

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

-----o0o-----

**TÌM HIỂU PHƯƠNG PHÁP PHÂN ĐOẠN TÁCH
CÁC NÉT CỦA CHỮ VIẾT TAY HẠN CHẾ**

ĐỒ ÁN TỐT NGHIỆP HỆ ĐẠI HỌC CHÍNH QUY

Ngành: Công nghệ thông tin

HẢI PHÒNG - 2012

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

-----o0o-----

**TÌM HIỂU PHƯƠNG PHÁP PHÂN ĐOẠN TÁCH
CÁC NÉT CỦA CHỮ VIẾT TAY HẠN CHẾ**

ĐỒ ÁN TỐT NGHIỆP HỆ ĐẠI HỌC CHÍNH QUY

Ngành: Công nghệ thông tin

Sinh viên thực hiện: Trần Văn Tuấn

Giáo viên hướng dẫn: PGS.TS. Ngô Quốc Tạo

Mã số sinh viên : 111159

HẢI PHÒNG – 2012

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh phúc
-----oOo-----

NHIỆM VỤ THIẾT KẾ TỐT NGHIỆP

Sinh viên: Trần Văn Tuấn

Mã số: 111159

Lớp: CT1201

Nghành: Công nghệ thông tin

Tên đề tài: Tìm hiểu phương pháp phân đoạn tách nét của chữ viết tay hạn chế

NHIỆM VỤ ĐỀ TÀI

1. Nội dung và các yêu cầu cần giải quyết trong nhiệm vụ đề tài tốt nghiệp

a. Nội dung:

- Tìm hiểu các giai đoạn của hệ thống nhận dạng chữ viết tay
- Đi sâu vào phương pháp tách các nét của chữ viết tay offline
- Nghiên cứu công cụ tiền xử lý trợ giúp cho tách các nét chữ viết tay
- Tìm hiểu về phương pháp làm mảnh ảnh
- Cài đặt chương trình thử nghiệm tách các nét viết

b. Các yêu cầu cần giải quyết

- Hiểu được một số khái niệm cơ bản về xử lý ảnh
- Tìm hiểu về phương pháp cộng, trừ ảnh
- Tìm hiểu phương pháp chuyển ảnh thành ảnh đa cấp xám
- Tìm hiểu phương pháp phân đoạn ảnh, lấy ngưỡng ảnh
- Tìm hiểu về phương pháp làm mảnh ảnh
- Tìm các điểm ngã 3 trong ảnh
- Tách các đoạn chữ viết tay
- Cài đặt chương trình thử nghiệm

2. Các số liệu cần thiết để thiết kế, tính toán

3. Địa điểm thực tập

CÁN BỘ HƯỚNG DẪN ĐỀ TÀI TỐT NGHIỆP

Người hướng dẫn thứ nhất:

Họ và tên: Ngô Quốc Tạo

Học hàm, học vị: PGS .TS

Cơ quan công tác: Viện Công Nghệ Thông Tin, Viện Khoa Học Và Công Nghệ Việt Nam

Nội dung hướng dẫn:

Tìm hiểu phương pháp phân đoạn tách nét chữ viết tay hạn chế

Người hướng dẫn thứ hai:

Họ và tên:

Học vị, học hàm:

Cơ quan công tác:

Nội dung hướng dẫn:

.....

Đề tài tốt nghiệp được giao ngày.....tháng..... năm 2012

Yêu cầu phải hoàn thành trước ngàytháng.....năm 2012

Đã nhận nhiệm vụ: Đ.T.T.N

Sinh viên

Đã nhận nhiệm vụ: Đ.T.T.N

Cán bộ hướng dẫn Đ.T.T.N

Hải Phòng, ngày....tháng....năm 2012

HIỆU TRƯỞNG

GS .TS .NGŨT Trần Hữu Nghị

PHẦN NHẬN XÉT TÓM TẮT CỦA CÁN BỘ HƯỚNG DẪN

1. Tinh thần thái độ của sinh viên trong quá trình làm đề tài tốt nghiệp:

.....
.....
.....
.....
.....
.....
.....
.....
.....

2. Đánh giá chất lượng của đề tài tốt nghiệp (so với nội dung yêu cầu đã đề ra trong đề tài tốt nghiệp)

.....
.....
.....
.....
.....
.....
.....

3. Cho điểm của cán bộ hướng dẫn:

(Điểm ghi bằng số và chữ)

.....
.....

Ngày....tháng.....năm 2012
Cán bộ hướng dẫn chính
(Ký, ghi rõ họ tên)

**PHÂN NHẬN XÉT ĐÁNH GIÁ CỦA CÁN BỘ CHĂM PHẢN BIỆN
ĐỀ TÀI TỐT NGHIỆP**

1. Đánh giá chất lượng đề tài tốt nghiệp (về các mặt như cơ sở lý thuyết, thuyết minh chương trình, giá trị thực tế,...)

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

2. Cho điểm của cán bộ phản biện

(Điểm ghi bằng số và chữ)

.....
.....
.....

Ngày.....tháng..... năm 2012
Cán bộ chăm phản biện
(Ký, ghi rõ họ tên)

LỜI CẢM ƠN

Để hoàn thành đề án này, em xin tỏ lòng biết ơn sâu sắc đến PGS.TS. Ngô Quốc Tạo, đã tận tình hướng dẫn em trong suốt quá trình làm đề án tốt nghiệp.

Em cũng xin chân thành cảm ơn quý Thầy, Cô trong khoa Công Nghệ Thông Tin trường Đại Học Dân Lập Hải Phòng đã tận tình truyền đạt kiến thức trong suốt những năm em học tập tại trường. Với vốn kiến thức được tiếp thu trong quá trình học không chỉ là nền tảng cho quá trình nghiên cứu đề án mà còn là hành trang quý báu để em bước vào đời một cách vững chắc và tự tin.

Em cũng thầm biết ơn sự ủng hộ của gia đình, bạn bè – những người thân yêu luôn là chỗ dựa vững chắc cho em.

Cuối cùng, em xin kính chúc quý Thầy, Cô và gia đình dồi dào sức khỏe và thành công trong sự nghiệp cao quý.

Hải Phòng, ngày 21 tháng 11 năm 2012

Sinh viên thực hiện

Trần Văn Tuấn

MỤC LỤC

LỜI CẢM ƠN	8
LỜI NÓI ĐẦU	11
CHƯƠNG I . GIỚI THIỆU	12
CHƯƠNG II . Cơ sở lý thuyết giai đoạn tiền xử lý ảnh kỹ tự	13
I .Lọc mịn ảnh:.....	13
II .Nhị phân ảnh:	13
III. Đánh nhãn thành phần liên thông:	15
1. Tách liên thông bằng kỹ thuật đệ quy:	15
2. Giải thuật cải tiến:	16
IV. Chuẩn kích thước:	18
V. Lập khoảng trống ảnh bằng phép đóng morphology:.....	18
1. Một số định nghĩa:.....	18
2. Phép giãn :	19
3. Phép co :	19
4 . Phép đóng:.....	19
VI. Lấy đường biên và làm trơn đường biên:	20
1. Phát hiện biên:	20
2. Dò biên và mã hóa đường biên:	21
3. Xác định hướng của điểm biên (Freeman code):	21
4. Làm trơn đường biên:.....	22
VII .Phương pháp chuyển ảnh RGB thành ảnh đa cấp xám.....	24
VIII .Phân đoạn ảnh và lấy ngưỡng ảnh	26
1. Phân đoạn (hay phân vùng).....	26
2. Các hướng tiếp cận phân đoạn ảnh	26
3. Một số phương pháp phân đoạn cụ thể	26
4 .Lấy ngưỡng (phân ngưỡng).....	27

5 .Phương pháp làm mảnh ảnh.....	29
CHƯƠNG III: RÚT ĐẶC TRUNG	31
1. Giới thiệu đặc trưng hướng:	31
2.Chia ô:.....	31
3. Đặc trưng hướng của đường biên:	32
CHƯƠNG IV : ĐIỂM NGÃ 3 VÀ PHƯƠNG PHÁP PHÂN ĐOẠN TÁCH NÉT DỰA TRÊN ĐIỂM NGÃ BA.....	33
1.Các khái niệm cơ bản	33
2) Xác định các láng giềng của một điểm ảnh.....	36
3.Điểm ngã ba :	38
4 .Phân đoạn tách nét dựa vào điểm ngã ba.....	39
CHƯƠNG V : CHƯƠNG TRÌNH VÀ MỘT SỐ KẾT QUẢ	42
THỬ NGHIỆM.....	42
TÀI LIỆU THAM KHẢO	48

LỜI NÓI ĐẦU

Nhận dạng ký tự đặc biệt là ký tự viết tay là bài toán có nhiều ứng dụng thực tiễn và rất được quan tâm trong những năm gần đây. Tuy nhiên hiện nay chưa có phương pháp nào hoàn chỉnh cho bài toán này do tính phức tạp và sự biến dạng của dữ liệu đầu vào.

Phương pháp phân đoạn tách nét chỉ là một bước cơ bản mang tính chất phát triển trong bài toán nhận dạng chữ viết tay.

Mục tiêu của đề án nhằm giới thiệu một phương pháp tiếp cận bài toán nhận dạng chữ viết tay với một số ràng buộc , nhằm từng bước đưa vào ứng dụng thực tiễn.

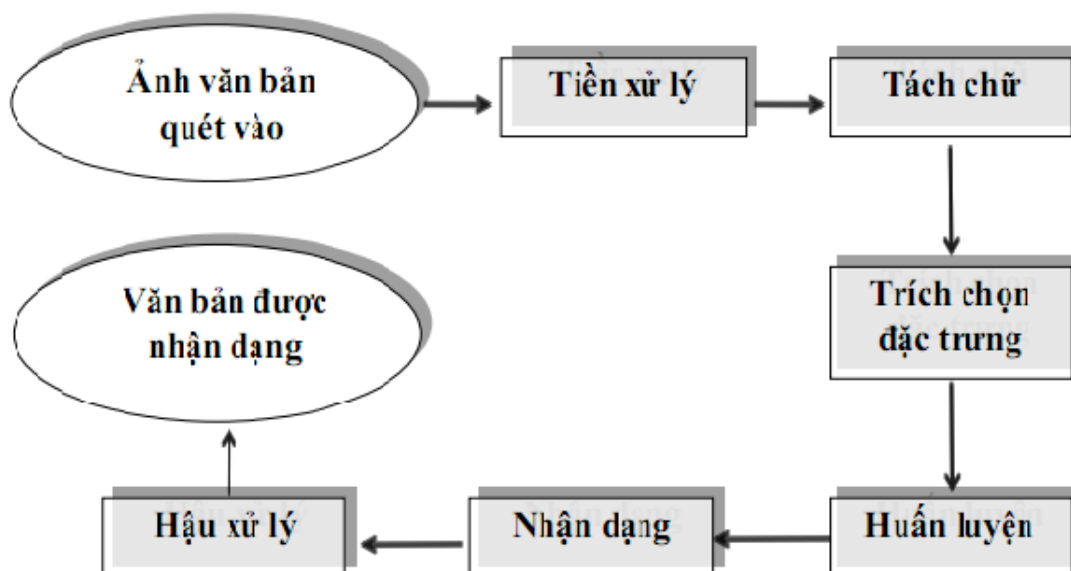
Mặc dù hết sức cố gắng , song do thời gian có hạn và những hạn chế bản thân nên đề án còn nhiều thiếu sót , mong tiếp tục nhận được sự chỉ bảo của Thầy , Cô và ý kiến đóng góp của các bạn sinh viên để đề án được hoàn thiện hơn . Em xin chân thành cảm ơn

CHƯƠNG I. GIỚI THIỆU

Nhận dạng ký tự , đặc biệt là kí tự viết tay là bài toán có nhiều ứng dụng thực tế .Máy tính sẽ tự xử lý , nhận dạng các biểu mẫu , phiếu điều tra tự động , bằng cách này ta có thể tiết kiệm được nhiều chi phí về thời gian , công sức cũng như các chi phí khác cho việc nhập dữ liệu.

Ngày nay có rất nhiều hướng đi cho việc giải quyết bài toán này như : nhận dạng kí tự dựa trên cấu trúc hay các cách tiếp cận khác như : dung logic mờ , giải thuật di truyền, mô hình xác suất thống kê, mô hình mạng nơ ron. Tuy nhiên, cho tới nay, bất kể hệ thống nhận dạng chữ viết nào cũng cần phải qua các giai đoạn cơ bản trong sơ đồ tổng quát dưới đây :

SƠ ĐỒ TỔNG QUÁT CỦA HỆ NHẬN DẠNG CHỮ



Hình 1 Sơ đồ tổng quát hệ thống nhận dạng chữ viết

CHƯƠNG II. Cơ sở lý thuyết giai đoạn tiền xử lý ảnh kỹ thuật

I. Lọc mịn ảnh:

Lọc mịn ảnh là một lọc thông thấp, giá trị của một điểm ảnh là trung bình trọng số của các điểm ảnh lân cận, hay giá trị điểm ảnh là kết quả của quá trình xoắn (convole) của các điểm ảnh lân cận với một nhân. Nhân có kích thước tùy ý 3x3, 5x5, kích thước nhân càng lớn thì càng nhiều điểm lân cận ảnh hưởng vào điểm ảnh kết quả. Ví dụ một số nhân lọc mịn ảnh như sau:

1	1	1
1	1	1
1	1	1

1/9

1	1	1
1	2	1
1	1	1

1/10

1	1	1
1	4	1
1	1	1

1/12

Hình I.1

Lọc mịn ảnh có tác dụng hạn chế ảnh nhiễu.

II. Nhị phân ảnh:

Nhị phân ảnh mức xám là tìm giá trị ngưỡng sao cho các điểm ảnh có giá trị lớn hơn ngưỡng được gọi là trắng (nền) và các điểm ảnh có giá trị nhỏ hơn ngưỡng được gọi là đen (đôi tượng).

Tiêu chuẩn xác định ngưỡng thường sử dụng nhất là sử dụng sai số bình phương trung bình giữa giá trị mẫu v và mức tái thiết $r(v)$. (ký hiệu MSE)

Ta có:

$$E\{e^2\} = E\{[v - r(v)]^2\} = \int_0^{\infty} [v - r(v)]^2 p(v) dv$$

Trong đó $p(v)$ là hàm mật độ xác suất của biến ngẫu nhiên v , có thể coi xấp xỉ bằng histogram của ảnh. Với một số cho trước L các mức xám MSE được biểu diễn bởi:

$$E\{e^2\} = \sum_j \int_j^{j+1} (v - v_j)^2 p(v) dv$$

Do $r(v) = r_j$ là hằng số trong đoạn $[v_j, v_{j+1}]$.

Với $p(v)$ cho trước và số mức tái thiết L cố định, các mức quyết định v_j , $L-1$ và các mức tái thiết r_j , $j=0..L-1$ cực tiểu hố MSE tuân theo quan hệ sau :

$$v_j = (r_{j-1} + r_j) / 2 \quad j = 1 \dots L - 1$$

Tuy nhiên, không có cách giải quyết dưới dạng khép kín nào tồn tại trừ khi chấp nhận một số phép xấp xỉ.

Bây giờ ta xem xét trường hợp ngoại lệ nhưng quan trọng với $L=2$. Đó là trường hợp nhị phân hóa ảnh. Khi đó MSE trở thành:

$$E\{e^2\} = \int_{v_0}^{v_1} (v - r_0)^2 p(v) dv + \int_{v_1}^{v_2} (v - r_1)^2 p(v) dv$$

Giả sử rằng $p(v)$ có thể ước lượng từ histogram và v_0, v_2 tương ứng với v_{min}, v_{max} . Còn lại ba tham số cần được tính toán đó là r_0, r_1 , và v_1 . Tham số v_1 gọi là ngưỡng nhị phân hóa. Hơn nữa $r_0(v_1)$ và $r_1(v_1)$ cực tiểu MSE, với một giá trị cho trước của v_j , đơn giản là những giá trị trung bình trong đoạn tương ứng:

$$r_0(v_1) = \left\{ \int_{v_0}^{v_1} v p(v) dv \right\} / \int_{v_0}^{v_1} p(v) dv$$

$$r_1(v_1) = \left\{ \int_{v_1}^{v_2} v p(v) dv \right\} / \int_{v_1}^{v_2} p(v) dv$$

Như vậy đủ để biến đổi v_1 từ v_0 đến v_2 . MSE được tính bằng cách thay r_0 và r_1 bằng $r_0(v_1)$ và $r_1(v_1)$ tương ứng và chọn v_1 sao cho MSE là cực tiểu.

Otsu đề nghị một cách tương tự nhưng tiêu chuẩn đơn giản hơn về mặt tính toán dựa trên phân tích biệt số. Trong công thức này, MSE tương đương với

phương sai lớp trong $\sigma_w^2(v_1)$. Nếu $\sigma_w^2(v_1)$ được bổ sung vào phương sai lớp giữa $\sigma_B^2(v_1)$, ta được toàn bộ biến đổi σ_T^2 (độc lập v_1). Như vậy, thay vì cực tiểu MSE, giải thuật của Otsu cực đại phương sai giữa lớp:

$$v_1^* = \arg \max \{p_0(v_1)[\mu_0(v_1) - \mu_T]^2 + p_1(v_1)[\mu_1(v_1) - \mu_T]^2\}$$

Thật ra, còn tồn tại một số tiêu chuẩn lượng hóa khác, chẳng hạn entropy, cũng trên histogram của mức xám. Histogram có thể được tính từ toàn bộ ảnh hoặc từ lân cận địa phương giới hạn xung quanh ảnh đang xét.

III. Đánh nhãn thành phần liên thông:

Khái niệm liên thông trong kí tự có thể xem như tập hợp các điểm ảnh liên nhau tạo nên kí tự, mục tiêu của giải thuật này là lọc ra những kí tự có trong ảnh.

1. Tách liên thông bằng kĩ thuật đệ quy:

Để tách liên thông ta nghĩ ngay đến kỹ thuật đệ quy cấp 8 (tại mỗi bước chúng ta có tối đa 8 lựa chọn tiếp theo).

Quy ước: Điểm (x, y) : cho biết giá trị điểm tại toạ độ (x, y) .

=1 màu chữ

=0 màu nền

Height: Chiều cao ảnh

Width: Chiều rộng ảnh

Ta có giải thuật như sau:

// Đưa tất cả các điểm liên thông với Điểm (x,y) vào liên thông t

Procedure Chonvao((x,y) :điểm, t :liênthông)

Begin

If (Điểm (x, y) =1) AND (NOT đã_xét $[x,y]$) then

Begin

Them_vao_lien_thong(t, (x, y));

Đã_xét $[x, y]$:=True ;

For i :=x-1 to x+1 do

For j:= y-1 to y+1 do

```

    If ( i <>x) or (j<>y) then Chonvao((i, j),t);
End;
End ;
// Ta có thủ tục tách liên thông đệ quy như sau :
Procedure TáchLiênThôngĐQ(VAR LT: Danh_Sách_Liên_Thông)
Begin
    T:=<danh sách rỗng>
    t:=<liên thông rỗng>
    For j:=1 to Height do
        For i:=1 to Width do đã_xét[i, j]:=False;

    For j:=1 to Height do

        For i:=1 to Width do
            If (Điểm(x, y)=1) AND( NOT Đã_xét[i, j]) then
                Begin
                    Chonvao( (i, j ),t);
                    Thêm_liên_thông_vào_Danh_Sách(T, t)  ;
                    t :=<liên thông rỗng> ;
                End ;
            End;
End;

```

Nhận xét:

Thuật toán này chỉ có ý nghĩa minh họa bản chất của tách liên thông. Ta không chọn thuật toán này cài đặt vì chi phí đệ quy quá cao, chưa kể tốc độ thực hiện.

2. Giải thuật cải tiến:

Để gán nhãn cho thành phần liên thông ta có thể duyệt theo từng đường chạy. Kỹ thuật này gán cho mỗi thành phần liên thông của ảnh nhị phân một nhãn riêng biệt. Nhãn thường là các số tự nhiên bắt đầu từ một đến tổng số các thành phần liên thông trong ảnh input.

Giải thuật phát biểu như sau:

Quét ảnh từ trái sang phải và từ trên xuống dưới. Trong dòng thứ nhất chứa pixel đen, một nhãn duy nhất được gán cho mỗi đường chạy liên tục của pixel đen. Với mỗi pixel đen của dòng tiếp theo, các pixel lân cận dòng trước và pixel bên trái được xem xét (hình vẽ a). Nếu bất kì pixel lân cận nào được gán nhãn, nhãn tương tự được gán cho pixel đen hiện thời; ngược lại, nhãn tiếp theo chưa được sử dụng được chọn. Thủ tục này được sử dụng cho đến dòng cuối của ảnh. Lúc kết thúc tiến trình này, một thành phần liên thông có thể chứa các pixel có các nhãn khác nhau vì khi chúng ta xem xét lân cận của pixel đen, chẳng hạn pixel “?”(trong hình c), pixel đối với lân cận trái và những lân cận trong dòng trước có thể gán nhãn một cách riêng biệt.(Trong ví dụ này, chúng ta sử dụng nhãn của lân cận trái). Một tình huống như vậy phải được xác định và ghi lại. Sau tiến trình quét ảnh, việc gán nhãn được hoàn tất bằng cách “thống nhất mâu thuẫn các nhãn” và gán lại các nhãn chưa sử dụng.

Để minh họa ta có các hình biểu diễn sau:

```

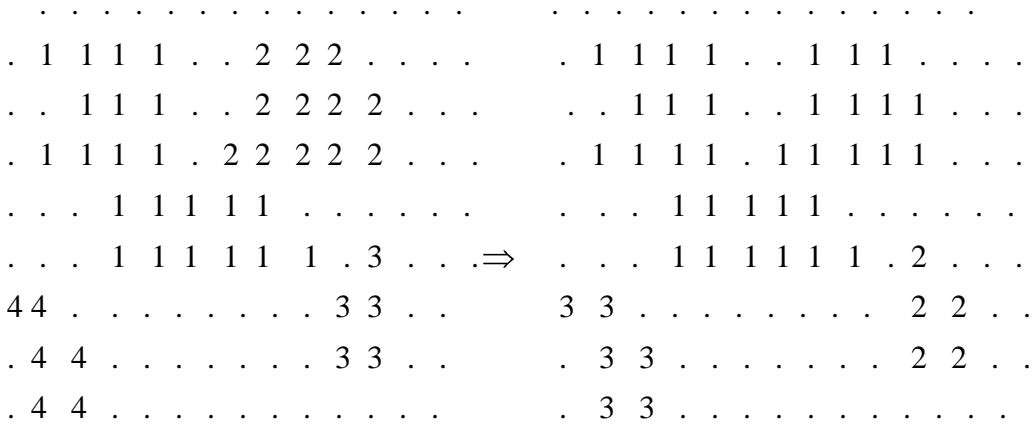
. . . . .
. P P P .
. L ? . .
. . . . .
    
```

Hình a lân cận của “?” P= dòng trước; L=lân cận trái.

.
. * * * * 1 1 1 1 . . 2 2 2
. . * * * . . * * * * 1 1 1 . . 2 2 2 2
. * * * * . * * * * * 1 1 1 1 . 2 2 2 2 2
. . . * * * * * 1 1 ? * *
. . . * * * * * * * . * ⇒	. . . * * * * * * * . *
* * * *	* * * *
. * * * * * * * *
. * * * *

Hình b. Ảnh ban đầu

Hình c. Tiến trình gán nhãn



Hình d . Sau khi quét đầy đủ

Hình e .Kết quả sau cùng

IV. Chuẩn kích thước:

Chuẩn kích thước ảnh kí tự về một kích thước cố định và phóng sát bốn biên của ảnh.

Phóng ảnh là thực hiện phép biến đổi sau:

$$f_s(x, y) = f(x/s_x, y/s_y)$$

Với (x, y) là toạ độ điểm ảnh sau khi phóng và s_x, s_y là tỷ lệ phóng theo trục x và y tương ứng, $f(x,y)$ là giá trị điểm ảnh kết quả ứng với giá trị toạ độ (x, y).

Chú ý:

Sau khi phóng ảnh, ảnh có thể bị rời rạc, răng cưa biên. Để khắc phục tình trạng này, ta thực hiện một số xử lý bằng phép đóng morphology:

V. Lấp khoảng trống ảnh bằng phép đóng morphology:

1. Một số định nghĩa:

Giả sử A và B là hai tập trong không gian Z^2 , $a \in A$ thì $a = (a_1, a_2)$

Phép dịch chuyển của tập A đối với $x = (x_1, x_2)$ ký hiệu $(A)_x$, được định nghĩa.

$$(A)_x = \{c | c = a + x, \text{ với } a \in A\}$$

Phép phản chiếu của tập B, ký hiệu B^* , được định nghĩa:

$$B^* = \{x | x = -b, \text{ với } b \in B\}$$

Phép bù của một tập A, ký hiệu A^c , được định nghĩa:

$$A^c = \{x | x \notin A\}$$

Hiệu của hai tập hợp A và B, ký hiệu $A-B$, được định nghĩa:

$$A - B = \{x | x \in A, x \notin B\}$$

2. Phép giãn :

Giả sử A, B là hai tập hợp thuộc Z^2 , \emptyset là tập rỗng, phép giãn của A đối với B, ký hiệu $A \oplus B$, được định nghĩa :

$$A \oplus B = \{x | (B^*)_x \cap A = \emptyset\}$$

Tập B được gọi là thành phần cấu trúc .

3. Phép co :

Giả sử A, B là hai tập thuộc Z^2 , phép co của A đối với B, ký hiệu $A \ominus B$ được định nghĩa:

$$A \ominus B = \{x | (B^*)_x \subseteq A\}$$

4 . Phép đóng:

Giả sử A, B là hai tập thuộc Z^2 , phép đóng của A đối với B, ký hiệu $A \bullet B$ được định nghĩa:

$$A \bullet B = (A \oplus B) \ominus B$$

Tức phép đóng là phép do thực hiện phép giãn rồi thực hiện phép co lên kết quả vừa có.

Phép đóng có tác dụng làm đầy những khoảng nhỏ (tùy thuộc vào thành phần cấu trúc B) thường xảy ra trên đường biên.

Thành phần cấu trúc thường được sử dụng là thành phần cấu trúc đối xứng có gốc (0, 0) là ở tâm như hình:

0	1	0
1	1	1
0	1	0

Nhưng do ảnh được quét với độ phân giải 300 dpi, và đối với những chữ có bụng được viết khá nghiêng thì khi thực hiện phóng với thành phần cấu trúc trên, tức thực hiện phép giãn rồi thực hiện phép co, thì phép giãn làm cho phần bụng bị dính lại với nhau do với mỗi hướng ngang và đứng đều được giãn 2 điểm ảnh. Để hạn chế điều này ta sử dụng 2 thành phần cấu trúc không đối xứng và thực hiện phép đóng 2 lần trên 2 thành phần cấu trúc này, vì khi thực hiện phép giãn thì chỉ cần giãn về 1 phía:

0	1	0
1	1	0
0	0	0

0	0	0
0	1	1
0	1	0

Các thành phần cấu trúc không đối xứng

VI. Lấy đường biên và làm trơn đường biên:

1. Phát hiện biên:

Biên của ảnh được thiết lập bằng cách nhân chập ảnh với phần tử có cấu trúc:

0	1	0
1	1	1
0	1	0

2. Dò biên và mã hóa đường biên:

Ta có giải thuật dò tìm biên như sau:

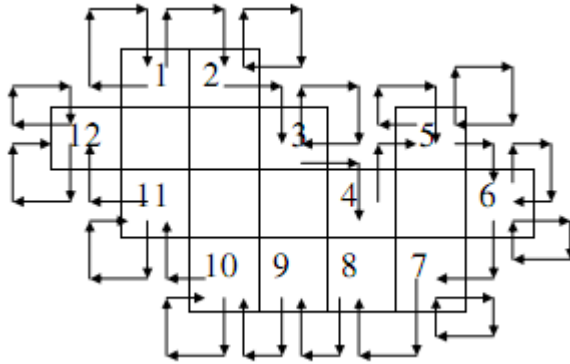
Bước 1: Quét ảnh đến khi gặp điểm ảnh đen. Gọi nó là pixel 1

Bước 2: Lặp

Nếu “điểm ảnh hiện thời là đen” thì “dò ngược”

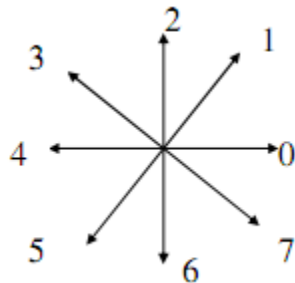
Ngược lại “sang phải” đến khi “gặp pixel 1”

Minh họa dò biên



3. Xác định hướng của điểm biên (Freeman code):

Các hướng được quy ước như sau:



Với hướng quy ước trên, đường biên được mã hóa như sau:



4. Làm trơn đường biên:

Làm trơn đường biên là duyệt theo đường biên, nếu hai điểm liên tiếp trên đường biên có hiệu số hướng lớn hơn 1 thì có thể hiệu chỉnh để có đường biên mà hai điểm liên tiếp có hiệu số hướng bằng 1.

Theo mã hướng Freeman, hiệu số hướng của 2 điểm liên tiếp nhau trên đường biên được định nghĩa :

Goi c_i là mã hướng tại điểm biên đang xét p_i , c_{i+1} là mã hướng của điểm kế tiếp trên đường biên p_{i+1}

Đặt $d=c_{i+1}-c_i$ và $Dabs=|d|$

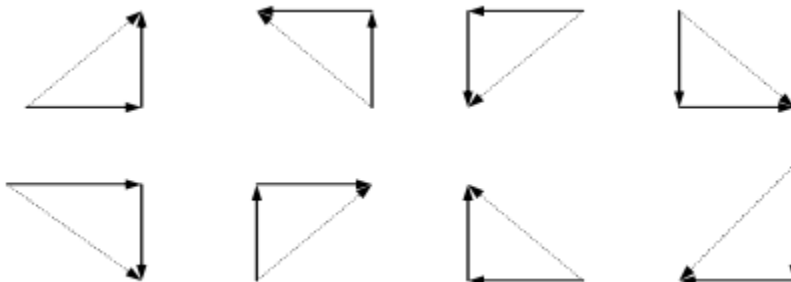
nếu $|d| \leq 4$

Và $dabs=8-|d|$ nếu $|d| > 4$

Ta có các trường hợp sau:

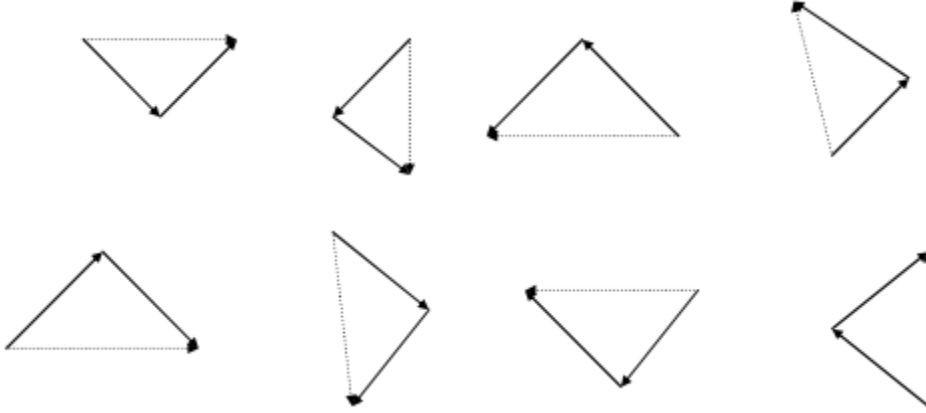
a. $dabs \leq 1$: Điểm biên trơn.

b. $dabs=2$ và c_i chẵn, c_{i+1} chẵn : bỏ p_{i+1} và thay hướng p_i như sau:

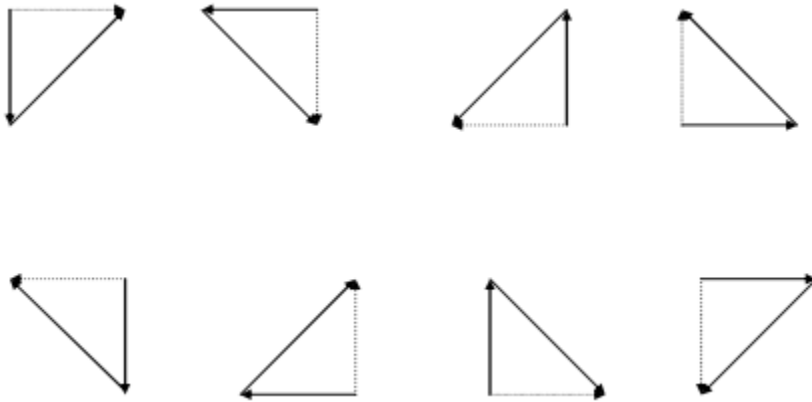


Dabs = 2 và c_i chẵn, c_{i+1} chẵn

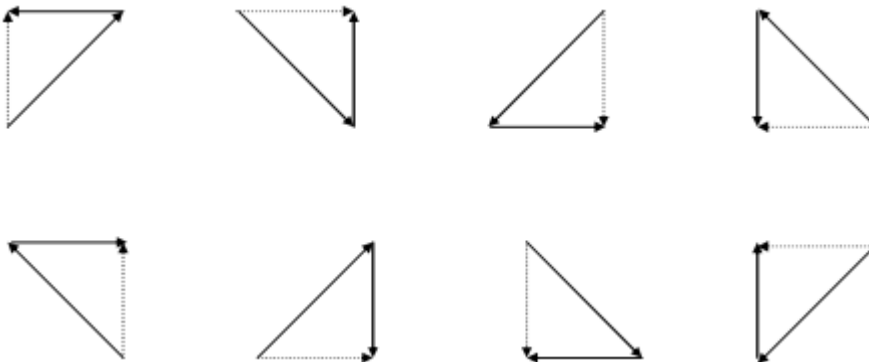
c. dabs=2 và c_i lẻ, c_{i+1} lẻ : Bỏ p_{i+1} và thay hướng p_i như hình



d. dabs=3, c_i chẵn, c_{i+1} lẻ



e. dabs=3, c_i lẻ, c_{i+1} chẵn



Minh hoạ ảnh kí tự sau quá trình tiền xử lý .



ảnh ban đầu



ảnh sau khi xử lý

VII .Phương pháp chuyển ảnh RGB thành ảnh đa cấp xám

Đơn vị cơ bản của một bức ảnh là điểm ảnh (pixel), mỗi điểm ảnh có thể được biểu diễn bằng n bytes dưới các hệ màu khác nhau .Việc chuyển đổi giữa các hệ màu thông thường được thực hiện thông qua các phép biến đổi ma trận. Trong bài viết này tôi sẽ giới thiệu phương thức chuyển đổi từ ảnh 24 bits RGB sang ảnh 8bits đa mức xám. Để thực hiện yêu cầu trên, thông thường ta sử dụng một trong những công thức sau đây, áp dụng cho từng điểm ảnh [x,y]

$$L_{x,y} = 0.3086 * Red_{x,y} + 0.6094 * Green_{x,y} + 0.0820 * Blue_{x,y}$$

$$L_{x,y} = 0.299 * Red_{x,y} + 0.587 * Green_{x,y} + 0.114 * Blue_{x,y}$$

Chú ý rằng giá trị mỗi điểm ảnh $L_{x,y}$ hay còn gọi là cường độ sáng (Luminance Intensity) trong ảnh đa mức xám tính được là tổng trọng số khác nhau của mỗi thành phần màu trong hệ màu RGB. Một trong những lý do của việc này là nếu chúng ta sử dụng cùng trọng số, ví dụ $(R + G + B) / 3$ thì màu đỏ, màu xanh nước biển hay màu xanh da trời sẽ có cùng mức xám sau khi chuyển đổi .Mặt khác theo khoa học đã chứng minh thì mắt người nhạy cảm hơn với thành phần màu xanh lá cây và màu đỏ so với xanh da trời.

Việc triển khai trong C/C++ của công thức trên khá đơn giản như sau với một ảnh có định dạng điểm ảnh liên tục (interleaved color) RGBRGB:

```
for ( i = 0; i < imgSize; i += 3 )
{
    out[i] = (unsigned char)(0.299*in[i] + 0.587*in[i+1] + 0.114*in[i+2] +
0.5);
}
```

Vì việc thực hiện phép nhân trong miền số nguyên thì nhanh hơn rất nhiều trong miền số thực nên công thức trên được viết lại trong miền số nguyên như sau:

$$L_{x,y} = (2 * Red_{x,y} + 5 * Green_{x,y} + 1 * Blue_{x,y}) / 8$$

Cộng với việc thay thế các phép chia và nhân bằng toán tử shift tốc độ của quá trình tính toán sẽ được đẩy lên một cách hiệu quả. Dưới đây là mã nguồn cho quá trình nâng cấp này:

```
int iTmp;
for ( i = 0; i < imgSize; i += 3 )
{
    iTmp = in[i] << 1;
    iTmp += in[i+1] << 2 + in[i+1];
    iTmp += in[i+2];
    out[i] = (unsigned char)(iTmp>> 3);
}
```

Lưu ý: trong thực hành, không phải lúc nào thứ tự của 3 bytes trong 1 điểm ảnh cũng là R - G - B, đôi khi thứ tự này có thể là B - G - R.

VIII .Phân đoạn ảnh và lấy ngưỡng ảnh

1. Phân đoạn (hay phân vùng)

Phân vùng ảnh là tách một ảnh đầu vào thành các vùng thành phần để biểu diễn phân tích, nhận dạng ảnh. Ví dụ: để nhận dạng chữ (hoặc mã vạch) trên phong bì thư cho mục đích phân loại bưu phẩm, cần chia các câu, chữ về địa chỉ hoặc tên người thành các từ, các chữ, các số (hoặc các vạch) riêng biệt để nhận dạng. Đây là phần phức tạp khó khăn nhất trong xử lý ảnh và cũng dễ gây lỗi, làm mất độ chính xác của ảnh. Kết quả nhận dạng ảnh phụ thuộc rất nhiều vào công đoạn này.

2. Các hướng tiếp cận phân đoạn ảnh

Phân đoạn ảnh là chia ảnh thành các vùng không trùng lắp. Mỗi vùng gồm một nhóm pixel liên thông và đồng nhất theo một tiêu chí nào đó. Tiêu chí này phụ thuộc vào mục tiêu của quá trình phân đoạn. Ví dụ như đồng nhất về màu sắc, mức xám, kết cấu, độ sâu của các layer... Sau khi phân đoạn mỗi pixel chỉ thuộc về một vùng duy nhất. Để đánh giá chất lượng của quá trình phân đoạn là rất khó. Vì vậy trước khi phân đoạn ảnh cần xác định rõ mục tiêu của quá trình phân đoạn là gì. Xét một cách tổng quát, ta có thể chia các hướng tiếp cận phân đoạn ảnh thành ba nhóm chính như sau:

- Các kỹ thuật phân đoạn ảnh dựa trên không gian đặc trưng.
- Các kỹ thuật dựa trên không gian ảnh.
- Các kỹ thuật dựa trên các mô hình vật lý.

3. Một số phương pháp phân đoạn cụ thể

3.1) Phương pháp phân đoạn của B.G. Prasad

Đây là phương pháp do B.G. Prasad đề xuất và được áp dụng trong hệ thống truy vấn ảnh theo chỉ mục màu sắc hình dạng và vị trí của tác giả. Phương pháp này sử dụng sự lượng tử hóa màu trong không gian màu RGB sử dụng 25 màu phân biệt (bằng giác quan) để phân đoạn ảnh dựa trên những màu trội. Vì 25 màu là đủ để phân biệt rõ tất cả các vùng màu trong cơ sở dữ liệu hình ảnh mà tác giả chọn.

3.2) Phương pháp phân đoạn dựa trên ngưỡng cục bộ thích nghi

Số ngưỡng cục bộ và giá trị của chúng không được chỉ định trước mà được trích lọc thông qua quá trình kiểm tra các thông tin cục bộ. Giải thuật gồm các bước tuần tự như sau:

Áp dụng giải thuật Watershed chia ảnh thành rất nhiều vùng con .

Trộn các vùng và đồng thời phát hiện ngưỡng cục bộ .Ngưỡng được tính từ thông tin cục bộ của vùng và các vùng lân cận

Giải thuật này cho kết quả tương đối tin cậy trên nhiều loại ảnh khác nhau

3.3) Phân đoạn sơ khởi bằng Watershed

Dữ liệu đầu vào của giải thuật Watershed là một ảnh xám. Vì vậy, trước tiên ta biến đổi ảnh đầu vào I thành ảnh xám. Sau đó, dùng giải thuật tìm cạnh Canny [20] để lấy cường độ gradient, kí hiệu là IG. Với ảnh gradient nhận được, ta hình liên tưởng đến một lược đồ địa hình, vùng có độ xám cao hơn là vùng trũng hơn và ngược lại. Tại mỗi pixel, việc đánh giá sẽ dựa vào giá trị mức xám của pixel đó.

4 .Lấy ngưỡng (phân ngưỡng)

4.1 Phân ngưỡng

Kỹ thuật này đặt ngưỡng để hiển thị các tông màu liên tục. Các điểm trong ảnh được so sánh với ngưỡng định trước. Giá trị của ngưỡng sẽ quyết định điểm có được hiển thị hay không.

Do vậy ảnh kết quả sẽ mất đi một số chi tiết. Có nhiều kỹ thuật chọn ngưỡng áp dụng cho các đối tượng khác nhau:

- Hiện thị 2 màu: chỉ dùng ảnh đen trắng có 256 mức xám. Bản chất của phương pháp này là chọn ngưỡng dựa trên lược đồ mức xám của ảnh. Để đơn giản có thể lấy ngưỡng với giá trị là 127.

Như vậy:

$$u(m,n) = \begin{cases} 1 & \text{khi } h(m,n) < 127 \\ 0 & \text{khác} \end{cases}$$

trong đó $u(m, n)$ là mức xám tại tọa độ (m, n) .

Nhìn chung kỹ thuật này khó chấp nhận vì ảnh mất khá nhiều chi tiết.

- Hiện thị 4 màu: hiện 4 màu để khắc phục nhược điểm của kỹ thuật hiện thị 2 màu. Một ví dụ của Bảng mã 4 màu được cho ở Bảng 2.1.

Bảng 4.1 Bảng mã 4 màu

Mã màu	Màn hình monochrome (đơn sắc)	Màn hình màu
0	Đen	Đen
1	Xám đậm	Đỏ
2	Xám nhạt	Xanh
3	Trắng	Vàng

4.2) Kỹ thuật chọn theo mẫu

Kỹ thuật này sử dụng một nhóm các phần tử trên thiết bị ra (máy in chẳng hạn) để biểu diễn một pixel trên ảnh nguồn. Các phần tử của nhóm quyết định độ sáng tối của cả nhóm. Các phần tử này mô phỏng các chấm đen trong kỹ thuật nửa cường độ. Nhóm thường được chọn có dạng ma trận vuông. Nhóm $n \times n$ phần tử sẽ

tạo nên n^2+1 mức sáng. Ma trận mẫu thường được chọn là ma trận Rylander. Ma trận Rylander cấp 4 có dạng như

Bảng 4.2 .

Bảng 4.2. Ma trận Rylander cấp 4

0	8	2	10
4	12	6	14
3	11	1	9
7	15	5	3

Việc chọn kích thước của nhóm như vậy sẽ làm giảm độ mịn của ảnh. Vì vậy kỹ thuật này chỉ áp dụng trong trường hợp mà độ phân giải của thiết bị ra lớn hơn độ phân giải của ảnh nguồn. Thí dụ: thiết bị ra có độ phân giải 640x480 khi sử dụng nhóm có kích thước 4x4 sẽ chỉ còn 160x120.

4.3) Kỹ thuật Dithering

Dithering là việc biến đổi một ảnh đa cấp xám (nhiều mức sáng tối) sang ảnh nhị phân (hai mức sáng tối). Kỹ thuật Dithering được áp dụng để tạo ra ảnh đa cấp sáng khi độ phân giải nguồn và đích là như nhau. Kỹ thuật này sử dụng một ma trận mẫu gọi là ma trận Dither. Ma trận này gần giống như ma trận Rylander.

Để tạo ảnh, mỗi phần tử của ảnh gốc sẽ được so sánh với phần tử tương ứng của ma trận Dither. Nếu lớn hơn, phần tử ở đầu ra sẽ sáng và ngược lại

5 .Phương pháp làm mảnh ảnh

5.1) Tổng quan về làm mảnh

Làm mảnh ảnh là phương pháp dùng để đưa ra xương của đối tượng, là các dạng thuật toán xóa các điểm biên của mẫu một cách có chọn lọc cho đến khi thu được xương. Mà thuật ngữ xương dùng để chỉ kết quả mà không quan tâm đến

hình dạng chuẩn của mẫu đối tượng. Xương được coi là hình dạng cơ bản của một đối tượng với số ít điểm ảnh cơ bản. Ta có thể lấy được các thông tin về hình dạng cơ bản của ảnh thông qua xương

Khi làm mảnh ảnh ta cần chú ý các vấn đề sau:

Không phải tất cả các đối tượng đều có thể làm mảnh, làm mảnh chỉ hữu dụng với các đối tượng là đường. Nghĩa là chúng chỉ thẳng hoặc cong và nó không có tác dụng với các đối tượng có hình dạng đóng trong một vùng.

Làm mảnh thông thường là bước chuẩn bị cho các bước tiếp theo xử lý một đối tượng ảnh. Các bước tiếp theo làm việc trên các thuộc tính cần thiết của xương.

5.2) Các thuật toán làm mảnh

Có 2 dạng thuật toán:

- Thuật toán tuần tự: Các điểm ảnh được xóa đi theo một trật tự nhất định trong mỗi vòng lặp và giá trị điểm ảnh sau mỗi vòng lặp không chỉ phụ thuộc vào các láng giềng mà còn phụ thuộc vào các điểm ảnh đã được xét trước đó trong chính vòng lặp đó. Gồm một số thuật toán như:

- Làm mảnh theo các điểm biên (bao gồm thuật toán của Chu và Suen, thuật toán của Arcelli, thuật toán của Pavlidis, thuật toán của Kwok)

- Làm mảnh theo loạt (gồm các thuật toán: thuật toán của Yakei, thuật toán SPKA,)

- Thuật toán song song: Các điểm ảnh có thể được xử lý cùng một lúc, giá trị của điểm ảnh chỉ phụ thuộc vào giá trị của các láng giềng. Chính vì thế mà thuật toán này được sử dụng trên các bộ vi xử lý song song để tăng khả năng tính toán. Tùy theo số chu trình con được sử dụng trong thuật toán mà làm mảnh song song được chia ra thành các kiểu như sau:

- Làm mảnh song song sử dụng 1 chu trình con (gồm thuật toán Rutovitz, thuật toán Holt, thuật toán của Favre và Keller, thuật toán của Huang, Wan và Liu)

- Làm mảnh song song 2 chu trình con (gồm thuật toán của Zang-Suen, thuật toán của Suzuki, thuật toán của Guo)

- Làm mảnh song song 4 chu trình con (gồm các thuật toán : thuật toán của Bel-lan và Monoto, thuật toán của Rosenfeld và điều kiện Hilditch, thuật toán của Stentiford)

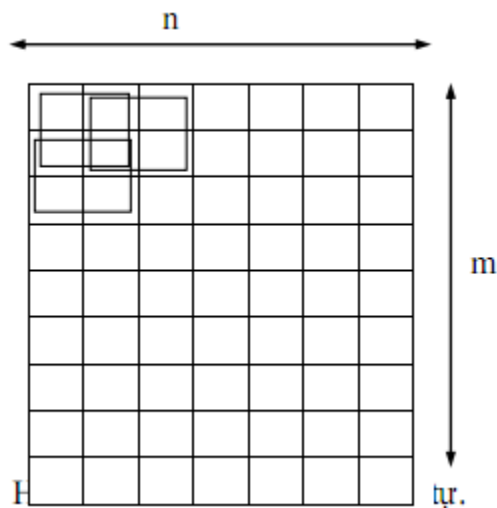
CHƯƠNG III: RÚT ĐẶC TRUNG

1. Giới thiệu đặc trưng hướng:

Hướng các điểm trên đường biên của kí tự mô tả khá đầy đủ về kí tự. Đặc trưng của ảnh được xác định là hướng của các điểm ảnh trên biên. Việc chọn đặc trưng để nâng cao độ chính xác của bài toán nhận dạng là hết sức khó khăn, đòi hỏi rất nhiều thời gian và quyết định rất nhiều đến độ chính xác. Hơn nữa, do biến dạng khá lớn trong chữ viết tay nên để hạn chế người ta thường chia ô trên ảnh và đặc trưng được rút trong các ô đó, việc chọn các ô phủ lấp lên nhau cũng không ngòi mục đích trên.

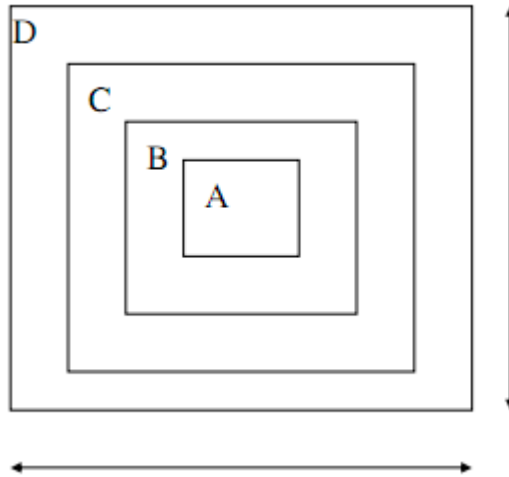
2. Chia ô:

Ảnh kí tự sau khi tiền xử lý kích thước được chuẩn về $m \times n$ điểm ảnh, ảnh được chia nhỏ thành các ô vuông nhỏ kích thước 8×8 điểm ảnh như hình:



Gom 4 ô kích thước 8x8 thành ô kích thước 16x16, và các ô kích thước 16x16 này được phủ lên nhau một nửa theo hai hướng ngang và dọc. Trên mỗi ô kích thước 16x16 sẽ rút đặc trưng theo 4 hướng (00, 450, 900, 1350). Mỗi ô này được chia làm 4 phần theo điểm tâm của ô, phần trung tâm A có kích thước 4x4 điểm ảnh, phần B có kích thước 8x8 điểm ảnh trừ đi phần A, phần C có kích thước 12x12 điểm ảnh trừ đi phần B và A, và phần D là phần còn lại của ô có kích thước 16x16 trừ đi phần C, B và A.

Ta có hình minh họa các phần A, B, C, D trong mỗi ô vuông 16x16



Đặt trọng số vùng A, B, C và D tương ứng là 4, 3, 2, và 1. Gọi x_j là một oại đặc trưng, x_j được tính cho một ô kích thước 16x16 như sau:

$$x_j = 4 x_j (A) + 3 x_j (C) + x_j (D)$$

3. Đặc trưng hướng của đường biên:

Để rút đặc trưng này là ảnh chỉ chứa đường biên. Với mỗi ô kích thước 16x16 điểm ảnh, ta rút ra 4 đặc trưng x_j ($j=1, 2, 3, 4$), x_j được tính như công thức trên, x_1 là số điểm biên có hướng 00 hay (1800), x_2 là số điểm biên có hướng 450 (hay -1350), x_3 là số điểm có hướng 900 (hay -900) và x_4 là số điểm có hướng 1350 (hay -450).

Như vậy ảnh kí tự sẽ được mô tả dưới dạng:

$$X = (x_1, x_2, x_3, x_4, \dots, x_n)$$

Trong đó $n = k \cdot 4$, với k là tổng số ô vuông 16×16 xếp chồng lên nhau.

CHƯƠNG IV : ĐIỂM NGÃ 3 VÀ PHƯƠNG PHÁP PHÂN ĐOẠN TÁCH NÉT DỰA TRÊN ĐIỂM NGÃ BA

1. Các khái niệm cơ bản

Xét một điểm ảnh $P(i, j)$ và các điểm lân cận

x_4	x_3	x_2
x_5	P	x_1
x_6	x_7	x_8

Hình điểm ảnh P và các láng giềng

1.1) Láng giềng của một điểm ảnh

Các điểm ảnh x_1, x_2, \dots, x_8 được gọi là 8_láng giềng của P và được biểu thị bởi $N(P)$, chúng còn được gọi là kề 8 của P . Các điểm ảnh x_1, x_3, x_5, x_7 được gọi là 4_láng giềng hay kề 4 của P. Ta sẽ sử dụng x_i để chỉ các điểm ảnh và x_i là điểm ảnh trắng hoặc đen tùy thuộc vào giá trị tương ứng 0 hay 1 của nó . Số các điểm ảnh trắng hoặc đen trong $N(P)$ được gọi là $b(P)$.

1.2) Đường đi

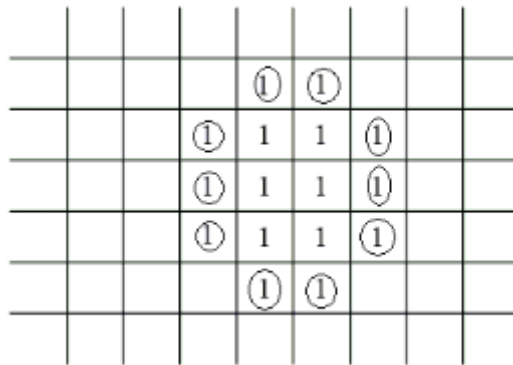
Một trật tự các điểm ảnh y_1, y_2, \dots, y_n được gọi là một 8_đường đi (hoặc 4_đường đi) nếu y_{i+1} là một trong 8_láng giềng (hay 4_láng giềng) của y_i ($i = 1, 2, \dots, n-1$)

Hai điểm được gọi là liên thông với nhau nếu tồn tại đường đi giữa chúng .

Một bộ con Q của ảnh P được gọi 8_liên thông (hay 4_liên thông) nếu mọi cặp điểm x.y trong Q đều tồn tại 8_đường đi (hoặc 4_đường đi) từ x đến y phù hợp với các điểm của Q. Trong trường hợp này, Q được gọi là một 8_ thành phần (hoặc 4_ thành phần) của P.

1.3) Điểm biên

Các điểm ảnh có ít nhất 1 trong các láng giềng là trắng được gọi là điểm biên . Có 2 loại biên 4 liên kết và 8 liên kết



Hình các điểm biên liên kết 8

1.4) Điểm trong

Các điểm đen mà không phải điểm biên được gọi là điểm trong của ảnh

1.5) Điểm cuối

Một điểm đen thỏa mãn điều kiện $b(P) = 1$ thì được coi là điểm cuối. Điều kiện này được một số tác giả đưa ra với các dạng khác nhau : p có thể được giữ lại khi có 2 hoặc 3 các điểm đen phối hợp trên một bên của $N(P)$, điều kiện này có thể được áp dụng chỉ sau hai vòng lặp đầu tiên , hoặc rất có thể nó sẽ bị bỏ qua hoàn toàn để tránh các nhánh giả.

Phần lớn sự khác nhau giữa các thuật toán là ở điều kiện đảm bảo tính liên thông .Điều kiện này được định nghĩa theo các thuật ngữ số giao, số liên thông, và điểm ảnh đơn

1.6) Số giao

Có hai định nghĩa về số giao của một điểm ảnh

Theo Rutovitz, số giao là số lần biến đổi từ điểm trắng sang điểm đen và ngược lại khi các các điểm ảnh này của $N(P)$ được đặt theo thứ tự ngược chiều kim đồng hồ

Hilditch lại định nghĩa số giao là số lần nhảy từ điểm trắng sang điểm đen khi các điểm này đang được đặt trong thứ tự cắt góc giữa kẻ 4 đen và 4_láng giềng.

Có thể thấy rằng , đối với cả 2 định nghĩa , 1 điểm ảnh có cả 8_láng giềng đều là đen thì sẽ có số giao bằng 0, như một điểm ảnh bị cô lập. Nếu $X_H(p) = 1$ thì việc xóa p đi không làm thay đổi tính 8_liên thông của mẫu.

1.7) Số liên thông

Ta có thể tính số 8_liên thông theo định nghĩa :

$$N_c^8 = \sum_{i=1}^4 (\bar{x}_{2i-1} - (\bar{x}_{2i-1} \cdot \bar{x}_{2i} \cdot \bar{x}_{2i+1}))$$

Với \bar{x} là phủ định của x, mặt khác với số 4_liên thông :

$$N_c^4 = \sum_{i=1}^4 (x_{2i-1} - (x_{2i-1} \cdot x_{2i} \cdot x_{2i+1}))$$

Các điểm ảnh được giữ lại (khi $N_c^8(p) > 1$) để không làm mất tính liên thông.

Các điểm ảnh có thể xóa được lại thường được gọi là đơn, một điểm ảnh biên không bị cô lập p là đơn nếu $N(p)$ có duy nhất một thành phần đen, điều này tương đương với $X_H(p)=1$.

Các điểm ảnh với số liên thông $N_c^8(p)$ lớn hơn 1 thuộc vào loại điểm ảnh bội. Chúng bao gồm các điểm cuối các nhánh, các nét vẽ có độ dày 2 điểm ảnh, các điểm ảnh được qui cho xương dựa trên tiêu chuẩn liên thông.

2) Xác định các láng giềng của một điểm ảnh

Để xác định điểm ngã ba ta chỉ cần tìm tất cả các điểm có nhiều hơn 3 láng giềng . Dưới đây là phần code demo để xác định các láng giềng của 1 điểm ảnh :

```

void IMAGE::LangGieng(char*pCot,struct LangGieng_8 *pixel,int
type=DEN){
    unsigned char next;
    pixel->nhomBienDen_8=0;
    pixel->nhomBien_4=0; pixel->nhomBienDen_4=0;
    pixel->langGieng_8=0; pixel->langGiengDen_8=0;
    pixel->langGieng_4=0; pixel->langGiengDen_4=0;

    for(unsigned char i=0;i<8;i++)
    {
        // tinh diem lang gieng
        if(pCot[bien[i].dpos]==DEN
        ||type>=0&& pCot[bien[i].dpos]==type
        ||type<0&& pCot[bien[i].dpos]!=TRANG)
        {
            pixel->langGieng_8++;
            if( (i&01)==1)// i le
                pixel->langGieng_4++;
        }//if
        if(pCot[bien[i].dpos]==DEN)
        {
            pixel->langGiengDen_8++;
            if( (i&01)==1)

```

```

        pixel->langGiengDen_4++;
    }//if
    // tinh nhom bien lien thong_4
    if(type<0)
    {

        if(pCot[bien[i].dpos]!=TRANG&&pCot[bien[i+1].dpos]==TRANG)
            pixel->nhomBien_4++;
        }
        Else
        {

            if((pCot[bien[i].dpos]==DEN||pCot[bien[i].dpos]==type)&&pCot[bien[i+1]
.dpos]==TRANG)

                pixel->nhomBien_4++;
            }
            // tinh nhomBienDen_4 lien thong 4

            if((pCot[bien[i].dpos]==DEN)&&pCot[bien[i+1].dpos]!=DEN)

                pixel->nhomBienDen_4++; // tinh nhom bien lien
thong_8 diem den
o goc

                if((i&01)==0&&pCot[bien[i].dpos]!=DEN) continue;// trang
o goc

                if((i&01)==1&&pCot[bien[i+1].dpos]!=DEN) next=2;
                else
                    next=1;

            if(pCot[bien[i].dpos]==DEN&&pCot[bien[i+next].dpos]!=DEN

                ||pCot[bien[i].dpos]!=DEN&&pCot[bien[i+next].dpos]==DEN

```

```

        )
        pixel->nhomBienDen_8++;
    } //of for
    // hieu chinh lai nhomBien_4
    if(pixel->nhomBien_4==0)
    {
        if( type<0&&pCot[1]!=TRANG||
            type>=0&&(pCot[1]==DEN||pCot[1]==type)
        )
            pixel->nhomBien_4=8;// xung quanh toan diem DEN
    } // of if
    // hieu chinh lai nhomBienDen_4
    if(pixel->nhomBienDen_4==0&&pCot[1]==DEN) // xung quanh toan diem
DEN
        pixel->nhomBienDen_4=8;
        // xac dinh tinh lien thong
        if(pixel->nhomBienDen_8==0&&pCot[1]!=DEN||pixel-
>nhomBienDen_8>2)
            pixel->lienThong=0;// khong lien thong
        else pixel->lienThong=1;//lien thong
        return;
    }

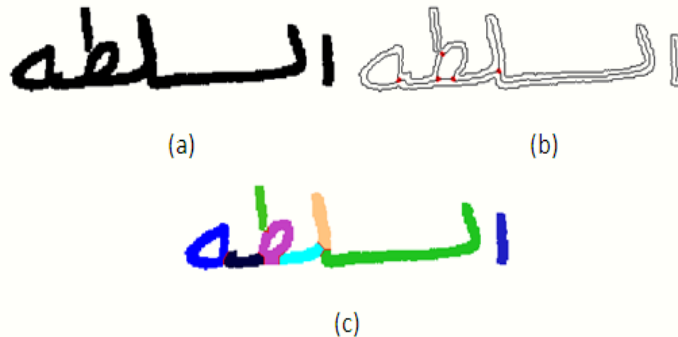
```

3.Điểm ngã ba:

Được định nghĩa như là một điểm giao nhau, một điểm pixel có 3 hoặc nhiều hơn các điểm lân cận, và các đường dẫn phân khúc có thể được định nghĩa theo các điểm giao nhau.

Trong giai đoạn phân khúc, kỹ thuật tách ngã ba dựa trên quyết định phân đoạn đầy đủ của các chuỗi chữ, số được kết nối. Sử dụng các giá trị mờ đặc trưng ở sự kết hợp đầy đủ các phân đoạn cô lập các phân đoạn chính (sáp nhập hoàn thành phân đoạn) từ các phân đoạn nhỏ. Các kết nối không mong muốn nằm trong phân khúc nhỏ của bộ xương ký tự.

Ở giai đoạn tách ký tự, tất cả các phân đoạn chính được kết hợp với nhau của các phân đoạn nhỏ để tạo ra các bản phác thảo kết nối khác nhau. Để nhận ra các đặc điểm cá nhân của chuỗi ký tự, các bản phác thảo này được xây dựng thành một hình ảnh ký tự mới đầu vào, sau đó có thể được sử dụng như một đầu vào cho một hệ thống nhận dạng ký tự.



Hình 3.1 Điểm ngã 3 và phân khúc trong ký tự

4. Phân đoạn tách nét dựa vào điểm ngã ba

Khi xem xét các ký tự được đưa vào phân đoạn, nó gần như là không thể phân đoạn thành phần ký tự trong xương của một ảnh. Điều này là do khả năng có kết dính giữa hai chữ số nhân vật khác nhau chuỗi. Do đó, giai đoạn phân đoạn của đề xuất công việc bao gồm hai giai đoạn cụ thể là, phân khúc giai đoạn ban đầu và phân chia tổng số pha. Trong phân khúc ban đầu, hình ảnh ký tự đầu vào trải qua xác định điểm ngã ba và phân đoạn dựa vào nó.

Trong nghiên cứu này đã được tìm thấy rằng, ban đầu phân khúc sẽ không hoàn chỉnh do kết nối điểm không tiếp giáp giữa phân khúc bộ xương. Như ví dụ hình ảnh 3.1 trên, ký tự viết tay trong bộ xương trên sẽ không được tách ra thành một "âm tính nghiêng" và hai "Ngang Lines" vì điểm không ngã ba kết nối ở giữa.

Để bù đắp cho điều này, một thuật toán phân đoạn riêng biệt được sử dụng với các phân khúc dựa trên quy tắc phương pháp tiếp cận. Để hiểu rõ thuật toán này ta cần tìm hiểu 1 số định nghĩa:

Định nghĩa 1 : Một điểm khởi là một điểm pixel trên bộ xương ký tự mà qua

bộ xương có thể được bắt đầu .khởi điểm có hai mặt chính khởi điểm và các điểm khởi nhỏ

Định nghĩa 2: Một điểm khởi đầu chính là một điểm khởi đầu được xác định trước khi bắt đầu traversal thông qua bộ xương.

Định nghĩa 3: Một điểm khởi nhỏ là một điểm khởi đầu được xác định trong cây qua bộ xương.

Định nghĩa 4: hướng traversal là hướng từ các điểm ảnh hiện tại pixel tiếp theo sẽ được truy cập trong traversal.

Định nghĩa 5: Một điểm cuối cùng là một điểm pixel trong khu vực tương quan, trong đó không có điểm ảnh lân cận để thăm tiếp theo .Sau khi tìm thấy tất cả các điểm khởi lớn, và các điểm giao nhau, các thuật toán bắt đầu đi ngang qua bộ xương ký tự kết dính, bắt đầu từ khởi điểm chính của danh sách các điểm khởi.Trong quá trình tách ban đầu này, các phân đoạn được xác định trong đường traversal dựa trên danh sách điểm giao nhau.

Một khi các traversal đạt đến một điểm đường giao nhau, hoặc một điểm kết thúc, mà là một điểm pixel không có điểm ảnh lân cận để thăm tiếp theo, tập trung được chuyển sang khởi động nhỏ được xác định điểm trong hàng đợi các khởi điểm nhỏ. Sau đó, các thuật toán bắt đầu đi qua những con đường ghé thăm thường xuyên của bộ xương bằng cách bắt đầu với mỗi điểm khởi nhỏ trong hàng đợi các khởi điểm nhỏ.Trong các traversal này, các thuật toán cũng phân đoạn con đường truy cập đến ngã ba điểm hoặc một điểm kết thúc thành các đoạn ban đầu.

Quá trình nêu trên được tiếp tục với tất cả các khởi điểm chính chưa được ghé thăm trong hàng đợi các khởi điểm chính, cho đến khi tất cả các con đường có liên quan trong khu vực truy cập được ghé thăm thường xuyên.

Dưới đây là thuật toán :


```

void IMAGE::CutThin(int MaxX,int MaxY,Pixel* storeXY,Pixel
*storeWhite)
{
    char *pCot,*pDong,*pPic;
    long j,i;
    int count=0;
    int ccount=0;
    char k;                                // chi so vong for

    int index=0;

    if(!hPic) return;
    pPic=(char*)hPic;
    if(!pPic) return;

    for(j=1,pDong=pPic+1+MaxX;j<MaxY-1;j++,pDong+=MaxX)
    {
        for(pCot=pDong,i=1;i<max-1;i++,pCot++)
        {
            count=0;

            if(pCot[0]==DEN)
            {
                for(k=1;k<9;k++)
                {
                    if(pCot[bien[k].dpos]==DEN&&
1].dpos]==TRANG)
                    {
                        count+=1;
                    }
                }
            }// of for_k
            if(count>=3)
            {
                storeXY[index].x=i;
                storeXY[index].y=j;

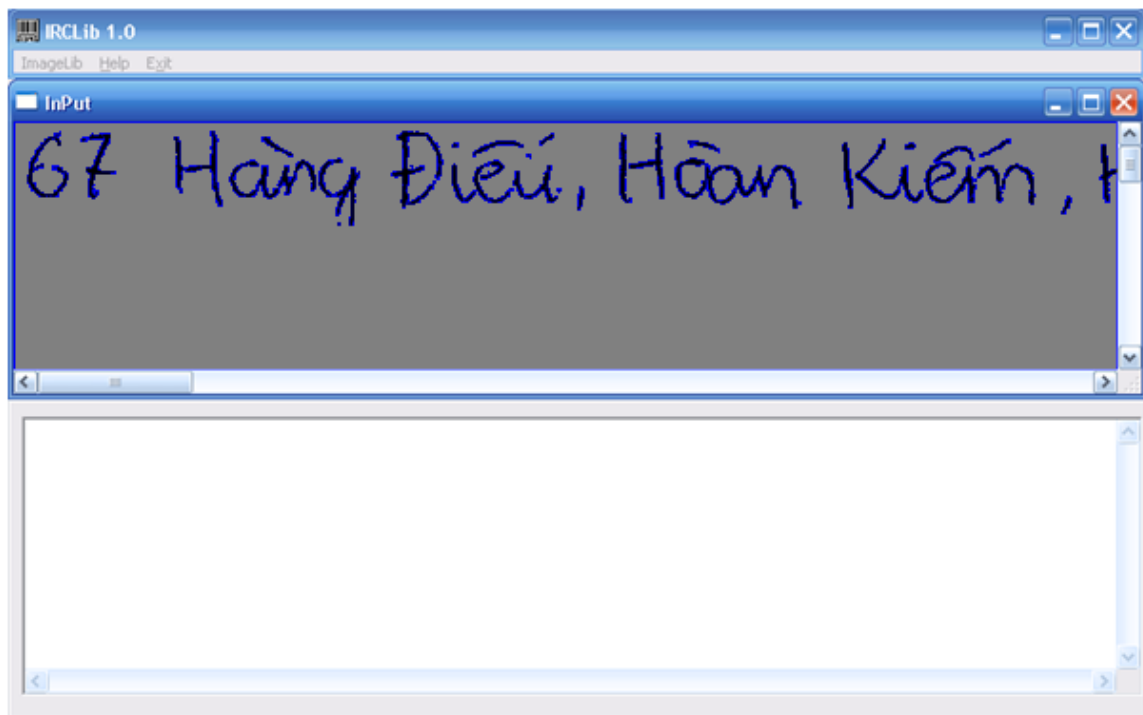
                ccount=0;
                for(k=1;k<9;k++)
                {
                    if(pCot[bien[k].dpos]==TRANG&&ccount<5)
                    {
                        storeWhite[index*5+ccount].x=bien[k].d
x;

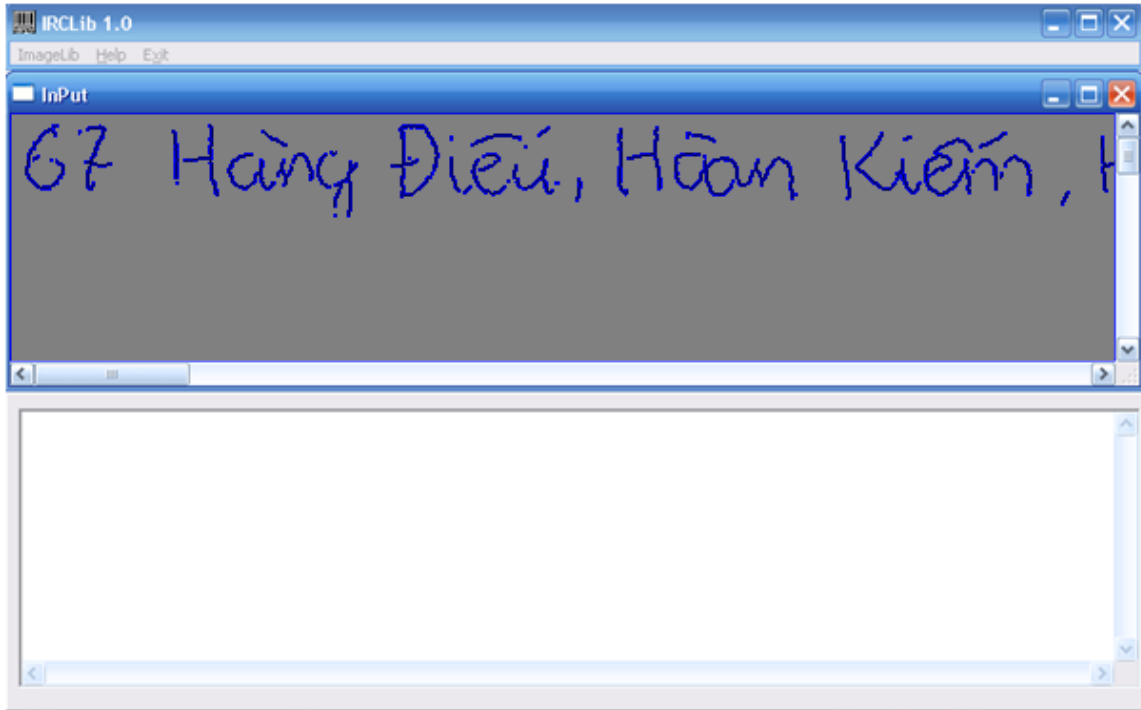
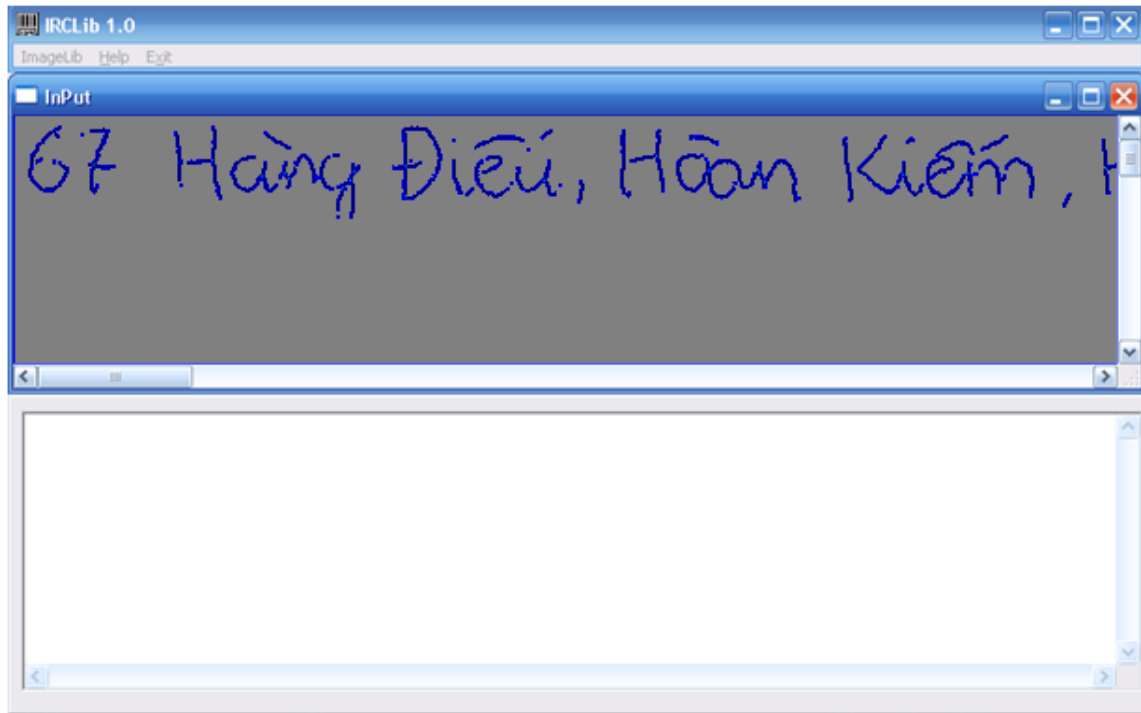
```

```
        storeWhite[index*5+ccount].y=bien[k].d
        y;
        ccount+=1;
    }
}
index++;
}
}
}
}
}
```

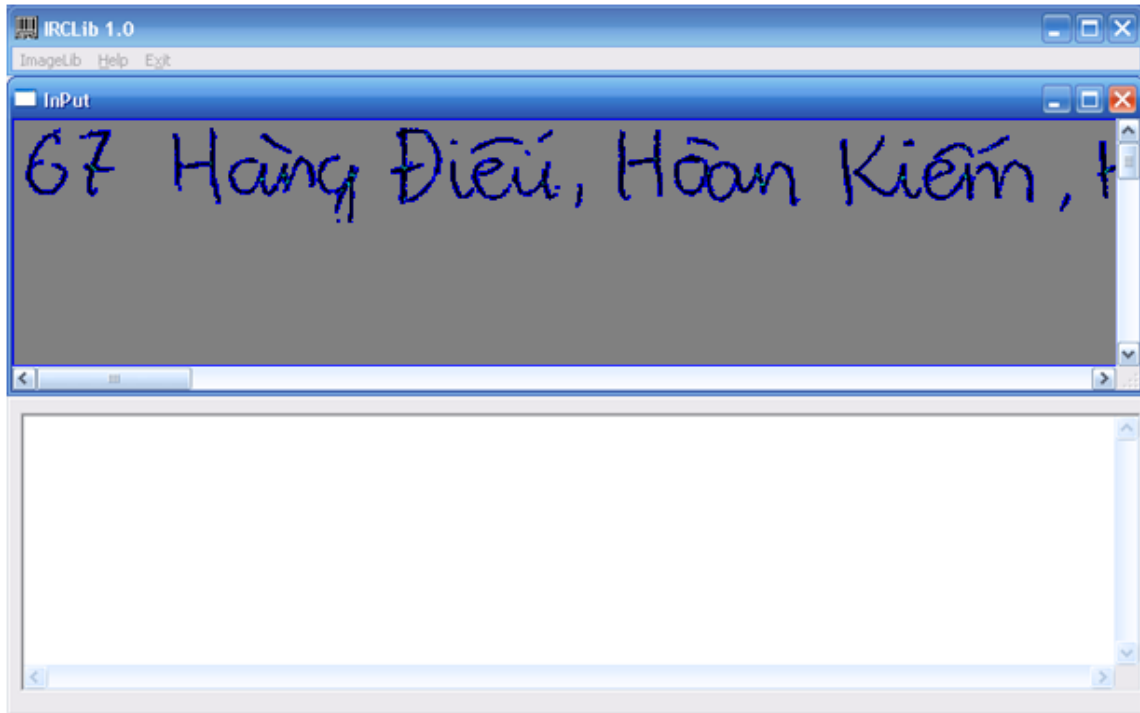
CHƯƠNG V : CHƯƠNG TRÌNH VÀ MỘT SỐ KẾT QUẢ THỬ NGHIỆM

Bước 1 : Load ảnh

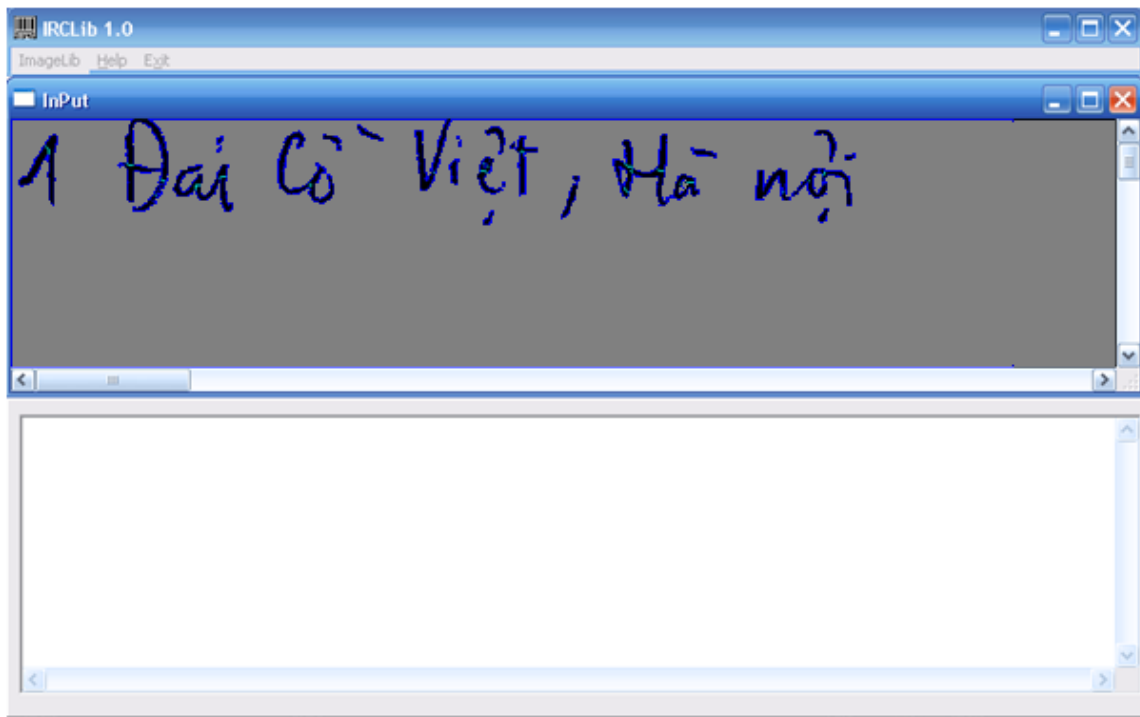


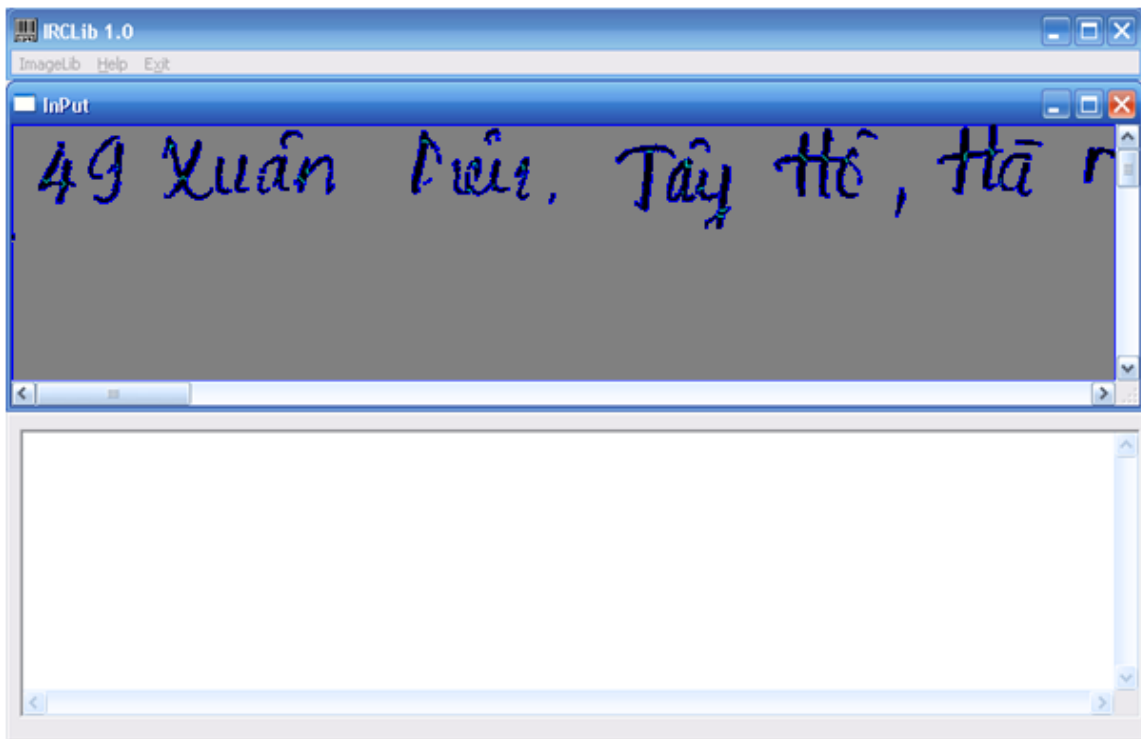
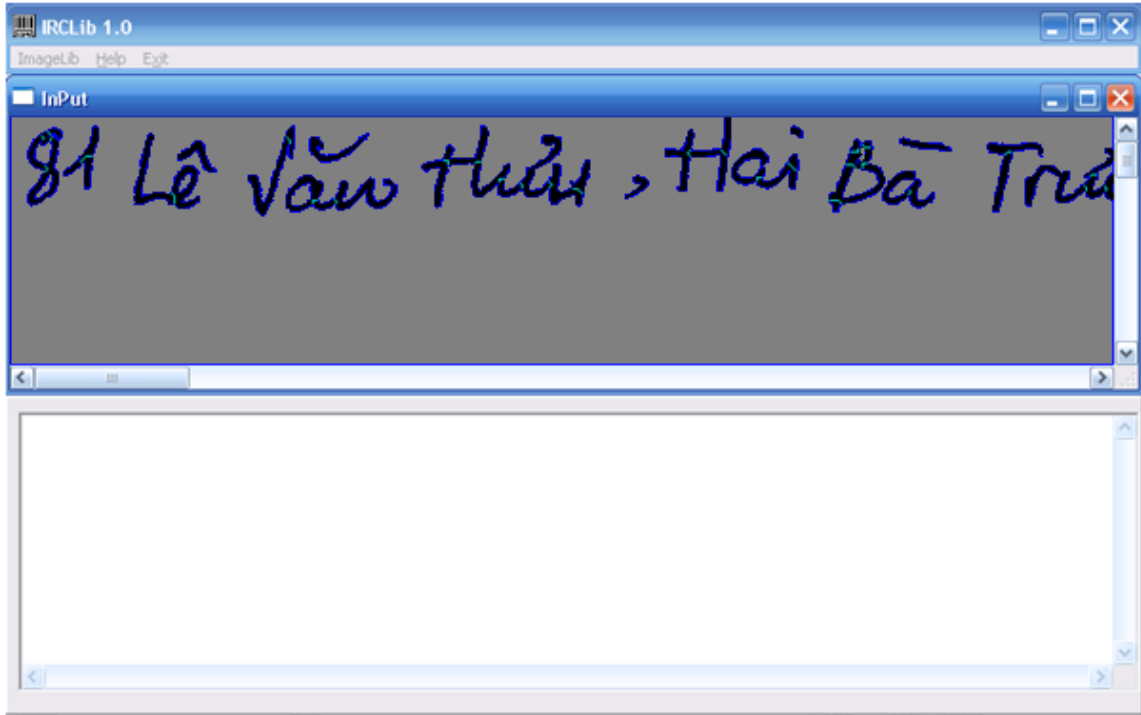
Bước 2 : Làm mảnh ảnh**Bước 3 : Phân đoạn tách nét và đánh dấu những điểm ngã ba**

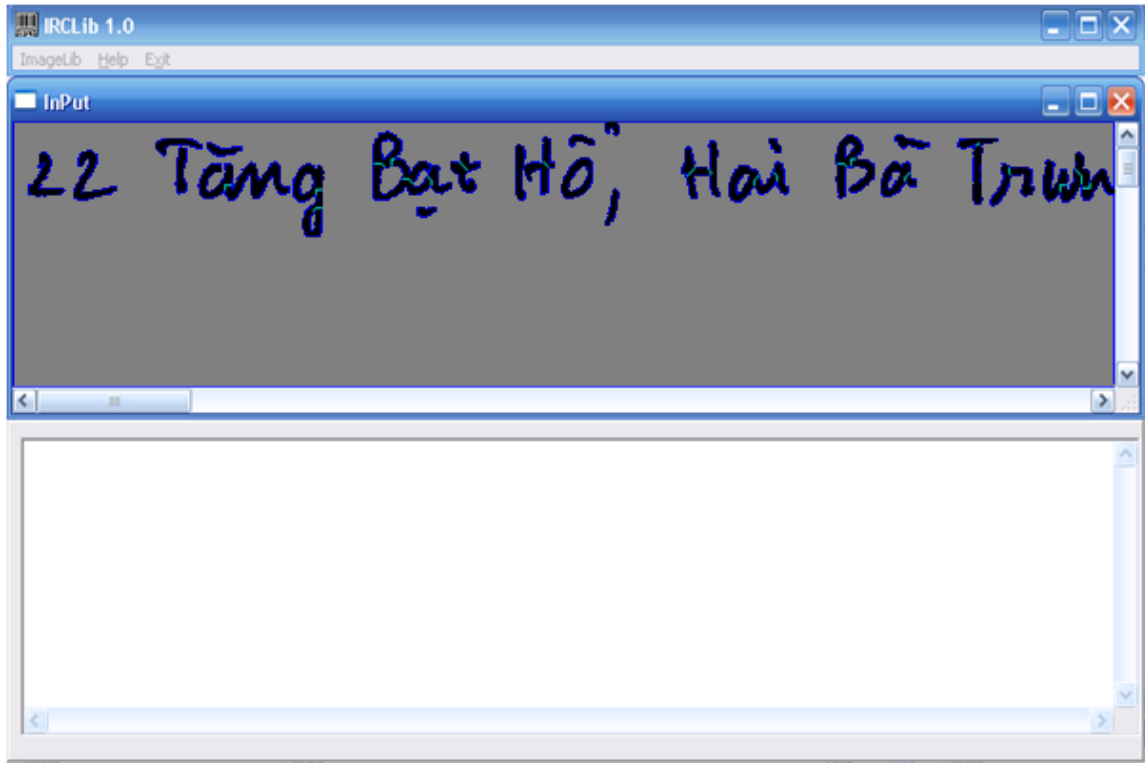
Đây là kết quả phân đoạn tách nét khi chưa làm mảnh :



Một số kết quả thử nghiệm khác :







Kết luận: Qua các trình bày ở trên đã cho thấy rằng đề xuất thuật toán tách các ký tự viết tay khỏi kết dính là cực kỳ đáng tin cậy và phương pháp tương đối đơn giản để tạo ra các mô tả mờ của mô hình ký tự bị kết dính. Các thuật toán phân khúc phát triển cho công việc này là có khả năng tách các kết nối cho các ký tự viết hoa . viết tay thành các đoạn có ý nghĩa chính xác. Tỷ lệ nhận dạng cao, khi đưa ký tự vào hệ thống để xác định các kết dính giữa các ký tự, trong đó có kết dính không mong muốn.

Mặc dù các phương pháp được sử dụng trong công việc này không đối phó với hai kết nối chồng từ hai kẻ chữ số chạm vào từ đầu đến cuối, hệ thống vẫn có thể được sử dụng cho việc tách của bất kỳ kết nối ký tự viết tay nào với các kết nối đơn hoặc kép. Bài toán được đề nghị là một phương pháp phát triển xác định, đó là có khả năng tách các nét kết dính của các ký tự viết tay đã được đề cập ở trên.

TÀI LIỆU THAM KHẢO

- [1] “Nhập môn xử lý ảnh số”, Lương Mạnh Bá – Nguyễn Thanh Thủy, Nhà xuất bản Khoa Học Kỹ Thuật, 1999.
- [2] “Nhận dạng chữ viết tay rời tiếng Việt”, luận án thạc sĩ, Phạm Đại Xuân, Đại học Khoa Học Tự Nhiên TPHCM.
- [3] Galaxy Bansal Dharamveer Sharma, Isolated Handwritten Words Segmentation Techniques in Gurmukhi Script, ©2010 International Journal of Computer Applications (0975 - 8887), Volume 1 – No. 24.
- [4] Dharam Veer Sharma and Gurpreet Singh Lehal, An Iterative Algorithm for Segmentation of Isolated Handwritten Words in Gurmukhi Script, The 18th International Conference on Pattern Recognition (ICPR'06)
- [5] A Junction Based Segmentation Algorithm for Offline Handwritten Connected Character Segmentation of UKS Jayarathna GEMDC Bandara Department of Statistics and Computer Science, Faculty of Science, University of Peradeniya, Sri Lanka
- [6] Segmentation of Arabic Handwriting based on both Contour and Skeleton Segmentation Safwan Wshah, Zhixin Shi and Venu Govindaraju Department of Computer Science and Engineering University at Buffalo, Amherst, NY, USA