

一种快速的多边形裁剪算法

黄文晓

(新疆伊犁财贸学校, 新疆 伊宁 835000)

摘 要: 提出并实现了一种有效的基于任意窗口的任意多边形裁剪算法. 通过简单的预处理和有效的数学方法, 不需要做复杂的“内点”、“外点”以及“进点”、“出点”的讨论就可准确地确定裁剪线段或折线. 建立了简单且有效的数据结构, 使得匹配这些有效线段和有效折线的过程十分简洁, 从而可快速输出裁剪结果.

关键词: 窗口; 多边形; 裁剪; 计算机图形学

中图分类号: TP391 **文献标识码:** A **文章编号:** 1673—999X (2011) 03—0052—05

多边形裁剪求窗口和多边形的交集, 是计算机图形学的一个基本问题, 解决该问题经典的算法有 Sutherland-Hodgeman 多边形裁剪算法和 Weiler-Atherton 多边形裁剪算法^[1]. Sutherland-Hodgeman 算法适合处理凸多边形而不宜处理凹多边形, 而 Weiler-Atherton 算法适合任意多边形, 但求交计算量却很大. 除了上述的两个经典算法之外, 还有许多的优秀算法, 如文献[2, 3]提出了“基于编码”的新算法, 适合于矩形窗口; 文献[4, 5, 6, 7, 8]提出的算法适合任意多边形, 在处理“内点”、“外点”, “进点”与“出点”的问题上给出了详细的讨论并提出了解决方案. 文献[9]对 Weiler-Atherton 算法做了改进, 节省了存储空间, 也提高了裁剪速度. 文献[10, 11]给出的算法不仅能处理一般的多边形, 还能够处理“自交多边形”. 文献[12, 13, 14]对求交步骤提出了新算法. 以上提到的算法, 几乎都是采用直观有效的算法确定窗口和多边形交集的线段. 本文试采用数学方法提出一种新算法, 可以裁剪任意多边形(凸、凹多边形)而不需要讨论“内点”和“外点”, 就可准确确定窗口和多边形交集的线段, 同时在“进”、“出”点的确定, 线段或折线的匹配等问题上给出了简洁且有效的解决办法, 并且很适合“并行计算”.

在提出算法之前, 先做一些准备工作.

1 定理

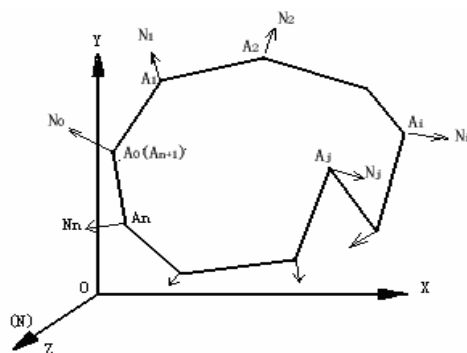


图 1 在 XOY 平面上, 待剪切的多边形

在提出定理之前, 先约定一些符号的意义:

在平面 XOY 上的一多边形 $\{A_0, A_1, A_2, \dots, A_n, A_{n+1}\}$, 其顶点按顺时针编号(见图 1), 符号 $\overline{N_i} (i=1, 2, 3, \dots, n)$ 表示多边形第 i 条边 $A_i A_{i+1} (i=1, 2, 3, \dots, n)$ 的法向量, 其定义为 $\overline{N_i} = \overline{N} \times \overline{A_i A_{i+1}} (i=0, 1, 2, 3, \dots, n)$, 其中 \overline{N} 的方向与 Z 轴正向相同. 直观上, 若 $\overline{N_i} (i=0, 1, 2, 3, \dots, n)$ 不经过所在边的端点, 显然指向所在边的外侧(线段 $A_i A_{i+1}$ 有两侧, 由距离线段 $A_i A_{i+1}$ 充分近的点组成, 但不包括与线段 $A_i A_{i+1}$ 共线的点, 一侧在多边形的内部, 另一侧在多边形的外部, 本文分别称这两侧为“内

收稿日期: 2011—06—18

作者简介: 黄文晓(1959—), 男, 讲师, 研究方向: 生物数学、数学建模.

侧”和“外侧”).

另外,我们约定:折线以及多边形不能“自交”,折线或多边形相邻的三个顶点不在同一条直线上.本文的“窗口”及“多边形”,其顶点均按顺时针编号,直线或线段用直线的参数方程表示.

引理 1 在平面内,设有向线段 p_0p_2 和一个多边形 $\{A_0, A_1, A_2, \dots, A_n, A_{n+1}\}$ (多边形各边的法向量的意义如上所述) 相交,其交点集设为 $\{q_i | q_i = \text{有向线段 } p_0p_2 \cap A_iA_{i+1} \text{ 且 } (i=0,1,2,3,\dots,n) q_i \neq A_i, A_{i+1}\}$.

(1) 若 $\overline{N_i} \cdot \overline{p_0p_2} < 0$, (“ \cdot ”表示“点积”) 则有向线段 p_0p_2 从边 A_iA_{i+1} 的外侧通过 q_i 点进入内侧;

(2) 若 $\overline{N_i} \cdot \overline{p_0p_2} > 0$, 则有向线段 p_0p_2 从边 A_iA_{i+1} 的内侧通过 q_i 点进入外侧.

证明: (1) 两向量点积小于零, 则两向量夹角大于 90° . 因为 $\overline{N_i}$ 指向边 A_iA_{i+1} 的外侧, 所以 $\overline{p_0p_2}$ 只能指向边 A_iA_{i+1} 的内侧.

(2) 证明过程和 (1) 类似.

引理 2 设有向线段 p_0p_2 和多边形相交, 其交点按线段 p_0p_2 方向顺序排列如下:

$$q_1, q_2, q_3, \dots, q_i, q_{i+1}, \dots, q_m,$$

其中 $m \geq 2$, 且可以有重合交点. 记这些交点所在的边为:

$$A_{j_1,1}A_{j_1+1,1}, A_{j_2,2}A_{j_2+1,2}, \dots, A_{j_i,i}A_{j_i+1,i}, \dots, A_{j_m,m}A_{j_m+1,m},$$

并记这些边对应的法向量为

$$\overline{n_1}, \overline{n_2}, \overline{n_3}, \dots, \overline{n_i}, \overline{n_{i+1}}, \dots, \overline{n_m},$$

(1) 若 $\overline{n_i} \cdot \overline{p_0p_2} < 0$ 且

$$\overline{n_{i+1}} \cdot \overline{p_0p_2} > 0 \quad (i \in \{1, 2, 3, \dots, m-1\}),$$

则线段 q_iq_{i+1} (其中 q_i 与 q_{i+1} 是相邻的两个交点) 在多边形内部, 如图 2 中线段 q_1q_2 的情形.

(2) 若 $\overline{n_i} \cdot \overline{p_0p_2} > 0$ 且 $\overline{n_{i+1}} \cdot \overline{p_0p_2} < 0$, 则线段 q_iq_{i+1} 在多边形的外部, 如图 2 中线段 q_2q_3 的情形.

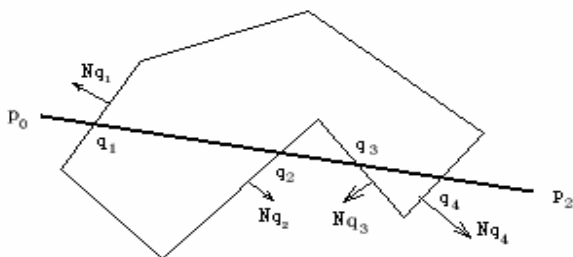


图 2 有向直线 p_0p_2 和多边形比较

证明: (1) 设 $q_i \neq q_{i+1}$ (这时 q_i, q_{i+1} 显然不是多边形的顶点), 用反证法: 若有一外部或者边界点 $P \in \text{线段 } q_iq_{i+1} \subset \text{有向线段 } p_0p_2$, 由于多边形的内部是一个开集 (外部也是开集), 由引理 1 可知, 存在一正整数 ε , 在区间 $(q_i, q_i + \varepsilon)$ ($\subset \text{线段 } q_iq_{i+1} \subset \text{有向线段 } p_0p_2$) 内至少有一个内部点, 不妨设为 q'_i 点. 由直线的连续性可知, 在 $(q'_i, P]$ ($\subset \text{有向线段 } p_0p_2$) 中必有一个边界点, 而这个边界点只能是有向线段 p_0p_2 与多边形的一个交点, 介于 q_i 与 q_{i+1} 之间, 这和“ q_i 与 q_{i+1} 是相邻的两个交点”矛盾. 故线段 q_iq_{i+1} 在多边形的内部. 当 $q_i = q_{i+1}$ 时, “线段” q_iq_{i+1} 在多边形的一个顶点上.

(2) 同理可证, 线段 q_iq_{i+1} 在多边形的外部.

定理 设 p_0p_2 是由若干段有向线段首尾相连构成的折线, 其中 p_0 是折线的始点, p_2 是折线的终点. 若折线 p_0p_2 和多边形相交, 其交点按 p_0 到 p_2 顺序排列如下: $q_1, q_2, q_3, \dots, q_i, q_{i+1}, \dots, q_m$ ($m \geq 2$, 且可以有重合交点). 记这些交点所在的直线段为: $s_{1,1}s_{1,2}, s_{2,1}s_{2,2}, \dots, s_{3,1}s_{3,2}, \dots, s_{m,1}s_{m,2}$. 并记这些交点所在多边形的边对应的法向量为 $\overline{n_1}, \overline{n_2}, \overline{n_3}, \dots, \overline{n_i}, \overline{n_{i+1}}, \dots, \overline{n_m}$.

(1) 若 $\overline{n_i} \cdot \overline{s_{i,1}s_{i,2}} < 0$ 且 $\overline{n_{i+1}} \cdot \overline{s_{i+1,1}s_{i+1,2}} > 0$ ($i \in \{1, 2, 3, \dots, m-1\}$), 则折线 q_iq_{i+1} (其中 q_i 与 q_{i+1} 是相邻的两个交点) 在多边形内部, 如图 3 中折线 q_1q_2 的情形.

(2) 若 $\overline{n_i} \cdot \overline{s_{i,1}s_{i,2}} > 0$ 且 $\overline{n_{i+1}} \cdot \overline{s_{i+1,1}s_{i+1,2}} < 0$ ($i \in \{1, 2, 3, \dots, m-1\}$), 则折线 q_iq_{i+1} (其中 q_i 与 q_{i+1} 是相邻的两个交点) 在多边形的外部, 如图 3 中折线 q_9q_{10} 的情形.

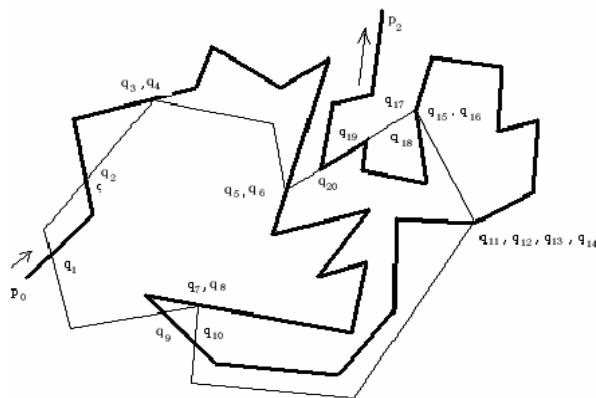


图 3 折线 (图中粗线者) p_0p_2 和多边形相交

在证明本定理之前,先讨论折线和多边形相交的相关问题:

(1)折线进入或出离多边形的方式.折线进入多边形的方式有两种,一是折线上一线段的一个端点(也是折线和多边形的一个交点)作为折线含在多边形内部的始点,如图4(a);二是折线上一线段的一个非端点(也是折线和多边形的一个交点)作为折线含在多边形内部的始点,如图4(b).

折线出离多边形的方式也有两种,一是折线上一线段的一个端点(也是折线和多边形的一个交点)作为折线出离多边形内部的始点,如图4(b);二是折线上一线段的一个非端点(也是折线和多边形的一个交点)作为折线出离多边形内部的始点,如图4(a).

(2)交点个数的计算.折线和多边形的交点集是由折线中所有线段和多边形所有的边求交而得,这样该交点集中就会出现相同的元素,如图3中的 q_3, q_4 相同, q_5, q_6 相同, q_7, q_8 相同, 相同 q_9, q_{10} , q_{11}, q_{12} 相同, q_{13}, q_{14} 相同, …….

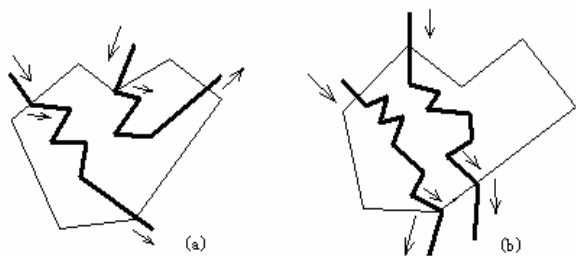


图4 折线(图中粗线者)进入或出离多边形方式的分类

证明:先证明定理的“(1)”部分,设 $q_i \neq q_{i+1}$:

①设 q_i 与 q_{i+1} 之间是两段线段构成的折线(设两线段的公共点为 q),用反证法:

若有一边界点 $P \in$ 折线 $q_i q_{i+1}$,且 $P \neq q_i, q_{i+1}$,这时容易证明 q_i, q, q_{i+1} 等三点同在一直线上,并且 $\overline{n_i \cdot s_{i,1} s_{i,2}} = 0$ 或 $\overline{n_{i+1} \cdot s_{i+1,1} s_{i+1,2}} = 0$,和条件相矛盾;

若有外部点 $P \in$ 折线 $q_i q_{i+1}$,由于 q_i 与 q_{i+1} 之间只有两段线段 $q_i q$ 与 $q q_{i+1}$,所以不妨设 $P \in$ 线段 $q_i q$,由于“内部”是“开集”,由引理1可知,存在一正数“ ε ”,使得在区间 $(q_i, q_i + \varepsilon) \subset (q_i, q)$ 上存在一内部点 q'_i .这样在线段 $q_i q$ 上就含有外部点 P 和内部点 q'_i ,由直线的连续性可知,在区间

(q_i, q) 存在一个边界点,这边界点就是折线 $p_0 p_2$ 和多边形的一个交点,夹在 q_i 与 q_{i+1} 之间,这和“ q_i 和 q_{i+1} 相邻”相矛盾.所以, $P \notin$ 线段 $q_i q$,同理可证 $P \notin$ 线段 $q q_{i+1}$.因此, q_i 与 q_{i+1} 之间的折线必含在多边形的内部.

②若 q_i 与 q_{i+1} 之间的折线是由多于两段线段构成,交点 q_i 所属的线段记为 $p_i q_i$,而 q_{i+1} 所属的线段记为 $p_m q_{i+1}$,而中间的线段则记为 $p_1 p_2, p_2 p_3, \dots, p_{m-1} p_m$,用反证法:若折线 $q_i q_{i+1}$ 上有一点 P 在多边形外部或者边界上,且 $P \neq q_i, q_{i+1}$,由①的讨论, $P \notin$ 线段 $p_i q_i$ 、 $P \notin$ 线段 $p_m q_{i+1}$,那只能 $P \in$ 线段 $[p_j, p_{j+1}] (j=1, 2, \dots, m-1)$.如果 $P \in$ 线段 $p_1 p_2$,由于 $[p_1, p_1 + \varepsilon) \subset$ 线段 $p_1 p_2$ 中存在内部点,记该内部点为 p' (\in 线段 $p_1 p_2$).由直线的连续性可知,线段 $[p', p]$ 上必存在一个边界点,这一个边界点也是折线 $p_0 p_2$ 与多边形的一个交点,夹在 q_i 与 q_{i+1} 之间,这和“ q_i 与 q_{i+1} 相邻”相矛盾.所以 $P \notin$ 线段 $p_1 p_2$.

用穷举法,可证得 $P \in [p_j, p_{j+1}] (j=1, 2, \dots, m-1)$.因此, q_i 与 q_{i+1} 之间的折线只能在多边形的内部.当 $q_i = q_{i+1}$ 时,“折线” $q_i q_{i+1}$ 实际上是一个点,在多边形的一个顶点上.

同理可证定理的“(2)”部分.

下面我们结合定理,提出基于任意多边形窗口的任意多边形裁剪算法(如果多边形包围了窗口,则直接演示窗口;如果窗口包围了多边形,则直接演示多边形.下面的算法步骤,默认窗口和多边形相交).

2 算法步骤

(1)定义任意多边形矩阵和窗口矩阵分别为 $A = \text{zeros}(2, nA)$ 和 $W = \text{zeros}(2, nW)$ (A 为2行 nA 列矩阵, W 为2行 nW 列矩阵).

定义几个备用矩阵: $NA = \text{zeros}(2, nA)$; $NW = \text{zeros}(2, nW)$; $AW = \text{zeros}(4, M)$; $\%M$ 是整形变量; $WW = \text{zeros}(1, 1)$; $AA = \text{zeros}(1, 1)$ (这里,矩阵初始化为一行一列,但在编程的过程中,可随意增加行数或列数).

(2)计算各边的法向量.

1)求出多边形 A 各边的法向量,使其指向相应边的“外侧”,并将这些法向量按其对应边的列

号顺序存入矩阵 NA 中;

2) 求出窗口 W 各边的法向量, 使其指向相应边的“外侧”, 并将这些法向量按其对应边的列号顺序存入矩阵 NW 中.

(3) 将多边形 A 和窗口 W 求交, 且将交点的“数据”存入矩阵 AW ($AW = \text{zeros}(2, nW)$), 其中 M 是交点个数, 而各行所接收的数据依次是: 第一行, 交点所在窗口边的序号 i ; 第二行, 交点在窗口第 i 条边的参数 twi ; 第三行, 交点所在多边形边的序号 j ; 第四行, 交点在多边形第 j 条边上的参数 taj .

(4) 将交点按 $i + twi$ 升序排序, 结果仍存入矩阵 AW 中, 并且将第二行的 twi 改成 $i + twi$.

(5) 在矩阵 AW 中, 顺序搜索交点, 应用定理, 确定交点对, 使得两交点之间的属于窗口的线段或折线含在多边形的内部, 同时将这条线段或折线存入矩阵 WW 中 (该矩阵的一行表示一条线段或折线, 相应第一列和最后一列表示线段或折线端点 (即交点) 参数, 其余列表示该折线的顶点参数).

(6) 将交点按 $j + taj$ 升序排序, 结果仍存入矩阵 AW 中, 并且将第四行的 taj 改成 $j + taj$.

(7) 在矩阵 AW 中, 顺序搜索交点, 应用定理确定交点对, 使得两交点之间的属于多边形的线段或者折线含在窗口的内部, 同时将这条线段或折线存入矩阵 AA 中 (该矩阵的一行表示一条线段或折线, 同一行的第三列和最后一列表示线段或折线端点 (即交点) 参数, 而第一列和第二列表示该交点对在矩阵 AW 中第二行的数据, 其余列表示该折线的顶点参数).

(8) 输出裁剪结果 (从矩阵 WW 和矩阵 AA 中匹配折线, 以构成闭合的多边形).

1) 若矩阵 WW 或 AA 为空, 则跳到第 9 步;

2) 从矩阵 WW 中输出第一行 (一条折线); $ts = WW(1,1)$; $te = WW(1, n11)$; 设第一行有 $n11$ 列, 此处是保留这条折线两个端点的 // 参数. 删除矩阵 WW 的第一行;

3) $\text{for}(j = 1; j \leq \text{size}(AA, 1); j++)$, 在矩阵 AA (有 $\text{size}(AA, 1)$ 行) 中, 逐行搜索:

{if ($te == AA(j, 1) || te == AA(j, 2)$)

{输出折线 AA(j,:);

这条折线的一端联上了矩阵 WW 中的一条折线, 结合点的参数为 te

if $te == AA(j, 1)$

$te = AA(j, 2)$;

else

$te = AA(j, 1)$;

保留折线 AA(j,:) 的另一端点, 以便在矩阵 WW 中匹配下一条折线; 删除矩阵 AA 的第 j 行;

break;

}

}

4) 如果 $te == ts$, 则说明输出了一个闭合的多边形, 可跳回第 (1) 步, 寻找下一个闭合多边形.

5) $\text{for}(i = 1; i \leq \text{size}(WW, 1); i++)$, 在矩阵 WW (有 $\text{size}(WW, 1)$ 行) 中, 逐行搜索.

{if ($te == WW(i, 1) || te == WW(i, ni)$)

设第 i 行有 ni 列.

{输出折线 WW(i,:);

if $te == WW(i, 1)$

$te = WW(i, ni)$;

else

$te = WW(i, 1)$;

删除矩阵 WW 的第 i 行;

break;

}

}

6) 跳到第 3) 步继续执行.

(9) 程序结束.

3 算法的实现

笔者用 Matlab 6.5 语言编程, 在微型机上实现了本文算法. 窗口分别为矩形, 一般多边形 (凸、凹多边形) 等情形. 图 5 和图 6 就是算法实现的结果.

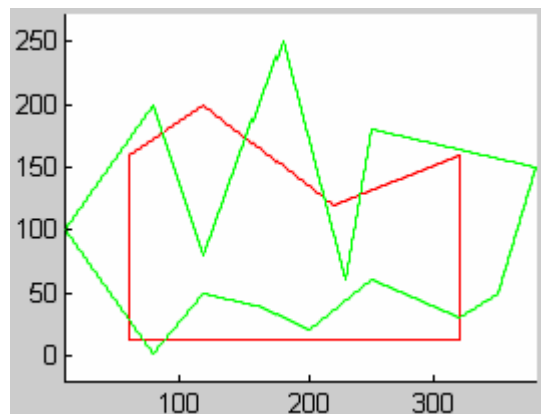


图 5 窗口 (深色线构成) 和多边形 (浅色线构成)

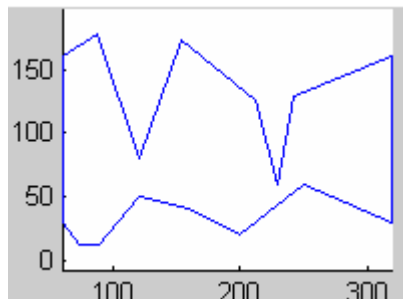


图 6 裁剪后的结果图

4 算法复杂度

设多边形和窗口的矩阵维数分别为 $2 \times nA$, $2 \times nW$. 记 $n = \max\{nA, nW\}$, 于是, 就最坏的情况而言, 计算量是:

(1) 求多边形和窗口各边法线所需的乘法次数为 $2 \times nA + 2 \times nW \leq 4n$;

(2) 求交所用乘法次数为 $2 \times 4 \times (nW \times nA) \leq 8n^2$; 同时需要作的除法次数为 $2 \times (nW \times nA) \leq 2n^2$; 所以, 算法的时间复杂度为 $O(n^2)$.

另外, 算法需要两个矩阵 ($2 \times nA$ 或 $2 \times nW$) 分别存储多边形和窗口, 同时还需要另一个矩阵 ($4 \times (nA \times nW)$) 来存储求交得到的交点, 故空间复杂度为 $O(n^2)$.

若窗口是一个矩形, 而待裁剪的多边形是任意多边形, 这时求交就简单得多. 对于窗口的一条边和多边形的一条求交, 仅需要 2 次除法 (不需要乘法), 故窗口与多边形求交时做除法的次数是 $(2n) \times 4$ (n 是多边形的边数), 所以时间复杂度为 $O(n)$.

5 结论

本文提出的算法是有效的, 对于任意窗口和任意多边形, 裁剪的结果都是正确的. 另外, 在求交之后, 确定有效线段或有效折线的方法严密、正确, 并且其过程不依赖“连贯性”, 从而求交及确定有效线段或有效折线的计算都可“并行计算”. 同时,

简单且有效的数据结构, 使得计算量大为减少. 缺点是为了追求算法简洁, 造成了部分数据冗余.

参考文献:

- [1] 赫恩, 贝克, 著. 计算机图形学 (C 语言版) [M]. 北京: 清华大学出版社, 2004.
- [2] 何阵棋. 基于编码与分类技术的任意多边形裁剪新算法[J]. 计算机工程与应用, 2002 (5): 56-59.
- [3] 彭欢. 基于端点与交点编码的矩形窗口多边形裁剪新算法[J]. 工程图学学报, 2006 (4): 72-76.
- [4] 杨哲海. 多边形裁剪算法的完善[J]. 测绘学院学报, 2003 (1): 43-46.
- [5] 温俊文. 对平面图形进行内外裁剪的统一算法[J]. 工程图学学报, 2001 (3): 71-74.
- [6] 陆毅, 孙万民. 多边形裁剪的特殊处理与实现[J]. 海洋测绘, 1999 (3): 36-40.
- [7] 刘勇奎. 一个有效的多边形裁剪算法[J]. 软件学报, 2003 (4): 845-856.
- [8] 韩俊卿. 多边形窗口的矢量图形裁剪算法[J]. 太原理工大学学报, 2005 (3): 160-163.
- [9] 付迎春, 袁修孝. 一种有效的任意多边形裁剪算法[J]. 计算机工程, 2006 (4): 278-280.
- [10] Bala R. Vatti. A generic solution to polygon clipping[J]. *Communications of the ACM*, 1992, 35 (7): 57-63.
- [11] Gunther Greiner and Kai Hormann. Efficient clipping of Arbitrary Polygons[J]. *Acm Transactions on Graphics*, 17(2): 71-83.
- [12] 孙春娟, 王文成, 李静. 基于凸片段分解的多边形窗口线裁剪算法[J]. 计算机辅助设计与图形学学报, 2006, 18 (12): 1799-1805.
- [13] 刘勇奎. 图形裁剪算法研究[J]. 计算机工程与应用, 2005 (5): 18-23.
- [14] 李静. 基于凸剖分的多边形窗口线裁剪算法[J]. 计算机辅助设计与图形学学报, 2007, 19 (4): 425-429.

[责任编辑: 李立]