

利用标准线性四元-八元树由正交 三轮廓图识别三维客体*

张田文 李仲荣

(哈尔滨工业大学, 哈尔滨)

摘要 线性八元树提供一种非常紧凑并适于传统客体处理运算的三维客体体积表示法。然而, 利用体积交技术由正交三轮廓图生成三维客体的线性八元树表示取决于观察点。匹配客体到模型达到识别目的需要客体表示独立于观察点。为了得到独立于观察点的表示, 应通过计算惯量矩阵的特征向量确定客体的三个主轴。线性八元树投影到三个主视(沿着主轴)的图象平面上得到三个标准线性四元树。客体匹配包括两步过程: 第一步是利用对称差的度量匹配待识别客体的标准线性四元树到库中一模型子集; 第二步是生成客体和这些候选模型的标准线性八元树, 然后匹配待识别客体的标准线性八元树到具有最小对称差的模型。

关键词 标准线性四元-八元树; 正交三轮廓图; 主轴; 客体匹配; 对称差

一、引言

有效的三维客体表示在计算机图形学、机器视觉和计算机辅助设计等方面起着重要作用。人们在实践中根据数据的获取方法和具体应用提出了三维客体的各种表示^[1]。近年来, 许多研究集中在线性八元树表示^[2,3]。它是二维图象的线性四元树表示推广到三维客体的一种体积表示法。如果连续分割包括客体的整立方体成八分体、八分体成子八分体、直至各个八分体具有单一标号(FULL 或 VOID), 客体的线性八元树表示为具有标准尺寸和位置的标号 FULL 八分体集合。作为线性八元树表示三维客体的一个例子, 考察图 1(a) 所示的采样客体。若采用整立方体 $2^3 \times 2^3 \times 2^3$, 它的块分割及其客体中各个八分体的编码如图 1(b) 所示。该采样客体的线性八元树可以表达为

$C = \{00x, 013, 02x, 03x, 04x, 12x, 20x, 21x, 22x, 23x, 26x, 30x, 31x, 32x, 33x, 6xx\}$

一般地, 如果八分体 c^i 的编码中含有 p 个 x , 称 c^i 为 p -八分体, 其中 p 是 c^i 的阶。整立方体 $2^n \times 2^n \times 2^n$ 记作 $U_n = \underbrace{xx \cdots x}_n$, 显然 U_n 是 n -立方体。在计算机上实现时, 各

个八分体 c^i 的编码用四个存储单元表示, 分别记作 V^i, T_1^i, T_2^i 和 T_3^i ^[3]。对于上述线性八元树, 若令 $c^i = 26x$, 它在机器中的具体数据结构: $V^i = 110, T_1^i = 010, T_2^i = 110, T_3^i = 000$ 。可见 $p = \|V^i\|$, 其中 $\|V^i\|$ 表示 V^i 中 0 的数目。

本文讨论利用标准线性四元-八元树由客体的正交三轮廓图识别它的方法。对于外形

* 1989年10月16日收到。

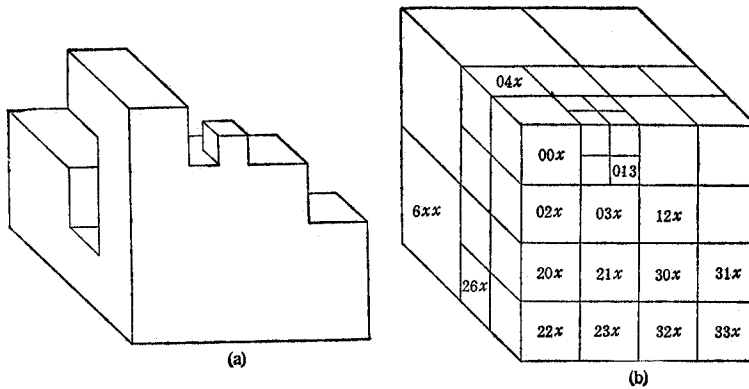


图 1

轮廓没有凹面的一大类三维客体如飞机、导弹和坦克等，可以利用体积交技术由客体的正交三轮廓图生成它的线性八元树表示。为了识别客体，客体的具体表示应是其位置、取向和相对尺寸的不变量，上述线性八元树不满足这种要求，因为它取决于正交三座标的方位。为此，这里提出一种客体的标准线性四元-八元树表示。首先以客体的惯量矩阵作为线性变换计算它的特征向量得到客体的三个主轴(或者称为主方向)，由正交三轮廓图生成的线性八元树投影到客体三个主方向的图象平面上得到客体的三个标准线性四元树表示。三个标准线性四元树通过体积交生成的线性八元树称为客体的标准线性八元树表示。标准线性四元-八元树仅仅取决于客体的形状，而不受其位置、取向和相对尺寸的影响，因此，它们可以作为形状描述符用于客体匹配。匹配过程分两步进行：“粗”匹配——利用对称差将待识别客体的三个标准线性四元树匹配到库(每个模型用三个标准线性四元树表示)中一模型子集；“精”匹配——分别生成待识别客体和子集中每个模型的标准线性八元树，待识别客体的标准线性八元树匹配到子集中具有最小对称差的模型，从而完成识别任务。

二、利用正交三轮廓图生成线性八元树

利用客体的正交三轮廓图通过体积交生成它的线性八元树表示包括下列步骤：首先产生三个正交轮廓图各自对应的线性四元树(有关线性四元树可参考文献[4])；将各个线性四元树变换成对应轮廓图“扫射”体积的线性八元树——伪线性八元树，再通过三个伪线性八元树的交生成客体的线性八元树表示。

通常，轮廓图的直接表示是二元数组。当轮廓图转换成线性四元树时，首先应将二元数组转换成0-四分形集合，数组中每个“1”元素对应一个0-四分形。若形成正交三轮廓图的方向如图2所示， F 、 T 和 S 分别表示前、俯和侧视方向。图3(a-c)分别表示 F 、 T 和 S 方向的二元数组 $A_F = [a_{ef}]_{2^2 \times 2^2}$ 、 $A_T = [a_{et}]_{2^2 \times 2^2}$ 、 $A_S = [a_{st}]_{2^2 \times 2^2}$ 中各个元素和 $U_n = xx$ 时各个0-四分形编码之间的对应关系。这些0-四分形和以后要用到的更高阶四分形在机器中的具体数据结构可以统一到与八分体相同的形式

$$\begin{aligned}
 c_j^i & 0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ x \\
 U_j^i & 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 0 \\
 r_{i,j}^i & 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 0 \\
 s_{i,j}^i & 0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 0 \\
 t_{i,j}^i & 0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 0
 \end{aligned}$$

利用文献[5]中的算法 1 可以把 A_F 转换成对应的 0-四分形集合 Q_F^0 。对于 A_T 和 A_S 也有类似算法。然后利用四分形算符#-积^[4]把 0-四分形集合 Q^0 转换成线性四元树 Q ，分别有

$$\begin{aligned}
 Q_F &= U_n \# (U_n \# Q_F^0) \\
 Q_T &= U_n \# (U_n \# Q_T^0) \\
 Q_S &= U_n \# (U_n \# Q_S^0)
 \end{aligned}$$

为了利用体积交生成线性八元树，需要分别产生三个正交线性四元树对应的伪线性八元树。所谓伪线性八元树就是从一个方向 (F 、 T 或 S) 看去，对应线性四元树复盖体积的线性八元树表示。一线性四元树包含的四分形复盖体积的八分体集合组成对应的伪线性八元树。下面以线性四元树 Q_F 为例说明伪线性八元树的形成方法。若 $c^i \in Q_F$ 在 U_n 中复盖体积的八分体子集用 C^i 表示，它包含 $2^{n-||V^i||}$ 个成员，根据前面所述的四分形编码的具体数据结构， c^i 是 C^i 的成员， C^i 的其余成员为 F 方向上 c^i 依次面邻接的同阶八分体序列。因此 C^i 中各个八分体具有相同的 V 、 T_2 和 T_3 ， T_1 在 T_1^i 的基础上依次递增 $2^{||V^i||}$ 。例如，若令 $c^i = 12x$ ，则有 $C^i = \{12x, 16x, 52x, 56x\}$ 。对于 Q_F 对应的伪线性八元树 C_F ，有

$$C_F = \bigcup_{c^i \in Q_F} C^i$$

文献[5]中的算法 2 给出具体步骤。计算 Q_T 和 Q_S 对应的伪线性八元树 C_T 和 C_S 也有类似算法。客体的线性八元树 (C) 表示是通过计算它的三个正交伪线性八元树 C_F 、 C_T 和 C_S 的交^[4]

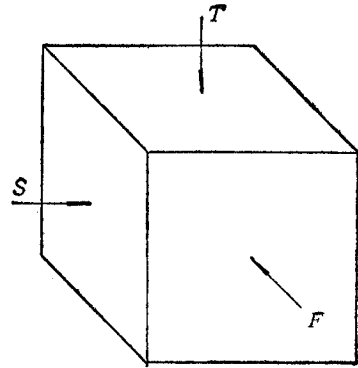


图 2

00	01	10	11
0x		1x	
02	03	12	13
xx			
20	21	30	31
2x		3x	
22	23	32	33

(a)

44	45	54	55
4x		5x	
40	41	50	51
xx			
04	05	14	15
0x		1x	
00	01	10	11

(b)

44	40	04	00
4x		0x	
46	42	06	02
xx			
64	60	24	20
6x		2x	
66	62	26	22

(c)

图 3

$$C = C_F \cap C_T \cap C_S$$

得到的。

例如,给出三个二元数组 A_F 、 A_T 和 A_S 如图 4 所示。利用上述方法可以得到

1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1
0	0	0	0	1	1	1	1
0	0	0	0	1	1	1	1
0	0	0	0	1	1	1	1
0	0	0	0	1	1	1	1
0	0	0	0	1	1	1	1
0	0	0	0	1	1	1	1
0	0	0	0	1	1	1	1

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	1	1	0	0
0	0	0	0	1	1	0	0
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
0	0	0	0	1	1	1	1
0	0	0	0	1	1	1	1
0	0	0	0	1	1	1	1
0	0	0	0	1	1	1	1

0	0	1	1	1	1	1	1
0	0	1	1	1	1	1	1
0	0	0	0	1	1	1	1
0	0	0	0	1	1	1	1
0	0	0	0	1	1	1	1
0	0	0	0	1	1	1	1
0	0	0	0	1	1	1	1
0	0	0	0	1	1	1	1
0	0	0	0	1	1	1	1
0	0	0	0	1	1	1	1

图 4

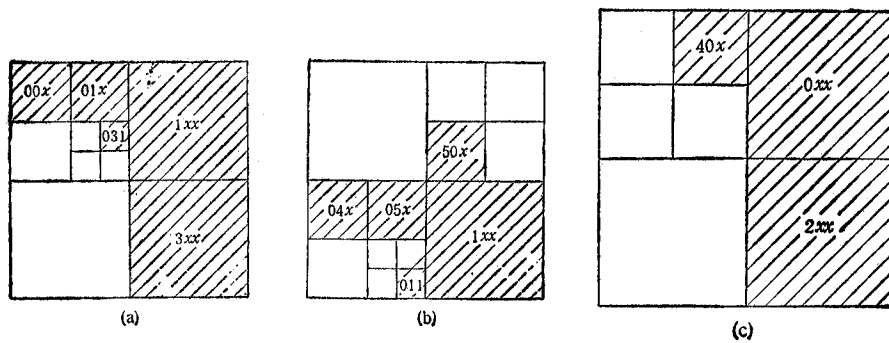


图 5

$$Q_F = \{00x, 01x, 031, 1xx, 3xx\}$$

$$Q_T = \{011, 04x, 05x, 1xx, 50x\}$$

$$Q_S = \{0xx, 2xx, 40x\}$$

它们依次表示在图 5 (a—c) 中。 Q_F 、 Q_T 和 Q_S 对应的伪线性八元树分别为

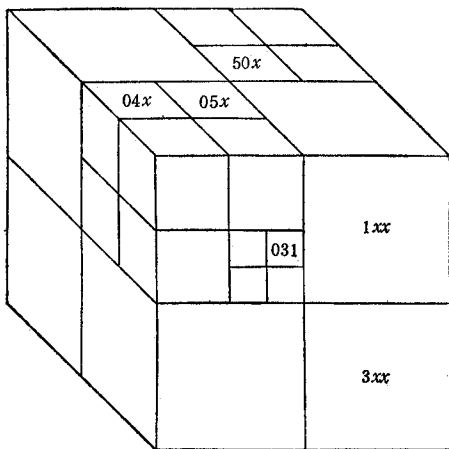


图 6

$$C_F = \{00x, 01x, 031, 035, 04x, 05x, 071, 075, 1xx, 3xx, 40x, 41x, 431, 435, 44x, 45x, 471, 475, 5xx, 7xx\}$$

$$C_T = \{011, 013, 031, 033, 04x, 05x, 06x, 07x, 1xx, 211, 213, 231, 233, 24x, 25x, 26x, 27x, 3xx, 50x, 52x, 70x, 72x\}$$

$$C_S = \{0xx, 1xx, 2xx, 3xx, 40x, 41x, 50x, 51x\}$$

进而,有

$$C = \{031, 04x, 05x, 1xx, 3xx, 50x\}$$

如图 6 所示。

三、标准线性四元-八元树

三维客体的线性八元树表示可以用于模式识别, 但利用上一节提出的方法得到的线性八元树取决于观察点, 即受客体的位置、取向和相对尺寸的影响。为了执行客体匹配, 线性八元树应是客体平移、旋转和相似变化的不变量, 即独立于观察点。为此, 利用客体的惯量矩阵^[6]

$$I = \begin{bmatrix} I_{11} & I_{12} & I_{13} \\ I_{12} & I_{22} & I_{23} \\ I_{13} & I_{23} & I_{33} \end{bmatrix} = \begin{bmatrix} M_{020} + M_{002} & -M_{110} & -M_{101} \\ -M_{110} & M_{002} + M_{200} & -M_{011} \\ -M_{101} & -M_{011} & M_{200} + M_{020} \end{bmatrix}$$

作为线性变换 T , 并使得

$$T \begin{bmatrix} x \\ y \\ z \end{bmatrix} = K \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

则称 K 为 T 的特征值, T 的特征向量 $([x, y, z])$ 对应于客体的三个主轴。主轴是客体的几何特性, 它是客体平移、旋转和相似变化的不变量。垂直于各个主轴的图象平面称为主平面, 由正交三轮廓图生成的线性八元树分别投影到三个主平面得到的线性四元树称为标准线性四元树, 三个标准线性四元树通过体积交生成的线性八元树称为标准线性八元树。

此时, 得到的标准线性四元-八元树只与客体的位置和取向无关, 还没有解决相似变化的影响。若 Q^1 和 Q^2 是两个标准线性四元树, 形成它们的整正方形分别为 U_a 和 U_b , 且 $a > b$, 即 Q^1 和 Q^2 包含的四分形编码分别有 a 和 b 个分量。如果将 Q^2 中各个四分形编码 c^i 对应的 V^i 、 T_1^i 、 T_2^i 和 T_3^i 分别左移 $(a - b)$ 位, 有 $Q^1 = Q^2$, 则称 Q^1 和 Q^2 相似, 上述处理过程称为相似处理。例如, $Q^1 = \{23x\}$ 和 $Q^2 = \{23\}$, 若以 8 位字节为一个存贮单元, 对于 $c^i = 23x$ 和 $c^i = 23$, 分别有 $V^i = 00000110$, $T_1^i = 00000000$, $T_2^i = 00000110$, $T_3^i = 00000010$, $V^i = 00000011$, $T_1^i = 00000000$, $T_2^i = 00000011$, $T_3^i = 00000001$ 。若将 c^i 的 V^i 、 T_1^i 、 T_2^i 和 T_3^i 分别左移一位后对应的四分形用 $c^i(L_1)$ 表示, 显然有 $c^i = c^i(L_1)$, 故 Q^1 和 Q^2 相似。

模型库中各个客体模型是用三个标准线性四元树 Q_R 、 Q_T 和 Q_S 表示, 若生成所有客体模型的整正方形均为 U_m , 形成各个待识别客体的三个标准线性四元树的整正方形用 U_i 表示, 并且对于所有的 i 有 $m \geq i$ 。若某一待识别客体的标准线性四元树 Q_R^i 、 Q_T^i 、 Q_S^i 包含的四分形具有 n 个分量 (即对应的整正方形为 U_n), 按照上述相似处理规则应将各个 $c^i \in Q_R^i$ 、 Q_T^i 、 Q_S^i 对应的 V^i 、 T_1^i 、 T_2^i 和 T_3^i 左移 $(m - n)$ 位。经过相似处理的标准线性四元树以及由这样的标准线性四元树通过体积交生成的标准线性八元树不仅是客体平移和旋转的不变量, 而且也是相似变化的不变量, 因此可以把它们作为客体的形状描述符用于客体识别。

四、客 体 匹 配

客体匹配是利用两个标准线性四元(八元)树对称差的定量分析过程。若 Q^1 和 Q^2 是两个经过相似处理(即在同一 U_m 中编码)的标准线性四元树, Q^1 和 Q^2 的对称差集 $Q^1 \Delta Q^2$ 定义为

$$Q^1 \Delta Q^2 = (Q^1 \# Q^2) \cup (Q^2 \# Q^1)$$

$Q^1 \Delta Q^2$ 是包含在 Q^1 而不在 Q^2 中和包含在 Q^2 而不在 Q^1 中的四分形集合。若

$$Q^1 \Delta Q^2 = \{c^1, c^2, \dots, c^r\},$$

Q^1 和 Q^2 的对称差 $S(Q^1 \Delta Q^2)$ 定义为

$$S(Q^1 \Delta Q^2) = \sum_{i=1}^r 2^{2^{i-1}|V^i|}$$

即对称差集 $Q^1 \Delta Q^2$ 中各个四分形的面积和。

客体匹配过程分两步进行: 第一步的粗匹配是利用待识别客体的三个标准线性四元树 Q_F^i 、 Q_T^i 、 Q_S^i 和库中客体模型 $Q_{F^i}^{m_i}$ 、 $Q_{T^i}^{m_i}$ 、 $Q_{S^i}^{m_i}$ 的对称差平均值

$$S(Q^i \Delta Q^{m_i}) = \frac{S(Q_F^i \Delta Q_{F^i}^{m_i}) + S(Q_T^i \Delta Q_{T^i}^{m_i}) + S(Q_S^i \Delta Q_{S^i}^{m_i})}{3}$$

将它匹配到具有较小对称差的模型子集 M' , 子集中的模型作为候选匹配者; 第二步的精匹配需要分别生成待识别客体的标准线性八元树 C^i 和子集中诸模型的标准线性八元树 C^{m_i} , $m_i \in M'$, 然后利用线性八元树的对称差

$$S(C^i \Delta C^{m_i}) = \sum 2^{2^{i-1}|V^i|}$$

即对称差集 $C^i \Delta C^{m_i}$ 中八分体的体积和, 将待识别客体匹配到子集中具有最小对称差的模型, 从而完成识别任务。需要指出, 为了消除 U_m 取值的影响, 通常可以取对称差相对于整正方形(立方体)的面积(体积)百分数, 即相对对称差。

五、结 束 语

本文提出的利用标准线性四元-八元树由正交三轮廓图识别三维客体的整个过程是一种代数方法。它比标准四元-八元树表示客体的识别方法^[6]要优越得多, 因为从轮廓图生成八元树, 利用八元树确定客体主轴以及根据标准四元-八元树计算对称差的所有算法都需要树遍历操作, 耗用机时太多。

如果客体的轮廓图是从非共平面(不必正交)的三个方向获得的, 利用体积交得到的线性八元树称为广义线性八元树, 其中各个八分体是其面分别平行于三个给定方向的平行六面体。此时, 为了确定客体的主轴, 仅仅需要惯量矩阵从三个给定方向指定的广义座标系统变换到正交座标系统, 计算变换后惯量矩阵的特征向量可以得到客体的三个主轴^[6]。

参 考 文 献

- [1] P. J. Besl, R. C. Jain, *Computing Surveys*, 17(1985), 135—145.
- [2] I. Gargantini, *Computer Graphics Image Processing*, 20(1982), 365—374.
- [3] Zhang Tianwen, Li Zhongrong, An Octant Structure for Three-Dimensional Objects, in Proceedings, 3rd International Technical Symposium on Optical and Optoelectronic Applied Sciences and Engineering, Innsbruck, Austria, (1986).
- [4] 张田文,李仲荣,自动化学报, 16(1989)1,45—49.
- [5] 张田文,李仲荣,宇航学报,待发表.
- [6] C. H. Chien, J. K. Aggarwal, Reconstruction and Matching of 3-D Objects Using Quadrees/Octrees, in Proceedings, 3rd Workshop on Computer Vision, Bellare, USA, (1985), 49—54.

IDENTIFICATION OF 3-D OBJECTS FROM THREE ORTHOGONAL VIEWS USING NORMALIZED LINEAR QUAD-OCTREES

Zhang Tianwen Li Zhongrong
(Harbin Institute of Technology, Harbin)

Abstract Linear octrees offer a volume representation of 3-D objects, which is quite compact and lends itself to traditional object processing operations. However, the linear octree structure for generating the representation of 3-D objects from three orthogonal views by using the volume intersection technique is dependent on viewpoints. The recognition achieved from matching object representations to model representations requires that the representations of objects are independent on viewpoints. In order to obtain independent representations on viewpoint, the three principal axes of the object should be found from the moment of inertia matrix by computing its eigenvectors. The linear octree is projected onto the image planes of the three principal views (along the principal axes) to obtain the three normalized linear quadrees. The object matching procedure has two phases: the first phase is to match the normalized linear quadrees of the unknown object to a subset of models contained in a library, utilizing a measure of symmetric difference; the second phase is to generate the normalized linear octrees of the object and these selected models and then match the normalized linear octree of the unknown object to the model with the minimum symmetric difference.

Key words Normalized linear quad-octrees; Three orthogonal views; Principal axes; Object matching; Symmetric difference