VERTICAL SCAN-CONVERSION FOR FILLING PURPOSES

R.D. Hersch

Swiss Federal Institute of Technology, Lausanne

ABSTRACT

Conventional scan-conversion algorithms were developed independently of filling algorithms. They cause many problems, when used for filling purposes. However, today´s raster printers and plotters require extended use of filling, especially for the generation of typographic characters and graphic line art. A new scan-conversion algorithm, called *vertical scan-conversion* has been specifically designed to meet the requirements of parity scan line fill algorithms. Vertical scan-conversion ensures the selection of exactly one pixel per intersecting scan line between a local minimum and a local maximum of the shape outline. Pairs of selected pixels define horizontal spans. All horizontal spans contain the full set of pixels interior to the original shape. Vertical scan-conversion greatly simplifies traditional edge-tracking filling algorithms, such as ordered edge fill, flag fill and descriptive contour fill, removing the need for testing and processing special cases.

Keywords

raster graphics, scan-conversion, filling

1. Introduction

The world of Computer Graphics is synthetic and artificial. Definitions and algorithms should help to represent and display selected real world items in the best possible way. Binary raster printing devices require the selection of configurations of discrete pixels to represent analog shapes like typographic characters or graphic line art.

Scan-conversion and filling algorithms are the basic tools used for the generation of any two-dimensional shape. Despite the fact that some algorithms are well known [BRESENHAM65], only few computer graphics specialists have real experience with them. These algorithms are generally implemented in firmware and are available to users as library calls. Recently, several researchers have pointed to limitations and side effects of existing scan-conversion algorithms [BRESENHAM87], [PITTEWAY85], [HERSCH87].

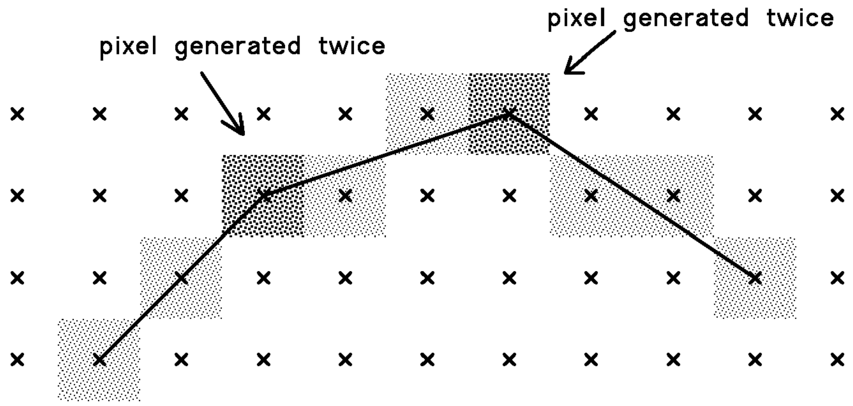
The various algorithms for scan-conversion and filling are the result of independent research. Therefore, traditional scan-conversion [BRESENHAM65] is not fully compatible with the requirements of most filling algorithms. Filling algorithms like ordered edge fill [NEWMAN79], flag fill [ACKLAND81] or descriptive contour fill [HERSCH86] require the detection and separate processing of special cases.

The purpose of this paper is to provide the basis for a new scan-conversion algorithm suiting the needs of edge tracking parity fill algorithms like ordered edge fill, flag fill and descriptive contour fill.

Proceedings CGInternational 1988, (Thalmann, Ed.), Springer Verlag, 1988, 318-327

2. Lack of coherence of previous filling algorithms, due to inadequate scan-conversion techniques

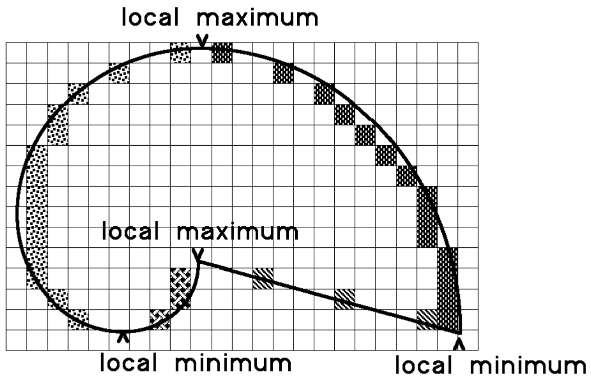
Conventional straight line segment, circular and conic arc scan-conversion is defined for integer departure and arrival points. Corresponding departure and arrival pixels belong to the scan-converted segment. Scan conversion of several connected segments leads to a duplication of pixels at junctions between segments (Fig. 1).



vscan1

Fig. 1 Traditional scan-conversion of connected segments

Ordered edge lists as well as edge-flag fill require that scan-conversion of contour segments lying between a local minimum and a local maximum (walls) should generate exactly one pixel on each intersected scan line (Fig. 2). This requirement is derived from the basic topological property stating that a straight line intersects a closed shape in an even number of points. Integer scan-conversion of connected segments does not meet this requirement. Therefore, implementors of filling algorithms have to analyze junctions between segments and rasterize them as two pixels or one pixel, after establishing whether the junction point is a local extrema or not. Furthermore, horizontal segments have to be detected and discarded [ROGERS85].



vscan2

Fig. 2 Scan conversion of contour parts (walls) lying between local minimas and local maximas

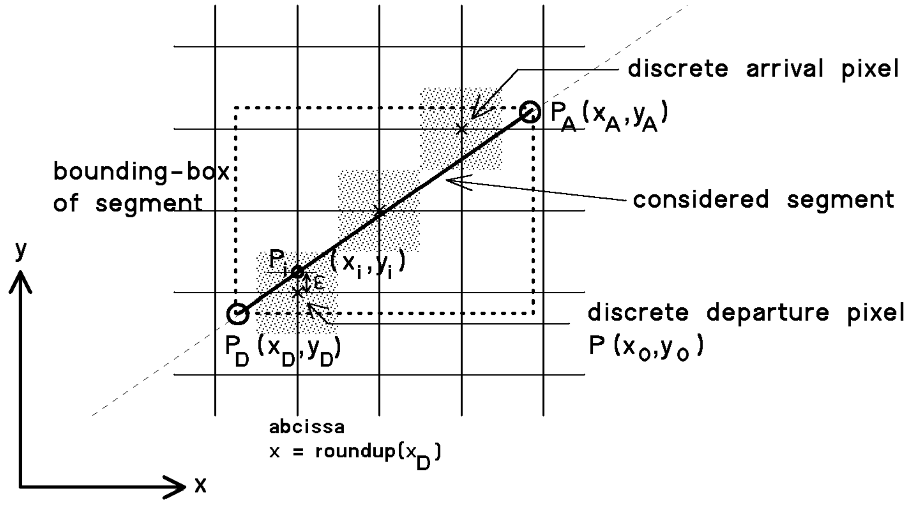
The problem of selecting exactly one pixel at intersections between contour parts (walls) and scan lines becomes even more intricate for the scan-conversion of cubic parametric splines. In such cases, recursive subdivision or forward differencing techniques are generally used [BARTELS87]. Recursive subdivision and forward differencing lead to a series of closely spaced points having real-number coordinates.

To obtain one simple discrete pixel on each intersection between walls and scan lines, heuristic methods are generally used [NEWMAN79]. Contour intersections with ordinates through pixel centers are calculated and coverage percentages of pixel surfaces are evaluated.

These coverage evaluation procedures are cumbersome and inefficient. There is a need for a scan-conversion algorithm able to scan-convert polysegments with real vertices and to produce exactly one pixel per scanline intersection.

3. First octant real-number scan-conversion

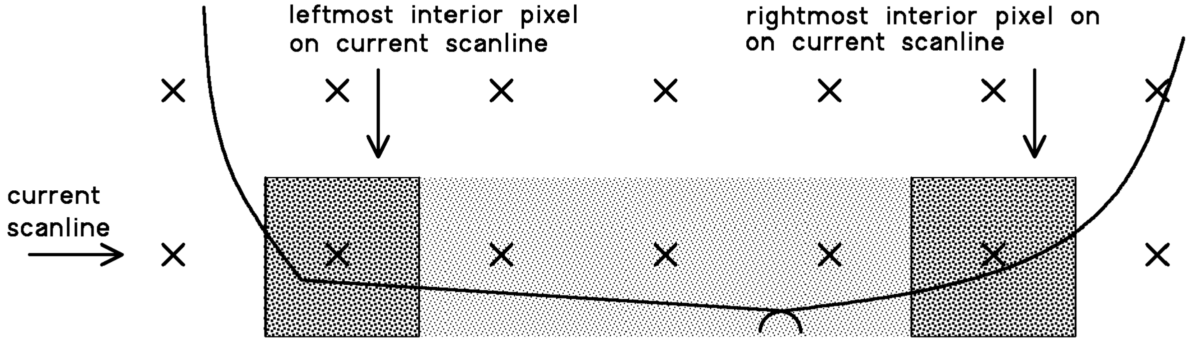
A previous approach [HERSCH87] has demonstrated the importance of *fixed-point real number scan-conversion* for the scan-conversion of straight line, circular arcs and spline segments. Real number scan-conversion can be used for filling, although it was not developed primarly for this purpose. Real number scan-conversion generates those pixels with pixel centers closest to the original continuous outline (Fig. 3). Selected pixel centers lie within the original segment´s bounding box. This algorithm is called *first octant real-number scan conversion*, since it is defined for straight line and circular arc segments in the first octant. Scan-conversion of segments in other octants is obtained by applying interoctant transformations (concatenation of symmetries along x, y or y=x axis).



vscan5

Fig. 3 First octant real number scan conversion

It has also been shown [HERSCH87] that first octant real-number scan-conversion does not ensure 8-point connectivity at the junction between segments having different primary directions. Furthermore, first octant scan-conversion generates many more pixels than necessary. Among all pixels lying on one scanline, only the leftmost or rightmost pixels are kept (Fig. 4) for later filling.



vscan7

Fig. 4 Keeping leftmost and rightmost pixels

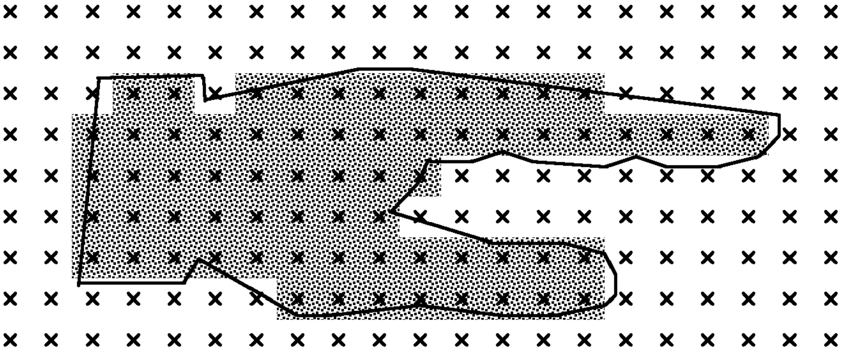
The aim of the new *vertical scan-conversion* algorithm is to generate directly, without ambiguity, leftmost or rightmost interior pixels.

4. Requirements for a new vertical real-number scan-conversion algorithm

The definition of a new vertical scan-conversion algorithm is directly derived from the requirements of scanline parity fill algorithms. These requirements can be expressed with the help of the following definitions.

Definition 1

The set of discrete pixels representing a closed continuous shape contains all pixels, whose pixel centers lie within the shape boundary (Fig. 5)



vscan8

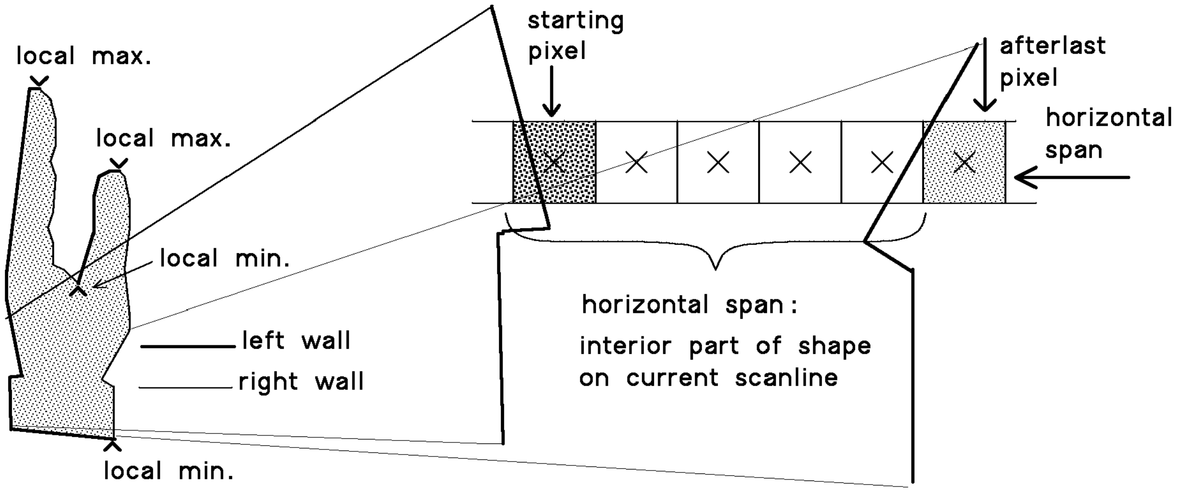
Fig. 5 Closed shape rendered by its interior pixels

Definition 2

The part of a closed contour outline lying between a local minimum and a local maximum in scanning direction is called a *wall*. A *left wall* is part of the left border of a closed shape. A *right wall* is part of the right border of a closed shape (Fig. 6).

Definition 3

A horizontal span is defined as the set of pixels lying on one scanline and whose pixel centers lie between a left wall and the following right wall (Fig. 6). A horizontal span is given by the coordinates of its starting pixel and by the coordinates of the pixel following its last pixel (afterlast pixel). A span with identical coordinates for starting and afterlast pixel is a null-span containing an empty set of pixels.



vscan9

Fig. 6 Closed shape with left and right walls

A scan line has an even number of intersections with a closed shape. Intersections with left walls and right walls provide horizontal spans. The set of all spans contains all pixels interior to the shape outline.

These statements enable us to derive the basic requirement for scan-conversion:

Requirement : Scan-conversion of a wall should lead to the selection of exactly one pixel on each scan line. The closest pixel to the wall, lying on its right side, should be selected.

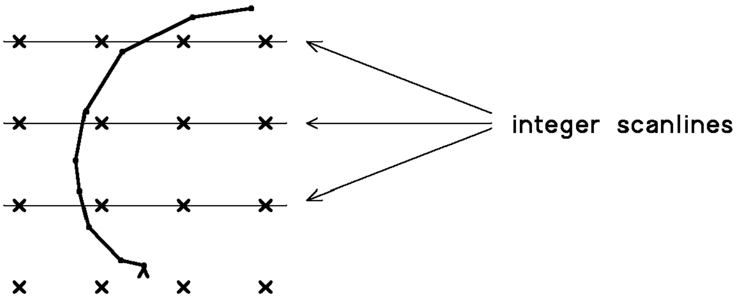
Scan-conversion of left walls generates starting pixels and scan-conversion of right walls generates afterlast pixels of horizontal spans. Filling a closed shape consists of marking all pixels belonging to all horizontal spans obtained by scan-conversion of left and right walls.

Each wall is given, after subdivision or forward differencing of its original spline description, by one polysegment containing many closely spaced vertices having real coordinates.

To fulfill the previously described requirement for *wall scan-conversion*, we should be able to scan-convert polysegments without taking into account special orientations or sizes of individual segments.

5. Vertical real-number scan-conversion

Let us analyze the problems we face when scan-converting a polysegment representing a shape wall (Fig. 7).



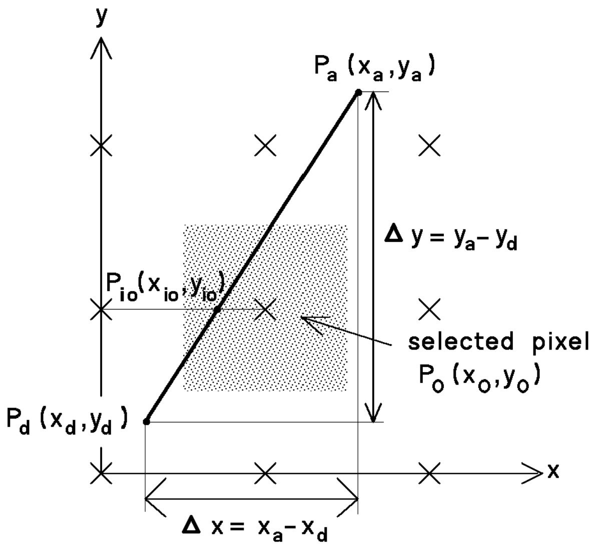
vscan10

Fig. 7 Continuous polysegment lying on top of a discrete pixel grid

Only segments crossing integer scan lines contribute to the determination of the closest pixels lying to the right of the continous polysegment. Thus, the new vertical real-number scan-conversion algorithm can be formulated in the following way :

```
FOR CurrentSegment:=FirstSegment TO LastSegment OF PolySegment DO
     IF  IntersectScanline(CurrentSegment) THEN
         VerticalScanConvert(CurrentSegment)
     ENDIF
ENDFOR
```

Vertical scan-conversion of segments intersecting scan lines is based on the computation of the intersection of the segment with the lowest intersecting scanline and on incremental calculation of intersections with the next scanlines (Fig. 8). Pixels to the right of scanline-segment intersections are chosen as discrete pixels defining either starting or afterlast pixels of spans.



vscan11

Fig. 8 Vertical scan-conversion of segment P_dP_a

In mathematical terms :

Intersection point $P_{i0}(x_{i0}, y_{i0})$ with scanline above real departure point:

$$y_{i0} = \text{roundup}(y_d)$$

$$x_{i0} = x_d + \frac{\Delta x}{\Delta y}(y_{i0} - y_d)$$

Selected first discrete pixel $P_0(x_0, y_0)$:

$$y_0 = y_{i0}$$

$$x_0 = \text{roundup}(x_{i0})$$

Recurrent formulation for the next real intersection points P_{ik+1}

$$y_{ik+1} = y_{ik} + 1$$

$$x_{ik+1} = x_{ik} + \frac{\Delta x}{\Delta y}$$

and for the selection of pixel P_{K+1}

$$y_{k+1} = y_{ik+1}$$

$$x_{k+1} = \text{roundup}\ (x_{ik+1});$$

The number of pixels to be scan-converted :

$$\text{nbpixel} = \text{trunc}(y_a) - \text{roundup}(y_d) + 1$$

This scan-conversion algorithm guarantees the selection of one discrete pixel on each intersection between walls (polysegments) and integer scanlines.

Let us analyze the behaviour of vertical scan-conversion in the cases which create problems with conventional scan-conversion.

6. Behaviour of vertical real-number scan conversion

When conventional scan-conversion algorithms are used for filling, one needs to recognize and appropriately process the following situations:

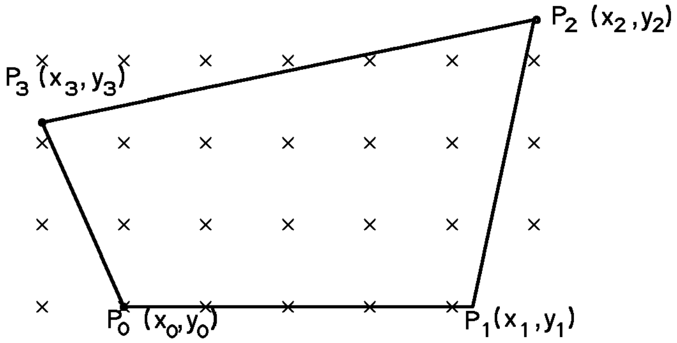
a) Existence, within a wall, of a long horizontal polysegment going through pixel centers

b) Local maximum lying on a pixel center

c) Vertex point of a polysegment wall lying exactly on a pixel center

d) Partly degenerated shape

Case a: vertical scan-conversion of horizontal segment P_0P_1



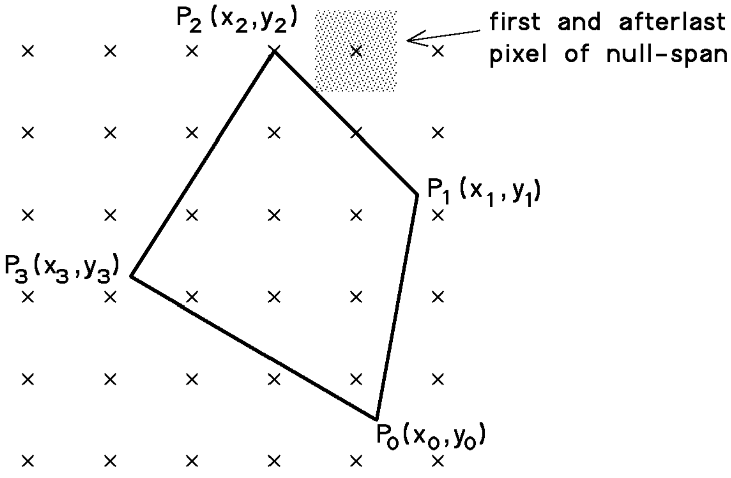
vscan12

Fig. 9 Long horizontal segment through pixel centers

$$\text{nbpixel}(P_0P_1) = \text{trunc}(y_1) - \text{roundup}(y_0) + 1 = 0$$

Line segment P_0P_1 does not generate discrete pixels. Horizontal segments need not to be treated as special cases.

Case b: local maximum on a pixel center



vscan13

Fig. 10 Vertex at local maximum lying on pixel center

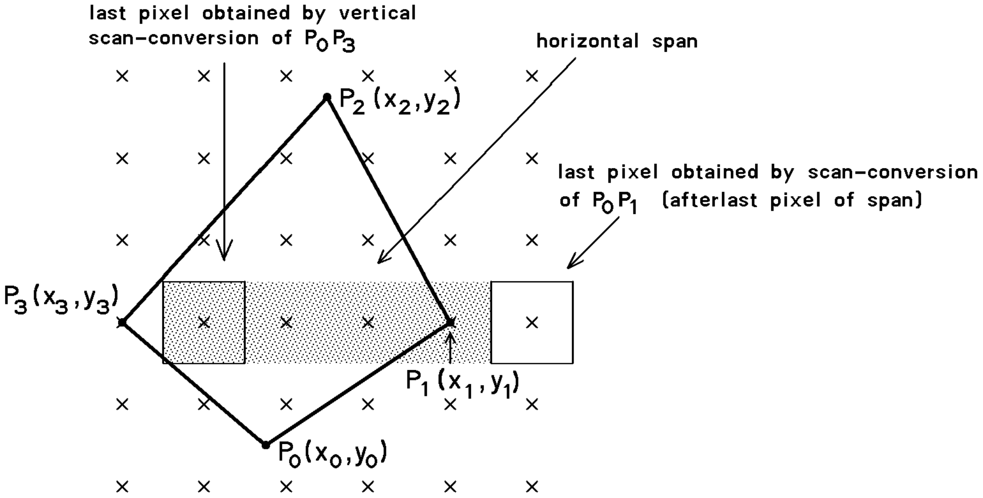
Vertical scan-conversion of segment P_3P_2 :

$$\text{nbpixel}(P_3P_2) = \text{trunc}(y_2) - \text{roundup}(y_3) + 1 = y_2 - \text{trunc}(y_3+1) + 1$$

P_3P_2 intersects scanline $y=y_2$ at (x_2, y_2).

The selected discrete pixel on scanline $y=y_2$ is (x_2+1, y_2). The same pixel is selected by vertical scan-conversion of segment P_1P_2. Therefore, pixel (x_2+1, y_2) is both the first and the afterlast pixel of a horizontal null-span.

Case c: intermediate vertex points on pixel centers



last pixel obtained by vertical scan-conversion of P_0P_3
horizontal span
$P_2\,(x_2,y_2)$
last pixel obtained by scan-conversion of P_0P_1 (afterlast pixel of span)
$P_3\,(x_3,y_3)$
$P_1\,(x_1,y_1)$
$P_0(x_0,y_0)$

vscan14

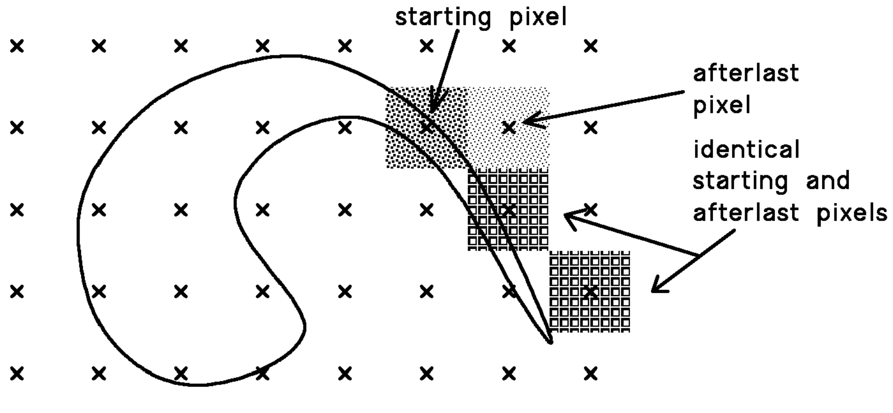
Fig. 11 Polysegment vertices P_3 and P_1 lying on pixel centers

Vertical scan-conversion of P_0P_3 leads to the selection of pixel $(x_3+1,\ y_3)$. The first pixel obtained by scan-conversion of P_3P_2 lies on scanline y_3+1. Therefore, pixel (x_3+1, y_3) is the only pixel selected on scanline y_3 by scan-conversion of wall $P_0P_3P_2$.

Pixel (x_3+1, y_3) is the first pixel of a horizontal span. Pixel $(x_1+1,\ y_1)$ is the afterlast pixel of a horizontal span. If $y_3=y_1$, the span contains the set of pixels $\{(x_3+1, y_3), (x_3+2, y_3), \ldots, (x_1, y_1)\}$.

Due to the mathematical definitions of vertical scan conversion and horizontal spans, pixels having non-extremal polysegment vertices on their pixel centers belong to the interior of the shape if they belong to a right wall. Pixels with non-extremal vertices on their pixel centers obtained by scan-conversion of a left wall do not belong to the interior pixel set.

Case d: partly degenerated shape



vscan15

Fig. 12 Null-spans given by identical starting and afterlast pixels

The definition of *starting* and *afterlast* pixels as being the first pixels lying on the right respectively of a left or of a right wall allows null-segments to be generated when there are no discrete pixel centers between a left and a right contour part (Fig. 12). Vertical scan-conversion therefore provides a basis for generating shapes scaled down by any factor.

7. Conclusion

In the synthetic world of computer graphics, algorithms are based on appropriate definitions. Filling algorithms are based on scan-conversion of shape outlines. Restrictive definition of scan-conversion leads to filling algorithms requiring tests to be carried out for many special cases.

A new algorithm called *vertical scan-conversion* has been specially developed to meet the requirements of shape filling. It is very useful for the generation of figures described by spline outlines. During scan-conversion, splines are subdivided into polysegments having real vertices. Vertical scan-conversion generates exactly one pixel for each intersection between scanlines and the polysegment. The generated pixel represents either the first pixel or the pixel following the last pixel of a horizontal span. Vertical scan-conversion of contour polysegments leads to an even number of selected pixels on each scanline. Conventional filling algorithms (ordered edge fill, flag fill, descriptive contour fill) can be used in a simplified form, removing the need to test special cases.

Vertical scan-conversion does not suit the requirements of seed or tint fill algorithms [ROGERS85], due to the fact that vertically scan-converted contour walls (Fig. 2) do not form 8-connected discrete shape boundaries.

The future of 2-d imaging devices lies in more general algorithms which cover all possible cases without exceptions. Only such algorithms can be efficiently integrated in sillicium to produce very fast raster drawing devices.

References:

[ACKLAND81] B.D. Ackland, N.H. Weste, "The Edge Flag Algorithm - A Fill Method for Raster Scan Displays," *IEEE Trans. on Computers,* Vol 30, No 1, January 1981, pp. 41-48

[BARTELS87] R.H. Bartels, J.C. Beatty, B.A. Barsky, *An Introduction to the Use fo Splines in Computer Graphics*, Morgan Kaufmann Publ., 1987

[BRESENHAM65] J.E. Bresenham, "Algorithm for computer control of a digital plotter," *IBM Systems Journal,* Vol 4, No 1, 1965, pp 25-30

[BRESENHAM87] J.E. Bresenham, "Ambiguities in Incremental Line Rastering," *IEEE Computer Graphics and Applications*, Vol 7, No 5, May 1987, pp. 31-43

[HERSCH86] R.D. Hersch, "Descriptive Contour Fill of Partly Degenerated Shapes," *IEEE Computer Graphics and Applications,* Vol 6, No 7, July 1986, pp.61-70

[HERSCH87] R.D. Hersch, "Real Scan-Conversion of Shape Contours," *Proceedings Computer Graphics International 87,* Karuizawa, Japan, Ed. T.L. Kunii, Springer Verlag, 1987, pp. 207-220

[NEWMAN79] W.M. Newman, R.F. Sproull, *Principles of Interactive Computer Graphics,* McGraw-Hill, 1979

[PITTEWAY85] M.L.V. Pitteway, "Algorithms for Conic Generation", *Fundamental Algorithms for Computer Graphics*, (Ed. R. Earnshaw), Nato Asi Series, Vol. F17, Springer Verlag Berlin, 1985, pp.219-237

[ROGERS85] D.F. Rogers, *Procedural Elements for Computer Graphics,* McGraw-Hill, 1985

Appendix

Shapes described by cubic splines, subdivided and generated with vertical scan-conversion

(resolution: 150 dots/inch)