

MỘT KỸ THUẬT CHỈ SỐ HOÁ TỰ ĐỘNG DỮ LIỆU VIDEO DỰA TRÊN ĐÁNH DẤU VÙNG NỀN

Lương Xuân Cường⁽¹⁾, Đỗ Xuân Tiến⁽¹⁾, Đỗ Trung Tuấn⁽²⁾

⁽¹⁾Học Viện Kỹ thuật Quân sự

⁽²⁾Trường Đại học Khoa học Tự nhiên, Đại học Quốc gia Hà Nội

1. Giới thiệu

Cùng với sự phát triển mạnh mẽ của công nghệ thông tin và truyền thông, video số đã và đang trở thành một phần không thể thiếu của nhiều ứng dụng. Số lượng dữ liệu video tăng nhanh cùng với các yêu cầu ngày càng cao của người sử dụng gây áp lực rất lớn đối với các hệ quản trị cơ sở dữ liệu (CSDL) video. Nói chung, việc xây dựng một hệ quản trị CSDL video luôn cần đến các kỹ thuật xử lý dữ liệu phức tạp hơn nhiều so với các dữ liệu đa phương tiện khác như văn bản, âm thanh, hay ảnh tĩnh. Đối với dữ liệu video, việc chỉ số hóa dữ liệu được xem là vấn đề quan trọng [2, 12].

Trong nhiều trường hợp, việc lưu trữ video dưới dạng các clip là quá thô không thuận tiện cho các yêu cầu duyệt và truy vấn. Thực tế, việc phân chia video thành một chuỗi các lia (shot, video shot) sẽ thuận tiện cho việc chỉ số hoá dữ liệu [4, 5, 7]. Lia là một đơn vị vật lý của dòng video [2, 12, 21]. Nó gồm một tập hợp các khung video ghi được từ một hoạt động camera đơn. Để có thể biểu diễn lại dữ liệu video dưới dạng các khoá đặc trưng (xây dựng được một tập chỉ số cho CSDL video), trước hết dòng video cần được phân chia về mặt thời gian thành các đơn vị. Như vậy luôn cần đến các kỹ thuật nhằm phát hiện ranh giới giữa các lia (xin gọi tắt là kĩ thuật PRL).

Các kỹ thuật PRL hiện nay dựa trên các đặc tính hình ảnh mức thấp của video gặp phải một số khó khăn do chúng đòi hỏi nhiều tham số đầu vào và thực hiện xử lý trực tiếp ở mức khung [1, 9, 17]. Kỹ thuật sử dụng biểu đồ màu cần ít nhất tới 3 giá trị ngưỡng màu [6, 7, 23]. Kỹ thuật sử dụng tỷ số biến đổi cạnh cần ít nhất 6 giá trị ngưỡng [3, 11, 18]. Việc lựa chọn các giá trị ngưỡng thích hợp phụ thuộc nhiều vào bản thân từng video và ảnh hưởng nhiều tới kết quả.

Bên cạnh đó cũng đã có một số nghiên cứu nhằm vào các đặc tính hình ảnh mức cao của video. Các khái niệm cảnh (scene) và chuỗi (dãy video, sequence) được xem là những đơn vị logic của dòng video. Cảnh có thể hiểu là một tập hợp các lia liên kế về không gian và thời gian. Các cảnh cùng mang đến một nghĩa nào đó được nhóm vào thành một chuỗi... Hiện một số cấu trúc đa mức nhằm vào các đặc tính video mức cao

[13, 19, 22], đã đạt kết quả trong việc xây dựng các mô hình cấu trúc của hệ thống phân cấp nội dung ngữ nghĩa dòng video. Các nghiên cứu này tập trung vào vấn đề mô hình hóa nhằm thiết kế cấu trúc phân cấp tốt nhất cho việc biểu diễn lại nội dung video. Tuy nhiên việc đưa ra các kỹ thuật tự động hoá việc xây dựng các cấu trúc đó còn nhiều hạn chế [10, 14, 16].

Một trong những khó khăn lớn nhất của các kỹ thuật PRL là phải đối mặt với tính liên tục theo thời gian của dòng video. Các khung video trong một lia thường nội dung có tính tương quan cao và các khung lân cận nhau thì rất giống nhau. Hầu hết các phương pháp PRL đang tồn tại dựa trên kết quả đo sự thay đổi nội dung giữa các khung liên kế. Và thường sử dụng các giá trị ngưỡng (đôi khi dùng cả các ngưỡng thích nghi) [1, 12] để ra quyết định về ranh giới lia. Theo đó mọi sự khác nhau nhỏ hơn ngưỡng đều được coi là do sự biến đổi về thời gian trong một lia và đương nhiên các khung đó vẫn thuộc về cùng một lia. Không nên nhầm lẫn giữa sự khác nhau về nội dung hình ảnh và sự khác nhau do tính liên tục thời gian của chuỗi video. Sự biến đổi trong một lia có thể do nhiều nguyên nhân (sự di chuyển của camera và các đối tượng, các hiệu ứng biên tập...). Chúng có thể tạo nên giá trị khác nhau giữa hai khung lớn hơn ngưỡng và sẽ dẫn đến lỗi trong PRL [20, 21].

Đối với hệ thống các camera quan sát, dòng video có tính thời gian thực cao và CSDL của chúng cũng cần được tổ chức phù hợp để phục vụ cho các yêu cầu duyệt và truy vấn riêng. Đặc biệt là các yêu cầu về tìm kiếm, phát hiện sự xuất hiện hay biến đi của đối tượng và nhận dạng đối tượng. Như vậy nảy sinh yêu cầu về việc phân chia các dòng video thành các cảnh dựa trên cơ sở nội dung của chúng (chỉ số hoá dữ liệu video dựa vào nội dung). Ngoài ra do đặc tính di chuyển của camera yêu cầu phải có các biện pháp tránh nhầm lẫn trong PRL.

Bài báo đề xuất kỹ thuật PRL dựa trên việc đánh dấu đường di chuyển camera qua vùng nền của video, hay đánh dấu vùng nền, cùng với các kỹ thuật nhằm giảm nhiễu để tăng khả năng phát hiện chính xác ranh giới các lia.

2. Kỹ thuật đánh dấu vùng nền nhằm phát hiện ranh giới lia

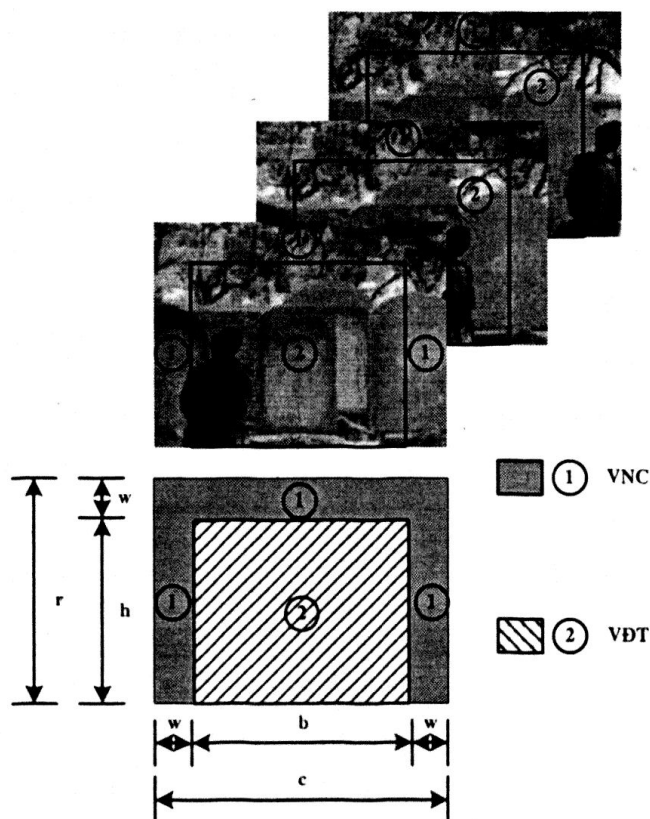
2.1 Cơ sở lý thuyết

Kỹ thuật PRL này dựa trên hai cơ sở sau:

1) Lựa chọn các vùng đặc biệt trên các khung ảnh video để đánh dấu đường di chuyển của camera nhằm tăng hiệu quả cho thuật toán so sánh giữa các khung trong PRL;

2) Vận dụng kỹ thuật nén ảnh theo phương pháp Tháp Gauss (Gaussian Pyramid) [8, 11, 15] để giảm nhỏ kích thước ảnh phục vụ cho PRL.

Khi một lia được thực hiện từ một hoạt động camera, theo vết di chuyển của camera là một cách để xác định ranh giới giữa các lia. Có thể làm được việc này bằng cách đánh dấu vùng nền trong các khung video. Vùng nền (VNC) được định nghĩa như hình 1. Lí do chọn VNC trong các khung video có dạng chữ Π : (i) trước hết là do phần đáy của một khung thường là phần thể hiện các đối tượng hay còn gọi là vùng cận cảnh, hay vùng đối tượng (VĐT); (ii) vùng nền chữ nhật nhỏ trên đỉnh phản ánh chuyển động ngang của camera; (iii) hai cột trái/ phải phản ánh chuyển động dọc của camera. Kết hợp giữa vùng đỉnh và các cột có thể vạch ra chuyển động bất kỳ theo hướng đường chéo.

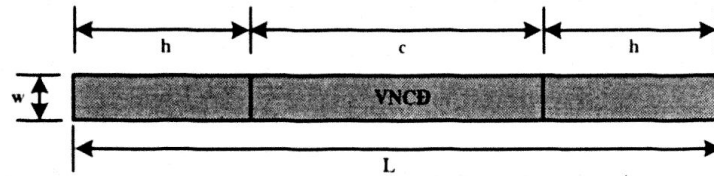


Hình 1. Vùng nền cố định

Dựa trên cách xác định vùng nền, chúng tôi thấy có thể thực hiện kỹ thuật PRL thông qua việc xác định mức độ phân biệt VNC của hai khung liền kề. Mặc dù độ phức tạp của thuật toán có tăng, nhưng bù lại phương pháp này lại tỏ ra hiệu quả trên các phương diện sau:

- 1) Số lượng phép so sánh giảm do kích thước vùng cần so sánh không phải là toàn bộ khung.
- 2) Giảm được phát hiện sai đáng kể so với một số phương pháp khác do các biện pháp lọc nhiễu (mục 3.1) và phân biệt khá tốt được tính liên tục thời gian của dòng video với các chuyển động của camera và/hoặc đối tượng.

Để so sánh các vùng nền thuận tiện, nên sử dụng các hình đã được chuyển dạng bằng cách kéo thẳng VNC. Vùng nền kéo thẳng của VNC được gọi là vùng nền chuyển đổi (kí hiệu là VNCD); và nó là một mảng các pixel hai chiều, như trong hình 2.



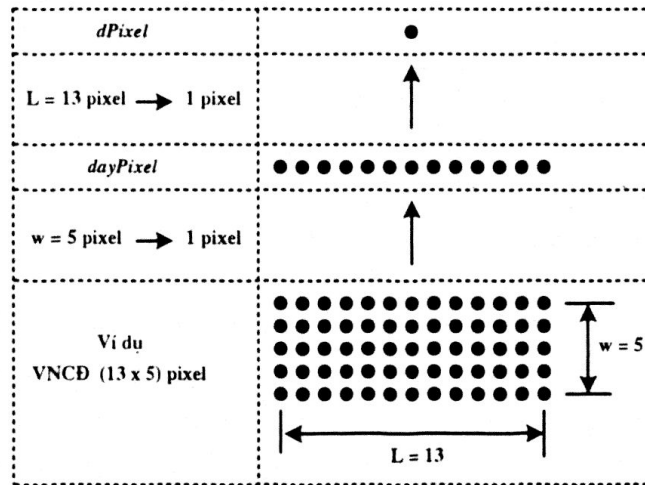
Hình 2. Vùng nền chuyển đổi

Trong kỹ thuật nén ảnh số, phương pháp Tháp Gauss có tác dụng làm giảm nhỏ kích thước ảnh trên cả hai phương diện (i) độ phân giải; và (ii) tốc độ lấy mẫu. Xuất phát từ một ảnh gốc, qua một số lần lặp lọc dải thấp, mỗi lần sẽ nhận được một ảnh đã được giảm nhỏ. Bộ lọc được thiết kế để tính trung bình cục bộ có trọng số trên các cửa sổ mẫu pixel (các nhân sinh) dựa trên các đáp ứng xung hai chiều rất gần với đường cong Gauss. Các nhân sinh chính là các tích chập Gauss, thường được chọn đối xứng, ví dụ (5×5) pixel hoặc (16×16) pixel. Nếu xếp chồng các ảnh nhận được sẽ có một cấu trúc hình tháp. Mỗi một pixel ở tầng trên của tháp sẽ là kết quả tính trung bình theo cách kể trên của cửa sổ mẫu ở tầng dưới.

Bằng cách vận dụng phương pháp Tháp Gauss, trong đó sử dụng tích chập Gauss là một cửa sổ mẫu thích nghi (biến đổi) để chuyển một VNCD hai chiều thành một dãy các pixel (kí hiệu là $dayPixel$) và cuối cùng sẽ chuyển thành một pixel đơn (kí hiệu là $dPixel$). Kỹ thuật này được minh họa như trong hình 3. Theo hình vẽ, một VNCD 13×5 sẽ được giảm lần lượt theo nhiều bước. Trước hết 5 pixel trong từng cột được giảm thành 1 pixel và VNCD sẽ chuyển thành một dãy 13 pixel. Sau đó $dayPixel$ này sẽ giảm thành $dPixel$, ký hiệu là $dPixel_i^N$ ($dPixel$ vùng nền của khung thứ i). Kích thước của VNCD được tính như sau:

Giả sử kích thước của khung là r và c ; của VNCD là w và L ; của VĐT là h và b (hình 1). Trước hết cần ước lượng các tham số này là w' , L' và h' , b' . Chọn w' bằng 10% chiều rộng của khung, tức là $w' = \lfloor c/10 \rfloor$. Giá trị này được chọn theo kinh nghiệm sử dụng các clip video. Kích thước đó là đủ lớn để bao phủ VNC và VĐT. Từ w' , có thể tính được các giá trị khác như sau: $b' = c - 2 \cdot w'$; $h' = r - w'$; và $L' = c + 2 \cdot h'$. Vận dụng phương pháp Tháp Gauss, kích thước của VNCD phải nằm trong tập kích thước sau: $\{1, 5, 13, 29, 61, 125, \dots\}$. Theo ví dụ minh họa, với kỹ thuật này sẽ giảm được 5 pixel xuống 1, 13 xuống 5, 29 xuống 13.... Nói chung, phần tử thứ j (s_j) trong tập này được tính theo:

$$s_j = 1 + \sum_{i=2}^j 2^i \quad \text{với } j = 1, 2, 3, \dots \quad (1)$$



Hình 3. Ví dụ dayPixel và dPixel

Sử dụng tập kích thước này, các giá trị thích hợp cho w là trong tập mà gần nhất với w' . Số gần nhất này được xác định: trước hết tính $j = 2 + \log_2 ((w\Box + 3)/6)$. Thay thế giá trị j này vào (1) sẽ nhận được w mong muốn. Tương tự có thể tính được L . Kết quả của phương pháp gần đúng này minh họa trên bảng 1. Ví dụ với $c=160$, nhận được $w\Box = \lfloor 160/10 \rfloor = 16$, giá trị j tương ứng là 3. Thay j vào (1) nhận được giá trị thích hợp của w là 13.

h', b', w', L'	Giá trị gần nhất	H, b, w, L
1, 2	1	1
3, 4, ..., 8	5	3
9, 10, ..., 20	13	13
21, 22, ..., 44	29	29
45, 46, ..., 92	61	61
...

Bảng 1. Thí dụ minh họa

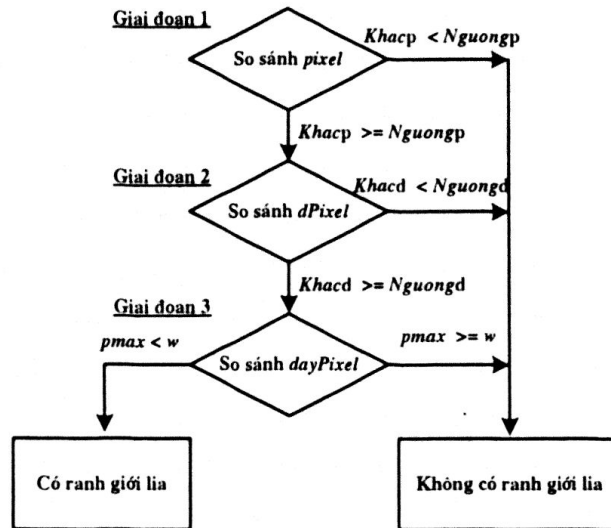
2.2 Thuật toán

Thuật toán sử dụng *dayPixel* và *dPixel* để PRL thể hiện trên hình 4. Hai giai đoạn đầu thường được sử dụng để kiểm tra nhanh cho những trường hợp rõ ràng. Chỉ khi nào không phát hiện được mới sử dụng đến giai đoạn 3. Thuật toán được mô tả chi tiết như sau:

1) Giai đoạn 1 - So sánh *pixel*: Một pixel trong mỗi VNC được so với pixel ở cùng vị trí và một số pixel lân cận (*Vungp*) trong VNC của khung liền kề nó. Nếu tổng số pixel trong hai VNC khác nhau *Khacp* nhỏ hơn một giá trị ngưỡng (*Khacp < Nguongp*), hai khung tương ứng với hai VNC coi như cùng một lia và xét tới khung tiếp theo. Ngược lại, sẽ chuyển tới giai đoạn 2.

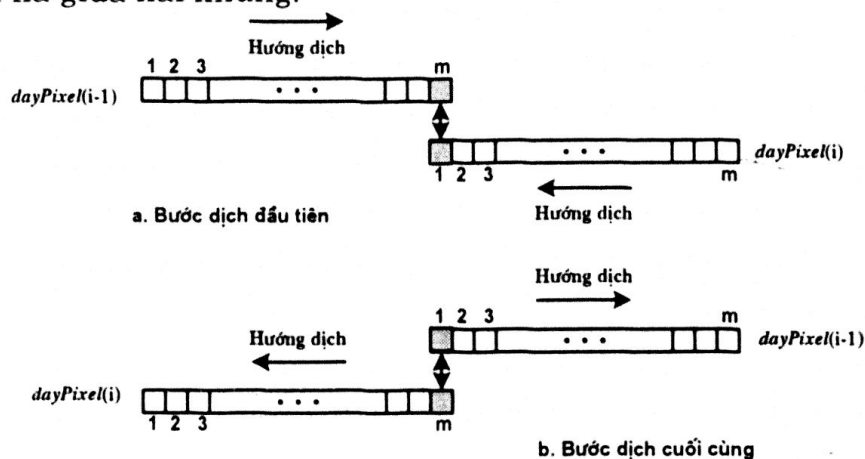
2) Giai đoạn 2 - So sánh *dPixel*: Xét đến sự khác nhau của 2 *dPixel* của hai khung. Nếu sự khác nhau của giá trị *dPixel* là *Khacd* nhỏ hơn một giá trị ngưỡng

($Khacd < Nguongd$), hai khung coi như cùng một lia và sẽ xét tới khung tiếp theo. Ngược lại sẽ chuyển sang giai đoạn tiếp theo.



Hình 4. Thuật toán

3) Giai đoạn 3 - So sánh *dayPixel*: Tính toán xem hai khung chia sẽ bao nhiêu phần nên chung. Hai *dayPixel* được so sánh bằng cách dịch chúng đi theo chiều ngược lại (hình 5). Mỗi bước dịch là một pixel và sẽ tính được số điểm khớp nhau p tương ứng (số các giá trị khớp nhau liên tiếp p của *dayPixel* của khung i (*dayPixel_i*) và của khung $i-1$ (*dayPixel_{i-1}*)). Ví dụ trong bước dịch đầu tiên, giá trị pixel đầu tiên của *dayPixel_i* được so sánh với giá trị pixel cuối m của *dayPixel_{i-1}*. Giai đoạn này kết thúc khi mà giá trị pixel cuối m của *dayPixel_i* được so với giá trị pixel đầu tiên của *dayPixel_{i-1}*. Nếu số cực đại của tất cả các giá trị điểm khớp ($pmax$) lớn hơn chiều rộng của VNC (w), hai khung coi như cùng một lia và sẽ xét tới khung tiếp theo. Ngược lại, đã phát hiện ra một chuyển đổi lia giữa hai khung.



Hình 5. Bước dịch đầu tiên và cuối cùng của *dayPixel*

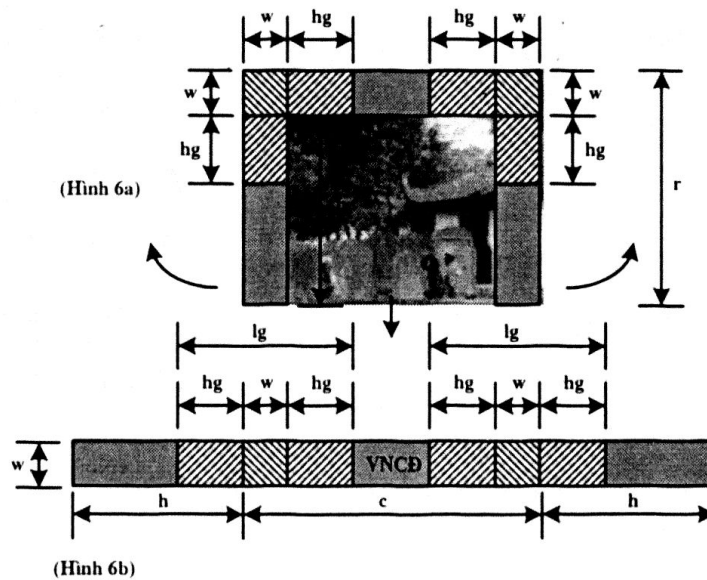
Để tăng tốc độ xử lý khi thực hiện các phép toán so sánh *pixel*, *dPixel*, *dayPixel*, có áp dụng các kỹ thuật nâng cao hiệu quả phân đoạn video mà các tác giả đã đề xuất [12].

3. Mở rộng kỹ thuật phát hiện ranh giới lia

Phần này đề cập hướng mở rộng kỹ thuật PRL, cho phép hạn chế các sai sót trong việc phân đoạn.

3.1 Lựa chọn các góc vùng nền để lọc nhiễu

Kỹ thuật PRL mô tả ở trên kết hợp các khung có màu trung bình của vùng nền giống nhau vào cùng một cảnh [8]. Về mặt ngữ nghĩa, kỹ thuật này phát hiện một sự kiện xuất hiện ở cùng một vị trí. Nếu kết quả chính xác nó sẽ bước quan trọng trong quá trình chỉ số hoá và chú thích dữ liệu video. Do đó nó chưa thật sự hoàn hảo trong một số trường hợp sau:



Hình 6. Lựa chọn các vùng góc nền của khung video

1) Trường hợp 1: Nếu một đối tượng đột nhiên xuất hiện trong vùng nền (ví dụ có cận cảnh hoặc các đối tượng di chuyển ngang qua camera), các khung này có thể được chia nhậm vào các cảnh khác nhau;

2) Trường hợp 2: Nếu vùng nền của hai lia khác nhau ngẫu nhiên giống nhau, nhưng chúng chứa các đối tượng khác nhau ở những vị trí khác nhau, các khung này có thể được chia nhậm vào cùng một cảnh;

3) Trường hợp 3: Nếu vùng nền khác nhau khá nhiều nhưng các đối tượng chính là giống nhau (ví dụ trong một loạt khung mà cùng một người nhưng đi lại từ một vị trí này tới một vị trí khác), những khung đó có thể được chia nhậm vào các cảnh khác nhau.

Trong trường hợp 1, các đối tượng xuất hiện như vậy được xem là nhiễu. Trong hai trường hợp sau, vùng nền được coi là nhiễu. Việc phát hiện ra vùng nhiễu là khó khăn, vị trí của nó biến đổi theo các cảnh. Tuy nhiên cần thiết phải có các giải pháp giảm nhiễu, chỉ sử dụng các thông tin đúng trong các khung video để tăng mức độ chính xác của phân đoạn.

Trong mỗi khung video có thể nhận thấy rằng các phần trung tâm của VNC là rất nhạy với nhiễu. Đặc biệt trong các trường hợp ngắt cảnh vì sự di chuyển của đối tượng vào VNC hoặc đột ngột xuất hiện và bành trướng về kích thước (cận cảnh). Bằng việc chọn tính toán thêm các vùng góc của VNC (hình 6), sẽ có thêm các thông tin để lọc nhiễu, giảm được phân đoạn sai. Ngoài số đo giá trị màu trung bình xấp xỉ của toàn bộ VNCD là $dPixel$, còn tính thêm các giá trị tương ứng cho góc trái và góc phải ($dPixelTrai$ và $dPixelPhai$).

Kích thước của các vùng góc (hình 6.a) và phần kéo thẳng của nó (hình 6.b), với độ dài lg và chiều cao hg được xác định:

$$lg = 2 \times hg + w \quad (2)$$

Theo yêu cầu của phương pháp Tháp Gauss, lg phải là một trong các giá trị trong tập S xác định theo biểu thức (1). Trong tập này chọn lg là số đầu tiên lớn hơn w . Sau đó xác định hg theo lg :

$$hg = (lg - w) / 2 \quad (3)$$

Các tính toán trên được thực hiện cho mỗi mặt phẳng màu R, G và B trong mô hình màu RGB [11].

3.2 Sử dụng $dPixelTrai$ và $dPixelPhai$ như bộ lọc nhiễu

Trong thuật toán đã trình bày (ii), bổ sung thêm giai đoạn 4 để lọc nhiễu. Sau khi đã thực hiện xong giai đoạn 3 và phát hiện ra hai khung i và j có khả năng là một ranh giới giữa hai lia.

Đặt $KhacTrai$ và $KhacPhai$ là các sự khác nhau của hai giá trị $dPixelTrai$ và của hai giá trị $dPixelPhai$ giữa hai khung i và j . Chúng được tính như sau:

$$KhacTrai = \frac{100 * |dPixelTrai_i - dPixelTrai_j|}{256} (\%) \quad (4)$$

Tương tự tính được $KhacPhai$. Từ đó xác định được giá trị khác nhau nhỏ nhất $KhacGoc$:

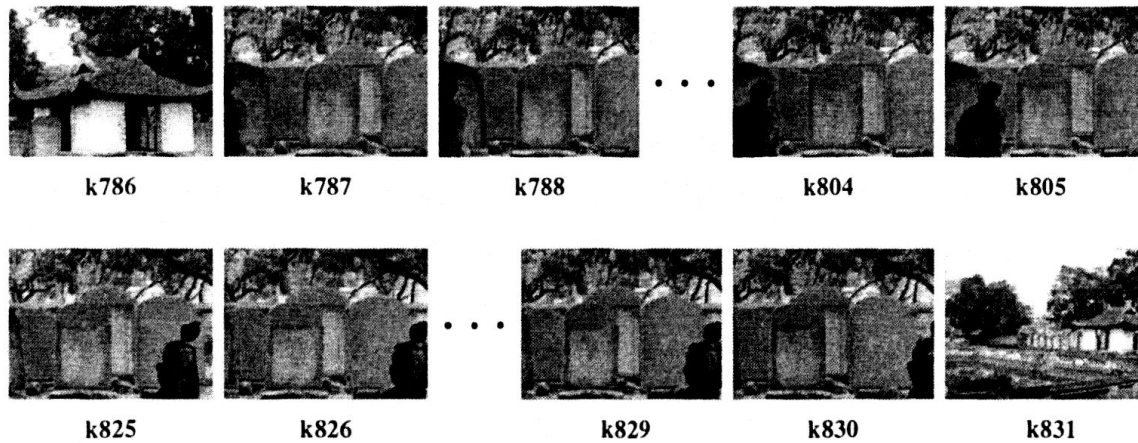
$$KhacGoc = \min(KhacTrai, KhacPhai) \quad (5)$$

Nếu $KhacGoc$ lớn hơn 10% giá trị khác nhau của các $dPixel$ có thể có, lúc này thực sự xuất hiện một ranh giới giữa hai khung i và j . Ngược lại, hai khung coi như nằm cùng trong một lia.

4. Kết quả thực nghiệm

Trong thực nghiệm chúng tôi đã chọn các giá trị ngưỡng là: $Vungd=24$, $Nguongd=Nguongp=10\%$. Hình 7 minh họa các ngắt cảnh khi có sự xuất hiện và biến mất của đối tượng video (con người), xảy ra giữa các khung $k786$ và $k787$ và giữa các khung $k830$ và $k831$ được phát hiện dựa trên kỹ thuật đánh dấu vùng nền. Nếu không có các biện pháp

lọc nhiễu thì sẽ có sự phát hiện nhầm thêm các ranh giới lia giữa các khung *k804* và *k805* và giữa các khung *k825* và *k826* (mặc dù thực tế chúng vẫn thuộc về cùng một lia), vì thay đổi VNC vượt ngưỡng cho phép do có sự chuyển động của đối tượng video.



Hình 7. Ví dụ minh họa các ngắt cảnh dựa trên sự kiện; hình Văn miếu Quốc tử giám

5. Kết luận

Kỹ thuật PRL trình bày trên tỏ ra hiệu quả trong việc phân đoạn các lia trong dòng dữ liệu kích thước lớn, do các camera thu nhận. Nó là cơ sở tốt cho việc phân đoạn các cảnh video dựa trên nội dung. Đây là bước quan trọng trong quá trình chỉ số hóa và chú thích dữ liệu video. Mặc dù thực tế không giảm được nhiều số phép so sánh do phải thực hiện các giai đoạn khá phức tạp trong thuật toán, nhưng nhờ các kỹ thuật lọc nhiễu và qua một số giai đoạn so sánh nên hiệu quả PRL là khá cao. Trong kho dữ liệu thu được của các camera quan sát, ngoài nhu cầu truy vấn dữ liệu theo thời gian, việc ngắt cảnh dựa trên sự xuất hiện hoặc biến mất của các đối tượng video là cơ sở cho các truy vấn theo sự kiện và cao hơn là nhận dạng các đối tượng. Bài báo còn mở ra hướng xây dựng hệ thống phân cấp các cảnh video dựa trên các kết quả PRL kể trên làm cơ sở cho việc chỉ số hóa tự động dữ liệu video.

TÀI LIỆU THAM KHẢO

1. Ahmed K. Elmagarmid, H. Jiang, A. Helal, A. Joshi, and M. Ahmed. *Video Database Systems - Issues, Products and Applications*. Kulwer Academic Publishers, 2001, 126pp.
2. Đỗ Trung Tuấn, Lương Xuân Cương, Khun Piseth, Nguyễn Văn Tảo. Về Xử lý dữ liệu video. *Tạp chí khoa học-Khoa học tự nhiên và công nghệ*, Đại học Quốc gia Hà Nội, T.XIX, No3(2003), p.48-56,
3. E. Saez, J.I Benavides and N.Gui. Combining Luminescence and Edge Based Metrics for Robust Temporal Video Segmentation, *IEEE Int'l. Conf. on Image Processing (ICIP2004)*, Singapore, October 24-27, 2004, p.126-132.

4. H. Aoki, S. Shimotsuji, and O. Hori. A shot classification method of selecting effective key-frame for video browsing. *In Proc. of ACM Int'l Conf. on Multimedia*, Boston, MA, November 1996, p.1-10.
5. H. H. Yu and W. Wolf. A hierarchical multiresolution video shot transition detection scheme. *Computer Vision and Image Understanding*, 75(1/2), July/August 1999, p.196–213.
6. H. J. Zhang, S. W. Smoliar, and J. Wu. Content- based video browsing tools. *In Proc. of IS&T/SPIE Con. on Multimedia Computing and Networking*, San Jose, CA, Jan.1995.
7. H. Yu and W. Wolf. A visual search system for video and image databases. *In Proc. IEEE Int'l Conf. on Multimedia Computing and Systems*, Ottawa, Canada, June 1997, p.517-524.
8. JungHwan Oh, Kien A. Hua, and Ning Liang. A content-based scene change detection and classification technique using background tracking. *In SPIE Conf. on Multimedia Computing and Networking 2000*, San Jose, CA, Jan. 2000, p.233-245.
9. JungHwan Oh Kien A. Hua. Efficient and Cost-effective Techniques for Browsing and Indexing Large Video Databases. *Computer Science Program, School of EECS University of Central Florida*, Orlando, 2001, p.2328-2362.
10. Kien A. Hua, W. Tavanapong, and J. Wang. 2psm: An efficient framework for searching video information in a limited-bandwidth environment. *ACM Multimedia Systems*, 7(5): September 1999, p.396-408.
11. Lương Mạnh Bá, Nguyễn Thanh Thủy, *Nhập môn xử lý ảnh số*, Nhà xuất bản Khoa học Kỹ thuật, Hà Nội, 1998, 175 tr.
12. Lương Xuân Cương, Đỗ Xuân Tiến, Đỗ Trung Tuấn, *Kỹ thuật nâng cao khả năng phân đoạn dữ liệu video ứng dụng trong E-Learning*, Hội thảo Quốc gia lần thứ VII: Một số vấn đề chọn lọc của công nghệ thông tin và truyền thông, Đà Nẵng, 8-2004, tr.103-109.
13. M. Abdel-Mottaleb and N. Dimitrova. Conivas: Content-based image and video access system, *In Proc. of ACM Int'l Conf. on Multimedia*, Boston, MA, November 1996, p.427-428.
14. P. Aigrain, P. Joly, and V. Longueville. Medium knowledge-based macro-segmentation of video into sequences. *In IJCAI Workshop on Intelligent Multimedia Information Retrieval*, 1995, p.5-14.
15. P. J. Burt and E. H. Adelson. The laplacian pyramid as a compact image code. *In IEEE Transactions on Communications V COM-31*, April 1983, p.532-540.
16. R. Hjelsvold and R. Midtstraum. Modeling and querying video data, *In Proc. of 20th Int'l Conf. on Very Large Database (VLDB '94)*, April 1994, p.132-140.
17. R. Lienhart. *Comparison of automatic shot boundary detection algorithms*. *In Proc. SPIE Vol. 3656, Storage and Retrieval for Image and Video Databases VII*, San Jose, CA, January 1999, p.290-301.
18. R. Zabih, J. Miller, and K. Mai. A feature- based algorithm for detecting and classifying scene breaks. *In Proc. of ACM Multimedia '95*, San Francisco, CA, 1995, p.189-200.
19. R. Weiss, A. Duda, and D. Gifford. *Content-based access to algebraic video*. *In Proc. of IEEE Int'l Conf. Multimedia Computing and Systems*, Los Alamitos, CA, 1994, p.456-468.

20. S. V. Porter, M. Mirmehdi and B. T. Thomas. *Temporal* video segmentation and classification of edit effects. In *Image and Vision Computing*, vol. 21, December 2003, p.1097–1106.
21. S. V. Porter. Video Segmentation and Indexing using Motion Estimation. *Science and Technology journal of University of Bristol*, February 2004, p.9–19.
22. Uma Srinivasan, Surya Nepal, Graham Reynolds. *Modelling High Level Semantic for Video Data Management*. CSIRO Mathematical and information Sciences, Australia, 2001, pp 144-155.
23. Y. Gong, H. Chua, and X. Guo. Image indexing and retrieval based on color histogram. In *Proc. Of Int'l Conf. Multimedia Modeling*, Singapore, Nov. 1995, pp 115-126.

VNU. JOURNAL OF SCIENCE, Nat., Sci., & Tech., T.XXII, N_o2, 2006

A TRACKING BACKGROUND-BASED AUTOMATIC TECHNIQUE FOR INDEXING VIDEO DATA

Luong Xuan Cuong⁽¹⁾, Do Xuan Tien⁽¹⁾, Do Trung Tuan⁽²⁾

⁽¹⁾ *Military Technic Academy*

⁽²⁾ *College of Science, VNU*

In general, the process of design and construction of database for video data is more complex and difficult than for other multimedia data, such as text, sound or still image. In order to represent video data and quickly retrieve them, it is necessary to arrange a technique allowing to indexing video data. Because of the richness and structure complexity of video data, the video indexing is usually challenge. In recent years, existing video data segmentation techniques based on low-level visual properties faced certain difficulty; therefore the approach of combining low-level visual properties with semantic content of video in segmentation and indexing video data is interesting. The paper aims at a shot boundary detection technique that based on tracking the camera motion through the background areas in the video frame (background tracking). We also proposed some steps to discard noise and to reduce the false segmentation to increase the accuracy of the video stream segmentation into shots. To building index set based on video content, video stream should segment into scenes that bases on results of proposed shot boundary detection technique. This technique have a more significant influence on the quality of the segmentation result when video frames are recorded by camera whilst it is moving. The experimental results have been applied to indexing for video databases of observational camera systems.