Годишен зборник на ПМФ
Том 38 (1997)
Скопје, Р.Македонија

ISSN 0351-7441

Liljana Babinkostova

Homotopy classification and trees of binary images

Abstract. *In this paper, based on the algorithm of the combinatoric contraction,*[2] *and on its dual algorithm of the combinatoric pulling, one modification of the algorithm that assigns to any class of homotopy equivalent images a directed tree with root,*[1] *is constructed. All algorithms considered in the work have been realized in a PC.*

1. Introduction and preliminaries

We give first a list of definitions and results needed in what follows.

1.1. Pixels p and q $(p \neq q)$ are called *directed* neighbors, if they have a common edge, and *angular* neighbors if they have only one common vertex.

1.2. A set $O_X(Y)$ is called an *edging* of a set Y in X if

$$O_X(Y) = \{q \in X \mid q \text{ is a neighbor of } p,\ p \in X\ \}$$

1.3. An image X is called *4-connected* (*8-connected*) if for any two pixels $p.q \in X$ there is a sequence of pixels $p_0, p_1, \ldots, p_n \in X$, where $p_0 = p_n$ and any two neighboring pixels of this sequence are direct (angular) neighbors.

1.4. A set F=N\X is called the *background* of the image X .

Remark. Bellow we will always suppose the 8-connectedness for an image and 4-connectedness for a background .This is result from the attempt of the famous paradox in (Pavlidis [5]) .

1.5. A pixel $p \in X$ is called *isolated* if $O_X(p) = \varnothing$. An image X is called *discrete*, if any pixel of X is isolated .

1.6. Any bounded 4-connected component of F is called a *hole* in X .

1.7. An external background is a nonbounded 4-connected component of F .

2. Combinatoric homotopy equivalences

2.1. **Definition.** An *index* of a pixel $p \in X (ind_X(p))$ with respect to the binary image X is :

1) the number -1 , if p is an isolated pixel ;
2) the Euler characteristic of the complex constituted from edges and vertices of p and pixels from X which are its neighbours .

2.2. **Definition.** We shall say , that an image X_1 is obtained from an image X by an *elementary push* , if $X_1 = X \setminus p$ and $ind_X(p) = 1$.

An image X is obtained from an image X_1 by an *elementary pull* if $X_1 = X \setminus p$ and $ind_X(p) = 1$.

2.3. **Definition.** Two images X and Y are called *combinatorically homotopy equivalent* , if Y can be obtained from X by a sequence of elementary pushings and elementary pullings .

2.4. **Theorem.** An elementary pushing and an elementary pulling don't change the homotopy type of X .

Let us be given a tree T=(*V*,*E*) with the marked vertex r , which is called a *root.* As it is known, we can assign to any vertex of the tree a number which is called the level of the vertex.

Suppose that $level(r) = 0$ and $V_0 = \{r\}$. To all vertices $v \in V_1$, where $V_1 = \{v \in V \setminus V_0 | \ \{v,r\} \in E\}$ we assign the $level(v) = 1$; to all vertices $v \in V_2$, $V_2 = \{v \in V \setminus (V_0 \cup V_1) | \ \exists u \in V_1, \{u,v\} \in E\}$ we assign the $level(v) = 2$. Suppose that the vertices of levels 0,1,...,n are already defined. Suppose that the vertices of levels 0,1,..., n are already defined. The set of all vertices with level n+1 we define inductively

$$V_{n+1} = \{v \in V \setminus (V_0 \cup \cdots \cup V_n) | \ \exists u \in V_n ; \{u,v\} \in E\}$$

This procedure separates all vertices of T into levels. Hence, we obtain the directed tree with the following property :

$$\forall v \in E \setminus \{r\}, \ \exists ! \ v_1, v_2, \cdots, v_n : \ r \to v_1 \to \ \cdots \ \to v_n \to v \quad .$$

By $T = (V, E, r)$ we denote *directed tree* with a root r .

3.1. **Definition.** Two trees $T = (V, E, r)$ i $T' = (V', E', r')$ are called *isomorphic* $(T \cong T')$, if there exists a bijection $f : V \to V'$ such that :

1) $f(v) = v'$
2) $\forall u, v \in V$ if $u \to v \Rightarrow f(u) \to f(v)$.

First we will describe the procedure to assign a tree with a root to any class of combinatorial homotopy equivalent binary images.

Let $C_1, C_2, \cdots, C_n$ be components of X , and $F_0, F_1, \cdots, F_m$ be components of the background, where F_0 is the external background. First we assign to X an undirected graph T_X . The vertices of this graph are $C_1, C_2, \cdots, C_n, F_0, F_1, \cdots, F_m$, and the edges have the form $C_i - F_j$. The edge joins two vertices C_i and F_j, iff $p \in C_i$ and $q \in F_j$, which are neighbours.

The proofs of this statements is considered in [1] .

3.2. **Theorem.** The graph T_X is a tree.

3.3. **Theorem.** Images X and Y are combinatorically homotopy equivalent iff their trees T_X and T_Y are isomorphic.

Assuming that F_0 is the root of T_X , we obtain the directed tree, which is called the *tree of the binary image* X .

All algorithms considered in this work are based on the following algorithms .

Algorithm of combinatoric contraction (*ACC*)

Step 1. We search for pixel p in X, such that $p \in X \; : ind_X(p) = 1$. If it does not exist, then we end the algorithm.

Step 2. $X := X \setminus \{p\}$ and go to the step 1.

Algorithm for combinatoric pulling (*ACP*)

Step 1. We search for pixel $p \in F : ind_{X \cup p}(p) = 1$. If it does not exist, then we end the algorithm.

Step 2. $X := X \cup \{p\}$ and go to the step 1.

Algorithm to determine that two marked pixels belong to the same component (ISTKOMP)

Step 0. Suppose that *label*(p_0)=2 , *label*(q_0)=3 .

Step 1. We carry out ACC with regard for marked pixels :

1.1. We search for pixel p with the label 1,2,3 and $ind_X(p) = 1$. If it exist we go to 1.2. , else we end step1.

1.2. If p exist, then:

a)if *label*(p)=1, then we remove pixel p (*label*(p)=*0*) .

b)if *label*(p)>1, then we search neighbour q of p and verify *label*(q)=2 or *label*(q)=3 .

If this is true, we end the algorithm (p_0 and q_0 belong to the same component); else *label*(q) := *label*(p) i *label*(p) := *0* and go to step 1.

Step 2. We search for isolated pixels of X . If they do not exist, then we go to step 3, or

2.1. If there are marked pixels among isolated pixels we end the algorithm (p_0 and q_0 do not belong to the same component) .

2.2. If there are no marked pixels among isolated pixels, then we remove these pixels and go to step 3.

Step 3. We carry out ACP with the following removing of isolated pixels of the background and go to step 1.

3. The homotopy recognition algorithm

The construction of the tree consists of the construction of arrays *ORIG* and *RECORDS* for some numeration of components of *X* and *F*. During the work of the algorithm to any pixel of X or F it always assigns some list of numbers (these numbers will correspond to the components of the image or of the background, i.e. to the vertices of T_X). At the beginning all these lists are empty. They are constructed and modifed during the work of the algorithm. The pixels which have non-empty lists will be called marked. If it is necessary to remove a marked pixels of X (or of F), then its lists must be given to one of its neighbours in X (in F), i.e. to the list of a neighbouring pixel, the list of the removed pixel is added. Only after that we can remove the marked pixel.

The main algorithm used following algorithms:

Algorithm for marking the components of binary image and the background (NOM)

Step1. We search for a pixel $p \in X : ind_X(p) = 1$ and $label(p) = 1$. If p is not found then go to step3 .

Step2. We carry out ISTKOMP for any pixel $q \in X$, $q \neq p$. If pixels p,q belong to the same component, then $label(q) := 2k, label(p) := 2k$, (k = 1,2,...) and go to step1.

Step3. We search for a pixel $p \in F$: $ind_{X \cup \{p\}}(p) = 1$ and $label(p) = 0$. If p is not found we end the algorithm .

Step4. $label(q) := 2(k+1), label(p) := 2(k+1), (k = 0,1,2,\ldots)$ for any pixel $q \in F$, thus p and q can join with sequence of pixels from the background F and any two neighboring pixels of this sequence are direct neighbors .Then go to step3.

Algorithm for marked combinatory contraction (*NAKK*)

*Step*1. We search for a pixel $p \in X : ind_X(p) = 1$ and $label(p) = 2k$,(k= 1,2,...) If p is not found we end the algorithm .

*Step*2. $X := X \setminus \{p\}$ and $label(p) := label(q)$ for any pixel $q \in F$ which belong to the component of p from F and go to step1 .

Algorithm for marked combinatory pulling (*NAKP*)

*Step*1.We search for a pixel $p \in F$: $ind_{X \cup \{p\}}(p) = 1$ and $label(p) = 2(k+1)$, (k=0,1,...).If p is not found we end the algorithm .

*Step*2. $X := X \cup \{p\}$ and $label(p) := label(q)$ for any pixel $q \in X$ which belong to the component of p from X and go to step1 .

Algorithm for construction of T_X

Step 1. We carry out (*NOM*) for binary image X .

Step 2. We carry out (*NAKK*).

Step 3. We search for a pixel $p \in X : ind_X(p) = -1$. If p is not found ,then we go to step 5 .

Step 4. We write the numbers $(label(q), label(p))$ in the list of incidence, where $q \in O_N(p)$; $label(p) := label(q)$ and go to step 3 .

Step 5. We carry out (*NAKP*).

Step 6. We search for a pixel $p \in F$: $ind_{X \cup \{p\}}(p) = 0$. If p is not found , then we go to step 8 .

Step 7. We write the numbers $(label(q), label(p))$ in the list of incidence, where $q \in O_N(p)$; $label(p) := label(q)$ and go to step 6.

Step 8. If $label(p) = 1$ for any pixel $p \in N$, then we go to step 9, else we go to step 2 .

Step 9. We draw the tree T_X according to the list of incidence .

End of algorithm .

REFERENCES

[1] A. Bykov & L. Zerkalov, Algorithms for Homotopy Classification of Binary Images, Pattern Recognition (1996).

[2] E.V.Shchepin & G.M.Nepomnyashchyi ,On Topology Analysis of Images, Mezvuzovskii sbornik nauchnih trutov , Moskva , (1990) , 13-25 .

[3] J.P.Mylopoulos & T.Pavlidis, Journal of the Association for Computing Machinery,Vol.18, No.2, (1971), 239-146.

[4] G.Tourlakis & J.P.Mylopoulos, Some results in Computational Topology, Journal of the Association for Computing Machinery,Vol.20, No.3, (1973), 439-455.

[5] E. M. Patterson , Topology , Interscience publishers , New York (1963).

[6] E. Spanier , Algebraic Topology , McGraw Hill , New York (1966).

[7] T.Pavlidis , Algorithms for Graphics and Image Processing , Computer Science Press (1982).

Бабинкостова Лилјана

Хомотопска класификација и дрва на бинарни слики

Резиме

Во овој труд, користејќи ги алгоритмите за комбинаторна контракција и комбинаторно проширување[(2)], направена е една модификација на алгоритмот за придружување ориентирано дрво на секоја класа од комбинаторно хомотопно еквивалетни бинарни слики[(1)]. Направена е компјутерска реализација на сите алгоритми содржани во овој труд.