

三维工业物体的一种关系数据结构 及协调标号算法

王新华 李德仁

(武汉测绘科技大学电子信息工程与精密机械学院,武汉市珞喻路 39号, 430070)

摘要 提出一种用于描述三维工业物体的关系数据结构, 将关系匹配归结为一个关系同构或协调标号问题。提出了3种修剪算子, 它充分利用关系数据结构的多重约束、一对一映射约束、关系符号表与基元符号表的映射作为知识启发来修剪搜索树, 从而大大简化了物体匹配中的协调标号或关系同构问题。

关键词 识别; 关系数据结构; 关系同构; 协调标号问题; 关系匹配

分类号 P235. 2 TP391; TP311. 12

摄影测量原理结合模式识别、计算机视觉等技术用于三维工业物体的识别、测量和定位的关键在于知识的表达方式, 即如何描述3D物体。它不仅决定了计算机视觉低级处理的任务和目标, 而且也决定了高级处理的算法。目前一些学者提出了许多物体的描述方法^[1~4], 其中关系结构描述是重要的描述方式之一。在关系数据结构描述中, 由于子图匹配具有指型计算的复杂性, 另外如此简单的图描述还远不能解决问题。对此, Tsai和Fu^[5]提出了属性关系图的概念, 使图的节点和边附带一些属性, 从而使图的描述能力大大增强, 也在一定程度上简化了匹配问题。但它对低级处理也提出了更高的要求。Shapiro^[6, 7]提出了一种广义的关系描述方法, 即关系数据结构。在这种描述下, 物体匹配变成了关系同构或关系同态问题。但他在寻找关系同构过程中提出的Rof算法并没有解决计算的复杂性问题。

关系同态是人工智能中著名的约束满足问题, 它可以表达为协调标号问题^[8], 其计算的复杂性仍没有解决。Haralick^[8]提出的算法尽管很有效, 但它往往从数学角度考虑问题。实际上, 我们可以利用物体的结构约束使标号协调问题大大简化。本文首先给出了一种利用结构约束来描述的简单物体的关系数据结构, 然后给出了利用这些约束简化标号问题的修剪算法, 这些修剪算法充分利用了关系数据结构的多重约束、一对一映射约束和关系符号表的映射约束作为知识启发。

1 描述3D物体的关系数据结构

一个关系数据结构是一个二元组 $D = (U, T)$, 其中 U 是基元集合(每个基元可附带有属性), 形如 $U = (p, x)$ 。其中 p 为结点符号, $x = (x_1, \dots, x_n)$ 为结点表征向量; $T = \{T_1, \dots, T_k\}$ 是关系集合, 每一个关系 $T_i (i=1, \dots, k)$ 是一个子集集合, 且 $T_i = \{t_{i1}, \dots, t_{im}\}$, 式中 $t_{ij} \subseteq U$, 且 \leqslant

$|t_{ij}| \leq |U|$ 。这里符号 $|U|$ 表示集合 U 的基数。

T_i 的子集 t 可以是有序或无序的,并可带有表示向量表达属性。

下面我们以图1所示的工业零件为例讨论该物体的关系数据结构。图1a为物体,图1b为模型。考虑到计算机视觉低级处理的困难,在我们的视觉系统中均采用面作为基元,而且基元可以带有相应的基元属性,如类型、面积、参数描述等。基元间的关系根据其几何或结构约束,可定义为临接交面关系 T_1 、平行关系 T_2 ,即

$$T_1 = \{(u_1, \dots, u_n) \mid u_i \in U, u_i \text{ 交面} \cap u_j \text{ 交面}, i = 1, \dots, n\}$$

$$T_2 = \{(u_1, \dots, u_n) \mid u_i \in U, u_i \text{ 相互平行}, i = 1, \dots, n\}$$

应指出,上式中的 T_i 可以是无序对集合,也可以是有序对集合。另外,关系也可带有关系属性,如夹角、交面、连接方式等。同时,上述的关系描述不是唯一的。为讨论问题方便起见,这里仅以上述两种关系为例。太多的关系描述(结构约束)虽然对标号协调有利,但会给低级处理带来很大困难。基于上述关系描述,我们可以得出图1所示的工业零件的关系描述如表1~4所示。表中 U 为物体基元集, T 表示其关系集, L 为模型的基元集, S 为模型关系集, 该物体和模型的关系数据结构分别为 $D_a = (U, T)$, $D_b = (L, S)$ 。应说明的是, 在关系子集 T_1 中, 引入了关系属性——交面。

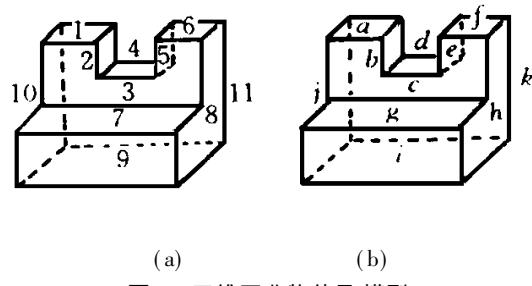


图1 三维工业物体及模型

表1 临接交面关系 T_1		
t_{ij}	基元	交面
t_{11}	2, 3, 10, 11	1
t_{12}	1, 3, 4, 11	2
t_{13}	1, 2, 4, 5, 6, 7, 8, 10	3
t_{14}	2, 3, 5, 11	4
t_{15}	3, 4, 6, 11	5
t_{16}	3, 5, 8, 11	6
t_{17}	3, 8, 9, 10	7
t_{18}	3, 6, 7, 9, 11	8
t_{19}	7, 8, 10	9
t_{110}	1, 3, 7, 9, 11	10
t_{111}	1, 2, 4, 5, 6, 8, 10	11

表2 临接交面关系 S_1		
s_{ij}	符号	交面
s_{11}	b, c, j, k	a
s_{12}	a, c, d, k	b
s_{13}	a, b, d, e, f, g, h, j	c
s_{14}	b, c, e, k	d
s_{15}	c, d, f, k	e
s_{16}	c, e, h, k	f
s_{17}	c, h, i, j	g
s_{18}	c, f, g, i, k	h
s_{19}	g, h, j	i
s_{110}	a, c, g, i, k	j
s_{111}	a, b, d, e, f, h, j	k

表3 平行关系 T_2	
t_{ij}	u_i
t_{21}	1, 4, 6, 7
t_{22}	2, 5, 8, 10
t_{23}	3, 9, 11

表4 平行关系 S_2	
s_{2i}	s_i
s_{21}	a, d, f, g
s_{22}	b, e, h, j
s_{23}	c, i, k

2 协调标号中的修剪算法

协调标号问题可表达为一个四元组 (U, L, T, R) , 其中 U 为观测物体的基元集 $U = \{u_1, u_2, \dots, u_n\}$, L 为模型的基元符号集 $L = \{l_1, l_2, \dots, l_n\}$, T 为物体的结构约束构成的基元约束关系, 形如 $U = \{u_1, u_2, \dots, u_n\}$, 它是 N 元组的集合, $u \in U$ 一般地并不是所有 N 组合或排列都可给 (u_1, \dots, u_n) 标号, 因此 R 就代表了对于一组 (u_1, \dots, u_n) 哪些 (l_1, \dots, l_n) 是允许的组合标号, 我们把它称为基元符号约束关系。它由模型的结构约束以及 T 中的约束关系来构成。这样, 我们可用各种树搜索算法来协调符号。在现有的算法中大都把全部约束归在一起搜索^[6], 从而造成计算量大。为此, 我们引入下述的修剪算法。

匹配要求所有匹配上的基元在结构上完全相同, 因此任何结构约束均可作为知识来修剪

树。最简单的方法是利用基元属性进行修剪,它可定义为一个基元 符号表:

$$H = \{(H(u_1), H(u_2), \dots, H(u_m))\} \quad (1)$$

式中, $H(u_i)$ 是可给基元 u_i 符号的集合。在此基础上可以根据各种约束进一步修剪。

2.1 利用关系子集修剪

设对一物体,已知其关系数据结构 $D_0 = (U, T)$ 和模型关系数据结构 $D_m = (L, S)$, 其中 U 、 L 分别为基元集和符号集, $T = \{T_1, \dots, T_k\}$, $S = \{S_1, \dots, S_k\}$, T_j 和 S_j 表示某种关系。我们定义一个关系子集表 F 为:

$$F(t) = \{S \in S_j \mid |S| \geq |t|\} \quad (2)$$

式中, t 与 S 有相同或相似的关系属性。

$F(t)$ 代表了可标给 t 的符号关系子集的集合。 $|S| \geq |t|$ 的物理意义是由于遮挡原因, 物体的某些基元可能丢失, 因此标给这些基元关系子集的符号关系子集的个数可以比它们大, 符号 $| \cdot |$ 表示子集的基数。

令 $w(u)$ 表示所有包含基元 u 的基元关系子集的集合, 即

$$w(u) = \{t \in T_i \mid t \in u\} \quad (3)$$

则

$$H_j(u) = \bigcap_{t \in w(u)} \bigcup_{S \in F(t)} S \quad (4)$$

式中 $H_j(u)$ 是根据约束 j , 所有可标给基元 u 的符号的集合。显然, 我们希望找到的关系子同构能满足所有 k 个关系, 故最终的基元 符号表 H 应为:

$$H(u) = \bigcap_{j=1}^k H_j(u) \quad u \in U, j = 1, \dots, k \quad (5)$$

2.2 利用一对映射修剪

令 U 、 L 分别为基元集和符号集, 若有 j 个基元 $(u_1, \dots, u_j) \in U$, 使得 $H(u_i) \subseteq L' \subset L$, 且 $|L'| = j$, 这意味着这 j 个符号只能标给这 j 个基元, 其它基元不能再标出 L' 中的符号。

对于子集 (U_n, L_n) , 其中 $U_n \subseteq U$, $|U_n| = n$, $L_n \subseteq L$, $|L_n| = m$, 且对于任何 $u \in U_n$, $H(u) \subseteq L_n$, 则若

(1) $m < n$, 则关系子同构不存在, 即一个符号不能同时标给两个以上基元, 一对映射不存在

(2) $m > n$, 不作修剪。

(3) $m = n$, 此时修剪算子定义为:

$$ZH(u_i) = \{l \in H(u_i) \mid \text{若 } u_i \notin U_n, l \notin L_n\} \quad (6)$$

而且

$$ZH = \{ZH(u_i)\} \quad i = 1, 2, \dots, n \quad (7)$$

式中 $u \in U$, 修剪算子可以迭代地应用于 H , 直到 H 满足:

$$Z^{*-1} H = Z H \quad (8)$$

一般地在修剪过程中, 我们总是从具有最少基数的子集对开始进行上述修剪。同样的道理, 对于关系子同构, 关系子集的映射也是一对一映射的, 故 Z 算子同样适用于关系子集表 F 的修剪。

2.3 利用关系符号表和基元符号表的映射修剪

当我们利用不同的结构约束形成了相应的关系符号表和基元符号表后, 事实上, 关系符号表与基元符号表之间存在的映射关系也可以帮助我们进一步修剪搜索树。

若对于基元符号表 $H(u)$, 我们已知了一个子集对 (U_n, L_n) , 其中 $U_n \subseteq U$, $L_n \subseteq L$ 。这意味着这 n 个符号 L_n 只能标给这 n 个基元 U_n , 其它基元不能再标以这些符号, 这种关系应该适用于

关系符号表中。另一方面,对于约束关系 T_i ,如果存在某基元关系子集 t_j 包含 U_n ,并且存在相应的某符号关系子集 S_k 包含 L_n ,令 $K(u)$ 表示包含基元 u 的基元关系子集, $u \in U_n$,令 $f(l)$ 表示包含 l 的符号关系子集, $l \in L_n$ 。则定义修剪算子 V 为:

$$VH_i(u) = \{l \in f(l) \mid \text{若 } u \notin K(u), \text{ 则 } l \notin f(l)\} \quad (9)$$

实际上,我们总是从基元符号表 H 中具有最少基数的子集开始进行 V 操作,直到满足:

$$V^{k-1} H_i = V^k H_i \quad (10)$$

应指出,同样也可以采用基元符号表与关系属性的映射进行 V 算子修剪。

3 实例

对于图 1 所示的被观测物体 (a) 和匹配模型 (b),表 1~4 分别列出了其相应的关系数据结构,其中在约束关系 T_1 中我们引入了关系属性——交面。

根据公式(2)以及约束关系 T_1 , T_2 , 可得相应的关系子集表 F_1 , F_2 (如表 5, 6 所示)。再根据公式(3), (4)可以得基于关系约束的各基元符号表 H_1 , H_2 (如表 7, 8 所示)。显然,最终的基元符号表应该同时满足 H_1 和 H_2 , 故使用公式(5)可得各基元符号表求交后的基元符号表 H , 见表 9 第 1 列。然后根据 H 与 T_1 , T_2 的映射关系进行 V 算子修剪, 例如采用基元 {8, 10} 和符号 $\{h, j\}$ 组成的子集对, 对关系符号子集表 t_2 和 s_2 的映射进行 V 算子修剪后, 可得新的基元符号表如表 9 第 2 列。对该符号表进行 Z 算子修剪可得 Z_H 。继续利用基元符号表 H 和 T_1 , T_2 的映射关系进行 V 算子修剪和 Z 算子修剪。表 9 第 4 列就是利用 H 和 T_1 的映射进行 V 算子操作的结果, 注意这里采用的是 T_1 中关系属性与 H 的映射。重复 V 操作和 Z 操作可得第 5 列。可以看出, 该列已是一个充分修剪的搜索树, 可以说已经完成了修剪任务。从被处理的对象形状上我们不难发现, 引起个别基元标号不定性(如 1, 2, 5, 6, 8, 10)的原因是物体的对称性。如果假设基元 1 分别标以标号 a 和 f , 则利用基元符号表 H 和 T_1 中关系属性的映射, 反复进行 V 及 Z 算子修剪, 不难得出两种相应的最终标号结果分别如第 6 列和第 7 列所示。如果我们建立了基元的属性描述, 则容易发现第 7 列才是真正协调的结果。

表 5 F_1

t_{ij}	$F_1(t_{ij})$
t_{11}	$S_{11}, S_{12}, S_{14}, S_{15}, S_{16}, S_{17}$
t_{12}	$S_{11}, S_{12}, S_{14}, S_{15}, S_{16}, S_{17}$
t_{13}	S_{13}
t_{14}	$S_{11}, S_{12}, S_{14}, S_{15}, S_{16}, S_{17}$
t_{15}	$S_{11}, S_{12}, S_{14}, S_{15}, S_{16}, S_{17}$
t_{16}	$S_{11}, S_{12}, S_{14}, S_{15}, S_{16}, S_{17}$
t_{17}	$S_{11}, S_{12}, S_{14}, S_{15}, S_{16}, S_{17}$
t_{18}	S_{18}, S_{10}
t_{19}	S_{19}
t_{110}	S_{18}, S_{10}
t_{111}	S_{111}

表 6 F_2

t_{2j}	$F_2(t_{2j})$
t_{21}	S_{21}, S_{22}
t_{22}	S_{21}, S_{22}
t_{23}	S_{23}

表 7 H_1

u	符 号
1	a, b, d, e, f, g
2	a, b, d, e, f, g
3	c
4	a, b, d, e, f, g
5	a, b, d, e, f, g
6	a, b, d, e, f, g
7	a, b, d, e, f, g
8	h, j
9	i
10	h, j
11	k

表 8 H_2

ui	符 号
1	a, b, d, e, f, g, h, j
2	a, b, d, e, f, g, h, j
3	c, i, k
4	a, b, d, e, f, g, h, j
5	a, b, d, e, f, g, h, j
6	a, b, d, e, f, g, h, j
7	a, b, d, e, f, g, h, j
8	a, b, d, e, f, g, h, j
9	c, i, k
10	a, b, d, e, f, g, h, j
11	c, i, k

使用本文的关系数据结构描述三维物体, 其标号协调算法可概括为下述几个步骤:

① 通过关系描述 T_1 , T_2 构造相应的关系子集表。

② 由关系子集表按公式(3), (4)构造相应的基元符号表 H_1 , H_2 , 然后按公式(5)求 H_1 和 H_2 的交集。

③根据基元符号表和关系符号子集(或关系属性)的映射对基元符号表作V算子修剪,一般每作一次V操作,同时进行一次Z算子操作,直到满足公式(8)和(10)式为止。

表 9

u_i	$H(H \cap H_2)$	$\nabla H(\leftrightarrow T_2)$	ZH	$\nabla(H \rightarrow T_1)$	$\nabla(H \rightarrow T_1)$	H	H
1	a, b, d, e, f, g	a, b, d, e, f, g	a, d, f, g	$a, d, f,$	a, f	f	a
2	a, b, d, e, f, g	$b, e,$	b, e	b, e	b, e	e	b
3	c	c	c	c	c	c	c
4	a, b, d, e, f, g	a, b, d, e, f, g	a, d, f, g	a, d, f	d	d	d
5	a, b, d, e, f, g	b, e	b, e	b, e	b, e	b	e
6	a, b, d, e, f, g	a, b, d, e, f, g	a, d, f, g	a, d, f	a, f	a	f
7	a, b, d, e, f, g	a, b, d, e, f, g	a, d, f, g	g	g	g	g
8	h, j	h, j	h, j	h, j	h, j	j	h
9	i	i	i	i	i	i	i
10	h, j	h, j	h, j	h, j	h, j	h	j
11	k	k	k	k	k	k	k

通过上述算法和实例不难看出,基于关系数据结构的描述使得3D物体的关系描述非常灵活。本文提出的利用子集、一对一映射、基元符号表和关系符号表的映射等修剪算法能大大减少协调标号(寻找关系同态)的过程,尤其是结合关系属性和基元属性进行修剪更为有效。经过上述修剪后,寻找关系同态(同构)的搜索树已基本上是一个充分修剪的树。若在此基础上,充分利用基元属性描述(每个基元属性均可构造相应的特征向量)构造代价函数,并以此评价标号结果,则可以实现精匹配。

应指出,对于更复杂的识别问题,只要我们适当地采用更多的基元属性和关系属性约束,同样能得到很好的修剪效果,文中的协调标号算法仍然有效。还应指出的是,文中的思想具有一般性,它同样适用于以曲面(柱面、锥面、球面等)为基元的情况,只要我们找到适当的结构(几何)约束来设计相应的关系数据结构,文中的各种修剪算法同样有效。

参 考 文 献

- 1 Flynn P J, Jain A K. CAD-based Computer Vision: From Model to Relation Graphs. IEEE Trans. PAMI, 1991, 13(2): 114~ 132
- 2 Wang X. CAD-based Line Feature Objects Measurement and Reconstruction. SPIE Proc. on the Fourth Int. Conf. on CAD/CV, 1995. 701~ 706
- 3 Fan T J, Medien G, Nevatia R. Recognition 3D Objects Using Surface Description. IEEE Trans. PAMI, 1986, 5(3): 27~ 52
- 4 Chen T W. A Neural Network Approach to CSG-based 3D Objects Recognition. IEEE Trans. PAMI, 1994, 16(7): 719~ 726
- 5 Tsai W H, Fu K. Error-correcting Isomorphisms of Attributed Relational Graphs for Pattern Analysis. IEEE Trans. SMC, 1979, 9(12): 757~ 768
- 6 Shapiro L G, Moriarty J D. Matching Three-dimensional Objects Using a Relational Paradigm. Pattern Recognition, 1984, 17(4): 385~ 405
- 7 Shapiro L G, Haralick R M. Relational Matching. Applied Optics, 1987, 26(10): 1845~ 1851
- 8 Haralick R M, Shapiro L G. The Consistent Labeling Problem. IEEE Trans. PAMI, 1980, 2(3): 193~ 203

- Photogrammetry and Remote Sensing, 1990(28): 34~ 39
- 5 Zhang Zuxun. Extraction of the Concept of Feature Bridge Mode Method and Global Image Matching. In Proc. Photogrammetry and Remote Sensing. Wuhan WTUSM, 1992. 25~ 35
- 6 Genis C T. Relaxation and Neural Learning Points of Convergence and Divergence. J Parallel Distributed Comput., 1989(6): 998~ 1 012
- 7 Hopfield J J. Neurons with Graded Response Have Collective Computational Properties Like Those of Two-state Neurons. In Proc. Natn. Acad. Sci. USA, 1984. 324~ 332

Image Matching by Neural Network-Based Relaxation

Shen Weiming Zhang Zuxun Zhang Jianqing

(LIESM ARS, WTUSM, 39 Luoyu Road, Wuhan, China, 430070)

Abstract This paper proposes a mapping method that makes the Hopfield net perform the relaxation process. In such a way, image matching by relaxation process can be realized in real time since the Hopfield net can be implemented by conventional analog circuits.

Key words image matching; relaxation labelling; neural network

(上接第 246页)

Relational Data Structure and Consistent Labelling Algorithms for 3D Industrial Objects

Wang Xinhua Li Deren

(School of Electronic Information Engineering & Precision Mechanics, WTUSM, 39 Luoyu Road, Wuhan, China, 430070)

Abstract In this paper, authors present a kind of relational data structure which is used to describe 3D industrial objects. Thus, the relational matching problem results in the problem of relational isomorphism or consistent labelling. In the procedure of consistent labelling, authors present three pruning algorithms, that use the multi-constraint, one to one correspondence constraint of relational data structure, the correspondence between relation-label tables and unit-label tables as the knowledge to prune the searching tree. These approaches simplify the problem of relational isomorphism or consistent labelling in relational matching. Finally, an example and its discussion are given.

Key words recognition; relational data structure; relational isomorphism; consistent labelling problem; relation