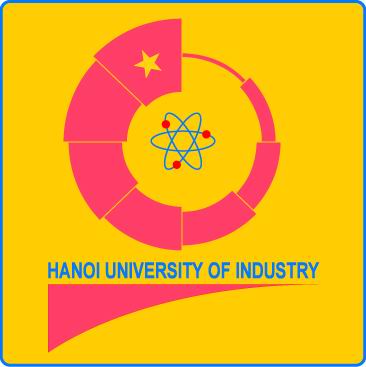
**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**



**ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CẤP TRƯỜNG**

**NGHIÊN CỨU KỸ THUẬT PHÂN ĐOẠN CHUYỂN ĐỘNG VIDEO**

Đơn vị thực hiện : Khoa công nghệ thông tin

Giáo viên hướng dẫn : Ths. **Ngô Đức Vĩnh**

Nhóm sinh viên : Nhóm DST

1. Nguyễn Hồng Gấm – ĐH KHMT3 – K5

2. Bùi Duy Hùng – ĐH KHMT1 – K3 *(Chủ nhiệm)*

3. Lê Văn Hinh – ĐH KHMT1– K3

4. Nguyễn Xuân Minh – ĐH KTPM CLC 1 – K6

5. Phạm Việt – ĐH KHMT1 – K3

*--- Hà Nội, tháng 4 – 2012 ---*

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

Giáo viên hướng dẫn

**NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN**

Giáo viên phản biện

# MỤC LỤC

[MỤC LỤC 3](#_Toc323885973)

[LỜI NÓI ĐẦU 8](#_Toc323885974)

[Chương 1: MỞ ĐẦU 9](#_Toc323885975)

[1.1. Lý do chọn đề tài 9](#_Toc323885976)

[1.2. Mục tiêu nghiên cứu 9](#_Toc323885977)

[1.3. Phương pháp nghiên cứu 10](#_Toc323885978)

[1.4. Đối tượng nghiên cứu 10](#_Toc323885979)

[1.5. Dự kiến kết quả nghiên cứu 10](#_Toc323885980)

[1.6. Cấu trúc báo cáo 10](#_Toc323885981)

[Chương 2: NỘI DUNG NGHIÊN CỨU 11](#_Toc323885982)

[2.1. Tổng quan về lĩnh vực nghiên cứu 11](#_Toc323885985)

[2.1.1. Khái quát chung về kỹ thuật phân đoạn chuyển động video 11](#_Toc323885989)

[2.1.2. Khái quát chung về kỹ thuật phân đoạn đối tượng chuyển động 12](#_Toc323885990)

[2.2. Một số bộ lọc trong xử lý tín hiệu số 15](#_Toc323885992)

[2.2.1. Tổng quan về các kỹ thuật lọc 15](#_Toc323885997)

[2.2.1.1. Mô hình không gian trạng thái rời rạc theo thời gian 16](#_Toc323885998)

[2.2.1.2. Ước lượng trạng thái tuyến tính 16](#_Toc323885999)

[2.2.1.3. Tiêu chí đánh giá các bộ lọc 17](#_Toc323886000)

[2.2.2. Bộ lọc Kalman 17](#_Toc323886001)

[2.2.2.1. Bản chất của bộ lọc Kalman 18](#_Toc323886003)

[2.2.2.2. Phương trình Kalman rời rạc 18](#_Toc323886004)

[2.2.2.3. Thuật toán bộ lọc Kalman rời rạc 20](#_Toc323886005)

[2.2.2.4. Đánh giá KF 21](#_Toc323886006)

[2.2.3. Bộ lọc Kalman mở rộng (Extended Kalman Filter - EKF) 21](#_Toc323886007)

[2.2.3.1. Phương trình EKF 22](#_Toc323886009)

[2.2.3.2. Bản chất của lọc Kalman mở rộng (EKF) 22](#_Toc323886010)

[2.2.3.3. Đánh giá EKF 25](#_Toc323886011)

[2.2.4. Bộ Lọc Unscented Kalman 25](#_Toc323886012)

[2.2.4.1. Biến đổi Unscented (UT – Unscented Transform) 25](#_Toc323886014)

[2.2.4.2. Unscented Kalman Filter (UKF) 26](#_Toc323886015)

[2.2.4.3. Đánh giá UKF 28](#_Toc323886016)

[2.2.5. Lọc đa mô hình (Multiple Model Estimator) 28](#_Toc323886017)

[2.2.5.1. Giới thiệu lọc đa mô hình 28](#_Toc323886019)

[2.2.5.2. Lọc đa mô hình cố định (Fixed Multiple Model - FMM) 29](#_Toc323886020)

[2.2.5.3. Lọc đa mô hình động (Dynamic Multiple Model) 30](#_Toc323886021)

[2.2.5.4. Lọc đa mô hình tương tác (Interactive Multiple Model) 31](#_Toc323886022)

[2.3. Bài toán phân đoạn và theo vết đa đối tượng chuyển động 32](#_Toc323886023)

[2.3.1. Bài toán phát hiện đối tượng chuyển động trong video 33](#_Toc323886025)

[2.3.1.1. Tổng quan 33](#_Toc323886028)

[2.3.1.2. Tiếp cận dựa trên đặc trưng của ảnh và các bộ phân lớp 33](#_Toc323886029)

[2.3.1.3. Cách tiếp cận dựa trên thuật toán trừ nền và ứng dụng 34](#_Toc323886030)

[2.3.2. Kỹ thuật theo dõi đối tuợng chuyển động sử dụng Optical flow 38](#_Toc323886031)

[2.3.2.1. Tổng quan 38](#_Toc323886033)

[2.3.2.2. Trường chuyển động (motion field) 39](#_Toc323886034)

[2.3.2.3. Phương trình optical flow 39](#_Toc323886035)

[2.3.2.4. Đánh giá 42](#_Toc323886036)

[2.3.3. Median Flow - một cải tiến hiệu quả của Optical flow 42](#_Toc323886037)

[2.3.3.1. Tổng quan về Median flow và trình tự thực hiện 42](#_Toc323886039)

[2.3.3.2. Tính lỗi và lọc 43](#_Toc323886040)

[2.3.3.3. Forward-Backward 43](#_Toc323886041)

[2.3.3.4. So khớp mẫu (Match Template) 43](#_Toc323886042)

[2.3.3.5. Lọc trung vị 44](#_Toc323886043)

[2.3.3.6. Tính vị trí và kích thước thay đổi của đối tượng 45](#_Toc323886044)

[2.3.4. Bài toán tương quan dữ liệu 46](#_Toc323886045)

[2.3.4.1. Vai trò của tương quan dữ liệu trong bài toán theo dõi đa đối tượng chuyển động 46](#_Toc323886047)

[2.3.4.2. Kỹ thuật tương quan dữ liệu ứng dụng trong theo dõi đa đối tượng chuyển động. 48](#_Toc323886048)

[2.3.4.3. Thuật toán Hungary 49](#_Toc323886049)

[Chương 3: Đề xuất mô hình cho kỹ thuật phân đoạn chuyển động 53](#_Toc323886050)

[3.1. Mô hình chung của bài toán đề xuất 53](#_Toc323886053)

[3.1.1. Phát hiện đối tượng chuyển động 53](#_Toc323886058)

[3.1.2. Sử dụng kỹ thuật trừ ảnh trung bình cho bài toán phát hiện đối tượng chuyển động 54](#_Toc323886059)

[3.1.3. Áp dụng bộ lọc kết hợp cùng thuật toán Median flow cho bài toán theo vết các vùng chuyển động 54](#_Toc323886060)

[3.1.3.1. Mô hình hệ thống 54](#_Toc323886066)

[3.1.3.2. Mô hình của bộ lọc Kalman 55](#_Toc323886067)

[3.1.4. Áp dụng thuật toán Hungary giải quyết bài toán tương quan dữ liệu 56](#_Toc323886068)

[Chương 4: Thực nghiệm và kết luận 57](#_Toc323886069)

[4.1. Điều kiện thực nghiệm 57](#_Toc323886071)

[4.1.1. Môi trường 57](#_Toc323886074)

[4.1.2. Thiết bị 57](#_Toc323886075)

[4.1.3. Cấu hình máy 57](#_Toc323886076)

[4.1.4. Khoảng cách thực nghiệm 57](#_Toc323886077)

[4.2. Thực nghiệm kỹ thuật phân đoạn video dựa trên các kỹ thuật lọc 57](#_Toc323886078)

[4.2.1. Phương pháp thực nghiệm 57](#_Toc323886080)

[4.2.2. Kết quả thực nghiệm 58](#_Toc323886081)

[4.3. Kết luận 61](#_Toc323886082)

[4.4. Hướng nghiên cứu 61](#_Toc323886083)

[TÀI LIỆU THAM KHẢO 62](#_Toc323886084)

DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| **STT** | **TỪ VIẾT TẮT** | **ĐẦY ĐỦ** |
|  | KF | Kalman Filter – Bộ lọc Kalman |
|  | EKF | Extended Kalman Filter – Bộ lọc Kalman mở rộng |
|  | UKF | Unscented Kalman Filter – Bộ lọc Unscented Kalman |
|  | PF | Particle Filter |
|  | IMM | Interactive Multiple Model – Mô hình lọc tương tác |
|  | OpenCV | Thư viên xử lý ảnh mã nguồn mở Open Computer Vision library |
|  | EmguCV | Thư viện xử lý ảnh OpenCV được đóng gói để lập trình trên môi trường .NET |
|  | Filter | Bộ lọc nhiễu trong bài toán xử lý tín hiệu số |

DANH MỤC BẢNG BIỂU

[Bảng 1: Xác suất ghép cặp đối tượng 48](#_Toc323884201)

[Bảng 2: Bảng kết quả ghép cặp 49](#_Toc323884202)

[Bảng 3: Bảng phân công m công việc cho n công nhân 50](#_Toc323884203)

[Bảng 4: Biểu diễn ma trận của thuật toán Hungary 50](#_Toc323884204)

[Bảng 5: Kết quả phân công cho bài toán giao m công việc cho n công nhân 51](#_Toc323884205)

DANH MỤC HÌNH ẢNH

[Hình 1: Kỹ thuật phân đoạn chuyển động video ứng dụng theo dõi phương tiện giao thông 11](#_Toc323884206)

[Hình 2: Mô hình kỹ thuật phân đoạn chuyển động Video 12](#_Toc323884207)

[Hình 3: Áp dụng GPCA để giảm thiểu số chiều của không gian con 13](#_Toc323884208)

[Hình 4: Áp dụng kỹ thuật học không giám sát cho bài toán phân đoạn chuyển động 14](#_Toc323884209)

[Hình 5: Thuật toán đề xuất của Bing Han, William Robert, Dapeng Wu, Jian Li 14](#_Toc323884210)

[Hình 6: Kỹ thuật đề xuất của A. Colombari, A. Fusiello, V. Murino 15](#_Toc323884211)

[Hình 7: Ứng dụng lọc Kalman để lọc tín hiệu thu được từ toạ độ con trỏ của người dùng 18](#_Toc323884212)

[Hình 8: Mô hình hoạt động của mạch lọc Kalman 18](#_Toc323884213)

[Hình 9: Hình trên tóm tắt cơ bản của mạch lọc Kalman 21](#_Toc323884214)

[Hình 10: Ví dụ sử dụng EKF của J. Hartikainen, A. Solin, S. Sarkka - Aalto University, August 2011 22](#_Toc323884215)

[Hình 11: Mô phỏng tư tưởng của UT 25](file:///G:\Video%20Segmentation\HungBD_TTTN_Filters_Apply_To_MotionSegmentation.docx#_Toc323884216)

[Hình 12: Ví dụ sử dụng UKF của Rambabu Kandepu a, Bjarne Foss a, Lars Imsland b 28](#_Toc323884217)

[Hình 13: Mô hình của kỹ thuật FMM 29](#_Toc323884218)

[Hình 14: Sự phức tạp của DMM 30](#_Toc323884219)

[Hình 15: Mô hình hoạt động của IMM 31](#_Toc323884220)

[Hình 16: Các thành phần cơ bản của hệ thống theo vết đa đối tượng 32](#_Toc323884221)

[Hình 17: Ví dụ về bài toán phát hiện bàn tay chuyển động 34](#_Toc323884222)

[Hình 18: Mô hình chung của kỹ thuật trừ nền 35](#_Toc323884223)

[Hình 19: Một ví dụ về kỹ thuật trừ ảnh 35](#_Toc323884224)

[Hình 20: Làm trơn ảnh sử dụng bộ lọc ảnh 36](#_Toc323884225)

[Hình 21: Khoanh vùng đối tượng chuyển động 36](#_Toc323884226)

[Hình 22: Kỹ thuật trừ nền trước bộc lộ nhược điểm 37](#_Toc323884227)

[Hình 23: Motion field trong Optical flow 38](#_Toc323884228)

[Hình 24: Hình ảnh minh họa theo dõi 2 đối tượng chuyển động bằng kỹ thuật xử lý ảnh và bộ lọc. 47](#_Toc323884229)

[Hình 25: Hình ảnh minh họa theo dõi 2 đối tượng chuyển động bằng kỹ thuật tương quan dữ liệu. 48](#_Toc323884230)

[Hình 26: Mô tả các khả năng ghép đối tượng 48](file:///G:\Video%20Segmentation\HungBD_TTTN_Filters_Apply_To_MotionSegmentation.docx#_Toc323884231)

[Hình 27: Mô hình bài toán đề xuất 53](#_Toc323884232)

[Hình 28: Mô hình đề xuất của kỹ thuật trừ nền trung bình 54](#_Toc323884233)

[Hình 29: Mô hình đề xuất sử dụng Median Flow và Kalman Filter cho bài toán phân đoạn chuyển động 54](#_Toc323884234)

[Hình 30: Mô hình đề xuất sử dụng thuật toán Hungary cho bài toán Data Association 56](#_Toc323884235)

# LỜI NÓI ĐẦU

*---*

Nhiều năm trở lại đây, đi cùng với sự phát triển của kỷ nguyên công nghệ thông tin, kỹ thuật xử lý ảnh và thị giác máy tính đã có những vị trí nhất định trong sản xuất công nghiệp, quốc phòng, đời sống hàng ngày của chúng ta.

Kỹ thuật phân đoạn chuyển động nói chung và phân đoạn chuyển động trong video nói riêng là một nhánh của Thị giác máy tính, được rất nhiều nhà nghiên cứu dày công xây dựng và phát triển các kỹ thuật làm phong phú thêm kho tàng tri thức và đưa ra những ứng dụng vô cùng thiết thực. Kỹ thuật phân đoạn chuyển động video xuất hiện trong quốc phòng, trong sản xuất công nghiệp, trong hệ thống giải trí, trong truyền thông, và trong đời sống hàng ngày.

Một trong những thành phần quan trọng của kỹ thuật phân đoạn chuyển động video không thể không kể đến là các bộ lọc, hay còn gọi là các kỹ thuật lọc. Các kỹ thuật lọc tín hiệu trong xử lý tín hiệu số có ứng dụng vô cùng quan trọng phục vụ việc chế tạo các thiết bị đo lường, ước lượng và đánh giá các số liệu kinh tế phục vụ sự phát triển kinh tế, theo dõi các đối tượng chuyển động phục vụ hệ thống giám sát trong an ninh và quốc phòng, …

Dựa vào những nhận định trên, chúng tôi nhận thấy các bộ lọc cũng như các kỹ thuật lọc là một sân chơi mở dành cho những nghiên cứu có giá trị. Vì vậy, dưới sự chỉ dẫn của Ths Ngô Đức Vĩnh, chúng tôi quyết định lựa chọn đề tài “*Nghiên cứu một số bộ lọc ứng dụng trong kỹ thuật phân đoạn chuyển động video*” làm đề tài thực tập tốt nghiệp của mình, với hy vọng có thể đóng góp những đánh giá – nhận định có giá trị góp phần làm phong phú hơn nữa lĩnh vực nghiên cứu này.

Sau cùng, chúng tôi xin chân thành cảm ơn Thạc sĩ Ngô Đức Vĩnh – trưởng khoa Công nghệ thông tin trường ĐH Công nghiệp Hà Nội đã giúp chúng tôi định hướng đề tài, lựa chọn hướng nghiên cứu, đánh giá và động viên chúng tôi trong suốt thời gian thực hiện đề tài. chúng tôi cảm ơn Ths Nguyễn Phương Nga, Ths Trần Hùng Cường – khoa CNTT đã giúp đỡ chúng tôi trong quá trình phát triển và hoàn thiện từng bước giải thuật cũng như gợi ý những ý tưởng mới để đi đến bước hoàn thiện đề tài.

# MỞ ĐẦU

## 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 lí 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…

Bằng chứng thấy rõ của kỹ thuật Phân đoạn chuyển động: 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 tách 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; Và một ứng dụng nổi bật không thể không nhắc đến, đó là kỹ thuật phân đoạn chuyển động ứng dụng trong việc nén video, 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 thế giới.

Xuất phát từ đó, chúng tôi quyết định lựa chọn đề tài **“Nghiên cứu một số bộ lọc ứng dụng trong kỹ thuật phân đoạn chuyển động video”**. Mong muốn của chúng 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 với những ứng dụng thực tiễn hơn nữa.

## Mục tiêu nghiên cứu

* Nghiên cứu và đánh giá các kỹ thuật nhận dạng hiện tại.
* Đề xuất sử dụng một số kỹ thuật khác và đưa ra sự so sánh với các nghiên cứu trước đó.
* Từ đó đưa ra các phương pháp cải tiến và khắc phục những hạn chế hiện tại.
* Xây dựng mô hình đề xuất cho bài toán đang nghiên cứu
* Thử nghiệm và đưa ra kết luận.

## 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, codeplex.com, sourceforge.net, 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 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 và đánh giá các kết quả nghiên cứu.

## Đối tượng nghiên cứu

* Kỹ thuật trừ ảnh và ứng dụng
* Nguyên lý của kỹ thuật phát hiện đối tượng chuyển động trong video
* Một số bộ lọc tuyến tính
* Một số bộ lọc phi tuyến
* Bộ lọc đa mô hình
* 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.

## Dự kiến kết quả nghiên cứu

* Hoàn thiện nghiên cứu, đánh giá về kỹ thuật lọc và bộ lọc đa mô hình.
* 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 cải tiến của Optical flow kết hợp cùng lọc Kalman để tối ưu hoá bài toán theo dõi đối tượng chuyển động.
* Xây dựng thử nghiệm hệ thống giám sát trong sân trường ĐH Công nghiệp Hà Nội và hệ thống giám sát giao thông
* Đánh giá kỹ thuật và mô hình đề xuất

## 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. Từ đó đưa ra mô hình mới và một số giải pháp cải tiến.

**Chương 3:** Thực nghiệm trên ngôn ngữ lập trình C# và đánh giá kết quả thực nghiệm

**Chương 4:** Kết luận đề tài

# NỘI DUNG NGHIÊN CỨU



## Tổng quan về lĩnh vực nghiên cứu



### Khái quát chung về kỹ thuật phân đoạn chuyển động video

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 [8].

Vai trò của kỹ thuật xử lý video nói chung và phân đoạn chuyển động video nói riêng được thể hiện rõ nhất qua các kỹ thuật nén, đánh chỉ mục và truyền tải video (trong các ứng dụng như Hội nghị trực tuyến, các mạng xã hội, các cuộc gọi video trong mạng 3G, các phần mềm tán gẫu như Yahoo, Skype, …). Và một trong những nhận định quan trọng vai trò của kỹ thuật xử lý video và phân đoạn chuyển động là định nghĩa về MPEG-7. [2]. MPEG-7 đã được định nghĩa như là một chuẩn nén công nghiệp trong tương lai, đáp ứng được sự bùng nổ của công nghệ thông tin và truyền thông trong giai đoạn tiếp theo. Đồng thời, thu hút được rất nhiều các nghiên cứu có giá trị trong các cuộc hội thảo Xử lý ảnh và nhận dạng, xử lý video trong những năm gần đây.



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

*Có thể chia thành hai hướng tiếp cận khác như:*

* Phân đoạn chuyển động với nền video tĩnh
* Phân đoạn chuyển động với nền video thay đổi

Trong đó, bài toán phân đoạn chuyển động với nền thay đổi vẫn đang làm các nhà nghiên cứu về xử lý video đau đầu. [15]

Kỹ thuật phân đoạn chuyển động video còn được chia thành hai hướng tiếp cận như sau:

* Tiếp cận dựa trên phân đoạn vùng và đặc trưng theo vùng
* Tiếp cận dựa trên phân đoạn các đối tượng chuyển động riêng biệt

Trong đó, kỹ thuật phân đoạn các đối tượng chuyển động riêng biệt có độ khó thấp hơn, đồng thời phạm vi ứng dụng cũng hẹp hơn.

### Khái quát chung về kỹ thuật phân đoạn đối tượng chuyển động

Phân đoạn các chuỗi ảnh thành các đối tượng chuyển động khác nhau là bước kế tiếp sau khi phát hiện đối tượng [15]. Việc phân đoạn này dựa trên thông tin vận tốc chuyển động ví dụ như từ các đối tượng ở giai đoạn đầu, ta kết hợp các đối tượng có cùng vận tốc chuyển động theo một ràng buộc nào đó chẳng hạn như tính lân cận.

Việc phát hiện các vùng có cường độ thay đổi đáng kể xảy ra giữa 2 chuỗi khung hình là một giải pháp cơ bản đầu tiên của kỹ thuật nén video [32], đã thể hiện được những giá trị khoa học, giá trị công nghiệp và giá trị kinh tế vô cùng to lớn. Không thể không kể đến kỹ thuật Motion JPEG 2 (MPEG2), làm nền tảng cho các kỹ thuật nén hiện nay.

Các bước cơ bản để phân đoạn từng đối tượng chuyển động được mô tả bởi hình sau:



**PHÁT HIỆN CHUYỂN ĐỘNG**

**THEO VẾT CHUYỂN ĐỘNG**

*Video*

*Chuỗi các hình ảnh*

*Phân đoạn chuyển động*

Hình : Mô hình kỹ thuật phân đoạn chuyển động Video

Một số kỹ thuật phân đoạn chuyển động nổi bật có thể kể đến như:

* ***Phân đoạn chuyển động dựa trên trường chuyển động (motion field) của kỹ thuật luồng quang học (Optical flow)*** [33]

Kỹ thuật trên được giới thiệu bởi Alessandro Verri, Sergio Uras & Enrico De Micheli năm 1989.



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.

* **Phân đoạn chuyển động dựa trên phân đoạn video thành từng vùng**

*Yucel Altunbasak, P. Erhan Eren, và A. Murat Tekalp* đề xuất kỹ thuật phân đoạn chuyển động dựa trên vùng màu sắc có tham số [26]. 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.



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

*M. Pawan Kumar, P.H.S. Torr, A.Zisserman* 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. [6] 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.

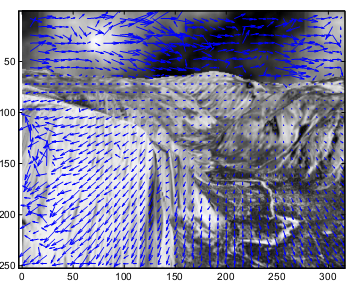
+ Sử dụng thuật toán tráo đổi và thuật toán mở rộng .



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

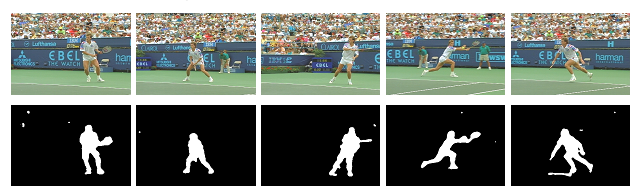
* **Phân đoạn và theo dõi các đối tượng chuyển động dựa trong video**

*Bing Han, William Roberts, Dapeng Wu, Jian Li* 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. [8].



Hình : Thuật toán đề xuất của Bing Han, William Robert, Dapeng Wu, Jian Li

*A. Colombari, A. Fusiello, V. Murino* 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. [7] 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 ccác nguyên lý của khoảng cách và nguyên lý của loại trừ tương ứng.



Hình : Kỹ thuật đề xuất của A. Colombari, A. Fusiello, V. Murino

Với các kỹ thuật đã tìm hiểu phía trên, chúng 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, chúng 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.

## 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 quác 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. [3]

Trong phần này, chúng 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.



### 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…) [22]

Trạng thái hệ thống được chúng ta mô hình hoá [[1]](#footnote-1) 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*
* *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, chúng tôi chỉ đưa ra các nghiên cứu và đánh giá về kỹ thuật lọc với trạng thái rời rạc theo thời gian.

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

***Trong đó:***

là trạng thái của hệ thống tại thời điểm k

là kết quả đo đạt được của hệ thống tại thời điểm k

là mô hình động thể hiện các đặc tính của hệ thống, cũng như cách mà hệ thống thay đổi theo thời gian

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 dựa trên và độc lập với các trạng thái và đo đạc trong quá khứ.

#### Ước lượng trạng thái tuyến tính

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

Trong đó:

* là trạng thái của hệ thống tại thời điểm k
* là kết quả đo đạt được của hệ thống tại thời điểm k
* ) là nhiễu hệ thống tại thời điểm k-1
* là nhiễu đo đạc tại thời điểm k
* là ma trận chuyển trạng thái của hệ thống
* là ma trận của mô hình đo đạc

#### 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). [3]

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.

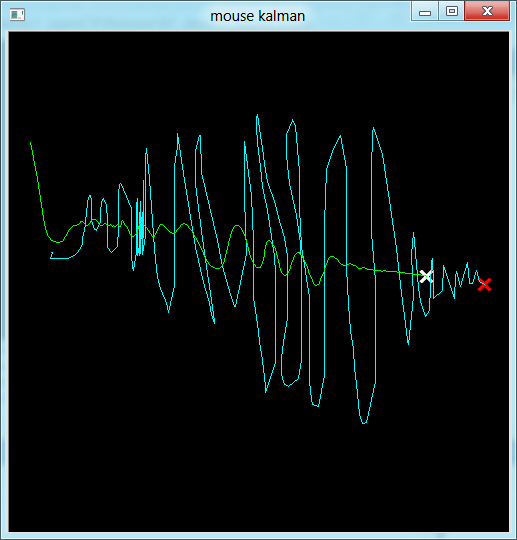
Bộ lọc nào có MSE càng thấp, chứng tỏ độ lệch giữa giá trị thực tế với giá trị ước lượng gần sát nhau. Vì vậy bộ lọc đó đáp ứng được yêu cầu của hệ thống.

### 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[38] [39]. 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 em, 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. [25]



Hình : Ứng dụng lọc Kalman để lọc tín hiệu thu được từ toạ độ con trỏ của người dùng



#### Bản chất của bộ lọc Kalman

Tín hiệu đo X

**Mạch lọc Kalman**

Mô hình hệ thống

X

Hình : 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 [37]. Để có thể ứng dụng một cách hiểu 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).

#### Phương trình Kalman rời rạc

##### 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:

*xk* = *Axk* – 1 + *wk* – 1

và kết quả đo đạc

*zk = H \* xk + vk*

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) ∼ N(0, Q)*

*p(v) ∼ N(0, R)*

Nếu vector trạng thái x có kích thước là n, thì ma trận A sẽ có kích thước là n x n. B (n x l) là ma trận phụ thuộc vào điều khiển tối ưu u với u la vector có kích thước là l. Vector đo đạc z có kích thước là m nên ma trận H sẽ là m x 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. [3]**

##### Các tính toán của bộ lọc

Nếu ta giả sử k- và klần 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 z­k (2). Khi đó sai số ước đoán ước đoán tiên nghiệm và hậu nghiệm lần lượt là:

*ek- = xk - k-*

*ek = xk - k*

Ước tính ma trận hiệp phương sai lỗi tiền nghiệm.

*Pk- = E[ ek- ek-T]*

e-T là ma trận chuyển vị của e -

Ước tính ma trận hiệp phương sai lỗi hậu nghiệm.

*Pk = E[ ekekT]*

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

k = k- + K( zk – H \* k- )

K gọi là độ lợi (grain)

*k : Giá trị hậu nghiệm của ước lượng x*

*k- : Giá trị tiền nghiệm của ước lượng x*

*Zk : giá trị đo đạc*

*H 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 *ek = K( zk – H \* k- )* là nhỏ nhất. Bằng cách thay ek vào biểu thức tính Pk rồi sau đó lấy đạo hàm của Pk theo K, ta sẽ tìm được K tương ứng với Pk nhỏ nhất.

*Kk = Pk- \* HT (H \* Pk- \* HT + R)-1*

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

##### 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[xk] = k*

*E[(xk – xk‑) \* [(xk – xk‑)T] = Pk*

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(xk / zk) ∼ N(E[xk], E[(xk – xk‑) \* [(xk – xk‑)T] ) = N( k, Pk )*

#### 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 nhập 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 tiền nghiệm
* **Measurement update:** Dựa vào kết quả đo để hiệu chỉnh lại ước đoán.

|  |
| --- |
| **CẬP NHẬT THỜI GIAN**  **(dự đoán)** |
| Dự đoán trạng thái trước    Dự đoán hiệp phương sai trước |

|  |
| --- |
| **CẬP NHẬT PHÉP ĐO**  **(sửa)** |
| 1. Tính độ lợi 2. Cập nhật dự đoán với đo đạc 3. Cập nhật hiệp phương sai lỗi |

Ước đoán ban đầu và

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

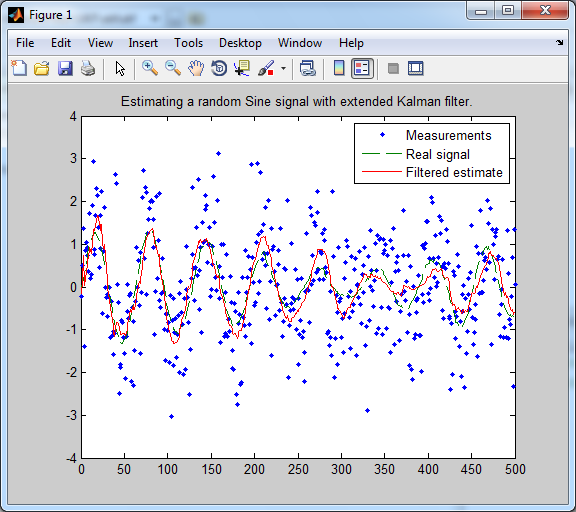
#### Đánh giá KF

* 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. [22]

### Bộ lọc Kalman mở rộng (Extended Kalman Filter - EKF)

Bộ lọc Kalman thông thường chỉ đáp ứng được các hệ thống phi tuyến có giả định nhiễu Gauss. Trên thực tế, có rất ít hệ thống có các điều kiện lý tưởng cho Kalman Filter hoạt động tốt. Vì vậy, Kalman Filter thường ít được triển khai trong các hệ thống thực tế. Tuy nhiên, không phải vì vậy mà ta kết luận Kalman Filter kém hơn so với các bộ lọc phi tuyến khác. Trên thực tế, có các hệ thống khi triển khai Kalman Filter cho ta hiệu quả tốt hơn hẳn, bởi đặc tính của hệ thống đó là trạng thái biến đổi tuyến tính.[22]

Jazwinski, 1996; Maybeck, 1982; Bar-Shalom et. al., 2001; Grewal and Andrews, 2001; Sarkka, 2006; [3] đã có những nghiên cứu để mở rộng bộ lọc Kalman tuyến tính, từ đó đề xuất một bộ lọc có tên Lọc Kalman mở rộng (extended Kalman filter - EKF).



Hình : Ví dụ sử dụng EKF của J. Hartikainen, A. Solin, S. Sarkka - Aalto University, August 2011

Trong ví dụ của cuốn sách Optimal Filtering with Kalman Filter and Smoothers của J. Hartikainen, A. Solin, S. Sarkka – trường ĐH Aalto, họ sử dụng EKF để ước lượng việc đo đạc một hệ thống có trạng thái biến đổi theo dạng sin – nghĩa là trạng thái biến đổi phi tu­­­yến.



#### Phương trình EKF

* là trạng thái của hệ thống tại thời điểm k
* là kết quả đo đạt được của hệ thống tại thời điểm k
* ) là nhiễu hệ thống tại thời điểm k-1
* là nhiễu đo đạc tại thời điểm k
* là hàm phi tuyến, mô tả sự biến thiên trạng thái của hệ thống
* là hàm phi tuyến của mô hình đo đạc

#### Bản chất của lọc Kalman mở rộng (EKF)

Xuất phát từ phương trình EKF ở mục a:

Về bản chất của bộ lọc là ta đi xấp xỉ giá trị của trạng thái hệ thống, vì vậy, ta hoàn toàn có thể xấp xỉ cả phương trình EKF như sau:

* Coi nhiễu hệ thống = 0
* Coi nhiễu đo đạc = 0 [[2]](#footnote-2)

Ta sẽ có được giá trị xấp xỉ của và như sau:

(\*\*)

Với là ước lượng hậu nghiệm của trạng thái hệ thống.

Khi đó, ta sẽ tính ngược trở lại trạng thái hệ thống ban đầu:

Trong đó:

* và lần lượt là trạng thái và đo đạc thực tế
* và lần lượt là trạng thái và đo đạc có được qua quá trình xấp xỉ
* là ước lượng hậu nghiệm tại thời điểm k
* và lần lượt là nhiễu xử lý và nhiễu đo đạc
* là ma trận Jacobi của đạo hàm riêng của *f( )* theo *x*
* W là ma trận Jacobi của đạo hàm riêng của *f( )* theo w
* H là ma trận Jacobi của đạo hàm riêng *h( )* theo x
* V là ma trận Jacobi của đao hàm riêng *h( )* theo v

Từ các bước trên, ta định nghĩa lỗi cho quá trình ước đoán :

⬄ (\*1)

Và lỗi của quá trình đo đạc:

⬄ (\*2)

Với và là biến ngẫu nhiên có trung bình bằng 0 và có ma trận hiệp phương sai là WQWT và VRVT, với Q, R được định nghĩa ở phương trình Kalman tổng quát.

*Chúng ta dễ thấy rằng (\*1) và (\*2)là phương trình ở dạng tuyến tính, có cấu trúc tương tự với phương trình Kalman. Điều này thôi thúc chúng ta sử dụng sai số của đo đạc thực tế và việc sử dụng Kalman Filter để ước lượng lỗi của quá trình dự đoán.*

Với ước lượng này, chúng ta coi nó là  *, có thể được sử dụng để thu được một ước lượng hậu nghiệm cho quá trình phi tuyến ban đầu:*

(\*3)

Các biến ngẫu nhiên được xấp xỉ với phân phối xác suất sau:

Khi đó, ta đưa ra giá trị xấp xỉ và để giá trị dự đoán của là 0

Thay vào (\*3) ta có :

Từ đây, ta có thể áp dụng Kalman Filter cho trường hợp tổng quát, với tính tương tự như phần 4.1.2.2.b và , được xấp xỉ tương tứng ở phương trình (\*\*)

***Tóm lại, bản chất của EKF thể hiện ở các yếu tố sau:***

* Từ mô hình hệ thống phi tuyến, chúng ta tuyến tính hoá
  + Phương trình chuyển trạng thái của hệ thống
  + Phương trình tương quan giữa trạng thái hệ thống và kết quả đo đạc

để áp dụng lọc Kalman Filter thông thường.

* Lan truyền và xấp xỉ của kỳ vọng có điều kiện và phương sai
* Tư tưởng chủ đạo của EKF là việc sử dụng khai triển chuỗi Taylor để xấp xỉ các ước lượng.

#### Đánh giá EKF

* *Ưu điểm:*
  + Nếu điều chỉnh đúng mô hình hệ thống và các tham số, EKF hoạt động rất tốt và ổn định.
  + Được sử dụng rộng rãi trong thực tế và chứng minh tính hiệu quả trong các hệ thống đó
* *Nhược điểm: [22]*
  + Nếu trạng thái khởi tạo không đúng hoặc hệ thống được mô hình hóa không đúng đắn, bộ lọc sẽ rất nhanh chóng đi chệch hướng và hoạt động kém hiệu quả
  + Một vấn đề khác với các bộ lọc Kalman mở rộng là ma trận hiệp phương sai ước tính có xu hướng đánh giá thấp các ma trận hiệp phương sai đúng và do đó rủi ro trở nên không phù hợp trong ý nghĩa thống kê mà không có sự bổ sung của "nhiễu bình ổn".
  + Không phải với hàm *f( )* nào của hệ thống cũng có thể xấp xỉ bởi một ma trận hệ số Jacobian. Vì vậy, không phải hệ thống phi tuyến nào cũng có thể áp dụng EKF
  + Rất khó để đưa ra mô hình hệ thống và triển khai. Trong rất nhiều trường hợp, việc tính toán ma trận Jacobian và Hessian là một quá trình phức tạp và khó, hơn nữa rất dễ gặp phải trường hợp có sai sót trong quá trình lấy đạo hàm của hàm phi tuyến *f( )* hoặc quá trình lập trình. Vì vậy, công việc debug rất phức tạp khi chỉ nhìn vào kết quả của ước lượng để tìm đến các lỗi sai.

### Bộ Lọc Unscented Kalman

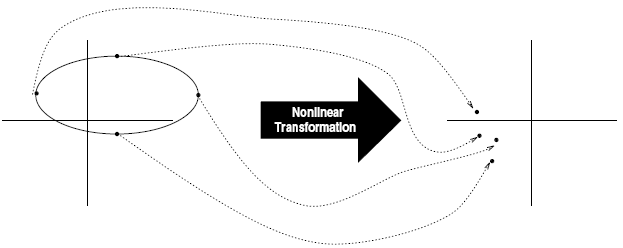


#### Biến đổi Unscented (UT – Unscented Transform)

UT là một phương pháp để tính thống kê của một biến ngẫu nhiên đã trải qua một phép biến đổi phi tuyến.

UT được xây dựng dựa trên tư tưởng: ***Xấp xỉ một phân phối xác suất dễ hơn là việc xấp xỉ một hàm phi tuyến.***

Hình : Mô phỏng tư tưởng của UT



*Sigma points*

Vấn đề của Unscented Transform

* + Lan truyền một biến ngẫu nhiên ***x*** có ***nx*** chiều qua một hàm phi tuyến để tính được ***y*** với
* **Công thức của phép biến đổi UT**

Giả sử x có mean là và hiệp phương sai là *Px*

Một tập gồm **2nx+1** điểm sigma được gán trọng số riêng  được chọn như sau:



Với là hệ số co giãn và  là dòng (hoặc cột) thứ i [[3]](#footnote-3) của ma trận 

*Wi* là trọng số của từng điểm sigma tương ứng, thoả mãn:



* **Lan truyền các điểm sigma**

Mỗi điểm sigma sẽ được lan truyền qua hàm phi tuyến 

Đồng thời ta cũng tính được giá trị mean ước lượng và hiệp phương sai tương ứng:



Ước lượng của mean và hiệp phương sai trên được tính dựa vào khai triển chuỗi Taylor (bậc 2) của hàm *g(x)* với mọi hàm phi tuyến.

#### Unscented Kalman Filter (UKF)

Lọc Unscented Kalman (UKF) là một ứng dụng trực tiếp và cơ bản của phép biến đổi Unscented (Trên thực tế, phép biến đổi được sử dụng là SUT – scaled unscented transform). Trong đó, trạng thái hệ thống là một vector có tham số và hiệp phương sai lỗi cũng vậy. [17]





**Trình tự thực hiện lọc UKF**

* **Khởi tạo**



* **Dự đoán (Time update)**

*+ Tính các điểm sigma*



*+ Time update*



*+ Measurement update*







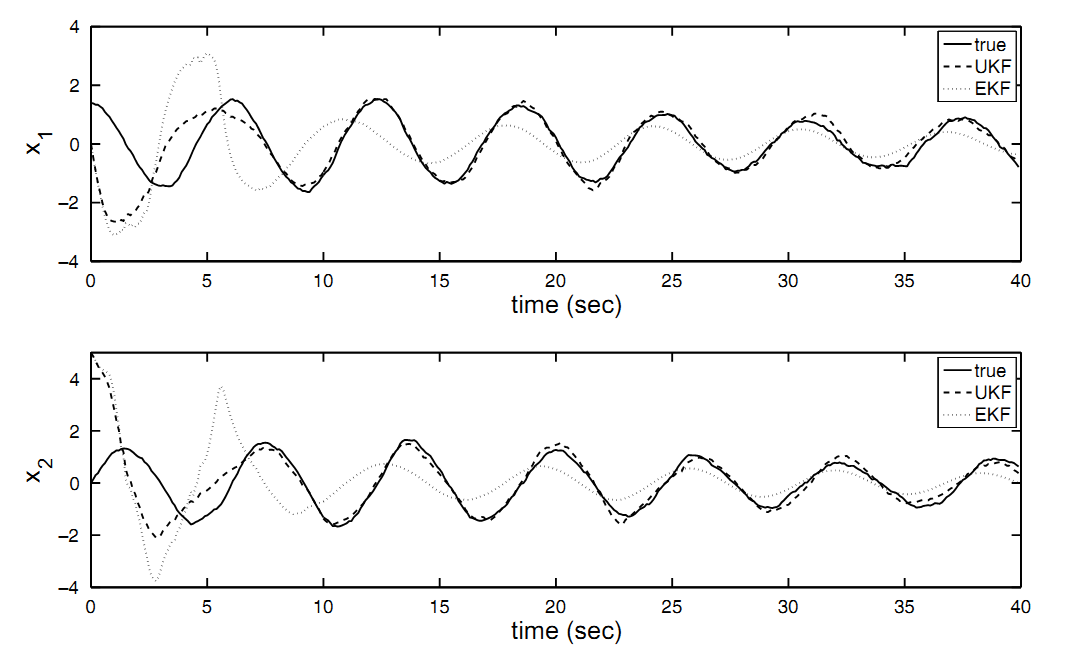




Trong đó:

* *na=nx+nv+nn ,*
* *Q là ma trận hiệp phương sai của nhiễu hệ thống.*
* *R là ma trận hiệp phương sai của nhiễu đo đạc.*
* *K lợi của Kalmain, W là trọng số*

Dưới đây là ví dụ áp dụng UKF của **Rambabu Kandepu a, Bjarne Foss a, Lars Imsland b** trong cuốn sách “*Applying unscented Kalman filter in non-linear state estimation*”



Hình : Ví dụ sử dụng UKF của Rambabu Kandepu a, Bjarne Foss a, Lars Imsland b

#### Đánh giá UKF

* Ưu điểm:
  + Không phải tính ma trận Jacobi và Hessian. Nên có thể triển khai với bất kỳ hàm phi tuyến nào
  + Khắc phục được một số nhược điểm của KF và EKF. Khắc phục nhược điểm “phức tạp” của EKF.
  + UKF có phương pháp tiếp cận dựa trên lấy mẫu rất hiệu quả.
* Nhược điểm:
  + Chỉ hoạt động tốt trên đạo hàm bậc 2 của hàm phi tuyến. Vì vậy có một số mô hình hệ thống, việc lấy đạo hàm bậc 2 thực sự là thử thách
  + Độ phức tạp của UKF phụ thuộc một phần vào số lượng điểm Sigma. *Tương ứng với độ chính xác tăng dần, số điểm sigma cũng tăng theo.*

### Lọc đa mô hình (Multiple Model Estimator)



#### Giới thiệu lọc đa mô hình

Như đã trình bày ở các phần trên, việc mô hình hoá hệ thống là một bài toán phức tạp – khi triển khai các bộ lọc. Các nghiên cứu cho thấy việc áp dụng các mô hình một cách hiệu quả là một quá trình tích luỹ kinh nghiệm. Nghĩa là chúng ta không thể biết chính xác mô hình hiệu quả đến mức nào mà chỉ có thể sử dụng một vài trường hợp mẫu để đánh giá bộ lọc đó.

Cũng từ điều này chúng ta nhận thấy, một bộ lọc – tương ứng với một mô hình sẽ chỉ hoạt động tốt trong các trường hợp được gọi là lý tưởng (Hoặc hiểu theo một cách khác là: hoạt động trong trường hợp đặc biệt – mà kỹ thuật cũng như mô hình đó sinh ra để giải quyết trường hợp này).

Từ đó, chúng ta nảy sinh ra một ý tưởng mới – đó là việc kết hợp nhiều mô hình lại. Các mô hình khác nhau sẽ có nhiệm vụ là khắc phục nhược điểm của nhau, để đưa ra một kết quả lọc tốt nhất có thể. Đó là lý do kỹ thuật lọc đa mô hình ra đời. [20]

Kỹ thuật lọc đa mô hình là một hệ thống lai giữa nhiều hơn một bộ lọc, được duy trì tại một thời điểm trong hệ thống. Các hệ thống lọc đa mô hình, thường triển khai đan xen giữa một vài bộ lọc tuyến tính và một vài bộ lọc phi tuyến.

Thông thường, có 3 loại lọc đa mô hình chính:

* Lọc đa mô hình cố định
* Lọc đa mô hình động
* Lọc đa mô hình tương tác

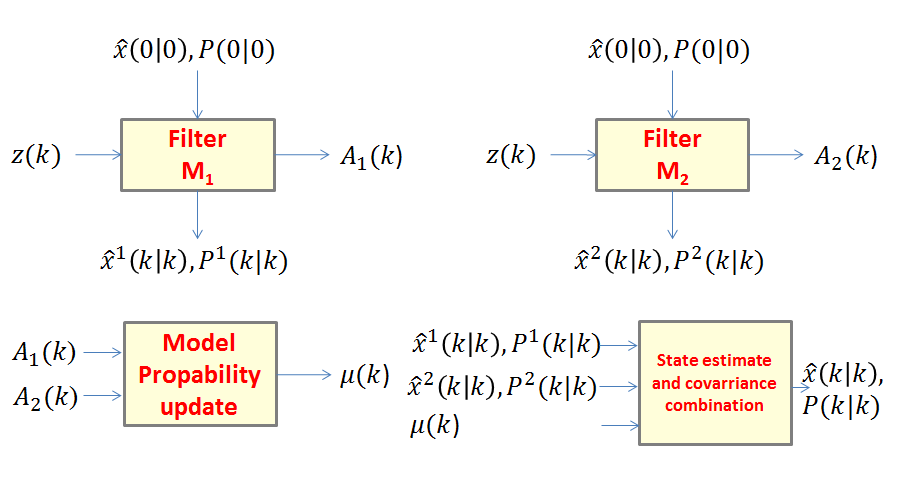
Chi tiết về 3 loại lọc đa mô hình sẽ được trình bày ở phía dưới.

#### Lọc đa mô hình cố định (Fixed Multiple Model - FMM)

Với kỹ thuật lọc đa mô hình cố định, tại một thời điểm hệ thống sẽ duy trì một số lượng bộ lọc cố định.

Mỗi bộ lọc sẽ có một trọng số riêng – trọng số thể hiện mức độ “đóng góp” của bộ lọc đó với kết quả lọc của hệ thống. Trọng số sẽ được cập nhật qua mỗi lần lọc.

Dưới đây là mô hình của kỹ thuật FMM với 2 bộ lọc M1 và M2



Hình : Mô hình của kỹ thuật FMM

#### Lọc đa mô hình động (Dynamic Multiple Model)

Với kỹ thuật lọc đa mô hình động, tại một thời điểm, hệ thống sẽ chỉ lựa chọn một hoặc một số bộ lọc để thực hiện quá trình lọc và đưa ra kết quả.

Dưới đây là mô hình chung của lọc đa mô hình động:

X(k) = F[ M(k)]\*X(k-1) + v[k-1, M(k)]

Z(k) = H[M(k)]\*X(k-1) + w[k, M(k)]

Với

* M(k) – mô hình hệ thống “*tại thời điểm k*” – M(k) có ảnh hưởng đến hệ thống cho đến hết chu kỳ lọc thứ k
* Với từng quá trình lọc, ta lưu lại lịch sử sử dụng các mô hình

Lịch sử sử dụng mô hình tại thứ lth (là chuỗi các mô hình tuần tự)

Với

* Quá trình lựa chọn các mô hình để lọc cho hệ thống chung – gọi là quá trình nhảy – quá trình này tuân theo tiến trình Markov.

**Vấn đề của lọc đa mô hình là:**

t

t

1

21

321

421

521

621

721

*Số vết cần lưu lại tăng theo cấp số nhân so với k*

Hình : Sự phức tạp của DMM

Ví dụ trên, ta sử dụng 2 mô hình M1, M2. Số bước phải lưu lại tại thời điểm k là 2k

Điều này có nghĩa là xuất hiện sự bùng nổ trong quá trình lưu vết.

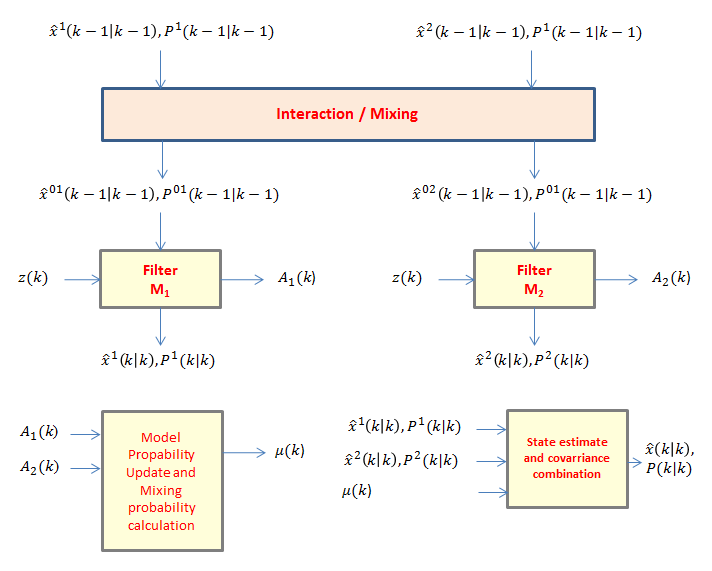
Để giải quyết vấn đề trên, chúng ta có 2 cách chính:

* Cách 1: Chỉ lưu vết các mô hình có xác suất lớn
* Cách 2: Sử dụng các thuật toán tối ưu:
  + IMM (Interactive Multiple Model) sẽ đuợc trình bày ở phần sau
  + GPB

#### Lọc đa mô hình tương tác (Interactive Multiple Model)

Tại thời điểm k, uớc luợng được tính toán dựa trên mỗi mô hình hiện tại của r bộ lọc. Tuơng ứng với mỗi bộ lọc, ta sẽ tính xác suất của giá trị ước lượng được thông qua bộ lọc đó, so với giá trị đo đạc đựoc.

Mỗi bộ lọc sẽ sử dụng các cách khác nhau để kết hợp các mô hình truớc đó lại thành một ước luợng có điều kiện, từ đó tìm ra giá trị cần xử lý là trạng thái hệ thống.



Hình : Mô hình hoạt động của IMM

Các bước thực hiện của IMM gồm có:

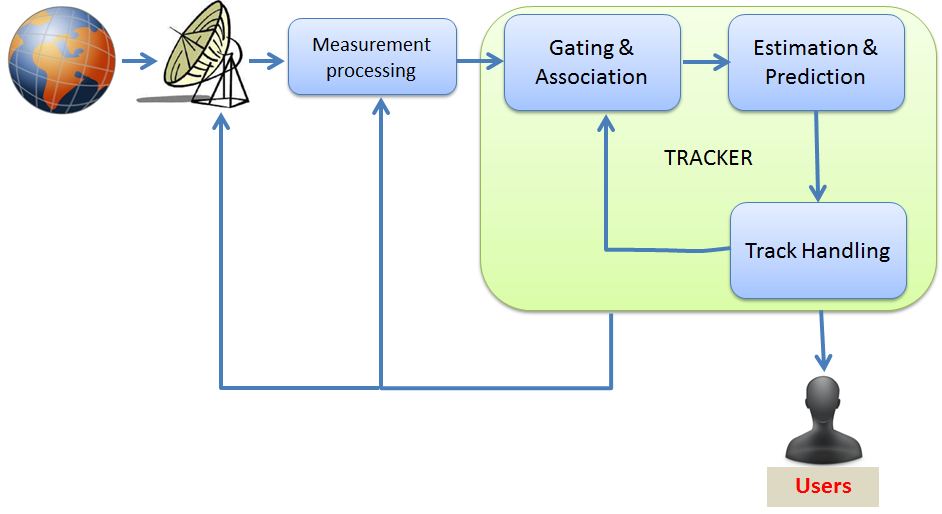
* Tính xác suất trộn giữa các mô hình
* Trộn giá trị uớc lượng từ các mô hình
* Cập nhật xác suất của từng mô hình (Xác suất này có thể coi như trọng số của từng mô hình)
* Ước lượng trạng thái của hệ thống và tính hiệp phương sai

Trên thực tế, IMM là mô hình đuợc sử dụng khá rộng rãi trong radar tracking.

## Bài toán phân đoạn và theo vết đa đối tượng chuyển động

Xuyên suốt những nội dung nghiên cứu, chúng tôi quyết định lựa chọn kỹ thuật theo từng thành phần chuyển động trong video làm nền tảng cho hướng giải quyết của kỹ thuật phân đoạn chuyển động.

Các thành phần cơ bản của bài toán phân đoạn và theo vết đa đối tượng chuyển động



Hình : Các thành phần cơ bản của hệ thống theo vết đa đối tượng

Bài toán theo vết đa đối tượng có một số đặc trưng cơ bản sau:

* Luôn gồm 2 quá trình: **Lọc** và **Tương quan dữ liệu** [10]
* Đảm bảo khả năng duy trì đối tượng
* Các cách tiếp cận rất phong phú, và đã có khá nhiều nghiên cứu có giá trị.
* Luôn phải quan tâm đến độ phức tạp thuật toán.

Với việc xử lý các đối tượng trong xử lý ảnh, không gian trạng thái cũng như số lượng đối tượng ít. Xuất phát từ điều này, chúng ta có thể dễ dàng áp dụng các kỹ thuật lọc và tương quan một cách “tùy ý”. [29]

Tuy nhiên, với bài toán theo dõi đối tượng trong Radar, số lượng đối tượng lên đến hàng nghìn, độ phức tạp của các thuật toán xử lý sẽ tăng lên theo hàng tỷ. Vì vậy, lúc này chúng ta phải quan tâm rất nhiều đến độ phức tạp thuật toán để đưa ra các giải pháp tối ưu – dĩ nhiên luôn cân bằng với yếu tố độ chính xác và độ tức thời.

Vì vậy, trong nội dung này, chúng tôi quan tâm với 3 vấn đề chính:

* Thuật toán phát hiện đối tượng chuyển động
* Kỹ thuật theo vết đa đối tượng chuyển động
* Kỹ thuật tương quan dữ liệu cho bài toán theo vết đa đối tượng chuyển động

### Bài toán phát hiện đối tượng chuyển động trong video



#### Tổng quan

Mục tiêu của bài toán là xác định các đối tượng chuyển động xuất hiện trong frame hiện tại. Kỹ thuật phát hiện đối tượng chuyển động được sử dụng rộng rãi trong thực tế, là bước đầu tiên của các hệ thống theo dõi đối tượng chuyển động, hệ thống giám sát, …

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 [34]. 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 [34]. 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: Trừ ảnh của thời điểm **ngay trước đó**, trừ ảnh với các điểm ảnh có **giá trị trung vị của K thời điểm trước** đó, trừ ảnh với các gia trị điểm ảnh là **trung bình của K thời điểm trước**…[7]
* Dựa trên phân bố Gaussian [15]
* Dựa trên ước lượng MeanShift
* Dựa trên ước lượng và lan truyền
* Dựa vào nền riêng (eigenbackgrounds) [15]

Ngoài ra, có các vấn đề đi song song cùng với bài toán phát hiện đối tượng chuyển động là kỹ thuật loại bỏ bóng (shadow) khỏi đối tượng. 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, chúng tôi sẽ trình bày thuật toán trừ nền với 2 kỹ thuật chủ đạo là trừ nền trước và trừ nền trung bình.

#### Tiếp cận dựa trên đặc trưng của ảnh và các bộ phân lớp

Để giải quyết bài toán phát hiện đối tượng chuyển động, ta có thể quy về bài toán phát hiện đối tượng trong ảnh. Với phương pháp này, ta dễ dàng áp dụng các kỹ thuật phân tích đặc trưng, phân lớp và nhận dạng đối tượng trên ảnh tĩnh để áp dụng cho bài toán phát hiện đối tượng chuyển động.

Tuy nhiên, với cách giải quyết này – đặc trưng của bài toán không được làm rõ.

Một số kỹ thuật có thể được áp dụng như:

* Sử dụng đặc trưng haarlike
* Sử dụng kỹ thuật HOG (Histogram Orriented Gradient)
* Sử dụng các kỹ thuật xử lý ảnh để phân tích đặc trưng cho một số đối tượng quan tâm riêng biệt (chẳng hạn như phát hiện khuôn mặt chuyển động, ta sẽ xây dựng kỹ thuật phân tích đặc trưng riêng cho khuôn mặt.)
* …

Dưới đây là ví dụ mô phỏng cho kỹ thuật phát hiện bàn tay chuyển động.



Hình : Ví dụ về bài toán phát hiện bàn tay chuyển động

*Tuy nhiên, việc áp dụng cách tiếp cận trên – chỉ phù hợp cho từng bài toán cụ thể. Chứ không phù hợp cho đa số bài toán phân đoạn chuyển động. Vì vậy, chúng tôi quyết định không lựa chọn kỹ thuật trên cho bài báo cáo của mình.*

#### Cách tiếp cận dựa trên thuật toán trừ nền và ứng dụng

**+ Tổng quan kỹ thuật trừ nền**

Kỹ thuật trừ ảnh là kỹ thuật sơ khai nhất được áp dụng trong quá trình phát hiện chuyển động trong video hoặc chuỗi hình ảnh. Ưu điểm của kỹ thuật trên là dễ triển khai và độ phức tạp tối thiểu. Tuy nhiên, nhược điểm của thuật toán này thể hiện ở việc chỉ hoạt động hiệu quả trong trường hợp có sự thay đổi rõ rệt trên từng vùng ảnh đến mức mắt thường có thể quan sát được.

**Các bước tổng quát của kỹ thuật ảnh**

Cập nhật nền

Ảnh hiện tại

Ảnh nền

Trừ ảnh

Làm trơn và khoanh vùng

đối tượng

Hình : Mô hình chung của kỹ thuật trừ nền

**Bước 1:** Trừ ảnh

Trong bước này, ta thực hiện việc duyệt qua từng pixel ảnh của 2 ảnh đưa vào 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.



**Trừ ảnh**

Ảnh nền từ thời điểm t – k -> t – 1

*k: số ảnh trong vết được lưu*

frame t+1

*Ảnh sai biệt*

Hình : 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, …

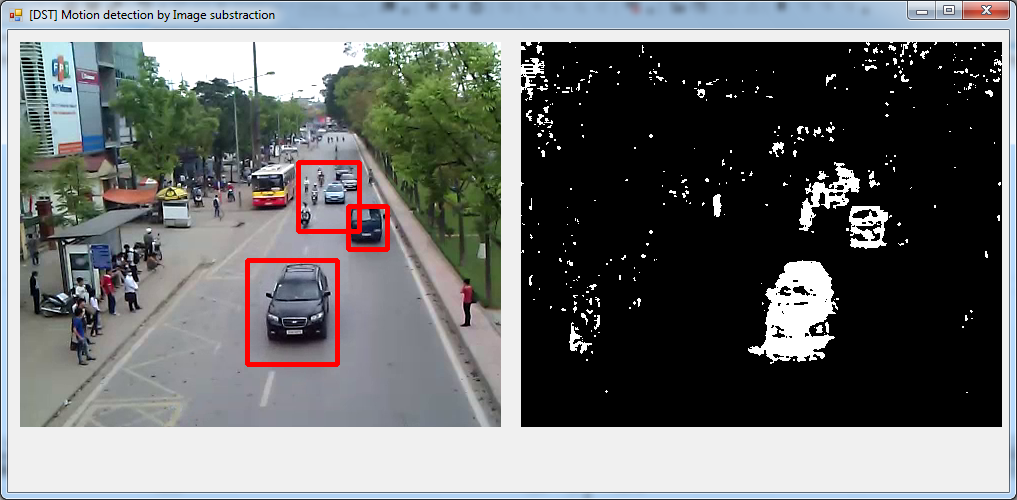
Để loại bỏ yếu tố trên, ta có thể sử dụng các kỹ thuật lọc ảnh để làm trơn ảnh.



Hình : 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. Vì vậy, chúng tôi nghiên cứu và lựa chọn một kỹ thuật khoanh vùng và gán nhãn khác hiệu quả hơn.



Hình : Khoanh vùng đối tượng chuyển động

Thuật toán khoanh vùng đối tượng nhanh – độ phức tạp O(n2)

|  |
| --- |
| Input: Ảnh sai biệt (I\_diff)  Output: Danh sách các đối tượng (listObject)  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]**  done |

**Bước 4:** Cập nhật ảnh nền

*Với kỹ thuật trừ nền, chúng ta thường có 3 cách triển khai:*

* 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

**+ Phát hiện đối tượng chuyển động dựa trên việc tính ảnh sai biệt giữa 2 frame liên tiếp**

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.

|  |
| --- |
| **Input :** Ảnh frame trước (Iprevious) & Ảnh hiện tại (Icurrent)  **Output :** Ảnh sai biệt (Idiff)  **BEGIN**  for pixel (x, y) in Icurrent  if ( ***| get(Icurrent, x, y) – get(Icurrent, x, y) | > motionThreshold*** )  Set(***Idiff***, x, y, white) ;  else  Set(***Idiff***, x, y, black)  **END.** |



Hình : Kỹ thuật trừ nền trước bộc lộ nhược điểm

**+ Phát hiện đối tượng chuyển động dựa vào kỹ thuật ảnh có giá trị điểm ảnh là trung bình của chuỗi K ảnh liên tiếp ngay sau đó**

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. 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. Kết thúc bằng việc cập nhật tập K ảnh bằng cách bớt đi ảnh đầu tiên và thêm ảnh hiện tại vào cuối danh sách.

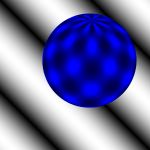
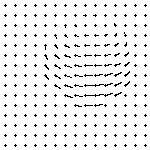
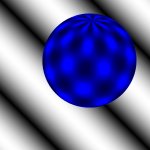
|  |
| --- |
| **Input :** tập gồm K trước đó (listImages) & Ảnh hiện tại (current)  **Output :** Ảnh sai biệt (diff)  **BEGIN**  avg <= getAverageImage(listImages)  for pixel (x, y) in current  if ( ***| get(current, x, y) – get(avg, x, y) | > motionThreshold*** )  Set(diff, x, y, white) ;  else  Set(diff, x, y, black)  listImage <= updateListImages(current)  **END.** |

### Kỹ thuật theo dõi đối tuợng chuyển động sử dụng Optical flow



#### Tổng quan

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 (mắt, camera…). Thuật toán Optical Flow là dùng để xác định xấp xỉ các chuyển động của các điểm ảnh từ frame này sang frame khác trong một khoảng thời gian t.



Ảnh tại thời điểm k

Ảnh tại thời điểm k+1

Trường chuyển động

Hình : Motion field trong Optical flow

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. [28]

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

#### Trường chuyển động (motion field)

Ảnh của một đối tượng di chuyển trước camera sẽ có sự thay đổi tương ứng theo thời gian. Xét một tập các điểm pi di chuyển với vận tốc vi giống như hình dưới đây, thì tập các điểm như vậy được gọi là trường chuyển động (motion field). Mục đích của Optical flow là tìm trường chuyển động này.



Frame n Frame n+1 Trường chuyển động

Ta có các nhận xét sau về trường chuyển động [27]

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

#### Phương trình optical flow

Có nhiều kĩ thuật xác định optical flow khác nhau như: Correlation, matching, feature tracking, energry 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 frame n sang frame n+1***”.

Nghĩa là:

*I(x, y, t) = I(x + u, y + v, t + 1) (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 ( frame thứ t). (u, v) là vecto vận tốc, thể hiện sự thay đổi vị trí của điểm ảnh từ frame thứ t sang frame 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)I(x, y, t) + = I(x, y, t) + Ix\*u + Iy\*v + It (2)*

Trong đó Ix, Iy, It lần lượt là đạo hàm riêng của I(x, y, t) theo x, y, t.

Từ (1) và (2) ta có:

*u \* Ix + v \* Iy + It = 0 (3)*

Phương trình (3) chính là phương trình optical flow. Trong đó (u, v) chính là flow cần tìm.

Phương trình 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 frame t sang t + 1.



Với u = , v =

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

v

u

0







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 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.

 (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:



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

 (5)

Trong đó:







Nghiệm của (5) là : 

***Chú ý:***

* Thông thường ta phải tính M và  tại mỗi điểm (x, y) trên ảnh. Do đó để tăng tốc ta có thể tính trước đạo hàm của ảnh ().
* Nếu dữ liệu trên ảnh đa phần biến thiên theo một chiều nhất định thì M có thể không khả nghịch dẫn đến hệ phương trình không thể giải được.

#### Đánh giá

Với giả thuyết đặt ra cho phương trình Optical Flow, ta phải coi cuờng độ sáng của ảnh không thay đổi từ frame này sang frame kia. Trên thực tế, cuờng 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, Optical flow 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

### Median Flow - một cải tiến hiệu quả của Optical flow



#### Tổng quan về Median flow và trình tự thực hiện

Như chúng ta đã biết, Optical flow 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 *frame n* sang *frame n+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. Và để khác phục điều này thì chúng ta sẽ cùng nhau tìm hiểu thuật toán Median Flow.

Median Flow là một thuật toán cải tiến của Optical flow. Ý tưởng chính của thuật toán này là sử dụng Optical flow để tính vận tốc của các điểm ảnh trên bề mặt đối tượng. Sau đó tính lỗi và chỉ lọc ra dưới 50% số điểm mà đáng tin cậy nhất bằng cách lấy trung vị lỗi.

Figure : Trình tự thực hiện Median Flow

OPTICAL FLOW

Tập hợp các điểm theo dõi được trong hai frame liên tiếp

Tính lỗi

Tập hợp các điểm đáng tin nhất

Dự đoán vị trí và kích thước đối tượng

Theo dõi các điểm trên bề mặt đôiis tượng

*Lọc ra đưới 50% số điểm*

#### Tính lỗi và lọc

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

Figure : Các bước lọc và tính lỗi

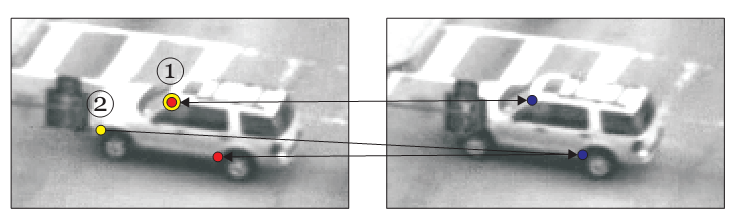
**Forward-Backward**

**Template matching**

**Median filter**

#### Forward-Backward

Là quá trình tìm vị trí của tương ứng của điểm ảnh p1 trong frame thứ i ta được điểm ảnh p2 trong frame thứ i + 1. Sau đó thực hiện ngược lại tìm vị trí của điểm ảnh p3 trong frame thứ i + 1 tương ứng với điểm ảnh p2 trong frame thứ i. Sau đó tính khoảng cách Euclide giữa hai điểm p1 và p3. Khoảng cách này được dùng để đo lỗi của điểm ảnh p1.



frame i frame i + 1

Ở hình trên ta thấy điểm số 1 khi tính điểm tương ứng ở frame thứ i+1 sau đó tính ngược lại ta lại thu được chính nó nên error[1]=0. Đối với điểm thứ hai thì 2 điểm p1, p3 là khác nhau nên error[2] = Euclide(p1, p3).

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.

#### So khớp mẫu (Match Template)

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)**

****

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

****

**+ Hệ số tương quan (CCOEFF\_NORMED)**

****

Trong đó:





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.

#### 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.

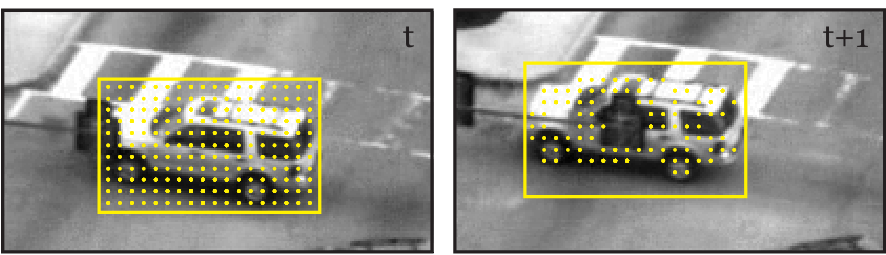


Figure : Lọc trung vị

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

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

**+ Tìm vị trí mới của đối tượng**

Giả sử mỗi điểm pi trong frame t sẽ chuyển động với vận tốc là vi (độ dịch chyển). Như vậy thì vị trí của điểm pi trong frame t+1 sẽ là pi + vi. 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 frame thứ t+1 một cách chính xác ta có thì ta lấy vị trí của frame thứ t cộng với trung vị của độ dịch chuyển.



(u, v)

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

Như ở phần motion field 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.



d2

d1

frame i frame i+1

Giả sử p1­­, p2 là 2 điểm nằm trên bề mặt của đối tượng như hình trên và p1i, p1i+1 tương ứng với các ký hiệu p1 ở trên frame i và p1 ở trên frame i+1. Như vậy độ co giản của đối tượng có thể tính được như sau:



Với d là khoảng cách euclide.

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 giản.

Cuối cùng kích thước của đối tượng trong frame thứ t + 1 ta có thể tính được như sau:

Wi+1 = Wi \* s

Hi­+1 = Hi \* s

### Bài toán tương quan dữ liệu

Kỹ thuật tương quan dữ liệu được sử dụng để thực hiện phép gán dữ liệu đã trải qua quá trình ước lượng cho các bộ theo dõi, ứng dụng trong bài toán theo dõi đa đối tượng chuyển động.



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

Đối với việc theo dõi 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. [10]

Một số kỹ thuật tương quan dữ liệu tiêu biểu:

* Kỹ thuật đơn giả thuyết (Single Hypothesis Tracking)
  + GNN: Sử dụng các thuật toán như Hungary, Munkres, …
  + JPDA
* Kỹ thuật đa giả thuyết (Multiple Hypothesis Tracking)

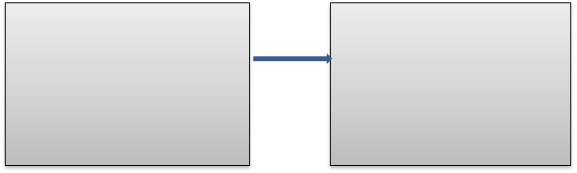
*Trong phạm vi bài báo, chúng tôi chỉ trình bày kỹ thuật sơ khai nhất là GNN sử dụng thuật toán Hungary.*

Xử lý ảnh và lọc

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

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

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

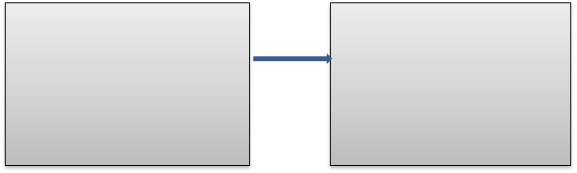


Xử lý ảnh và lọc

Hình : 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 vẫn tốc v tại thời điểm t+1*



Tương quan dữ liệu

Hình : 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 vẫn tốc v tại thời điểm t+1

#### Kỹ thuật tương quan dữ liệu ứng dụng trong theo dõi 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 xuất với *n* x *m* cặp giá trị xác xuất 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 xuất có được ta sử dụng các thuật toán như Hungary, …. Để xác định danh tính cho các đối tượng.

Đối tượng tại thời điểm t

Đối tượng tại thời điểm t+1

Hình : Mô tả các khả năng ghép đối tượng

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

Bảng : Xác suất ghép cặp đối tượng

|  |  |  |
| --- | --- | --- |
|  | **a** | **B** |
| **A** | 80% | 20% |
| **B** | 41% | 59% |

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

Bảng : Bảng kết quả ghép cặp

|  |  |  |
| --- | --- | --- |
|  | **a** | **B** |
| **A** | 80% | 20% |
| **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

#### 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 Cij >=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 x N với các bước thực hiện.

**Bước 1:** Tìm N với: N = MAX(n,m);

**Bước 2:** Biểu diễn ma trận N x N, với việc bổ sung N-n hàng hoặc N-m cột để tạo ma trận vuông với giá trị Cxy = MAX(Cij | 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 : 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 : 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 ví 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 : Kết quả phân công cho bài toán giao m công việc cho n công nhân

(*Chú thích: 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 bỏ 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 mông 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

# Đề xuất mô hình cho kỹ thuật phân đoạn chuyển động







## Mô hình chung của bài toán đề xuất

Dựa trên những nội dung đã nghiên cứu trong chương 2, chúng 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, chúng 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, chúng tôi đề xuất mô hình bài toán phân đoạn chuyển động của chúng tôi với các nội dung sau:

* Kỹ thuật phát hiện đối tượng chuyển động
* Kỹ thuật theo dõi đối tượng sử dụng median flow
* Tương quan dữ liệu sử dụng thuật toán Hungary



### Phát hiện đối tượng chuyển động

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

Vì vậy, bài toán của chúng tôi có cấu trúc tổng quát như sau:

Theo vết chuyển động

Phát hiện chuyển động

*(Sử dụng trừ nền trung bình)*

Xử lý các đo đạc

*(Median flow)*

Ước lượng và dự đoán

*(Sử dụng lọc Kalman)*

Xử lý tương quan

*(Sử dụng thuật toán Hungary)*

Track Handling



Hình : Mô hình bài toán đề xuất

### Sử dụng kỹ thuật trừ ảnh trung bình cho bài toán phát hiện đối tượng chuyển động

Em 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, … chúng 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.

Cập nhật nền

Ảnh hiện tại

Ảnh

Trung bình

Trừ ảnh

Làm trơn và khoanh vùng

đối tượng

*Thuật toán khoanh vùng nhanh*

*Chuỗi K ảnh nền*

Hình : Mô hình đề xuất của kỹ thuật trừ nền trung bình

### Áp dụng bộ lọc kết hợp cùng thuật toán Median flow cho bài toán theo vết các vùng chuyển động



#### Mô hình hệ thống



Theo vết sử dụng **Median flow**

*(x, y, w, h)*

Lọc sử dụng

**Kalman Filter**

Hình : Mô hình đề xuất sử dụng Median Flow và Kalman Filter cho bài toán phân đoạn chuyển động

#### Mô hình của bộ lọc Kalman

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

Trong đó:

* là trạng thái của hệ thống tại thời điểm k
* là kết quả đo đạt được của hệ thống tại thời điểm k
* ) là nhiễu hệ thống tại thời điểm k – 1
* là nhiễu đo đạc tại thời điểm k
* là ma trận chuyển trạng thái của hệ thống
* là ma trận của mô hình đo đạc

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

**A**

**B**

**C**

**A’**

**B’**

**C’**

**A**

**B**

**C**

Tính khoảng cách

ảnh tại thời điểm k

ảnh tại thời điểm k+1

Tính khoảng cách

Kết quả tương quan

**TƯƠNG QUAN**

Hình : Mô hình đề xuất sử dụng thuật toán Hungary cho bài toán Data Association

Trình tự thực hiện Data Association 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

# Thực nghiệm và kết luận



## Điều kiện thực nghiệm



### Môi trường

* Các môi trường thông thường như: *sân trường, đường xá,…* với các yếu tố sau:
  + Màu sắc nền và đối tượng có độ tương phản nhất định
  + 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.

### Thiết bị

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

### Cấu hình máy

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

*(Đó là điều kiện để cài đặt .NET Framework 3.5)*

### Khoảng cách thực nghiệm

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

## Thực nghiệm kỹ thuật phân đoạn video dựa trên các kỹ thuật lọc



### 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. chúng 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:

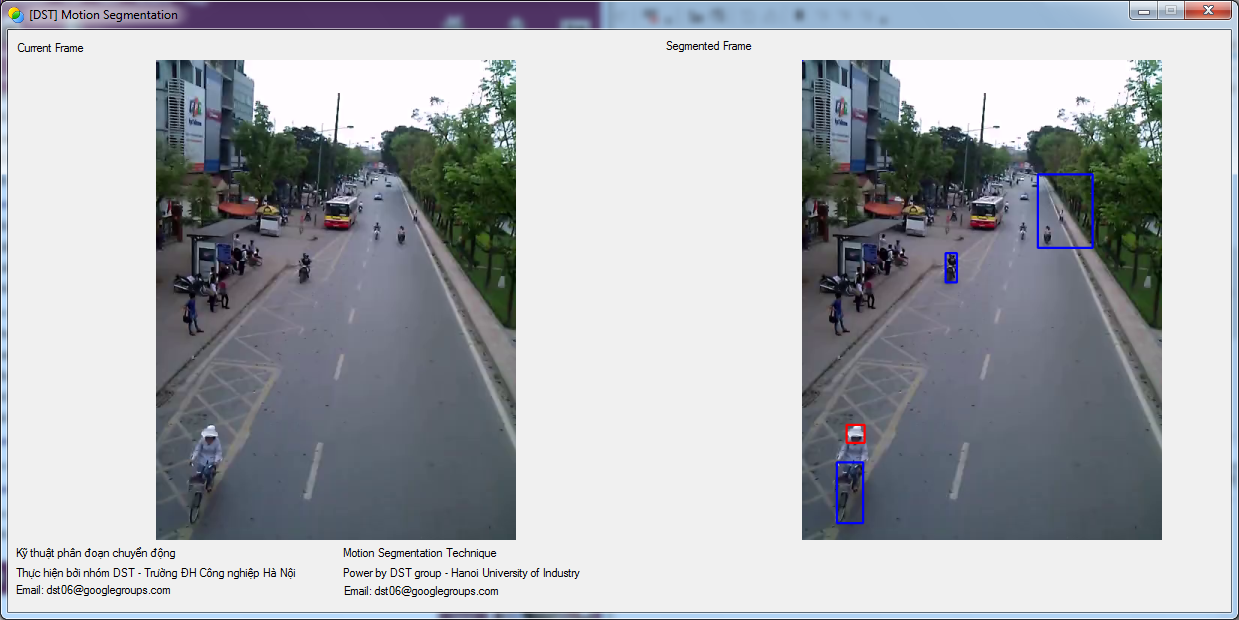
* Cầu vượt gần số 252 Hoàng Quốc Việt (gần trường Cao đẳng Du Lịch Hà Nội)
* Cầu vượt giao cắt giữa đường Xuân Thủy và đường Phạm Văn Đồng (gần trường ĐH Quốc gia Hà Nội)
* Sân Tenis – khu A trường ĐH Công nghiệp Hà Nội (quay từ tầng 4 – nhà A7)

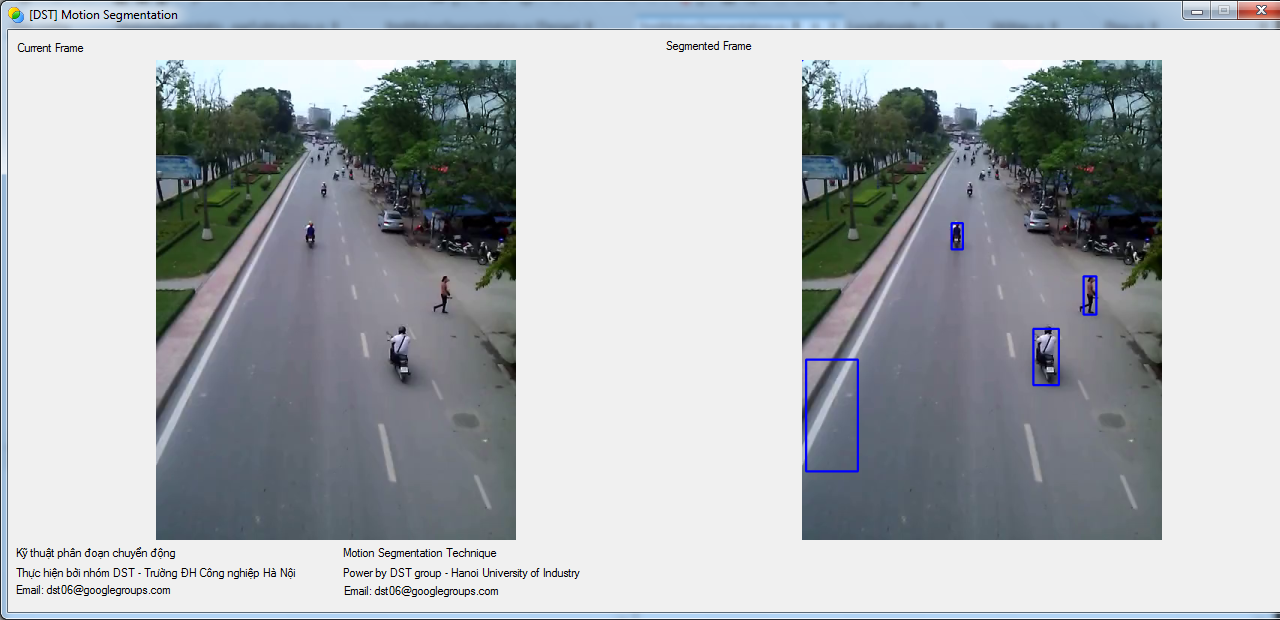
Các video thu được có chiều dài và kích thước như sau:

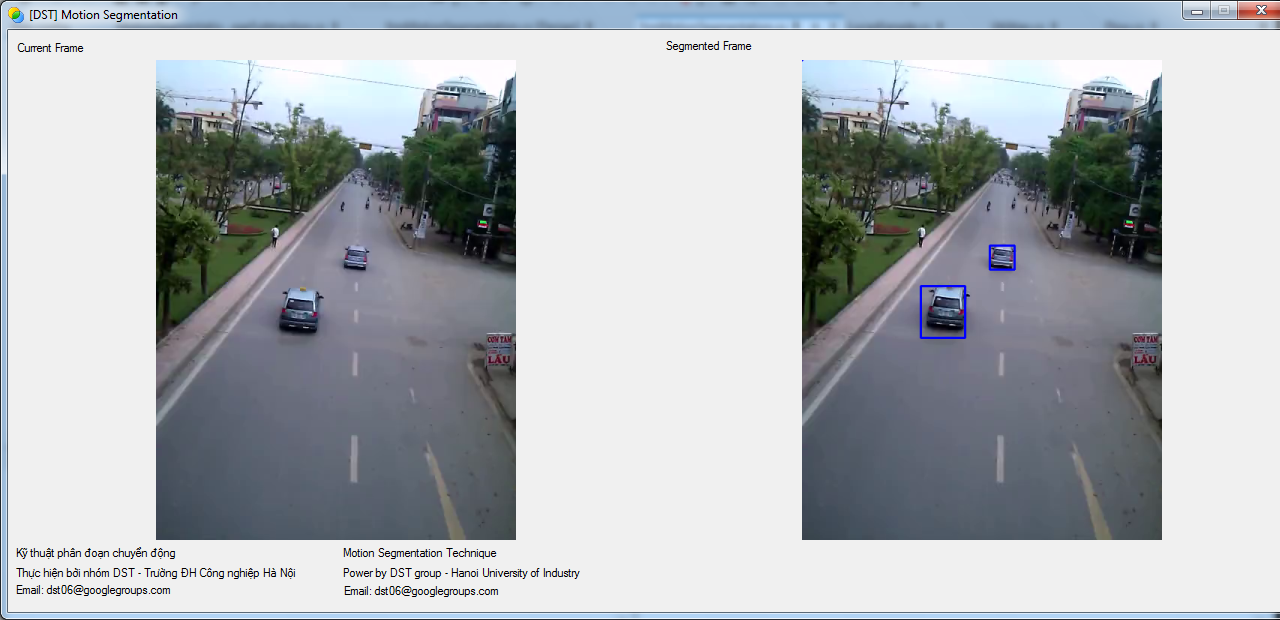
|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **TÊN VIDEO** | **THỜI GIAN** | **ĐỊA ĐIỂM** |
|  | test1.avi | 30s | 252 Hoàng Quốc Việt |
|  | test2.avi | 21s | 252 Hoàng Quốc Việt |
|  | test3.avi | 15s | 252 Hoàng Quốc Việt |
|  | test4.avi | 8s | 252 Hoàng Quốc Việt |
|  | test5.avi | 22s | Cầu vượt ĐH Quốc gia HN |
|  | test6.avi | 16s | Cầu vượt ĐH Quốc gia HN |
|  | test7.avi | 22s | Sân trường khu A - ĐHCNHN |
|  | test8.avi | 7s | Sân trường khu A - ĐHCNHN |

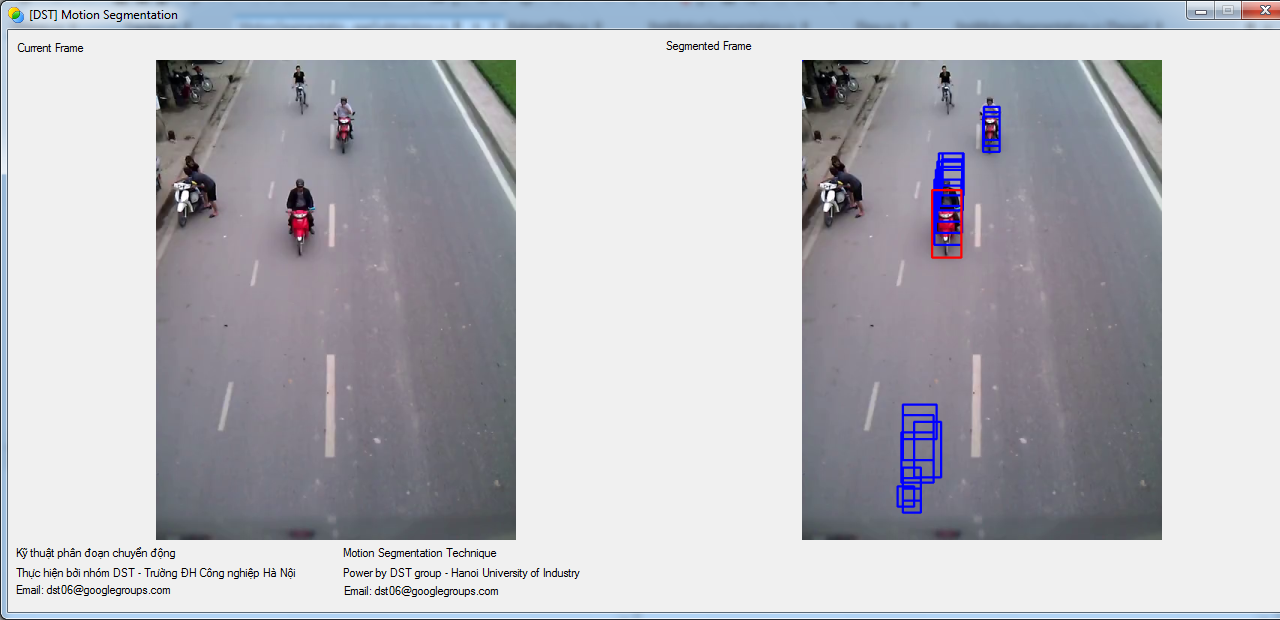
### Kết quả thực nghiệm

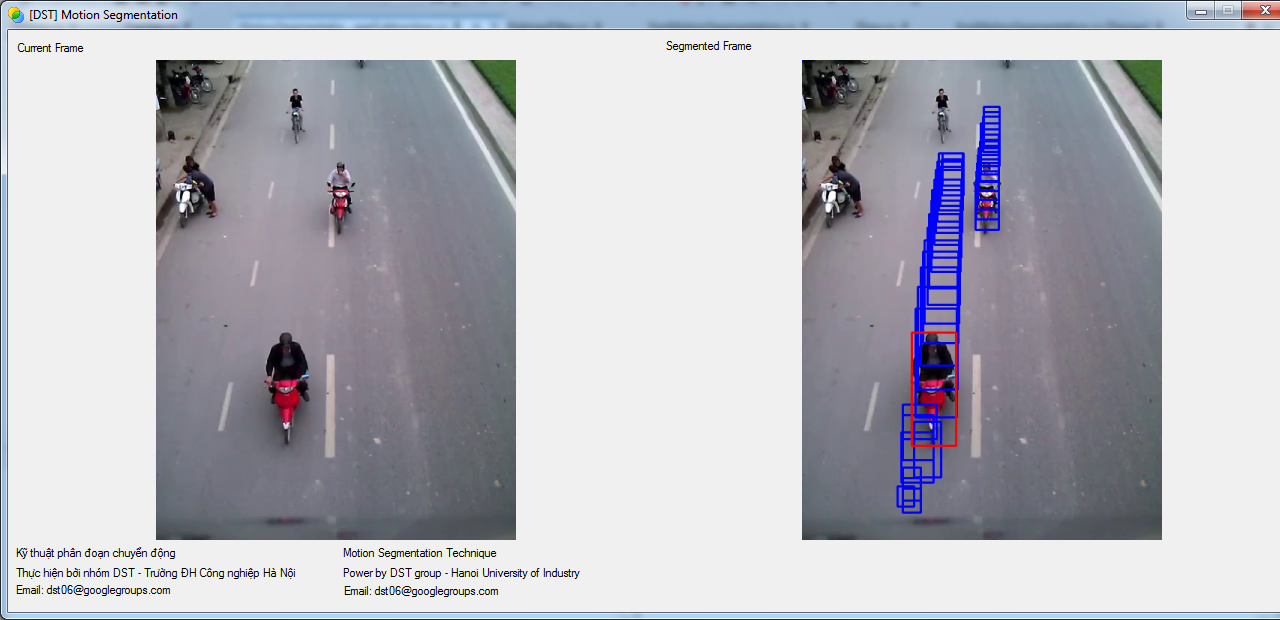
*Dưới đây là một số hình ảnh của kết quả thực nghiệm:*

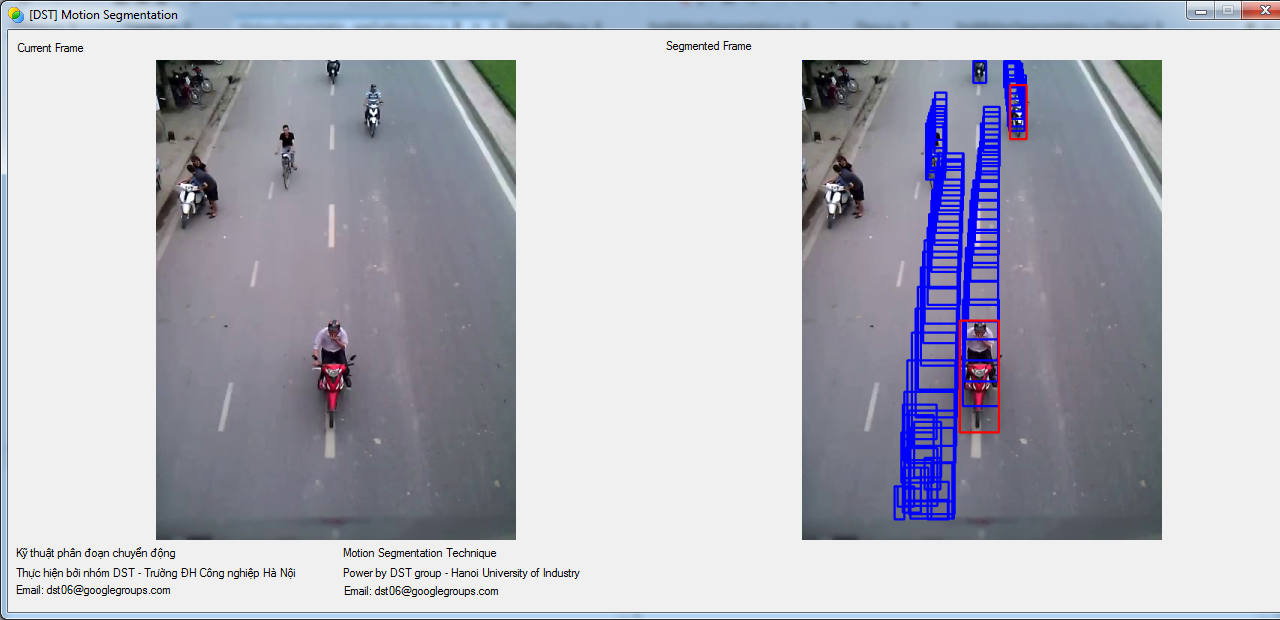












*Dưới đây là bảng kết quả thực nghiệm*

Table : Kết quả thực nghiệm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **TÊN VIDEO** | **THỜI GIAN** | **TỶ LỆ** | **FRAME RATE** |
|  | test1.avi | 30s | 67.3% | 12,8fps |
|  | test2.avi | 21s | 81.3% | 11,7fps |
|  | test3.avi | 15s | 83.3% | 9,2fps |
|  | test4.avi | 8s | 69.7% | 18,5fps |
|  | test5.avi | 22s | 76.6% | 13,2fps |
|  | test6.avi | 16s | 60.4% | 8,9fps |
|  | test7.avi | 22s | 31.5% | 11,2fps |
|  | test8.avi | 7s | 84.4% | 9,5fps |

## Kết luận

Các kỹ thuật phân đoạn chuyển động video rất đa dạng và phong phú, đã có một số kỹ thuật được áp dụng vào sản xuất công nghiệp – chứng tỏ được tính hiệu quả rõ rệt. Tuy nhiên, cách tiếp cận dựa trên việc áp dụng các bộ lọc để theo dõi và phân tích chuyển động của đối tượng thì có rất ít nghiên cứu và đánh giá.

Sau quá trình triển khai và thực nghiệm, chúng tôi rút ra những kết luận như sau:

* Việc áp dụng bộ lọc, yếu tố tiên quyết là thuật toán phát hiện đối tượng chuyển động phải chính xác.
* Việc áp dụng kỹ thuật theo dõi đơn giả thuyết (Single Hypothesis) chỉ phù hợp với bài toán có mật độ các đối tượng thưa. (thuật toán sử dụng là Hungary cho bài toán Data association)
* Áp dụng cải tiến của Optical flow là Median flow, 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 Optical flow hoạt động.
* Việc áp dụng Median flow 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.

## 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 Median flow
* Nghiên cứu lọc Particle và Unscented Kalman cho bài toán theo dõi đa đối tượng
* Nghiên cứu kỹ thuật theo dõi đa giả thuyết (Multiple Hypothesis Tracking) tối ưu bài toán data association

# TÀI LIỆU THAM KHẢO

1. <http://www.labbookpages.co.uk/software/imgProc/blobDetection.html> last access March 20th, 2012

1. <http://mpeg.chiariglione.org/standards/mpeg-7/mpeg-7.htm> last access March 10th, 2012
2. Jouni Hartikainen, Arno Solin, Simo Sarkka, “*Optimal filtering with Kalman filter and smoothers*”, Aalto University, 8-2011
3. D.J. Fleet, M.J. Black, Y. Yacoob and A.D. Jepson, “*Design and use of linear models for image motion analy-sis*”, Int. J. Comput. Vis. Vol.36, No.3, 2000, pp.171-193.
4. 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
5. M. Pawan Kumar, P.H.S. Torr, A.Zisserman, “*Learning Layered Motion Segmentations of Video*”, Oxford Brookes University, 2008
6. A. Colombari, A. Fusiello, V. Murino, “*Segmentation and Tracking of Multiple Video Object*”, Dipartimento di Informatica, University of Verona, 2008
7. Bing Han, William Roberts, Dapeng Wu, Jian Li, “*Motion-Segmentation-Based Change Detection*”, University of Florida Gainesville, 2007
8. 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
9. 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
10. A. G. Amitha Perera, Glen Brooks, Anthony Hoogs, Gianfranco Doretto, “*Moving Object Segmentation using Scene Understanding*”, One Research Circle, Niskayuna, New York 12309, 2006
11. Ren´e Vidal, “*Generalized Principal Component Analysis (GPCA)*”, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 27, NO. 12, PAGE 1945-1959, 2005
12. Conf. on NEURAL NETWORKS, Lisbon, Portugal, June 16-18, 2005 (pp20-25)
13. Vladimir Cherkassky & Yunqian Ma, “*Multiple Model Estimation: A New Formulation for Predictive Learning*”, University of Minnesota, 2005
14. 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.
15. Marco Benvenuti, Sara Colantonio, Maria Graz Ia Di Bono, Gabriele Pieri, “*Tracking of Moving Targets in Video Sequences*”, Proceedings of the 6th WSEAS Int.
16. Julier, S.J.; Uhlmann, J.K, [“*Unscented filtering and nonlinear estimation*”](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1271397&tag=1). Proceedings of the IEEE: 401–422, 2004
17. P. Smith, T. Drummond and R. Cipolla, “*Layered Mo tion Segmentation and Depth Ordering by Tracking Edges*”, IEEE Trans. on Pattern Analysis and M achine Intelligence, Vol. 26, No.4, 2004, pp.479-494.
18. Samuel S. Blackman, “*Multiple Hypothesis Tracking For Multiple Target Tracking*”, IEEE A&E Systems Magazine, 2004
19. Henk A.P. Blom and Edwin A. Bloe, “*Interacting Multiple Model Joint Probabilistic Data Association avoiding track coalescence*”, National Aerospace Laboratory NLR, Amsterdam, The Netherlands, 2002
20. David STweed, “*MotionSegmentationAcrossImageSequences*”, University of Bristol, 2001
21. Yaakov Bar-Shalom, X.-Rong Li, Thiagalingam Kiruba, “*Estimation with Applications to Tracking & Navigation*”, A Wiley-Interscience Publication, Canada, 2001
22. R. Szeliski, S. Avidan, P. Anandan, “*Layer extraction from multiple images containing reflections and transpar-ency”,* IEEE Conf. on Computer Vision and Pattern Recognition, Vol.I, June, 2000, pp.246-253.
23. 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.
24. Thomas Bak, “*Lecture Notes - Estimation and Sensor Information Fusion*”, Aalborge University, 2000
25. 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
26. T. Camus, “*Real-Time Quantized Optical Flow*”, Journal of Real-Time Imaging, Volume 3, pages 71-86, 1997.
27. M. Proesmans, L. Van Gool, E. Pauwels and A. Oosterlinck, “*Determination of optical flow and its Discontinuities using non-linear diffusion*”, In 3rd Eurpoean Conference on Computer Vision, ECCV'94, Volume 2, pages 295-304, 1994.
28. Matt M. Miller & Harold S. Stone and Ingemar J. Cox, “*Optimizing Murty’s Ranked Assignment Method*”, NEC Research Institute, Technical Report 95-08
29. Til Aach and Andr´ e Kaup and Rudolf Mester , “Statistical Model-Based Change Detection in Moving Video”, Institute of Imaging & Computer Vision, 1993
30. 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.
31. Til Aach, Andre Kaup, Rudolf Mester, “*Statical model-based change detection in moving video*”, Signal Processing 31, 165-180, 1993
32. Alessandro Verri, Sergio Uras & Enrico De Micheli, “*Motion Segmentation from Optical Flow*”, Dipartimento di Fisica dell' Universita di Genova, 1989
33. Donald B. Reid, “*An Algorithm for Tracking Multiple Targets*”, IEE Transaction on automatic control, 1979
34. 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
35. 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
36. R.E. Kalman; R.S. Bucy (1961). “*New results in linear filtering and prediction theory*”. Journal of Basic Engineering: 95–108, 1961
37. R.E. Kalman (1960). “*A New Approach to Linear Filtering and Prediction Problems”*. Journal of Basic Engineering: 35–45.
38. R.E. Kalman (1960). “*Contributions to the theory of optimal control*”, Bol. Soc. Mat. Mexicana: 102–119

1. 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) = xo + 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 [↑](#footnote-ref-1)
2. Việc coi nhiễu đo đạc và nhiễu hệ thống không có sẽ giúp ta có thể tuyến tính hoá hệ thống, phục vụ cho việc áp dụng cải tiến của Kalman Filter là EKF được thực hiện. Tuy nhiên, đây cũng là một hạn chế của EKF (được trình bày trong phần c. Ưu nhược điểm của EKF) [↑](#footnote-ref-2)
3. Để tính được  là dòng thứ i (hoặc cột thứ i), ta sử dụng phép phân tích ma trận Cholesky để lấy căn của ma trận  [↑](#footnote-ref-3)