

MỘT PHƯƠNG PHÁP ĐỐI SÁNH ẢNH THỜI GIAN THỰC

Trần Thị Thanh Hải*, Eric Marchand**

Địa chỉ: * Trung tâm MICA, Trường Đại học Bách Khoa Hà Nội

** IRISA Rennes, Campus Beaulieu, Rennes, France

Emails: thanh-hai.tran@mica.edu.vn; eric.marchand@irisa.fr

Tóm tắt: Đối sánh ảnh là một bài toán cơ bản có mặt trong nhiều ứng dụng khác nhau của lĩnh vực thị giác máy tính như nhận dạng ảnh, theo dõi đối tượng, tìm kiếm, vv. Trong một số bài toán đòi hỏi tính toán thời gian thực như theo dõi giám sát (tracking), điều khiển tự động bằng hình ảnh (visual servoing), việc đối sánh ảnh phải được thực hiện nhanh và phải đạt được độ chính xác mong muốn. Trong bài báo này, chúng tôi đề xuất một phương pháp cho phép đối sánh ảnh thời gian thực. Phương pháp đối sánh ảnh mà chúng tôi đề xuất dựa trên các kỹ thuật trích chọn nhanh các điểm đặc trưng trong ảnh, ánh xạ các điểm này vào không gian đặc trưng có số chiều thấp và tìm kiếm nhanh điểm lân cận gần nhất trong không gian đặc trưng để đối sánh các điểm đặc trưng, từ đó đưa ra kết luận về các vị trí giống nhau trên hai ảnh. Phương pháp đã được thử nghiệm với một tập các ảnh khác biệt nhau về vị trí, ánh sáng, góc thu nhận ảnh từ camera. Ngoài ra chúng tôi cũng đã thử nghiệm thuật toán trong một ứng dụng điều khiển tự động bằng hình ảnh để kiểm tra độ chính xác và thời gian tính toán của thuật toán. Giải thuật hoạt động với tốc độ 10-14Hz trên máy tính Pentium IV 2.6Ghz, chứng minh khả năng tích hợp của nó vào trong các ứng dụng thời gian thực

Abstract: Image matching is a primitive problem for many computer vision applications like image recognition, tracking, indexation, image retrieval, ect. In some applications as tracking, visual servoing, image matching needs to be as **correct** and **fast** as possible. In this paper, we propose a real-time image matching method. Our main contributions are to propose: i) a fast method for keypoint detection; ii) a compact representation of keypoint in a low-dimentional feature space based on PCA technique and iii) a reliable method for matching feature points in that feature space. Experiments have been conducted with natural real images to measure the performance of the proposed method. The image matching algorithm works at 10-14Hz, shows its capability to be applied to realtime applications.

I. GIỚI THIỆU BÀI TOÁN ĐỐI SÁNH ẢNH

Đối sánh ảnh là một bài toán đã và đang thu hút được sự quan tâm của các nhà nghiên cứu và phát triển. Mỗi khi bài toán này được giải quyết, nó mở ra rất nhiều các ứng dụng hữu ích như: tìm kiếm ảnh, nhận dạng, theo dõi và phát hiện đối tượng, vv. Đối sánh hai ảnh là tìm ra những vùng giống nhau trên hai ảnh. Thông thường, để so sánh hai ảnh, người ta so sánh các phần tử cơ bản cấu thành nên nó. Đơn giản nhất là so sánh các điểm ảnh (pixel). Tuy nhiên phép so sánh này đòi hỏi nhiều thời gian tính toán và thường không đạt được độ chính xác mong muốn. Các phương pháp sau này đề xuất trích chọn các đặc trưng để biểu diễn ảnh. Khi đó bài toán đối sánh ảnh sẽ quy về bài toán so sánh các đặc trưng trích chọn [25, 24, 23, 3, 19, 27, 20, 22, 13, 28]. Các đặc trưng cho phép biểu diễn ảnh đã được nghiên cứu bao gồm đường biên, vùng ảnh, điểm đặc trưng, histogram, vv.

Bài toán đối sánh ảnh đã được đề cập vào những năm 50. Hai thập kỷ gần đây, số lượng các công trình nghiên cứu và phát triển các giải thuật đối sánh ảnh tăng một cách đáng kể. Dù vậy, đối sánh ảnh vẫn còn là một bài toán mở. Có hai vấn đề cơ bản thường được đặt ra trong bài toán đối sánh ảnh: i) làm sao có thể biểu diễn thông tin một cách hiệu quả nhằm thực hiện việc đối sánh hai ảnh một cách chính xác và nhanh nhất có thể; ii) làm thế nào để giải pháp đối sánh vẫn hoạt động hiệu quả khi có sự thay đổi của môi trường: nhiễu trong quá trình thu nhận ảnh, sự thay đổi về ánh sáng, sự che khuất, vv.

Các phương pháp đối sánh ảnh dựa trên việc đối sánh các điểm đặc trưng được đề xuất rất nhiều và đã gặt hái được những thành công đáng kể [26, 14, 20]. Tuy nhiên để đạt được một độ chính xác nhất định, các phương pháp này đều đòi hỏi rất nhiều thời gian tính toán. Trong những ứng dụng thời gian thực như theo dõi đối tượng trong ảnh (tracking), điều khiển tự động bằng hình ảnh (visual servoing), việc đưa ra một **phương pháp đối sánh ảnh thực hiện trong thời gian thực** là một công việc cần thiết [2, 4, 5, 6, 7, 8, 10, 15, 16, 17, 18].

Đóng góp cơ bản trong bài báo này là đề xuất một phương pháp so sánh ảnh có khả năng dung hòa được hai yêu cầu: độ chính xác và thời gian tính toán để nhắm tới các ứng dụng thời gian thực. Để làm được điều đó, phương pháp dựa trên ba mô đun sau:

Mô đun 1: Phát hiện nhanh các điểm đặc trưng trong ảnh: Việc phát hiện này phải được thực hiện nhanh, các điểm phát hiện phải biểu diễn các đặc trưng phân biệt cho phép đối sánh ảnh hiệu quả.

Mô đun 2: Đặc tả các vùng đặc trưng bằng các vector mô tả: Mỗi điểm đặc trưng sẽ được mô tả bởi một vector. Vector này phải có độ dài hữu hạn để giảm thời gian tìm kiếm trong không gian đặc trưng, xong cũng không được quá tối giản để vẫn chứa các thông tin phân biệt của mỗi vùng ảnh.

Mô đun 3: Đối sánh các vector mô tả để so sánh các vùng đặc trưng trong hai ảnh từ đó cho phép so sánh hai ảnh.

Trong các phần tiếp theo của bài báo này, chúng tôi trình bày cụ thể từng mô đun. Phần II trình bày một phương pháp phát hiện (mô đun 1) và biểu diễn các điểm đặc trưng trong không gian đặc trưng (mô đun 2). Trong phần III, chúng tôi trình bày một phương pháp đối sánh các điểm trong không gian đặc trưng (mô đun 3). Kết quả thử nghiệm phương pháp đối sánh ảnh thông qua việc đối sánh các vector đặc trưng được trình bày trong phần IV. Phần V đưa một số kết luận và hướng phát triển.

II. PHÁT HIỆN VÀ BIỂU DIỄN CÁC ĐẶC TRƯNG

II.1 Điểm đặc trưng: định nghĩa và phương pháp trích chọn

II.1.1 Định nghĩa điểm đặc trưng

Định nghĩa: Chúng tôi định nghĩa điểm đặc trưng trong ảnh là một điểm ảnh có **chứa nhiều thông tin** hơn các điểm ảnh lân cận. Biểu diễn ảnh thông qua các điểm đặc trưng sẽ cô đọng (compact) hơn, vì thế giảm kích thước trong không gian tìm kiếm kéo theo giảm thời gian đối sánh ảnh.

II.1.2 Phát hiện điểm đặc trưng trong ảnh

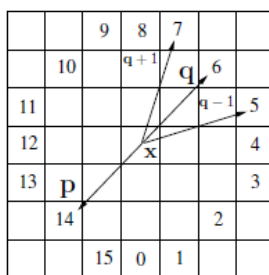
Định nghĩa trên đây cho phép hiểu một cách nôm na về điểm đặc trưng. Để phát hiện các điểm đặc trưng trong ảnh, cần phải định nghĩa điểm đặc trưng một cách toán học. Có nhiều phương pháp phát hiện điểm đặc trưng tồn tại. Harris đưa ra định nghĩa điểm đặc trưng dựa trên sai khác về độ lớn của các vector riêng của ma trận đạo hàm hướng tính tại điểm đó [24]. Lowe phát hiện các điểm cực đại hoặc cực tiểu trong không gian 3 chiều (x, y, scale) của Laplacian và gọi các điểm đặc trưng là các blob [14]. Việc phát hiện điểm đặc trưng theo các phương pháp này tương đối phức tạp và tốn nhiều thời gian tính toán. Rosten đề xuất một phương pháp cho phép phát hiện nhanh điểm đặc trưng [22]. Rosten kiểm tra xem liệu giá trị mức xám của n điểm liên tiếp trên đường tròn 16 (xem định nghĩa dưới đây) có lớn hay nhỏ hơn giá trị mức xám của điểm xem xét. Khi n lớn, việc kiểm tra này cũng mất thời gian và điều kiện này đôi khi không loại bỏ được các điểm đặc trưng nằm gần nhau.

Chúng tôi đơn giản hóa phương pháp của Rosten và đề xuất thêm một số cải tiến, nhất là trong phần loại bỏ điểm đặc trưng kề nhau. Chúng tôi đưa ra định nghĩa một điểm **không** đặc trưng trong ảnh như sau

Định nghĩa điểm không đặc trưng: Giả thiết có ảnh I mà ta muốn trích chọn các đặc trưng. x sẽ không là điểm đặc trưng nếu tồn tại hai điểm p, q sao cho:

$$\begin{cases} |I(x) - I(p)| < \epsilon \\ |I(x) - I(q)| < \epsilon \end{cases} \quad (1)$$

trong đó ϵ là một ngưỡng có giá trị nhỏ. p, q là hai điểm nằm trên đường tròn cấu thành từ 16 pixels bao quanh điểm x như trong hình vẽ 1.



Hình 1: Đường tròn 16pixels.

Điều kiện (1) sẽ cho phép loại bỏ một cách nhanh chóng các điểm đường biên và các điểm thuộc vùng có mức xám đồng đều. Để tránh phát hiện các điểm nằm trên các đường biên cong (skewed edge), chúng tôi thực hiện việc kiểm tra cả trên các điểm lân cận của p và q, cụ thể là q+1 và q-1. Việc kiểm tra bắt đầu từ một điểm nằm trên đường tròn - 16 và kết thúc ngay sau khi điều kiện (1) thỏa mãn. Như thế trong rất nhiều trường hợp, điều kiện (1) đã được thỏa mãn ngay từ lần kiểm tra đầu tiên, vì thế giải thuật loại bỏ các điểm không đặc trưng được tiến hành một cách rất nhanh chóng.

Mỗi khi các điểm trên đường biên và các điểm thuộc vùng sáng đồng đều được loại bỏ, các điểm đặc trưng nằm sát gần nhau (do nhiễu) sẽ được loại bỏ để chỉ giữ lại những điểm đặc trưng nhất. Điểm đặc trưng nhất là điểm có đáp ứng Laplacian lớn nhất so với các điểm đặc trưng lân cận.

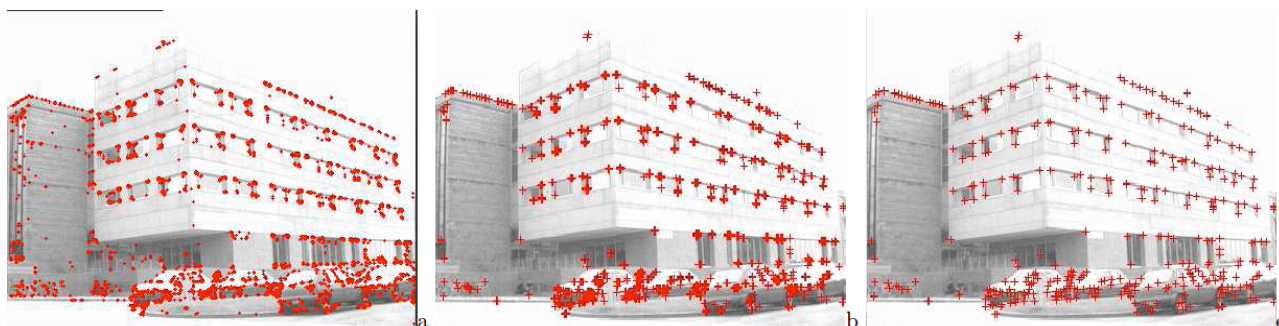
Laplacian được tính dựa trên đạo hàm bậc hai của ảnh. Để giảm nhẹ khâu tính toán, chúng tôi sử dụng một công thức tính nhanh Laplacian xấp xỉ như sau:

$$L(x) = \sum_{(p,q)} |I(p) + I(q) - I(x)| \quad (2)$$

trong đó p, q là hai điểm đối xứng qua x trên đường tròn 16.

Những điểm vượt qua được hai vòng kiểm tra là những điểm đặc trưng. Một cách trực quan, việc phát hiện các điểm đặc trưng như đề xuất ở trên là rất nhanh, nhanh hơn các phương pháp được đề xuất trong [14] (phát hiện các điểm cực trị trong không gian ba chiều x, y, scale). Tuy nhiên do chỉ làm với ảnh gốc (ảnh có độ phân giải đầy đủ), các điểm đặc trưng phát hiện được sẽ biến đổi khi kích thước đối tượng ảnh thay đổi. Trong các bài toán như theo dõi phát hiện đối tượng, kích thước đối tượng ảnh thường biến đổi không nhiều, vì thế các điểm đặc trưng phát hiện như phương pháp đề xuất vẫn có giá trị cho bài toán đối sánh ảnh.

Hình 2 minh họa kết quả thu nhận được từ phương pháp phát hiện điểm đặc trưng từ một ảnh. Điều kiện loại bỏ điểm không đặc trưng của chúng tôi đơn giản hơn nhiều so với điều kiện đề xuất bởi Rosten *et al.* [22], tuy nhiên kết quả phát hiện điểm đặc trưng phần lớn là giống nhau (xem hình 2a và 2b). Ngoài ra, do áp dụng thêm điều kiện về Laplacian, một số lượng lớn các điểm đặc trưng nằm gần nhau được loại bỏ, chỉ giữ lại các điểm đặc trưng nhất (hình 2c)



Hình 2: a) Điểm đặc trưng phát hiện theo phương pháp của [22]. b) Điểm đặc trưng phát hiện theo phương pháp đề xuất c) Các điểm đặc trưng còn lại sau khi đã loại bỏ các điểm lân cận.

II.1.3 Hướng của điểm đặc trưng

Trong phần II.1.2, chúng tôi đã trình bày một phương pháp cho phép phát hiện và định vị nhanh các điểm đặc trưng trên ảnh. Trong phần này, chúng tôi đề xuất gán cho mỗi điểm đặc trưng một hướng, gọi là

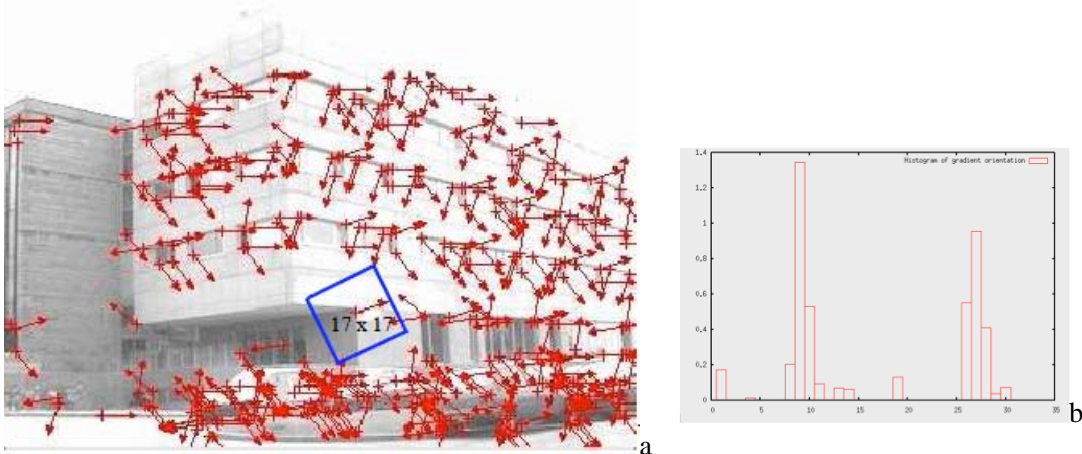
hướng đặc trưng của điểm. Việc gán hướng đặc trưng cho điểm ảnh sẽ cho phép mô tả đặc trưng độc lập với hướng của đối tượng ảnh (do có sự thay đổi về góc chụp).

Để xác định hướng đặc trưng cho một điểm ảnh (x, y) , [14] đề xuất tính histogram hướng các gradient tính trong miền lân cận của điểm xem xét. Thông thường, tất cả các điểm nằm trong đường tròn bán kính r , tâm (x, y) sẽ được coi là lân cận- r của điểm (x, y) , ta gọi là $\mathcal{C}(x, y, r)$. Với mỗi điểm (x_i, y_i) trong đường

tròn $\mathcal{C}(x, y, r)$, ta tính hướng vector gradient tại điểm này. Độ lớn của vector gradient được coi như trọng

số để tính histogram. Trong bài báo này, chúng tôi thử nghiệm với $r = 7$ (pixels). Histogram có 36 bin phủ không gian 360 độ. Hình 3 bên phải minh họa histogram được tính trong lân cận của một điểm ảnh. Trục hoành của histogram biểu diễn số bin, trục tung là số điểm ảnh có đánh trọng số theo biên độ của gradient có cùng hướng tương ứng. Điểm cực đại trên histogram cho phép xác định một hướng chính (canonical orientation), ta gán nó như là hướng đặc trưng của điểm xem xét.

Phương pháp tính hướng đặc trưng mà chúng tôi đề xuất phức tạp hơn một chút so với một phương pháp đề xuất bởi [13] do việc đánh trọng số của hướng theo độ lớn của vector gradient và việc tính histogram hướng. Tuy nhiên với cách tính này, hướng xác định được sẽ ít bị tác động bởi nhiễu hơn. Hình 3 minh họa các hướng đặc trưng tính được trên mỗi điểm đặc trưng. Ta có thể thấy những điểm đặc trưng biểu diễn các góc của cửa sổ của tòa nhà sẽ được phân biệt bởi các hướng đặc trưng.



Hình 3: a) Các điểm đặc trưng phát hiện với các hướng đặc trưng (vector màu đỏ). b) Histogram hướng gradient tính tại một điểm ảnh. Trục hoành biểu diễn số lượng bins của histogram. Trục tung biểu diễn số điểm ảnh đánh trọng số theo biên độ của vector gradient của các điểm ảnh.

II.2 Biểu diễn các điểm đặc trưng - Không gian đặc trưng

II.2.1 Xây dựng không gian đặc trưng

Chúng ta có một tập các điểm đặc trưng cho phép biểu diễn ảnh, mỗi điểm đặc trưng được gán một hướng đặc trưng. Để so sánh hai điểm đặc trưng trong ảnh, mỗi điểm phải được mô tả bởi một vector đặc trưng. Các mô tả nên độc lập với các biến đổi về hướng, ánh sáng và vị trí chụp ảnh. Có nhiều phương pháp để mô tả các đặc trưng. Cách đơn giản nhất là sử dụng chính mức xám của vùng ảnh bao xung quanh điểm đặc trưng và việc so sánh hai điểm đặc trưng quy về tính sự tương quan (correlation measure) giữa hai vùng mức xám. Nếu vùng mức xám xem xét có kích thước $N \times N$, không gian đặc trưng sẽ có kích thước N^2 chiều và việc tính toán và tìm kiếm trong không gian có số chiều lớn sẽ rất tốn kém về thời gian và bộ nhớ [1, 11].

Để giảm số chiều của không gian đặc trưng, chúng tôi đề xuất sử dụng phương pháp phân tích thành các thành phần chính (PCA - Principal Component Analysis). Phương pháp này cho phép chuyển không gian

N^2 chiều về không gian có số chiều nhỏ hơn để nâng cao hiệu suất đối sánh ảnh. Ngoài ra, do mức xám rất dễ bị nhiễu và phụ thuộc nhiều vào cường độ ánh sáng, chúng tôi đề nghị thay mức xám bằng biên độ của vector gradient chuẩn hóa (normalized gradient of the image). Trong [12], Y. Ke cũng đã chỉ ra rằng việc sử dụng PCA cho phép tăng tốc giải thuật đối sánh các điểm đặc trưng trong ảnh một cách đáng kể

Không gian đặc trưng được xây dựng thông qua các bước sau

1) Lấy ra một tập ảnh để học. Tập này phải đủ lớn và các ảnh nên phân biệt để có thể học được nhiều mẫu khác nhau.

2) Trên mỗi ảnh, phát hiện các điểm đặc trưng, tính hướng đặc trưng cho các điểm đặc trưng, xác định vùng phủ tương ứng với điểm đặc trưng (theo hướng đặc trưng). Vùng phủ là một vùng hình vuông có kích thước $N \times N$ và có hướng là hướng đặc trưng (xem một ví dụ hình 3a). Gọi số lượng vùng đặc trưng thu được từ tập ảnh học là M .

3) Với mỗi vùng phủ P_i , tính gradient của các pixel trong vùng phủ: $G(x, y) = \Delta I_x^2(x, y) + \Delta I_y^2(x, y)$ trong đó $\Delta I_x^2, \Delta I_y^2$ là gradient của ảnh theo các hướng x, y . Mỗi điểm đặc trưng được biểu diễn bởi một vector Γ_i có $(N-2)^2$ thành phần (do không tính các điểm nằm trên cạnh của hình vuông kích thước $N \times N$).

4) Chuẩn hóa vector gradient Γ_i :

4a) Tính vector gradient trung bình:
$$\psi = \frac{1}{M} \sum_{i=1}^M \Gamma_i$$

4b) Chuẩn hóa vector đặc trưng:

$$\Omega_i = \Gamma_i - \psi$$

5) Tính ma trận hiệp phương sai:

$$C = \frac{1}{M} \sum_{i=1}^M \Omega_i \Omega_i^T$$

C là một ma trận có kích thước $(N-2)^2 \times (N-2)^2$.

6) Tính các giá trị riêng và vector riêng của ma trận C bằng phương pháp phân tích thành các giá trị riêng (Singular Value Decomposition - SVD). Gọi e_1, e_2, \dots, e_K là các K giá trị riêng lớn nhất tìm được trong tập các giá trị riêng

7) Giữ lại K vector riêng để làm hệ cơ sở cho không gian đặc trưng. K thông thường nhỏ hơn rất nhiều so với N^2 . Gọi các vector đặc trưng là v_1, v_2, \dots, v_K .

Lưu ý rằng giai đoạn học để xây dựng hệ cơ sở của không gian đặc trưng được làm offline, độc lập với lúc chạy chương trình.

II.2.2 Biểu diễn trong không gian đặc trưng

Sau khi không gian đặc trưng đã được xây dựng, mỗi điểm đặc trưng có thể được ánh xạ sang không gian đặc trưng thông qua các bước sau:

1) Chuẩn hóa vector biên độ gradient $\Omega = \Gamma - \psi$

2) Ánh xạ vector Ω sang không gian đặc trưng:

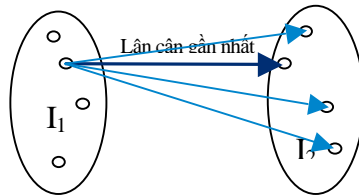
$$\tilde{\Omega} = \sum_{i=1}^K w_i v_i$$

3) Biểu diễn Ω như một vector K phần tử $\tilde{\Omega} = (w_1, w_2, \dots, w_K)$

Mỗi điểm đặc trưng, trước đó được biểu diễn bởi một vector Γ - $(N-2)^2$ chiều (khi $N = 41$, số chiều là $39 \times 39 = 1521$ chiều), giờ chỉ còn được biểu diễn bởi vector $\tilde{\Omega}$ - K chiều, với $K = 20$ trong trường hợp thử nghiệm của chúng tôi. Số chiều của không gian đặc trưng giảm một cách đáng kể, cho phép giảm lượng thời gian tính toán và tìm kiếm trong không gian này.

III. Đối sánh các vector đặc trưng

Để so sánh hai ảnh, ta so sánh các điểm đặc trưng phát hiện được từ hai ảnh. Với mỗi ảnh tập các điểm đặc trưng được phát hiện bởi phương pháp được trình bày trong phần II.1.2. Mỗi điểm sau đó được gán một hướng đặc trưng (phần II.1.2) và được mô tả bởi một vector đặc trưng (phần II.2.2). Việc đối sánh các hai tập hợp điểm đặc trưng quy về bài toán tìm điểm lân cận gần nhất (nearest neighbour) của mỗi điểm đặc trưng (xem hình 4).



Hình 4 : Đối sánh hai ảnh quy về đối sánh hai tập hợp điểm đặc trưng trong không gian đặc trưng

Có hai vấn đề cần được quan tâm: i) tổ chức tập hợp điểm cho phép việc tìm kiếm lân cận một cách hiệu quả; ii) việc so sánh phải đạt được một độ chính xác nhất định.

Một phương pháp đề xuất bởi D. Mount cho phép tìm kiếm nhanh các điểm lân cận được sử dụng [21]. ANN là viết tắt của **A**pproximative **N**earest **N**eighbour. Nó cho phép tổ chức dữ liệu dưới dạng *kd-tree*. Việc tìm kiếm lân cận gần nhất mang tính xấp xỉ trên *kd-tree*. Vì thế thời gian tìm kiếm giảm đáng kể. Tuy nhiên việc tìm kiếm lân cận xấp xỉ đôi khi đưa ra một lân cận không hẳn là gần nhất. Để tránh các lỗi có thể xảy ra, chúng tôi đề xuất sử dụng thêm một tiêu chí liên quan đến việc tính lân cận gần thứ hai (second nearest neighbour) như đề xuất trong [14]. Cụ thể là hai điểm trong không gian đặc trưng được coi là giống nhau nếu như khoảng cách Euclide giữa hai điểm là nhỏ nhất và tỷ số giữa khoảng cách gần nhất và khoảng cách gần nhì phải nhỏ hơn một ngưỡng cho trước. Khoảng cách giữa hai điểm trong không gian đặc trưng là khoảng cách Euclide được tính như sau

$$\|\Omega_1 - \Omega_2\| = \sum_{i=1}^K \frac{1}{e_i} (w_{1_i} - w_{2_i})^2$$

trong đó e_i là các giá trị riêng của ma trận hiệp tương quan tính ở trên.

Ngoài ra, giải thuật RANSAC cũng được áp dụng để loại bỏ các lỗi đối sánh. Chi tiết về RANSAC được trình bày trong [9].

IV. MỘT SỐ KẾT QUẢ THỰC NGHIỆM

Các bước thực hiện đối sánh hai ảnh, I_1, I_2 :

- 1) Phát hiện các điểm đặc trưng trên mỗi ảnh, I_1, I_2
- 2) Với mỗi ảnh I_i , tập các điểm đặc trưng sẽ được ánh xạ sang không gian đặc trưng. Gọi S_i là tập các điểm đặc trưng của ảnh I_i trong không gian đặc trưng.
- 3) Sử dụng phương pháp ANN và phương pháp RANSAC tìm các cặp điểm giống nhau giữa hai tập S_1 và S_2

Dữ liệu sử dụng để thử nghiệm thuật toán: Bộ dữ liệu ảnh có được bằng cách chụp ảnh các đồ vật trong phòng thí nghiệm, các ảnh trong khuôn viên đại học. Số ảnh sử dụng cho việc học là 30 ảnh. Giải thuật được thử nghiệm trong hai trường hợp nhằm kiểm tra:

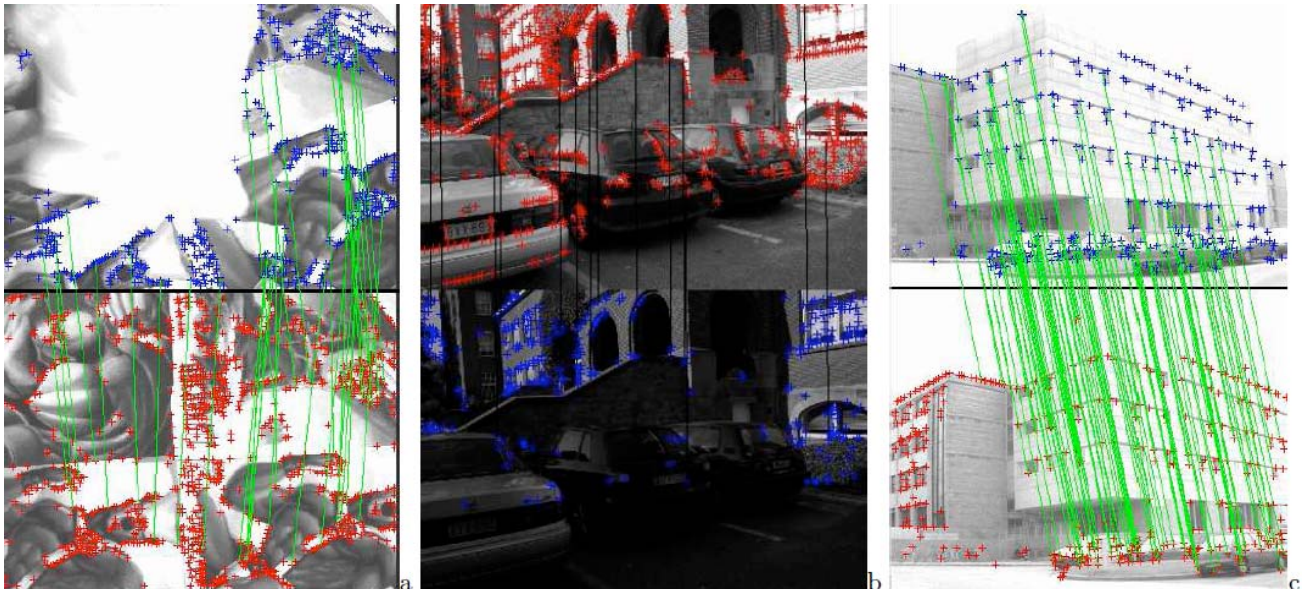
1) *Độ chính xác của giải thuật đối với sự thay đổi của môi trường:* Thử nghiệm với các cặp ảnh khi có sự dịch chuyển của camera, sự thay đổi của ánh sáng, sự che khuất.

2) *Thời gian tính toán của giải thuật:* Thử nghiệm để xác định xem thuật toán có đủ nhanh để ứng dụng vào các bài toán cần **đối sánh ảnh thời gian thực** (như là điều khiển tự động bằng hình ảnh, theo dõi đối tượng).

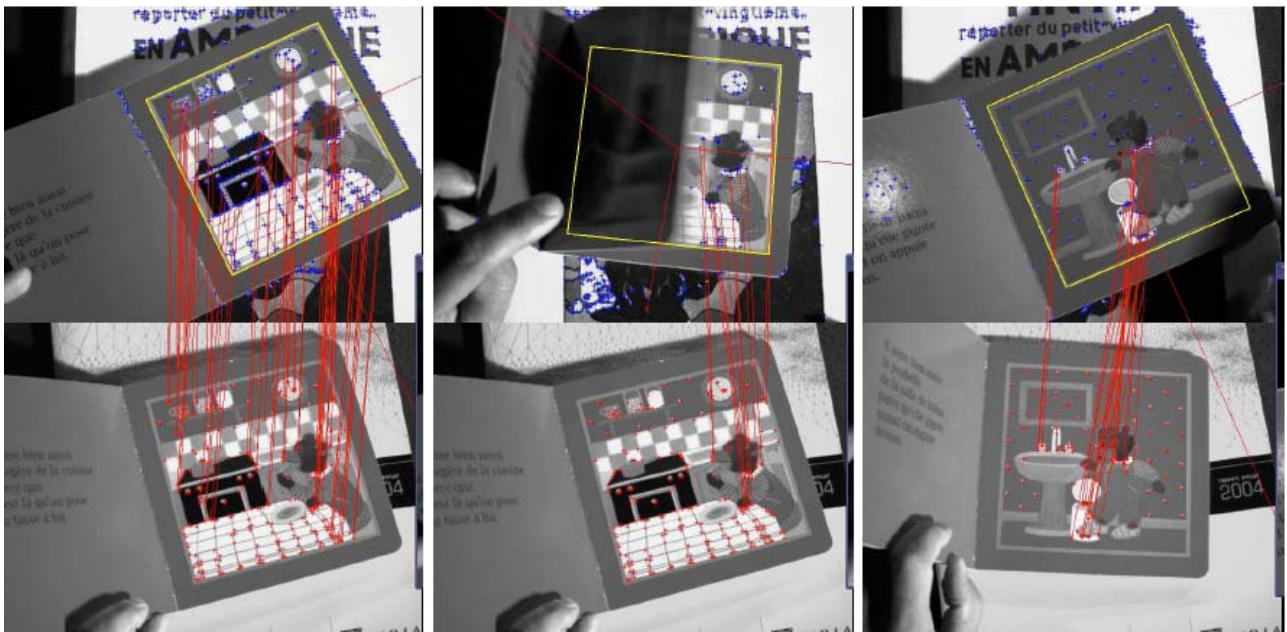
Một số tham số sử dụng khi thử nghiệm thuật toán: Số chiều của không gian đặc trưng: $K = 20$. Kích thước của vùng bao phủ $N = 17$. Đây là các tham số được lựa chọn một cách thực nghiệm. Thực tế, thuật

toán được thử nghiệm với các giá trị khác nhau của N, K, bộ N = 17, K = 20 là bộ tham số cho độ chính xác/thời gian tính toán tốt nhất trên bộ thử nghiệm của chúng tôi.

Hình 5, 6, 7 cung cấp một số kết quả đối sánh ảnh. Trên hình, các điểm đặc trưng được đánh dấu bằng các dấu +, hai điểm giống nhau trên hai ảnh được nối bằng một đường thẳng. Ta có thể thấy rằng, giải thuật vẫn cho kết quả tốt khi ánh sáng thay đổi (hình 5b), khi camera di chuyển (hình 5c), khi có sự che khuất (hình 5a) hay trong trường hai ảnh khác nhau về góc quay camera (hình 6).



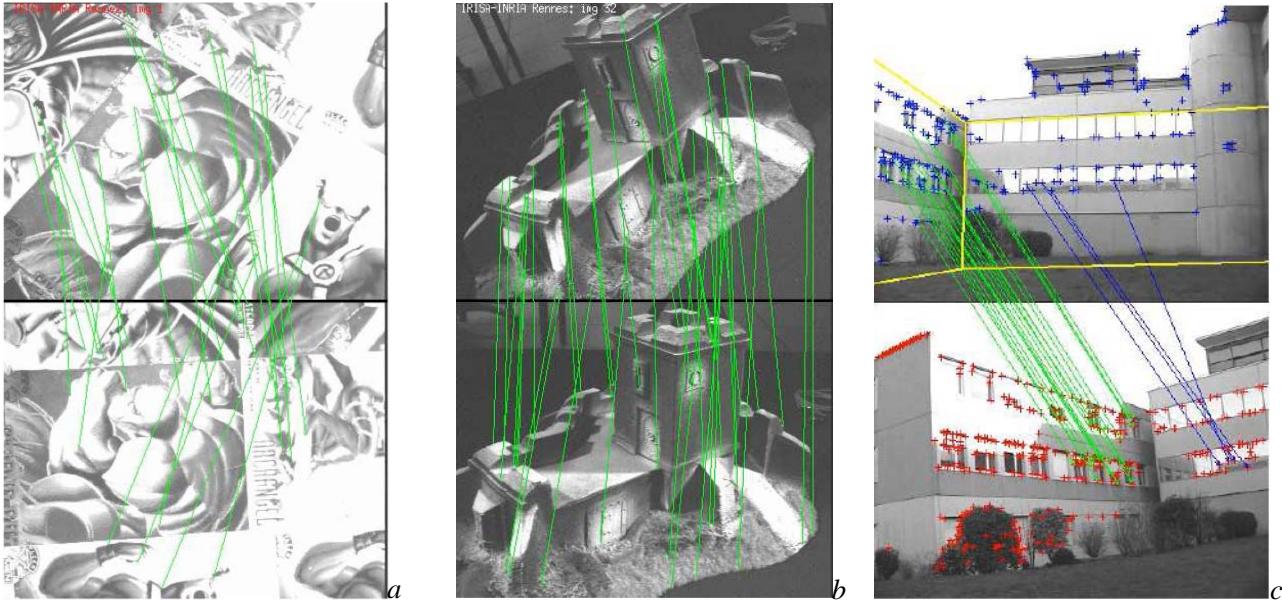
Hình 5: Đối sánh các điểm đặc trưng giữa hai ảnh khác nhau về góc nghiêng và vị trí thu nhận ảnh



Hình 6 : Một số kết quả đối sánh hai ảnh của thu nhận được từ những vị trí khác nhau, trong điều kiện ánh sáng khác nhau.

Hình 7 được trích ra từ việc thử nghiệm giải thuật trong một ứng dụng sử dụng hình ảnh để điều khiển robot (robot có gắn camera). Ứng dụng có thể tóm tắt như sau: Người ta muốn điều khiển robot (có gắn camera) di chuyển từ một vị trí A sang một vị trí B bằng hình ảnh. Để làm được việc này, ảnh quan sát được I_A từ vị trí A sẽ được đối sánh với ảnh I_B ở vị trí B. Sự sai khác giữa hai ảnh cho phép điều khiển robot chuyển từ vị trí A đến một vị trí T gần vị trí B hơn vị trí A. Tại vị trí T, ta lại so sánh ảnh quan sát được I_T với I_B và dịch chuyển robot từ vị trí T đến một vị trí gần B hơn vị trí T. Cứ làm như vậy cho đến khi I_T trùng với I_B có nghĩa là robot đã di chuyển đến vị trí mong muốn B. Lý thuyết về việc điều khiển di chuyển robot dựa trên sự sai khác hình ảnh được trình bày cụ thể trong các bài báo [5, 6, 8].

Thử nghiệm cho thấy số lượng cặp điểm ảnh giống nhau tìm thấy trên hai ảnh I_A và I_B luôn đủ để cho phép thuật toán xác định phép biến đổi giữa hai ảnh (transformation) hoạt động tốt (tối thiểu là 4 hay 8 cặp điểm, tùy thuộc vào scene quan sát là 2D hay 3D). Lỗi đối sánh thường gặp phải khi các điểm đặc trưng phát hiện được ở các vị trí khác nhau trên ảnh biểu thị những đối tượng giống nhau (ví dụ các góc cửa của các toà nhà hình 7c). Để tránh các lỗi này, giải thuật tìm kiếm điểm lân cận gần nhất nên tính đến quan hệ không gian giữa các điểm ảnh.



Hình 7: Thử nghiệm giải thuật đối sánh các trong ứng dụng điều khiển tự động bằng hình ảnh. Trong mỗi hình, phía bên trên là một ảnh I_A trong lúc chạy chương trình, bên dưới là ảnh tham chiếu (I_B). c) Minh họa một ví dụ về lỗi đối sánh do các điểm đặc trubiểu diễn các đối tượng trên thực tế khá giống nhau.

Bảng I cung cấp một số thông tin về thời gian tính toán trong mỗi giai đoạn của giải thuật đề xuất. Giải thuật hoạt động với tốc độ 10Hz trên máy Pentium IV, 2.6GHz khi các ảnh tương đối khác nhau về vị trí thu nhận ảnh cũng như điều kiện ánh sáng. Trong trường hợp đối sánh ảnh trong ứng dụng điều khiển robot bằng hình ảnh, giải thuật có lúc đạt đến tốc độ 14Hz.

Các công đoạn trong bài toán đối sánh	(ms)
Phát hiện điểm đặc trưng	10
Mô tả điểm đặc trưng (PCA)	30
Đối sánh các điểm đặc trưng (giải thuật ANN)	20
Loại bỏ các đáp án sai (giải thuật RANSAC)	30

Bảng 1: Thời gian tính toán của mỗi bước trong giải thuật đối sánh ảnh

V. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Trong bài báo này, chúng tôi đề xuất một phương pháp đối sánh ảnh nhanh, có khả năng ứng dụng vào các bài toán đòi hỏi việc đối sánh phải được thực hiện trong thời gian thực. Đóng góp chính của chúng tôi là đã đưa ra: 1) một phương pháp phát hiện nhanh các điểm đặc trưng trong ảnh; 2) một cách thức xác định hướng đặc trưng cho ảnh và việc biểu diễn điểm đặc trưng trong không gian đặc trưng sử dụng kỹ thuật PCA; 3) một đề xuất sử dụng giải thuật ANN để thực hiện tìm kiếm nhanh các lân cận gần nhất.

Phương pháp đã được thử nghiệm với các ảnh tự nhiên và trong bài toán điều khiển tự động bằng hình ảnh. Giải thuật hoạt động với tốc độ 10-14Hz trên máy tính Pentium IV, 2.6GHz phù hợp với các ứng dụng thời gian thực (giải thuật đề xuất bởi D. Lowe chạy với tốc độ 3Hz trên máy Pentium IV, 2GHz). Độ chính xác của giải thuật đã được kiểm nghiệm thông qua việc hội tụ của các ứng dụng điều khiển tự động bằng hình ảnh. Hướng phát triển tiếp theo của chúng tôi là loại bỏ một số lỗi đối sánh bằng cách xem xét

và đưa vào các ràng buộc về quan hệ không gian giữa các điểm đặc trưng. Sau đó chúng tôi sẽ thử nghiệm giải thuật đối sánh ảnh trong một hệ thống theo dõi giám sát (video surveillance).

TÀI LIỆU THAM KHẢO

- [1] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Y. Wu. An optimal algorithm for approximate nearest neighbor searching. *Journal of ACM*, (45): 891.923, 1998.
- [2] S. Benhimane and E. Malis. Homography-based 2d visual tracking and servoing. *Int. Journal of Computer Vision*, 2007. Special IJCV/IJRR issue on vision for robots.
- [3] S. Boukir, P. Bouthemy, F. Chaumette, and D. Juvin. A local method for contour matching and its parallel implementation. *Machine Vision and Application*, 10(5/6):321.330, April 1998.
- [4] A.I. Comport, E. Marchand, and F. Chaumette. Robust model-based tracking for robot vision. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'04*, volume 1, pages 692.697, Sendai, Japan, September 2004. (extended version version published in *Advanced Robotics*, 16(10):1097.1013, december 2005 (special issue on Selected paper from IROS'04).
- [5] A.I. Comport, E. Marchand, and F. Chaumette. Statistically robust 2d visual servoing. *IEEE Trans. on Robotics*, 22(2):415.421, apr 2006.
- [6] A.I. Comport, E. Marchand, M. Pressigout, and F. Chaumette. Realtime markerless tracking for augmented reality: the virtual visual servoing framework. *IEEE Trans. on Visualization and Computer Graphics*, 12(4):615.628, July 2006.
- [7] T. Drummond and R. Cipolla. Real-time visual tracking of complex structures. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(7):932.946, July 2002.
- [8] B. Espiau, F. Chaumette, and P. Rives. A new approach to visual servoing in robotics. *IEEE Trans. on Robotics and Automation*, 8(3):313.326, June 1992.
- [9] N. Fischler and R.C. Bolles. Random sample consensus: A paradigm for model _tting with application to image analysis and automated cartography. *Communication of the ACM*, 24(6):381.395, June 1981.
- [10] S. Hutchinson, G. Hager, and P. Corke. A tutorial on visual servo control. *IEEE Trans. on Robotics and Automation*, 12(5):651.670, October 1996.
- [11] P. Indyk. *Nearest Neighbors in high dimensional spaces*. Handbook of Discrete and Computational Geometry, CRC Press, 2004.
- [12] Y. Ke and R. Sukthankar. Pca-sift: A more distinctive representation for local image descriptor. In *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, 2004.
- [13] V. Lepetit and P. Fua. Keypoint recognition using randomized trees. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28(9):1465.1479, September 2006.
- [14] D. Lowe. Distinctive image features from scale-invariant keypoints. *Int. Journal of Computer Vision*, 60(2):91.110, 2004.
- [15] E. Malis, F. Chaumette, and S. Boudet. 2 1/2 D visual servoing with respect to unknown objects through a new estimation scheme of camera displacement. *Int. Journal of Computer Vision*, 37(1):79. 97, June 2000.
- [16] E. Marchand. Visp: A software environment for eye-in-hand visual servoing. In *IEEE Int. Conf. on Robotics and Automation, ICRA'99*, volume 4, pages 3224.3229, Detroit, Michigan, Mai 1999.
- [17] E. Marchand and F. Chaumette. Feature tracking for visual servoing purposes. *Robotics and Autonomous Systems*, 52(1):53.70, June 2005. special issue on .Advances in Robot Vision., D. Kragic, H. Christensen (Eds.).
- [18] E. Marchand, F. Spindler, and F. Chaumette. ViSP for visual servoing: a generic software platform with a wide class of robot control skills. *IEEE Robotics and Automation Magazine*, 12(4):40.52, December 2005. Special Issue on "Software Packages for Vision-Based Control of Motion", P. Oh, D. Burschka (Eds.).
- [19] K. Mikolajczyk and C. Schmid. Indexing based on scale invariant interest point. In *IEEE, Int. Conf. on Computer Vision, ICCV'01*, pages 525.531, Seattle, 2001.
- [20] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A comparison of affine region detectors. *Int. Journal of Computer Vision*, 65:43.72, 2005.
- [21] D.M. Mount. *ANN programming Manual*. Dpt. of Computer Science and Inst. of Advanced Computer Studies, Univ. of Maryland, College Park, Maryland, 2005.
- [22] E. Rosten and T. Drummond. Fusing points and lines for high performance tracking. In *IEEE Int. Conf. on Computer Vision*, pages 1508.1515, Beijing, China, 2005.

- [23] J. Shi and C. Tomasi. Good features to track. In *IEEE Int. Conf. on Computer Vision and Pattern Recognition, CVPR'94*, pages 593-600, Seattle, Washington, June 1994.
- [24] C. Harris and M.J. Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, pages 147-152, 1988.
- [25] H. Moravec. Obstacle avoidance and navigation in the real world by a seeing robot rover. *Technical Report CMU-RI-TR-3*, Carnegie-Mellon University, Robotics Institute, 1980.
- [26] C. Schmid, R. Mohr, and C. Bauckhage. Evaluation of interest point detectors. *International Journal of Computer Vision*, 37(2):151-172, June 2000.
- [27], H. Tran, Etude des lignes d'intérêt naturelles pour la représentation d'objets en vision par ordinateur, *Ph.D dissertation (in french)*, INPG, France, Mars 2006
- [28], H. Tran, A. Lux, A method for ridge extraction, In *Proc. of Asean Conference on computer vision ACCV05*, Volume 1, Pages 96-111, Jeju, Korea, 2004