

PHƯƠNG PHÁP PHÂN LOẠI NHANH PHƯƠNG TIỆN GIAO THÔNG DỰA TRÊN ĐƯỜNG VIÊN

Nguyễn Văn Căn, Nguyễn Tiên Hưng, Dương Phú Thuần, Nguyễn Đăng Tiến

Trường Đại học Kỹ thuật – Hậu cần CAND

cannv@truongt36.edu.vn, hungnt.bca@gmail.com, thuanktt36@gmail.com, dangtienT36@gmail.com

TÓM TẮT - Mục đích của bài viết này là trình bày phương pháp biểu diễn đường viền của đối tượng trong ảnh nhị phân bằng vector số phức; phân tích một số tính chất của vector số phức để áp dụng cho bài toán nhận dạng; so sánh mẫu trên cơ sở đường viền; xây dựng lược đồ chung cho việc nhận dạng và phân loại đối tượng bằng phương pháp đường viền vector. Chúng tôi đã tiến hành thử nghiệm nhận dạng và phân loại các tập ảnh có mức độ phức tạp về xe máy, xe ô tô cho kết quả chính xác và tốc độ tính toán nhanh.

Từ khóa: Contour Analysis, Car Recognition, complex vector, Vector Contour

I. GIỚI THIỆU

Phân loại ảnh là một bài toán đã và đang thu hút được sự quan tâm của các nhà nghiên cứu và phát triển, được ứng dụng rộng rãi nhiều ứng dụng hữu ích như: tìm kiếm ảnh, nhận dạng, theo dõi và phát hiện đối tượng... Trong giám sát đối tượng chuyển động từ video chẳng hạn như giám sát phương tiện giao thông, thì phân loại ảnh là bài toán kế tiếp sau bài toán phát hiện đối tượng chuyển động [15]. Cách tiếp cận phổ biến của bài toán phân loại ảnh là đối sánh ảnh, tức là tìm ra những vùng giống nhau trên hai ảnh dựa trên những điểm đặc trưng. Khi đó bài toán đối sánh ảnh sẽ quy về bài toán so sánh các đặc trưng trích chọn [3], [7], [8], [9], [13]. Các đặc trưng cho phép biểu diễn ảnh đã được nghiên cứu bao gồm đường biên vùng ảnh, điểm ảnh đặc trưng, lược đồ xám,...

Có hai vấn đề cơ bản thường 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 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,...

Các phương pháp đối sánh ảnh dựa trên đối sánh các điểm đặc trưng được đề xuất rất nhiều và đạt được sự thành công đáng kể [9], [12], [14]. 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 giám sát giao thông tự động, 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], [10], [11].

Nghiên cứu này trình bày và giới thiệu một phương pháp tiếp cận phân loại mạnh mẽ đối với ô tô và xe máy trong ảnh giao thông. Đóng góp cơ bản trong bài báo này là đề xuất một phương pháp biểu diễn đường viền đối tượng bằng đường viền vector trên trường số phức [1], phân tích và áp dụng tính chất của một số phép toán trên trường số phức, áp dụng cho việc nhận dạng và phân loại đối tượng.

Trong các phần tiếp theo của bài báo này, chúng tôi trình bày chi tiết các nội dung về phương pháp tiếp cận của chúng tôi. Mục II, trình bày phương pháp biểu diễn đường viền đối tượng trên trường số phức và áp dụng cho thuật toán phân loại đường viền. Mục III, trình bày các kết quả thực nghiệm và bàn luận đánh giá. Mục IV là kết luận và hướng phát triển.

II. PHƯƠNG PHÁP BIỂU DIỄN ĐƯỜNG VIÊN ĐỐI TƯỢNG BẰNG SỐ PHỨC

A. Biểu diễn đường viền theo vector

Phân tích đường viền (Contour Analysis - CA) cho phép mô tả, lưu trữ, so sánh và tìm ra các đối tượng biểu diễn dưới dạng đường viền. Đường viền chứa thông tin cần thiết về hình dạng đối tượng. Không quan tâm nhiều đến các điểm bên trong của đối tượng, nhưng quan tâm nhiều về thể hiện đường viền bên ngoài thì cho phép chuyển về không gian đường viền, từ đó cho phép giảm thời gian tính toán và độ phức tạp tính toán. CA cho phép giải quyết hiệu quả các bài toán cơ bản của nhận dạng mẫu – biến đổi, quay và tỷ lệ của ảnh đối tượng. Phương pháp CA là bất biến đối với phép biến đổi.

Đường viền là đường bao của đối tượng, thường là các điểm ảnh, phân tách đối tượng với nền. Trong các hệ thống thị giác máy tính, một vài định dạng mã hóa đường viền được sử dụng như mã hóa Freeman, mã hóa 2 chiều, mã hóa đa giác thường được sử dụng nhất. Nhưng tất cả những định dạng này thường không sử dụng trong CA.

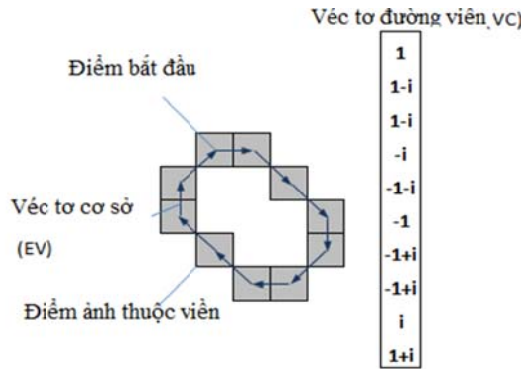
Định nghĩa 1. Đường viền Vector [16]

Đường viền Vector là đường viền được biểu diễn bằng một dãy các số phức. Trên một đường viền, điểm bắt đầu cần được xác định. Tiếp theo, đường viền sẽ được quét (xoay theo chiều kim đồng hồ), và mỗi vector được biểu diễn bằng một số phức $a+ib$. Với a là điểm trên trục x , b là điểm trên trục y . Các điểm được biểu diễn kế tiếp nhau (Hình 1).

Do tính chất vật lý của các đối tượng ba chiều, đường viền của chúng luôn khép kín và không tự giao nhau. Nó cho phép xác định rõ ràng một duyệt qua một đường viền (đề trong một hướng - hoặc ngược chiều kim đồng hồ). Vector cuối cùng của một đường viền luôn luôn dẫn đến điểm khởi đầu.

Mỗi vector của một đường viên được gọi vector cơ sở EV (*Elementary Vector*). Và chuỗi giá trị các số phức gọi là đường viên vector (VC).

VC ký hiệu bằng chữ cái Γ , và EV ký hiệu là γ . Như vậy, đường viên vector Γ có độ dài k xác định là $\Gamma=(\gamma_0, \gamma_1, \dots, \gamma_{k-1})$.



Hình 1. Biểu diễn đường viên bằng vector số phức

Khi biểu diễn đường viên bằng VC thì thao tác trên đường viên như là thao tác trên vector số phức có chứa nhiều đặc tính toán học hơn là các mã biểu diễn khác. Về cơ bản, mã số phức là gần với mã hai chiều khi mà đường viên được định nghĩa phổ biến bằng vector cơ bản trong tọa độ 2 chiều. Nhưng sự khác biệt giữa thao tác tỷ lệ của các vector đối với số phức là khác nhau. Trường hợp này cũng ưu tiên cho các phương pháp CA.

Định nghĩa 2. Tích vô hướng của đường viên [16]

Hai số phức của 2 đường viên Γ và N , tích vô hướng của nó là

$$\eta = (\Gamma, N) = \sum_{n=0}^{k-1} (\gamma_n, \nu_n) \tag{1}$$

với k – kích thước của VC, γ_n là vector cơ sở của đường viên Γ , ν_n là vector cơ sở của đường viên N .

Tích vô hướng của các số phức được tính bằng công thức:

$$(a+ib, c+id) = (a+ib)(c+id) = ac+bd+i(bc+ad) \tag{2}$$

Chú ý rằng trong một tích vô hướng CA chỉ có một đường viên vector VC của chiều đồng nhất. Do vậy một số vectors cơ sở trong đường viên là trùng nhau.

Tích vô hướng của các vector thường và tích vô hướng của các số phức là khác nhau: Nếu ta nhân một vector cơ sở EV như một vector đơn, tích vô hướng của chúng sẽ là:

$$((a,b), (c,d)) = ac + bd \tag{3}$$

So sánh công thức này với công thức (2) chú ý rằng:

- Kết quả tích vô hướng của các vectors là một số thực. Và kết quả tích của các số phức là một số phức.
- Phần thực của tích vô hướng của các số phức trùng với tích vô hướng của các vector phù hợp. Tích số phức bao gồm tích vô hướng vector.

Theo đại số tuyến tính, để xác định được chính xác chiều vật lý và các đặc tính của tích vô hướng. Tích vô hướng bằng với tích của độ dài vector của góc cosin ở giữa. Tức là 2 vector vuông góc sẽ luôn có tích vô hướng bằng 0. Ngược lại, tích của vector thẳng hàng sẽ cho giá trị tích vô hướng tối đa.

Những đặc tính này của tích được sử dụng như một cách để đo lường độ gần của các vector. Nếu tích càng lớn, góc càng nhỏ giữa các vector, thì các vector này sẽ càng gần nhau. Với những vector vuông góc, tích này bằng 0, và hơn nữa có thể nhận giá trị âm cho những vector có hướng khác nhau theo cách này. Tích vô hướng (1) cũng có các đặc tính tương tự.

Mệnh đề 1. Đặc tính của đường viên

1. Cho $\Gamma=(\gamma_0, \gamma_1, \dots, \gamma_{k-1})$, tổng các EV là S, ta có: $S= \text{SUM}(\gamma_0, \gamma_1, \dots, \gamma_{k-1})=0$, S là tầm thường khi $k=1$.
2. Cho I_1 là ảnh nguồn, Γ_1 thuộc I_1 ; I_2 là I_1 tịnh tiến theo một phương p, Γ_2 thuộc I_2 ; thì $\Gamma_2 = \Gamma_1$.
3. Quay ảnh theo một góc độ nào đó tương đương với quay mỗi EV của đường viên trên cùng góc độ đó.
4. Việc thay đổi điểm khởi đầu tiến hành theo vòng tròn VC. Vì các EV được mã hóa liên quan đến các điểm trước đó, điều này rõ ràng là thay đổi điểm khởi đầu, trình tự của một EV sẽ là như nhau, nhưng EV đầu tiên sẽ là bắt đầu từ điểm khởi đầu.
5. Thay đổi tỷ lệ ảnh nguồn có thể được coi là phép nhân của mỗi EV của đường viên với một hệ số tỷ lệ.

Bắt nguồn từ cách biểu diễn đường viền bằng số phức, giả sử có 8 hướng cơ bản của đường viền, bắt đầu từ đỉnh cao nhất, quay theo chiều kim đồng hồ, chúng ta có vector số phức $\{1, 1-i, -i, -1, -1+i, 1+i\}$. Rõ ràng ta có tổng vector này bằng 0. Bằng trực quan hóa hình học, dễ dàng chứng minh được đặc tính 2,3 và các đặc tính còn lại.

Định nghĩa 3. Tích vô hướng chuẩn hóa [16]

Tích vô hướng chuẩn hóa (NSP) của hai đường viền vector Γ và N được xác định bằng công thức:

$$\eta = \frac{(\Gamma, N)}{|\Gamma||N|} \tag{4}$$

$|\Gamma|$ và $|N|$ - Chuẩn (chiều dài) của đường viền được tính bằng công thức:

$$|\Gamma| = (\sum_{n=0}^{k-1} |\gamma_n|^2)^{\frac{1}{2}} \tag{5}$$

Mệnh đề 2. NSP cực đại [16]

Dạng chuẩn của NSP đạt giá trị lớn nhất, chỉ khi đường viền Γ giống với đường viền N , nhưng xoay theo một số góc và tỉ lệ bởi một hệ số xác định.

Thật vậy, do NSP trong không gian phức cũng là một số phức, tính đồng nhất là giá trị lớn nhất có thể của chuẩn NSP (Theo bất đẳng thức Cauchy-Bunyakovsky Schwarz): $|ab| \leq |a||b|$, và chỉ có thể đạt được giá trị này khi và chỉ khi:

$$\Gamma = \mu N \tag{6}$$

Với μ – Là một số phức tùy chọn.

Khi nhân các số phức, độ dài của chúng được nhân, và các góc được cộng lại với nhau. Đường viền μN giống với đường viền N , ngoại trừ xoay và tỉ lệ. Tỉ lệ và hướng xoay được định nghĩa bởi một số phức μ .





Ví dụ, ta coi một tích vô hướng của một đường viền với chính nó, nhưng xoay một góc nhất định:

Nếu đếm một NSP của một vector với chính nó, ta sẽ nhận giá trị NSP=1. Nếu xoay một đường viền 90 độ, sẽ nhận giá trị NSP=0+i, xoay một góc 180 độ nhận giá trị NSP=-1. Do vậy, phần thực của một NSP sẽ cho ta cosin của một góc giữa các đường viền, và chuẩn của NSP luôn bằng 1.

Tương tự, nếu ta tăng một VC với một hệ số thực nào đó, ta sẽ nhận giá trị NSP=1 (Có thể thấy ở công thức (4))

Theo mệnh đề 1 ta có tích vô hướng NSP thì bất biến đối với phép dịch chuyển, xoay và tỷ lệ của đường viền.

Do vậy, chuẩn của tích vô hướng chuẩn hóa của đường viền cho giá trị đồng nhất chỉ khi 2 đường viền này bằng nhau cả về độ xoay và tỉ lệ. Mặt khác chuẩn của NSP sẽ bị giới hạn ít đồng nhất hơn.

	NSP	Re(NSP)=cos(a)	NSP
	1	1	1
	i	0	1
	-1	-1	1
	-i	0	1

Hình 2. Tích vô hướng chuẩn trên đường viền

Chuẩn NSP là bất biến trong phép chuyển dịch, xoay và tỉ lệ của đường viền. Nếu 2 đường viền tương đồng nhau, NSP của chúng sẽ luôn cho giá trị đồng nhất, không phụ thuộc vào vị trí của đường viền, hay độ xoay của góc và tỉ lệ của chúng. Tương tự, nếu các đường viền khác nhau, NSP sẽ bị giới hạn nhỏ hơn 1, và độc lập trong không gian, độ xoay và tỉ lệ.

Chuẩn của NSP là phép đo độ gần của các đường viền. Chuẩn đưa ra giá trị đo của một đường viền và tham số của một NSP (bằng atan(b/a)) – đưa ra một góc xoay của đường viền.

Định nghĩa 4. Hàm tương quan của 2 đường viền [16]

Hàm tương quan ICF của 2 đường viền:

$$\tau(m) = (\Gamma, N^{(m)}), m=0, \dots, k-1 \tag{7}$$

Trong đó $N^{(m)}$: là đường viền nhận từ N bởi vòng dịch chuyển bởi vector cơ sở EV của chính nó trên m phần tử. Độ lớn của chuẩn tối đa giữa các giá trị của ICF:

$$\tau_{max} = \max \left(\frac{\tau(m)}{|\Gamma||N|} \right), m = 0, \dots, p - 1$$

Từ việc xác định một NSP và một ICF, có thể thấy τ_{max} là giá trị đo lường sự giống nhau của 2 đường viền, bất biến với việc dịch chuyển, xoay, tỉ lệ và sự thay đổi điểm khởi đầu.

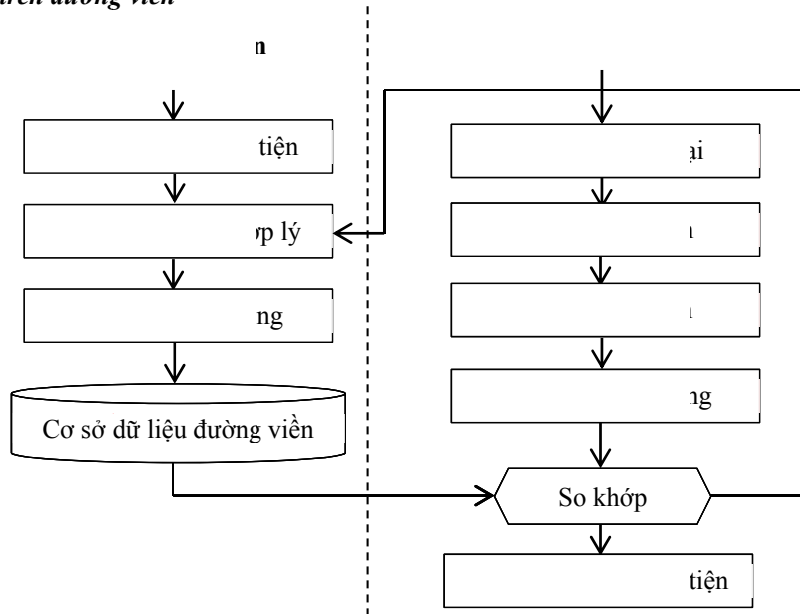
Định nghĩa 5. Hàm tự tương quan (ACF) [16]

ACF là hàm tương quan khi $N=\Gamma$. Tích vô hướng của đường viền trên chính nó tại các dịch chuyển khác nhau của điểm khởi đầu được tính theo công thức:

$$v(m) = (\Gamma, \Gamma^{(m)}), m=0, \dots, k-1 \tag{8}$$

Nếu đường viền có nhiều đối xứng để xoay thì ACF của chúng sẽ có đối xứng tương tự nhau. Có thể coi một đường viền ACF là một đặc tính của hình dạng của đường viền.

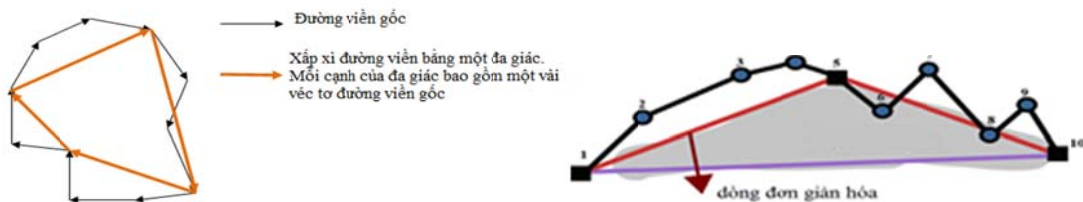
B. Thuật toán phân loại dựa trên đường viền



Hình 3. Sơ đồ khối tổng quát phân loại theo đường viền

Phân loại dựa trên đường viền được chia thành 2 pha: Pha huấn luyện và Pha phân loại. Sơ đồ khái quát được minh họa trong hình 3. Các khối tăng cường ảnh, tìm đường viền áp dụng các thuật toán đã có trong xử lý ảnh. Điều quan trọng ở đây là tính toán đặc trưng của đường viền. Tuy nhiên để chuẩn hóa đường viền ở cả pha huấn luyện cũng như pha phân loại cần phải thực hiện phép cân bằng hóa đường viền hay còn gọi là xấp xỉ độ dài đường viền.

Như đã trình bày ở trên về phương pháp CA, cần xác định độ dài của đường viền. Trong một bức ảnh thực, đường viền có độ dài bất kỳ. Do đó việc tìm kiếm và so sánh đường viền, tất cả chúng cần có số đỉnh đồng nhất. Quá trình này gọi là quá trình cân bằng. Đầu tiên sẽ cố định số đỉnh của VC chuẩn (ở pha huấn luyện) sẽ sử dụng trong hệ thống nhận diện, ký hiệu là k. Sau đó với mỗi đường viền A mới được tạo ra, ta tạo một đường viền vector N với độ dài k. Và có thể có 2 biến thể, hoặc đường viền ban đầu có số đỉnh lớn hơn số k hoặc nhỏ hơn số k. Nếu một đường viền ban đầu cần thiết để được sắp xếp bởi EV, ta sẽ quan tâm tới thành phần N như tổng của các EV. Hình 4 dưới đây chỉ ra ý nghĩa của sự cân bằng hóa hay còn gọi là đơn giản hóa đường viền.



Hình 4. Xấp xỉ hóa đường viền

Trong hình 4, bên trái mô tả đường viền gốc (đường màu đen, mỏng) và đường viền sau khi đơn giản hóa (đường màu cam, đậm); bên phải mô tả hình tượng phương pháp đơn giản hóa đường cong theo thuật toán Douglas Peucker [17]. Với d là số đỉnh của đường viền sau khi đơn giản hóa, hình dáng đường viền, chu vi hay diện tích của đường viền mới sẽ đảm bảo phải xấp xỉ với đường viền ban đầu. Điều này được thực hiện và chứng minh trong công trình [17] về ứng dụng của thuật toán Douglas Peucker.

Pha huấn luyện. Chuẩn bị cơ sở dữ liệu Template (Thực hiện thủ công).

Input: Hình ảnh (Image), ngưỡng đường viền (ThresoldContour), số đỉnh đường viền (d)

Output: Template (Tập mẫu)

Nội dung thuật toán:

1. Chuẩn hóa về độ phân giải mong muốn
Image \leftarrow ChuanHoaDoPhanGiai(Image)
2. Tìm các đường viền
n \leftarrow SoDuongVien(Image)
Contour(i) \leftarrow TimDuongVien(Image), i=1..n
3. Chuẩn hóa các đường viền
For i=1 to n { //Áp dụng thuật toán đơn giản hóa
Contour(i) \leftarrow DonGianHoa(Contour(i), d)
For i=1 to n { //Tính chu vi
ChuVi(i) \leftarrow TinhChuVi(Contour(i))
For i=1 to n { //Loại đường viền quá nhỏ
j \leftarrow i
if ChuVi(i) > ThresoldContour then { Countour(j) \leftarrow Countour(i); j++}
m=j; //m là số đường viền sau khi loại bỏ đường viền nhỏ
4. Tìm đặc trưng các đường viền
For i=1 to m { //Tính toán tham số đặc trưng
CV(i) \leftarrow Chuvi(Contour(i))
DT(i) \leftarrow DienTich(Contour(i))
for j=1 to d { //Tính góc tại các đỉnh
goc(i,j) = TinhGoc(Contour(i),j)
}
}
}
}
5. Cập nhật Template
For i=1 to m {
Template \leftarrow CV(i), DT(i)
For j=1 to d
Template \leftarrow goc(i,j)
}
}
6. Return Template

Pha phân loại. Nhận dạng trên các tập ảnh thực tế (Thực hiện online – thời gian thực):

Input: Video/Ảnh

Output: ImageCountour (Ảnh có chứa đường viền phương tiện)

Nội dung thuật toán:

1. Thu nhận và Xử lý sơ bộ ảnh (Làm mịn, lọc nhiễu, tăng độ tương phản)
Image \leftarrow Capture(Video)
Image \leftarrow ChuanHoaAnh(Image)
2. Tìm các đường viền
n \leftarrow SoDuongVien(Image)
Contour(i) \leftarrow TimDuongVien(Image), i=1..n
3. Chuẩn hóa các đường viền
For i=1 to n { //Áp dụng thuật toán đơn giản hóa
Contour(i) \leftarrow DonGianHoa(Contour(i), d)
For i=1 to n { //Tính chu vi
ChuVi(i) \leftarrow TinhChuVi(Contour(i))
For i=1 to n { //Loại đường viền quá nhỏ
j \leftarrow i
if ChuVi(i) > ThresoldContour then { Countour(j) \leftarrow Countour(i); j++}
m=j; //m là số đường viền sau khi loại bỏ đường viền nhỏ
4. Tìm đặc trưng các đường viền
For i=1 to m { //Tính toán tham số đặc trưng
CV(i) \leftarrow Chuvi(Contour(i))
DT(i) \leftarrow DienTich(Contour(i))
for j=1 to d { //Tính góc tại các đỉnh

- $goc(i,j)=TinhGoc(Contour(i,j))$
- }
5. So sánh đường viền với Template.
 For each đường viền phát hiện {
 Chọn vùng chỉ vi để đối sánh
 Chọn vùng diện tích để đối sánh
 So sánh sự đồng dạng giữa 2 đường viền}
6. Return

Mệnh đề 3. Độ phức tạp ước tính của thuật toán phân loại dựa trên VC

Giả sử bức ảnh đã được nhị phân hóa có kích thước $n*n$ pixels, tìm đường viền bằng cách duyệt qua toàn bộ ảnh 2 chiều, do đó độ phức tạp tương ứng là $O(n^2)$.

Giả sử k là độ dài đường viền, t là số các đường viền có trong ảnh. Thuật toán so sánh đường viền có độ phức tạp ước tính là $O(n^2k^2t)$.

Thuật toán này thực là phức tạp, đặc biệt với độ dài của một số k lớn. Tuy nhiên trong thực tế, đường viền ban đầu là một số k nhỏ. Ta thêm vào phép nội suy và coi là giá trị xấp xỉ.

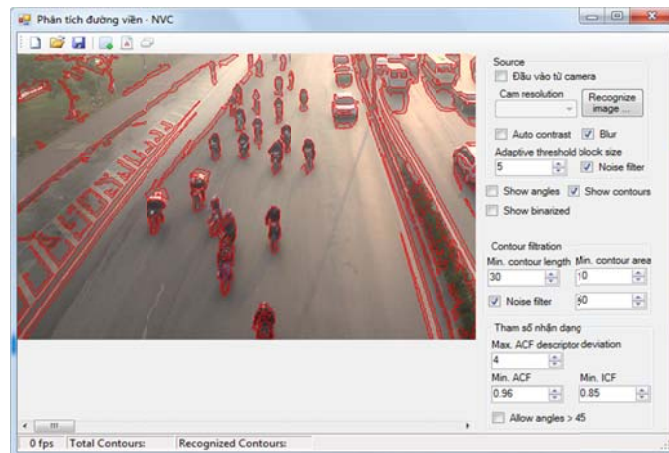
Vấn đề là cần chọn giá trị k . Vậy giá trị nào là tối ưu nhất? Câu trả lời cho vấn đề này là hoàn toàn được định nghĩa bởi một vùng ứng dụng cụ thể. Mặt khác, độ dài lớn k có nghĩa là tiêu tốn một lượng phí lớn vào việc đánh giá. Còn giá trị k nhỏ cần ít thông tin, độ chính xác của việc nhận dạng cũng giảm và việc nhận dạng nhiều tăng lên.

III. THỰC NGHIỆM VÀ BÀN LUẬN

Mặc dù những giới hạn trên, phương pháp CA vẫn được chú ý với những sự thực hiện đơn giản và tốc độ cao. Trong việc thể hiện các đối tượng trong nền tương phản và ít nhiễu, CA sẽ thực hiện tốt việc nhận dạng.

Việc kiểm nghiệm phương pháp CA bằng cách kiểm thử cho ra kết quả 90% phương tiện được nhận diện. Đó là một kết quả khá tốt. Và kết quả này chứa một số lượng các ảnh đọc xấu của các phương tiện. Do đó CA xử lý 249 ảnh với các kích thước khác nhau (từ $400*400$ tới $1280*960$) trong vòng 30 giây. So sánh với kết quả trong [14], độ chính xác đã được cải thiện và tốc độ nhanh hơn. Bên cạnh việc nhận dạng các ảnh tĩnh, thực hiện tốc độ cao của CA cho phép xử lý video trong chế độ thời gian thực.

Thực nghiệm được thiết kế trên 2 dự án. Dự án đầu tiên là ContourAnalysis - thực hiện các chức năng cơ bản của phân tích đường viền, tạo đường viền, tích vô hướng của đường viền, cân bằng hóa, đánh giá ICF và ACF, so sánh và tìm kiếm các mẫu.



Hình 5. Hệ thống thực nghiệm

Lớp Contour tạo ra và lưu trữ các đường viền. Chứa các vận hành cơ bản cho đường viền, tích vô hướng, tỉ lệ, cân bằng hóa, chuẩn hóa và đánh giá quang phổ, đánh giá ACF, ICF.

Lớp Template được dùng để tạo các mẫu cơ bản. Lớp này lưu trữ các đường viền, bộ mô tả ACF, tham số tuyến tính của đường viền ban đầu, chuẩn của đường viền. Đồng thời lớp mẫu này có tên được dùng trong giá trị nhận dạng.

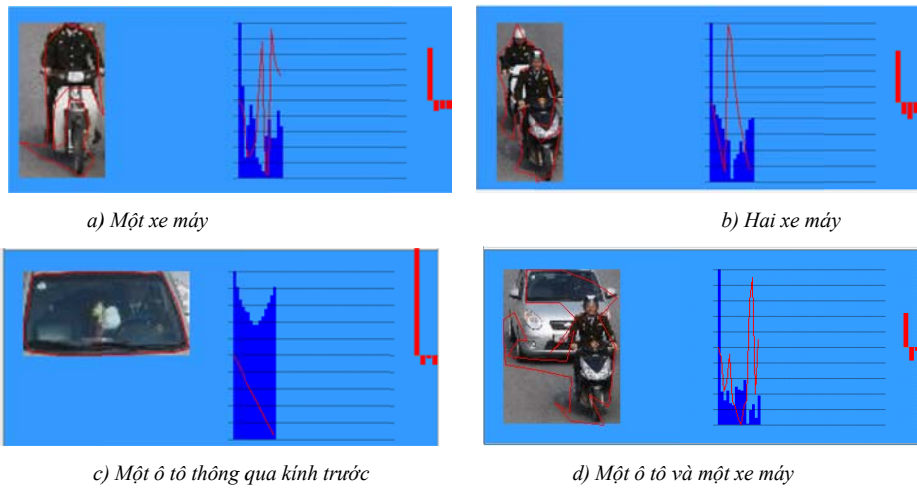
Lớp TemplateFinder thực hiện việc tìm kiếm nhanh các mẫu của đường viền cho sẵn. Kết quả của sự vận hành của lớp này là FoundTemplateDesc, chứa đường viền ban đầu và mẫu được phát hiện bởi đường viền cho sẵn. Bên cạnh đó FoundTemplateDesc chứa tỉ lệ tương tự, góc xoay và tỉ lệ đường viền, các liên hệ mẫu.

Dự án thứ 2, ContourAnalysisProcessing, chứa các phương pháp để xử lý sơ bộ ảnh, chọn đường viền, lọc và nhận dạng. Đồng thời nó cũng chứa các công cụ để tự động tạo ra các mẫu cho việc nhận dạng các loại xe, sử dụng thư viện OpenCV (EmguCV .NET wrapper) để xử lý ảnh.

Lớp ImageProcessor được sử dụng để xử lý ảnh. Đồng thời lưu trữ mẫu. Phương thức ProcessImage() ImageProcessor nhận ảnh ở đầu vào. Kết quả xử lý liệt kê các đường viền được phát hiện ImageProcessor.samples và liệt kê danh sách đường viền được nhận diện FoundTemplates. Lớp ImageProcessor chứa cài đặt cho việc tìm kiếm đường viền.

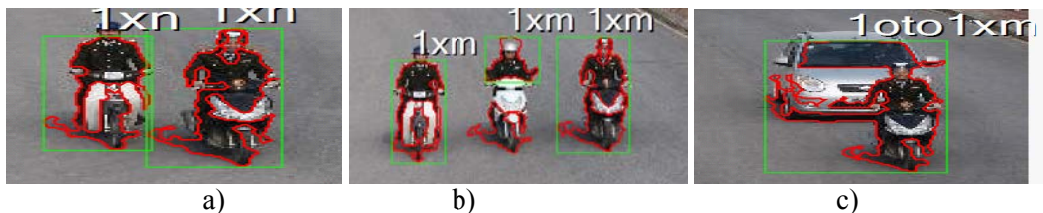
Các tham số trong thực nghiệm: Kích thước ngưỡng khối thích nghi (Adaptive Thresold block size) = 5 pixel; Độ dài đường viền nhỏ nhất (Min contour length) = 30 pixel; Diện tích đường viền nhỏ nhất (Min contour area) = 10 pixel; Độ phân giải ảnh đầu vào: 640 x 480 pixel.

Cơ sở dữ liệu mẫu: Thực nghiệm đã thực hiện tạo ra một cơ sở dữ liệu tập mẫu các đường viền gồm 50 mẫu đường viền khác nhau từ các hình dạng 1 xe máy, 1 xe ô tô, 2 xe máy, 2 ô tô, 1 xe máy 1 ô tô. Đường viền mẫu của xe máy được tập trung lưu trữ toàn bộ hình dạng đường viền bao quanh xe máy. Thêm một số mẫu về đường viền phần nửa trên người đi xe máy. Đối với ô tô, tập mẫu tạo ra bằng cách lưu trữ khung đường viền của kính trước ô tô.



Hình 6. Ví dụ về tập mẫu để so sánh

Trong quá trình nhận dạng, gán nhãn cho đường viền phát hiện được tương ứng là 1xm (một xe máy), 1oto (một ô tô), 2xm (hai xe máy), 2oto (hai ô tô), 1xm1oto (một xe máy kèm một ô tô).



a) Nhận dạng được 2 đường viền, gán nhãn cho mỗi xe một nhãn là 1xm.
 b) Nhận dạng được 3 xe máy. Trong đó có 2 xe nhận dạng theo đường viền toàn bộ, 1 xe nhận dạng theo phần trên xe.
 c) Nhận dạng được một ô tô và 1 xe máy đi gần nhau.

Hình 7. Ví dụ kết quả nhận dạng xe ô tô và xe máy

Phương pháp được thực nghiệm với các ảnh tự nhiên và trong bài toán đếm phương tiện giao thông, so sánh và nhận dạng ra nhanh một xe máy, 1 ô tô, 2 xe máy dính liền nhau, 2 ô tô dính liền nhau, 1 ô tô và 1 xe máy dính liền nhau trong 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. Độ chính xác của giải thuật đã được kiểm nghiệm thông qua việc đối sánh ảnh giao thông chụp tại một số cung đường ở Việt Nam.

Trong quá trình thực nghiệm, chúng tôi thấy rằng còn có một số những hạn chế của phương pháp, cần phải nghiên cứu tiếp tục như sau:

- Hạn chế đầu tiên có liên quan tới vấn đề lựa chọn đường viền trên ảnh. Đường viền được giới hạn với một cấu trúc rời rạc nhất định. Tuy nhiên các đối tượng này được thể hiện trong môi trường thực có thể xảy ra những trường hợp: Có một số lượng lớn các đường viền liên quan và không liên quan đến đối tượng nhận dạng; Đối tượng trong ảnh không thể có đường biên rõ ràng, có thể nhận diện dựa trên độ sáng hoặc màu sắc so với nền, có thể bị nhiễu... Tất cả những nhân tố trên dẫn tới việc đường viền không thể được lựa chọn hoặc được chọn không chính xác, không tương đồng với đường bao của đối tượng. Mặt khác, trên thực tế một đối tượng có thể có nhiều đường viền tồn tại, chẳng hạn như một ô tô: đường viền kính trước, đường viền gương quan sát hậu, đường viền hai đèn trước xe,... Việc tổ hợp các đường viền này trong cùng một đối tượng cũng cần phải đặt ra.

- Hạn chế thứ hai, gây phức tạp cho phương pháp CA có liên quan tới các quy tắc của phân tích đường viền. Phương pháp CA giả sử rằng đường viền mô tả khung của các đối tượng và không quan tâm đến các phần phía sau hoặc các phần nhìn thấy không hoàn toàn của đối tượng. Do đó CA có độ ổn định kém trong các trường hợp nhiễu, không hỗ trợ sự giao cắt hoặc các phần nhìn thấy của đối tượng.

IV. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Trong bài báo này, đề xuất phương pháp phân loại dựa trên đường viền, có khả năng ứng dụng vào các bài toán đối sánh ảnh đòi hỏi thời gian thực. Đóng góp chính: 1) một phương pháp biểu diễn đường viền đối tượng bằng đường viền vector trên trường số phức; 2) một số phép toán xử lý trên đường viền vector số phức, xấp xỉ đường viền số phức bằng đa giác; 3) sử dụng giải thuật phân tích đường viền, tìm kiếm độ dài đường viền để thực hiện tìm kiếm và đối sánh hai đường viền.

Hướng phát triển tiếp theo là: 1) loại bỏ nhanh một số lỗi bằng cách xem xét kích thước chiều dài, chiều rộng đối tượng, ngưỡng xấp xỉ hình dạng đường viền mịn hơn, sau đó thử nghiệm giải thuật đối sánh ảnh trong một hệ thống giám sát giao thông thời gian thực; 2) xem xét đến trường hợp một đối tượng có nhiều đường viền để tăng độ chính xác và khả năng nhận dạng đối tượng đa dạng hơn.

V. TÀI LIỆU THAM KHẢO

- [1] Kasana, H. S. (2005), *Complex Variables: Theory And Applications (2nd ed.)*, PHI Learning Pvt. Ltd, p. 14, ISBN 81-203-2641-5.
- [2] S. Benhimane and E. Malis (2007), *Homography based 2d visual tracking and servoing*. Int. Journal of Computer Vision, 2007. Special UJCV/IJRR issue on vision for robots.
- [3] H. Tran, A. Lux (2004), *A method for ridge extraction*. In Proc. of Asean Conference on computer vision ACCV05, Volume 1, Pages 96-111, Jeju, Korea.
- [4] A. I. Comport, E. Marchand, and F. Chaumette (2004), *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 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 (2006), *Statistically robust 2d visual servoing*. IEEE Trans. on Robotics, 22(2):415-421.
- [6] A. I. Comport, E. Marchand, M. Pressigout, and F. Chaumette (2006), *Realtime markerless tracking for augmented reality: the virtual visual servoing framework*. IEEE Trans. on Visualization and Computer Graphics, 12(4):615-628.
- [7] H. Tran (2006), *Etude des lignes d'intérêt naturelles pour la représentation d'objets en vision par ordinateur*. Ph.D dissertation (in french).
- [8] E. Rosten and T. Drummond (2005), *Fusing points and lines for high performance tracking*. In IEEE Int. Conf. on Computer Vision, pages 1508-1515, Beijing, China.
- [9] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool (2005), *A comparison of affine region detectors*. Int. Journal of Computer Vision, 65:43-72.
- [10] E. Marchand, F. Spindler, and F. Chaumette (2005), *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.).
- [11] E. Marchand and F. Chaumette (2005), *Feature tracking for visual servoing purposes*. Robotics and Autonomous Systems, 52(1):53-70. Special issue on Advances in Robot Vision., D. Kragic, H. Christensen (Eds.).
- [12] D. Lowe (2004), *Distinctive image features from scale-invariant keypoints*. Int. Journal of Computer Vision, 60(2):91-110.
- [13] V. Lepetit and P. Fua (2006), *Keypoint recognition using randomized trees*. IEEE Trans. on Pattern Analysis and Machine Intelligence, 28(9):1465-1479.
- [14] A. J. Lipton and at al (1998), *Moving target classification and tracking from real-time video*. In Proc. of Workshop Applications of Computer Vision, pages 129-136.

Bảng kết quả của nghiên cứu trong [14]

Target	Tot.	Unclass.	Misclass	Correct
Vehilce	319	10.7%	2.5%	86.8%
Human	291	11.0%	6.2%	82.8%
False	4			

- [15] Ngô Quốc Tạo, Nguyễn Văn Căn, Huỳnh Văn Huy (2014), Nghiên cứu phương pháp đếm xe ô tô sử dụng mô hình hỗn hợp Gaussian và luồng quang học. Kỷ yếu hội nghị Fair 2014.

- [16] Rohit Kolar, Akshay Thakar, Muzaffar Shabad (2014), *Image Segmentation for Text Recognition using Boundary Analysis*. International Journal of Emerging Technology and Advanced Engineering. ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 4, Issue 2, February 2014) 294.
- [17] Wu, Shin-Ting and Márquez (2004), Mercedes R. G (2004), A non-self-intersection Douglas-Peucker Algorithm, Proceedings of Sibgrapi. © 2004 IEEE.

FAST METHOD CLASSIFICATION BASED ON TRANSPORTATION CONTOURS

Nguyen Van Can, Nguyen Tien Hung, Duong Phu Thuan, Nguyen Dang Tien

***ABSTRACT** - The purpose of this paper is: first, the present method of performing contour of the object in a binary image using complex numbers vector. Monday analyzing some properties of the complex vector to apply to problem identification, comparison on the basis of contour form. Construction general scheme for the identification and classification of objects by means of vector contours. We have conducted experiments to identify and classify the photo with the complexity of motorcycles and cars for accurate results and fast calculation speed.*