

面向三维重建的工程图信息预处理的研究

Research on pre-processing of Engineering Drawing information for 3D Reconstruction

赵晓峰 孙静波 孔庆 (长春工业大学 计算机科学与工程学院 吉林长春 130012)

摘要: 本文在基于多视图边界自动识别的基础上实现了视图分离,研究了确定投影原点坐标的算法,并建立起了空间投影坐标系,同时提出了利用数据库技术来存储图元的几何信息和拓扑信息。该方法为工程图的三维重建打下了良好的技术基础。

关键词: 工程图 视图分离 投影原点 数据库

在现实生活中,二维工程图是表达工程人员设计思想的重要手段,尽管现在已涌现出很多三维设计软件(如 SolidWork, SolidEdge, Pro/E 等),但在相当长时期内二维工程图仍将发挥重要作用,三维重建系统的研究日益迫切。而在进行工程图的三维重建时,二维视图投影信息的获取和预处理是个重要的基础性工作,它关系后面三维重建的复杂度和处理的效率与效果。从已有的三维重建预处理算法来看,大多数要求有一定的交互性,处理的自动化程度不高。本文在基于多视图边界自动识别的基础上实现了视图分离和空间投影坐标系的建立,笔者所研究的确定投影原点坐标的算法更接近实际的工程图样,同时提出了利用数据库技术来存储图元的几何信息和拓扑信息,为接下来进行数据分析和三维建模打下了基础。

1 视图分离

视图分离是进行三维重建的第一步,分离的目的是为了识别。工程图大多数是由多个视图组成,而以往的分算法一般基于三视图,这时可通过计算各连通域的凸包矩形来达到分离的目的,本文的分算法基于多视图边界自动识别,主要过程如下:

(1) 过滤掉视图中的辅助图元(如尺寸标注、各种文字信息等)。

(2) 进行多视图边界的自动识别,确定每个视图的外边界。设所识别出的外边界的个数为 n ,将每个外边界及其所包含的基本图元(直线、圆、圆弧)构成

一个集合 $ER_i (i=1,2,\dots,n)$,则 ER_i 构成了一个视图。

(3) 设 $EN_i \in ER_i (i=1,2,\dots,m, m$ 为第 i 个视图中所包含的基本图元的数量)为第 i 个视图中的任意一个图元。从每个视图的图元集合中各任取一个图元 $EN_i \in ER_i (i=1,2,\dots,n)$,并提取该图元的特征点的坐标,比较各个特征点的相对位置,便可确定该特征点所在的视图属于哪个视图。以三视图为例:设 $P_1(P_{1x}, P_{1y})$ 、 $P_2(P_{2x}, P_{2y})$ 、 $P_3(P_{3x}, P_{3y})$ 为分别属于从三个视图中任选出的三个基本图元的特征点,则:这三个点中,具有最大 x 坐标值的点所对应的视图便为左视图,其余两点中,具有较大 y 坐标的点所对应的视图为主视图,余者为俯视图。

2 视图坐标转换

2.1 不同坐标系下的表示形式

利用 CAD 软件所绘制的二维工程图通常是在如图 1 所示的坐标系下进行的,而三维重建的目标坐标系是如图 2 所示的坐标系,因此,它们之间需要进行转换。工程图样中,为了便于画图和看图,通常采用如图 3 所示的坐标系,该坐标系是把空间三个投影面展开在一个平面上,具体展开方法参看相关书籍。

2.2 投影原点的确定

为了建立如图 3 所示的视图间的空间投影坐标系,需在绘图坐标系上选取一个合适的点 (x_0, y_0) 作为空间投影坐标系的原点坐标。投影原点要满足以该点为中心所建立的坐标系使得三视图处于各自的象限

里,同时要满足三维重建的基本出发点。为了使算法

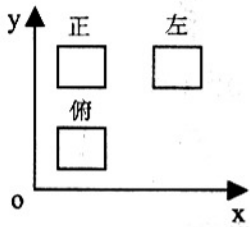


图 1 绘图坐标系

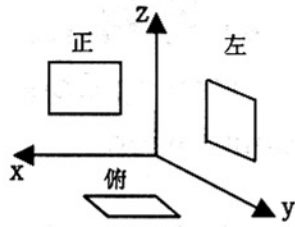


图 2 空间投影坐标系

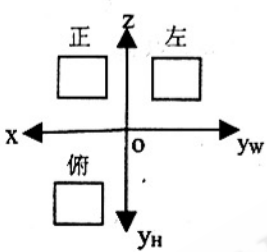


图 3 展开后的三面投影体系

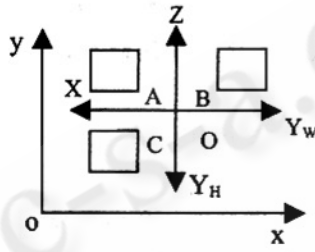


图 4 坐标转换

更具一般性,并不要求主视图和左视图之间以及主视图和俯视图之间分别是严格对齐的,处理过程如下:

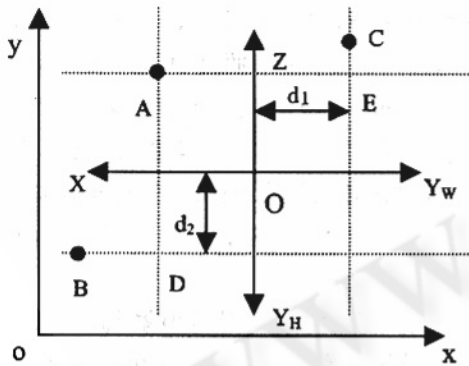


图 5 确定投影原点

(1) 分别做三个视图的凸包矩形,在 3 个矩形上各选取一个顶点,如图 4,坐标分别为 $A(x_1, y_1)$, $B(x_2, y_2)$ 和 $C(x_3, y_3)$ 。

(2) 计算 $\text{Max}(x_1, x_2)$,过该值所在顶点作 y 轴平行线,然后计算 $\text{Min}(y_1, y_3)$,过该值所在顶点作 x 轴平行线,接着过 B 作 x 轴平行线,过 C 作 y 轴平行线,则这四条平行线围成一个矩形,以矩形内任一点为中心建立坐标系都可使得三视图处于各自的象限里。

(3) 如图 5,将俯视图由 B 点平移至 D 点,将左视图由 C 点平移至 E 点,这样投影原点的选取只需使得 $d_1 = d_2$,记 $O(x_0, y_0)$ 为投影原点,则 $y_0 - y_2 = x_3 - x_0$,移项得到 $x_0 + y_0 = x_3 + y_2$ 。

可见,投影原点的纵横坐标之和是一个定值,确定一个便可得到第二个。

2.3 坐标转换

当空间投影坐标原点在视图平面上的对应原点确定后,就可以将各视图在统一坐标系下的二维坐标转换成三维空间坐标。关于这方面主要是建立映射算子,有关论文已有说明,这里不再赘述。

3 图元几何信息的存储

将图元的几何信息和拓扑信息进行完整的存储是进行后期重建的基础,本文通过灵活设计数据库,不仅使得重建效率大大提高,而且发现错误易于修改。常见的图元有:直线、圆弧、圆,等,每个图元都有自己的特征点和特征信息,如直线的特征点为端点,圆、圆弧的特征点是圆心,特征信息如圆的半径、圆弧的起始角和终止角等等。通过对 DXF 文件进行读取,完全可以找到所有图元的几何信息,我们将这些信息以数据库的方式进行存储。

表的设计与表间关系如下图:

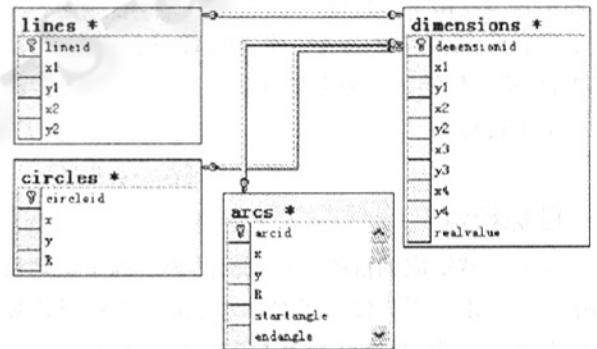


图 6 表间关系

假定在一幅工程图中绘制每种图元的数量不超过 999,我们设定表 lines、circles、arcs 的主键是分别是以 1、1000、2000 为种子数自动增长的, dimensions 表中含有所有图元的真实信息,同时多段线通过打断函

(下转第 54 页)

(上接第 49 页)

数被分解成若干条线段,顶点信息包含在 lines 表中。为了提高查询效率,dimensionid 的值是由程序自动生成的,具体的生成策略是:根据所标注图元的不同,dimensionid 的值或者为 lineid、或者为 circleid、或者为 arcid,同时还要考虑另外一种情况,标注的是两个图元,比如两条直线所成的角度,这时候我们设定 dimensionid 以 3000 为种子数自动增长。表 dimensions 的内容根据不同的图元会有所不同,比如标注直线用的 x_1 、 y_1 、 x_2 、 y_2 ,realvalue 代表线段的长度,而标注圆则只用 x_1 、 y_1 ,代表的是圆心坐标,realvalue 则代表半径的长度,标注两条直线的角度则所有字段都用到。

参考文献

- 1 韩云鹏、孟磊,面向三维重建的工程图信息自动获取与处理,制造业信息化,2004.
- 2 张爱军、诸昌铃等,面向三维重建的视图坐标转换,西南交通大学学报,2001.2.
- 3 谭建荣、张树有等,图学基础教程[M],北京:高等教育出版社,2006.5.
- 4 张彩明、杨兴强等,计算机图形学[M],北京:科学出版社,2005.2.
- 5 陆国栋、张树有等,工程图样数字化转换与智能理解[M],北京:机械工业出版社,2001.5.