TẠP CHÍ KHOA HỌC TÍNH TOÁN VÀ ĐIỀU KHIỂN TẬP II (1986), Số 3 (14–18; 27)

MỘT CÁCH TIẾP CẬN VIỆC BIỂU DIỄN ẢNH CỦA ĐỐI TƯỢNG QUA BÓNG CỦA NÓ

PHẠM NGỌC KHÔI

TÓM TẮT NỘI DUNG

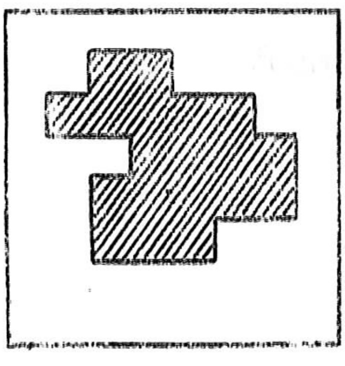
Bài báo đề cập đến một kiểu dữ liệu đặc biệt dùng để biểu diễn đối tượng trên ảnh, đó là cây tứ phân ảo. Các toán tử và hàm thao tác trên cây tứ phân ảo được đưa ra, cùng với các tính chất của chúng. Một vài áp dụng trực tiếp của cây tứ phân ảo được trình bày. Phần cuối bài báo có đưa ra ước lượng hiệu quả của biểu diễn cây tứ phân so với mã độ dài loạt và mã dây chuyền đối với một ảnh cụ thể cho trước.

I – MỞ ĐẦU

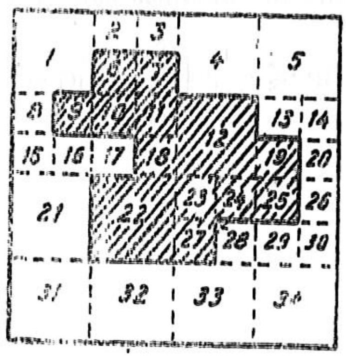
Bài toán xử lý ảnh dựa trên bóng các đối tượng được đề cập đến trong nhiều lĩnh vực áp dụng khác nhau: điều tra địa hình, phân tích sinh–y học và đặc biệt là trong việc kiểm định chất lượng các sản phẩm công nghiệp [1, 2]. Đối với các ảnh nhị phân, để biểu diễn các đối tượng trên ảnh, có 3 cách mã hóa sau được dùng một cách phổ biến nhất: mã độ dài loạt (run length code), mã dây chuyền (chain code), và cây tứ phân (quadtree) [7]. Bài báo này là mở rộng một vài kết quả nghiên cứu về cây tứ phân đã được công bố trong một bài báo trước [9]. Để cho tiện, ta giữ nguyên các ký pháp đã dùng.

Ở đây, một ảnh nhị phân được hiểu là một ma trận có cỡ N×N, ($N = 2^n$), mỗi phần tử của nó nhận giá trị 0 hoặc 1 tùy thuộc giá trị của điểm ảnh tương ứng là trắng hoặc đen. Tập hợp các phần tử 1 liên thông với nhau thể hiện một đối tượng trên ảnh.

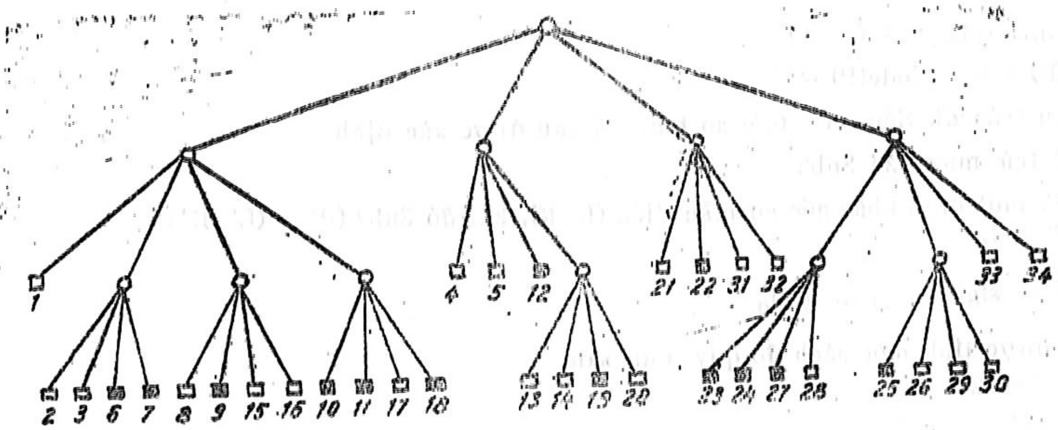
Cây tứ phân là một cấu trúc dữ liệu phân cấp dùng để biểu diễn cô đọng ảnh nhị phân 2 chiều. Cấu trúc này dựa trên việc chia một cách liên tiếp ma trận thành các khối phần tư cho đến khi ta thu được các khối đồng màu (tối thiểu là một điểm ảnh), sao cho trong một khối chỉ bao gồm toàn giá trị 1 hoặc 0. Để ý rằng nếu ảnh có cỡ $2^n \times 2^n$ thì sau lần chia thứ k, mỗi khối phần tư có cỡ $2^{n-k} \times 2^{n-k}$. Ví dụ như ảnh trong hình 1a tương ứng với quá trình làm mịn vẽ ở hình 1b. Quá trình này có thể biểu diễn bởi một cây tứ phân như sau: gốc của cây tương ứng với toàn bộ ảnh, 4 con của một đỉnh tương ứng với 4 khối phần tư, và các lá tương ứng với các khối phần tư đồng màu (không cần chia nữa) và nhận giá trị BLACK hoặc WHITE tùy theo các phần tử ảnh trong đó là 1 hay 0. Các đỉnh không phải là lá ứng với các khối phần tư có chứa cả hai loại điểm ảnh 1 và 0, mang giá trị thuộc tính GRAY. Biểu diễn cây cấp 4 của hình 1a được vẽ ở hình 1c.



Hình 1a
Đối tượng trên ảnh



Hình 1b
Việc chia mịn ảnh thành các khối phần tư



Hình 1c :

Biểu diễn cây tứ phân của ảnh trên hình 1a

Mặc dù ở đây một đối tượng được biểu diễn bởi hợp của một số khối vuông, nhưng các khối ở đây có độ dài chuẩn (lũy thừa của 2) và các vị trí chuẩn. Vì ma trận ban đầu có kích thước $2^n \times 2^n$, chiều cao của cây nhiều nhất là n.

Một trong những nghiên cứu đầu tiên về cây tứ phân và áp dụng của nó trong xử lý ảnh là bài báo của Sidhu và Boute [3]. Gần đây, Samet và Rosenfeld đã có những bài tổng quan về lịch sử của cây tứ phân và việc biểu diễn ảnh bởi bóng của đối tượng [4].

II – CẤU TRÚC DỮ LIỆU CHO CÂY TỨ PHÂN

2.1. Cây tứ phân ảo

Liên quan đến biểu diễn cây tứ phân, một cấu trúc dữ liệu truyền thống được sử dụng để tính toán các đặc trưng ảnh và phân tích ảnh [5]. Mỗi đỉnh của cây tứ phân có thể được lưu trữ dưới dạng một record chứa 6 trường, 5 trường đầu chứa tên các con trỏ trỏ tới đỉnh cha và 4 đỉnh con với nhãn : NW, NE, SW, SE, tương ứng với 4 khối con của một khối: Tây–Bắc, Đông–Bắc, Tây–Nam, Đông–Nam. Trường thứ 6 có tên là NODETYPE mô tả nội dung của khối: WHITE nếu khối chỉ chứa các điểm ảnh có giá trị 0, BLACK nếu khối chỉ chứa các điểm ảnh có giá trị 1, và GRAY nếu khối chứa cả 2 loại điểm ảnh.

Mới đây có một số tác giả đề xướng những cấu trúc dữ liệu cho riêng biểu diễn cây tứ phân, không sử dụng cơ chế con trỏ. Sau đây ta sẽ dùng một cấu trúc dữ liệu, gọi là cây tứ phân ảo, do John và Iyengar đưa ra [6].

Giả sử rằng một khối vuông có 4 khối phần tư được đánh chỉ số như sau

0	1
2	3

tương ứng với các khối phần tư NW, NE, SW, SE. Mỗi khối phần tư trong ảnh được ứng với một đỉnh trong cây tứ phân, đỉnh này có 4 con được đánh số từ trái sang phải theo thứ tự 0, 1, 2, 3. Ta sẽ biểu diễn mỗi đỉnh P của cây tứ phân bởi một cặp số nguyên (L, K) trong đó :

– L là mức của đỉnh (tức là khoảng cách từ gốc đến đỉnh), $0 \leqslant L \leqslant n$,

– K được xác định bởi $K = \sum_{i=0}^{L-1} a_i\, 4^i \quad 0 \leqslant a_i \leqslant 3$

ở đây, $(a_{L-1}, \ldots, a_1, a_0)$ là đường đi từ gốc tới đỉnh P.
Ký hiệu

$K = a_{L-1} \dots a_1 a_0$, $P' = (L, K)$

Level (P) = L , Code(P) = K.

Trên cấu trúc dữ liệu này, một số toán tử sau được xác định

1) Phép trừ mượn 2; Sub2

Đặt P là một đỉnh khác gốc có biểu diễn (L, K), khi đó Sub2 (P) = (L', K')
trong đó $L' = L$

$$K' = \text{sign}\ a'_{L-1} \dots a'_1 a'_0$$

với các a_i được tính một cách đệ quy như sau

- $b_0 := 2$
- $a'_i := ((a_i + 4) - b_i) \bmod 4$

$$b_{i+1} := \begin{cases} 0 & \text{nếu } a_i \geqslant b_i; \quad i = 0,1,\dots L-1 \\ 2 & \text{ngược lại} \end{cases}$$

$$\text{sign} := \begin{cases} + & \text{nếu } b_{L-1} = 0 \\ - & \text{nếu } b_{L-1} = 2 \end{cases}$$

Ví dụ: Sub2 (3,311) = (3,133), Sub2 (3,111) = (3, −333)

2) Phép quay thuận: Rot^+

Trước hết, Rot^+ được định nghĩa trên tập hữu hạn $\{0, 1, 2, 3\}$

$Rot^+(0) = 1$, $Rot^+(1) = 3$,

$Rot^+(2) = 0$, $Rot^+(3) = 2$.

Sau đó Rot^+ được mở rộng một cách tự nhiên trên tập vô hạn $\{0, 1, 2, 3\}^*$:

$$Rot^+(xa) = Rot^+(x)\, Rot^+(a)$$

trong đó $x \in \{0, 1, 2, 3\}^*$, $a \in \{0, 1, 2, 3\}$.
Tiếp theo, Rot^+ được mở rộng một lần nữa trên tập các đỉnh của cây tứ phân

$$Rot^+(L, K) = (L, Rot^+(K)).$$

3) Phép quay nghịch: Rot^-

Rot^- được định nghĩa là: $Rot^- = (Rot^+)^{-1}$.

2.2. Một vài tính chất của cây tứ phân ảo

Ta có thể nhận được một số tính chất sau đây của cây tứ phân ảo dùng để biểu diễn ảnh nhị phân [8].

Tính chất 1: Mỗi đỉnh P = (L, K) ứng với một khối phần tư của ảnh có cỡ 2^{n-L}, nói riêng, nếu L = n thì đỉnh P ứng với một điểm ảnh.

Tính chất 2: Rot^+ (P) cho ta biểu diễn của chính đỉnh P sau khi quay ảnh một góc 90^o thuận chiều kim đồng hồ.

Tính chất 3: Rot^- (P) cho ta biểu diễn của chính đỉnh P sau khi quay ảnh một góc 90^o ngược chiều kim đồng hồ.

Định lý 4: Nếu Sign (Code Sub2(P))) > 0 thì Sub2(P) xác định khối phần tư giáp P về phía Bắc, ngược lại, P nằm sát biên phía bắc của ảnh.

Hệ quả 5: Cho biến side thuộc tập {Bắc, Tây, Nam, Đông} tập này được mã hóa bởi {0, 1, 2, 3}. Đặt P là một khối phần tư của ảnh 1. Khối phần tư Q được xác định bởi công thức

$$Q = Rot^{-side}(Sub2(Rot^{side}(P))).$$

Khi đó, nếu sign (Code (Sub2 (Rot^{side} (P)))) > 0 thì Q là khối phần tư nằm kề P về phía side và có cùng cỡ như P.

Tính chất 6: Nếu một điểm ảnh P = (L, K) có các tọa độ biểu diễn dưới dạng nhị phân bởi

$x = x_{n-1} \ldots x_1 x_0$, $y = y_{n-1} \ldots y_1 y_0$, $x_i, y_i \in \{0, 1\}$ thì ta có hệ thức sau

$$K = \sum_{i=0}^{i-1} (2y_i + x_i) \cdot 4^i.$$

Tính chất 7: Nếu đỉnh P ứng với một khối vuông $(L < n)$, thì các tọa độ của góc trên bên trái được tính bởi

$$x(P) = \sum_{i=0}^{i-1} (a_i \bmod 2) 2^{i+n-L}, \quad y(P) = \sum_{i=0}^{L-1} (a_i \operatorname{div} 2) 2^{i+n-L}.$$

Các tính chất trên đóng một vai trò cơ bản trong việc xây dựng các thuật toán xử lý ảnh trên biểu diễn mã tứ phân: tính mômen và các đặc trưng hình học của ảnh, phân tích tính liên thông của ảnh, đối sánh ảnh, chuyển đổi từ biểu diễn mã cây tứ phân sang các biểu diễn mã khác và ngược lại. Có thể xem các thuật toán chi tiết trong [8]

III – ĐÁNH GIÁ HIỆU QUẢ CỦA BIỂU DIỄN MÃ CÂY TỨ PHÂN SO VỚI BIỂU DIỄN KHÁC

Các kỹ thuật mã hóa ảnh khác nhau đều có một lợi điểm chung là rút gọn bộ nhớ lưu trữ ảnh. Tuy nhiên đối với một ảnh cụ thể, kỹ thuật mã hóa này lại tỏ ra tốt hơn kỹ thuật khác. Trong phần này ta sẽ chỉ ra các điều kiện để chuyển sang các loại mã hóa khác, khi cho ảnh biểu diễn bởi mã cây tứ phân

3.1. Đối với mã độ dài loạt

Giả sử một ảnh có cỡ $2^n \times 2^n$ được biểu diễn bởi cây tứ phân Q với Q đỉnh. Mỗi đỉnh P thuộc Q có 3 thuộc tính: Level (P), Code (P) và Color (P), chúng tương ứng cần $\log_2 n$ bit, 2n bit và 2 bit để lưu trữ. Vậy, bộ nhớ cần thiết cho Q là:

$$Q(2 + 2n + \log_2 n).$$

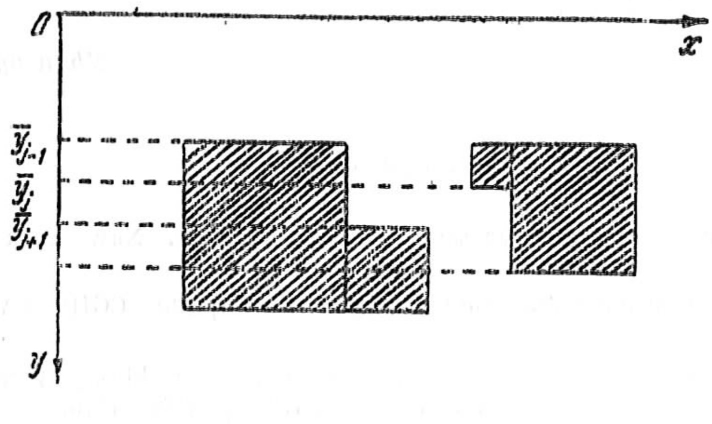
Với mỗi đỉnh $P = (L, K) \in Q$, theo các tính chất 1 và 7, ta định nghĩa các hàm

$$X(P) = \sum_{t=0}^{L-t} (a_i \bmod 2)\, 2^{i+n-L}$$

$$y(P) = \sum_{t=0}^{L-1} (a_i \operatorname{div} 2)\, 2^{i+n-L}$$

$$s(P) = 2^{n-L}.$$

Đặt $\mathcal{B}$ là tập các lá đen của cây tứ phân, ta có tập số nguyên BY được xác định như sau $BY = \{y(P) : P \in \mathcal{B}\} \cup \{y(P) + s(P) + 1,\ P \in \mathcal{B}\}$ Với quan hệ bằng nhau « = » theo nghĩa thông thường trên BY, ta có tập $\overline{BY} = BY/=$. Ký hiệu b là lực lượng của tập $\overline{BY}$ mỗi $\overline{y_j} \in \overline{BY}$ xác định dòng ảnh trên đó có thể xuất hiện một hình trạng mới (hình 2).



Hình 2: Các tọa độ y trên đó có thể xuất hiện hình trạng mới của loạt chạy các điểm ảnh

Để xác định số loạt chạy của các điểm ảnh, với mỗi $\overline{y_j} \in \overline{BY}$ được đặt tương ứng tập $H_j \in \mathcal{B}$ và quan hệ $\gamma_j \subset H_j \times H_j$ theo cách sau

$$H_j = \{P \in \mathcal{B} : \overline{y_j} \in [y(P), y(P) + s(P)]\}$$

$$P_m \gamma_j P_l \leftrightarrow x(P_m) = x(P_l) + s(P_l) + 1 \text{ hoặc } x(P_l) = x(P_m) + s(P_m) + 1 ; \; P_m, P_l \in H_j$$

Tiếp theo, nếu đặt γ_j^* là bao đóng phản xạ, bắc cầu của γ_j trên H_j thì ta có tập $H_j^* = H_j/\gamma_j^*$. mỗi phần tử của nó chứa các khối kề nhau liên tiếp theo chiều ngang.

Tổng số loạt chạy trong biểu diễn mã độ dài loạt sẽ là:

$$\sum_{j=1}^{b} (\overline{y}_{j+1} - \overline{y_j}) \operatorname{card} \sum (H_j^*), \quad \text{trong đó } \overline{y}_{b+1} = 2^n.$$

Như vậy, tổng số khối lượng bộ nhớ cần dùng cho ảnh trong biểu diễn mã độ dài loạt là

$$2(n+1) \sum_{j=1}^{b} (\overline{y_{j+1}} - \overline{y_j}) \operatorname{card} (H_j^*) \text{ bit.}$$

Dựa trên cơ sở đó, ta có thể kết luận rằng khi cho một ảnh biểu diễn bởi mã cây tứ phân với Q đỉnh, mã độ dài loạt sẽ tối ưu hơn về mặt lưu trữ khi

$$\sum_{j=1}^{b} (\overline{y_{j+1}} - \overline{y_j}) \operatorname{card} (H_j^*) < \frac{Q\,(2 + 2n + \log_2 n)}{2(n+1)}.$$

3. 2. Đối với mã dây chuyền

Với mỗi B thuộc tập các lá đen $\mathcal{B}$, bằng một thủ tục đơn giản (xem [9] có thể xác định được tập

$$N_B = \{P \in \mathcal{B} : P \text{ là kề với } B\}.$$

Độ dài biên thuộc B được tính bởi

$$4\,(n - \mathrm{Level}(B)) - \sum_{P \in N_B} \min\,(n - \mathrm{Level}(B),\, n - \mathrm{Level}(P))$$

Do đó số vecto chỉ phương trong biểu diễn mã dây chuyền là

$$4 \sum_{B \in \mathcal{B}} [n - \mathrm{Level}(B)] - \sum_{B \in \mathcal{B}} \sum_{P \in \mathcal{B}} \min\,(n - \mathrm{Level}(B),\, n - \mathrm{Level}(P)).$$

Ta nhận được công thức tính tổng số bộ nhớ cần dùng để lưu trữ ảnh trong mã dây chuyền

$$3\,(4 \sum_{B \in \mathcal{B}} n - \mathrm{Level}(B) - \sum_{B \in \mathcal{B}} \sum_{P \in N_B} \min\,(n - \mathrm{Level}(B),\, n - \mathrm{Level}(P))).$$

Đến đây, ta có thể kết luận rằng khi cho một ảnh biểu diễn bởi mã cây tứ phân, mã dây chuyền sẽ tối ưu hơn về mặt lưu trữ khi

$$4 \sum_{B \in \mathcal{B}} [n - \mathrm{Level}(B)] - \sum_{B \in \mathcal{B}} \sum_{P \in N_B} \min\,(n - \mathrm{Level}(B),\, n - \mathrm{Level}(P)) < \frac{Q(2 + 2n + \log_2 n)}{3}$$

Nhận ngày 21-4-1986

TÀI LIỆU THAM KHẢO

1. E.L. Hall, Computer image processing and recognition. New York – Academic Press 1979.

2. M. Onoe et al, Chromosome Analysis by minicomputer. CGIP – Vol 2, 1973, p. 402 – 416.

3. G.S. Sidhu, R.T.Boute, Property encoding: Application in binary picture encoding and boundary following. IEEE Trans. Comput. Vol C-21, 1972, p. 1206 – 1216.

(Xem tiếp trang 27)

J. of Comp. and Syst. Sciences 28, 167 — 189 (1984).

6. **Ullman J. D.**, Principles of Database Systems.
Comp. Science Prese 1980.

7. **Talbot S.**, An investigation into logical optimization of relational quary languages
The Computer Journal Vol 27, N°4, 1984.

8. **Berge C.**, Graphs and Hypergraphs 1973.

9. **Đỗ Xuân Thọ**, Biểu mở rộng và sự tương đương một lớp các biểu thức quan hệ.
Báo cáo tại Hội nghị toán học toàn quốc lần thứ 3 Hà-nội tháng 7/1985.

10. ДО СУАН ТХО, Эквивалентность одного класса реляционных выражений.
Acta Matematica Vietnamica N° 1, 1985.

ABSTRACT

Equivalences among simple generalized tableaux

Tableaux are used as a instrument for the optimization of query processing in relational database systems. But it has been proved that the problem of testing the equivalence of two tableaux is a NP - complete one. Therefore the problem which arises is to find such class of tableaux that the procedure for testing the equivalence of which is solvable in polynomial time. The class of simple tableaux possesses such property. The paper investigates the equivalence among simple generalized tableaux and conditions for the existence of corresponding relational expression to a simple generalized tableau.

MỘT CÁCH TIẾP CẬN...

Tiếp theo trang 18)

4. **H. Samet, A. Rosenfeld**, Quadtree structure for region processing. In Proceeding of IJCPR-80, 1980, p. 36 — 41.

5. **L.G. Shapiro**, Data structure for picture processing: A survey. CGIP — Vol 11, 1979, p. 162 — 184.

6. **L.P. Johu, S.S. Iyengar**, Space and Time efficient virtual Quadtree. IEEE Trans. PAMI, Vol PAMI-6, N°2, 1984, p. 244—248.

7. **P.N. Khôi, H. Kiểm**, Code conversion in image processing. Institute of Tech. Cybernetics, Bratislava, Rept. N°14, 1984.

8. **P.N. Khôi**, Image processing based on compact representation. Institute of Tech. Cybernetics, Bratislava, Rept., 1985.

9. **P.N. Khôi**, A data structure for quadtree codes and application. Kozlemények, Budapest. N°32, 198 .

ABSTRACT

An approach to represent image objects by shaps

The paper deals with a special data structure — virtual quadtree — for representing and approximating shaps of image objects. Some operator manipulating on this data structure is proposed and followed by their properties. The last section presents the evaluation of memory space efficiency of quadtree code in relation to run length code and chain code for a given image