**LỜI CÁM ƠN**

Em xin chân thành cảm ơn Khoa Công Nghệ Thông Tin, trường Đại Học Khoa Học Tự Nhiên, Tp.HCM đã tạo điều kiện tốt cho chúng em thực hiện đồ án thạc sĩ này. Em cảm ơn thầy Trần Thái Sơn là người đã tận tình hướng dẫn, chỉ bảo em trong suốt thời gian thực hiện đồ án.

Em cũng xin gởi lời cảm ơn sâu sắc đến quí thầy cô trong Khoa đã tận tình giảng dạy, trang bị cho chúng em những kiến thức quí báu trong những năm học vừa qua.

Em xin gởi lòng biết ơn sâu sắc đến ba, mẹ, các anh chị và bạn bè đã ủng hộ, giúp đỡ và động viên chúng em trong những lúc khó khăn cũng như trong suốt thời gian học tập, nghiên cứu và thực hiện đồ án.

Mặc dù đã cố gắng hoàn thành luận văn trong phạm vi và khả năng cho phép, nhưng chắc chắn sẽ không tránh khỏi những thiếu sót, kính mong sự cảm thông và tận tình chỉ bảo của quý thầy cô và các bạn.

TP.HCM, 11/2015

Sinh viên thực hiện

Nguyễn Thành Công

**MỤC LỤC**

[CHƯƠNG 1 GIỚI THIỆU 1](#_Toc436839575)

[1.1 Động lực nghiên cứu 1](#_Toc436839576)

[1.2 Phát biểu bài toán 1](#_Toc436839577)

[1.4 Các thách thức của bài toán 2](#_Toc436839578)

[1.5 Bố cục tổ chức chương 2](#_Toc436839579)

[CHƯƠNG 2 CÁC HƯỚNG NGHIÊN CỨU LIÊN QUAN VÀ HƯỚNG TIẾP CẬN 4](#_Toc436839580)

[2.1 Các hướng nghiên cứu liên quan đến việc giải quyết bài toán 4](#_Toc436839581)

[2.2 Hướng tiếp cận bài toán của đồ án 5](#_Toc436839582)

[2.2.1 Tóm tắt video 6](#_Toc436839583)

[2.2.2 Rút trích keyframe đại diện cho shot 6](#_Toc436839584)

[2.2.3 Truy tìm frame ảnh trong video 7](#_Toc436839585)

[CHƯƠNG 3 TÓM TẮT VIDEO VÀ RÚT TRÍCH KEYFRAME 8](#_Toc436839586)

[3.1 Tóm tắt video 8](#_Toc436839587)

[3.1.1 Tách shot 8](#_Toc436839588)

[3.1.2 Ghép shot 8](#_Toc436839589)

[3.2 Rút trích keyframe đại diện cho shot 8](#_Toc436839590)

[3.2.1 Giới thiệu đặc trưng MPEG-7 8](#_Toc436839591)

[3.2.1.1 Bố cục màu (Color Layout Desriptor - CLD) 8](#_Toc436839592)

[3.2.1.2 Cấu trúc màu (Color Structure Descriptor - CSD) 9](#_Toc436839593)

[3.2.1.3 Lược đồ biên cạnh (Edge Histogram Descriptor - EHD) 10](#_Toc436839594)

[3.2.1.4 Vân đồng dạng (Homogeneous Texture Descriptor - HTD) 11](#_Toc436839595)

[3.2.2 Chọn lọc keyframe 12](#_Toc436839596)

[3.2.2.1 Rút trích đặc trưng 12](#_Toc436839597)

[3.2.2.2 Thuật toán phát hiện điểm đường cong cao 14](#_Toc436839598)

[CHƯƠNG 4 TRUY TÌM FRAME ẢNH TRONG VIDEO 19](#_Toc436839599)

[4.1 Rút trích đặc trưng keyframe 19](#_Toc436839600)

[4.2 Truy vấn video 19](#_Toc436839601)

[4.3 Truy vấn shot 19](#_Toc436839602)

[CHƯƠNG 5 THỰC NGHIỆM 22](#_Toc436839603)

[5.1 Giới thiệu tập dữ liệu thực nghiệm 22](#_Toc436839604)

[5.2 Kết quả truy vấn video 22](#_Toc436839605)

[CHƯƠNG 6 KẾT LUẬN 25](#_Toc436839606)

[THAM KHẢO 27](#_Toc436839607)

**DANH MỤC CÁC BẢNG**

[Bảng 3.1: Bảng chi tiết các giá trị khoảng cách, góc mở tại mỗi frame trong shot. 17](#_Toc436862388)

[Bảng 5.1: Số lượng mẫu video trong tập dữ liệu 22](#_Toc436862389)

[Bảng 5.2: Số lượng shot và keyframe rút trích được từ các mẫu video 22](#_Toc436862390)

[Bảng 5.3: Kết quả đánh giá phương pháp truy vấn video của đồ án 23](#_Toc436862391)

**DANH MỤC CÁC HÌNH VẼ**

[Hình 3.1: Cấu trúc nguyên tố cho các ảnh với các độ phân giải khác nhau 10](#_Toc436864747)

[Hình 3.2: Cách rút trích lược đồ biên cạnh 11](#_Toc436864748)

[Hình 3.3: Bố cục tần số trong quá trình rút trích đặc trưng vân 12](#_Toc436864749)

[Hình 3.4: Biểu đồ phát hiện điểm đường cong cao 16](#_Toc436864750)

[Hình 3.5: Ví dụ các frame có trong một shot 16](#_Toc436864751)

[Hình 3.6: Đồ thị thể hiện khoảng cách giữa các frame liên tiếp nhau. 18](#_Toc436864752)

[Hình 3.7: Các frame được xem là keyframe và rút trích từ shot 18](#_Toc436864754)

[Hình 4.1: Hình mô tả phương pháp truy vấn dùng Feature Line 20](#_Toc436864755)

# CHƯƠNG 1 GIỚI THIỆU

*Nội dung chương giới thiệu tổng quan về đề tài, mục tiêu cụ thể của đề tài, cũng như các nội dung chính trong khoá luận.*

## Động lực nghiên cứu

Ngày nay, nhu cầu xử lý thông tin ảnh số và video càng ngày càng gia tăng do lượng thông tin lưu trữ trong đó có thể được khai thác và sử dụng cho nhiều mục đích khác nhau. Một trong số các ứng dụng đó là truy vấn ảnh và video trong cơ sở dữ liệu. Truy vấn ảnh và video hiện nay vẫn còn đang là một mảng mới đang được nghiên cứu, đòi hỏi có sự kết hợp giữa các mảng tri thức khác nhau như thống kê, thị giác máy tính, xử lý tín hiệu số,… để giải quyết.

Bài toán truy vấn có thể được chia thành nhiều bài toán con. Một trong số các bài toán con đó là bài toán truy tìm frame ảnh trong cơ sở dữ liệu video lớn. Bằng việc áp dụng các thuật toán xử lý ảnh, video và rút trích đặc, ta có thể truy tìm ra các video có chứa frame ảnh gần giống với ảnh mà người dùng đưa vào. Giải quyết được bài toán này sẽ có lợi ích rất lớn khi áp dụng vào thực tế như tìm kiếm nhanh các phân cảnh quan trọng trong các bộ phim, tìm ra video quay được từ các camera quan sát cảnh vi phạm giao thông hoặc hành động phạm tội, v.v…

## Phát biểu bài toán

Mục đích của đồ án: Sử dụng các đặc trưng cấp thấp và cấp cao truy tìm nhanh video trong cơ sở dữ liệu lớn dựa vào ảnh đầu vào của người dùng.

**Đầu vào:** Ảnh truy vấn *I*, cơ sở dữ liệu video lớn *V*.

**Đầu ra:** Tập các cặp giá trị (S, *t*) với *S* là video có chứa frame tương đồng với ảnh truy vấn *I* tại thời điểm *t*.

1. **Giới hạn bài toán**

Các video được đưa vào sử dụng trong đồ án được thu thập từ nhiều nguồn khác nhau (DVD, Internet, v.v…). Chủ đề chính của các video là những bộ phim đã được công chiếu hoặc cũng có thể là các video thể thao như bóng đá, tennis.

Đồ án tập trung vào việc truy tìm được frame ảnh tương đồng với ảnh đầu vào trong một video thuộc cơ sở dữ liệu video lớn. Do giới hạn về mặt lưu trữ và thời gian chuẩn bị nên đồ án chỉ xét trên cơ sở dữ liệu của 100 video.

## Các thách thức của bài toán

Bài toán bao gồm các thách thức sau:

Đồ án vẫn chưa tìm được phương pháp nào xử lý các video có đính kèm phụ đề (subtitle). Do đó hầu hết các video thu thập được đều không có phụ đề để tránh gây nhiễu kết quả truy tìm. Điều này giới hạn tầm xử lý của hệ thống.

Tốc độ xử lý khi tóm tắt video và rút trích các keyframe rất lâu (thường nằm trong khoảng từ 2 đến 3 giờ cho một video). Đồ án đã nghiên cứu qua một số phương pháp rút trích keyframe có tốc độ nhanh hơn nhưng thường phải đánh đổi với mức độ tổng quát hóa video của các keyframe.

## Bố cục tổ chức chương

**Có 3 bài toán chính sau cần được giải quyết**

**1.5.1 Phân đoạn video**

* Tách shot
* Ghép shot

**1.5.2 Rút trích keyframe đại diện cho shot**

* Rút trích đặc trưng MPEG-7
* Chọn lọc keyframe

**1.5.3 Truy tìm frame ảnh trong video**

* Rút trích đặc trưng keyframe.
* Xác định video có khả năng chứa ảnh đầu vào cao nhất.
* Xác định shot có chứa keyframe tương đồng với ảnh đầu vào.

# CHƯƠNG 2 CÁC HƯỚNG NGHIÊN CỨU LIÊN QUAN VÀ HƯỚNG TIẾP CẬN

*Nội dung chương nói về các hướng nghiên cứu liên quan đến việc giải quyết bài toán, và hướng tiếp cận bài toán của đồ án*

## Các hướng nghiên cứu liên quan đến việc giải quyết bài toán

Giải quyết bài toán phân đoạn video thành các shot là bước đầu cơ bản để thực hiện các thao tác khác như đánh chỉ mục video hoặc truy vấn video dựa vào nội dung, v.v… Tác giả Liu [1] đã đề xuất một giá trị tỉ lệ báo động sai để phân đoạn video. Để tìm ra những đoạn cắt cảnh trong video, một phương pháp xác định ngưỡng dựa trên tỉ lệ báo động sai phi tham số được phát triển. Trong khi đó, Fang [2] lại đề xuất phương pháp kết hợp logic mờ vào các đặc trưng kết hợp để phát hiện đúng các khung hình chuyển cảnh. Phương pháp này có thể phát hiện ra các cảnh chuyển tiếp từ từ hoặc đột ngột. Các đặc trưng khác nhau được sử dụng cho hướng tiếp cận sử dụng logic mờ để phân đoạn video theo chiều thời gian. Một phương pháp khác cũng đang được nghiên cứu để tách shot là phương pháp sử dụng chuyển động của camera để đo lường thị sai của các vật hiển thị trong cảnh của Donate [3]. Ý tưởng được đưa ra mô hình hóa cấu trúc 3 chiều của cảnh và sau đó theo vết các đặc trưng qua các khung hình để xác định ranh giới của shot. Ngoài ra còn có các phương pháp tách shot video khác như sử dụng các đặc trưng cấp cao (moment [4], contour [5], v.v…) và các mô hình thống kê (Bayesian [6], Hidden Markov Model [7], v.v…) để nâng cao khả năng tách shot chính xác.

Bài toán tiếp theo cần được giải quyết là bài toán rút trích key frame từ các shot để đại diện cho một video, giảm chi phí lưu trữ mà vẫn giữ được những thông tin quan trọng trong video. Do đó việc giải quyết bài toán này đóng vai trò chủ chốt trong việc xây dựng hầu hết các hệ thống xử lý video số. Đã có một số phương pháp được đề xuất để giải quyết bài toán này: Ajay [8] cho rằng càng nhiều thông tin chuyển động thì càng cần nhiều key frame để tổng quát hóa. Do đó, tác giả xác định key frame bằng cách chia nhỏ các shot theo thông tin chuyển động và chọn các frame nằm giữa các phân đoạn con đó. Nắm bắt được nhu cầu sử dụng ảnh hoặc mẫu video nhỏ để truy vấn video trong cơ sở dữ liệu, tác giả Li Zhao, Wei Qi [9] đã tìm ra phương pháp coi các frame trong một shot video như các điểm trong không gian đặc trưng mà trong đó các điểm được nối với nhau bằng các đường cong. Các điểm đứt gãy (gọi là breakpoint) trên quỹ đạo đặc trưng là các key frame và đường thẳng đi qua các điểm breakpoint là biểu diễn của shot video đó. Tác giả Xianglin [10] lại đề xuất phương pháp xác định key frame dựa vào nhóm chiếm ưu thế và có thể xác định được số lượng key frames dựa vào độ phức tạp của video với chi phí tính toán thấp. Ngoài ra vẫn có một số phương pháp khác như sử dụng độ đo Hausdorff cải tiến để tính độ phân tán trực tiếp giữa 2 lược đồ của 2 frame cạnh nhau [11] hoặc các phương pháp khác dựa vào nội dung của video và đặc trưng chuyển động, v.v… Như vậy, ta xây dựng được một cơ sở dữ liệu các key frame cho mỗi video để lưu trữ và thực hiện các thao tác rút trích đặc trưng và đối sánh như đối với ảnh.

Ta có thể sử dụng các loại đặc trưng cấp thấp (màu, vân, dáng) hoặc các đặc trưng cấp cao (SIFT, SURF, v.v…) cùng mô hình giỏ đặc trưng và sử dụng các độ đo đối sánh thông dụng như Minkowski, Euclidean, v.v… để đối sánh ảnh đầu vào với các key frame để xác định được video có chứa thông tin tương đồng với ảnh đầu vào.

## Hướng tiếp cận bài toán của đồ án

Đồ án sử dụng các đặc trưng màu, vân, dáng thuộc nhóm đặc trưng MPEG-7 (Motion Picture Expert Group – 7) để rút trích keyframe và dùng độ đo Euclidean để đối sánh các vector đặc trưng. Phương pháp đồ án sử dụng để giải quyết vấn đề có thể được chia thành nhiều bài toán con. Để giải quyết các bài toán con này, đồ án đã thực hiện như sau:

**Phân đoạn video**

* Tách shot
* Ghép shot dựa vào độ dài của shot và số khung hình/giây của video

**Rút trích keyframe đại diện cho shot**

* Đặc trưng rút trích: bố cục màu, cấu trúc màu, lược đồ biên cạnh, vân đồng dạng [12]
* Phương pháp chọn lọc keyframe: High Curvature Point [12]
* Độ đo: Euclidean.

**Truy tìm frame ảnh trong video**

* Đặc trưng rút trích: bố cục màu, lược đồ biên cạnh [14]
* Phương pháp: dựa trên Nearest Feature Line [9]
* Độ đo: Euclidean.

### Tóm tắt video

Một video bất kỳ cần được tóm tắt và phân đoạn thành các shot video để giảm bớt độ phức tạp tính toán khi đưa vào rút trích keyframe. Một shot là một đơn vị vật lý của dòng video, gồm các chuỗi một hay nhiều khung hình liên tiếp, không thể chia nhỏ hơn, ứng với một thao tác camera đơn. Các phương pháp phát hiện ranh giới shot thường tập trung vào việc nhận biết sự không tương đồng về đặc trưng thị giác do sự chuyển đổi cảnh. Đồ án sử dụng phương pháp đặt ngưỡng đáp ứng cục bộ trong một cửa sổ duyệt để phát hiện ra các điểm xảy ra sự chuyển đổi cảnh và từ đó xác định ranh giới của các shot. Sau khi phân đoạn video thành các shot, đồ án sẽ thực hiện ghép hai shot liên tiếp nhau nếu tồn tại shot có số lượng frame quá ít để đảm bảo bước rút trích keyframe luôn thực hiện được đúng.

### Rút trích keyframe đại diện cho shot

Rút trích keyframe đại diện toàn bộ video là một bước quan trọng trong tất cả các hệ thống truy vấn video. Các keyframe là các frame có chứa nội dung và thông tin nổi bật trong video. Việc rút trích keyframe sẽ giúp cải thiện hiệu quả của một hệ thống truy vấn. Để giải quyết bài toán này, đồ án đã thực hiện rút trích keyframe trên các shot video bằng cách rút trích các đặc trưng màu và vân trên tất cả các frame trong shot, tính khoảng cách giữa hai frame liên tiếp nhau và đặt ngưỡng để phát hiện ra các frame mà tại đó có sự biến động lớn về mặt đặc trưng thị giác, cụ thể hơn đó là frame có khoảng cách với frame tiếp theo lớn hơn một ngưỡng cho trước.

### Truy tìm frame ảnh trong video

Tại thời điểm này, mỗi video đã được đại diện bằng một tập các keyframe và mỗi keyframe đã được rút trích đặc trưng màu và vân để lưu trữ. Khi người dùng đưa vào một ảnh bất kỳ, đồ án sẽ thực hiện rút trích đặc trưng tương ứng và trả ra kết quả truy vấn video có khả năng chứa frame tương đồng với ảnh đầu vào bằng độ đo Euclidean. Với mỗi kết quả video được trả về, đồ án cũng trả ra kết quả truy vấn shot cùng với khung thời gian mà ảnh đầu vào có khả năng thuộc về sau khi đã áp dụng phương pháp Nearest Feature Line.

# CHƯƠNG 3 TÓM TẮT VIDEO VÀ RÚT TRÍCH KEYFRAME

*Nội dung chương 3 trình bày mô hình rút trích keyframe của một video bất kỳ trong cơ sở dữ liệu*

## Tóm tắt video

### Tách shot

Đồ án sử dụng phương pháp của Dugad để phát hiện sự chuyển cảnh trong video. Phương pháp sử dụng một cửa sổ duyệt một dãy các khung hình liên tiếp cùng với độ đo trung bình bình phương lỗi để tính ngưỡng và xác định khung hình mà tại đó là ranh giới của hai cảnh khác nhau. Các khung hình liên tiếp nhau được xét độ biến động cục bộ bằng các phương pháp xử lý ảnh như trừ nền, làm trơn và chia lưới để phân loại vào các nhóm khác nhau: khung hình bắt buộc, khung hình tiềm năng, khung hình dư thừa loại 1 và khung hình dư thừa loại 2. Như vậy, video đã được chia thành các shot với ranh giới là các khung hình bắt buộc.

### Ghép shot

Kết quả của quá trình tách shot đôi khi sẽ tồn tại những shot có số lượng frame quá ít để có thể sử dụng cho các bước tiếp theo. Do đó, đồ án đã thực hiện ghép shot. Gọi *Si* là shot hiện tại đang xét có *n* frame của video *V* với số khung hình/giây là *fpsV*. Nếu *n* < *fpsV*  thì shot *Si*sẽ được nối liền với shot *Si+1*.

## Rút trích keyframe đại diện cho shot

### Giới thiệu đặc trưng MPEG-7

#### Bố cục màu (Color Layout Desriptor - CLD)

Đặc trưng bố cục màu được thiết kế để bắt được sự phân bố về mặt không gian của màu sắc trong một ảnh hoặc một vùng có hình dáng ngẫu nhiên. Phân bố trong không gian của màu sắc có vai trò quan trọng trong các bộ mô tả để sử dụng cho việc truy vấn ảnh dựa vào phác thảo hoặc nội dung. Bộ mô tả bố cục màu là một bộ mô tả sử dụng các màu đại diện trong các ô 8×8 được sử dụng trong phép biến đổi cosine rời rạc (Discrete Cosine Transform – DCT) và mã hóa tham số kết quả.

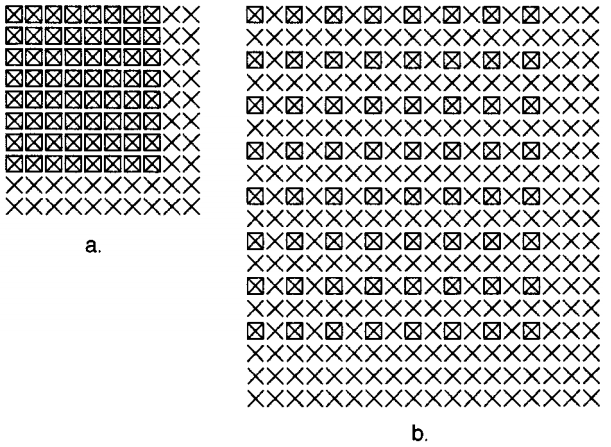
Quá trình rút trích đặc trưng diễn ra trong 2 quá trình: lựa chọn màu sắc theo ô và biến đổi cosine rời rạc có định lượng. Một ảnh khi đưa vào sẽ được phân thành 64 block với kích thước 8×8 và trung bình màu tại mỗi block được tính. Các block được biến đổi thành một chuỗi các tham số là kết quả của phép biến đổi cosine rời rạc 8×8. Một số tham số tần số thấp được sử dụng sử dụng phương pháp quét zigzag và định lượng để tạo ra vector đặc trưng. Bộ mô tả sử dụng không gian màu YCrCb. Gọi *nY*, *nCr* và *nCb* lần lượt là số lượng bin cho kênh màu Y, Cr và Cb, ta được histogram *hCLD* có dạng:

#### Cấu trúc màu (Color Structure Descriptor - CSD)

Bộ mô tả cấu trúc màu biểu diễn cấu trúc màu cục bộ trong một ảnh sử dụng cấu trúc nguyên tố (structure element). Với mỗi khu vực mà cấu trúc nguyên tố này quét qua trên ảnh, các màu nằm trong cấu trúc nguyên tố được đếm số lần xuất hiện. Gọi *c0*, *c1*, *c2*, … , *cM-1*là các màu định lượng. Một lược đồ màu *hCSD(m)* (*m*=0,1,…,M-1) được xây dựng. Mỗi giá trị trong lược đồ đại diện cho số lượng vị trí mà tại đó màu *cm* nằm trong cấu trúc nguyên tố. Bộ mô tả này sử dụng không gian màu HMMD để biểu diễn. Để xây dựng bộ mô tả cấu trúc màu, ta cần sử dụng cấu trúc nguyên tố có kích thước 8×8. Mặc dù tổng số lượng mẫu được giữ nguyên ở mức 64, kích thước của cấu trúc nguyên tố được thay đổi tỷ lệ với kích thước ảnh. Một luật đơn giản được dùng để xác định kích thước của cấu trúc nguyên tố được định nghĩa như sau:

Trong đó, W, H lần lượt là chiều dài và rộng của ảnh. E × E là kích thước của cấu trúc nguyên tố. K là tham số phụ (sub-sampling).

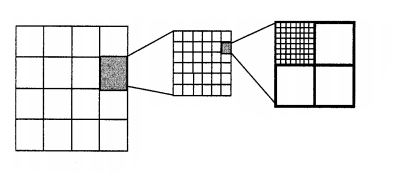
Khi *hCSD(m)* được xây dựng, các giá trị trong *hCSD(m)* được chuẩn hóa bởi tổng số vị trí xuất hiện của nguyên tố cấu trúc và có miền giá trị trong đoạn [0.0, 1.0]. Các giá trị này sau đó được định lượng phi tuyến tính vào 8 bits/bin. Như vậy ta có lược đồ màu: .



Hình 3.1: Cấu trúc nguyên tố cho các ảnh với các độ phân giải khác nhau  
*(a) 320×240 và (b) 640×480 (Nguồn [15])*

#### Lược đồ biên cạnh (Edge Histogram Descriptor - EHD)

Lược đồ biên cạnh có thể bắt được tần suất và hướng phân bố của sự thay đổi độ sáng trong ảnh. Sự phân bố của biên cạnh là một tín hiệu vân hữu dụng cho việc đối sánh 2 ảnh mặc dù vân cơ bản không cùng chung dạng với nhau. Một ảnh khi đưa vào được chia thành các ảnh con có kích thước 4×4 và các lược đồ biên cạnh cục bộ được tính cho mỗi ảnh con đó. Biên cạnh được nhóm vào 5 loại: ngang, dọc, nghiêng 45o, nghiêng 135o và không có hướng xác định. Do đó, mỗi lược đồ cục bộ được có 5 bin tương ứng với 5 nhóm trên. Các bin này được định lượng không đồng đều với 3bits/bin cho ra một bộ mô tả với 240 bits. Như vậy lược đồ biên có tất cả 80 bin.

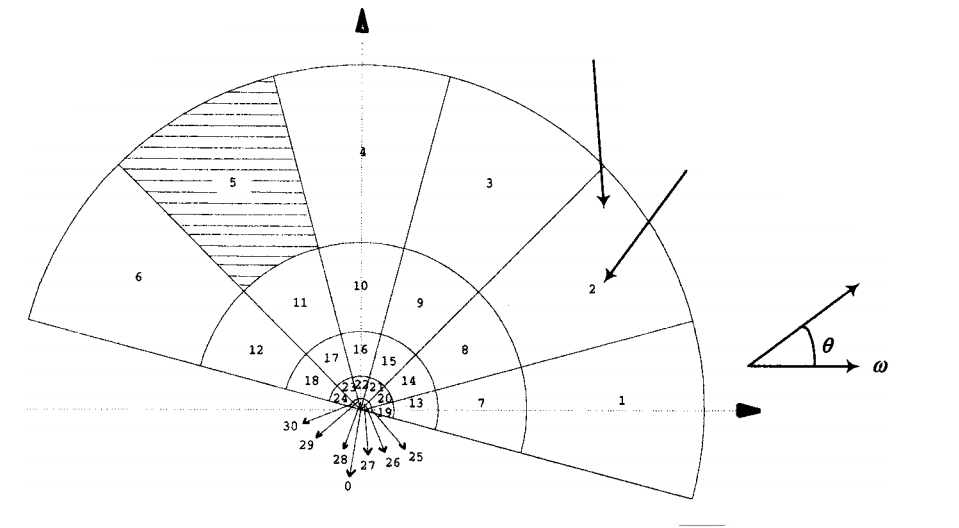


Hình 3.2: Cách rút trích lược đồ biên cạnh (Nguồn [15])

Để tính lược đồ biên cạnh, mỗi ảnh trong số 16 ảnh con được chia nhỏ thêm thành các khối ảnh. Kích thước mỗi khối ảnh tỷ lệ với kích thước ảnh gốc và có giá trị là lũy thừa của 2. Số lượng khối ảnh mỗi ảnh con được giữ nuyên và độc lập với số chiều của ảnh gốc bằng cách điều chỉnh tỷ lệ tương ứng. Một thuật toán phát hiện biên cạnh được áp dụng cho các phân vùng có kích thước 2×2 của mỗi khối ảnh. Cường độ độ xám của các phân vùng 2×2 được tính bằng cách lấy trung cường độ độ xám của các điểm ảnh có trong đó. Cường độ của các khối ảnh khi vượt qua một ngưỡng cho trước sẽ đóng góp vào việc tính lược đồ và được xem như một “khối biên cạnh”. Mỗi “khối biên cạnh” được đưa vào bin tương ứng trong lược đồ và các giá trị bin này được chuẩn hóa về đoạn [0,1]. Như vậy, lược đồ biên cạnh sẽ có dạng: .

#### Vân đồng dạng (Homogeneous Texture Descriptor - HTD)

Bộ mô tả vân đồng dạng cung cấp một đặc tính định lượng của vân dùng cho việc đối sánh độ tương đồng giữa 2 ảnh. Bộ mô tả này được tính bằng việc lọc ảnh bằng một bộ các bộ lọc nhạy cảm với tỷ lệ và hướng và sau đó tính trung bình và độ lệch chuẩn của đầu ra của các bộ lọc trong miền tần số. Đây là một bộ mô tả đặc trưng có tính ổn định, hiệu quả và dễ tính toán.



Hình 3.3: Bố cục tần số trong quá trình rút trích đặc trưng vân (Nguồn [15])

Để xây dựng bộ mô tả này, miền tần số được phân vùng thành 30 kênh được chia đều theo hướng góc (ngưỡng 30o) và các octave được chia đều theo hướng xuyên tâm (5 octave). Mỗi kênh được lọc bởi một tập các hàm Gabor 2-D. Công thức biến đổi Fourier của hàm Gabor 2-D trong hệ tọa độ cực được viết như sau:

Vector đặc trưng của bộ mô tả vân đồng dạng có dạng như sau:

Trong đó, hai giá trị đầu tiên lần lượt là trung bình và độ lệch chuẩn của vân ảnh. Các giá trị *ci* và *di*lần lượt là năng lượng và độ lệch năng lượng tại kênh tần số thứ i.

### Chọn lọc keyframe

#### Rút trích đặc trưng

Để giải quyết bài toán chọn lọc keyframe, đồ án đã chọn phương pháp được đề xuất bởi G. Ciocca [12]. Theo như G. Ciocca, đa số các thuật toán chọn lọc keyframe thường chỉ sử dụng một bộ mô tả đặc trưng để thể hiện sự thay đổi nội dung hình ảnh qua các frame trong cùng một shot. Do đó, tác giả đã đề xuất sử dụng kết hợp đặc trưng màu, vân và thống kê wavelet. Các đặc trưng được lựa chọn bởi ba tiêu chí: tương đồng tri giác (độ dị biệt giữa 2 ảnh chỉ lớn khi 2 ảnh thật sự khác nhau), hiệu quả (đặc trưng có thể được tính nhanh) và “kinh tế” (số chiều của đặc trưng không làm ảnh hưởng tới hiệu quả).

Khác với G. Ciocca, đồ án đã sử dụng các đặc trưng màu và vân thuộc bộ đặc trưng MPEG-7 mà cụ thể là: lược đồ cấu trúc màu, lược đồ biên cạnh và đặc trưng vân đồng nhất. Lược đồ màu nói chung đều bền vững đối với những thay đổi nhỏ từ góc nhìn của camera nên thường được sử dụng nhiều trong các công trình nghiên cứu về bài toán truy vấn và nhận dạng đối tượng. Lược đồ biên cạnh phù hợp với việc thể hiện các ảnh tự nhiên và được xây dựng với mục đích đối sánh hai ảnh với nhau. Đặc trưng vân đồng dạng có thể được tính nhanh khi tính trên miền tần số và có thể sử dụng chung với hai đặc trưng trên để mở rộng hiệu quả truy vấn. Gọi *It* là frame đang xét, ta thực hiện rút trích ba đặc trưng vừa nêu trên và tính độ dị biệt với ba đặc trưng tương ứng của frame *It+1*. Độ dị biệt đối với từng đặc trưng được tính như sau:

* Bố cục màu:
* Lược đồ biên cạnh:
* Vân đồng dạng:   
  với *w(j)* là trọng số và *a(j)* là tham số chuẩn hóa (có thể được tính từ độ lệch chuẩn của các giá trị thứ *j* trong vector đặc trưng vân đồng dạng trong một cơ sở dữ liệu tham chiếu nào đó).

Với ba giá trị khoảng cách tính được *dC*, *dE* và *dH*, ta có thể kết hợp thành một giá trị khoảng cách cuối cùng *dCEH* có dạng như sau:

Việc kết hợp các đặc trưng sẽ làm tăng hiệu quả của quá trình chọn lọc keyframe do các đặc trưng có thể bù đắp khuyết điểm của nhau và làm tăng độ dị biệt giữa hai ảnh đang xét. Vấn đề cần giải quyết là việc quyết định trọng số cho từng đặc trưng khi kết hợp chúng với nhau. Thông thường, các trọng số cần được điều chỉnh tùy vào tác vụ đang thực hiện nhưng thay vào đó, G. Ciocca đã đề xuất dùng chính con số khoảng cách của từng đặc trưng làm trọng số. *dCEH*chỉ có giá trị lớn khi và chỉ khi có ít nhất 2 giá trị khoảng cách riêng lớn làm nổi bật sự khác nhau giữa hai ảnh đang xét.

#### Thuật toán phát hiện điểm đường cong cao

Để rút trích được các keyframe đại diện cho một shot, G. Ciocca đã đề xuất một phương pháp dựa trên phương pháp phát hiện các điểm đường cong cao của Chetverikov [13]. Phương pháp này từng được sử dụng để phân tích dáng và phát hiện các điểm mà tại đó có sự biến đổi đột ngột về giá trị trong một đồ thị 2-D. Đối với bài toán rút trích keyframe, ta có thể xem như các giá trị khoảng cách *dCEH* của mỗi frame *t* trong shot như một điểm trên một đồ thị và biểu diễn thành một đường cong. Thuật toán phát hiện điểm đường cong cao có hai giai đoạn. Ở giai đoạn 1, gọi *P* là điểm đang xét, *O* và *R* lần lượt là các điểm nằm bên trái và phải *P* trên đồ thị. Gọi *α(P)* là góc mở tại điểm P. *α(P)* được tính dựa vào các bộ ba điểm *O*, *P* và *R* trong một cửa sổ duyệt có kích thước cố định và tâm tại *P* (*O* và *R* thay đổi) và góc mở nhỏ nhất có thể tại *P* trong số các bộ ba sẽ được gán cho *α(P)*. Gọi dOP, dPR và dOR lần lượt là khoảng cách từ *O* đến *P*, *P* đến *R* và *O* đến *R*. Để tính được góc mở α tại P đối với một tam giác được tạo bởi ba điểm *O*, *P*, *R* bất kỳ, ta có công thức như sau:

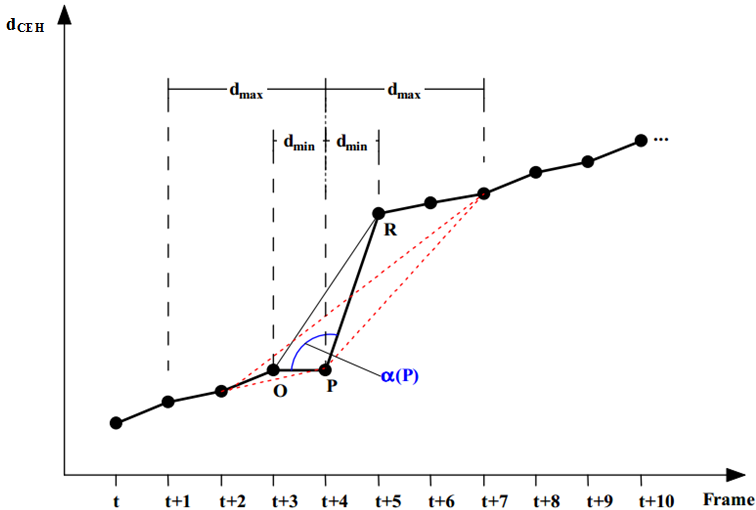
Một tam giác *OPR* thỏa mãn điều kiện về khoảng cách giữa các điểm (theo chiều *x* trong đồ thị):

và ràng buộc của giá trị góc:

là một tam giác hợp lệ. *dmin*và *dmax* giới hạn số lượng tam giác được đưa vào xét và *αmax* được dùng để loại bỏ các góc α quá tù. Như vậy, ta có góc mở tại *P* là:

Nếu tại *P* không tồn tại một tam giác hợp lệ nào thì *α(P)* được gán bằng *π*.

Trong giai đoạn 2, ta tiến hành đối sánh góc mở của các điểm ứng viên với các ứng viên khác lân cận. Gọi N là điểm ứng viên lân cận với P thỏa mãn điều kiện với *w* là kích thước cửa sổ duyệt (thông thường *w* = *dmax*). Điểm ứng viên P sẽ bị loại nếu tồn tại một điểm ứng viên N lân cận sao cho . Các điểm ứng viên vượt qua giai đoạn 2 này sẽ được xem như một điểm đường cong cao. Tại đây, các keyframe có thể được rút trích bằng cách lấy điểm giữa (midpoint) giữa 2 điểm đường cong cao liên tiếp nhau. Frame đầu tiên và cuối cùng của một shot bất kỳ được ngầm hiểu là điểm đường cong cao. Do đó nếu shot không có thêm điểm đường cong cao nào, tức là không có nhiều thay đổi lớn trong shot thì shot có thể được biểu diễn chỉ bằng một frame duy nhất.

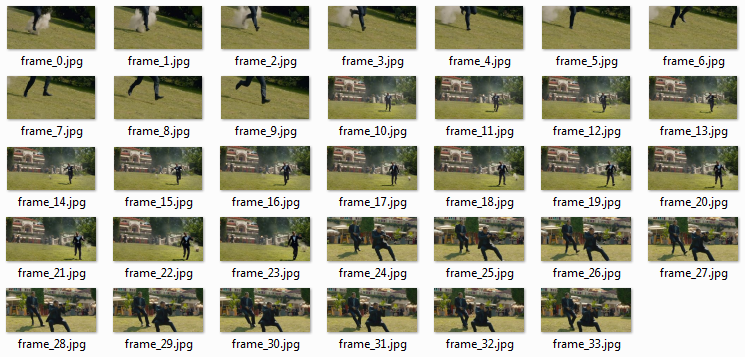


*Hình 3.4: Biểu đồ phát hiện điểm đường cong cao* (Nguồn [12])

Đồ án đã chọn tham số được đề xuất bởi G. Ciocca cho thuật toán này, trong đó: *dmax*= 3, *dmin* = 1 và *αmax* ≤ 140o. *αmax* càng lớn thì số lượng keyframe rút trích được càng lớn và ngược lại.

Để làm rõ hơn phương pháp này, đồ án đã thực hiện một ví dụ như sau:

Trong ví dụ này, *dmax*= 3, *dmin* = 1, *αmax* = 90o và *w* = 5;

Hình 3.5: Ví dụ các frame có trong một shot

Như *Hình 3.5*, ta có một shot có *n* = 34 frame. Mỗi frame được rút trích đặc trưng cấu trúc màu, lược đồ biên cạnh và vân đồng dạng và được tính khoảng cách *dCEH* với frame kế tiếp. Như vậy, ta có được *n* – 1 = 33 giá trị khoảng cách.

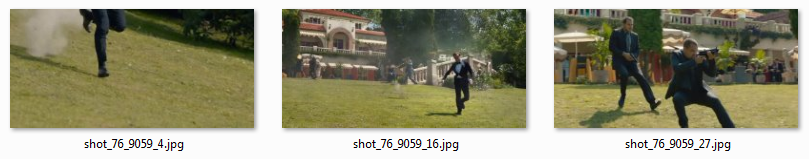
|  |  |  |
| --- | --- | --- |
| **Frame (t)** | **Khoảng cách** | **αt (deg)** |
| 0 | 3.91919494 | 0 |
| 1 | 4.81263828 | 180 |
| 2 | 8.37200069 | 31 |
| 3 | 4.7795558 | 59 |
| 4 | 5.83391619 | 69 |
| 5 | 3.72500944 | 51 |
| 6 | 5.83270407 | 52 |
| 7 | 3.80714607 | 28 |
| 8 | 3.97494364 | 48 |
| **9** | **58.1901855** | **2** |
| 10 | 0.294803947 | 64 |
| 11 | 0.814338326 | 180 |
| 12 | 0.294877499 | 83 |
| 13 | 0.464750707 | 180 |
| 14 | 0.35809347 | 180 |
| 15 | 0.463378221 | 180 |
| 16 | 0.254099101 | 180 |
| 17 | 0.310577899 | 180 |
| 18 | 0.33345142 | 180 |
| 19 | 0.346814245 | 180 |
| 20 | 0.408902615 | 180 |
| 21 | 0.21164462 | 83 |
| 22 | 0.373784721 | 180 |
| **23** | **28.2221813** | **4** |
| 24 | 2.83512044 | 58 |
| 25 | 3.50673914 | 85 |
| 26 | 2.3561368 | 53 |
| 27 | 3.31931233 | 180 |
| 28 | 2.58951283 | 180 |
| 29 | 2.17835355 | 180 |
| 30 | 2.27420473 | 180 |
| 31 | 2.0084312 | 180 |
| 32 | 2.03931856 | 0 |

Bảng 3.1: Bảng chi tiết các giá trị khoảng cách, góc mở tại mỗi frame trong shot. Nếu frame t bất kỳ có góc mở αt >αmax, αt sẽ được tự động gán bằng 180o.

Sau đó, các giá trị này được thể hiện lên đồ thị với trục hoành là số thứ tự frame từ 0 đến 33 và trục tung là giá trị khoảng cách, ta được một đồ thị như hình sau:

*Hình 3.6: Đồ thị thể hiện khoảng cách giữa các frame liên tiếp nhau.*

Gọi αt là góc mở tại frame thứ t. Theo như Hình 3.6 và Bảng 3.1, các frame thứ 9 và 23 là các điểm đường cong cao được phát hiện do α9 = 2o và α23 = 4o nhỏ hơn ngưỡng *αmax* = 90o và là góc mở nhỏ nhất trong bán kính *w* = 5. Frame 0 và 33 được mặc định là điểm đường cong cao do chúng nằm ở đầu và cuối shot. Các keyframe được rút trích bằng cách lấy frame nằm giữa hai điểm đường cong cao liên tiếp. Như vậy, frame thứ 4, 16 và 27 được chọn làm keyframe cho shot này.



Hình 3.7: Các frame được xem là keyframe và rút trích từ shot

# CHƯƠNG 4 TRUY TÌM FRAME ẢNH TRONG VIDEO

*Sau quá trình rút trích keyframe, chương 4 của đồ án sẽ trình bày cách thức rút trích đặc trưng keyframe và thực hiện truy vấn*

## Rút trích đặc trưng keyframe

Như đã trình bày ở trên, sử dụng kết hợp nhiều loại đặc trưng sẽ làm tăng hiệu quả truy vấn. Vấn đề được đặt ra là đặc trưng nào được chọn và cách kết hợp chúng như thế nào. Theo như nghiên cứu của Cvetković [14] và để bảo đảm tốc độ tính toán nhanh mà vẫn giữ được hiệu quả truy vấn cao, đồ án đã chọn cách kết hợp sớm đặc trưng bố cục màu (CLD) với 10 tham số cho kênh Y và 6 tham số cho mỗi kênh Cr và Cb) và lược đồ biên cạnh (EHD). Như vậy, mỗi keyframe sẽ được biểu diễn bằng một vector đặc trưng có chiều dài *nCLD*+ *nEHD* = 22 + 80 = 102. Ngoài ra các đặc trưng thuộc nhóm MPEG-7.

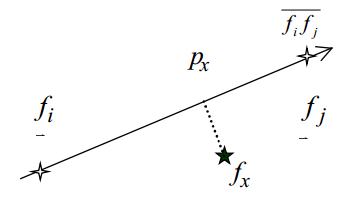
## Truy vấn video

Mỗi video được biểu diễn bởi một bộ đặc trưng rút trích từ các keyframe. Với một ảnh bất kỳ được đưa vào từ người dùng, ảnh sẽ được rút trích đặc trưng và tính khoảng cách đối với từng video. Gọi *V* là cơ sở dữ liệu video hiện có, mỗi video *Vi* có keyframe và *I* là ảnh mà người dùng đưa vào truy vấn. Khoảng cách từ ảnh đến video *Vi*được tính như sau:

Với , *Vij* là keyframe thứ *j* của video *Vi*và ***D*** là hàm tính khoảng cách giữa ảnh *I* với keyframe *Vij*. Do ảnh *I* có khả năng xuất hiện trong nhiều hơn 1 video, kết quả đạt được là một danh sách video truy vấn được sắp xếp theo thứ tự tăng dần khoảng cách.

## Truy vấn shot

Đối với các video truy vấn được, ta cần xác định ảnh truy vấn *I* thuộc vào shot nào của video để có thu hẹp không gian tìm kiếm khi truy tìm frame ảnh tương đồng với ảnh *I*. Do việc đối sánh trực tiếp giữa hai ảnh như **phần 4.2** không bảo đảm ảnh *I* thật sự thuộc shot có chứa keyframe tương đồng với ảnh *I* (đặc biệt là đối với video có phụ đề), đồ án đã áp dụng phương pháp truy vấn shot được đề xuất bởi Li Zhao [9] sử dụng một độ đo mới được phát triển từ cơ sở lý thuyết của Nearest Feature Line. Phương pháp Nearest Feature Line xét tới mối quan hệ theo thời gian và tương quan giữa hai keyframe trong cùng một shot bằng cách dùng một đường nối để xấp xỉ quỹ đạo thay đổi liên tục giữa hai keyframe.



*Hình 4.1: Hình mô tả phương pháp truy vấn dùng Feature Line* (Nguồn [9])

Gọi *Fi* và *Fj*là hai keyframe được ánh xạ vào hai điểm *fi* và *fj*trong không gian đặc trưng. Gọi là đường thẳng nối *fi*và *fj*và được gọi là một đường đặc trưng (Feature Line). Gọi *fx* là một điểm trong không gian đặc trưng của ảnh truy vấn *I*, khoảng cách giữa điểm *fx* và đường , để tìm *px* là hình chiếu của *fx* trên , ta dùng công thức sau:

để

khoảng cách giữa hai điểm trong không gian đặc trưng được tính bằng độ đo Euclidean. Ta cần tính khoảng cách giữa *fx* với mọi đường đặc trưng trong tất cả các shot và sắp xếp tất cả giá trị khoảng cách theo hướng tăng dần. Shot có khả năng chứa ảnh *I* cao nhất nếu tồn tại một đường đặc trưng có khoảng cách đến ảnh *I* nhỏ nhất. Keyframe trên đường đặc trưng có khoảng cách đến ảnh *I* nhỏ nhất được chọn làm kết quả cuối cùng và xuất ra khung thời gian mà keyframe đó xuất hiện trong video.

# CHƯƠNG 5 THỰC NGHIỆM

## Giới thiệu tập dữ liệu thực nghiệm

Nguồn gốc: Các video được thu thập từ nhiều nguồn khác nhau (DVD, Youtube, Mail.ru, v.v…). Mỗi video có độ dài trong khoảng từ 1 đến 3 giờ, độ phân giải nằm trong khoảng từ 640×480 đến 1280×720.

Thống kê mẫu:

|  |  |
| --- | --- |
| **Loại video** | **Số lượng** |
| Phim | 30 |
| Tennis | 10 |
| Bóng đá | 10 |

Bảng 5.1: Số lượng mẫu video trong tập dữ liệu

|  |  |  |
| --- | --- | --- |
| **Loại video** | **Số lượng shot** | **Số lượng keyframe** |
| Phim | 492 ~ 1768 | 944 ~ 5179 |
| Tennis | 360 ~ 1319 | 620 ~ 7337 |
| Bóng đá | 343 ~ 849 | 1645 ~ 4911 |

Bảng 5.2: Số lượng shot và keyframe rút trích được từ các mẫu video

Bộ dữ liệu kiểm thử được xây dựng bằng cách lấy ngẫu nhiên 100 frame trong mỗi video trong cơ sở dữ liệu. Khi thực hiện kiểm thử, đồ án chỉ thực hiện kiểm thử trên các video trong cùng một loại.

## Kết quả truy vấn video

Độ chính xác (precision), độ phủ (recall) và độ chính xác trung bình (Mean Average Precision – MAP) là các giá trị được sử dụng trong hầu hết các bài toán truy vấn để đánh giá hiệu quả của phương pháp truy vấn. Ta có công thức tính các giá trị đánh giá này như sau:

Với *rij­*là hạng của video đúng thứ *j* đối với ảnh truy vấn *Qi*, |*Ri*| là số ảnh đúng đối với lần truy vấn thứ *i*, n là số ảnh dùng để truy vấn. Để thực hiện kiểm thử, đồ án đã chọn ra 100 ảnh ngẫu nhiên trong mỗi video trong cơ sở dữ liệu để làm tập dữ liệu kiểm thử. Do các video trong cơ sở dữ liệu không có liên hệ gì về mặt nội dung nên tất cả ảnh kiểm thử chỉ thuộc vào một video duy nhất. Với mỗi lần truy vấn, đồ án chọn ra 3 kết quả có hạng cao nhất. Kết quả đồ án đạt được sau khi đã lấy trung bình độ chính xác, độ phủ và MAP qua các lần truy vấn như sau:

|  |  |  |  |
| --- | --- | --- | --- |
| **Loại video** | **Đánh giá** | | |
| **Độ chính xác (%)** | **Độ phủ (%)** | **MAP (%)** |
| Phim | 30.1994 | 90.6 | 89.3623 |
| Tennis | 32.7214 | 96.1841 | 93.2553 |
| Bóng đá | 32.4667 | 97.4 | 95.5332 |

Bảng 5.3: Kết quả đánh giá phương pháp truy vấn video của đồ án

1. **Kết quả truy vấn shot**

Đối với bài toán truy vấn shot, đồ án sử dụng độ chính xác (precision) và trọng số (weight score) để đánh giá chất lượng truy vấn. Gọi *q* là ảnh truy vấn, *r1*, *r2*, … , *rm*là *m* shot có thứ hạng cao nhất truy vấn được. Ta có công thức tính độ chính xác*η(q, m)* như sau:

*Nq* là tổng số shot truy vấn được, *m* < *Nq*. Khi *k* càng lớn thì wk càng nhỏ, do đó shot truy vấn được đúng có thứ hạng càng cao thì càng đóng góp nhiều vào *η(q, m)*. *wk* được chuẩn hóa sao cho nếu *m* = *Nq*, tức là mọi shot truy vấn được đều đúng thì *η(q, m)* sẽ đạt giá trị cao nhất là 1. Với mỗi lần truy vấn, đồ án chọn ra 5 kết quả có hạng cao nhất. Độ chính xác trung bình qua các lần truy vấn tính được theo từng loại video được thể hiện trong bảng sau:

|  |  |
| --- | --- |
| **Loại video** | **Độ chính xác (%)** |
| Phim | 34.3044 |
| Tennis | 27.7467 |
| Bóng đá | 23.3095 |

*Bảng 5.4: Kết quả đánh giá phươg pháp truy vấn shot*

# CHƯƠNG 6 KẾT LUẬN

*Nội dung chương này gồm có nhận định lại vấn đề, tiềm năng phát triển của phương pháp được nghiên cứu và các hướng phát triển trong tương lai*

Đồ án đã tìm hiểu các phương pháp phân đoạn shot và rút trích keyframe để đại diện cho một video. Đồ án cũng tìm hiểu các đặc trưng thuộc nhóm đặc trưng MPEG-7 và cách kết hợp chúng để cho ra kết quả truy vấn hiệu quả mà vẫn bảo đảm được tốc độ truy vấn nhanh. Về mặt thực nghiệm, đồ án đã ứng dụng phương pháp rút trích keyframe của G. Ciocca [14] và truy vấn shot của Li Zhao [9] để cài đặt một hệ thống truy tìm frame ảnh trong cơ sở dữ liệu video với một ảnh đầu vào cho trước.

Hiệu suất của hệ thống truy tìm frame ảnh trong cơ sở dữ liệu video phụ thuộc vào hiệu suất của các bài toán con:

* Tóm tắt video và rút trích keyframe.
* Truy vấn video.
* Truy vấn shot.

Một lỗi trong bất kỳ công đoạn nào cũng có thể dẫn tới kết quả thẩm định sai.

Vì vậy, trong nghiên cứu lần này đồ án đã trình bày cách làm của từng bài toán con

**Tóm tắt video và rút trích keyframe**

Đồ án đã thực hiện tóm tắt video và phân đoạn thành các shot. Thao tác ghép shot diễn ra khi và chỉ khi số lượng frame trong một shot nhỏ hơn số khung hình/giây của video gốc. Đồ án áp dụng phương pháp rút trích keyframe từ các shot của G. Ciocca [14] cùng với các đặc trưng MPEG-7 để xây dựng một cơ sở dữ liệu đại diện cho video.

**Truy vấn video**

Sau khi rút trích được các keyframe cho từng video trong cơ sở dữ liệu. Đồ án đã dựa vào phân tích của Cvetković [14] và kết hợp đặc trưng bố cục màu (CLD) và lược đồ biên cạnh (EHD) để mô tả từng keyframe và đưa vào lưu trữ. Với một ảnh bất kỳ được đưa vào từ người dùng, ảnh được rút trích đặc trưng tương ứng và truy vấn các video có khả năng chứa frame tương đồng với ảnh đó cao nhất.

**Truy vấn shot**

Sau khi có được danh sách các video có khả năng chứa ảnh đầu vào, đồ án tiếp tục áp dụng phương pháp truy vấn shot video bằng phương pháp Nearest Feature Line được đề xuất bởi Li Zhao [9] để truy tìm shot có khả năng chứa frame tương đồng với ảnh đầu vào cao nhất. Keyframe trong shot được chọn có khoảng cách ngắn nhất với ảnh đầu vào được chọn làm kết quả và trả ra khung thời gian mà frame đó xuất hiện trong video.

**Hướng phát triển**

Trong nghiên cứu này, đồ án chỉ mới giải quyết được một góc nhỏ trong bài toán truy vấn video và vẫn có nhiều khả năng mở rộng:

* Giai đoạn rút trích keyframe tuy thực hiện tốt nhưng mất quá nhiều thời gian để rút trích nên cần được cải tiến để thực thi nhanh hơn.
* Hệ thống có thể được mở rộng ra cho phép người dùng đưa vào một shot video để truy vấn.

*Như vậy chương cuối cùng đã nhận định lại vấn đề, tiềm năng phát triển của phương pháp được nghiên cứu, tóm tắt lại hướng tiếp cận của khoá luận và kết quả đạt được, cùng với một số hướng phát triển bài toán trong tương lai đã được đưa ra.*

# THAM KHẢO

[1] Liu, T.-Y., Lo, K.-T., Zhang, X.-D., Fengc, J, **A new cut detection algorithm with constant false-alarm ratio for video segmentation**, J. Vis. Commun. Image R. 15, 132–144 (2004).

[2] Fang, H., Jiang, J., Feng, Y, **A fuzzy logic approach for detection of video shot Boundaries**, Pattern Recognition 39, 2092–2100 (2006).

[3] R. Kasturi and R.C. Jain. **Dynamic vision.** In R. Kasturi and R.C. Jain, editors, Computer Vision: Principles, pages 469–480. IEEE Computer Society Press, Washington, 1991.

[4] A. Dailianas, R.B. Allen, and P. England. **Comparison of automatic video segmentation algorithms**. In SPIE Conference on Integration Issues in Large Commercial Media Delivery Systems, volume 2615, pages 2–16, Philadelphia, PA, October 1995.

[5] R. Zabih, J. Miller, and K. Mai. **A feature-based algorithm for detecting and classifying production effects.** Multimedia Systems, 7(2):119–128, March 1999.

[6] N. Vasconcelos and A. Lippman. **Statistical models of video structure for content analysis and characterization**. IEEE Transactions on Image Processing, 9(1):3–19, January 2001.

[7] S. Eickeler and S. Muller. **Content-based video indexing of tv broadcast news using hidden markov models**, International Conference on Acoustics, Speech, and Signal Processing, pages 2997–3000, Phoenix, AZ, March 1999.

[8] A. Akutsu, Y. Tonomura, H. Hashimoto, Y. Ohba. **Video indexing using motion vectors**, SPIE Conference on Visual Communications and Image Processing, volume 1818, pages 1522–1530, Boston, MA, November 1992.

[9] O. Fatemi, S. Zhang, and S. Panchanathan. **Optical ﬂow based model for scene cut detection**, Canadian Conference on Electrical and Computer Engineering, volume 1, pages 470–473, Calgary, Canada, May 1996.

[8] Ajay divakaran, Regunathan Radhakrishnan, Kadir A Perker, **Motion activity based extraction of key frames from video shots**, IEEE 2002.

[9] Li Zhao, Wei Qi, Stan Z. Li, Shi-Qiang Yang, H. J. Zhang, **Key frame extraction and shot retrieval using nearest feature line**, Microsoft Research China.

[10] Xianglin Zeng, Weiming Hu, Wanqing Liy, Xiaoqin Zhang, Bo Xu, **Key-frame extraction using dominant set clustering**, SCSSE, University of Wollongong, Australia.

[11] Sang Hyun Kim, Rae-Hong Park, **An effective algorithm for video sequence matching using the modified Hausdorff distance and the directed divergence**, IEEE transactions on circuits and systems for video technology, vol. 1.

[12] G. Ciocca, R. Schettini, **An innovative algorithm for key frame extraction in video**, Bicocca degli Arcimboldi 8, 20126, Milano, Italy.

[13] Chetverikov D, Szabo Zs, **A Simple and Efficient Algorithm for Detection of High Curvature Points in Planar Curves**, Proc. 23rd Workshop of the Austrian Pattern Recognition, Group, 1999;175-184.

[14] Stevica Cvetković, Saša V. Nikolić, Slobodan Ilić, **Effective combining of color and texture descriptors for indoor-outdoor image classification**, Electronics and Energetics Vol. 27, No 3, September 2014, pp. 399 – 410

[15] B. S. Manjunath, Jens-Rainer Ohm, Vinod V. Vasudevan, Akio Yamada, **Color and Texture Descriptors**, IEEE transactions on circuits and systems for video technology, vol. 11, no. 6, June 2001