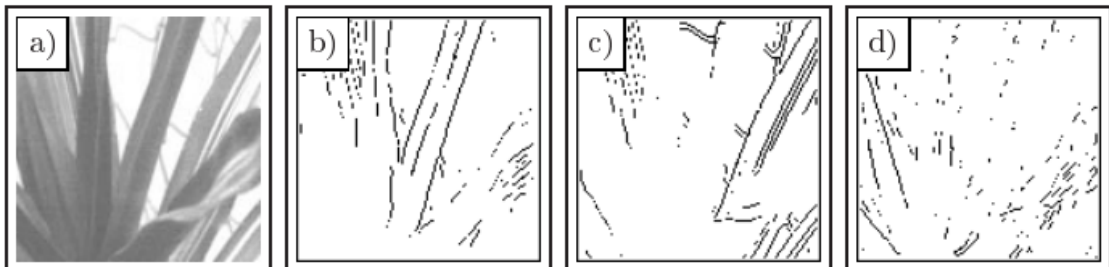


Figure 4: Mô hình của Alessandro Verri và cộng sự đề xuất

- ***Phân đoạn chuyển động Dựa trên biến đổi wavelets***

Năm 1997, Laurenz Wiskott [63] kết hợp hai kỹ thuật khác nhau dựa trên biến đổi Gabor-wavelet và Mallat-wavelet để khắc phục những vấn đề về độ mở và sự tương ứng.



- ***Phân đoạn chuyển động dựa trên sự phân lớp, phân vùng***

Yucel Altunbasak đề xuất kỹ thuật phân đoạn chuyển động dựa trên vùng màu sắc có tham số [60]. Tư tưởng chính của họ là lặp đi lặp lại quá trình tính toán phân đoạn chuyển động có tham số dựa vào véc-tơ chuyển động, tính và so khớp cường độ chuyển

động. Sau đó phân tách thành từng vùng có kích thước thay đổi, nhân dạng và phân loại các vùng dựa theo hình dạng biên bao bọc lấy đối tượng.

*Ren'e Vidal* coi video là sự chồng lên nhau của các đối tượng (các vùng đối tượng), mỗi đối tượng hay mỗi vùng là một không gian con. Việc phân đoạn chuyển động lúc này chuyển về bài toán theo dõi và phân đoạn các không gian con, để phân lớp mỗi điểm trên không gian con này. Kỹ thuật GPCA được sử dụng để giảm thiểu số không gian con, phục vụ cho các bước tiếp theo của phân đoạn chuyển động video theo vùng.

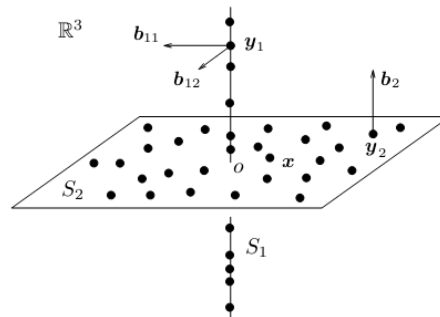


Figure 5: Áp dụng GPCA để giảm thiểu số chiều của không gian con

*M. Pawan Kumar* chia khung hình video thành nhiều lớp (vùng). Mỗi vùng sẽ được đưa vào bộ máy học – mô hình học không giám sát, nhằm tối ưu việc theo dõi các đối tượng có chuyển động phức tạp và có hình trạng không đoán biết trước. [9] Các bước của thuật toán gồm có:

- + Đưa ra thuật toán mới nhằm thu được ước lượng khởi tạo cho mô hình, sử dụng phép lan truyền đặc biệt.
- + Sử dụng thuật toán trao đổi  $\alpha\beta$  và thuật toán mở rộng  $\alpha$ .



Figure 6: Áp dụng kỹ thuật học không giám sát cho bài toán phân đoạn chuyển động

#### - Phân đoạn và theo dõi các đối tượng chuyển động dựa trong video

*Bing Han* sử dụng kỹ thuật theo dõi đối tượng chuyển động (cụ thể là kỹ thuật luồng quang học) để tính toán trường chuyển động. Từ đó làm nền tảng cho việc bám sát các đối tượng dịch chuyển. [17].



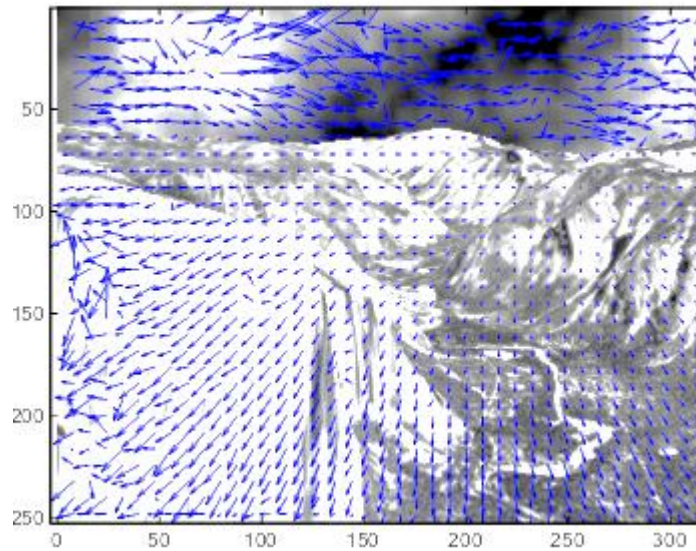


Figure 7: Thuật toán đề xuất của Bing Han

Với các kỹ thuật đã tìm hiểu phía trên, tôi nhận thấy: Việc áp dụng các bộ lọc tín hiệu kết hợp cùng xử lý ảnh đem lại hiệu quả khả quan. Vì vậy trong chương hai, tôi sẽ tập trung trình bày về các kỹ thuật lọc cũng như mô hình đề xuất cho bài toán phân đoạn đối tượng chuyển động trong video dựa trên kỹ thuật phân đoạn và theo dõi đa đối tượng chuyển động.

A. Colombari đề xuất kỹ thuật theo dõi đa đối tượng áp dụng cho phân đoạn các đối tượng chuyển động trong video. Giải pháp mà họ đưa ra gồm có: Mô hình hóa ảnh nền sử dụng phương pháp mô hình hóa hiệu quả, tìm và khớp các vùng ảnh được phân tách sau quá trình trừ ảnh và phân cụm đối tượng. [12] Thuật toán khớp vùng được sử dụng dựa trên khoảng cách Mahalanobis giữa 2 chuỗi khung hình và sử dụng phân tách giá trị để tính toán một tập hợp các nguyên lý của khoảng cách và nguyên lý của loại trừ tương ứng.



Figure 8: Kỹ thuật đề xuất của A. Colombari

Với các kỹ thuật đã tìm hiểu phía trên, tôi nhận thấy: Việc áp dụng các bộ lọc tín hiệu kết hợp cùng xử lý ảnh đem lại hiệu quả khả quan. Vì vậy trong chương hai, tôi sẽ tập

trung trình bày về các kỹ thuật lọc cũng như mô hình đề xuất cho bài toán phân đoạn đối tượng chuyển động trong video dựa trên kỹ thuật phân đoạn và theo dõi đa đối tượng chuyển động.

### 2.1.3. Vai trò và ứng dụng của kỹ thuật phân đoạn chuyển động

Kỹ thuật xử lý video là một trong những lĩnh vực nghiên cứu quan trọng trong xử lý đa phương tiện – phục vụ đắc lực cho sự phát triển kinh tế, xã hội. Xử lý video bao gồm việc thao tác các dữ liệu trực quan nhằm phân tích, phát hiện, đánh chỉ mục... Phát hiện và phân đoạn chuyển động của đối tượng là một trong những nhiệm vụ quan trọng và cần thiết của công nghệ thị giác máy tính [17].



Figure 9: Kỹ thuật phân đoạn chuyển động video ứng dụng theo dõi phương tiện giao thông

Kỹ thuật phân đoạn chuyển động ứng dụng trong công nghiệp giải trí và truyền thông có thể nói là rộng nhất và đem lại nhiều lợi ích kinh tế cũng như ý nghĩa thực tiễn nhất. Việc áp dụng kỹ thuật phân đoạn chuyển động giúp ta nén các chuỗi hình ảnh liên tiếp có kích thước rất lớn, không phù hợp với việc lưu trữ, truyền tải vào tập các điểm ảnh chuyển động kết hợp với chuỗi hình ảnh tương ứng [51]. Từ đó, giảm thiểu kích thước của video phục vụ việc lưu trữ, truyền dẫn, tổ chức dữ liệu video. Góp phần tạo nên sự phát triển bùng nổ của hệ thống giải trí đa phương tiện, đàm thoại video, hội nghị trực tuyến, hệ thống học liệu trực tuyến, cũng như các mạng xã hội ... – trở thành bước đệm vô cùng lớn cho sự thay đổi cơ bản cục diện công nghệ thông tin – truyền thông.

Trong quốc phòng, các hệ thống radar theo dõi và phân đoạn chuyển động các đối tượng, trong hệ thống phòng thủ tên lửa, các radar theo dõi và phân tích các đối tượng chuyển động với vận tốc cực nhanh, nhằm đưa ra những phản ứng kịp thời



Trong An ninh, giám sát, hệ thống sẽ phân tích và bám sát các đối tượng chuyển động phục vụ quá trình quan sát hành vi của các đối tượng đang hiện hữu trước ống kính camera; Trong đời sống hàng ngày, các chú robot thông minh phân tích các chuyển động xung quanh, từ đó đưa ra những quyết định để thao tác, xử lý các công việc phục vụ con người.

## **2.2. Phương pháp tiếp cận thứ nhất – Dựa trên sự sai biệt ảnh**

Phương pháp tiếp cận dựa trên sai biệt ảnh, có thể gọi bằng cách khác là phân đoạn dựa trên sự thay đổi [17].

Phát hiện các vùng có sự thay đổi trong ảnh thu nhận được ở các thời điểm khác nhau là một lĩnh vực nghiên cứu được rất nhiều nhà khoa học quan tâm bởi nó có tính ứng dụng rất cao. Ứng dụng quan trọng của phát hiện sự thay đổi bao gồm: Giám sát video, cảm biến từ xa, chuẩn đoán và điều trị y học, cảm biến dưới nước. [17]

Phát hiện sự thay đổi thường bao gồm 3 bước:

- **Sự điều chỉnh về hình học:** Loại bỏ một số sự thay đổi không có ý nghĩa (khi camera bị dịch chuyển, các đối tượng đứng yên nhưng ta vẫn thu nhận được sự thay đổi).
- **Điều chỉnh cường độ:** Nhằm giảm thiểu ảnh hưởng của sự thay đổi điều kiện chiếu sáng và nhiễu thu nhận
- **Sử dụng mô hình ngẫu nhiên để kiểm định các giả thuyết:** Từ đó quyết định xem vùng nào/điểm ảnh nào có sự thay đổi mà chúng ta cần quan tâm

### **2.2.1. Tổng quan về kỹ thuật trừ nền**

Mục tiêu của bài toán trừ ảnh là xác định được các vùng ảnh sai biệt giữa 2 hoặc nhiều ảnh thu nhận được ở các thời điểm khác nhau.

Bài toán được đề xuất và được rất nhiều nhà khoa học dày công nghiên cứu từ giai đoạn đầu tiên của các hệ thống giám sát 1960-1980 [72]. Từ đó đến nay, đã có rất nhiều nghiên cứu có giá trị thực tiễn và vẫn đang tiếp tục được phát triển theo chiều sâu.

Với kỹ thuật phát hiện đối tượng chuyển động, người ta chia thành 2 phân nhánh chính:

- Nền cố định
- Nền thay đổi

Với bài toán nền cố định, ta có thể áp dụng kỹ thuật sai phân tạm thời hoặc thuật toán trừ nền [72]. Tuy nhiên, với bài toán có nền thay đổi – chúng ta phải áp dụng rất nhiều kỹ thuật phức tạp. Nhánh nghiên cứu với nền thay đổi đòi hỏi sự thách thức lớn hơn.

Các bài toán với nền cố định có thể giải quyết bằng một số kỹ thuật cơ bản như sau:

- Thuật toán trừ ảnh:
  - o Trừ ảnh để tìm ảnh sai biệt giữa 2 ảnh liên tiếp
  - o Trừ ảnh với các điểm ảnh có **giá trị trung vị của K thời điểm trước đó**
  - o Trừ ảnh với các giá trị điểm ảnh là **trung bình của K thời điểm trước...**[12]
- Dựa trên phân bố Gaussian [32]
- Dựa trên ước lượng MeanShift
- Dựa vào nền riêng (Eigenbackgrounds) [32]

Ngoài ra, có các vấn đề đi song song cùng với bài toán trên là các kỹ thuật loại bỏ bóng khỏi đối tượng khi thu nhận hình ảnh của đối tượng dưới các góc của các nguồn sáng. Đây cũng là loạt bài toán rất thú vị và rất hấp dẫn đối với các phòng nghiên cứu của một số trường ĐH trên thế giới.

Trong phạm vi bài báo cáo này, tôi sẽ trình bày thuật toán trừ nền với 2 kỹ thuật chủ đạo là trừ nền trước, trừ nền trung vị, trừ nền trung bình và sử dụng mô hình hỗn hợp Gauss (Mixture of Gaussian) – *riêng kỹ thuật sử dụng mô hình hỗn hợp Gaussian tôi sẽ kết hợp trình bày ở phần Error! Reference source not found..* Từ đó đưa ra đánh giá về các kỹ thuật trên, cũng như mức độ phù hợp cho bài toán phân đoạn chuyển động video.

### Các giai đoạn chính

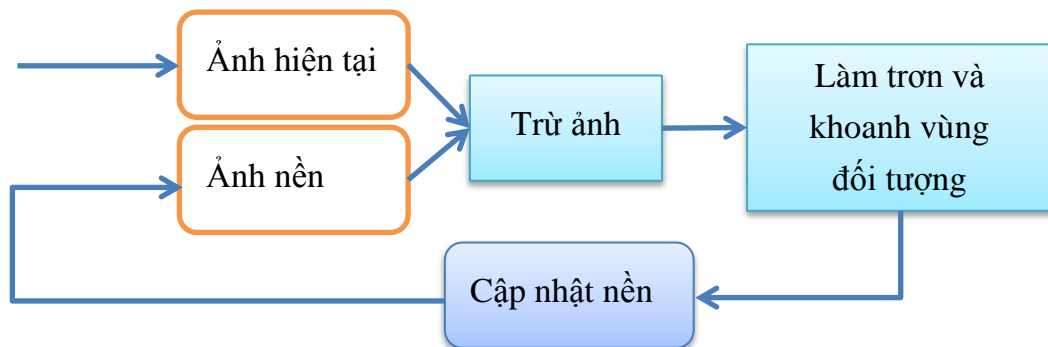


Figure 10: Mô hình chung của phương pháp tiếp cận dựa trên sai biệt

### Bước 1: Trừ ảnh

Trong bước này, ta thực hiện việc duyệt qua từng điểm ảnh của 2 ảnh đưa vào thực hiện phép trừ. Ta so sánh trị tuyệt đối giữa giá trị điểm ảnh này với một ngưỡng cho trước nào đó. Nếu trị tuyệt đối sai biệt lớn hơn ngưỡng thì ta gán màu cho ảnh sai biệt tại điểm đang xét là trắng, ngược lại là màu đen.

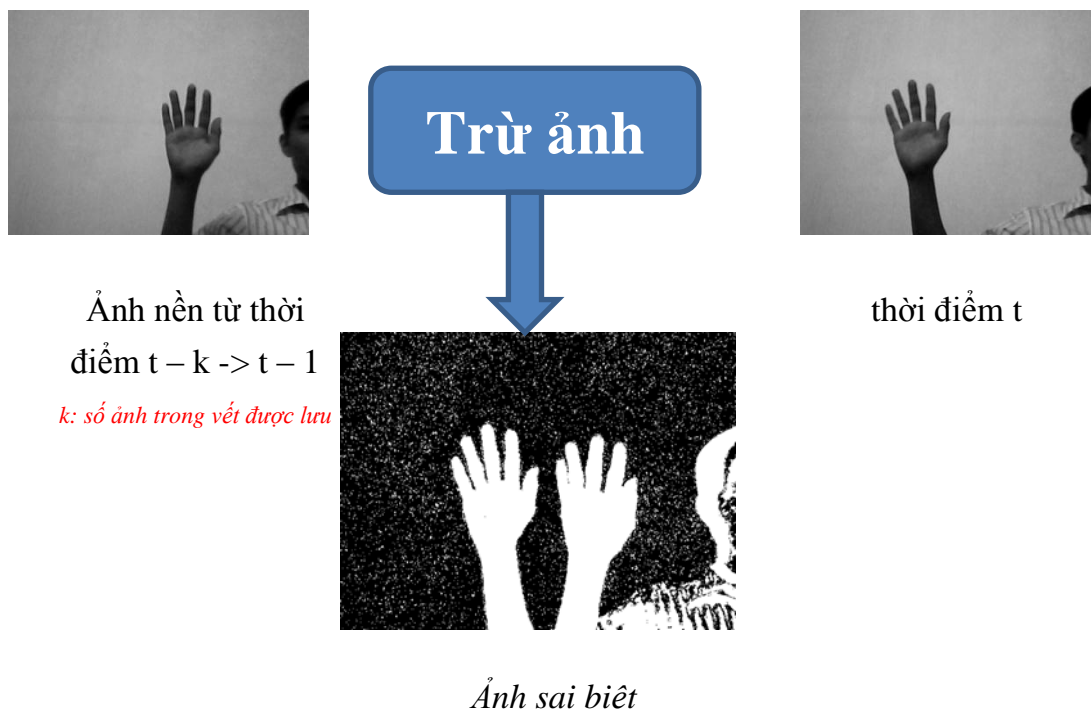


Figure 11: Một ví dụ về kỹ thuật trừ ảnh

## Bước 2: Làm trơn ảnh

Trên thực tế, việc thu nhận hình ảnh từ các thiết bị ghi hình có thể gặp sai sót. Hoặc bị ảnh hưởng trong quá trình nén và giải nén video. Nên sẽ có rất nhiều điểm nhiễu, ta thường gọi là nhiễu sạn, ...

Để loại bỏ yếu tố trên, ta có thể sử dụng các kỹ thuật lọc ảnh để làm trơn ảnh.

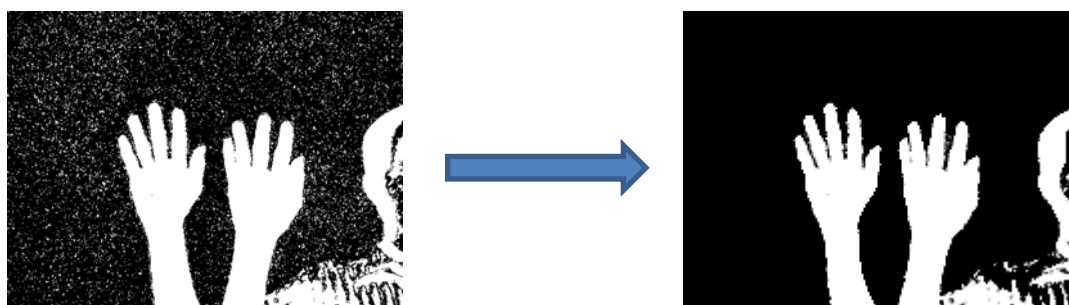


Figure 12: Ví dụ về làm trơn ảnh sử dụng bộ lọc ảnh

## Bước 3: Khoanh vùng và gán nhãn đối tượng

Đây là khâu cuối giúp ta khoanh vùng để xác định được đối tượng chuyển động trong video. Bài toán khoanh vùng và gán nhãn – tương ứng với bài toán loang. Tuy nhiên, nếu gọi lời gọi đệ quy cho kỹ thuật loang thông thường, độ phức tạp của bài toán sẽ cao và không ổn định [1]. Vì vậy, tôi nghiên cứu và lựa chọn một kỹ thuật khoanh vùng và gán nhãn khác hiệu quả hơn.

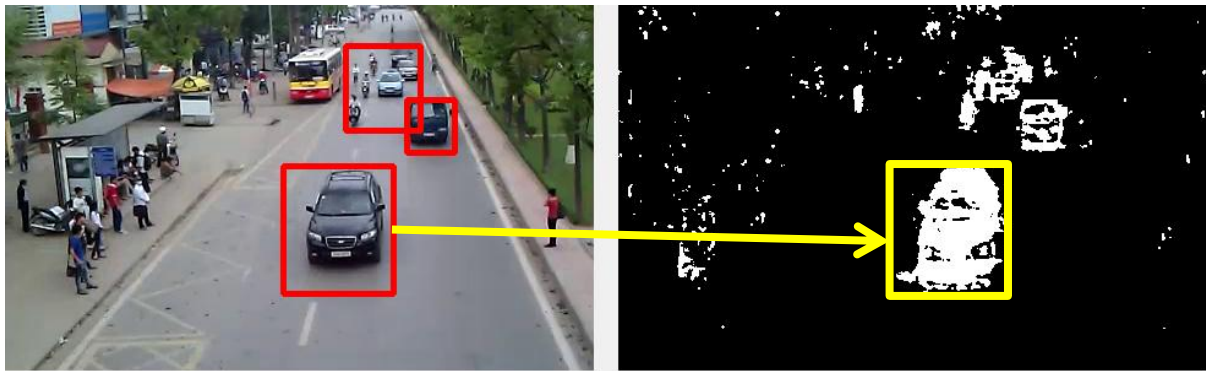


Figure 13: Ví dụ về kỹ thuật khoanh vùng đối tượng chuyển động

Thuật toán khoanh vùng đối tượng nhanh – độ phức tạp  $O(n^2)$

```

Input : Ảnh sai biệt (I_diff)
Output : Danh sách các đối tượng (listObject)
BEGIN
    L = 1 // Khởi tạo số thứ tự nhãn
    Duyệt qua từng pixel của ảnh I_diff
    if pixel X không phải là nền
        if lân cận A,B,C & D đã được gán nhãn( = 0)
            gán nhãn pixel X là L
            L = L+1
        else
            num = MIN(nhãn của các lân cận A,B,C & D) điều kiện #0
            gán nhãn pixel X và pixels A, B, C & D if không phải là nền với
            giá trị num
        end if
    end if
done

    Duyệt qua từng pixel(x, y) của ảnh I_diff
    Index = nhãn của (x, y)
    if(Index > 0) cập nhật tọa độ, kích thước của listObject[Index]

END

```

#### Bước 4: Cập nhật ảnh nền

Với kỹ thuật trừ nền, chúng ta thường có 3 cách triển khai cơ bản nhất:

- Ảnh nền là ảnh trước đó
- Ảnh nền là ảnh trung bình tại K thời điểm trước đó
- Ảnh nền là ảnh với từng điểm ảnh là trung vị của các điểm ảnh tương ứng trong K ảnh trước đó

#### 2.2.2. Trừ nền đơn giản

Trong kỹ thuật này, ta duyệt qua từng điểm ảnh và lấy trị tuyệt đối của điểm ảnh hiện tại trên 2 ảnh previous và current. Nếu độ chênh lệch trên điểm (x, y) đó lớn hơn ngưỡng **motionThreshold** thì ta gán màu của ảnh sai biệt là trắng. Ngược lại gán màu đen cho ảnh sai biệt.



Figure 14: Ảnh mô phỏng thuật toán trừ nền đơn giản

### Thuật toán trừ nền đơn giản

```

Input      : Ảnh frame trước ( $I_{previous}$ ) & Ảnh hiện tại ( $I_{current}$ )
Output    : Ảnh sai biệt ( $I_{diff}$ )
BEGIN
for pixel ( $x, y$ ) in  $I_{current}$ 
    if (  $| get(I_{current}, x, y) - get(I_{previous}, x, y) | > motionThreshold$  )
        Set( $I_{diff}, x, y, white$ ) ;
    else
        Set( $I_{diff}, x, y, black$ )
END .

```

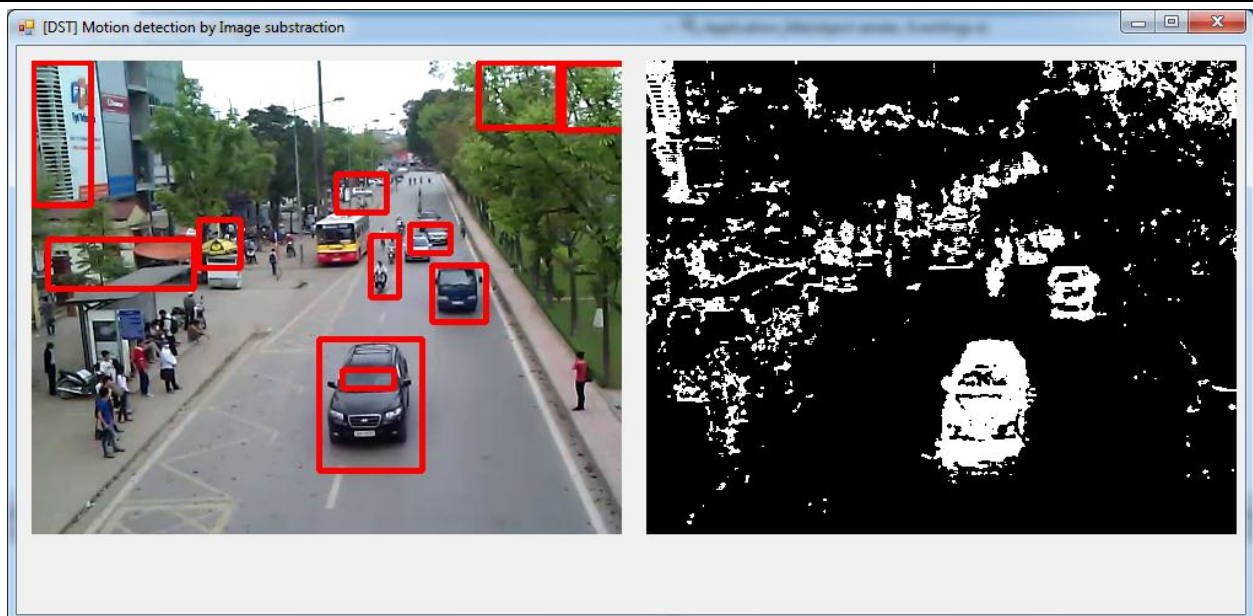


Figure 15: Kỹ thuật trừ nền trước bộc lộ nhược điểm

Thuật toán trừ ảnh đơn giản có những đặc điểm cơ bản:

- Lựa chọn ngưỡng **Th** ảnh hưởng đến kết quả
- Phụ thuộc vào cấu trúc đối tượng, tốc độ
- Cách tiếp cận này thường không được sử dụng

$Th = 25$



$Th = 50$



$Th = 100$



$Th = 200$



Figure 16: Ví dụ về kết quả của kỹ thuật trừ nền đơn giản

## 2.2.3. Trừ nền trung vị và nền trung bình

### 2.2.3.1. Trừ nền trung vị

Với kỹ thuật trừ nền trung bình, ta luôn lưu lại tập các ảnh tại K thời điểm liên tiếp trước đó. Trước tiên, ta sẽ tính ảnh trung vị (**median**) từ tập các ảnh tại K thời điểm ngay trước đó.

Nền trung vị được tính bằng công thức

$$B(x, y, t) = \text{median}_{i=0}^K \{I(x, y, t - i)\}$$

Để tính ảnh sai biệt, ta duyệt qua từng điểm ảnh và lấy trị tuyệt đối của điểm ảnh hiện tại trên 2 ảnh **median** và **current**. Nếu độ chênh lệch trên điểm (x, y) đó lớn hơn ngưỡng **motionThreshold** thì ta gán màu của ảnh sai biệt là trắng. Ngược lại gán màu đen cho ảnh sai biệt.

Với kỹ thuật trừ nền trung vị

$$|I(x, y, t) - \text{median}_{i=0}^K \{I(x, y, t - i)\}| > Th$$



### Thuật toán được thực hiện như sau:

```
Input      : ảnh nền (median) & Ảnh hiện tại (current)
Output    : Ảnh sai biệt (diff)
BEGIN
for pixel (x, y) in current
    if ( | get(current, x, y) - get(median, x, y) | > motionThreshold )
        Set(diff, x, y, white) ;
    else
        Set(diff, x, y, black)
median <= updateImages(current)
END.
```

Ví dụ: Với kỹ thuật trừ nền trung vị

+ Chọn  $N = 10$



+ Chọn  $N = 20$



+ Chọn  $K = 50$



Figure 17: Ví dụ về kết quả của kỹ thuật trừ nền trung vị

### 2.2.3.2. Trừ nền trung bình

Với kỹ thuật trừ nền trung bình, ta luôn lưu lại tập các ảnh tại K thời điểm liên tiếp trước đó. Trước tiên, ta sẽ tính ảnh trung bình (avg) từ tập các ảnh tại K thời điểm ngay trước đó.

Nền trung bình được tính bằng công thức sau:

$$B(x, y, t) = \frac{1}{K} \sum_{i=0}^K I(x, y, t - i)$$

Để tính ảnh sai biệt, ta duyệt qua từng điểm ảnh và lấy trị tuyệt đối của điểm ảnh hiện tại trên 2 ảnh avg và current. Nếu độ chênh lệch trên điểm (x, y) đó lớn hơn ngưỡng **motionThreshold** thì ta gán màu của ảnh sai biệt là trắng. Ngược lại gán màu đen cho ảnh sai biệt.

$$\left| I(x, y, t) - \frac{1}{K} \sum_{i=0}^K I(x, y, t - i) \right| > Th$$

Ví dụ: Với kỹ thuật trừ nền trung bình

+ Chọn **K = 10**



+ Chọn **K = 20**





+ Chọn **K = 50**



Figure 18: Ví dụ về kết quả của kỹ thuật trừ nền trung bình

**Thuật toán được thực hiện như sau:**

```
Input      : ảnh nền (avg) & Ảnh hiện tại (current)
Output    : Ảnh sai biệt (diff)
BEGIN
for pixel (x, y) in current
    if ( | get(current, x, y) - get(avg, x, y) | > motionThreshold )
        Set(diff, x, y, white) ;
    else
        Set(diff, x, y, black)
avg <= updateImages(current)
END.
```

### 2.2.3.3. Cập nhật nền

Để cập nhật nền, cách đơn giản nhất, ta sử dụng hàng đợi kép, mỗi lần cập nhật ta lấy ảnh ở đầu hàng đợi ra, và thêm ảnh cần cập nhật vào cuối hàng đợi. Nhược điểm của việc này là tốn bộ nhớ và độ phức tạp cao (vì mỗi lần cập nhật ta lại phải tính ảnh trung bình mới bằng cách lặp qua K ảnh để tính trung bình)

Để khắc phục nhược điểm tốn bộ nhớ, chúng ta không lưu trữ lại lịch sử các ảnh, mà sẽ sử dụng phương pháp học cập nhật:

$$B(x, y, t) = \frac{t-1}{t} B(x, y, t-1) + I(x, y, t)$$

Hoặc tổng quát hơn

$$B(x, y, t) = (1 - \alpha) B(x, y, t-1) + \alpha I(x, y, t)$$

Với  $\alpha$  là tốc độ học

$\alpha$  được chọn dựa trên thực nghiệm

#### **2.2.3.4. Đánh giá**

##### **Ưu điểm:**

- Thật sự dễ triển khai và sử dụng
- Tốc độ tốt
- Nền không cố định, được cập nhật thường xuyên

##### **Nhược điểm:**

- Độ chính xác phụ thuộc vào sự sai biệt giữa các ảnh thu được
- Độ chính xác còn phụ thuộc vào tốc độ chuyển động của đối tượng
- Tốn bộ nhớ (vì phải lưu trữ nhiều ảnh), tuy nhiên có thể sử dụng hệ số học để giải quyết vấn đề trên

## 2.3. Phương pháp tiếp cận thứ hai – Dựa trên kỹ thuật luồng quang học

### 2.3.1. Tổng quan về luồng quang học

Luồng quang học (Optical flow) là khái niệm chỉ sự chuyển động tương đối của các điểm trên bề mặt đối tượng, vật thể nào đó gây ra dưới góc quan sát của một điểm mốc. Thuật toán luồng quang học là dùng để xác định xấp xỉ các chuyển động của các điểm ảnh từ thời điểm này sang thời điểm khác trong một khoảng thời gian  $t$ .

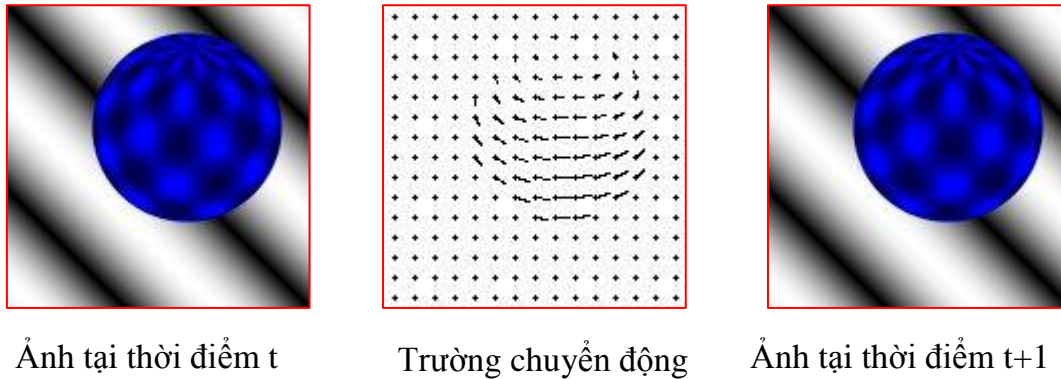


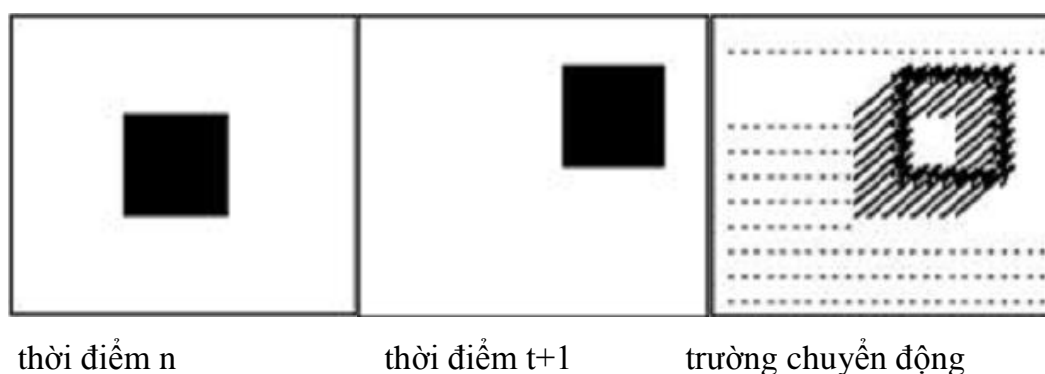
Figure 19: Trường chuyển động trong kỹ thuật luồng quang học

Optical flow được giới thiệu lần đầu tiên trong nghiên cứu của James J. Gibson, công bố bởi Nhà XB Tâm lý học Mỹ những năm 1940. [16]

Optical thường được áp dụng trong các bài toán như: phát hiện chuyển động, phân đoạn đối tượng, ...

### 2.3.2. Trường chuyển động

Ảnh của một đối tượng di chuyển trước camera sẽ có sự thay đổi tương ứng theo thời gian. Xét một tập các điểm  $p_i$  di chuyển với vận tốc  $v_i$  giống như hình dưới đây, thì tập các điểm như vậy được gọi là trường chuyển động. Mục đích của luồng quang học là tìm trường chuyển động này



Ta có các nhận xét sau về trường chuyển động [62]

- Hướng của chuyển động là hướng của các vector vận tốc  $v_i$
- Độ lớn của vector  $v_i$  chính là quãng đường chuyển động của vật thể
- Trong trường hợp các mũi tên tỏa ra các phía  $\Rightarrow$  chuyển động vào gần camera
- Trong trường hợp hướng mũi tên hội tụ  $\Rightarrow$  chuyển động ra xa camera

### 2.3.3. Phương trình luồng quang học

Có nhiều kĩ thuật xác định luồng quang học khác nhau như: Correlation, matching, feature tracking, energy based... nhưng bài này sẽ tập trung vào các phương pháp dựa trên gradient. Ý tưởng quan trọng của phương pháp dựa trên gradient là giả định sau:

***“Bề mặt ngoài của đối tượng không có nhiều thay đổi về cường độ sáng khi xét sự thay đổi từ thời điểm  $t$  sang thời điểm  $t+1$ ”.***

Nghĩa là:

$$I(x, y, t) = I(x + u, y + v, t + 1) \quad (2.3.4.1)$$

Trong đó  $I(x, y, t)$  là hàm trả về cường độ sáng của điểm ảnh tại tọa độ  $(x, y)$  ở thời điểm  $t$  ( thời điểm thứ  $t$ ).  $(u, v)$  là véc-tơ vận tốc, thể hiện sự thay đổi vị trí của điểm ảnh từ thời điểm thứ  $t$  sang thời điểm thứ  $t + 1$ .

Sử dụng khai triển Taylor ta được:

$$I(x + u, y + v, t + 1) = I(x, y, t) + (u, v) * I'(x, y, t) + O((u, v)^2 * I''(x, y, t))$$

Bỏ qua các phần tử có đạo hàm cao hơn 1 ta được xấp xỉ:

$$I(x+u, y + v, t + 1) \approx I(x, y, t) + \frac{\partial I}{\partial x} * u + \frac{\partial I}{\partial y} * v + \frac{\partial I}{\partial t} = I(x, y, t) + I_x * u + I_y * v + I_t \quad (2.3.4.2)$$

Trong đó  $I_x, I_y, I_t$  lần lượt là đạo hàm riêng của  $I(x, y, t)$  theo  $x, y, t$ .

Từ (2.3.4.1) và (2.3.4.2) ta có:

$$u * I_x + v * I_y + I_t = 0 \quad (2.3.4.3)$$

Phương trình (2.3.4.3) chính là phương trình luồng quang học. Trong đó  $(u, v)$  chính là luồng cần tìm.

Phương trình 2.3.4.3 cũng có thể đạt được bằng cách dẫn xuất ngay từ giả thiết ban đầu của chúng ta giá trị độ sáng của điểm ảnh  $(x, y)$  không thay đổi từ thời điểm  $t$  sang thời điểm  $t + 1$ .

$$\begin{aligned}
\frac{d}{dt} I(x(t), y(t), t) &= 0 \\
\Leftrightarrow \frac{\partial I}{\partial x} \frac{\delta x}{\delta t} + \frac{\partial I}{\partial y} \frac{\delta y}{\delta t} + \frac{\partial I}{\partial t} \frac{dt}{dt} &= 0 \\
\Leftrightarrow I_x * u + I_y * v + I_t &= 0
\end{aligned}$$

Với  $u = \frac{\delta x}{\delta t}$ ,  $v = \frac{\delta y}{\delta t}$

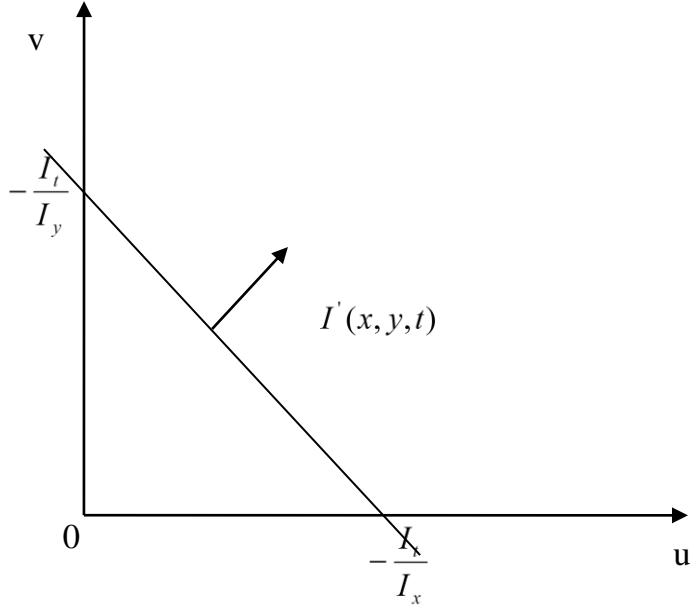


Figure 20: Phương trình Optical flow biểu diễn trên hệ trục tọa độ

Ở phần trên chúng ta đã tìm được phương trình optical flow. Tuy nhiên phương trình này có dạng  $A * u + B * v + C = 0$  với 2 ẩn là  $u, v$  nên ta không thể giải được. Chúng ta cũng nhận thấy phương trình (3) với ẩn là  $(u, v)$  là một đường thẳng trong không gian 2D với vector pháp tuyến là  $I'(x, y, t)$ . Mọi phương pháp ước lượng optical flow đều xoay quanh việc tìm thêm một phương trình nữa để tính  $(u, v)$ . Trong bài này chúng tôi xin giới thiệu các phương pháp ước lượng Cực tiểu bình phương.

#### + Phương pháp Cực tiểu bình phương (Least-Squares estimation)

Để tìm thêm một phương trình nữa cho  $(u, v)$ , ta có dùng các điểm ảnh lân cận điểm ảnh đang xét, với giả sử các điểm ảnh lân cận này cũng nằm trên vật thể và chuyển động cùng với vận tốc đang xét. Trong thực tế các điểm lân cận có thể chuyển động không cùng vận tốc với điểm ảnh đang xét, vì vậy ta tìm vector vận tốc làm cực tiểu bình phương lỗi.

$$E(u, v) = \sum_{x, y} g[u * I_x + v * I_y + I_t]^2 \quad (4)$$

Trong đó  $g$  xác định “độ rộng” của các điểm lân cận điểm đang xét hay nói cách khác  $g$  là cửa sổ bao quanh điểm đang xét. Vận tốc cần tìm là vector  $(u, v)$  làm cực tiểu  $E(u, v)$ .

Người ta đã chứng minh được  $E(u, v)$  cực tiểu tại đạo hàm riêng của nó bằng 0. Tức là:

$$\frac{\partial E(u, v)}{\partial u} = \sum_{x, y} g \left[ u * I_x^2 + v * I_x I_y + I_x I_t \right] = 0$$

$$\frac{\partial E(u, v)}{\partial v} = \sum_{x, y} g \left[ v * I_y^2 + u * I_x I_y + I_y I_t \right] = 0$$

Hai phương trình này có thể viết dưới dạng ma trận.

$$M * \vec{v} = \vec{b} \quad (5)$$

Trong đó:

$$M = \begin{bmatrix} \sum g I_x^2 & \sum g I_x I_y \\ \sum g I_x I_y & \sum g I_y^2 \end{bmatrix}$$

$$\vec{v} = \begin{bmatrix} u \\ v \end{bmatrix}$$

$$\vec{b} = - \begin{bmatrix} \sum g I_x I_t \\ \sum g I_y I_t \end{bmatrix}$$

Nghiệm của (5) là :  $\vec{v} = M^{-1} * \vec{b}$

**Chú ý:**

- Thông thường ta phải tính  $M$  và  $\vec{b}$  tại mỗi điểm  $(x, y)$  trên ảnh. Do đó để tăng tốc ta có thể tính trước đạo hàm của ảnh  $(I_x, I_x^2, I_y, I_y^2, I_t, I_t^2)$ .
- Nếu dữ liệu trên ảnh đa phân biến thiên theo một chiều nhất định thì  $M$  có thể không khả nghịch dẫn đến hệ phương trình không thể giải được.

#### 2.3.4. Đánh giá

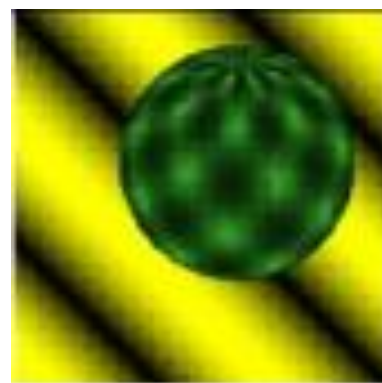
Với giả thuyết đặt ra cho phương trình luồng quang học, ta phải coi cường độ sáng của ảnh không thay đổi từ frame này sang frame kia. Trên thực tế, cường độ sáng của cùng một đối tượng – trong các thời điểm khác nhau thì không giống nhau. Có thể có nhiễu trong quá trình thu nhận từ các thiết bị ghi hình... Vì vậy, luồng quang học hoạt động kém hiệu quả trong một số trường hợp phổ biến như:

- Cường độ sáng không ổn định
- Có ít sự phân biệt về màu sắc giữa đối tượng và nền

### 2.3.5. Áp dụng luồng quang học trong phân đoạn chuyển động video

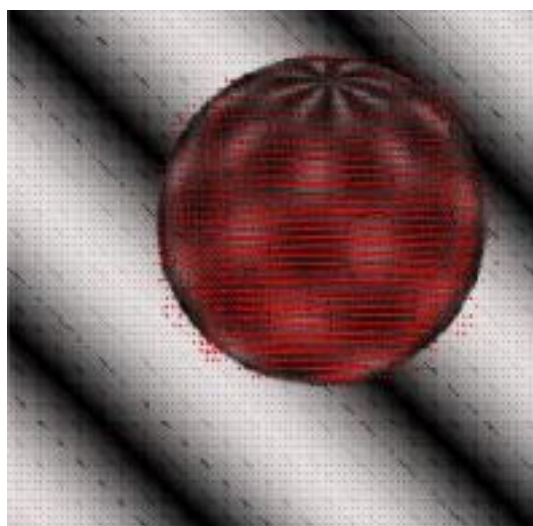


Thời điểm  $t$



Thời điểm  $t+1$

Đầu tiên, với chuỗi ảnh thu được từ video, ta tính trường chuyển động của chúng



Trường chuyển động

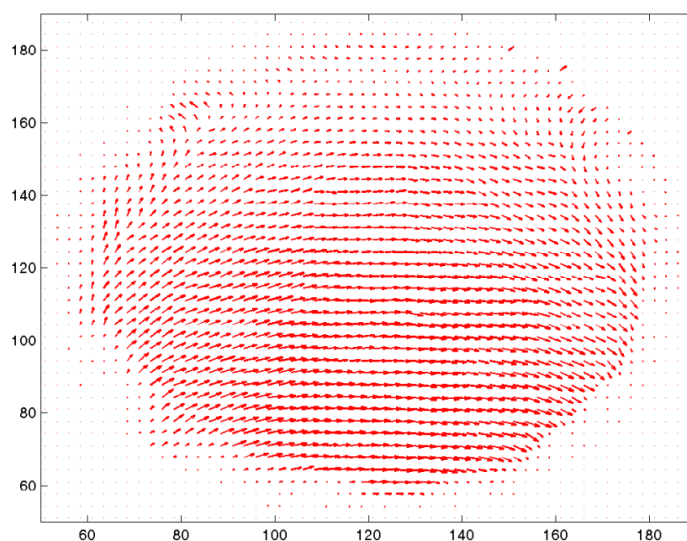


Figure 21: Ứng dụng luồng quang học vào phân đoạn chuyển động

Sau đó, chúng ta sử dụng các kỹ thuật phụ trợ khác để gom cụm các véc-tơ chuyển động lại. Kết quả các cụm là các vùng chuyển động đã được phân đoạn.

## 2.4. Phương pháp tiếp cận thứ ba – Dựa trên mô hình thống kê

### 2.4.1. Mô hình hỗn hợp Gaussians

#### 2.4.1.1. Mô hình Gaussian đơn

**Phân phối Gauss**, là một phân phối xác suất cực kì quan trọng trong nhiều lĩnh vực. Nó là họ phân phối có dạng tổng quát giống nhau, chỉ khác tham số *vị trí* (giá trị trung bình  $\mu$ ) và *tỉ lệ* (phương sai  $\sigma^2$ ). Đây là dạng phân phối dễ tính toán và thường được sử dụng nhất. Phân phối Gauss có trung bình, trung vị và yếu vị trùng nhau, và có đồ thị đối xứng dạng hình chuông.

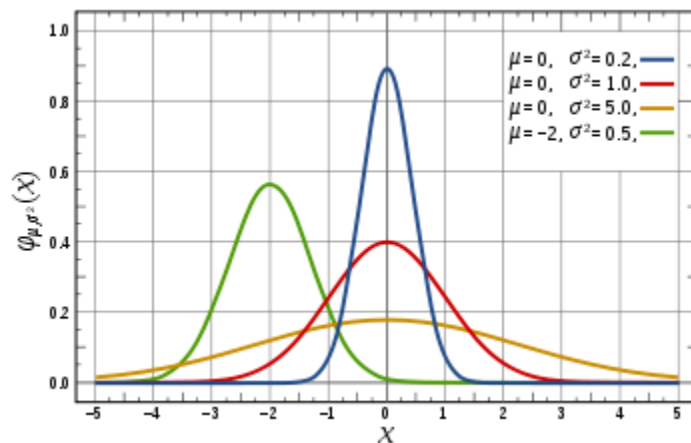


Figure 22 Đồ thị hàm mật độ Gauss

Hình trên minh họa bốn hàm mật độ Gauss với các tham số khác nhau. Một cách hình thức, hàm mật độ xác suất của phân phối Gauss  $f_N(x, \mu, \sigma^2)$  được cho bởi công thức:

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

trong đó,  $\mu$  là giá trị trung bình,  $\sigma$  là độ lệch chuẩn. Trong trường hợp  $x$  là vector gồm  $D$  thành phần, hàm mật độ xác suất của phân phối Gauss  $f_N(x, \mu, \Sigma)$  được cho bởi công thức:

$$p(\vec{x}) = \frac{1}{(2\pi)^{D/2} (\Sigma)^{1/2}} \exp\left(-\frac{1}{2} (\vec{x} - \vec{\mu})' \Sigma^{-1} (\vec{x} - \vec{\mu})\right)$$

Trong đó  $\vec{\mu}$  là vector trung bình,  $\Sigma$  là ma trận hiệp phương sai.



### 2.4.1.2. Mô hình hỗn hợp Gaussian

Về cơ bản, mô hình hỗn hợp Gaussians (Gaussian Mixture Model – GMM) xấp xỉ một hàm mật độ xác suất bằng hợp các hàm mật độ Gauss.

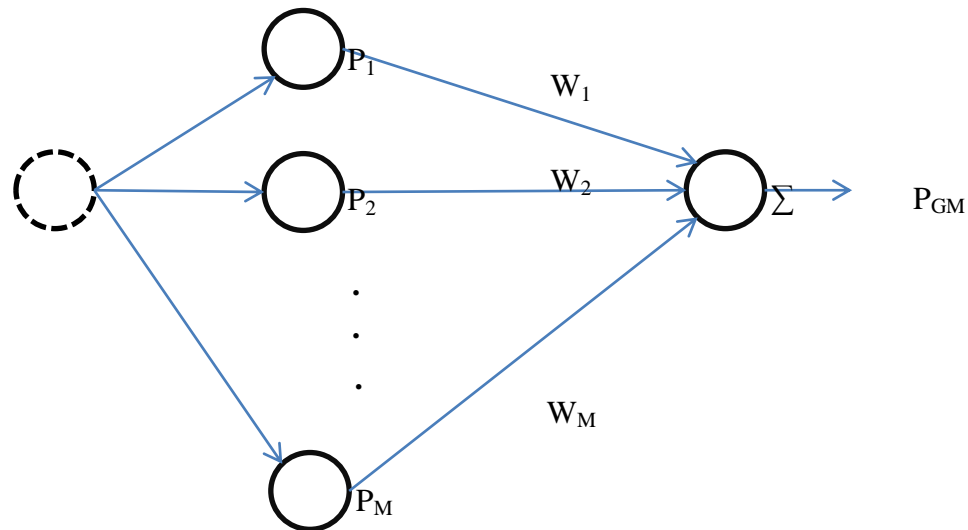


Figure 23 Mô hình GMM

Cho trước  $M$  phân phối Gauss  $p_1, p_2, \dots, p_M$ , hàm mật độ xác suất của mô hình GMM chính là tổng trọng của  $M$  phân phối Gauss theo công thức:

$$P_{(GMM)}(\vec{x}) = \sum_{i=1}^M w_i p_i(\vec{x})$$

Trong đó  $w_i$  là trọng số của phân phối Gauss thứ  $i$ , thỏa mãn ràng buộc  $0 \leq w_i \leq 1$  và  $\sum_{i=1}^M w_i = 1$ . Các trọng số này thể hiện mức độ ảnh hưởng của mỗi phân phối Gauss đối với mô hình GMM. Như vậy, phân phối Gauss có phương sai và trọng số lớn bao nhiêu thì có mức độ ảnh hưởng lớn bấy nhiêu đối với kết xuất của mô hình.

Một mô hình GMM có  $M$  phân phối Gauss sẽ được biểu diễn bởi một bộ tham số  $\lambda = \{w_i, \mu_i, \Sigma_i\}, i \in [1, M]$ .

Tùy thuộc vào cách tổ chức của ma trận hiệp phương sai (covariance matrix), GMM có thể có một số biến thể khác nhau:

- Nodal covariance matrices GMM: mỗi phân phối Gauss trong GMM có một ma trận hiệp phương sai riêng.
- Grand covariance matrix GMM: mọi phân phối Gauss trong một GMM dùng chung một ma trận hiệp phương sai.
- Global covariance matrix GMM: mọi phân phối Gauss trong tất cả các GMM dùng chung một ma trận hiệp phương sai.

Ngoài ra, xét về dạng thức, ma trận hiệp phương sai gồm hai loại: full (dạng đầy đủ) và diagonal (dạng ma trận đường chéo). Thông thường, dạng nodal-diagonal covariance matrices GMM được sử dụng phổ biến nhất.

### 2.4.1.3. Áp dụng GMM cho bài toán ước lượng nền

#### a. Mô hình Gauss hỗn hợp

Tại thời điểm  $t$ , Giá trị của một điểm ảnh trong hệ màu RGB được biểu diễn bằng  $X_t$ . Như vậy ta có thể biểu diễn tập giá trị tại điểm ảnh  $\{x_0, y_0\}$  từ thời điểm 1 đến  $t$ :

$$A = \{X_1, X_2, \dots, X_t\} = \{I(x_0, y_0, i) : 1 \leq i \leq t\}$$

Tập  $A$  có thể được mô phỏng bởi mô hình hỗn hợp gồm  $k$  hàm phân phối Gauss:

$$p(X_T) = \sum_{i=1}^k w_{i,t} * N(X_t | \mu_{i,t}, \Sigma_{i,t})$$

Trong đó:

$$N(X_t | \mu_{i,t}, \Sigma_{i,t}) = \frac{1}{(2\pi)^{D/2} (\Sigma_{i,t})^{1/2}} \exp\left(-\frac{1}{2} (X_t - \mu_{i,t})' \Sigma_{i,t}^{-1} (X_t - \mu_{i,t})\right)$$

#### b. Điều chỉnh mô hình

Trong thực tế, ánh sáng trong khung cảnh có thể bị thay đổi dần dần (ban ngày, điều kiện thời tiết đối với khung cảnh ngoài trời...) hoặc thay đổi đột ngột (bật, tắt đèn đối với khung cảnh trong nhà...). Điều này làm cho các đối tượng mới (foreground) có thể được đưa vào cảnh hoặc các đối tượng hiện tại có thể bị mất đi. Để thích ứng với thay đổi này chúng ta có thể áp dụng thuật toán K-means để cập nhật các hàm phân phối Gauss thành phần.

- Nếu giá trị của một điểm ảnh mới  $X_{t+1}$  trùng khớp với một trong các hàm phân phối Gauss. Giả sử hàm phân phối Gauss trùng khớp là hàm phân phối thứ  $j$  thì giá trị trung bình và độ lệch chuẩn của hàm phân phối đó tại thời điểm  $t + 1$  sẽ được sửa lại như sau:

$$\mu_{j,t+1} = (1 - \rho) \mu_{j,t} + \rho X_{t+1}$$

$$\text{Và } \sigma_{j,t+1}^2 = (1 - \rho) \sigma_{j,t}^2 + \rho (X_{t+1} - \mu_{j,t+1})^2$$

Trong đó  $\rho = \alpha N(X_{t+1} | \mu_{j,t}, \sigma_{j,t}^2)$  và  $\alpha$  là tốc độ học.

- o Trọng số của tất cả các hàm Gauss trong hỗn hợp Gauss được sửa theo công thức dưới đây.

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

Với  $1 \leq i \leq M$ ,  $M_{i, t+1} = 1$  cho hàm phân phối trùng khớp với  $X_{t+1}$  và bằng 0 cho các hàm phân phối còn lại.

- Nếu  $X_{t+1}$  không trùng khớp với bất kỳ hàm phân phối nào trong hỗn hợp Gauss thì hàm phân phối có độ chắc chắn nhỏ nhất sẽ được thay thế bởi hàm phân phối mới có:

$$\mu_{t+1} = X_{t+1}, w_i \text{ là trọng số nhỏ nhất trước đó, } \sigma \text{ là giá trị khởi tạo.}$$

### c. Ước lượng nền

Thông thường các đối tượng foreground sẽ được đại diện bởi một số các phân phối có trọng số nhỏ. Vì vậy, chúng ta có thể xấp xỉ mô hình nền bởi  $b$  phân phối lớn nhất.

- Các phân phối Gauss sẽ được sắp xếp giảm dần theo  $w/\sigma$ .
- $b$  phân phối đầu tiên sẽ được chọn cho mô hình background:

$$B = \underset{b}{\operatorname{argmin}} \sum_{i=1}^b w_i > T$$

Trong đó  $T$  là giá trị nhỏ nhất được cho là background.

## 2.4.2. Mô hình RANSAC

### 2.4.2.1. Tổng quan về mô hình RANSAC

Giả sử Việt Nam có 19 người nghèo và 1 tỉ phú. Mỗi người nghèo có thu nhập 5 đồng, tỉ phú có thu nhập 1 tỉ. Nếu tính theo phương pháp trung bình cộng, thì thu nhập trung bình của dân Việt Nam vào khoảng 50 triệu. Nếu tính theo phương pháp trung vị, thì ra 5 đồng. Cách tính thứ 2 cho phép giảm nhiễu, và nhiễu ở đây chính là 01 người tỉ phú.

Phương pháp trung vị thật ra chỉ hạn chế chứ vẫn bao gồm cả nhiễu. Hầu hết giải thuật lập trình bình thường biết chỉ liên quan đến dữ liệu hoàn hảo, chính xác tuyệt đối, không chứa nhiễu (nói cách khác, coi nhiễu cũng là dữ liệu chính xác) [2]. Vì vậy, trong phần này tôi giới thiệu giải thuật khử nhiễu RANSAC, có tính ứng dụng cao làm nền tảng cho phân đoạn chuyển động video.

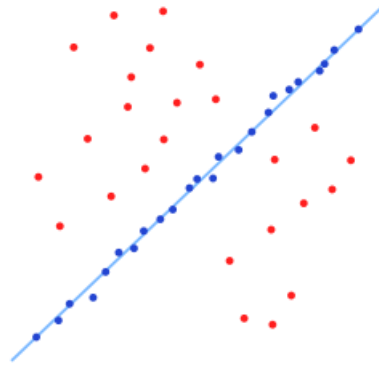


Figure 24: Ví dụ 1 về bài toán áp dụng RANSAC

### Khái quát về khái niệm mô hình

Về mặt toán học, mô hình là hình thức biểu diễn một cách tổng quát quan hệ giữa **các thuộc tính của cùng một phần tử**, thường bằng đẳng thức.

Ví dụ để biểu diễn tập hợp các điểm  $(x, y)$  nằm trên đường tròn tâm  $(a, b)$  bán kính  $r$ , ta dùng mô hình sau:

$$(x - a)^2 + (y - b)^2 = r^2$$

Mô hình trên được tham số hoá bởi bộ ba  $(a, b, r)$ . Tùy cách hiểu mà có mô hình khác nhau, ví dụ chỉ cần thay đổi hệ quy chiếu là ta có bộ ba  $(a, b, r)$  mới.

Việc tìm ra mô hình tổng quát dựa vào thuộc tính của các phần tử của tập hợp hữu hạn là nội suy, vì ta suy từ những cái cụ thể ra cái tổng quát, từ trong ra. Sau khi suy ra mô hình, không nhất thiết tất cả các phần tử đều phải khớp ngược trở lại vào mô hình. Ví dụ mô hình suy ra từ 19 người nghèo và tỉ phú là: Thu nhập = 5 [đồng], nhưng tỉ phú không khớp vào mô hình này.

Về mặt xác suất thống kê, khi kiểm tra phần tử có khớp vào mô hình hay không, ta không dùng dấu  $=$ , mà dùng dấu gần bằng, nghĩa là có xét đến nhiễu. Phần tử không khớp gọi là phần tử gây nhiễu (*outlier*)

#### 2.4.2.2. RANSAC là gì?

**RANSAC** là viết tắt của **RAN**dom **SAM**ple **C**onsensus.

##### + Xét bài toán

Để dễ hiểu, phần tiếp theo sử dụng bài toán cụ thể. Từ cụ thể ta suy ra tổng quát.

- Cho toạ độ của  $N$  điểm trên mặt phẳng.
- Giả sử có một số điểm nằm trên đường thẳng, còn các điểm khác là nhiễu (nghĩa là biết chắc có 2 tập hợp con, còn gọi là cluster).

- Hãy tìm phương trình đường thẳng, và cho biết những điểm nào được coi là nằm trên đường thẳng, những điểm nào được coi là nhiễu.



Figure 25: Ví dụ 1 về bài toán lọc nhiễu mô hình bằng RANSAC

Người ta phân biệt 2 loại cách giải: theo xác suất thống kê cổ điển (classical statistics) và theo xác suất thống kê chống nhiễu (robust statistics). Đặc điểm của các phương pháp theo xác suất thống kê cổ điển là luôn gồm luôn cả nhiễu, nên đáp số bị ảnh hưởng bởi nhiễu.

#### + Giải bằng phương pháp bình phương tối thiểu

Cách giải thuộc loại xác suất thống kê cổ điển nổi bật nhất có lẽ là phương pháp bình phương tối thiểu. Nó cho phép tìm ra một đường khớp nhất cho **tất cả** các điểm, với tổng các khoảng cách (sai số) từ từng điểm đến đường thẳng là nhỏ nhất. Tuy nhiên nếu dữ liệu chứa quá nhiều outlier (ví dụ một nửa số điểm), thì tất cả cách giải thuộc loại xác suất thống kê cổ điển đều cho kết quả sai lệch lớn.

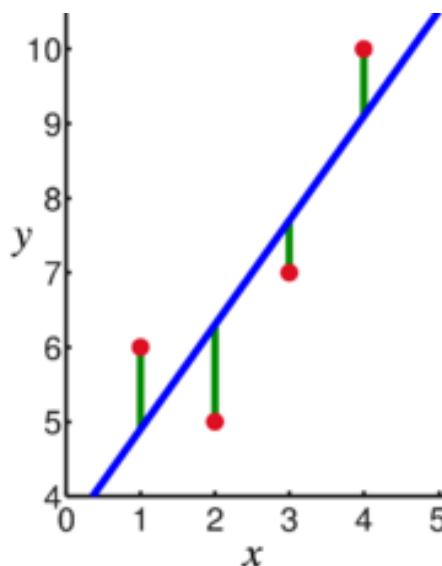


Figure 26: Kết quả ví dụ về bài toán lọc nhiễu mô hình bằng RANSAC

## + Thuật toán RANSAC

Bí quyết của các cách giải thuộc loại xác suất thống kê chống nhiễu là làm sao **loại bỏ outlier**, để **không sử dụng outlier khi tính toán mô hình**.

Giải thuật như sau:

### Đầu vào:

data - tập hợp các điểm  
k - số lần lặp  
t - ngưỡng (threshold) sai số để xác định điểm nào đó có khớp mô hình không

### Đầu ra:

best\_model - mô hình tốt nhất  
best\_consensus\_set - tập hợp các điểm khớp với best\_model

best\_model = nil

best\_consensus\_set = nil

best\_num\_points = 0

loop k lần

consensus\_set = tập hợp 2 điểm ngẫu nhiên thuộc data

model = mô hình đường thẳng suy ra từ 2 điểm trên

với mỗi điểm point thuộc data nhưng không thuộc consensus\_set

distance = khoảng cách từ point đến đường thẳng

if distance < t (point được coi là khớp với model nếu sai số nhỏ hơn t)

thêm point vào consensus\_set

num\_points = số lượng phần tử trong consensus\_set

if num\_points > best\_num\_points

best\_model = model

best\_consensus\_set = consensus\_set

best\_num\_points = num\_points

return best\_model và best\_consensus\_set

Trong giải thuật, ta chọn ra 2 điểm ngẫu nhiên, vì 2 là số lượng điểm tối thiểu để có thể tính toán mô hình đường thẳng. Với bài toán tổng quát, ta cần chọn ra số phần tử tối thiểu để có thể tính toán mô hình.

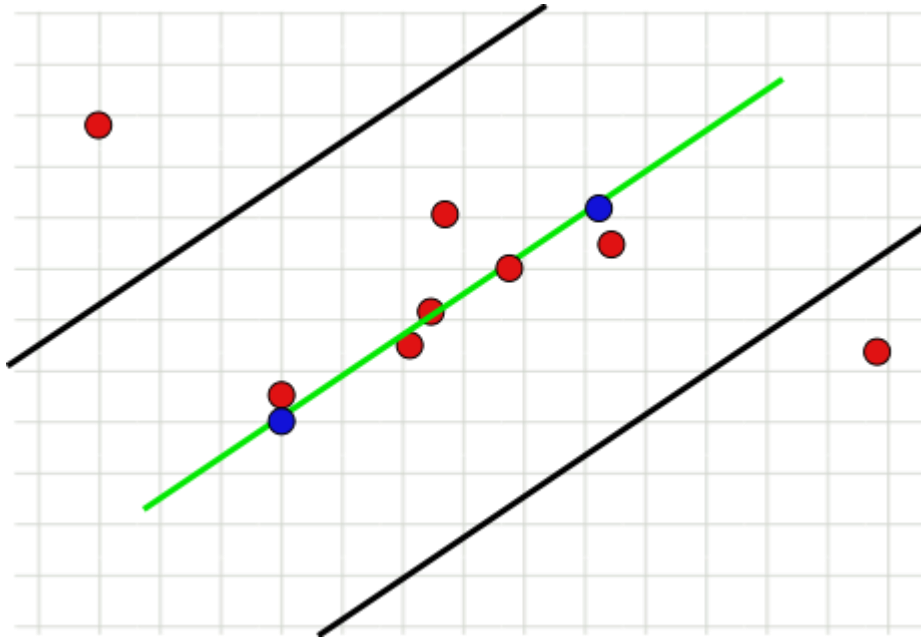


Figure 27: Kết quả của ví dụ về bài toán lọc nhiễu mô hình bằng RANSAC

### 2.4.2.3. Áp dụng RANSAC cho bài toán phân đoạn chuyển động

Các nghiên cứu [11] đề xuất việc áp dụng kỹ thuật luồng quang học để tìm ra trường chuyển động giữa ảnh thu được từ hai thời điểm.

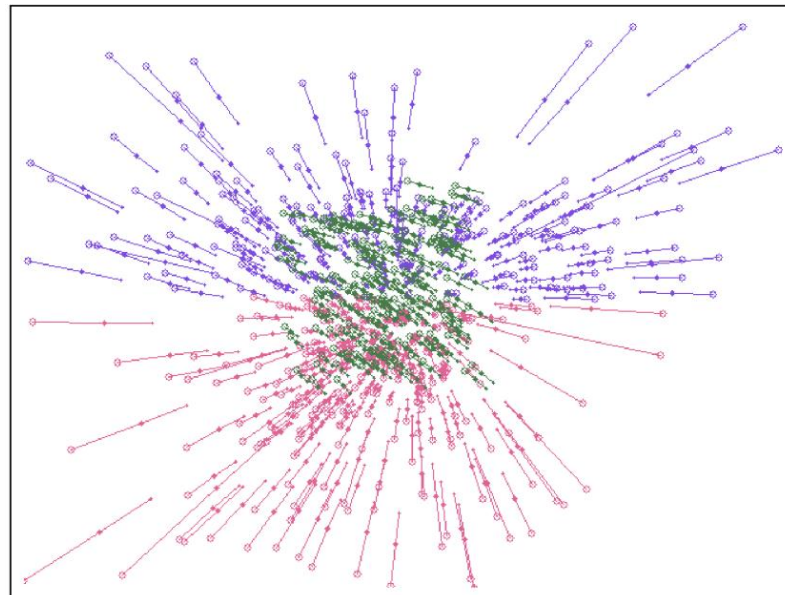


Figure 28: Trường chuyển động khi ô-tô rẽ trái

Từ trường chuyển động, chúng ta trích xuất ra được các véc-tơ chuyển động.



Ảnh thu nhận khi xe máy đi thẳng



Các véc-tơ chuyển động trích xuất được



Khi 2 xe đạp đi ngược chiều nhau



Các véc-tơ chuyển động trích xuất được

Điểm mấu chốt để ta áp dụng RANSAC trong bài toán phân đoạn chuyển động là ***dữ liệu thu nhận được luôn bao gồm nhiễu***

Và RANSAC giúp chúng ta **loại bỏ nhiễu** trong mô hình đó.

Quay trở lại với ví dụ trên, ta nhận thấy:

- Việc thu nhận hình ảnh từ camera thường có nhiễu. Từ đó, kết quả của quá trình tính toán trường chuyển động để tìm ra véc-tơ chuyển động bị ảnh hưởng.
- Thuật toán luồng quang học là một phép ước lượng, vì vậy kết quả luôn luôn bao gồm nhiễu.
- Nhiễu trong ví dụ trên tồn tại trong: **Véc-tơ dịch chuyển**

Vì vậy, chúng ta sẽ áp dụng mô hình RANSAC để loại bỏ các véc-tơ dịch chuyển là nhiễu trong mô hình.



Trở lại ví dụ ban đầu, trường chuyển động thu được khi ô-tô rẽ trái như sau:

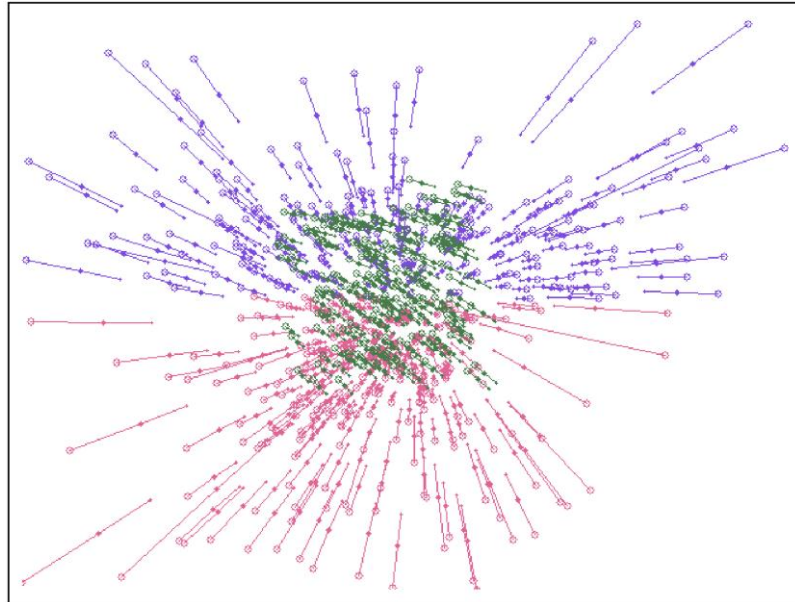


Figure 29: Ví dụ véc-tơ chuyển động được trích ra khi ô-tô rẽ trái

Trong số các véc-tơ chuyển động thu được, ta dùng mô hình RANSAC để lọc nhiễu, và thu được:

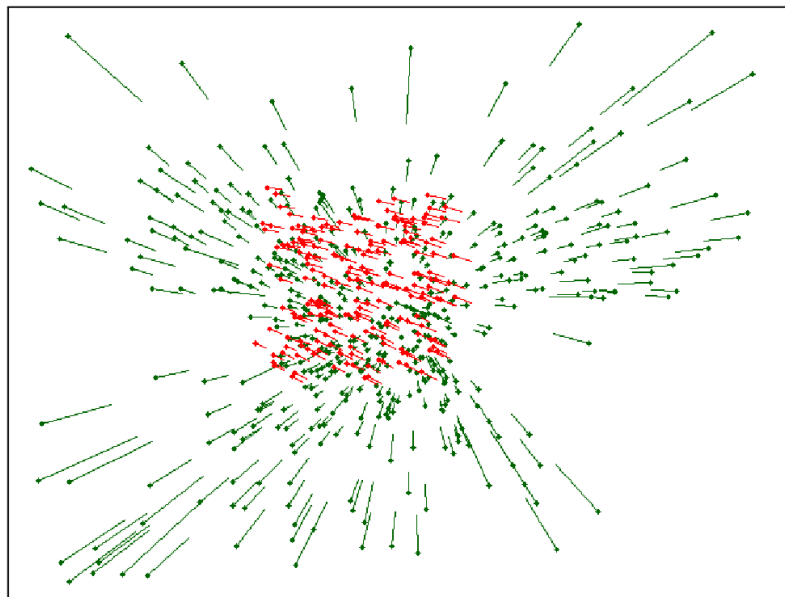


Figure 30: Kết quả lọc sử dụng RANSAC cho ví dụ véc-tơ chuyển động được trích ra khi ô-tô rẽ trái

Với các véc-tơ **màu xanh** là nhiễu được loại bỏ, giữ lại những véc-tơ **màu đỏ** là véc-tơ chuyển động thực tế thu được khi chiếc ô-tô rẽ trái.

## 2.5. Một số kiến thức nền tảng hỗ trợ cho quá trình nghiên cứu

### 2.5.1. Một số bộ lọc trong xử lý tín hiệu số

Với kỹ thuật lọc trong xử lý tín hiệu số, người ta thường phân ra 2 loại bộ lọc:

- *Bộ lọc tuyến tính (Kalman Filter, Median Filter...)*
- *Bộ lọc phi tuyến (Extended Kalman Filter, Unscented Kalman Filter, Particle Filter...)*

Với kỹ thuật lọc, thường được phân tách ra thành 2 loại:

- *Lọc đơn mô hình: Duy trì một bộ lọc cho quá trình xử lý của hệ thống*
- *Lọc đa mô hình: Duy trì nhiều hơn 1 bộ lọc tại một thời điểm xử lý của hệ thống*

Với bài toán theo dõi đối tượng chuyển động đơn giản, chúng ta thường sử dụng kỹ thuật lọc đơn mô hình (Sử dụng một loại bộ lọc tuyến tính hoặc lọc phi tuyến)

Tuy nhiên, trong thực tế với bài toán theo dõi đối tượng chuyển động phức tạp như bài toán theo dõi đối tượng chuyển động Radar, người ta thường duy trì nhiều bộ lọc (sử dụng cả hai loại bộ lọc tuyến tính và phi tuyến) trong một mô hình. [75]

Trong phần này, tôi đưa ra nguyên lý và một số vấn đề cơ bản của 3/4 loại bộ lọc phổ biến, đồng thời đưa ra so sánh và đánh giá mức độ phù hợp của từng bộ lọc với các bài toán cụ thể.

### 2.5.2. Tổng quan về các kỹ thuật lọc

Với một hệ thống bất kỳ, trạng thái hệ thống là điều mà chúng ta quan tâm, cũng là trung tâm của các bài toán cần giải quyết.

**Ví dụ:**

- Với hệ thống giám sát tại Sân bay, trạng thái của hệ thống là thông tin về các đối tượng hiện hữu trong khung hình của camera (Bao gồm: Toạ độ, kích thước, vận tốc di chuyển, gia tốc...)
- Với hệ thống radar, trạng thái của hệ thống là độ phương vị của đối tượng (Toạ độ tương đối so với radar và trạng thái góc của đối tượng...) [50]

Trạng thái hệ thống được chúng ta mô hình hoá<sup>1</sup> tương ứng với từng bài toán nhất định. Trạng thái hệ thống phải đảm bảo các yếu tố sau:

- *Miêu tả đủ các thuộc tính phục vụ cho yêu cầu bài toán*

---

<sup>1</sup> Trạng thái của hệ thống được mô hình hoá theo từng bài toán nhất định. Ví dụ: Với bài toán theo dõi đối tượng chuyển động, ta sẽ mô hình hoá trạng thái hệ thống đơn giản nhất theo phương trình  $x(t) = x_0 + v * t$

Việc mô hình hoá hệ thống, chưa có cơ sở toán học thực tế để chứng minh tính đúng đắn của mô hình đó, vì vậy tất cả các mô hình hệ thống đều được thực hiện dựa trên cơ sở thực nghiệm

- Đặc tả tốt nhất cho hệ thống thực tế sau khi được mô hình hoá
- Có thể triển khai được các bộ lọc

Trạng thái hệ thống được chia thành 2 loại chính:

- Trạng thái rời rạc theo thời gian
- Trạng thái liên tục theo thời gian

Trong giới hạn của đề tài này và kỹ thuật phân đoạn chuyển động video, tôi chỉ đưa ra các nghiên cứu và đánh giá về kỹ thuật lọc với trạng thái rời rạc theo thời gian.

### 2.5.2.1. Mô hình không gian trạng thái rời rạc theo thời gian

$$x_k \sim p(x_k | x_{k-1})$$

$$z_k \sim p(z_k | x_k)$$

**Trong đó:**

$x_k \in R^n$  là trạng thái của hệ thống tại thời điểm k

$z_k \in R^m$  là kết quả đo đạt được của hệ thống tại thời điểm k

$p(x_k | x_{k-1})$  là mô hình động thể hiện các đặc tính của hệ thống, cũng như cách mà hệ thống thay đổi theo thời gian

$p(z_k | x_k)$  là mô hình cho quá trình đo đạc, mô tả cách mà các đo đạc phân bố (phụ thuộc) vào trạng thái hệ thống

Một hệ thống được định nghĩa theo cách trên, được gọi là thuộc tính Markov. Có nghĩa là trạng thái  $x_k$  dựa trên  $x_{k-1}$  và độc lập với các trạng thái và đo đạc trong quá khứ.

$$p(x_k | x_{1:k-1}, y_{1:k-1}) = p(x_k | x_{k-1})$$

### 2.5.2.2. Ước lượng trạng thái tuyến tính

Trạng thái tuyến tính, được thể hiện qua phương trình sau

$$x_k = A_{k-1}x_{k-1} + w_{k-1}$$

$$z_k = H_k x_k + v_k$$

**Trong đó:**

- $x_k \in R^n$  là trạng thái của hệ thống tại thời điểm k
- $z_k \in R^m$  là kết quả đo đạt được của hệ thống tại thời điểm k
- $w_{k-1} \sim N(0, Q_{k-1})$  là nhiễu hệ thống tại thời điểm k-1
- $v_k \sim N(0, R_k)$  là nhiễu đo đạc tại thời điểm k
- $A_{k-1}$  là ma trận chuyển trạng thái của hệ thống
- $H_k$  là ma trận của mô hình đo đạc

### 2.5.2.3. Tiêu chí đánh giá các bộ lọc

Để đánh giá một bộ lọc, hiển nhiên chúng ta sẽ quan tâm đến độ chính xác của bộ lọc. Và để đánh giá độ chính xác của một bộ lọc, ta sẽ sử dụng tiêu chuẩn MSE (mean-square error). [50]

Phương pháp đánh giá : Sinh ra một tập dữ liệu mẫu phù hợp với hệ thống. Thêm nhiễu và đưa vào hệ thống xử lý. Kết quả lấy sự sai biệt của trạng thái hệ thống xử lý với trạng thái của hệ thống ước đoán được – lấy làm tiêu chí để đánh giá một bộ lọc.

Bộ lọc nào có MSE càng thấp, chứng tỏ độ lệch giữa giá trị thực tế với giá trị ước lượng gần sát nhau. Vì vậy bộ lọc đó đáp ứng được yêu cầu của hệ thống.

### 2.5.3. Bộ lọc Kalman

Bộ lọc Kalman được R. E. Kalman giới thiệu lần đầu tiên vào năm 1960, trong một bài báo tại hội thảo khoa học của ông [76]. Mục tiêu của lọc Kalman rời rạc theo thời gian là đưa ra công thức đệ quy cho quá trình ước lượng của các hệ thống rời rạc.

Lọc Kalman rời rạc là một bộ lọc tuyến tính. Trong giới hạn bài báo cáo của tôi, lý do để nêu lên Kalman rời rạc là một bộ lọc tuyến tính, sẽ được trình bày ở phần sau.

Tư tưởng cơ bản của lọc Kalman gần giống với việc xấp xỉ đường cong của các điểm dữ liệu, sử dụng phương pháp cực tiểu bình phương xấp xỉ. Và cho phép dự đoán trạng thái tại thời điểm kế tiếp. [76]

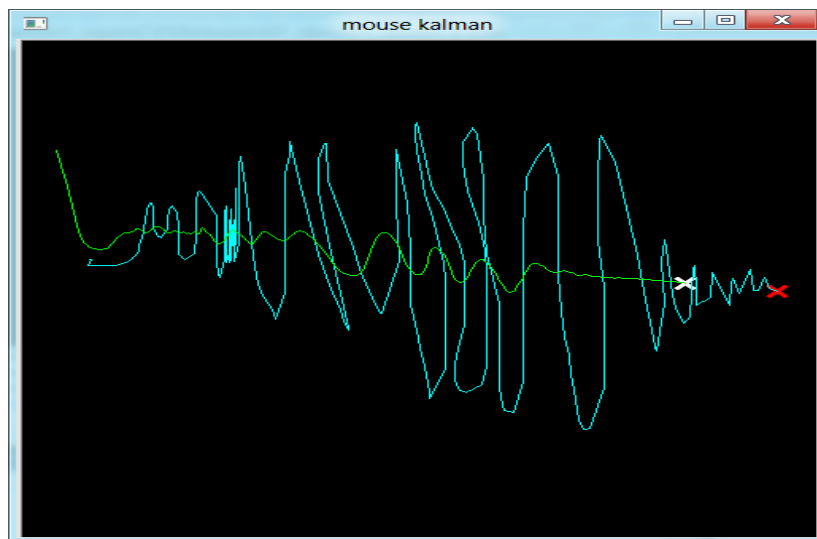


Figure 31: Ứng dụng lọc Kalman để lọc tín hiệu thu được từ tọa độ con trỏ của người dùng

### 2.5.3.1. Bản chất của bộ lọc Kalman

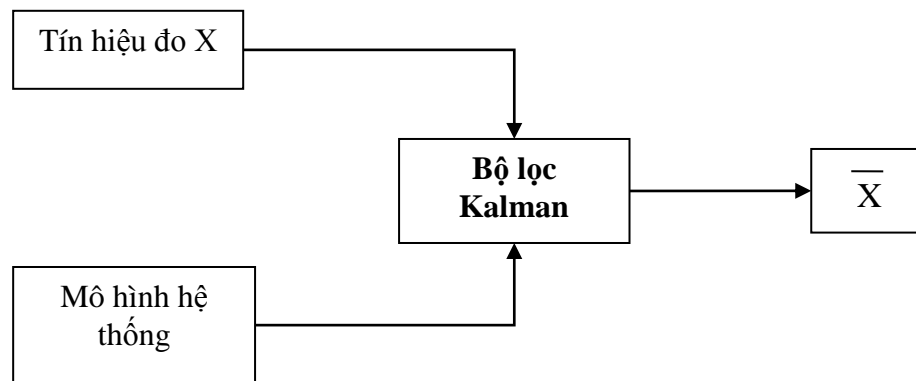


Figure 32: Mô hình hoạt động của mạch lọc Kalman

Chúng ta có tín hiệu đo được và mô hình của tín hiệu đo được (đòi hỏi tuyến tính) và sau đó là áp dụng vào trong hệ thống phương trình của mạch lọc để ước lượng trạng thái quan tâm. Thực ra tín hiệu đo là không khó, phương trình đã có sẵn, cái chúng ta cần chính là mô hình hoá hệ thống [50]. Để có thể ứng dụng một cách hiệu quả mạch lọc Kalman thì chúng ta phải mô hình hóa được một cách tuyến tính sự thay đổi của trạng thái cần ước lượng (estimate) hoặc ước đoán (predict).

### 2.5.3.2. Phương trình Kalman rời rạc

#### a. Quá trình ước đoán

Bộ lọc Kalman đề cập đến bài toán tổng quát đi ước lượng trạng thái của một quá trình được mô hình hóa một cách rời rạc theo thời gian bằng một phương trình ngẫu nhiên tuyến tính như sau:

$$x_k = Ax_{k-1} + w_{k-1}$$

và kết quả đo đạc

$$z_k = H^* x_k + v_k$$

Trong đó  $w$  và  $v$  là hai vecto biến ngẫu nhiên đại diện cho nhiễu hệ thống và nhiễu đo đạc.

Hai biến ngẫu nhiên này độc lập và giả sử tuân theo luật phân phối chuẩn, với giá trị trung bình = 0 và ma trận hiệp biến (covariance) lần lượt là  $Q$  và  $R$

$$p(w) \sim N(0, Q)$$

$$p(v) \sim N(0, R)$$

Nếu vector trạng thái  $x$  có kích thước là  $n$ , thì ma trận  $A$  sẽ có kích thước là  $n \times n$ .  $B$  ( $n \times 1$ ) là ma trận phụ thuộc vào điều khiển tối ưu  $u$  với  $u$  là vector có kích thước là 1.

Vector đo đạc  $z$  có kích thước là  $m$  nên ma trận  $H$  sẽ là  $m \times n$ . Chú ý rằng các ma trận  $Q$ ,  $R$ ,  $A$ ,  $H$  có thể thay đổi theo thời gian (từng bước  $k$ ).

Đến đây ta thấy bài toán lọc Kalman chính là đi tìm giá trị ước lượng và ước đoán của trạng thái  $x$  khi ta biết được sự biến thiên của nó và ta đo được một đại lượng  $z$  mà phụ thuộc tuyến tính vào  $x$ . [75]

## b. Các tính toán của bộ lọc

Nếu ta giả sử  $\bar{x}_k^-$  và  $\bar{x}_k$  lần lượt là tiên nghiệm và hậu nghiệm ước lượng của giá trị  $x$  tại thời điểm  $k$ . Giá trị tiên nghiệm thu được chỉ dựa vào mô hình hệ thống (1), còn giá trị hậu nghiệm là giá trị thu được sau khi đã có kết quả đo đạc  $z_k$  (2). Khi đó sai số ước đoán ước đoán tiên nghiệm và hậu nghiệm lần lượt là:

$$e_k^- = x_k - \bar{x}_k^-$$

$$e_k = x_k - \bar{x}_k$$

Ước tính ma trận hiệp phương sai lỗi tiên nghiệm.

$$P_k^- = E[e_k^- e_k^{-T}]$$

$e^{-T}$  là ma trận chuyển vị của  $e^-$

Ước tính ma trận hiệp phương sai lỗi hậu nghiệm.

$$P_k = E[e_k e_k^T]$$

Chúng ta bắt đầu với mục tiêu tìm kiếm một phương trình tuyến tính của một trạng thái tiên nghiệm ước tính và sai số của giá trị đo đạc và giá trị đo đạc ước đoán để ước tính trạng thái hậu nghiệm:

$$\bar{x}_k = \bar{x}_k^- + K(z_k - H * \bar{x}_k^-)$$

$K$  gọi là độ lợi (gain)

$\bar{x}_k$  : Giá trị hậu nghiệm của ước lượng  $x$

$\bar{x}_k^-$  : Giá trị tiên nghiệm của ước lượng  $x$

$z_k$  : giá trị đo đạc

$H\bar{x}_k^-$  : giá trị đo đạc ước đoán

Vậy làm sao để chọn được  $K$  tối ưu nhất ? Có nghĩa là covariance của sai số ước lượng hậu nghiệm  $e_k = x_k - \bar{x}_k$  là nhỏ nhất. Bằng cách thay  $e_k$  vào biểu thức tính  $P_k$  rồi sau đó lấy đạo hàm của  $P_k$  theo  $K$ , ta sẽ tìm được  $K$  tương ứng với  $P_k$  nhỏ nhất.

$$K_k = P_k^- * H^T (H * P_k^- * H^T + R)^{-1}$$

$K_k$  thay đổi trong thời gian  $k$  và chính là độ lợi cần tìm của mạch lọc Kalman trong mỗi ước đoán.

### c. Các xác suất của bộ lọc

Các chứng minh cho (5) bắt nguồn từ khả năng dự đoán trước một sự kiện dựa trên tất cả các phép đo trước.

$$E[x_k] = \bar{x}_k$$

$$E[(x_k - \bar{x}_k) * [(x_k - \bar{x}_k)^T] = P_k$$

Dự đoán trạng thái hậu nghiệm (5) phản ánh giá trị trung bình của trạng thái tuân theo quy luật phân phối chuẩn nêu như (3) và (4) được đáp ứng. Hiệp phương sai lỗi hậu nghiệm ước tính phản ánh sự khác biệt về sự phân bố trạng thái. Nói cách khác

$$P(x_k / z_k) \sim N(E[x_k], E[(x_k - \bar{x}_k) * [(x_k - \bar{x}_k)^T] = P_k)$$

#### 2.5.3.3. Thuật toán bộ lọc Kalman rời rạc

Bộ lọc Kalman ước một trạng thái bằng cách sử dụng kiểm soát thông tin phản hồi dưới dạng phép đo. Như vậy các phương trình cho bộ lọc Kalman rơi vào hai nhóm: phương trình cập nhật thời gian và phương trình cập nhật đo lường. Phương trình cập nhật thời gian có trách nhiệm tham chiếu đến các trạng thái hiện tại và hiệp phương sai lỗi ước tính để có được ước lượng trước cho thời gian tới. Phương trình cập nhật đo lường chịu trách nhiệm về thông tin phản hồi, tức là tích hợp một phép đo mới vào sự tiên đoán để cải thiện một ước tính hậu nghiệm.

Phương trình cập nhật thời gian cũng có thể xem như một phương trình dự báo. Phương trình cập nhật phép đo có thể xem là phương trình sửa.

Tóm lại mạch lọc Kalman gồm 2 bước:

- **Time update:** Ước đoán trạng thái tiên nghiệm
- **Measurement update:** Dựa vào kết quả đo để hiệu chỉnh lại ước đoán.

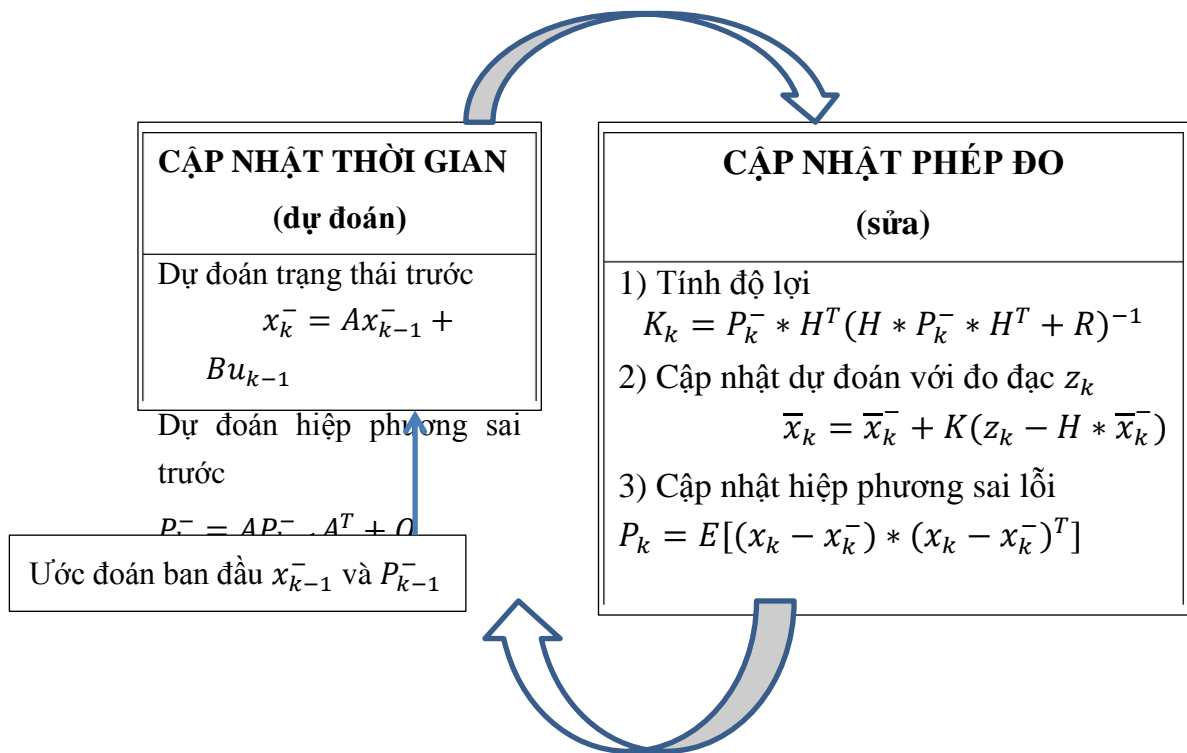


Figure 33: Hình trên tóm tắt cơ bản của mạch lọc Kalman

#### 2.5.3.4. Đánh giá lọc Kalman

- **Ưu điểm:**
  - Dễ triển khai trên các thiết bị hạn chế về tài nguyên
  - Độ phức tạp thuật toán thấp, phù hợp với các bài toán phải xử lý nhiều đối tượng
  - Kalman filter hoạt động tốt (có thể nói là lý tưởng) trong trường hợp có nhiễu Gauss và trạng thái của hệ biến đổi tuyến tính.
- **Nhược điểm:**
  - Chỉ hoạt động tốt trong bài toán có trạng thái biến đổi tuyến tính và có nhiễu Gauss. Trên thực tế, chỉ có một phần nhỏ các hệ thống có các yêu cầu trên. Vì vậy, Kalman Filter không được áp dụng rộng rãi như các bộ lọc phi tuyến khác. [76]

#### 2.5.4. Kỹ thuật tương quan dữ liệu

##### 2.5.4.1. Vai trò của tương quan dữ liệu trong bài toán theo dõi đa đối tượng chuyển động

Đối với việc theo dõi/phân đoạn một đối tượng chuyển động nhờ vào kỹ thuật xử lý ảnh và bộ lọc có thể đo đạc và mô hình hóa được vị trí của đối tượng tại thời điểm tiếp theo. Nhưng bài toán trở nên khó khăn với việc theo dõi nhiều đối tượng cùng một lúc bởi việc xử lý như trên chỉ cho biết vị trí mới của các đối tượng tại thời điểm sau mà khó xác định được danh tính của đối tượng đó. Chính bởi thế, với  $n$  đối tượng tại thời gian  $t$ ,



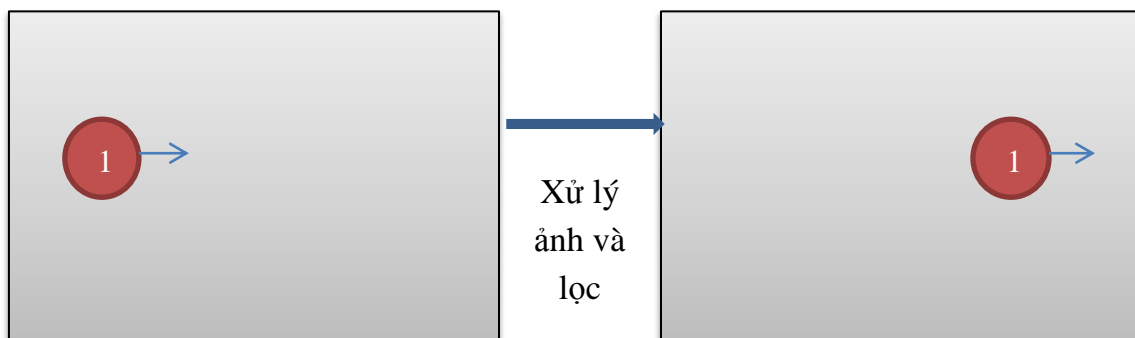
tại thời gian  $t+1$  ta sẽ thu được  $m$  đối tượng mới, điều đó tương đương với việc ta phải duy trì và làm việc trên  $m!$  khả năng về danh tính của các đối tượng điều này là không khả thi.

Kỹ thuật tương quan dữ liệu sẽ giúp giải quyết khó khăn về việc xác định chính xác danh tính của các đối tượng. [76]

Một số kỹ thuật tương quan dữ liệu tiêu biểu:

- Kỹ thuật đơn giả thuyết (Single Hypothesis Tracking)
  - o GNN: Sử dụng các thuật toán như Hungary, Munkres, ...
  - o JPDA
- Kỹ thuật đa giả thuyết (Multiple Hypothesis Tracking)

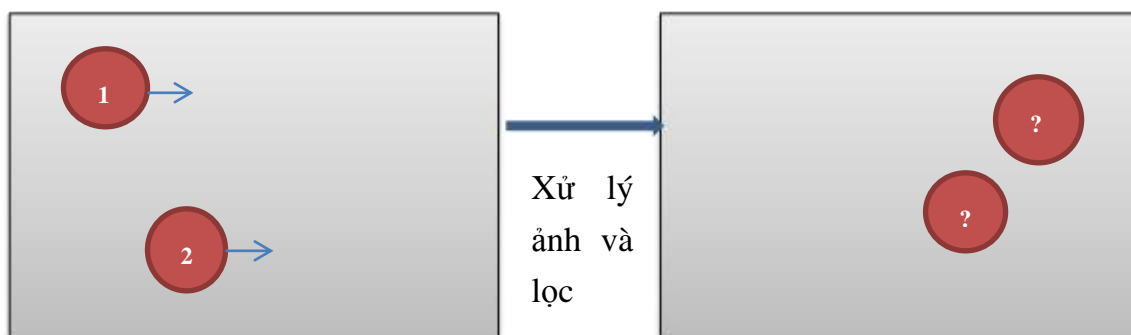
*Trong phạm vi bài báo, tôi chỉ trình bày kỹ thuật sơ khai nhất là GNN sử dụng thuật toán Hungary.*



*1a: vị trí đối tượng chuyển động với vận tốc  $v$  tại thời điểm  $t$*

*1b: vị trí đối tượng chuyển động với vận tốc  $v$  tại thời điểm  $t+1$*

**Hình 1: Hình ảnh minh họa theo dõi 1 đối tượng chuyển động.**



**Figure 34: Hình ảnh minh họa theo dõi 2 đối tượng chuyển động bằng kỹ thuật xử lý ảnh và bộ lọc**

*2a: vị trí các đối tượng chuyển động với vận tốc  $v$  tại thời điểm  $t$*

*2b: vị trí các đối tượng chuyển động với vận tốc  $v$  tại thời điểm  $t+1$*

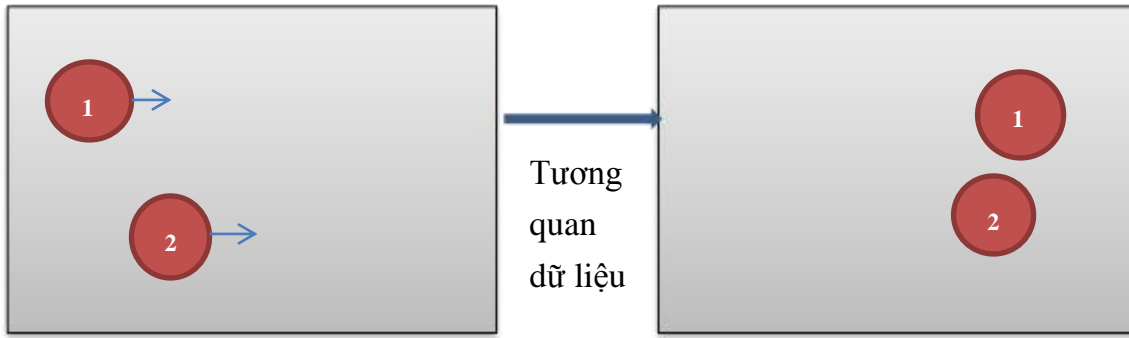


Figure 35: Hình ảnh minh họa theo dõi 2 đối tượng chuyển động bằng kỹ thuật tương quan dữ liệu.

3a: vị trí các đối tượng chuyển động với vận tốc  $v$  tại thời điểm  $t$

3b: vị trí các đối tượng chuyển động với vận tốc  $v$  tại thời điểm  $t+1$

#### 2.5.4.2. Kỹ thuật tương quan dữ liệu ứng dụng trong phân tách đa đối tượng chuyển động.

Kỹ thuật tương quan dữ liệu được ứng dụng trong việc theo dõi đa đối tượng chuyển động với bài toán cụ thể như sau:

Bộ theo dõi các đối tượng chuyển động được gắn vào đối tượng sau khi phát hiện được đối tượng. Tại mỗi thời điểm  $t+1$  (sau thời điểm  $t$ ) ta sẽ phải tính xác suất để xác định danh tính của  $m$  đối tượng so với thời điểm  $t$  trước đó. Ta sẽ thu được bảng xác suất với  $n \times m$  cặp giá trị xác suất với mỗi cặp thể hiện khả năng để đối tượng thứ  $i$  ban đầu sẽ trở thành đối tượng  $j$  tại thời điểm sau. Từ bảng xác suất có được ta sử dụng các thuật toán như Hungary, .... Để xác định danh tính cho các đối tượng.

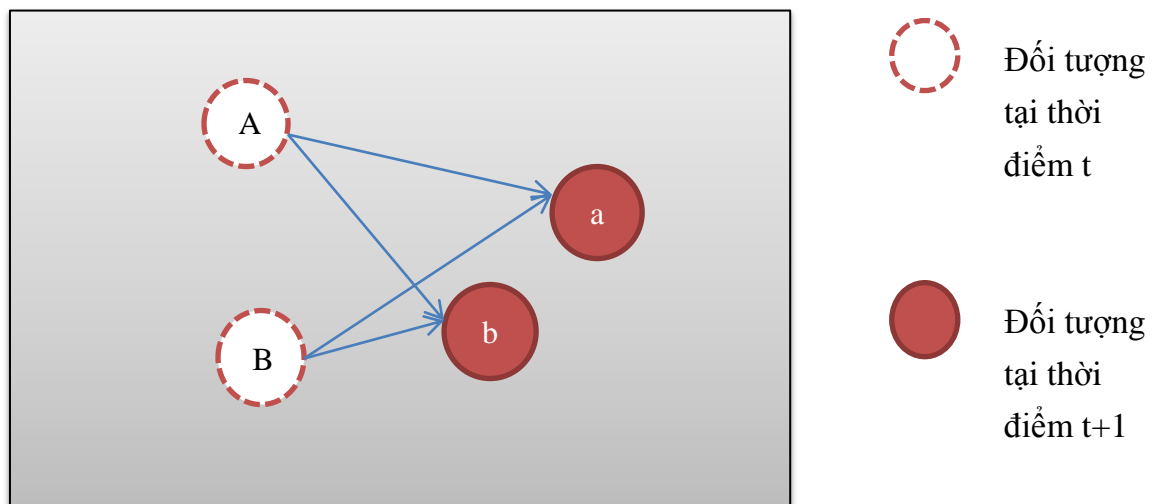


Figure 36: Mô tả các khả năng ghép đối tượng

- Giả sử thu được bảng xác suất ghép các đối tượng:

Bảng 1: Xác suất ghép cặp đối tượng

	<b>a</b>	<b>B</b>
<b>A</b>	80%	20%
<b>B</b>	41%	59%

- Sử dụng thuật toán Hungary ta được kết quả lựa chọn

Bảng 2: Bảng kết quả ghép cặp

	<b>a</b>	<b>B</b>
<b>A</b>	80%	20%
<b>B</b>	41%	59%

Kết quả thu được xác định tại thời điểm  $t+1$ :

- Đối tượng **a** là **A** tại thời điểm  $t$
- Đối tượng **b** là **B** tại thời điểm  $t$

#### 2.5.4.3. Thuật toán Hungary

Thuật toán được sử dụng nhằm để giải quyết các bài toán tìm bộ ghép cực đại với trọng số cực tiểu trên đồ thị hai phía, với dạng toán cụ thể là bài toán phân công.

Cho  $n$  người thợ và  $m$  công việc, người thứ  $i$  sẽ giải quyết công việc thứ  $j$  với chi phí là  $C_{ij} \geq 0$ . Cần phân cho mỗi thợ một công việc sao cho tổng chi phí phải trả để thực hiện  $m$  công việc này là nhỏ nhất. Thuật toán thực hiện lựa chọn các bộ ghép người và công việc tối ưu trên ma trận vuông  $N \times N$  với các bước thực hiện.

**Bước 1:** Tìm  $N$  với:  $N = \text{MAX}(n, m)$ ;

**Bước 2:** Biểu diễn ma trận  $N \times N$ , với việc bổ sung  $N-n$  hàng hoặc  $N-m$  cột để tạo ma trận vuông với giá trị  $C_{xy} = \text{MAX}(C_{ij} \mid i = 1, n; j = 1, m)$ .

**Bước 3:** Tìm giá trị nhỏ nhất của mỗi hàng và lấy hiệu của mỗi vị trí trên hàng với giá trị nhỏ nhất của hàng đó.

**Bước 4:** Tìm giá trị nhỏ nhất trên mỗi cột và lấy hiệu của mỗi vị trí trên cột đó với giá trị nhỏ nhất của cột đó.

**Bước 5:** Chọn trên mỗi hàng và cột một vị trí 0 là duy nhất, nếu có nhiều hơn 1 ta chọn và loại bỏ các vị trí 0 khác.

**Bước 6:**

- Nếu tồn tại ít nhất một hàng chưa chọn được giá trị 0 thì:

Đánh dấu các hàng chưa chọn được vị trí 0, trên hàng sẽ có ít nhất một số 0 bị loại bỏ, đánh dấu cột chứa các số 0, trên cột có thể có những vị trí 0 khác, ta tiếp tục đánh dấu các hàng có số 0 khác đó và lại tìm trên hàng vừa đánh dấu. Việc đánh dấu hàng và cột lặp đi lặp lại cho tới khi không tồn tại vị trí 0 nào có thể dẫn tới việc đánh dấu 1 dòng hoặc cột khác.

- Nếu các hàng đều chọn được thì đưa ra danh sách phân công rồi kết thúc bài toán.

**Bước 7:** Vạch những cột bị đánh dấu và các hàng không bị đánh dấu.

**Bước 8:** Tìm giá trị nhỏ nhất của các ô không bị vạch, tính mới giá trị các ô không bị vạch bằng việc lấy hiệu giá trị của ô đó với giá trị nhỏ nhất vừa tìm được. Sau đó, chuyển về bước 5.

### Ví dụ minh họa:

Cho 3 thợ và 3 công việc, với chi phí mà phải trả cho các công nhân nếu thuê họ thực hiện cho mỗi công việc được thể hiện bằng bảng:

	Công việc 1	Công việc 2	Công việc 3
Người thợ 1	12	34	15
Người thợ 2	22	21	18
Người thợ 3	11	21	26

Bảng 3: Bảng phân công m công việc cho n công nhân

Bước 1: Bỏ qua vì  $n = m = 3$

Bước 2: Biểu diễn ma trận:

12	34	15
22	21	18
11	21	26

Bảng 4: Biểu diễn ma trận của thuật toán Hungary

Bước 3: Lấy hiệu trên hàng của các vị trí trên hàng và số trừ là giá trị nhỏ nhất của hàng. (Thu được giá trị 0 trên hàng i là công việc mà người thứ i làm với chi phí nhỏ nhất.)

0	22	3
4	3	0
0	10	15

Bước 4: Lấy hiệu trên cột của các vị trí trên cột và số trừ là giá trị nhỏ nhất trên cột đó. ( Thu được các vị trí 0 mới trên các cột ban đầu không chứa giá trị 0, các vị trí 0 trên cột thứ j là những người làm công việc j với chi phí nhỏ nhất).

0	19	3
4	0	0
0	7	15

Bước 5: Chọn trên mỗi hàng và cột một vị trí 0 là duy nhất, nếu có nhiều hơn 1 ta chọn và loại bỏ các vị trí 0 khác. ( Chọn cho mỗi người làm một công việc)

0	19	3
4	0	0
0	7	15

Bảng 5: Kết quả phân công cho bài toán giao m công việc cho n công nhân

(Chú thích: ô đỏ là vị trí được chọn và ô màu xám là những ô loại bỏ, hàng không có ô đỏ là hàng mà chưa chọn được công việc)

Bước 6:

- Vì có một hàng chưa chọn được giá trị 0 nên:

Đánh dấu các hàng chưa chọn được vị trí 0, trên hàng sẽ có ít nhất một số 0 bị loại bỏ, đánh dấu cột chứa nó, trên cột có thể có những vị trí 0 khác, ta tiếp tục đánh dấu các hàng có số 0 khác đó và lại tìm trên hàng vừa đánh dấu. Việc đánh dấu hàng và cột lặp đi lặp lại cho tới khi không tồn tại vị trí 0 nào có thể dẫn tới việc đánh dấu 1 dòng hoặc cột khác.

✓			
0	19	3	✓
4	0	0	
0	7	15	✓

Bước 7: Vạch những cột bị đánh dấu và các hàng không bị đánh dấu. (Nhưng ô không bị kẻ qua là những ô mà ta có thể tìm ra 1 cặp người thợ và công việc có chi phí nhỏ nhất để có thể bổ sung vào danh sách phân công.)

	✓			
0		19	3	✓
4	0	0		
0		7	15	✓

Bước 8: Tìm cặp người thợ và công việc mà cho chi phí nhỏ nhất trên các ô chi phí không bị vạch tại bước 7, sau đó lấy hiệu của các ô đó với chi phí nhỏ nhất vừa tìm được.  
Chuyển tới bước 5

0	16	0
4	0	0
0	4	12

Ví dụ quay lại bước 5 chọn được kết quả:

0	16	0
4	0	0
0	4	12

Bước 6: Đưa ra danh sách phân công. (vì bước 5 chọn được công việc trên mỗi dòng)

Thợ 1 làm công việc 3

Thợ 2 làm công việc 2

Thợ 3 làm công việc 1

Với tổng chi phí là : 48

## Chương 3: Một số đề xuất và các bước thực hiện

### 3.1. Đề xuất thứ nhất – cải tiến kỹ thuật luồng quang học

#### 3.1.1. Tổng quan kỹ thuật cải tiến luồng quang học

Như chúng ta đã biết, kỹ thuật luồng quang học giúp chúng ta có thể tính được vận tốc của các điểm ảnh trên bề mặt đối tượng. Nó dựa trên ý tưởng “**Bề mặt ngoài của đối tượng không có nhiều thay đổi về cường độ sáng khi xét sự thay đổi từ thời điểm  $t$  sang thời điểm  $t + 1$** ”. Nhưng trong thực tế điều kiện ánh sáng, và nhiễu của thiết bị thu nên cường độ sáng của các điểm ảnh trên bề mặt đối tượng khó mà được ổn định. Do vậy vận tốc tính được của các điểm ảnh mà có cường độ sáng thay đổi nhiều là không đúng và sẽ làm cho kết quả theo dõi không được chính xác. Để khắc phục điều này, tôi đề xuất một thuật toán cải tiến kỹ thuật luồng quang học.

Ý tưởng chính của thuật toán này là sử dụng luồng quang học để tính vận tốc của các điểm ảnh trên bề mặt đối tượng. Sau đó tính lỗi và chỉ lọc ra và giữ lại dưới 50% số điểm mà đáng tin cậy nhất bằng cách lấy trung vị lỗi. [5]

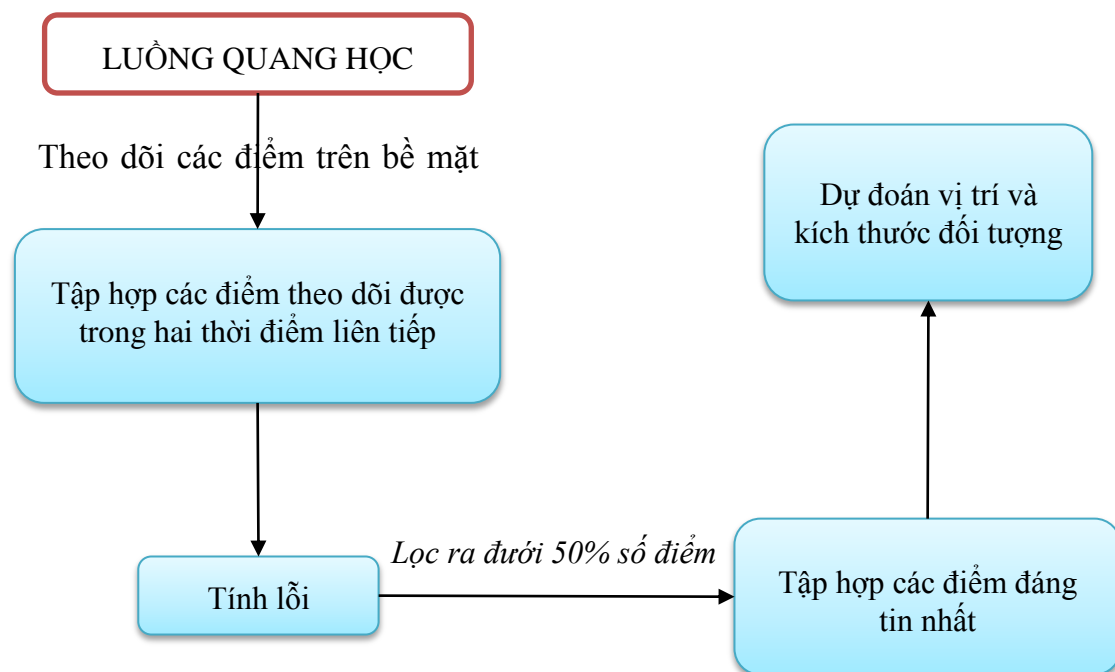


Figure 37: Trình tự thực hiện

### 3.1.2. Tính lỗi và lọc

Quá trình tính lỗi và lọc được mô tả bởi lược đồ sau:

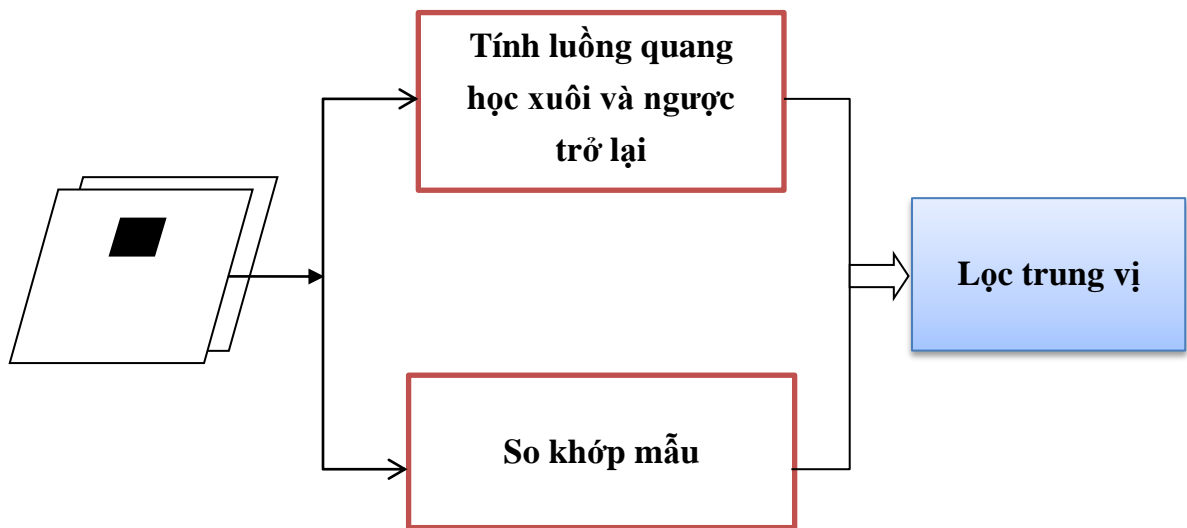


Figure 38: Các bước lọc và tính lỗi

### 3.1.3. Tính luồng quang học tiến và tính ngược trở lại

Là quá trình tìm vị trí của tương ứng của điểm ảnh  $p_1$  trong thời điểm thứ  $i$  ta được điểm ảnh  $p_2$  trong thời điểm thứ  $i + 1$ . Sau đó thực hiện ngược lại tìm vị trí của điểm ảnh  $p_3$  trong thời điểm thứ  $i + 1$  tương ứng với điểm ảnh  $p_2$  trong thời điểm thứ  $i$ . Sau đó tính khoảng cách Euclide giữa hai điểm  $p_1$  và  $p_3$ . Khoảng cách này được dùng để đo lỗi của điểm ảnh  $p_1$ .

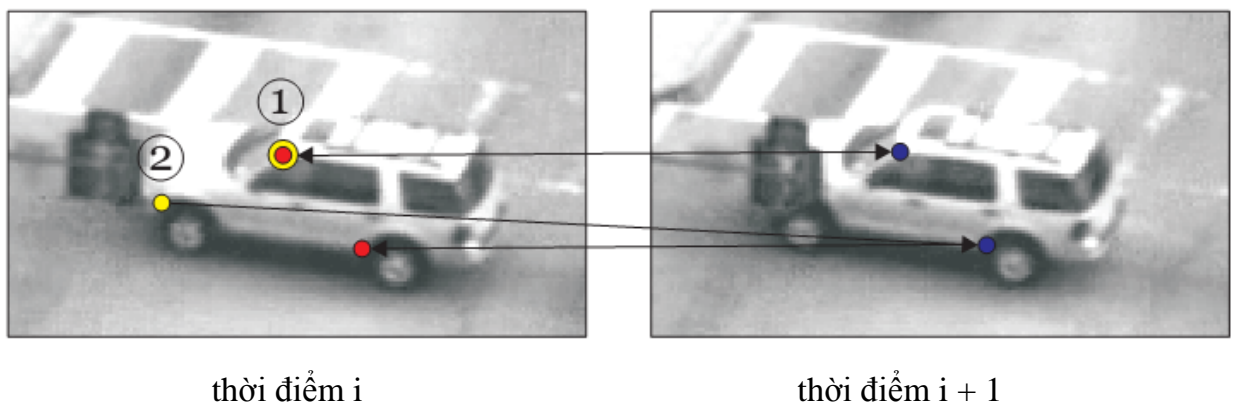


Figure 39: Ví dụ về tính luồng quang học xuôi - ngược

Ở hình trên ta thấy điểm số 1 khi tính điểm tương ứng ở thời điểm thứ  $i+1$  sau đó tính ngược lại ta lại thu được chính nó nên  $\text{error}[1]=0$ . Đối với điểm thứ hai thì 2 điểm  $p_1, p_3$  là khác nhau nên  $\text{error}[2] = \text{Euclide}(p_1, p_3)$ .



Sau khi tính được các giá trị của mảng giá trị lỗi tương ứng với từng điểm, ta có mảng error[ ]. Mảng lưu giá trị lỗi được sử dụng ở bước lọc.

### 3.1.4. So khớp mẫu

Để đảm bảo rằng điểm theo dõi được từ frame thứ i sang frame thứ i+1 được chính xác hơn thì ta dùng thêm một cơ chế nữa đó là so khớp mẫu.

Có thể hiểu so khớp mẫu là quá trình so sánh 2 vùng ảnh tương ứng của 2 điểm. Để so sánh được ta có thể dùng các công thức sau:

+ **Tương quan (CCORR\_NORMED)**

$$R = \frac{\sum_{x',y'} T(x', y') * I'(x', y')}{\sqrt{\sum_{x',y'} T(x', y')^2 * \sum_{x',y'} I'(x', y')^2}}$$

+ **Bình phương sai biệt (SQDIFF\_NORMED)**

$$R = \frac{\sum_{x',y'} (T(x', y') - I'(x', y'))^2}{\sqrt{\sum_{x',y'} T(x', y')^2 * \sum_{x',y'} I'(x', y')^2}}$$

+ **Chuẩn hóa tương quan (CCOEFF\_NORMED)**

$$R = \frac{\sum_{x',y'} T'(x', y') * I'(x', y')}{\sqrt{\sum_{x',y'} T'(x', y')^2 * \sum_{x',y'} I'(x', y')^2}}$$

Trong đó:

$$T'(x', y') = T(x', y') - \frac{1}{w * h} \sum_{x,y} T(x, y)$$

$$I'(x', y') = I(x', y') - \frac{1}{w * h} \sum_{x,y} I(x, y)$$

I và T là hai ảnh cần so khớp.

$x' = 0..width-1$ ,  $y = 0..height-1$ .

R: kết quả.

Áp dụng các công thức trên ta sẽ thu được kết quả là R. R sẽ tiến dần đến 1 nếu hai ảnh càng giống nhau.

### 3.1.5. Lọc trung vị

Là quá trình loại bỏ các điểm xấu (các điểm mà làm cho theo dõi sai) chỉ giữ lại dưới 50% số điểm. Việc lọc được thực hiện bằng cách tính ra trung vị của lỗi và trung vị so khớp. Sau đó loại bỏ tất cả các điểm có lỗi lớn hơn trung vị lỗi và các điểm có giá trị so khớp nhỏ hơn trung vị so khớp. Như vậy ta sẽ thu được một tập các điểm tối ưu nhất để từ các điểm này ta có thể tính được vị trí và kích thước của đối tượng theo dõi.

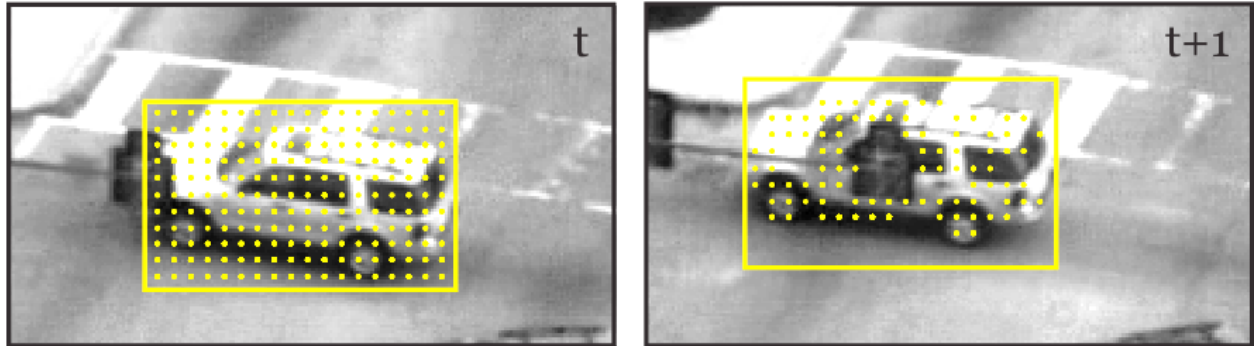


Figure 40: Lọc trung vị

Hình bên phải là các điểm thu được sau khi lọc trung vị.

### 3.1.6. Tính vị trí và kích thước thay đổi của đối tượng

#### + Tìm vị trí mới của đối tượng

Giả sử mỗi điểm  $p_i$  trong thời điểm  $t$  sẽ chuyển động với vận tốc là  $v_i$  (độ dịch chuyển). Như vậy thì vị trí của điểm  $p_i$  trong thời điểm  $t+1$  sẽ là  $p_i + v_i$ . Vì các điểm trên bề mặt đối tượng có thể chuyển động không cùng vận tốc nên để tính được vị trí của đối tượng ta ở thời điểm thứ  $t+1$  một cách chính xác ta có thì ta lấy vị trí của thời điểm thứ  $t$  cộng với trung vị của độ dịch chuyển.

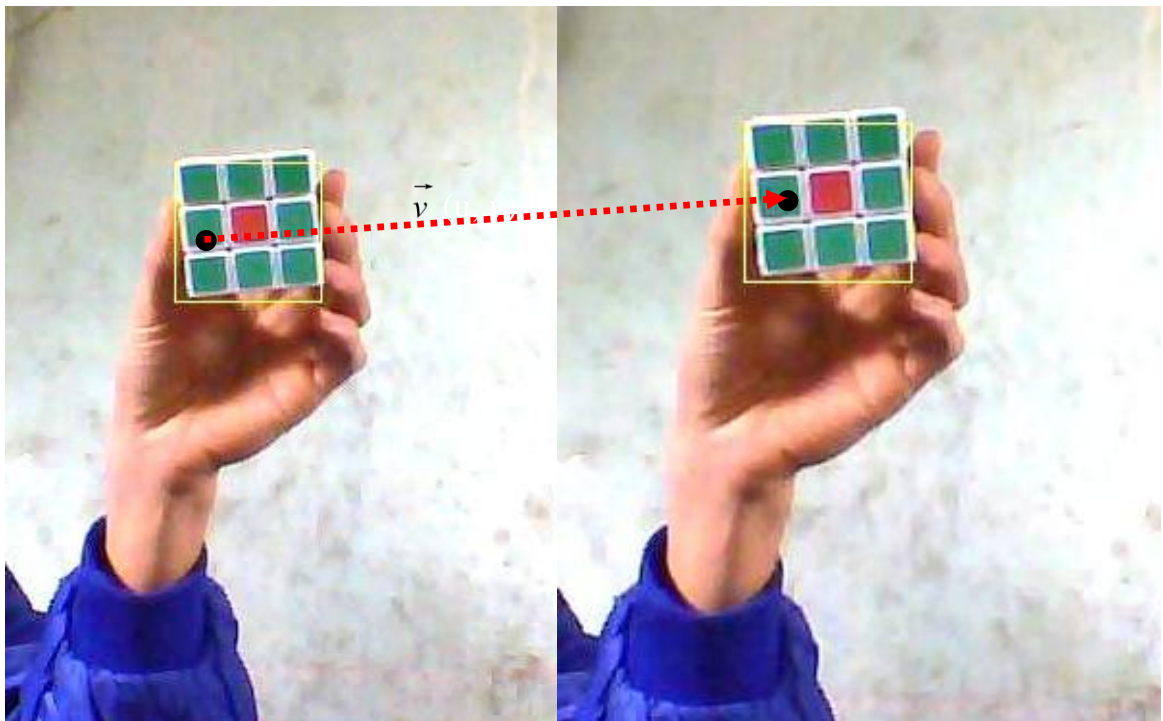
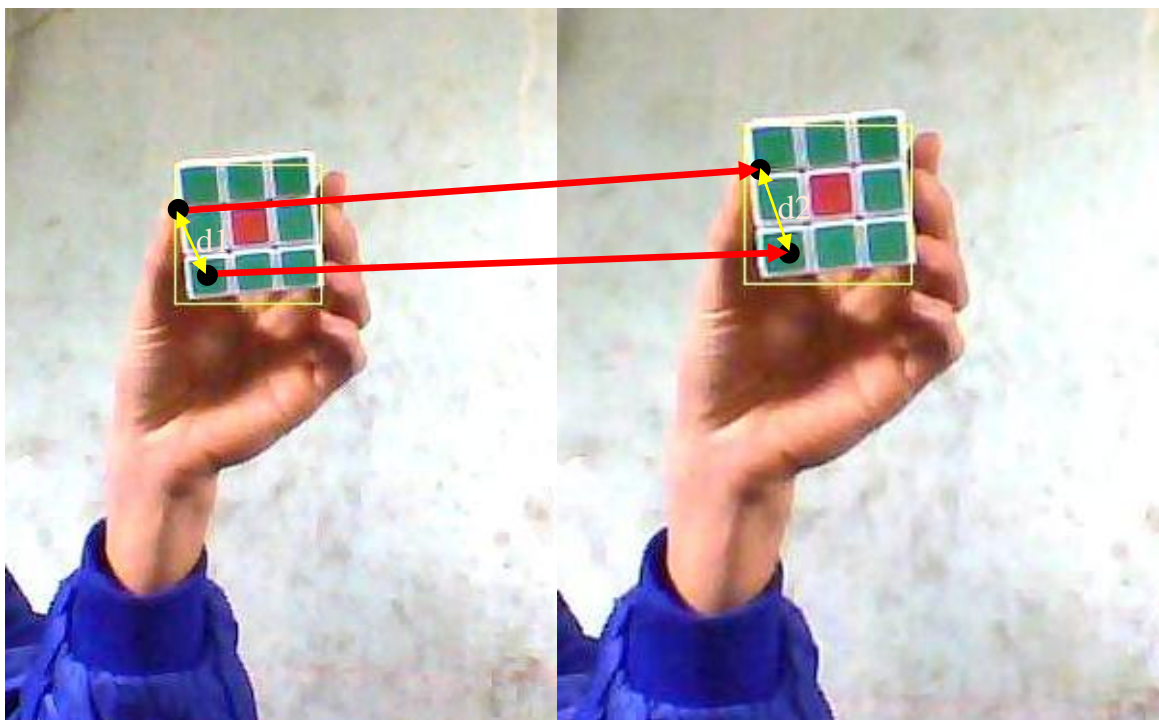


Figure 41: Ví dụ áp dụng luồng quang học theo dõi chuyển động của rubik

#### + Tính kích thước mới của đối tượng

Như ở phần trước chuyển động chúng ta được biết các điểm ảnh sẽ di chuyển hội tụ lại khi vật di chuyển ra xa camera hay nói cách khác ảnh của vật sẽ có kích thước nhỏ hơn khi đi ra xa camera.



thời điểm  $i$

thời điểm  $i+1$

Figure 42: Ví dụ tính kích thước mới của đối tượng dùng cải tiến của luồng quang học

Giả sử  $p_1, p_2$  là 2 điểm nằm trên bề mặt của đối tượng như hình trên và  $p_{1_i}, p_{1_{i+1}}$  tương ứng với các ký hiệu  $p_1$  ở trên frame  $i$  và  $p_1$  ở trên frame  $i+1$ . Như vậy độ co giãn của đối tượng có thể tính được như sau:

$$s = \frac{d_2}{d_1} = \frac{d(p_{1_{i+1}}, p_{2_{i+1}})}{d(p_{1_i}, p_{2_i})}$$

Với  $d(p_1, p_2)$  là khoảng cách euclide giữa 2 điểm  $p_1$  và  $p_2$ .

Chú ý ở trên bề mặt của đối tượng có rất nhiều điểm ảnh nên để đảm bảo xấp xỉ được tốt nhất ta phải tính tất cả các khoảng cách sau đó lấy trung vị của độ co giãn.

Cuối cùng kích thước của đối tượng tại thời điểm thứ  $t + 1$  ta có thể tính được như sau:

$$W_{i+1} = W_i * s$$

$$H_{i+1} = H_i * s$$

### 3.2. Đề xuất thứ hai - Mô hình chung của bài toán thực hiện

Dựa trên những nội dung đã nghiên cứu trong chương 2, tôi nhận thấy rằng nhánh nghiên cứu Phân đoạn chuyển động dựa trên phân tách, phát hiện và bám sát đối tượng chuyển động là một nhánh nghiên cứu có một số ứng dụng thực tế rất hiệu quả. Vì vậy, tôi quyết định lựa chọn nội dung nghiên cứu này.

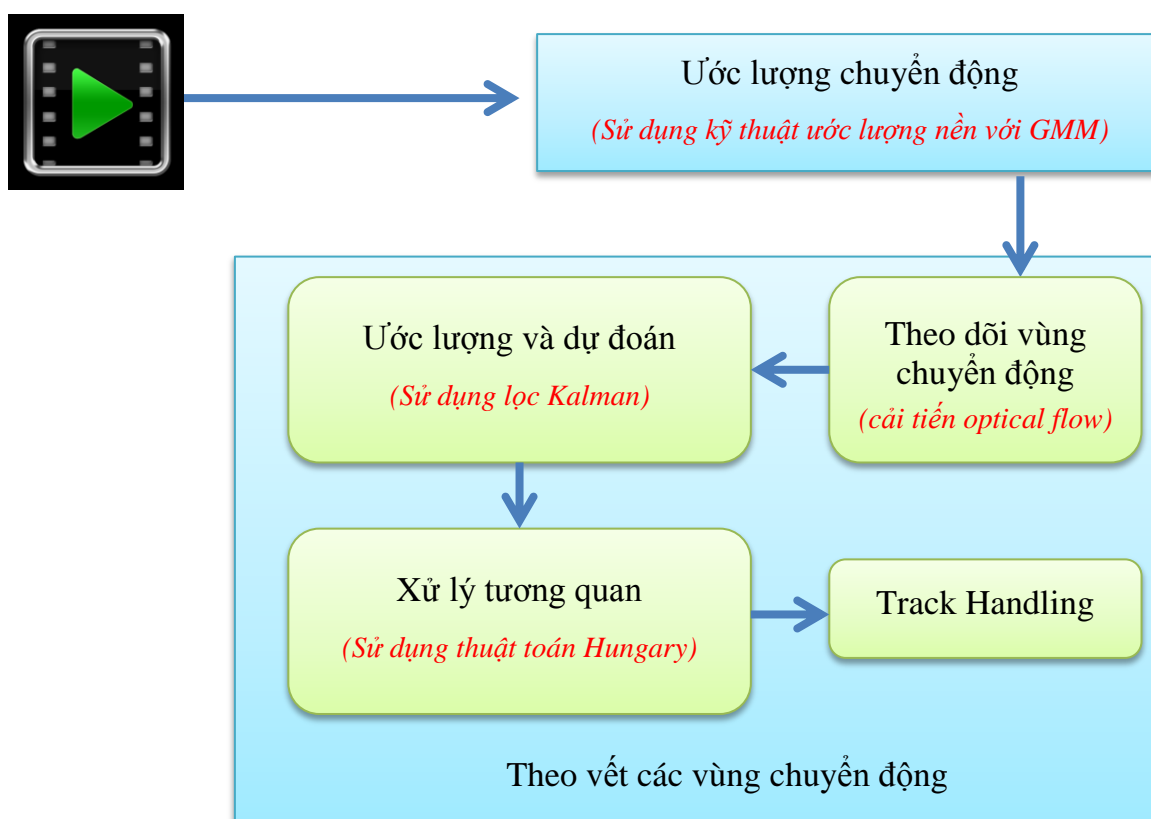
Hơn nữa, với nền tảng về các bộ lọc được tìm hiểu xuyên suốt thời gian thực tập, rất phù hợp với việc áp dụng cho bài toán bám sát vùng đối tượng chuyển động phục vụ cho quá trình phân đoạn các đối tượng.

Vì vậy, trong phần này, tôi đề xuất mô hình bài toán phân đoạn chuyển động của tôi với các nội dung sau:

- Kỹ thuật ước lượng chuyển động
- Kỹ thuật theo dõi đối tượng sử dụng cải tiến của kỹ thuật luồng quang học
- Tương quan dữ liệu sử dụng thuật toán Hungary

#### 3.2.1. Phát hiện đối tượng chuyển động

Thay vì cách tiếp cận dựa trên phân đoạn theo vùng, tôi lựa chọn cách tiếp cận theo đối tượng chuyển động trong video.



Hình 1: Mô hình bài toán đề xuất

### 3.2.2. Sử dụng hỗn hợp Gaussian để ước lượng nền cho bài toán ước lượng chuyển động

Tôi đề xuất sử dụng  $K = 10$ , nghĩa là sử dụng chuỗi 10 hình ảnh liên tiếp trước đó. Việc lựa chọn tham số  $K$  có ảnh hưởng trực tiếp đến hiệu quả thuật toán. Dựa trên kinh nghiệm cá nhân khi trải qua một số thực nghiệm kỹ thuật trừ ảnh áp dụng trong theo dõi bàn tay chuyển động, theo dõi phương tiện giao thông, ... tôi thấy  $K = 10$  là lựa chọn tốt để cân bằng cả yếu tố độ chính xác và tốc độ thực thi.

Hình dưới đây mô tả các bước làm việc của thuật toán trừ ảnh đề xuất.

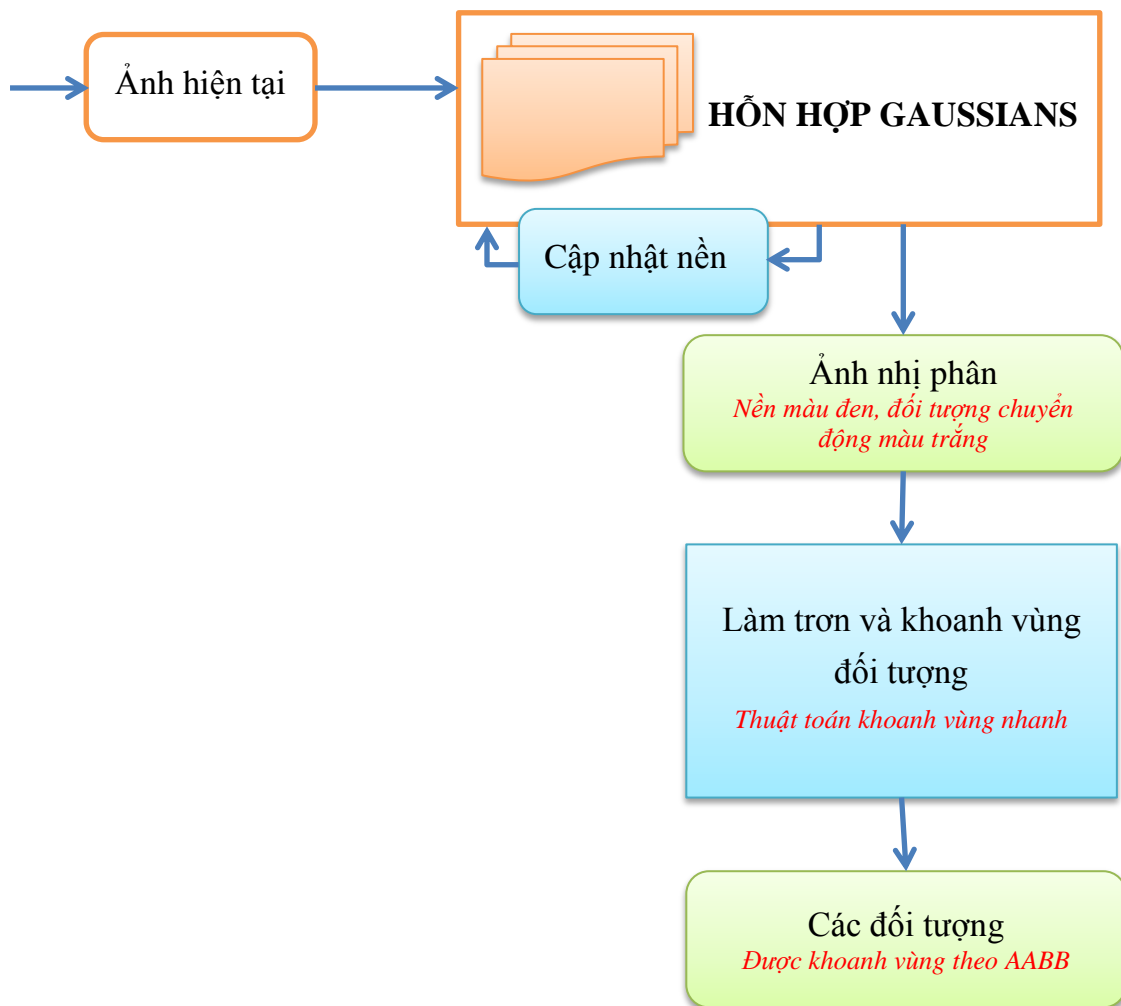


Figure 43: Mô hình đề xuất của kỹ thuật ước lượng chuyển động sử dụng GMM

### 3.2.3. Áp dụng bộ lọc kết hợp cùng thuật toán cải tiến luồng quang học cho bài toán theo vết các vùng chuyển động

#### 3.1.3.1. Mô hình hệ thống

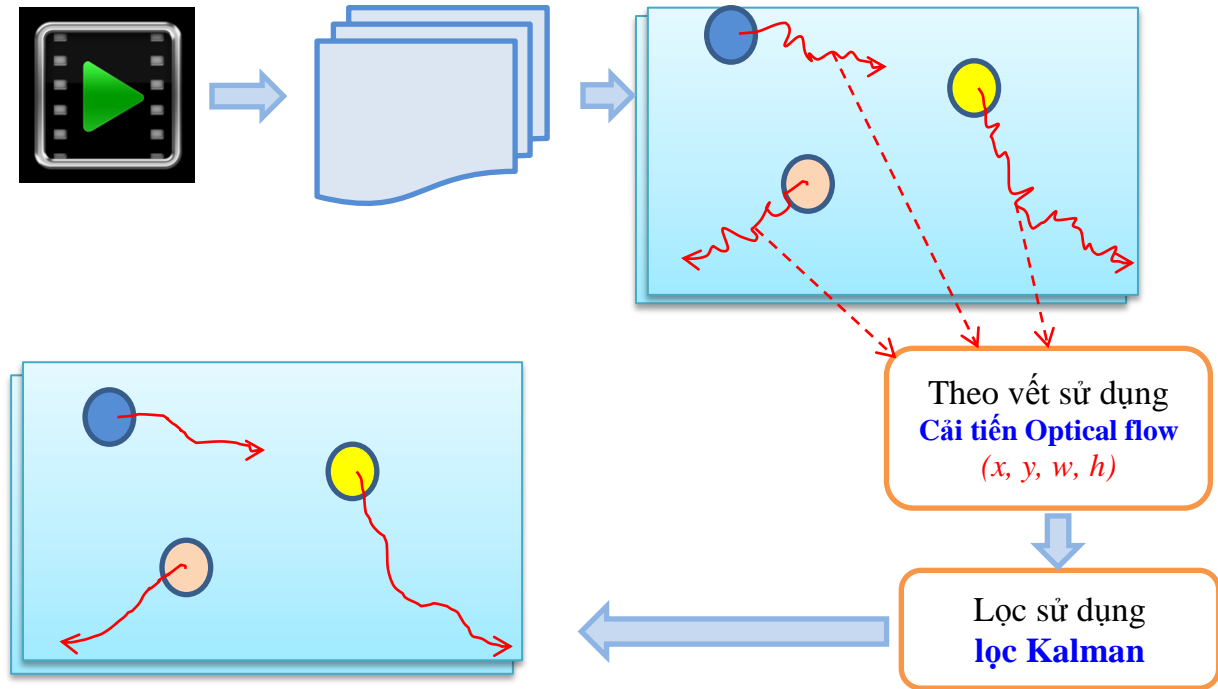


Figure 44: Mô hình đề xuất sử dụng cải tiến optical flow và lọc Kalman cho bài toán phân đoạn chuyển động

#### 3.1.3.2. Mô hình của bộ lọc Kalman

- Với lọc Kalman, tôi sử dụng mô hình hệ thống sau:

$$x_k = A_{k-1}x_{k-1} + w_{k-1}$$

$$z_k = H_k x_k + v_k$$

Trong đó:

- $x_k \in R^n$  là trạng thái của hệ thống tại thời điểm k

$$x_k = \begin{bmatrix} x & 0 & T & 0 \\ 0 & y & 0 & T \\ 0 & 0 & w & 0 \\ 0 & 0 & 0 & h \end{bmatrix}$$

- $z_k \in R^m$  là kết quả đo đạt được của hệ thống tại thời điểm k

$$z_k = \begin{bmatrix} X \\ Y \\ X - X_{prev} \\ Y - Y_{prev} \end{bmatrix}$$

- $w_{k-1} \sim N(0, Q_{k-1})$  là nhiễu hệ thống tại thời điểm  $k - 1$

$$Q = \begin{bmatrix} 10^{-4} & 10^{-4} & 10^{-4} & 10^{-4} \\ 10^{-4} & 10^{-4} & 10^{-4} & 10^{-4} \\ 10^{-4} & 10^{-4} & 10^{-4} & 10^{-4} \\ 10^{-4} & 10^{-4} & 10^{-4} & 10^{-4} \end{bmatrix}$$

- $v_k \sim N(0, R_k)$  là nhiễu đo đạc tại thời điểm  $k$

$$R = \begin{bmatrix} 10^{-6} & 10^{-6} & 10^{-6} & 10^{-6} \\ 10^{-6} & 10^{-6} & 10^{-6} & 10^{-6} \\ 10^{-6} & 10^{-6} & 10^{-6} & 10^{-6} \\ 10^{-6} & 10^{-6} & 10^{-6} & 10^{-6} \end{bmatrix}$$

- $A_{k-1}$  là ma trận chuyển trạng thái của hệ thống

$$A = \begin{bmatrix} 1 & 0 & 0.1 & 0 \\ 0 & 1 & 0 & 0.1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- $H_k$  là ma trận của mô hình đo đạc

$$H = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$



### 3.2.4. Áp dụng thuật toán Hungary giải quyết bài toán tương quan dữ liệu

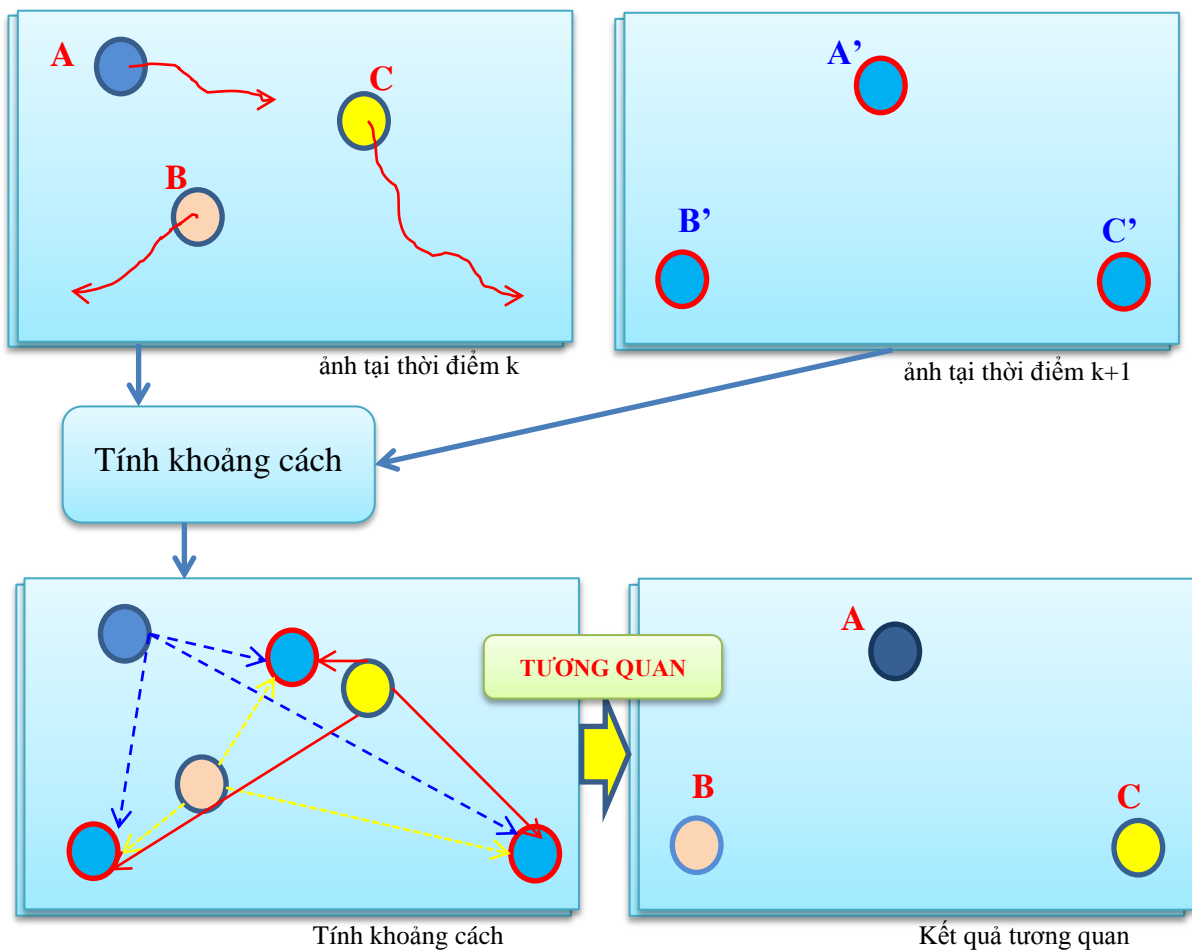


Figure 45: Mô hình đề xuất sử dụng thuật toán Hungary cho bài toán tương quan

Trình tự thực hiện tương quan dữ liệu như sau:

- Tính khoảng cách tương ứng giữa các đối tượng giữa 2 frame.
- Áp dụng kỹ thuật GNN (thuật toán Hungary) để chọn ra cách ghép cặp các đối tượng sao cho tổng khoảng cách thu được là nhỏ nhất

## **Chương 4: Thực nghiệm và kết luận**

### **4.1. Điều kiện thực nghiệm**

#### **4.1.1. Môi trường**

- Các môi trường thông thường như: *sân trường, đường xá,...* với các yếu tố sau:
  - o Màu sắc nền và đối tượng có độ tương phản nhất định
  - o Các đối tượng di chuyển với mật độ và khoảng cách tương ứng ở mức trung bình
- Yếu tố màu sắc: Các vùng màu sắc có độ tương phản nhất định. Trong trường hợp màu sắc của nền và đối tượng đồng nhất, thì chương trình hoạt động rất kém hiệu quả.

#### **4.1.2. Thiết bị**

- Máy tính
- Thiết bị ghi hình hoặc file Video:
  - o File video: Kích thước lớn hơn 320x240 pixel

#### **4.1.3. Cấu hình máy**

- CPU tối thiểu: 1.0 GHz
- RAM tối thiểu: 512 MB

*(Điều kiện để cài đặt .NET Framework 3.5)*

#### **4.1.4. Khoảng cách thực nghiệm**

- Camera đặt trước đối tượng với khoảng cách từ 2m-25m

#### **4.1.5. Môi trường triển khai**

- Ngôn ngữ lập trình: Visual C#, trên nền .NET framework 4.0
- Thư viện mã nguồn mở EmguCV (<http://emgu.com>)

### **4.2. Thực nghiệm kỹ thuật phân đoạn video**

#### **4.2.1. Phương pháp thực nghiệm**

Để thực nghiệm và đánh giá cách tiếp cận dựa trên bộ lọc, ứng dụng cho kỹ thuật phân đoạn chuyển động video. tôi sử dụng 08 video với kích thước và thời gian nhất định. Tỷ lệ chính xác được đánh giá bằng tỷ lệ đối tượng thu được/tổng số đối tượng di chuyển trong video đó.

Các video xuất phát từ việc tự quay và tổng hợp tại các địa điểm:

- Cầu vượt gần số 252 Hoàng Quốc Việt (gần trường Cao đẳng Du Lịch Hà Nội)

- Cầu vượt giao cắt giữa đường Xuân Thủy và đường Phạm Văn Đồng (gần trường ĐH Quốc gia Hà Nội)
- Sân Tennis – khu A trường ĐH Công nghiệp Hà Nội (quay từ tầng 4 – nhà A7)

Các video thu được có chiều dài và kích thước như sau:

STT	TÊN VIDEO	THỜI GIAN	ĐỊA ĐIỂM
1.	video1.avi	3s	252 Hoàng Quốc Việt
2.	video2.avi	8s	252 Hoàng Quốc Việt
3.	video3.avi	10s	252 Hoàng Quốc Việt
4.	video4.avi	7s	252 Hoàng Quốc Việt
5.	video5.avi	14s	Cầu vượt ĐH Quốc gia HN
6.	video6.avi	20s	Cầu vượt ĐH Quốc gia HN
7.	video7.avi	28s	252 Hoàng Quốc Việt
8.	video8.avi	27s	252 Hoàng Quốc Việt
9.	Video9.avi	23s	Sân trường khu A - ĐHCNHN

#### 4.2.2. Kết quả thực nghiệm

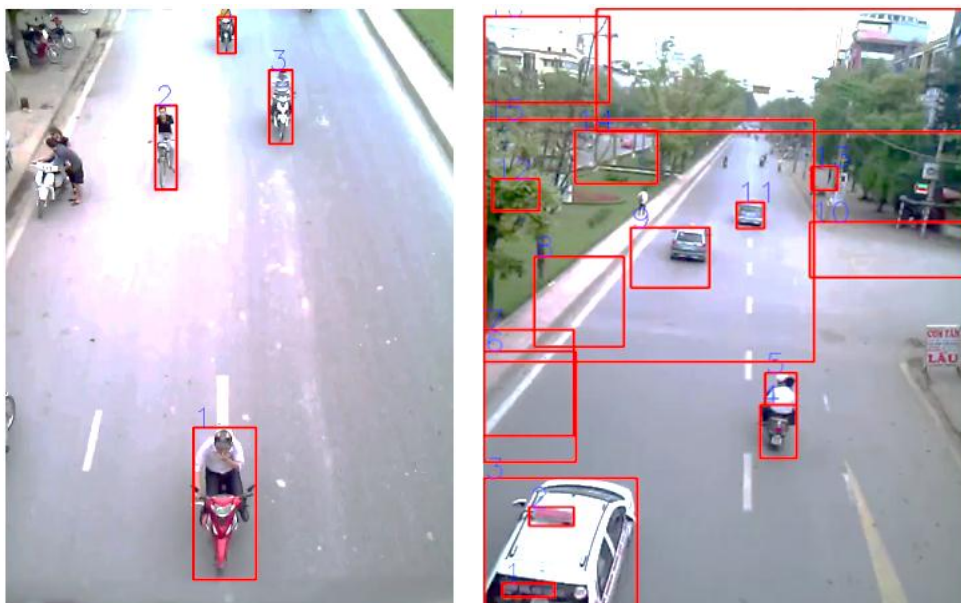


Figure 46: Thuật toán đề xuất hoạt động kém hiệu quả khi camera bị dịch chuyển

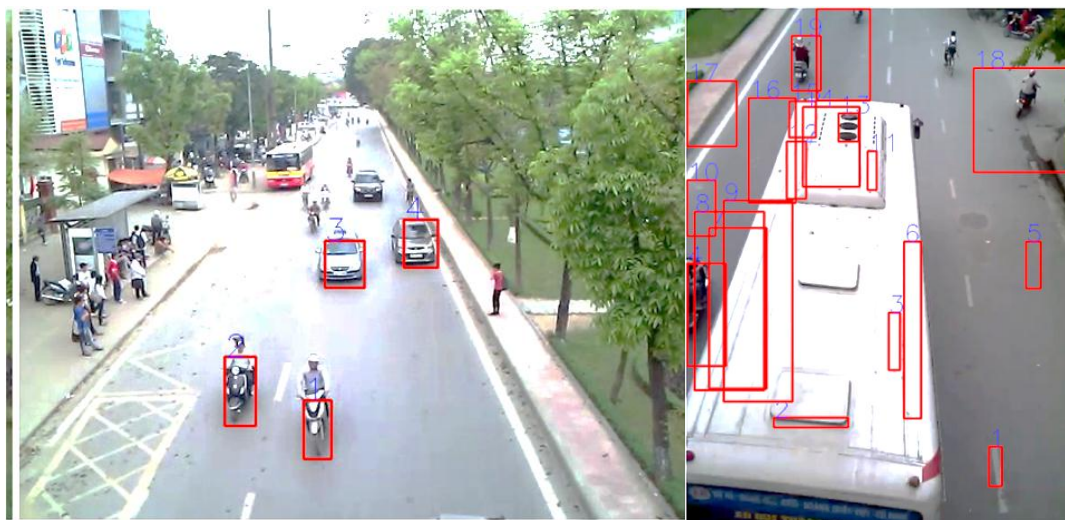


Figure 47: Ước lượng nền sử dụng GMM hoạt động kém hiệu quả khi camera bị dịch chuyển

Dưới đây là bảng kết quả thực nghiệm

STT	TÊN VIDEO	THỜI GIAN	PHÁT HIỆN SAI	PHÁT HIỆN THIẾU	PHÁT HIỆN ĐÚNG	TỶ LỆ	FRAME RATE	LÝ DO
10.	video1.avi	3s	4	0	8	66.7%	11,5 fps	
11.	video2.avi	8s	2	0	6	75%	9,7 fps	
12.	video3.avi	10s	18	0	9	33.3%	9,5 fps	Camera bị di chuyển
13.	video4.avi	7s	26	0	7	21.2%	5,3fps	Camera bị di chuyển
14.	video5.avi	14s	43	0	25	36.8%	5,3fps	MoG kém hiệu quả
15.	video6.avi	20s	18	1	30	61.2%	9,2fps	Camera bị di chuyển
16.	video7.avi	28s	82	3	77	47.5%	5,5fps	Camera bị di chuyển
17.	video8.avi	27s	45	2	28	37.3%	4,5fps	Camera bị di chuyển
18.	video9.avi	23s	2	1	23	88.5%	2,3fps	

### 4.3. Kết luận

Dựa vào kết quả thực nghiệm, tôi rút ra những kết luận như sau:

- Trong trường hợp camera dịch chuyển trong một khoảng thời gian, rồi trở lại vị trí ban đầu: Thuật toán trừ ảnh trước, trừ nền trung vị và trung bình bậc lợ nhiều nhược điểm. Kỹ thuật ước lượng nền sử dụng mô hình hỗn hợp Gaussians khắc phục được nhược điểm trên.
- Các kỹ thuật trừ ảnh, ước lượng nền kết hợp với kỹ thuật tương quan có một số ưu điểm: *Dễ triển khai, thực thi nhanh, hoạt động tốt trong môi trường nền và đối tượng có màu sắc khá tương phản và khoảng cách tương đối giữa các đối tượng khá thưa thớt.*
- Việc áp dụng bộ lọc, yếu tố tiên quyết là thuật toán phát hiện đối tượng chuyển động phải chính xác.
- Áp dụng cải tiến của kỹ thuật luồng quang học, kết hợp cùng lọc Kalman đã từng bước cải tiến nhược điểm của các kỹ thuật trên khi hoạt động trong môi trường kém lý tưởng cho giả thuyết để luồng quang học hoạt động.
- Việc áp dụng lọc trung vị lỗi cho kỹ thuật luồng quang học với bài toán đa đối tượng làm tăng độ phức tạp thuật toán, từ đó hạn chế phạm vi ứng dụng của thuật toán trong các hệ thống thực tế.

### 4.4. Hướng nghiên cứu

- Áp dụng lọc Kalman đa mô hình: Lọc Kalman tương tác đa mô hình
  - Mô hình dựa trên trừ nền (tuyến tính)
  - Mô hình dựa trên kỹ thuật cải tiến luồng quang học
- Nghiên cứu mô hình thống kê dựa trên lọc hạt
- Nghiên cứu kỹ thuật theo dõi đa giả thuyết (Multiple Hypothesis Tracking) tối ưu bài toán tương quan dữ liệu
- Nghiên cứu phương pháp bao bọc đối tượng OORB, KDrop

# TÀI LIỆU THAM KHẢO

- [1]. <http://www.labbookpages.co.uk/software/imgProc/blobDetection.html> last access March 20<sup>th</sup>, 2012.
- [2]. <http://cntt.tv/nodes/show/485> last access April 20<sup>th</sup>, 2012
- [3]. <http://mpeg.chiariglione.org/standards/mpeg-7/mpeg-7.htm> last access March 10<sup>th</sup>, 2012.
- [4]. Jouni Hartikainen, Arno Solin, Simo Sarkka, “*Optimal filtering with Kalman filter and smoothers*”, Aalto University, 8-2011.
- [5]. Zdenek Kalal, “*Predator, a smart camera that learns from its errors*”, UK ICT Pioneers 2011 Competition, 4-2011
- [6]. R. Stolkin, A. Greig, M. Hodgetts, and J. Gilby, “*An em/e-mrf algorithm for adaptive model based tracking in extremely poor visibility.*” Image and Vision Computing, vol. 26, no. 4, pp. 480–495, 2008.
- [7]. Ronghua Guo, Zheng Qin, Xiangnan Li, and Junliang Che, “*Interacting Multiple Model Particle-type Filtering Approaches to Ground Target Tracking*”, Tsinghua University, Beijing, China, 2008.
- [8]. Rishabh Malhotra, “*Detection and segmentation of moving objects in video using optical vector flow estimation*”, PhD Thesis, University of Saskatchewan, 2008
- [9]. M. Pawan Kumar, P.H.S. Torr, A.Zisserman, “*Learning Layered Motion Segmentations of Video*”, Oxford Brookes University, 2008.
- [10]. M. P. Kumar, P. H. Torr, and A. Zisserman, “*Learning layered motion segmentations of video,*”International Journal of Computer Vision, vol. 76, no. 3, pp. 301–319, 2008.
- [11]. Mercè Mora Romera, “*Motion Segmentation in Monocular Image Sequences*”, Master of thesis, Karlsruhe University, 2008
- [12]. A. Colombari, A. Fusiello, V. Murino, “*Segmentation and Tracking of Multiple Video Object*”, Dipartimento di Informatica, University of Verona, 2008.
- [13]. Yogesh Rathi, Namrata Vaswani, Allen Tannenbaum, Anthony Yezzi, “*Tracking Deforming Objects using Particle Filtering for Geometric Active Contours*”, IEEE transactions on pattern analysis and machine intelligence, 2008.
- [14]. H. Shen, L. Zhang, B. Huang, and P. Li, “*A map approach for joint motion estimation, segmentation, and super resolution,*”IEEE Transactions on Image Processing, vol. 16, no. 2, pp. 479–490, 2007.

- [15]. R. Li, S. Yu, and X. Yang, “*Efficient spatio-temporal segmentation for extracting moving objects in video sequences*,” IEEE Transactions on Consumer Electronics, vol. 53, no. 3, pp. 1161–1167, Aug. 2007.
- [16]. J. Zhang, F. Shi, J. Wang, and Y. Liu, “*3d motion segmentation from straight-line optical flow*”, in Multimedia Content Analysis and Mining, 2007, pp. 85–94.
- [17]. Bing Han, William Roberts, Dapeng Wu, Jian Li, “*Motion-Segmentation-Based Change Detection*”, University of Florida Gainesville, 2007.
- [18]. C. Julià, A. Sappa, F. Lumberras, J. Serrat, and A. Lopez, “*Motion segmentation from feature trajectories with missing data*”, in Iberian Conference on Pattern Recognition and Image Analysis, 2007, pp. I: 483–490.
- [19]. A. Colombari, A. Fusiello, and V. Murino, “*Segmentation and tracking of multiple video objects*,” Pattern Recognition, vol. 40, no. 4, pp. 1307–1317, 2007.
- [20]. A. Goh and R. Vidal, “*Segmenting motions of different types by unsupervised manifold clustering*”, IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–6, 17-22 June 2007
- [21]. N. Vaswani, A. Tannenbaum, and A. Yezzi, “*Tracking deforming objects using particle filtering for geometric active contours*,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 8, pp. 1470–1475, 2007.
- [22]. J. Yan and M. Pollefeys, “*A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate*”, in European Conference on Computer Vision, 2006, pp. IV: 94–106.
- [23]. X. Llado, A. D. Bue, and L. Agapito, “*Euclidean reconstruction of deformable structure using a perspective camera with varying intrinsic parameters*”, 18th International Conference on Pattern Recognition, vol. 1, pp. 139–142, 2006.
- [24]. A. Gruber and Y. Weiss, “*Incorporating non-motion cues into 3d motion segmentation*”, in European Conference on Computer Vision, 2006, pp. 84–97.
- [25]. A. G. Amitha Perera, Glen Brooks, Anthony Hoogs, Gianfranco Doretto, “*Moving Object Segmentation using Scene Understanding*”, One Research Circle, Niskayuna, New York 12309, 2006.
- [26]. R. Danch ick and G.E. Newnam, “*Reformulating Reid’s MHT method with generalised Murty K-best ranked linear assignment algorithm*”, IEE P roc.-Radar Sonar Navig, 2006.
- [27]. F.-H. Cheng and Y.-L. Chen, “*Real time multiple objects tracking and identification based on discrete wavelet transform*,” Pattern Recognition, vol. 39, no. 6, pp. 1126–1139, 2006.

- [28]. E. Trucco and K. Plakas, “*Video tracking: A concise survey*,” IEEE Journal of Oceanic Engineering, vol. 31, no. 2, pp. 520–529, 2006.
- [29]. Ren e Vidal, “*Generalized Principal Component Analysis (GPCA)*”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.27, no.12, page 1925-1959,2005.
- [30]. Conf. on NEURAL NETWORKS, Lisbon, Portugal, June 16-18, 2005 (pp20-25)
- [31]. Vladimir Cherkassky & Yunqian Ma, “*Multiple Model Estimation: A New Formulation for Predictive Learning*”, University of Minnesota, 2005.
- [32]. A.S. Ogale, C. Fermuller and Y. Aloimonos, “*Motion Segmentation Using Occlusions*”, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.27, No.6 , 2005, pp.988-992.
- [33]. D. Cremers and S. Soatto, “*Motion competition: A variational approach to piecewise parametric motion segmentation*,”International Journal of Computer Vision, vol. 62, no. 3, pp. 249–265, May 2005.
- [34]. A. Cavallaro, O. Steiger, and T. Ebrahimi, “*Tracking Video Objects in Cluttered Background*,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 15, no. 4, pp. 575–584, 2005.
- [35]. Y. Sugaya and K. Kanatani, “*Geometric structure of degeneracy for multi-body motion segmentation*” , inStatistical Methods in Video Processing, 2004, pp. 13–25.
- [36]. P. Smith, T. Drummond and R. Cipolla, “*Layered Motion Segmentation and Depth Ordering by Tracking Edges*”, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 26, No.4, 2004, pp.479-494.
- [37]. A. Gruber and Y. Weiss, “*Multibody factorization with uncertainty and missing data using the t i algo-rithm*” , Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. I–707–I–714 Vol.1, 27 June-2 July 2004.
- [38]. Samuel S. Blackman, “*Multiple Hypothesis Tracking For Multiple Target Tracking*”, IEEE A&E Systems Magazine, 2004.
- [39]. R. Vidal and R. Hartley, “*Motion segmentation with missing data using powerfactorization and gpca*”, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. II–310–II–316 Vol.2, 27 June-2 July 2004.
- [40]. L. Zelnik-Manor and M. Irani, “*Temporal factorization vs. spatial factorization*”, inEuropean Conference on Computer Vision, vol. 2. Springer Berlin / Heidelberg, 2004, pp. 434–445.



- [41]. Marco Benvenuti, Sara Colantonio, Maria Grazia Di Bono, Gabriele Pieri, “*Tracking of Moving Targets in Video Sequences*”, Proceedings of the 6th WSEAS Int.
- [42]. S. Borman, “*The expectation maximization algorithm – a short tutorial*”, Jul. 2004.
- [43]. Julier, S.J.; Uhlmann, J.K, “*Unscented filtering and nonlinear estimation*”. Proceedings of the IEEE: 401–422, 2004.
- [44]. I. Rekleitis, “*Cooperative localization and multi-robot exploration*”, PhD in Computer Science, School of Computer Science, McGill University, Montreal, Quebec, Canada, 2003.
- [45]. L.Zelnik-Manor and M. Irani, “*Degeneracies, dependencies and their implications in multi-body and multi-sequence factorizations*”, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. II–287–93 vol.2, 18-20 June 2003.
- [46]. H. Zhou and T. S. Huang, “*Recovering articulated motion with a hierarchical factorization method*”, in Gesture Workshop, 2003, pp. 140–151.
- [47]. Henk A.P. Blom and Edwin A. Bloem, “*Interacting Multiple Model Joint Probabilistic Data Association avoiding track coalescence*”, National Aerospace Laboratory NLR, Amsterdam, The Netherlands, 2002.
- [48]. M. Brand, “*Incremental singular value decomposition of uncertain data with missing values*”, inEuro-pean Conference on Computer Vision, 2002, pp. 707–720.
- [49]. K. Kanatani and C. Matsunaga, “*Estimating the number of independent motions for multibody motion segmentation*”, in Proceedings of the Fifth Asian Conference on Computer Vision, vol. 1, Jan 2002, pp. 7–12.
- [50]. Yaakov Bar-Shalom, X.-Rong Li, Thiagalingam Kiruba, “*Estimation with Applications to Tracking & Navigation*”, A Wiley-Interscience Publication, Canada, 2001.
- [51]. David STweed, “*Motion Segmentation Across Image Sequences*”, University of Bristol, 2001.
- [52]. C. Rasmussen and G. D. Hager, “*Probabilistic data association methods for tracking complex visual objects*,”IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 6, pp. 560–576, 2001.
- [53]. D.J. Fleet, M.J. Black, Y. Yacoob and A.D. Jepson, “*Design and use of linear models for image motion analysis*”, Int. J. Comput. Vis. Vol.36, No.3, 2000, pp.171-193.

- [54]. R. Szeliski, S. Avidan, P. Anandan, "*Layer extraction from multiple images containing reflections and transparency*", IEEE Conf. on Computer Vision and Pattern Recognition, Vol.I, June, 2000, pp.246-253.
- [55]. Thomas Bak, "*Lecture Notes - Estimation and Sensor Information Fusion*", Aalborg University, 2000.
- [56]. M.J. Black and D.J. Fleet, "*Probabilistic detection and tracking of motion boundaries*", Int. J. Comput. Vis., Vol.38, No.3, 2000, pp.231-245.
- [57]. N. Ichimura and F. Tomita, "*Motion segmentation based on feature selection from shape matrix*", Systems and Computers in Japan, vol. 31, no. 4, pp. 32–42, 2000.
- [58]. Y. Boykov, O. Veksler, and R. Zabih, "*Fast approximate energy minimization via graph cuts*", in International Conference on Computer Vision, 1999, pp. 377–384.
- [59]. J. P. Costeira and T. Kanade, "*A multibody factorization method for independently moving objects*," International Journal of Computer Vision, vol. 29, no. 3, pp. 159–179, 1998.
- [60]. Yucel Altunbasak, P. Erhan Eren, A. Murat Tekalp, "*Region-Based Parametric Motion Segmentation Using Color Information*", Graphical Models and Image Processing, vol. 60, No. 1, January, pp. 13–23, 1998.
- [61]. M. Kong, J.-P. Leduc, B. Ghosh, and V. Wickerhauser, "*Spatio-temporal continuous wavelet transforms for motion-based segmentation in real image sequences*," Proceedings of the International Conference on Image Processing, vol. 2, pp. 662–666 vol.2, 4-7 Oct 1998.
- [62]. T. Camus, "*Real-Time Quantized Optical Flow*", Journal of Real-Time Imaging, Volume 3, pages 71-86, 1997.
- [63]. L. Wiskott, "*Segmentation from motion: Combining Gabor- and Mallat-wavelets to overcome aperture and correspondence problem*," in Proceedings of the 7th International Conference on Computer Analysis of Images and Patterns, G. Sommer, K. Daniilidis, and J. Pauli, Eds., vol. 1296. Heidelberg: Springer-Verlag, 1997, pp. 329–336.
- [64]. A. Bobick and J. Davis, "*An appearance-based representation of action*", IEEE International Conference on Pattern Recognition, pp. 307–312, 1996.
- [65]. "*Optical flow and its Discontinuities using non-linear diffusion*", In 3rd European Conference on Computer Vision, ECCV'94, Volume 2, pages 295-304, 1994.
- [66]. Matt M. Miller & Harold S. Stone and Ingemar J. Cox, "*Optimizing Murty's Ranked Assignment Method*", NEC Research Institute, Technical Report 95-08.
- [67]. J.Y.A. Wang, E.H. Adelson, "*Layered representation for motion analysis*", in: IEEE Conf. on Computer Vision and Pattern Recognition, June, 1993, pp.361–366.

- [68]. Til Aach and Andr e Kaup and Rudolf Mester , “*Statistical Model-Based Change Detection in Moving Video*”, Institute of Imaging & Computer Vision, 1993.
- [69]. C. Tomasi and T. Kanade, “*Shape and motion from image streams under orthography: a factorization method*” , International Journal of Computer Vision, vol. 9, no. 2, pp. 137–154, 1992.
- [70]. Alessandro Verri, Sergio Uras & Enrico De Micheli, “*Motion Segmentation from Optical Flow*”, Dipartimento di Fisica dell' Universita di Genova, 1989.
- [71]. B. K. Horn and B. G. Schunck, “*Determining optical flow*” , Cambridge, MA, USA, Tech. Rep., 1980.
- [72]. Donald B. Reid, “*An Algorithm for Tracking Multiple Targets*”, IEE Transaction on automatic control, 1979.
- [73]. Bruce A. McElhoe, “*An Assessment of the Navigation and Course Corrections for a Manned Flyby of Mars or Venus*”. Aerospace and Electronic Systems, IEEE Transactions on: 613–623, 1966.
- [74]. G.L. Smith; S.F. Schmidt and L.A. McGee, “*Application of statistical filter theory to the optimal estimation of position and velocity on board a circumlunar vehicle*”. National Aeronautics and Space Administration, 1962.
- [75]. R.E. Kalman; R.S. Bucy (1961). “*New results in linear filtering and prediction theory*”. Journal of Basic Engineering: 95–108, 1961.
- [76]. R.E. Kalman (1960). “*A New Approach to Linear Filtering and Prediction Problems*”. Journal of Basic Engineering: 35–45.
- [77]. R.E. Kalman (1960). “*Contributions to the theory of optimal control*”, Bol. Soc. Mat. Mexicana: 102–119.