

BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG  
-----o0o-----

**TRA CỨU ẢNH DỰA TRÊN LƯỚI ĐỒ  
KHOẢNG CÁCH VÀ BIỂU DIỄN HÌNH DẠNG**

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY**

Ngành: Công nghệ Thông tin

Sinh viên thực hiện: **TRẦN NGỌC D- ỜNG**

Giáo viên hướng dẫn: **PGS - TS. NGÔ QUỐC TẠO**

Mã số sinh viên: 1013101001

## LỜI MỞ ĐẦU

Cùng sự phát triển của công nghệ thông tin, mạng internet toàn cầu là sự gia tăng nhanh chóng của các dữ liệu đa phương tiện, khiến cho nhu cầu chia sẻ và tìm kiếm những loại tài nguyên này cũng tăng theo một cách nhanh chóng. Một loại tài nguyên trong đó là các dữ liệu kiểu hình ảnh. Mỗi người tìm kiếm hình ảnh đều có mục đích khác nhau nhưng chung quy lại cái người dùng muốn tìm kiếm là những thông tin kèm theo và hình ảnh họ đang cần. hoặc một số khác lại tìm kiếm hình ảnh để xác nhận tính chính xác của thông tin họ nhận được, cũng có thể là tìm kiếm chỉ với mục đích giải trí, thỏa mãn nhu cầu cá nhân. Nó phục vụ cho nhiều lĩnh vực quan trọng trong cuộc sống như trong các hệ thống bảo mật, an ninh, y tế, hay các hệ thống phát hiện chuyển động ... Vì thế việc nghiên cứu và phát triển các hệ thống tra cứu ảnh ngày càng trở nên cấp thiết.

Có 2 kiểu tìm kiếm đó là tìm kiếm theo từ khóa và tìm kiếm theo nội dung ảnh, tìm kiếm theo từ khóa dễ thỏa mãn được nhu cầu người dùng với các nhu cầu tìm kiếm hình ảnh mới theo như mong muốn xuất hiện trong suy nghĩ của họ, tìm kiếm theo từ khóa thì nhanh hơn tìm kiếm theo nội dung bởi vì nó hoạt động trên việc phân tích và so sánh các từ hoặc cụm từ tương ứng với nhau để đưa ra kết quả, kiểu dữ liệu này dạng các văn bản, từ ngữ cho nên nhanh chóng đưa ra được kết quả, và không đòi hỏi người dùng phải có ảnh mẫu. Phương pháp này có nhược điểm là hình ảnh kết quả không phải lúc nào cũng chính xác, và nó phù hợp nhất với việc đáp ứng nhu cầu của người dùng thông qua các mô tả bằng từ ngữ. Một phương pháp khác để tra cứu hình ảnh là tra cứu theo nội dung của hình ảnh. Phương pháp này cần một ảnh mẫu cho đầu vào để tìm ra những bức ảnh tương ứng. Phương pháp này cho kết quả tốt hơn về tính đúng đắn, bởi vì thông qua nội dung của bức ảnh sẽ được biểu diễn và đưa ra những kết quả tương ứng với nội dung bức ảnh đầu vào. Nó đáp ứng tốt hơn cho người dùng, tuy nhiên người dùng sẽ cần phải có một ảnh mẫu để trích chọn và biểu diễn các đặc trưng trong bức ảnh đó trước khi tìm kiếm. Tra cứu ảnh theo nội dung là một phương pháp phù hợp với những hệ thống máy tự động, hoặc các hệ thống an ninh nơi mà họ cần những kết quả hình ảnh tương tự với thông tin được lấy trực tiếp từ nội dung của ảnh.

Nói chung, đối với hệ thống này người dùng sẽ cung cấp ảnh truy vấn và hệ thống sẽ trả về kết quả là tập các ảnh tương tự. Do đó, làm thế nào để mô tả và mô hình một hình ảnh, để so sánh các ảnh khác nhau, để đánh chỉ số cho các ảnh trong cơ sở dữ liệu, và để tìm kiếm ảnh một cách hiệu quả là một vấn đề hết sức quan trọng.

Một ảnh có thể được mô tả theo các đặc trưng mức thấp (low level features). Các đặc trưng đó, bao gồm hình dạng, màu sắc, kết cấu và mối liên hệ không gian, đó còn được gọi là nội dung của ảnh. Bằng việc sử dụng các đặc trưng, chúng ta không chỉ mô tả và mô hình một ảnh, mà còn dùng để so sánh các bức ảnh với nhau. Vì thế, một hệ thống tra cứu ảnh theo các đặc trưng mức thấp còn gọi là hệ thống tra cứu ảnh theo nội dung (CBIR).

Ảnh dùng truy vấn được chia làm nhiều loại, mỗi loại mang một đặc trưng trội, chúng ta sẽ có các phương pháp khác nhau để phân tích và đạt được hệ thống tra cứu có kết quả tốt nhất, cho thí dụ như ảnh vân gỗ, vân vải có đặc trưng riêng về kết cấu và hướng, còn ảnh thiên nhiên lại mang nhiều đặc trưng màu sắc với sự bài trí phức tạp, thường thì phương pháp này người chúng ta sử dụng lược đồ màu sắc dựa trên màu trong vùng hoặc toàn bộ ảnh để tìm kiếm thì sẽ

đạt hiệu quả tốt, đối với hình ảnh mang đối tượng và bố cục độ phức tạp không cao nhưng đòi hỏi về sự thay đổi vị trí, thay đổi về kích thước theo tỷ lệ, hay góc quay đối tượng thì lại cần tới phương pháp trích chọn và biểu diễn theo hình dạng đối tượng. Đề tài này tập trung vào loại ảnh mang đặc trưng hình dạng đối tượng. Đã có rất nhiều phương pháp được đề xuất để biểu diễn hình dạng, tuy nhiên có những nhược điểm như khó có thể bảo toàn được tính bất biến khi quay, thu nhỏ, hay vị trí của đối tượng, thí dụ phương pháp dựa trên góc quay, phương pháp dựa trên lưới... phương pháp được đề xuất trong đề tài này là phương pháp có thể đảm bảo được tính bất biến hình dạng đó.

Báo cáo được chia làm 4 chương :

**Chương 1:** Giới thiệu về tra cứu ảnh, các đặc trưng của ảnh và cấu trúc hệ thống tra cứu ảnh.

**Chương 2:** Xây dựng cơ sở lý thuyết phương pháp biểu diễn hình dạng đối tượng theo lược đồ khoảng cách và tính toán độ tương tự giữa hai ảnh truy vấn.

**Chương 3:** Từ cơ sở lý thuyết đã xây dựng ở **Chương 2** để đưa ra ý tưởng, thuật toán và áp dụng thử nghiệm đưa ra kết quả, và đánh giá hiệu năng.

## LỜI CẢM ƠN

Em xin chân thành cảm ơn PGS, TS. Ngô Quốc Tạo, người đã trực tiếp hướng dẫn và tận tình giúp đỡ em trong quá trình thực hiện đề án này, những kiến thức, và phương pháp nghiên cứu em học từ Thầy thực sự rất quý giá, không những giúp ích cho em ở hiện tại mà còn là tiền đề để em có thể tiếp thu kiến thức mới một cách tốt hơn, một lần nữa em xin cảm ơn Thầy rất nhiều. Em xin cảm ơn Thạc sỹ Ngô Trường Giang vì thông qua môn học Đồ họa máy tính và Xử lý ảnh đã giúp em có niềm đam mê với lĩnh vực đồ họa máy tính, những kiến thức từ hai môn học đã góp phần giúp em hoàn thành đề án này.

Em xin gửi lời cảm ơn tới khoa CNTT trường ĐHDL Hải Phòng, vì trong thời gian học tập ở trường em đã học hỏi được những kiến thức, và tư duy, giúp em phát triển ý tưởng trong đề này.

Cuối cùng em xin gửi lời cảm ơn tới Gia đình và bạn bè đã bên cạnh giúp đỡ đồng thời ủng hộ em trong quá trình thực hiện đề án này.

Hải Phòng, tháng 6 năm 2012

Sinh viên thực hiện

**Trần Ngọc Dương**

## MỤC LỤC

LỜI MỞ ĐẦU

LỜI CẢM ƠN

MỤC LỤC ..... 12

DANH MỤC CÁC HÌNH ..... 14

KÝ HIỆU CÁC CỤM TỪ VIẾT TẮT ..... 15

Chương 1: TỔNG QUAN VỀ TRA CỨU ẢNH ..... 16

1.1 Giới thiệu về hệ thống tra cứu ảnh ..... 16

1.2 Hệ thống CBIR ..... 18

1.3 Ứng dụng của CBIR ..... 20

1.4 Cấu trúc của hệ thống CBIR ..... 20

1.5 Kết luận chương ..... 21

Chương 2: BIỂU DIỄN HÌNH DẠNG ..... 22

2.1 Giới thiệu về biểu diễn hình dạng ..... 22

2.2 Tầm quan trọng của biểu diễn hình dạng ..... 23

2.3 Xấp xỉ hình dạng đối tượng ..... 24

2.4 Trọng tâm của đa giác ..... 26

2.5 Chọn điểm mẫu ( Sample Point ) - tính khoảng cách giữa điểm mẫu và trọng tâm đa giác ..... 27

2.6 Lược đồ khoảng cách ..... 29

2.7 Chuẩn hóa ..... 30

2.8 Đo độ tương tự ..... 31

2.9 Kết luận chương ..... 32

<b>Chương 3: CÀI ĐẶT CHƯƠNG TRÌNH THỰC NGHIỆM .....</b>	<b>33</b>
<b>3.1 Cài đặt thuật toán.....</b>	<b>33</b>
3.1.1 Thu thập biên đa giác và lấy đỉnh đa giác .....	33
3.1.2 Tính diện tích và xác định trọng tâm của đa giác .....	36
3.1.3 Lựa chọn điểm mẫu và tính khoảng cách giữa trọng tâm đa giác và điểm mẫu	38
3.1.4 Chuẩn hóa khoảng cách .....	41
3.1.5 Xây dựng lược đồ khoảng cách .....	42
3.1.6 Độ đo tương tự .....	45
<b>3.2 Giao diện chương trình .....</b>	<b>46</b>
3.2.1 Giao diện tìm kiếm .....	46
3.2.2 Cơ sở dữ liệu ảnh .....	47
3.2.3 Lược đồ khoảng cách .....	48
<b>3.3 Kết luận chương .....</b>	<b>48</b>
<b>KẾT LUẬN .....</b>	<b>49</b>
<b>DANH MỤC TÀI LIỆU THAM KHẢO .....</b>	<b>50</b>

## DANH MỤC CÁC HÌNH

Số thứ tự	Nội dung	Số trang
Hình 1.1	Quá trình thực thi	17
Hình 1.2	Hình dạng đặc trưng	17
Hình 1.3	Hình kết cấu	18
Hình 1.4	Biểu diễn hình dạng qua mối liên hệ không gian	18
Hình 1.5	Cấu trúc hệ thống tra cứu ảnh theo nội dung	21
Hình 2.1	(a)Đa giác P, (b)P sau khi dịch chuyển, (c)P sau khi quay, (d)Thu nhỏ theo tỷ lệ	22
Hình 2.2	(a)Hình dạng và trọng tâm, (b)Hình dạng và trọng tâm sau khi di chuyển, (c)Hình dạng và trọng tâm sau khi quay	23
Hình 2.3	Mô tả hình dạng hình tròn	26
Hình 2.4	Quá trình mô phỏng đối tượng	27
Hình 2.5	Đa giác có n cạnh	28
Hình 2.6	Đa giác có n cạnh và có các điểm mẫu căng đều trên biên Li	29
Hình 2.7	Đa giác, điểm mẫu và các bán kính	29
Hình 2.8	Lược đồ khoảng cách của đa giác	30
Hình 2.9	Hai hình dạng tương tự nhưng kích thước khác nhau	30
Hình 2.10	Lược đồ khoảng cách sau khi chuẩn hóa	32
Hình 3.1	Một đa giác phóng to với mỗi ô tương ứng một điểm ảnh	37
Hình 3.2	Lược đồ khoảng cách đa giác hình 3.1	49
Hình 3.3	Giao diện tìm kiếm và kết quả	46
Hình 3.4	Cơ sở dữ liệu ảnh	47
Hình 3.5	Lược đồ khoảng cách	48

## KÝ HIỆU CÁC CỤM TỪ VIẾT TẮT

CBIR	Content Base Image Retrieval	Tra cứu ảnh dựa vào nội dung
QBIC	Query By Image Content	Truy vấn theo nội dung ảnh
SIM	Similar Image Measure	Độ đo ảnh tương tự



# Chương 1: TỔNG QUAN VỀ TRA CỨU ẢNH

## 1.1 Giới thiệu về hệ thống tra cứu ảnh

Hệ thống tra cứu ảnh là một hệ thống máy tính cho phép việc tìm kiếm và tra cứu ảnh từ một cơ sở dữ liệu ảnh số lớn. Hầu hết các phương pháp chung và truyền thống của việc tra cứu ảnh dựa trên một vài công thức về thêm metadata như là: Từ khóa, Chú thích hoặc Các miêu tả về ảnh. Sau đó việc tra cứu có thể được thực hiện qua các từ chú thích. Việc chú thích ảnh một cách thủ công là công việc tốn thời gian, công sức và tốn kém. Để giải quyết vấn đề này đã có nhiều nghiên cứu nhằm tự động hóa quá trình này. Ngoài ra, sự gia tăng của các ứng dụng trên mạng xã hội và mạng ngữ nghĩa đã thúc đẩy sự phát triển của hàng loạt các công cụ chú thích hình ảnh dựa trên nền web.

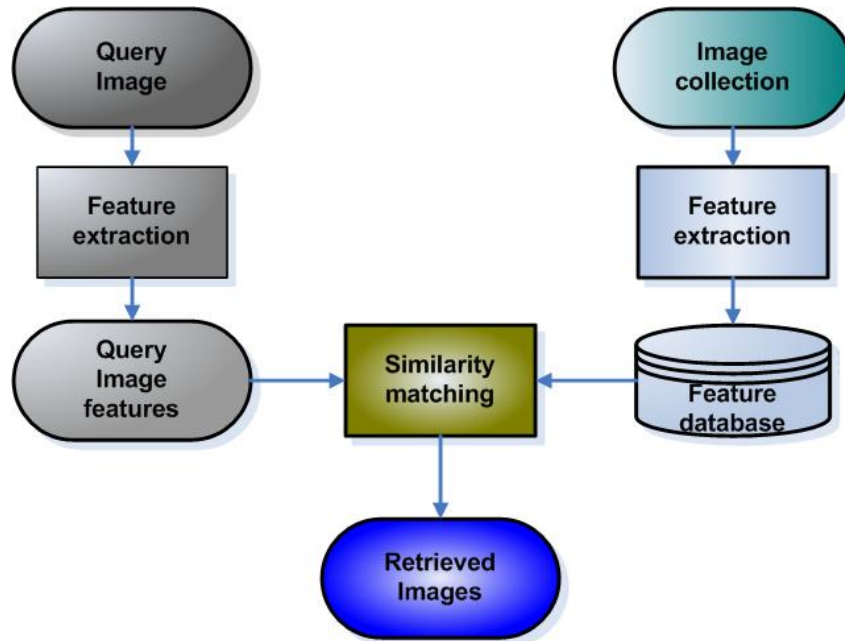
Hệ thống tra cứu cơ sở dữ liệu ảnh dựa trên microcomputer đầu tiên được phát triển tại Học viện công nghệ MIT vào những năm 1980 bởi Banireddy Prasaad, Amar Gupta, Hoo-min Toong, và Stuart Madnick.

Việc tìm kiếm ảnh là sự tìm kiếm dữ liệu chuyên biệt được sử dụng để tìm kiếm hình ảnh. Để tìm kiếm ảnh, người dùng có thể nhập vào các truy vấn như là: Từ khóa, File ảnh, Link ảnh hoặc bấm chuột vào một ảnh nào đó; sau đó hệ thống sẽ trả về các ảnh tương tự với truy vấn. Sự tương tự được sử dụng cho việc tìm kiếm có thể là: Các thẻ meta, Phân bố màu sắc, Thuộc tính vùng hoặc hình dạng.

Việc tra cứu ảnh có thể chia thành hai loại:

+ Tra cứu ảnh dựa trên từ khóa: Việc tìm kiếm dựa trên các dữ liệu liên quan như từ khóa, văn bản, ...

+ Tra cứu ảnh dựa trên nội dung (CBIR): Ứng dụng của thị giác máy (Computer Vision) vào việc tra cứu ảnh. Mục tiêu của CBIR là tránh sử dụng các miêu tả bằng từ và thay vào đó là sử dụng các sự tương tự trong nội dung của ảnh như: Kết cấu, Màu sắc, Hình dạng, ...



Hình 1.1: Quá trình thực thi

Quá trình thực thi của hệ thống tra cứu ảnh:

- + Người dùng đưa ra truy vấn hoặc ảnh có sẵn.
- + Hệ thống đón nhận truy vấn hoặc ảnh, sau đó trích chọn các đặc trưng.
- + Hệ thống so sánh truy vấn hoặc ảnh với cơ sở dữ liệu đặc trưng đã có.
- + Hệ thống trả ra kết quả tra cứu.

Một hệ thống tra cứu ảnh cần đáp ứng được:

- + Nhu cầu sử dụng hình ảnh của người dùng và thông tin đi kèm ảnh.
- + Cách mô tả nội dung ảnh.
- + Trích chọn đặc trưng từ ảnh.
- + Lưu trữ cơ sở dữ liệu ảnh.
- + Truy vấn và lưu trữ hình ảnh tương tự.
- + Truy xuất hình ảnh trong cơ sở dữ liệu hiệu quả.
- + Giao diện thân thiện, phù hợp.

## 1.2 Hệ thống CBIR

Thuật ngữ CBIR lần đầu tiên xuất hiện trên giấy tờ bởi T. Kato nhằm miêu tả việc tự động tra cứu ảnh từ một cơ sở dữ liệu dựa trên các đặc trưng nhìn thấy được như là màu sắc và hình dạng. Các đặc trưng mức thấp của ảnh trong cơ sở dữ liệu sẽ được trích chọn một cách tự động.

Tra cứu ảnh dựa theo nội dung (Content-based image retrieval - CBIR), còn được biết đến với tên gọi Truy vấn theo nội dung ảnh (Query by image content - QBIC) và Tra cứu thông tin theo nội dung trực quan (Content-based visual information retrieval - CBVIR) là một ứng dụng của kỹ thuật thị giác máy để giải quyết vấn đề tra cứu ảnh. Ở đây là tìm kiếm ảnh số trong cơ sở dữ liệu lớn.

Tra cứu ảnh theo nội dung trái ngược với đánh chỉ số ảnh dựa trên khái niệm (Concept-based image indexing ) còn được biết đến với tên gọi Dựa trên mô tả hoặc Dựa trên văn bản.

Việc tra cứu theo nội dung dựa trên một số đặc trưng mức thấp của ảnh (Low-level features): Màu sắc (Colors), Hình dạng (Shapes), Kết cấu (Textures) và Liên hệ không gian (Spatial relationship).

+ Màu sắc: Là đặc trưng được sử dụng phổ biến nhất, cho phép con người dễ dàng nhận ra sự khác biệt giữa hình ảnh. Lược đồ màu sắc (Color Histogram) là kỹ thuật thường dùng để biểu diễn.

+ Hình dạng: Là một đặc trưng khá quan trọng trong nội dung ảnh. Chúng ta có thể dễ dàng nhận dạng một đối tượng chỉ qua hình dạng của chúng. Có hai kiểu tiếp cận được sử dụng:

- Dựa trên vùng kín hình dạng.
- Dựa trên biên của đối tượng.



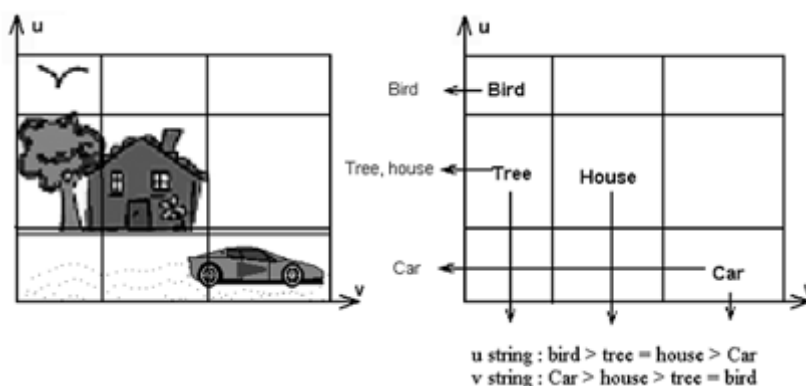
Hình 1.2: Hình dạng đặc trưng

+ **Kết cấu:** Cũng là một đặc trưng quan trọng trong tra cứu ảnh. Các thuộc tính của kết cấu: Tương phản, Tính thô, Hướng, Quy luật, Chu kỳ và Tính ngẫu nhiên.



Hình 1.3: Hình kết cấu

+ **Liên hệ không gian:** Được dùng nhiều trong xử lý ảnh, để phân biệt các đối tượng trong một ảnh. Có hai cách biểu diễn: Theo đối tượng và Theo quan hệ.



Hình 1.4: Biểu diễn hình dạng qua môi liên hệ không gian

Những phương pháp dựa trên những đặc trưng mức thấp được phát triển hoàn thiện trong thời gian gần đây. Tuy nhiên chúng chưa hẳn là tốt nhất. Để có được những hệ thống tra cứu hiệu quả, đáp ứng tốt nhu cầu sử dụng thì phải kết hợp và đưa ra những cách tiếp cận tốt hơn. Có những hệ thống tra cứu hiệu quả đối với đặc trưng màu sắc nhưng lại không hiệu quả trong những bức hình kết cấu, có những bức hình đạt hiệu quả tra cứu tốt trong tra cứu ảnh kết cấu nhưng đối với một vài loại ảnh có bố cục đầy đủ lại phải sử dụng phương pháp liên hệ không gian mới mong đạt được hiệu quả tốt nhất, nhiều hệ thống đã kết hợp các kỹ thuật khác nhau để đạt được hiệu quả tra cứu tối ưu.

Ngày nay, một vài hệ thống CBIR đã được đưa vào sử dụng: Cho thí dụ, hệ thống truy vấn theo nội dung ảnh QBIC của IBM được thiết kế vào đầu những năm 90, hệ thống đó hỗ trợ màu sắc và hình dạng, đặc biệt là độ đo tương tự cấu trúc. Ngoài ra có thể kể đến hệ thống Google Search Image của Google; Bing Image Search của Microsoft, ...

Trong bài báo cáo này chúng ta chỉ tập chung tìm hiểu và trích chọn hình dạng từ dữ liệu ảnh thô, cách mô tả, biểu diễn hình ảnh và tra cứu hình ảnh thông qua đặc trưng hình dạng của chúng.

### 1.3 Ứng dụng của CBIR

Ứng dụng của tra cứu ảnh có rất nhiều trong đời sống xã hội, phục vụ cho nhiều mục đích khác nhau, nhằm xác nhận, tra cứu thông tin. Giảm bớt công việc của con người nhằm tăng hiệu suất làm việc: Album ảnh số của người dùng, ảnh y khoa, bảo tàng ảnh, tìm kiếm nhãn hiệu, mô tả nội dung MPEG-7, ảnh tội phạm, hệ thống tự động nhận biết điều khiển giao thông, ...

Sau đây là một vài hệ thống lớn đại diện cho các lĩnh vực đặc trưng:

+ Hệ thống truy vấn ảnh theo nội dung (QBIC-query by image content) được nghiên cứu và phát triển bởi nhóm nghiên cứu Visual Media Management thuộc tập đoàn IBM, đây là một hệ thống tra cứu ảnh thương mại được phát triển từ rất sớm. Hiện nay, hệ thống này hỗ trợ một vài độ đo tương tự cho ảnh như: trung bình màu sắc, lược đồ màu sắc, và kết cấu. Công nghệ sử dụng trong hệ thống bao gồm 2 phần chính là: đánh chỉ số và tìm kiếm. Hơn nữa, hệ thống này còn cung cấp vài cách tiếp cận truy vấn theo đơn đặc trưng, đa đặc trưng và đa giai đoạn.

+ Hệ thống VisualSEEK tại trường đại học Columbia. Hệ thống cho phép người dùng nhập vào truy vấn, sử dụng các đặc trưng mức thấp của hình ảnh như: màu sắc, bố cục không gian, và kết cấu. Các đặc trưng đó được mô tả theo tập các màu sắc và biến đổi Wavelet dựa trên đặc trưng kết cấu.

+ Hệ thống NeTra sử dụng các đặc trưng của ảnh: Màu sắc, hình dạng, kết cấu, không gian.

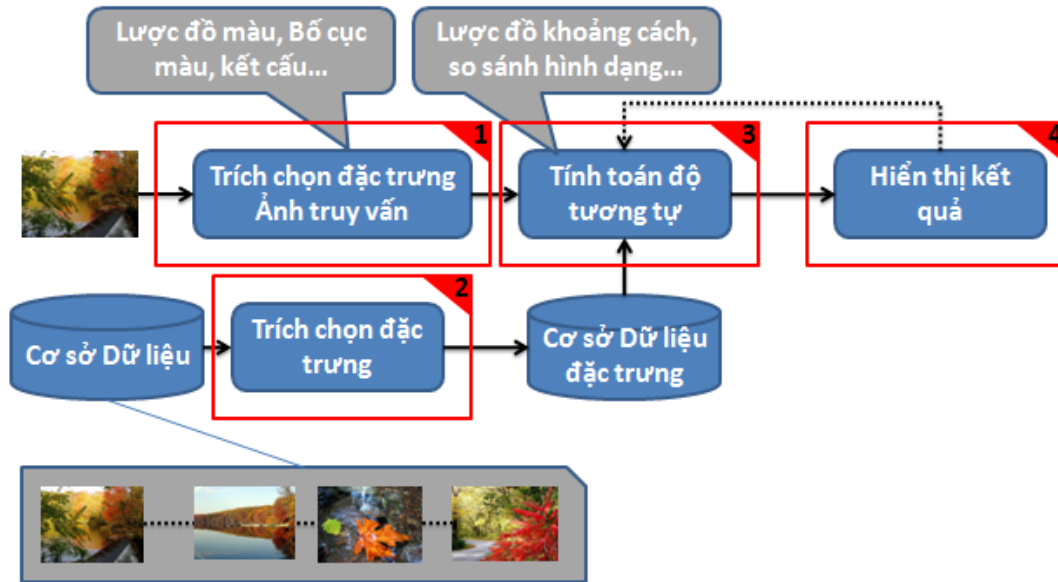
+ Ngoài ra còn một vài hệ thống khác như: Virage system, Stanford SIMPLICity system, NEC PicHunter system, ...

### 1.4 Cấu trúc của hệ thống CBIR

Một hệ thống tra cứu ảnh có thể thực hiện qua nhiều công đoạn: nhập ảnh truy vấn, nhập dữ liệu ảnh cho csdl, chuẩn hóa ảnh, trích chọn đặc trưng của ảnh truy vấn và ảnh trong cơ sở dữ liệu, tính toán độ tương tự và cách hiển thị kết quả lên màn hình .... Tuy nhiên chúng ta có miêu tả khái quát một hệ thống tra cứu ảnh thông qua những công đoạn chính sau – *Hình 1.5*:

1. Trích chọn đặc trưng cho ảnh truy vấn: Ở công đoạn này ảnh truy vấn ngay khi ảnh được nhập vào hệ thống sẽ xử lý để trích chọn đặc trưng theo đặc trưng nhất định nào đó và phục vụ tính toán độ tương đồng sau đó đưa ra kết quả, có thể nói công đoạn này sẽ được tính toán online.
2. Trích chọn đặc trưng: Đây là công đoạn tính toán đặc trưng cho ảnh trong cơ sở dữ liệu sinh ra cơ sở dữ liệu lưu trữ các đặc trưng, công đoạn này thường sẽ được tính toán từ khi nhập ảnh vào cơ sở dữ liệu, hoặc tiến hành khi người dùng cho phép thực hiện hay nói cách khác nó được tiến hành offline.

3. Đo độ tương đồng: Công đoạn này là công đoạn so sánh các ảnh tồn tại trong cơ sở dữ liệu và ảnh truy vấn thông qua đặc trưng đã trích chọn trước đó.
4. Tra cứu và hiển thị kết quả: Hiển thị kết quả vừa thu được cho người dùng theo một giá trị ngưỡng tương tự nào đó.



Hình 1.5: Cấu trúc hệ thống Tra cứu ảnh theo nội dung

Các thành phần cơ bản của hệ thống CBIR:

- Cơ sở dữ liệu ảnh: Là cơ sở dữ liệu phục vụ lưu trữ ảnh. Có thể là trên ổ cứng thường, cũng có thể là hệ quản trị cơ sở dữ liệu.
- Cơ sở dữ liệu đặc trưng: Các đặc trưng đã được trích chọn offline sẽ được lưu trữ trong cơ sở dữ liệu như tệp tin matlab, bảng tính excel ...

### 1.5 Kết luận chương

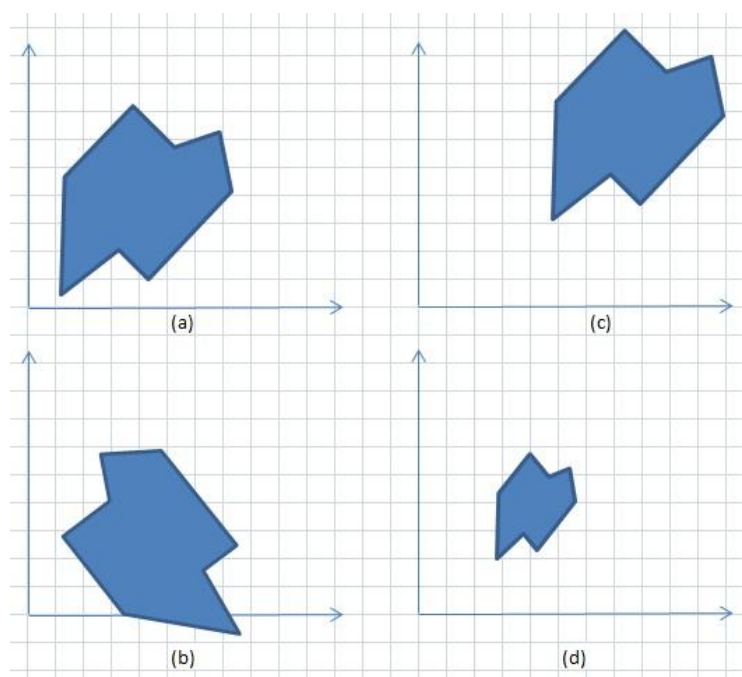
Tra cứu ảnh theo nội dung (CBIR) là một lĩnh vực khoa học được phát triển dựa trên cơ sở lý thuyết và ứng dụng của xử lý ảnh. Hệ thống cho phép người dùng tra cứu các ảnh tương tự trong một cơ sở dữ liệu hình ảnh. Các hình ảnh này có thể được thu thập thông qua các thiết bị chụp hình, cảm biến, và thiết bị quét hình ảnh, cũng có thể được chia sẻ thông qua hệ thống mạng máy tính toàn cầu.

Tra cứu ảnh theo nội dung là việc tính độ tương tự giữa hai bức ảnh được biểu diễn bởi một trong số các đặc trưng của ảnh như: Màu sắc, hình dạng, kết cấu... Kết quả là tập các bức ảnh tương tự với ảnh truy vấn được sắp xếp theo thứ tự giảm dần độ tương tự.

## Chương 2: BIỂU DIỄN HÌNH DẠNG

### 2.1 Giới thiệu về biểu diễn hình dạng

Thị giác con người có thể nhận biết đối tượng thông qua hình dạng của chúng, mặc dù những đối tượng này ở các trạng thái khác nhau trong môi trường cũng như tỷ lệ lớn nhỏ, vậy để đánh giá một phương pháp biểu diễn hình dạng tốt cũng phải thông qua tính chất bất biến khi dịch chuyển, quay và tỷ lệ kích thước, dễ dàng bổ sung, và đạt được kết quả giống kết quả nhận biết của thị giác con người (Hình 2.1).

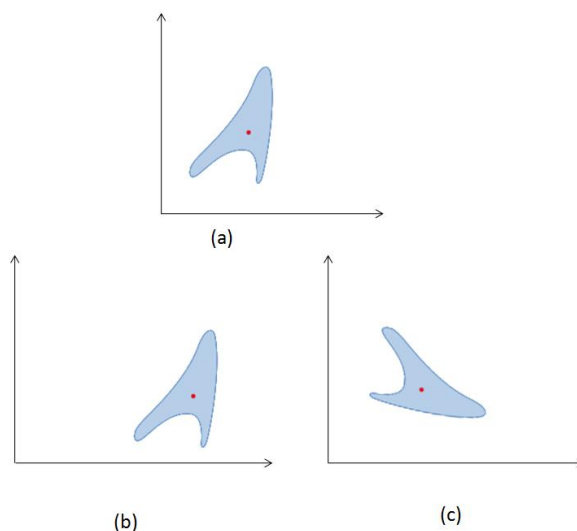


Hình 2.1: (a) Đa giác P (b) P Sau khi dịch chuyển (c) P sau khi quay (d) Sau khi thu nhỏ theo tỷ lệ

Khi xét đặc trưng hình học của hình dạng, chúng ta quan tâm tới tâm của hình dạng, nó rất quan trọng. Nói cách khác, với một hình dạng bất kỳ, vị trí tâm tương ứng với đường bao viền của chúng sẽ không thay đổi khi hình dạng được dịch chuyển hoặc quay.

Như chúng ta đã biết, đối với hình tròn thì khoảng cách từ trọng tâm tới điểm bất kỳ trên đường bao ngoài của chúng luôn bằng nhau, và khoảng cách này gọi là bán kính. Cho nên chúng ta có thể dễ dàng biểu diễn được hình tròn với đường viền bao và chỉ với một bán kính. Tuy nhiên, đối với hình dạng khác thì bán kính chưa đủ bởi vì khoảng cách từ các điểm khác nhau từ biên tới trọng tâm có thể sẽ khác nhau. Do đó chúng ta phải sử dụng tới tập hợp các bán kính, bán kính này bắt đầu từ trọng tâm của hình dạng và kết thúc ở đường viền bao của hình dạng, mới đủ để biểu diễn được hình dạng.

Từ đó, chúng ta có thể đưa ra một phương thức mô tả một hình dạng. Đồng thời thảo luận phương pháp so sánh và tìm kiếm hình dạng.



Hình 2.2: (a) Hình dạng và trọng tâm (b) Hình dạng và trọng tâm sau khi di chuyển (c) Hình dạng và trọng tâm sau khi quay

Cho một ảnh có một đối tượng duy nhất, chúng ta giả sử đã trích chọn được biên của đối tượng, hình dạng của đối tượng .... Vấn đề chúng ta cần quan tâm là làm thế nào để mô tả một đối tượng khi đã trích chọn được biên.

Để phân biệt được các lớp đối tượng với nhau hoặc các trạng thái của đối tượng thì chúng ta cần mô tả được hình dạng của chúng thông qua một vài yếu tố như, viền bao đối tượng và tập các bán kính tính từ trọng tâm của đối tượng.

## 2.2 Tầm quan trọng của biểu diễn hình dạng

Đối với các ảnh đa màu sắc, và có bố cục phức tạp như hình ảnh thiên nhiên thì phương pháp truyền thống được sử dụng là phương pháp tra cứu thông qua độ tương tự giữa các lược đồ màu sắc của ảnh. Một số phương pháp cải tiến đạt hiệu quả cao hơn là chia bức hình thành nhiều khối sau đó mới xây dựng lược đồ, số khác có thể phân đoạn bức ảnh và tìm ra vùng màu sắc đặc trưng nhất của ảnh, đây gọi là tra cứu ảnh dựa trên các màu sắc chủ đạo của ảnh. Các phương pháp này đều hướng tới hiệu quả tìm kiếm hình có độ tương tự đạt yêu cầu cao nhất. Tuy nhiên, chưa thể đạt được tốc độ tra cứu tốt đối với cơ sở dữ liệu ảnh lớn. Chính vì thế, gần đây các nhà khoa học đã nghiên cứu dựa trên các phương pháp phân cụm ảnh để giảm thiểu thời gian thực hiện tra cứu.



Với ảnh tra cứu là ảnh thiên nhiên thì chúng ta có thể tìm kiếm thông qua phương pháp trên, nhưng đối với những bức ảnh đòi hỏi tìm kiếm các đối tượng cụ thể như hình dạng một phi cơ hay hình ảnh một loài động vật hoặc là một vật dụng trong gia đình, công sở hay bất cứ nơi nào nếu thực hiện theo phương pháp đó thì có thể kết quả sẽ không được như mong muốn. Trong trường hợp này, phương pháp biểu diễn hình dạng sẽ hiệu quả hơn.

Muốn phát hiện được đối tượng cụ thể thông qua hình dạng thì chúng ta cần phải biểu diễn hình dạng đó trước. Đã có rất nhiều phương pháp được đưa ra để biểu diễn hình dạng, như: Phương pháp dựa trên mô tả Fourier (Fourier Descriptors), phương pháp dựa trên lưới (Grid Descriptors), phương pháp dựa trên đôi hướng góc (Turning Angle), .... Trong tất cả các phương pháp biểu diễn hình dạng đã nêu, phương pháp dựa trên mô tả Fourier được quan tâm nhất, bởi vì việc sử dụng thông tin từ biên của đối tượng để chỉ ra hình dạng đối tượng là một ý tưởng đơn giản và hiệu quả, phương pháp này tập trung giải quyết các vấn đề như sự thay đổi về vị trí đối tượng, quay đối tượng và sự thay đổi về kích thước đối tượng theo một tỷ lệ nhất định. Về khía cạnh biểu diễn hình dạng thì phương pháp này là phương pháp gần với phương pháp được đề xuất trong báo cáo này.

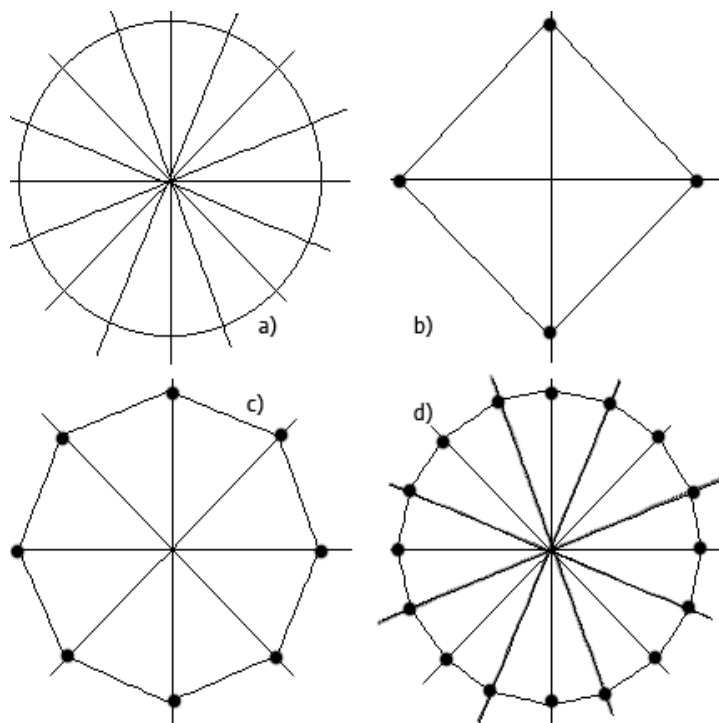
Phương pháp được đưa ra trong bản báo cáo này có 6 bước chính:

- + Xác định biên của đối tượng
- + Xác định trọng tâm của đối tượng.
- + Xác định các điểm mẫu trên biên của đối tượng.
- + Tính khoảng cách từ trọng tâm tới các điểm mẫu trên biên.
- + Chuẩn hóa khoảng cách và vẽ lược đồ khoảng cách.
- + Biểu diễn lược đồ khoảng cách và so sánh để đưa ra kết luận.

Trong đề tài này chúng ta sẽ tập trung vào giải quyết vấn đề sử dụng lược đồ khoảng cách để biểu diễn hình dạng bắt đầu từ xác định biên đối tượng.

### 2.3 Xấp xỉ hình dạng đối tượng

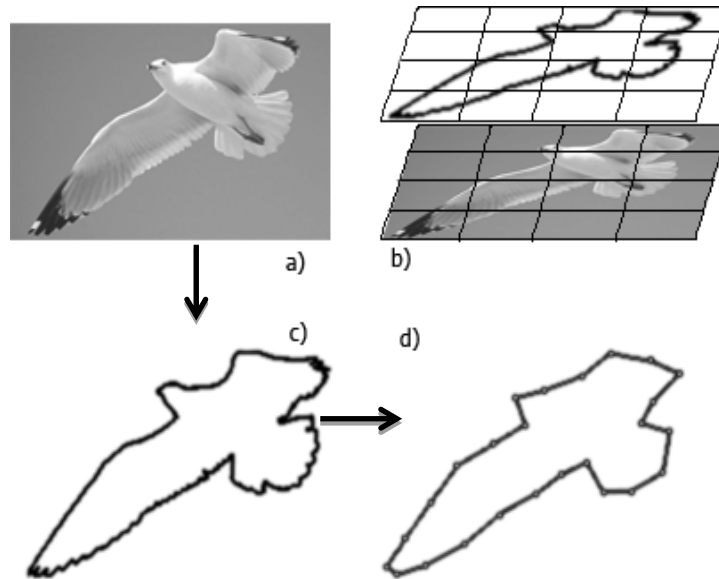
Phương pháp biểu diễn hình ảnh thông qua lược đồ khoảng cách thực hiện dựa trên các hình đa giác và trọng tâm của đa giác, cho nên trước khi đối tượng được biểu diễn chúng ta phải thực hiện tìm xấp xỉ của hình dạng đó ( Thuộc tính hình học). Phương pháp xấp xỉ hình dạng được nói tới trong đề tài này đã được sử dụng rất nhiều trong các công cụ thiết kế và mô phỏng hình dạng đối tượng. Để nhanh chóng nắm bắt được ý tưởng này đầu tiên chúng ta sẽ xét một hình tròn với các mức độ xấp xỉ khác nhau (Hình 2.3).



Hình 2.3: Mô tả hình dạng hình tròn, a) Với 4 điểm cơ bản, b) 8 điểm cơ bản, c) 16 điểm cơ bản

Quan sát hình 2.3 chúng ta rút ra một điều quan trọng là, khi số lượng điểm cơ bản trên biên của hình tròn càng tăng thì hình mô tả sẽ gần giống hơn đối với hình ảnh gốc, và các điểm biên cơ bản này luôn được căng đều trên biên đồng thời dây cung nối giữa các điểm này sẽ tạo lên đường mô phỏng hình dạng gốc, tất nhiên là số lượng điểm này không thể nhiều hơn số lượng điểm ảnh cấu thành nên hình tròn, lúc này các điểm cơ bản sẽ được cho là các đỉnh của một đa giác với nhiều cạnh, số lượng cạnh lớn thì kết quả tra cứu càng tốt, nhưng điều này sẽ dẫn đến việc hao tổn bộ nhớ và tăng thời gian biểu diễn hình dạng. Vì vậy tùy vào nhu cầu mà chúng ta sẽ đưa ra số lượng điểm cơ bản biểu diễn cho hợp lý.

Áp dụng cho một đối tượng cụ thể hơn, hình 4.2a) là ảnh gốc đầu vào đối tượng cụ thể là một con chim, hình 4.2 b) cho thấy việc tách bao viền ngoài, hình 4.2c) là bao viền ngoài của đối tượng được tách và lưu trong một mảng không gian hai chiều và hình 4.2d) cho thấy mô tả hình dạng bởi tập các điểm cơ bản.



Hình 2.4: Quá trình thực hiện mô phỏng đối tượng

Công việc xác định điểm cơ bản được thực hiện bằng cách, duyệt lần lượt các điểm ảnh bao viền theo thứ tự ngược chiều kim đồng hồ hoặc xuôi chiều kim đồng hồ. Thu được tổng số điểm ảnh bao viền ngoài của đối tượng, sau đó chia đều theo số điểm cơ bản cho trước theo công thức sau:

$$L_r = \frac{L_s}{N} \quad (\text{eq 4.1})$$

Với  $L_r$  là khoảng cách giữa các điểm cơ bản trên biên đã được làm tròn,  $L_s$  là tổng chiều dài của biên ảnh, và  $N$  là số lượng điểm cơ bản cho trước.

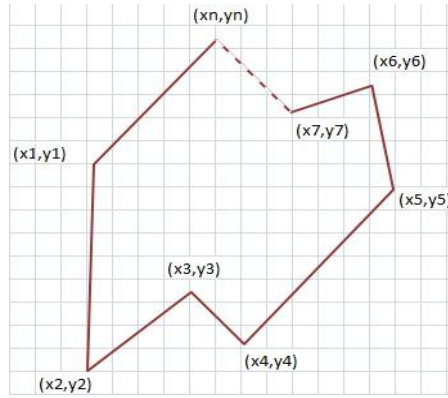
Cho thí dụ, nếu cho vào một hình dạng có tổng số lượng điểm ảnh là 31, và chúng ta cần đặt 5 điểm cơ bản thì chúng ta sẽ có:

$$L_r = 5$$

Thực hiện duyệt mảng chứa tọa độ các điểm ảnh theo thứ tự, và cứ 5 điểm chúng ta đặt một điểm ảnh làm điểm cơ bản, điểm bắt đầu chính là điểm kết thúc, chúng ta sẽ có đa giác 6 cạnh và 5 đỉnh.

## 2.4 Trọng tâm của đa giác

Hình đa giác có các đỉnh  $(x_i, y_i)$  với  $i = 0, 1, 2, \dots, n$ ,  $x_0 = x_n, y_0 = y_n$ .



Hình 2.5: Đa giác có  $n$  cạnh

Để tính được tâm của hình đa giác chúng ta cần tính được diện tích của đa giác đó, ở đây chúng ta sử dụng công thức tính diện tích của Gauss:

$$A = \frac{1}{2} \left| \sum_{i=1}^{n-1} x_i y_{i+1} + x_n y_1 - \sum_{i=1}^{n-1} x_{i+1} y_i - x_1 y_n \right| \quad (\text{eq. 1})$$

Với:  $A$  là diện tích của đa giác

$n$  là số mặt (số cạnh) của đa giác

$(x_i, y_i)$ ,  $i = 1, \dots, n$  là các đỉnh của đa giác.

Tiếp theo chúng ta sẽ tìm hiểu công thức tính trọng tâm của đa giác

Diện tích đa giác được tính theo cách này là một giá trị được đánh dấu theo các điều kiện sau: nếu dấu mang giá trị âm thì các đỉnh được xếp theo chiều quay kim đồng hồ, và ngược lại dấu giá trị âm thì các đỉnh được xếp theo chiều ngược kim đồng hồ. Bây giờ chúng ta sẽ tính trọng tâm dựa trên diện tích. Tọa độ trọng tâm được tính theo công thức:

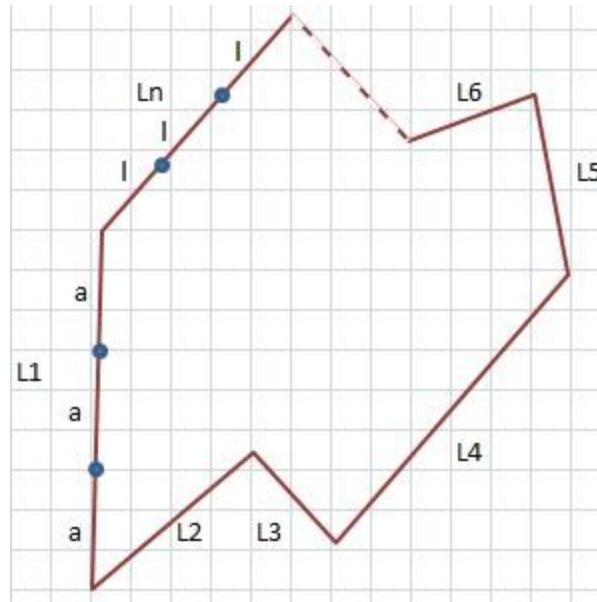
$$\bar{x} = \frac{1}{6A} \sum_{i=0}^{n-1} (x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \quad (\text{eq. 2})$$

$$\bar{y} = \frac{1}{6A} \sum_{i=0}^{n-1} (y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \quad (\text{eq. 2})$$

Trọng tâm đa giác là:  $C(\bar{x}, \bar{y})$

## 2.5 Chọn điểm mẫu - tính khoảng cách giữa điểm mẫu và trọng tâm đa giác

Điểm mẫu là tập hợp các điểm được chọn trên đường biên của hình dạng để mô tả đầy đủ cho một hình dạng. Tất nhiên không phải việc chọn ngẫu nhiên mà có thể mô tả được hình dạng thì là điều không phải, thực hiện theo nguyên tắc nhất định, nhiệm vụ cần giải quyết trong mục này là phương pháp lựa chọn điểm mẫu trên đường biên, và tính toán khoảng cách từ điểm điểm mẫu đó tới trọng tâm của đa giác.



Hình 2.6: Đa giác có  $n$  cạnh và có các điểm mẫu được căng đều trên mỗi biên  $L_i$

Các điểm mẫu có thể gán ngẫu nhiên trên biên. Tuy nhiên điều này có thể gây ra vấn đề: Có thể biên ngắn nhưng lại gán nhiều điểm mẫu, nếu hai đa giác có hình dạng giống nhau có thể có số lượng điểm mẫu khác nhau. Do đó chúng ta dựa theo độ dài của biên để xác định điểm mẫu.

Ví dụ một đa giác có  $n$  cạnh với  $L_i$  là chiều dài một cạnh biên,  $L_s$  là tổng chiều dài các biên và  $N$  là số lượng điểm mẫu dự kiến. Số lượng điểm mẫu trên cạnh biên  $i$  là:

$$N_i = \frac{L_i}{L_s} N \quad (\text{eq. 4})$$

Sau đó chúng ta bố trí các điểm mẫu này sao cho chúng chia đều cạnh biên  $i$  thành các đoạn bằng nhau.

Cuối cùng, dựa vào trọng tâm đã tìm được và các điểm mẫu, chúng ta tính khoảng cách giữa chúng. Chúng ta sử dụng công thức tính khoảng cách giữa hai điểm.

Ví dụ: Điểm mẫu có tọa độ  $P_i = (x_i, y_i)$  và trọng tâm có tọa độ  $C = (x_c, y_c)$

Khoảng cách giữa hai điểm:

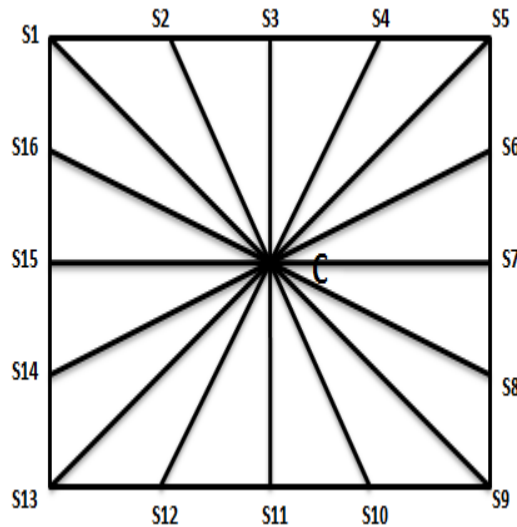
$$D(P_i, C) = \sqrt{(x_c - x_i)^2 + (y_c - y_i)^2} \quad (\text{eq. 5})$$

Từ đó chúng ta sẽ tính được khoảng cách từ trọng tâm tới tất cả các điểm mẫu nằm trên biên. Với các giá trị đó, chúng ta sẽ xây dựng được lược đồ khoảng cách (Distance Histogram) để mô tả một hình dạng.

## 2.6 Lược đồ khoảng cách

Lược đồ là một công cụ khá hữu dụng trong việc miêu tả các thuộc tính của dữ liệu, và chúng được ứng dụng khá rộng rãi trong các chương trình, đặc biệt trong lĩnh vực xử lý ảnh.

Như đã trình bày ở trên, các điểm mẫu sẽ được đặt cách đều nhau trên biên của đa giác, chúng ta đặt khoảng cách giữa trọng tâm đến một điểm mẫu trên biên là  $D$ , và khoảng cách lớn nhất từ trọng tâm đến điểm mẫu là  $D_{\max}$ .



Hình 2.7: Đa giác, điểm mẫu và các bán kính

Thí dụ, đa giác trong hình 2.7 là hình vuông, chúng ta chọn 16 điểm mẫu trên biên,  $D$  là khoảng cách từ điểm trọng tâm  $C$  tới điểm mẫu  $s_n$ , với  $n \in (1, 2, 3, \dots, 16)$ . Chiều dài mỗi cạnh là như nhau và được chia làm 4 phần. Khi đó:

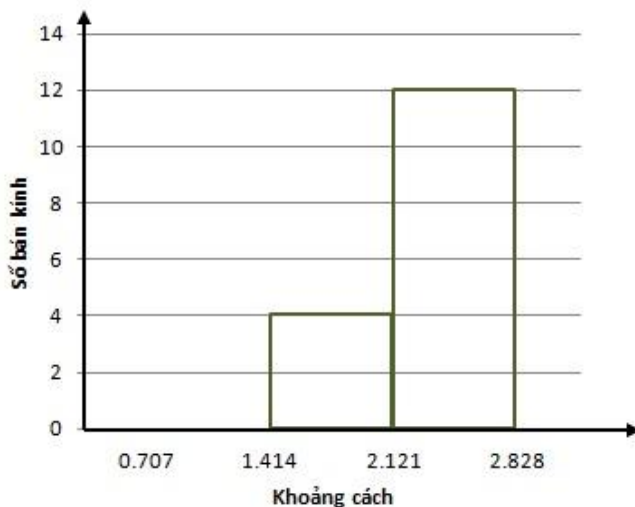
$$D_1 = D_5 = D_9 = D_{13} = 2.828$$

$$D_2 = D_4 = D_6 = D_8 = D_{10} = D_{12} = D_{14} = D_{16} = 2.236$$

$$D_3 = D_7 = D_{11} = D_{15} = 2.000$$

Giá trị lớn nhất là  $D_{\max} = 2.828$ . Chúng ta chia toàn bộ vùng  $[0, D_{\max}]$  thành 4 vùng, đó là  $[0, 0.707]$ ,  $[0.707, 1.414]$ ,  $[1.414, 2.121]$ ,  $[2.121, 2.828]$ . Chúng ta biết rằng  $D_3, D_7, D_{11}, D_{15}$  nằm trong khoảng  $[1.414, 2.121]$  và  $D_2, D_4, D_6, D_8, D_{10}, D_{12}, D_{14}, D_{16}$  nằm trong khoảng  $[2.121, 2.828]$ .

Chúng ta có lược đồ như sau:

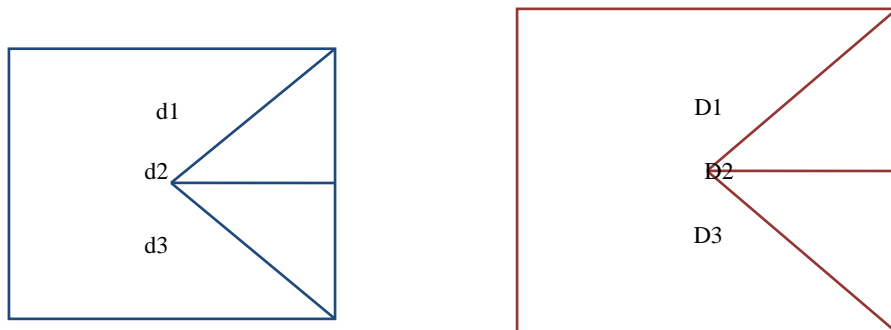


Hình 2.8: Biểu đồ khoảng cách của đa giác

Biểu đồ này có thể biểu diễn hình dạng của đa giác, việc lựa chọn điểm mẫu không ảnh hưởng tới quá trình biểu diễn hình dạng khi hình bị quay. Vấn đề nảy sinh ở đây là hai đa giác tương tự nhau nhưng kích thước khác nhau do đó lược đồ khoảng cách sẽ khác nhau. Điều này làm cho kết quả tra cứu không như mong muốn. Để giải quyết vấn đề này chúng ta thực hiện việc chuẩn hóa trước khi lập lược đồ khoảng cách.

## 2.7 Chuẩn hóa

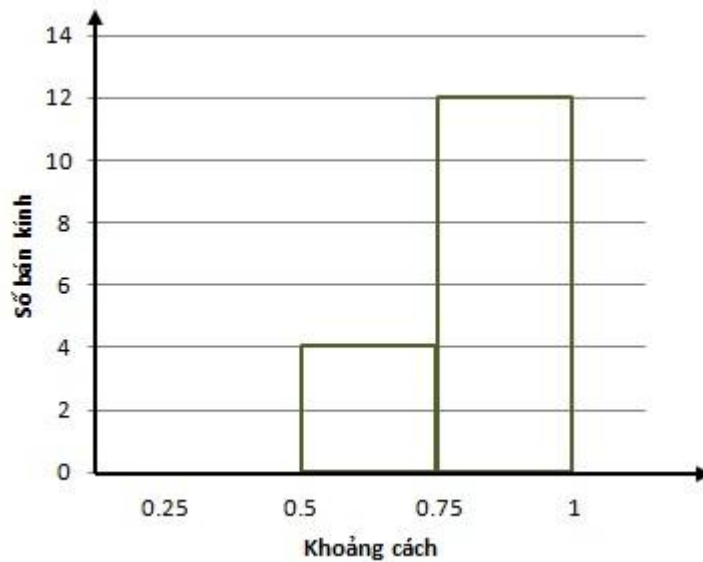
Cho hai hình đa giác với kích thước khác nhau, hình dạng giống nhau, trên cơ sở lý thuyết thì chúng là giống nhau. Tuy nhiên, nếu không qua quá trình chuẩn hóa mà để nguyên kích thước gốc để mô tả hình dạng và so sánh thì sẽ cho kết quả sai.



Hình 2.9: Hai hình dạng tương tự nhau nhưng kích thước khác nhau

Hai hình trên có hình dạng giống nhau nhưng kích thước bán kính từ trọng tâm đến biên khác nhau là  $D$  và  $d$ . Khi dùng  $(d_1, d_2, d_3)$  và  $(D_1, D_2, D_3)$  để biểu diễn hình dạng bằng lược đồ khoảng cách và thực hiện so sánh thì ta sẽ nhận được kết quả là không tương tự nhau. Do đó cần chuẩn hóa hình dạng trước khi biểu diễn và so sánh.

Việc chuẩn hóa được thực hiện như sau: Xác định bán kính lớn nhất trong các bán kính từ trọng tâm đến các điểm mẫu trên biên  $D_{max}$ . Sau đó lần lượt lấy các bán kính từ  $D_{min}$  đến  $D_{max}$  chia cho  $D_{max}$ . Kết quả mà ta thu được sau phép chia đã được chuẩn hóa và nằm trong khoảng  $[\frac{D_{min}}{D_{max}}, 1]$ . Do việc gán điểm mẫu phụ thuộc vào chiều dài biên và các điểm mẫu chia đều biên thành các phần bằng nhau nên giá trị sau khi chuẩn hóa sẽ là như nhau.



Hình 2.7: Lược đồ khoảng cách sau khi chuẩn hóa

Hình 2.10 là lược đồ tương ứng với lược đồ 2.9, sau khi đối tượng đã được chuẩn hóa về khoảng cách.

## 2.8 Đo độ tương tự

Sau khi chuẩn hóa chúng ta sẽ thu được lược đồ khoảng cách qua biểu diễn hình dạng của đa giác. Lược đồ khoảng cách được mô tả:

$D : (d_0, d_1, d_2, \dots, d_{n-1})$  với  $n$  là số lượng khoảng cách trong lược đồ;  $d_i$  với  $i \in [0, n-1]$  là số khoảng cách trong vùng khoảng cách này.

Ví dụ với hai lược đồ khoảng cách của hai đa giác:

$$D1 : (d1_0, d1_1, \dots, d1_{n-1})$$

$$D2 : (d2_0, d2_1, \dots, d2_{n-1})$$



Độ tương tự của hai lược đồ khoảng cách này tính theo công thức Euclid:

$$\text{Sim}(D1, D2) = \sqrt{\sum_{i=0}^{n-1} (d1_i - d2_i)^2} \quad (\text{eq. 6})$$

Với  $\text{Sim}(D1, D2)$  càng nhỏ thì hai lược đồ càng giống nhau và kết quả là hai đa giác sẽ càng giống nhau.

## 2.9 Kết luận chương

Đã có rất nhiều phương pháp được đề xuất với mục đích biểu diễn hình dạng của đối tượng trong ảnh: Phương pháp lưới (Grid Descriptors), phương pháp biểu diễn dựa trên mô tả Fourier (Fourier Descriptors), phương pháp biểu diễn quay góc (Turning Angle), .... Trong khía cạnh biểu diễn hình dạng thì phương pháp biểu diễn dựa trên mô tả Fourier gần với mục đích của phương pháp được đề xuất trong báo cáo này.

Hình dạng đối tượng sẽ được tính xấp xỉ về đa giác, sau đó tính trọng tâm, xác định các điểm mẫu, tính khoảng cách từ trọng tâm này tới các điểm mẫu trên các cạnh của đa giác, cuối cùng đưa ra lược đồ khoảng cách biểu diễn đối tượng. Phương pháp được đề xuất giải quyết được tính bất biến của đối tượng khi thay đổi vị trí, tỷ lệ kích thước của đối tượng và tính bất biến của đối tượng khi bị quay theo một góc bất kỳ.

## Chương 3: CÀI ĐẶT CHƯƠNG TRÌNH THỰC NGHIỆM

### 3.1 Cài đặt thuật toán

Các bước cài đặt thuật toán:

- Thu thập biên đa giác, xác định các đỉnh đa giác theo thứ tự.
- Tính diện tích và tính trọng tâm của đa giác.
- Lựa chọn điểm mẫu trên cạnh đa giác, tính khoảng cách giữa trọng tâm đa giác và điểm mẫu.
- Chuẩn hóa khoảng cách và xây dựng lược đồ khoảng cách.
- Sử dụng lược đồ để so sánh độ tương tự.

#### 3.1.1 Thu thập biên đa giác và lấy đỉnh đa giác

##### 3.1.1.1 Ý tưởng

Thuật toán này bắt nguồn từ nguyên tắc liền kề của các điểm ảnh trên tọa độ.

Phương thức sẽ có giá trị true nếu hai điểm ảnh thỏa mãn điều kiện các cặp giá trị tương ứng là:

$$g(x) = f(x_2, y_2) - f(x_1, y_1) = \begin{cases} \{x_2 - x_1 = 1 \\ y_2 - y_1 = 0 \} \\ \{x_2 - x_1 = 0 \\ y_2 - y_1 = 1 \} \\ \{x_2 - x_1 = 1 \\ y_2 - y_1 = 1 \} \\ \{x_2 - x_1 = 1 \\ y_2 - y_1 = -1 \} \\ \{x_2 - x_1 = -1 \\ y_2 - y_1 = 1 \} \\ \{x_2 - x_1 = -1 \\ y_2 - y_1 = -1 \} \\ \{x_2 - x_1 = 0 \\ y_2 - y_1 = -1 \} \\ \{x_2 - x_1 = -1 \\ y_2 - y_1 = 0 \} \end{cases}$$

Trường hợp hai điểm ảnh không thỏa mãn điều kiện trên thì phương thức kiểm tra sẽ trả về giá trị là false.

Để xác định được đỉnh của một đa giác chúng ta áp dụng một tính chất của đường thẳng  $y = mx + b$  trong hệ trục tọa độ 2 chiều chúng ta chia ra 3 trường hợp:

- Nếu  $m < 0$  thì sau mỗi vòng lặp  $x$  sẽ buộc phải tăng hoặc giảm một giá trị trên hệ trục tọa độ, đồng thời giá trị của  $y$  sẽ thay đổi tương ứng với giá trị của  $m$ .
- Nếu  $m > 0$  thì sau mỗi vòng lặp  $y$  luôn luôn tăng 1 giá trị trên hệ trục tọa độ đồng thời giá trị của  $x$  sẽ thay đổi tương ứng với giá trị của  $m$ .
- Nếu  $m = 0$  thì cả  $x$  và  $y$  sẽ cùng tăng hoặc là cùng giảm 1 giá trị hoặc  $x$  tăng  $y$  giảm,  $x$  giảm  $y$  tăng 1 giá trị sau mỗi vòng lặp.

Mảng sau khi đã được sắp xếp theo thứ tự nhất định chúng ta bắt đầu tiến hành xét trường hợp, nếu  $x$  luôn tăng một giá trị hoặc giảm 1 giá trị thì tiếp tục vòng lặp mới, ngược lại thì đặt điểm cuối cùng là đỉnh và lưu vào mảng dùng để chứa đỉnh đồng thời đảo sang xét  $y$ , nếu  $y$  luôn tăng hoặc giảm 1 giá trị thì tiếp tục vòng lặp với cặp giá trị thứ 2. Cứ lặp đi lặp lại việc xét như thế chúng ta sẽ tìm được tất cả các đỉnh của hình đa giác.

### 3.1.1.2 Thuật toán

**Đầu vào:** Danh sách chứa tọa độ đa giác chưa sắp xếp **UnsortedCoords** kiểu danh sách **List<Point>**

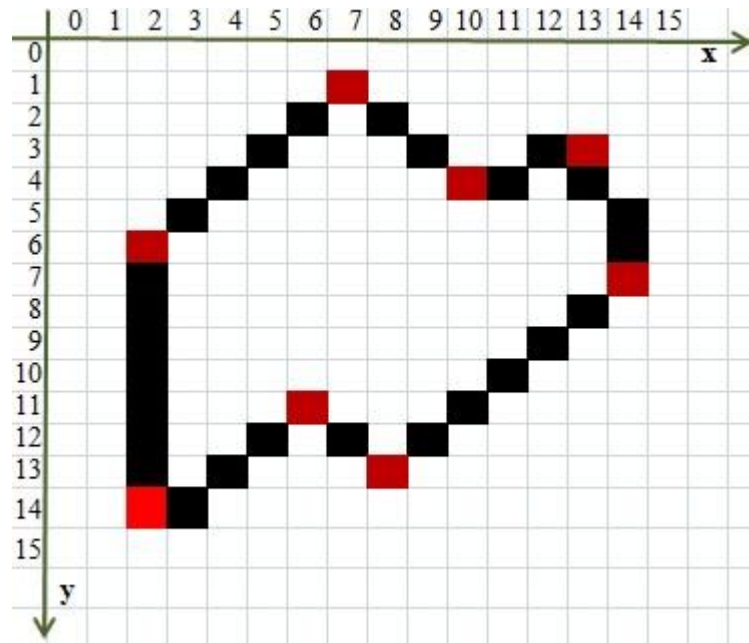
**Xử lý:**

```
List<point> SortedCoords
int Index;
Index = 1;
if(length(UnsortedCoords) > 0)
{
    SortedCoords[Index].X = UnsortedCoords[Index].X;
    SortedCoords[Index].Y = UnsortedCoords[Index].Y;
    UnsortedCoords.RemoveAt(Index);
    for(i = 0; i < length(UnsortedCoords); i++)
        if(Neighbour(SortedCoords[Index].X, SortedCoords[Index].Y,
UnsortedCoords[i+1].X, UnsortedCoords[i+1].Y) == true)
        {
            Index = Index + 1;
            SortedCoords[Index].X = UnsortedCoords[i].X;
            SortedCoords[Index].Y = UnsortedCoords[i].Y;
            UnsortedCoords.RemoveAt(i);
        }
}
```

**Đầu ra:**

Danh sách **SortedCoords** gồm các điểm ảnh được sắp xếp theo thứ tự.

### 3.1.1.3 Bài toán cụ thể



Hình 3.1: Một đa giác phóng to với mỗi ô tương ứng với một điểm ảnh

Các tọa độ thu thập được từ đa giác được quét theo thứ tự từ trên xuống và từ trái sang phải:

(7,1), (6,2), (8,2), (5,3), (9,3), (12,3), (13,3), (4,4), (10,4), (11,4), (13,4), (3,5), (14,5), (2,6), (14,6), (2,7), (4,7), (2,8), (13,8), (2,9), (12,9), (2,10), (11,10), (2,11), (6,11), (10,11), (2,12), (5,12), (7,12), (9,12), (2,13), (4,13), (8,13), (2,14), (3,14).

Chúng ta sử dụng thuật toán điểm ảnh liên kề để sắp xếp lại theo thứ tự nhất định.

+ Xét cặp điểm ảnh đầu tọa độ (7,1) và (6,2)

$$x_2 - x_1 = 6 - 7 = -1$$

$$y_2 - y_1 = 2 - 1 = 1$$

Thỏa mãn điều kiện nên tọa độ (7,1) được xếp vào danh sách mới và loại khỏi danh sách hiện tại.

Xét tiếp từ điểm ảnh (6,2).

+ Xét cặp điểm ảnh tọa độ (6,2) và (8,2):

$$x_2 - x_1 = 8 - 6 = 2$$

$$y_2 - y_1 = 2 - 2 = 0$$

Không thỏa mãn điều kiện liên kề. Bỏ qua tọa độ (8,2) và xét tiếp điểm ảnh tiếp theo trong danh sách hiện tại.

+ Xét cặp điểm ảnh tọa độ (6,2) và (5,3):

$$x_2 - x_1 = 5 - 6 = -1$$

$$y_2 - y_1 = 3 - 2 = 1$$

Thỏa mãn điều kiện nên tọa độ (6,2) được xếp vào danh sách mới và loại khỏi danh sách hiện tại. Xét tiếp từ điểm ảnh (5,3).

Làm tương tự với các tọa độ còn lại ta sẽ được danh sách mới chứa tọa độ các điểm ảnh đã được xếp theo thứ tự ngược chiều kim đồng hồ.

(7,1), (6,2), (5,3), (4,4), (3,5), (2,6), (2,7), (2,8), (2,9), (2,10), (2,11), (2,12), (2,13), (2,14), (3,14), (4,13), (5,12), (6,11), (7,12), (8,13), (9,12), (10,11), (11,10), (12,9), (13,8), (14,7), (14,6), (14,5), (13,4), (13,3), (12,3), (11,4), (10,4), (9,3), (8,2).

Sau khi có danh sách tọa độ đã sắp xếp, chúng ta xét từ điểm đầu của danh sách:

+ Điểm ảnh tọa độ (7, 1), tiếp theo là (6, 2): x giảm, y tăng,  $m = 0$ , hai điểm ảnh tạo với gốc tọa độ một góc  $45^\circ$ . Lưu tọa độ (7, 1) vào danh sách đỉnh. Tiếp tục vòng lặp với (5, 3): x giảm, y tăng,  $m = 0$ . Tương tự cho tới tọa độ (2, 7).

+ Xét điểm ảnh có tọa độ (2, 6) và (2, 7):  $x_2 - x_1 = 0$ , lưu (2, 6) vào danh sách đỉnh, chuyển sang xét y.

Làm tiếp tục chúng ta sẽ được đỉnh tiếp theo là (2, 14). Cuối cùng, khi kết thúc việc duyệt danh sách trên chúng ta sẽ thu được các đỉnh của đa giác với tọa độ lần lượt là:

(7, 1), (2, 6), (2, 14), (6, 11), (8, 13), (14, 7), (13, 3), (10, 4).

### 3.1.2 Tính diện tích và xác định trọng tâm của đa giác

#### 3.1.2.1 Ý tưởng

Để tìm được trọng tâm của đa giác, đầu tiên chúng ta phải tính được diện tích đa giác, áp dụng Công thức Gauss để tính diện tích hình đa giác trên mặt phẳng khi đã biết được tọa độ các đỉnh của đa giác, từ đó mới tìm được trọng tâm đa giác phục vụ mục đích xây dựng lược đồ khoảng cách.

#### 3.1.2.2 Thuật toán

\* Thuật toán tính diện tích đa giác

**Đầu vào:** Danh sách chứa các tọa độ đỉnh đa giác **Pvertex** kiểu **List<Point>**

**Xử lý:**

```
Var  
int i,j;  
double Area;  
area = 0;  
for(i = 1; i < length(Pvertex); i++)
```

```

{
    j = (i + 1) / length(Pvertex);
    Area = Area + Pvertex[i].X * Pvertex[j].Y;
    Area = Area - Pvertex[j].X * Pvertex[i].Y;
}
Area = (abs)Area/2;

```

**Đầu ra:** Giá trị diện tích của đa giác **Area**

**\*Thuật toán tính tọa độ trọng tâm của đa giác**

**Đầu vào:** **Pvertex** kiểu **List<Point>** gồm danh sách các Point lưu trữ giá trị các đỉnh đa giác.

**Xử lý:**

```

Var
int i,j;
float alpha;
int C[][];
for(i = 0; i < length(Pvertex); i++)
{
    j = (i + 1) / length(Pvertex);
    alpha = (Pvertex [i].x*Pvertex [j].y-Pvertex [j].x*Pvertex [i].y);
    c.x = c.x + +(Pvertex [i].x+Pvertex [j].x)*alpha;
    c.y:= c.y +(Pvertex [i].y+Pvertex [j].y)*alpha;
}
c.x = c.x / (6 * Area);
c.y = c.y / (6 * Area);

```

**Đầu ra:** Trả về tọa độ trọng tâm đa giác **C[x,y]**

### 3.1.2.3 Bài toán cụ thể

Từ đa giác ở hình 3.1, chúng ta đã có một danh sách chứa tọa độ các đỉnh của đa giác.

(7,1), (2,6), (2,14), (6,11), (8,13), (14,7), (13,3), (10,4).

Diện tích đa giác:

$$A = \frac{1}{2} \left| \sum_{i=1}^{n-1} x_i y_{i+1} + x_n y_1 - \sum_{i=1}^{n-1} x_{i+1} y_i - x_1 y_n \right| \quad (\text{eq. 1})$$

$A = 93$  Vậy diện tích của đa giác là 93.

Xác định trọng tâm của đa giác:  $C(\bar{x}, \bar{y})$

$$\bar{x} = \frac{1}{6A} \sum_{i=0}^{n-1} (x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \quad (\text{eq. 2})$$

$$\bar{y} = \frac{1}{6A} \sum_{i=0}^{n-1} (y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \quad (\text{eq. 2})$$

Ta có  $\bar{x} = 7$ ;  $\bar{y} = 7$

Vậy trọng tâm của đa giác hình 3.1 là: C(7,7).

### 3.1.3 Lựa chọn điểm mẫu và tính khoảng cách giữa trọng tâm đa giác và điểm mẫu

#### 3.1.3.1 Ý tưởng

Trong mục này chúng ta cần giải quyết được hai vấn đề, thứ nhất là lựa chọn điểm mẫu trên các cạnh biên sao cho hợp lý, thứ 2 là tính khoảng cách giữa các điểm mẫu đó tới trọng tâm đa giác.

#### \* Xác định vị trí và số lượng các điểm mẫu:

Đầu vào xử lý bao gồm :

- Tổng số lượng điểm ảnh của các cạnh biên của đối tượng, giá trị này được tính trong quá trình lưu các tọa độ của biên.
- Số lượng điểm mẫu ước lượng sử dụng trên biên đó, do người dùng nhập vào hoặc được gán trong quá trình phát triển chương trình.
- Số lượng điểm ảnh (chiều dài) của cạnh biên đang xét, có thể xác định được số lượng điểm ảnh trong quá trình xác định đỉnh đa giác.

Về mặt bản chất của đồ họa máy tính, các điểm ảnh được bố trí theo dạng ma trận và tọa độ là kiểu số nguyên. Cho nên việc xác định điểm ảnh tương ứng với vị trí của tọa độ thực tế phải dựa trên nguyên tắc làm tròn và chuyển đổi về dạng số nguyên.

Sau khi đã xác định được số điểm mẫu trên cạnh biên chúng ta duyệt từng điểm ảnh trên biên đó theo thứ tự và đánh dấu điểm ảnh đó lại sao cho các điểm mẫu này được cách đều trên biên, và chia biên đó thành các đoạn bằng nhau.

#### \* Tính khoảng cách giữa Các điểm mẫu và trọng tâm của đa giác

Đầu vào là tọa độ của điểm mẫu và điểm trọng tâm đa giác, theo công thức (eq. 5) chúng ta xây dựng phương thức tính khoảng cách giữa hai điểm trên ma trận tọa độ.

### 3.1.3.2 Thuật toán

#### \* Xác định vị trí và số lượng các điểm mẫu

**Đầu vào:** Danh sách **p** kiểu **List<Point>** chứa tập tọa độ của đa giác được xếp theo thứ tự ngược chiều kim đồng hồ.

**Xử lý:**

```
List<int>; // Số điểm ảnh trên một cạnh
int count, tempcount, SNum,a,SDistance, Index;
List<Point> ls_Point; // Danh sách đỉnh
List<Point> SCoordinates; // Danh sách tọa độ điểm mẫu
Index = 0;
for(int i = 0; i < length(p); i++)
{
    count = count + 1;
    Side[Index].X = p[i].X;
    Side[Index].Y = p[i].Y;
    p.RemoveAt(i);
    if(FindVertex(p[i].X,p[i].Y) == true)
    {
        SNum = (count / length(p)) * a;
        SDistance = count / (SNum + 1);
        for(i = SDistance; i < Side; i++)
        {
            SCoordinates[j].X = p[i].X;
            SCoordinates[j].Y = p[i].Y;
            SNum = 0;
            SDistance = 0;
        }
    }
}
```

**Đầu ra:** Trả về danh sách tọa độ các điểm mẫu **SCoordinates**

#### \* Tính khoảng cách giữa Các điểm mẫu và trọng tâm của đa giác

**Đầu vào:**

- Danh sách chứa tọa độ các điểm mẫu **SCoordinates** kiểu **List<Point>**
- Tọa độ trọng tâm của đa giác kiểu **Point**



### Xử lý:

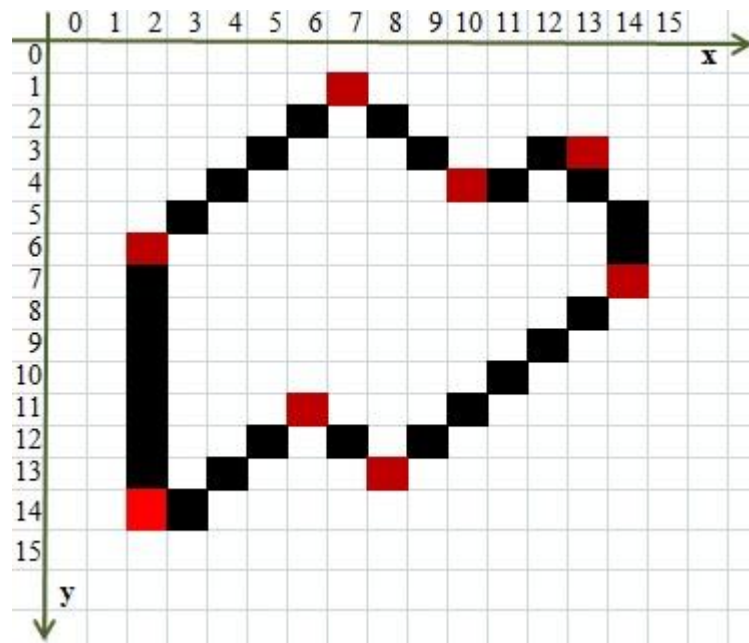
```
List<float> Distances;  
for(int i = 0; i < length(SamplePointCoordinates); i++)  
{  
    Distance[i] = Distance(Point.X, SCoordinates[i].X, Point.Y, SCoordinates[i].Y);  
}
```

**Đầu ra:** Trả về danh sách **Distances** chứa các khoảng cách từ trọng tâm tới các điểm mẫu theo thứ tự.

### 3.1.3.2 Bài toán cụ thể

Ví dụ hình 3.1: Tổng số điểm ảnh là 35 pixel, có các đỉnh lần lượt là:

(7,1), (2,6), (2,14), (6,11), (8,13), (14,7), (13,3), (10,4).

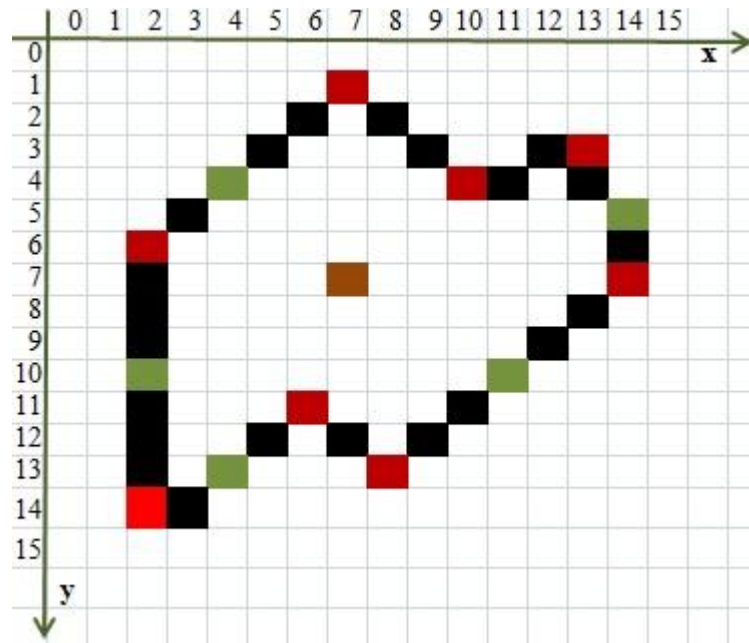


Xét từ (7,1) đến (2,6): Có 6 điểm ảnh cạnh nhau nên chiều dài cạnh này là 6. Số lượng điểm mẫu khoảng 8 điểm.

$$\text{Từ công thức (eq. 4) } N_i = \frac{L_i}{L_{\text{sum}}} N = \frac{6}{35} 8 = 1$$

Vậy có 1 điểm căng đều 2 đỉnh trên đoạn từ (7,1) đến (2,6) là (4,4).

Tiếp tục thực hiện với các đỉnh còn lại chúng ta sẽ thu được kết quả như dưới đây:



(4,4), (2,10), (4,13), (11,10), (14,5)

Trọng tâm đa giác là  $C(7,7)$

Tính lần lượt khoảng cách từ trọng tâm đến các điểm mẫu:

$D^* = \{5; 5.385; 5.83; 4.472; 7.615\} \{0.656; 0.707; 0.765; 0.587; 1\}$

### 3.1.4 Chuẩn hóa khoảng cách

#### 3.1.4.1 Ý tưởng

Nhằm đảm bảo khi thay đổi kích thước đối tượng theo tỉ lệ bất kì mà không làm ảnh hưởng đến việc so sánh, trước khi xây dựng lược đồ khoảng cách chúng ta cần chuẩn hóa các khoảng cách. Chúng ta xác định khoảng cách lớn nhất trong những khoảng cách đã thu được ở trên. Sau đó lấy lần lượt các khoảng cách chia cho khoảng cách lớn nhất đấy. Chúng ta sẽ thu được một dãy khoảng cách thuộc khoảng  $[0,1]$ . Khi đó sự thay đổi về kích thước bất kì của đối tượng sẽ không làm ảnh hưởng tới kết quả so sánh hình dạng đối tượng.

#### 3.1.4.2 Thuật toán

**Đầu vào:** Danh sách chứa các giá trị khoảng cách của đa giác đang xét **D** kiểu **List<float>**

**Xử lý:**

```
List<float> DNormal;
float Max;
Max = D[0];
for(int i = 0; i < length(D); i++)
```

```

{
    if(D[i] > Max)
        Max = D[i];
}
for(int i = 0; i < length(D); i++)
{
    DNormal [i] = D[i] / Max;
}

```

**Đầu ra:** Danh sách chứa các khoảng cách đã chuẩn hóa **DNormal**

### 3.1.4.3 Bài toán cụ thể

Áp dụng vào ví dụ một đa giác trong hình 3.1, với  $D^*$  thu được, chúng ta chuẩn hóa về khoảng  $[0,1]$ .

$D^* = \{5; 5.385; 5.83; 4.472; 7.615\}$   $\{0.656; 0.707; 0.765; 0.587; 1\}$

Giá trị cực đại trong các khoảng cách này là  $D_{\max} = 7.615$

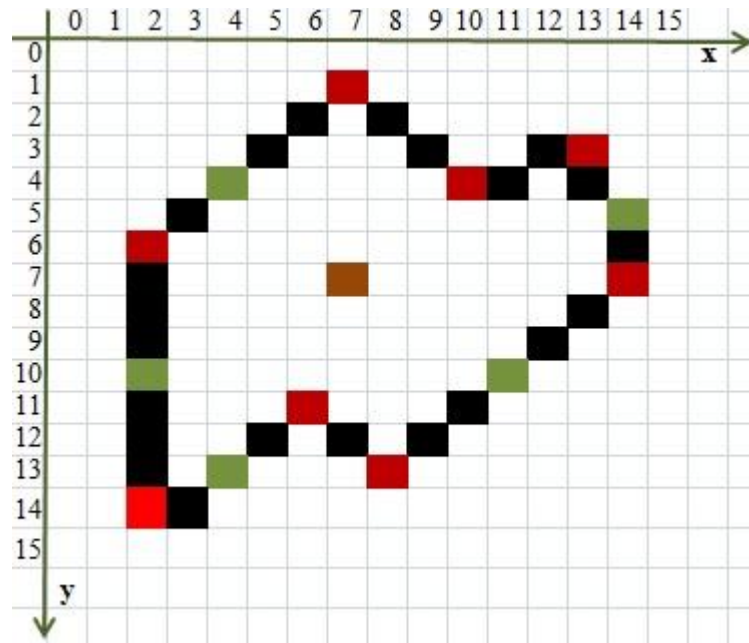
Chúng ta sẽ có tập giá trị sau chuẩn hóa:

$$d[] = \{0.656; 0.707; 0.765; 0.587; 1\}$$

## 3.1.5 Xây dựng lược đồ khoảng cách

### 3.1.5.1 Ý tưởng

Ý tưởng xây dựng lược đồ khoảng cách là phân chia các giá trị khoảng cách, hay nói cách khác kích thước tập các bán kính đa giác vào các nhóm dựa trên các ngưỡng xác định, ở đây đề xuất chia thành 5 khoảng.



Ta sẽ có các khoảng ngưỡng sau:

[0, 0.2], [0.2, 0.4], [0.4, 0.6], [0.6, 0.8], [0.8, 1]

Cơ sở dữ liệu chúng ta tạo ra một bảng lưu giá trị lược đồ khoảng cách Distance\_Histogram liên kết với ảnh bao gồm các trường như sau: (idhistogram, d<sub>0</sub>,d<sub>1</sub>,d<sub>2</sub>,d<sub>3</sub>,d<sub>4</sub>) trong đó idhistogram là khóa chính liên kết với khóa ngoại của bảng chứa ảnh và thông tin ảnh, d<sub>0</sub> là vùng 1, d<sub>1</sub> là vùng 2... tương tự chúng ta sẽ có 5 vùng tương ứng với 5 trường d<sub>0</sub> tới d<sub>4</sub> trong bảng Distance\_Histogram.

### 3.1.6.2 Thuật toán

#### Đầu vào:

- Bao gồm một danh sách chứa tập các giá trị khoảng cách **d** kiểu List<float>.

#### Xử lý:

```
int c1, c2, c3, c4, c5;
c1 = c2 = c3 = c4 = c5 = 0;
for(int i = 0; i < length(d); i++)
{
    if(d[i] >= 0 and d[i] <= 0.2)
        c1 = c1 + 1;
    Insert(c1,InsertTo_d1);
    else if(d[i] >= 0.2 and d[i] <= 0.4)
        c2 = c2 + 1;
    Insert(c2,InsertTo_d2);
    else if(d[i] >= 0.4 and d[i] <= 0.6)
```

```

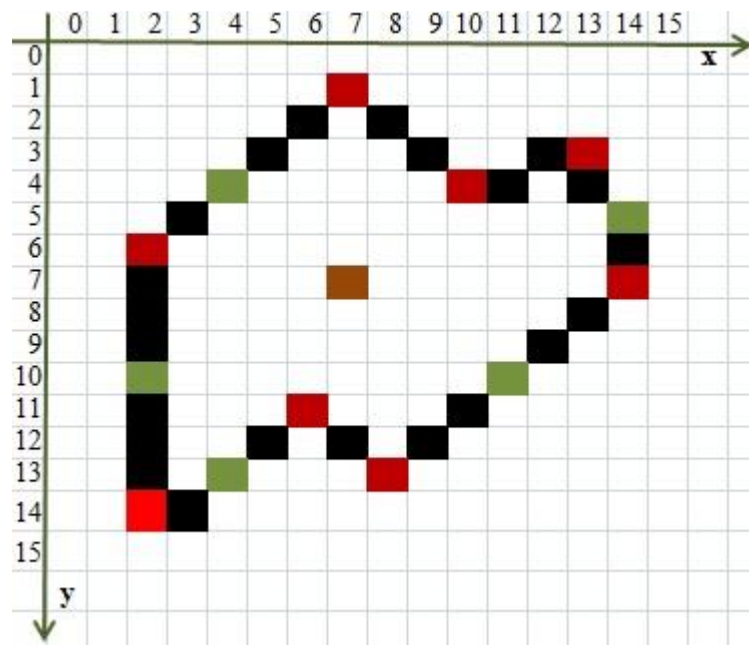
c3 = c3 + 1;
Insert(c3,InsertTo_d3);
else if(d[i] >= 0.6 and d[i] <= 0.8)
c4 = c4 + 1;
Insert(c4,InsertTo_d4);
else if(d[i] >= 0.8 and d[i] <= 1)
c5 = c5 + 1;
Insert(c5,InsertTo_d5);
}

```

**Đầu ra:** Đây là phương thức không có giá trị trả về.

### 3.1.6.3 Bài toán cụ thể

Trở lại thí dụ hình 3.1:



$d[] = \{0.656; 0.707; 0.765; 0.587; 1\}$

$i \in [0, n-1]$  là tập chứa khoảng cách; Cho  $R = 5$ ;  $d[i]_{\max} = 1$ .

Ta sẽ thu được các khoảng:

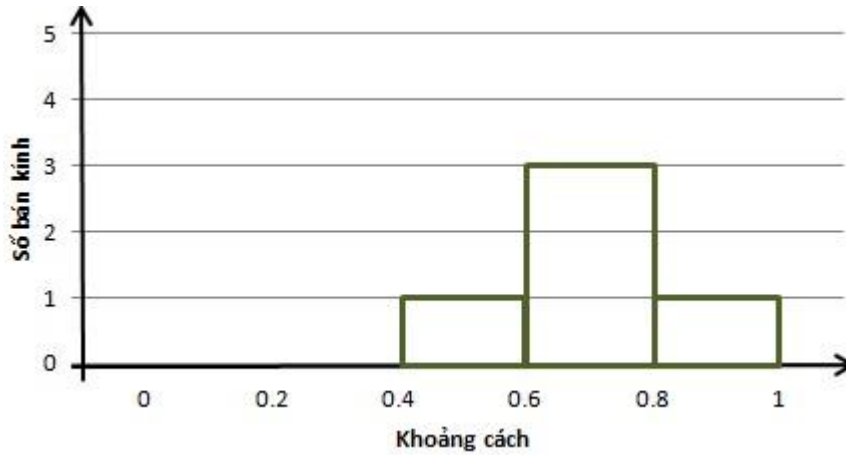
$[0, 0.2]$ ,  $[0.2, 0.4]$ ,  $[0.4, 0.6]$ ,  $[0.6, 0.8]$ ,  $[0.8, 1]$

Xếp  $d[i]$  vào các khoảng:

$d[4] \in [0.8, 1]$

$d[0], d[1], d[2] \in [0.6, 0.8]$

$d[3] \in [0.4, 0.6]$



Hình 3.2: Lược đồ khoảng cách đa giác hình 3.1

### 3.1.6 Độ đo tương tự

#### 3.1.6.1 Ý tưởng

Phương pháp đo độ tương tự được đề xuất trong báo cáo này là dựa trên biểu diễn hình dạng của lược đồ khoảng cách, trục khoảng cách của lược đồ được chia thành nhiều phần ký hiệu mỗi vùng là  $d[i]$ , với  $i \in [0, n - 1]$ , Lược đồ khoảng cách biểu diễn bởi  $D$ :

$D : (d_0, d_1, d_2, \dots, d_{n-1})$

Giả sử có hai lược đồ:

$D_1 : (d1_0, d1_1, d1_2, \dots, d1_{(n-1)})$ .

$D_2 : (d2_0, d2_1, d2_2, \dots, d2_{(n-1)})$ .

Áp dụng công thức eq. 6 chúng ta sẽ thu được khoảng cách giữa hai lược đồ

$$\text{Sim}(D1, D2) = \sqrt{\sum_{i=0}^{n-1} (d1_i - d2_i)^2} \quad (\text{eq. 6})$$

#### 3.1.6.2 Thuật toán

**Đầu vào:** Danh sách  $D_1$  và Danh sách  $D_2$  kiểu **int**

**Xử lý:**

```
float Sum;
double Sim;
for(int i = 0; i <= 7; i++)
{
    Sum = (D1[i] - D2[i]) * (D1[i] - D2[i]);
}

```

Sim = sqrt((double)Sum);

**Đầu ra:** Trả về giá trị độ tương tự **Sim**

### 3.1.6.3 Áp dụng thực tế

Hình 3.1 cho chúng ta lược đồ khoảng cách  $D1 = \{0, 0, 1, 3, 1\}$

Giả sử một đa giác sau khi xử lí ta được lược đồ khoảng cách  $D2 = \{0, 2, 2, 3, 0\}$

Độ tương đồng giữa  $D1$  và  $D2$  là:  $Sim = \sqrt{6} = 2.4495$

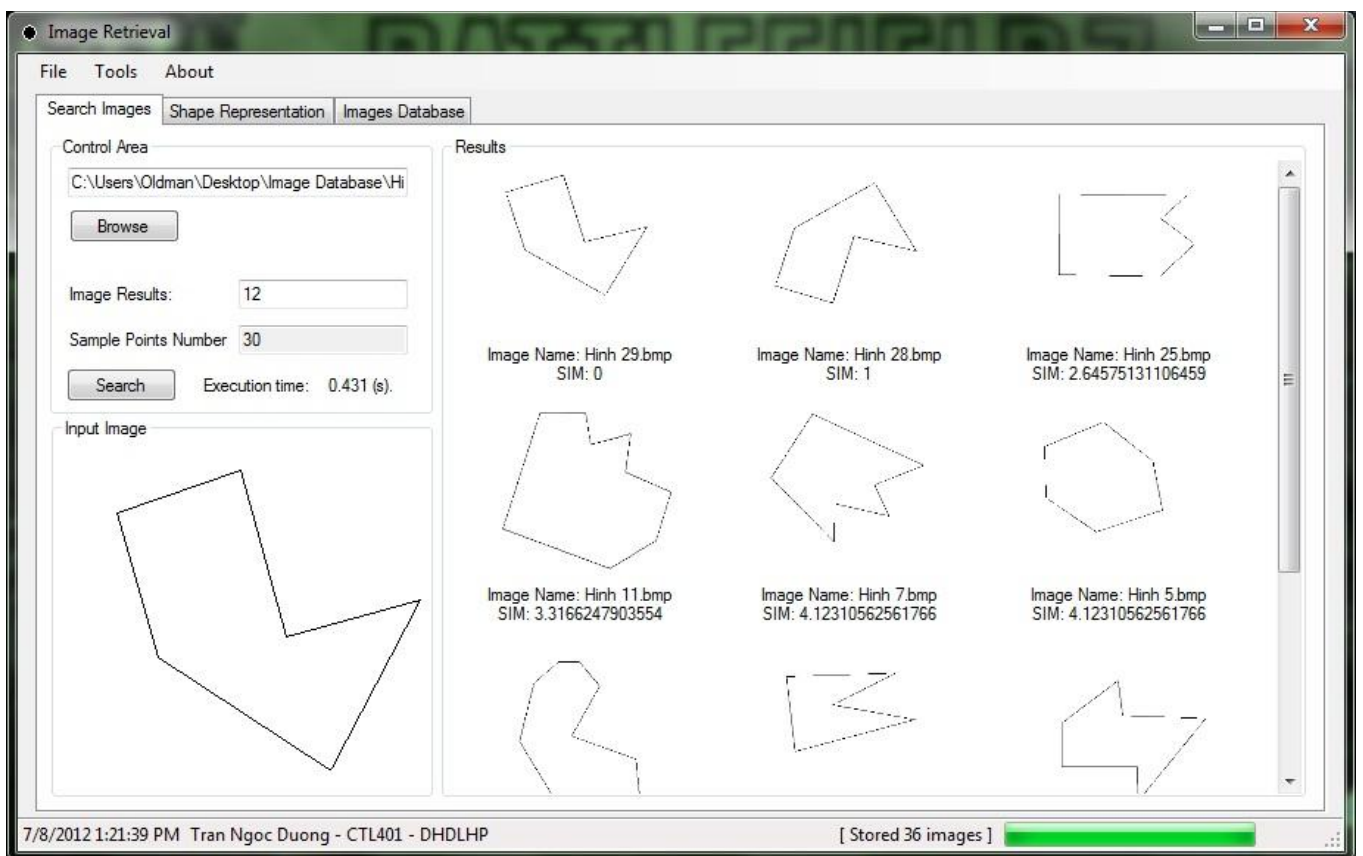
Có một đa giác khác với lược đồ khoảng cách  $D3 = \{2, 1, 0, 2, 3\}$

Độ tương đồng giữa  $D1$  và  $D3$  là:  $Sim = \sqrt{11} = 3.3166$

Vậy đa giác có lược đồ khoảng cách  $D2$  và  $D1$  là tương tự nhau.

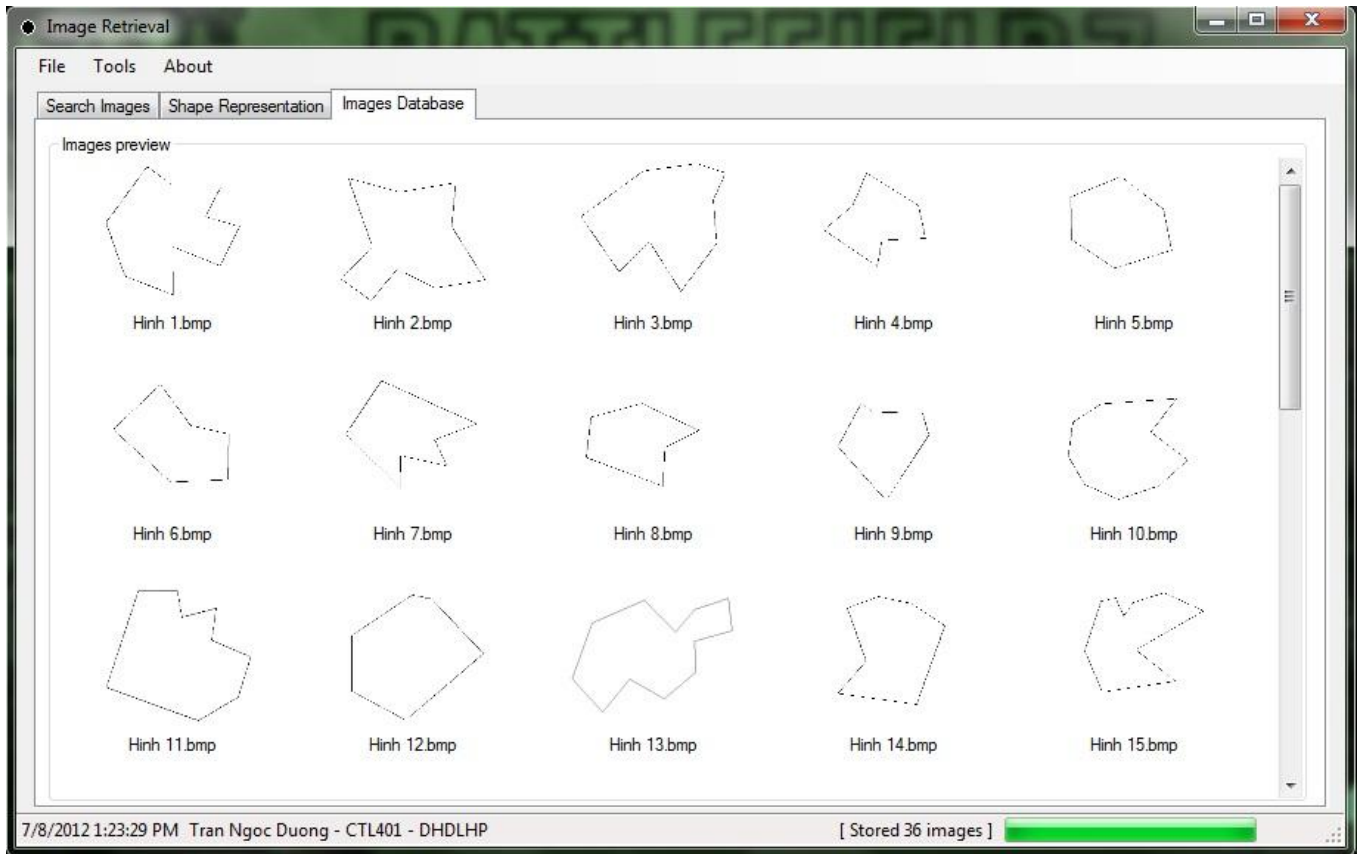
## 3.2 Giao diện chương trình

### 3.2.1 Giao diện tìm kiếm và kết quả



Hình 3.3: Giao diện tìm kiếm và kết quả

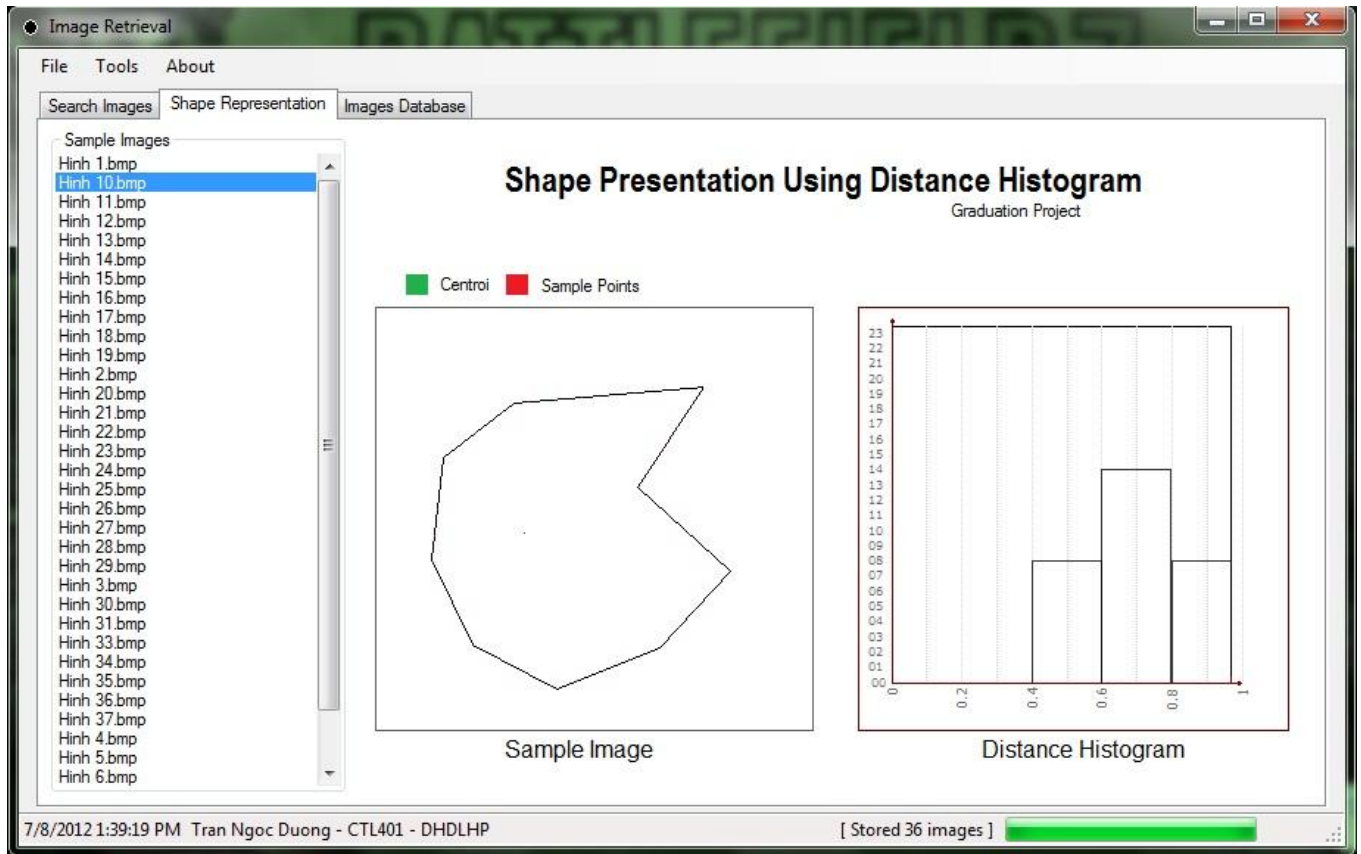
### 3.2.2 Cơ sở dữ liệu ảnh



Hình 3.4: Cơ sở dữ liệu ảnh



### 1.2.3 Lược đồ khoảng cách



Hình 3.5: Lược đồ khoảng cách

### 3.3 Kết luận chương

Qua quá trình phát biểu ý tưởng, trình bày thuật toán và lấy ví dụ thực tế chúng ta đã từng bước đưa ra được lược đồ khoảng cách biểu diễn hình dạng cho đa giác và tính toán, đo độ tương tự của đa giác khác so với đa giác ban đầu.

## KẾT LUẬN

Đồ án đã có các kết quả sau:

- Đồ án đã trình bày phương pháp tra cứu hình dạng tương tự dựa trên lược đồ khoảng cách và biểu diễn hình dạng, qua đó giúp tra cứu hình ảnh.
- Đồ án đã trình bày được cấu trúc một hệ thống tra cứu ảnh, các đặc trưng thường được sử dụng trong hệ thống này. Ngoài ra còn các ứng dụng của tra cứu ảnh trong đời sống cũng như trong các ngành khoa học, an ninh, bảo mật, ...
- Thông qua ý tưởng về lược đồ khoảng cách và biểu diễn hình dạng, đồ án đã nêu lên các bước thực hiện, cơ sở lý thuyết để xây dựng một hệ thống tra cứu ảnh dựa trên nội dung.
- Xây dựng được chương trình mô phỏng theo phương pháp đã trình bày.

Hướng phát triển: Trong thời gian tới nếu có điều kiện phát triển tiếp, em sẽ phát triển hệ thống từ lấy xấp xỉ hình dạng của đối tượng, có thể tối ưu hơn nữa dựa vào màu sắc và hình dạng của đối tượng.

## DANH MỤC TÀI LIỆU THAM KHẢO

### Tiếng Việt

[1] Lương Mạnh Bá, Nguyễn Thanh Thủy, Nhập môn xử lý ảnh số, NXBKHK, năm 1999, Trang 7-8.

### Tiếng Anh

[2] Xiaojun Q, Content-based Image Retrieval (CBIR) – Page. 14.

[3] Pradya Rane, Pallavi Kulkarni, Suchita Patil and B.B. Mesram, IJCEM International Journal of Computational Engineering & Management, Vol. 14, October 2011, ISSN (Online): 2230-7893 - Feature based image retrieval of images for CBIR.

[4] Jie-xian Zeng, Yong-gang Zhao, Xiang Fu: A Novel Shape Representation and Retrieval Algorithm: Distance Autocorrelogram. JSW 5(9): 1022-1029 (2010)

### Website

[5] [http://en.wikipedia.org/wiki/Content-based\\_image\\_retrieval](http://en.wikipedia.org/wiki/Content-based_image_retrieval)

[6] <http://mathworld.wolfram.com/PolygonArea.html>

[7] <http://code.activestate.com/recipes/578047-area-of-polygon-using-shoelace-formula/>