



[12] 发明专利说明书

[21] ZL 专利号 93117077.X

[51]Int.Cl⁶

G06K 9/00

[45]授权公告日 1996年10月23日

[24]颁证日 96.7.19

[21]申请号 93117077.X

[22]申请日 93.8.30

[30]优先权

[32]92.9.1 [33]JP[31]259137/92

[32]92.9.11 [33]JP[31]269646/92

[73]专利权人 寅市和男

地址 日本狭山市

[72]发明人 寅市和男

[74]专利代理机构 中国国际贸易促进委员会专利商
标事务所

G06K 11/00

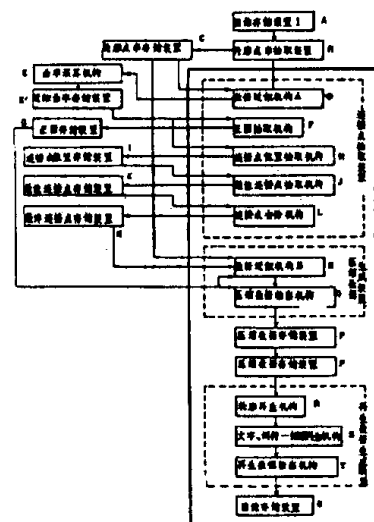
代理人 王以平

权利要求书 19 页 说明书 74 页 附图页数 52 页

[54]发明名称 文字数据、词符-插图数据的输入输出装置及其方法

[57]摘要

本发明能用很短时间把文字字体、词符-插图变换为少量数据进行存储，并能以任意尺寸、在任意的位置再生。其构成是以光学读取文字字体、词符-插图，求轮廓点串，利用自变量 t 存储轮廓点串的坐标，求轮廓点串的连接点，利用直线、圆弧、自由曲线近似连接点之间的区间，存储近似的参数的连接点坐标。输出联结连接点的坐标和相邻连接点的线段的参数，由此再生轮廓点串，将轮廓点串所围区域同其它区域进行区别后，再生文字、词符-插图。



权 利 要 求 书

1. 一种文字数据输入输出装置,其特征在于包括:

文字读取装置,用于光学读取文字字体数据,使这些数据与有限个纵横排列的象素对应并进行存储;

轮廓线抽取机构,用于抽取与从横排列象素对应的已读取的文字的轮廓线;

轮廓点串存储机构,用于对每个连续的群,把抽取出的轮廓线的2维坐标(X, Y)进行存储;

数据近似机构A,其用途是,设自变量为 t ,因变量为 x, y ,使用以 t 为自变量, x, y 为因变量的2次分段多项式近似上述各群中轮廓点串的 x, y 坐标,反复进行最小二乘法近似直到近似精度达到指定的范围,并求出各轮廓点串群的近似多项式;

曲率运算机构,用于根据上述近似结果,求出在 x, y 空间中各群的点串的每一点的曲率;

正圆抽取机构;用于根据各群的曲率数据抽取正圆;

临时连接点位置抽取机构,用于根据点串的曲率数据抽取空间不可微点作为临时连接点;

最佳连接点抽取机构,用于把位于临时连接点近旁的其它候补连接点根据相关关系概率地实施曲线拟合,求出最佳连接点;

不必要的连接点除去机构,用于从最佳连接点中找出并除去那些即使去掉也可以保证近似精度的不必要连接点;

数据近似机构 B ,其用途是当按直线、圆弧的顺序近似同一点串群内的相邻连接点之间不能得到指定的精度时,就用以 t 为自变量, x 、 y 为因变量的 2 次分段多项式进行近似,并在增加 2 次分段多项式因次数的同时,反复进行最小二乘近似,直到近似精度达到指定的数值后,再利用直线、圆弧、分段多项式近似相邻连接点之间的连线。

压缩数据存储机构,用于对每个点串群存储近似上述连接点的坐标和相邻连接点之间的函数的参数;

轮廓再生机构,用于输入存储的压缩数据,得到近似每个点串群的连接点的坐标和相邻连接点函数的参数,再生出轮廓线;

文字再生机构,用于使再生的轮廓线内的象素和外部象素对应不同的值;

再生数据输出机构,用于把再生的文字字体作为文字输出。

2. 一种文字数据输入输出装置,其特征在于包括:

文字读取装置,用于光学读取文字字体数据,使这些数据与有限个纵横排列的象素对应,并进行存储;

轮廓线抽取机构,用于抽取与纵横排列象素对应的已读取的文字的轮廓线;

轮廓点串存储机构,用于对每个连续的群把抽取出的轮廓线

的 2 维坐标(x, y)进行存储;

数据近似机构, 用于用以 t 为自变量, x, y 为因变量的 2 次分段多项式近似上述各群的轮廓点串的 x, y 坐标, 反复进行最小二乘法近似直到近似精度达到指定的范围, 并求出各轮廓点串群的近似多项式;

曲率运算机构, 用于根据上述近似结果, 求出在 x, y 空间的各群内点串的每一点的曲率;

正圆抽取机构, 用于根据各群的曲率数据抽取出正圆;

临时连接点位置抽取机构, 用于根据点串的曲率数据抽取出空间不可微点作为临时连接点;

最佳连接点抽取机构, 用于把位于临时连接点近旁的其它候补连接点根据相关关系概率地实施曲线拟合, 求出最佳连接点;

数据近似机构, 用于当按直线、圆弧的顺次近似同一点串群的相邻连接点之间不能得到指定的近似精度时, 就用以 t 为自变量, x, y 为因变量的 2 次分段多项式进行近似, 并在增加 2 次分段多项式的因次的同时, 反复进行最小二乘近似直到近似精度达到指定的数值后, 再利用直线、圆弧、分段多项式近似相邻连接点之间的连线;

压缩数据存储机构, 用于对每个点串群存储近似上述连接点的坐标和相邻连接点之间的函数的参数;

轮廓再生机构, 用于输入存储的压缩数据, 得到近似每个点串

群的连接点的坐标和相邻连接点的函数的参数,并再行出轮廓线;

文字再生机构,用于使再生的轮廓线内的象素和外部象素对应不同的值;

再生数据输出机构,用于把再生的文字字体作为文字输出。

3. 一种文字数据输入输出装置,其特征在于包括:

文字读取装置,用于光学读取文字字体数据,使这些数据与有限个纵横排列的象素对应并进行存储;

轮廓线抽取机构,用于抽取与纵横排列象素对应的已读取的文字的轮廓线;

轮廓点串存储机构,用于对每个连续的群把抽取出的轮廓线的2维坐标(x, y)进行存储;

数据近似机构,用于用以 t 为自变量, x, y 为因变量的分段多项式近似上述各群的轮廓线点串的 x, y 坐标,求出各轮廓线点串的近似多项式;

曲率运算机构,用于根据上述近似结果,求出在 x, y 空间的各群的点串的每一点的曲率;

正圆抽取机构,用于根据各群的曲率数据抽取正圆;

连接点位置抽取机构,用于根据点串的曲率数据抽出空间不可微点作为连接点;

数据近似机构,用于当按直线、圆弧的顺序近似同一点串群内的相邻连接点之间不能得到指定的近似精度时,就用以 t 为自变

量, x 、 y 为因变量的分段多项式进行近似, 从而利用直线、圆弧、分段多项式近似相邻连接点之间的连线;

压缩数据存储机构, 用于对每个点串群存储近似上述连接点坐标和相邻连接点之间的函数的参数;

轮廓再生机构, 用于输入存储的压缩数据, 得到近似各轮廓点串群的连接点的坐标和相邻连接点的函数的参数, 并再生出轮廓线;

文字再生机构, 用于使再生轮廓线内的象素和外部象素对应不同的值;

再生数据输出机构, 用于把再生的文字字体作为文字输出。

4. 一种文字数据输入输出装置, 其特征在于包括:

文字读取装置, 用于光学读取文字字体数据, 使这些数据与有限个纵横排列的象素对应并进行存储;

轮廓线抽取机构, 用于抽取与纵横排列象素对应的已读取的文字的轮廓线;

轮廓点串存储机构, 用于对每个连续的群把抽取出的轮廓线的 2 维坐标(X 、 Y)以 t 为自变量, X 、 Y 为因变量进行存储;

曲率运算机构, 用于求出 x 、 y 空间的各群内点串的每一点的离散曲率;

正圆抽取机构, 用于根据各群的曲率数据抽取正圆;

连接点位置抽取机构, 用于根据点串的曲率数据抽出空间不

可策点作为连接点；

数据近似机构,用于当按直线、圆弧的顺序近似同一点串群的相邻连接点之间不能得到指定的近似精度时,就用以 t 为自变量, x 、 y 为因变量的分段多项式进行近似,并在增加分段多项式的因次数的同时,反复进行最小二乘近似,直到近似精度达到指定的数值,再利用直线、圆弧、分段多项式近似相邻连接之间的连线;

压缩数据存储机构,用于对每个点串群存储近似上述连接点的坐标和相邻连接点之间的函数的参数;

轮廓再生机构,用于输入存储的压缩数据,得到近似各点串群的连接点的坐标和相邻连接点的函数的参数,并再生出轮廓线;

文字再生机构,用于使再生的轮廓线内的象素和外部的象素对应不同的值;

再生数据输出机构,用于把再生的文字字体作为文字输出。

5. 一种文字数据的输入输出方法,其特征在于:

光学读取文字字体数据,使这些数据与有限个纵横排列的象素对应,并存储这些数据;抽取出与纵横排列象素对应的已读取的文字的轮廓线,对每个连续的群,以 t 为自变量,以 X 、 Y 为因变量存储已抽取出的轮廓线的2维坐标 (X, Y) ;设自变量为 t ,因变量为 x, y ,用以 t 为自变量, x, y 为因变量的分段多项式近似上述各群的轮廓点串的 x, y 坐标;根据上述近似结果;求 x, y 空间的各

群中点串的每一点的曲率;根据各群的曲率数据抽取正圆;根据点串的曲率数据抽取出空间不可微点作为连接点;当按直线、圆弧的顺序近似同一点串群内的相邻连接点之间不能得到指定的近似精度时,用以 t 为自变量, x 、 y 为因变量的分段多项式近似;通过对各点串群存储近似上述连接点的坐标和相邻连接点之间的函数的参数,输入文字字体数据;而在输出文字时对各点串群输出存储的近似连接点坐标和相邻连接点之间的函数的参数;然后再生文字字体的轮廓线;通过使再生的轮廓线内的象素和外部象素对应不同的值,再生出文字,最后把再生的文字字体作为文字输出。

6. 一种文字数据输入输出方法,其特征在于:

光学读取文字字体数据,使这些数据与有限个纵横排列的象素对应并存储这些数据;抽取与纵横排列象素对应的已读取的文字的轮廓线;对各连续群以 t 为自变量, X 、 Y 为因变量存储已抽取的轮廓线的2维坐标(X , Y);设自变量为 t ,因变量作为 x 、 y ,用以 t 为自变量, x 、 y 为因变量的分段多项式近似上述各群的轮廓线点串的 x 、 y 坐标;根据上述近似结果,求出 x 、 y 空间的各群中点串的每一点的曲率;根据各群的曲率数据抽取正圆;根据点串的曲率数据抽取出空间不可微点作为临时连接点;把位于临时连接点近旁的其它候补连接点根据相关关系概率地进行曲线拟合,求出最适宜的连接点;从最佳连接点中找出并除去即使没有也能保证近似精度的不必要的连接点;当按直线、圆弧的顺序近似同

一点串群内相邻连接点之间不能得到指定的近似精度时,用以 t 为自变量、 x 、 y 为因变量的分段多项式进行近似,并在增加分段多项式的因次数的同时,反复用最小二乘近似直到近似精度达到指定的数值后,再利用直线、圆弧、分段多项式近似相邻连接点之间;通过对各点串群存储近似上述连接点的坐标和相邻连接点之间的函数的参数,输入文字字体数据;另一方面,输出文字时,对各点串群输出存储的近似连接点的坐标和相邻连接点之间的函数的参数;然后再生文字字体的轮廓线;通过使再生的轮廓线内的象素和外部的象素对应不同的值,再生出文字;最后把再生的文字字体作为文字输出。

7. 一种文字数据输入输出方法,其特征在于:

光学读取文字字体数据,使这些数据与有限个纵横排列的象素对应并存储这些数据;抽取与纵横排列象素对应的已读取的文字的轮廓线;对各连续群以 t 为自变量, X 、 Y 为因变量存储已抽出的轮廓线的 2 维坐标 (X, Y) ; 求出 x 、 y 空间各群的点串的每一点的离散的曲率;根据各群的曲率数据抽取正圆;根据点串的曲率数据抽取空间不可微点作为连接点;当按直线、圆弧的顺序近似同一点串群的相邻连接点之间不能得到指定的近似精度时,用以 t 为自变量,以 x 、 y 为因变量的分段多项式进行近似,并在增加分段多项式的因次数的同时,反复进行最小二乘近似,直到近似精度达到指定的数值后,再利用直线、圆弧、分段多项式近似相邻连接

点之间的连线；通过对各点串群存储近似上述连接点的坐标和相邻连接点之间的函数的参数，输入文字字体数据；另一方面，输出文字时，对各点串群输出存储的近似连接点的坐标和相邻连接点之间的函数的参数；然后再生文字字体的轮廓线；通过使再生的轮廓线内的象素和外部象素对应不同的数值，再生出文字；最后把再生的文字字体作为文字输出。

8. 一种词符—插图数据输入输出装置，其特在于包括：

词符—插图读取装置，用于光学读取词符—插图数据，使这些数据与有限个纵横排列的象素对应并进行存储；

轮廓线抽取机构，用于抽取与纵横排列象素对应的已读取的词符—插图的轮廓线；

轮廓点串存储机构，用于对每个连续的群把抽取出的轮廓线的2维坐标(X, Y)进行存储；

数据近似机构A，用于用以 t 为自变量， x, y 为因变量的2次分段多项式近似上述各群轮廓线点串的 x, y 坐标，反复进行最小二乘近似直到近似精度达到指定的范围，并求出各轮廓点串群的近似多项式；

曲率运算机构，用于根据上述近似结果求出 x, y 空间各群的点串的每一点的曲率；

正圆抽取机构，用于根据各群的曲率数据抽取正圆；

临时连接点位置抽取机构，用于根据点串的曲率数据抽取空

间不可微点作为临时连接点；

最佳连接点抽取机构，用于把位于临时连接点近旁的其它候补连接点根据相关关系概率地进行曲线拟合，求出最佳连接点；

不必要的连接点除去机构，用于从最佳连接点中找出并除去即使没有也能保证近似精度的不必要的连接点；

数据近似机构 B ，用于当按直线、圆弧的顺序近似同一点串群的相邻连接点之间不能得到指定的近似精度时，就用以 t 为自变量， x 、 y 为因变量的 2 次分段多项式进行近似，并在增加 2 次分段多项式的因次数的同时，反复进行最小二乘近似直到近似精度达到指定的数值后，再利用直线、圆弧、分段多项式近似相邻连接点之间的连线；

压缩数据存储机构，用于对各点串群存储近似上述连接点的坐标和相邻连接点之间的函数的参数；

轮廓再生机构，用于输入存储的压缩数据，得到近似各点串群的连接点的坐标和相邻连接点之间的函数的参数，并再生出轮廓线；

词符—插图再生机构，用于使再生的轮廓线内的象素和外部象素对应不同的值；

再生数据输出机构，用于把再生的词符—插图作为词符—插图输出。

9. 一种词符—插图输入输出装置，其特征在于包括：

词符—插图读取装置,用于光学读取词符—插图数据,使这些数据与有限个纵横排列的象素对应并存储这些数据;

轮廓线抽取机构,用于抽取与纵横排列象素对应的已读取的词符—插图的轮廓线;

轮廓点串存储机构,用于存储对每个把抽取出的轮廓线的2维坐标(X 、 Y)连续起来的群;

数据近似机构,用于用以 t 为自变量,以 x 、 y 为因变量的分段多项式近似上述各群的轮廓线点串的 x 、 y 坐标,反复进行最小二乘近似直到近似精度达到指定的范围,并求出各轮廓点串群的近似多项式;

曲率运算机构,用于根据上述近似结果,求出 x 、 y 空间各群的点串的每一点的曲率;

正圆抽取机构,用于根据各群的曲率数据抽取正圆;

临时连接点位置抽取机构,用于根据点串的曲率数据抽取空间不可微点作为临时连接点;

最佳连接点抽取机构,用于把位于临时连接点近旁的其它候补连接点,根据相关关系概率地进行曲线拟合,求出最佳连接点;

数据近似机构,用于当按直线、圆弧的顺序近似同一点串群的相邻连接点之间不能得到指定的近似精度时,就用以 t 为自变量,以 x 、 y 为因变量的分段多项式进行近似,并在增加2次分段多项式的因次数的同时反复进行最小二乘近似直到近似精度达到指定的数值后,再利用直线、圆弧、分段多项式近似相邻连接点之

间的连线；

压缩数据存储机构，用于对各点串群存储近似上述连接点的坐标和相邻连接点之间的函数的参数；

轮廓再生机构，用于输入存储的压缩数据，得到近似各点串群的连接点的坐标和相邻连接点的函数的参数，并再生出轮廓线；

词符—插图再生机构，用于使再生的轮廓线内的象素和外部象素对应不同的值；

再生数据输出机构，用于把再生的词符—插图作为词符—插图输出。

10. 一种词符—插图输入输出装置，其特征在于包括：

词符—插图读取装置，用于光学读取词符—插图数据，使这此数据与有限个纵横排列的象素对应并存储这些数据；

轮廓线抽取机构，用于抽取与纵横排列的象素对应的已读取的词符—插图的轮廓线；

轮廓点串存储机构，用于存储 对每个把抽取出的轮廓线的2维坐标(X 、 Y)连续起来的群；

数据近似机构，用于用以 t 为自变量，以 x 、 y 为因变量的分段多项式近似上述各群的轮廓线点串的 x 、 y 坐标，求出各轮廓线点串群的近似多项式；

曲率运算机构，用于根据上述近似结果，求出 x 、 y 空间各群的点串的每一点的曲率；

正圆抽取机构,用于根据各群的曲率数据抽取正圆;

连接点位置抽出机构,用于根据点串的曲率数据抽取空间不可微点作为连接点;

数据近似机构,用于用直线、圆弧、分段多项式近似相邻连接点间,当按直线、圆弧的顺序近似同一点串群的相邻连接点之间不能得到指定的近似精度时,就用以 t 为自变量, x 、 y 为因变量的分段多项式进行近似;然后,再利用直线、圆弧、分段多项式近似相邻连接点之间;

压缩数据存储机构,用于对各点串群存储近似上述连接点的坐标和相邻连接点的函数的参数;

轮廓再生机构,用于输入存储的压缩数据,得到近似各点串群的连接点的坐标和相邻连接点的函数的参数,并再生出轮廓线;

词符—插图再生机构,用于使再生的轮廓线内的象素和外部象素对应不同的值;

再生数据输出机构,用于把再生的词符—插图作为词符—插图输出。

11. 一种词符—插图输入输出装置,其特征在于包括:

词符—插图读取装置,用于光学读取词符—插图数据,使这些数据与有限个纵横排列的象素对应并存储这些数据;

轮廓线抽取机构,用于抽取与纵横排列象素对应的已读取的词符—插图的轮廓线;

轮廓点串存储机构,用于以 t 为自变量,以 X, Y 为因变量存储对每个把抽取出的轮廓线的 2 维坐标 (X, Y) 连续起来的群;

曲率运算机构,用于求出 x, y 空间各群的点串的每一点的离散曲率;

正圆抽取机构,用于根据各群的曲率数据抽取正圆;

连接点位置抽取机构,用于根据点串的曲率数据抽取空间不可微点作为连接点;

数据近似机构,用于当按直线、圆弧的顺序近似同一点串群的相邻连接点之间不能得到指定的近似精度时,就用以 t 为自变量,以 x, y 为因变量的分段多项式进行近似,并在增加分段多项式的因次数的同时,反复进行最小二乘近似,直到近似精度达到指定的数值后,再利用直线、圆弧、分段多项式近似相邻连接点;

压缩数据存储机构,用于对各点串群存储近似上述连接点的坐标和相邻连接点之间的函数的参数;

轮廓再生机构,用于输入存储的压缩数据,得到近似各点串群的连接点的坐标和相邻连接点的函数的参数,并再生出轮廓线;

词符一插图再生机构,用于使再生的轮廓线内的象素和外部象素对应不同的值;

再生数据输出机构,用于把再生的词符一插图作为调符一插图输出。

12. 一种词符一插图数据输入输出方法,其特征在于:

光学读取词符—插图数据,使这些数据与有限个纵横排列的象素对应并存储这些数据;抽取与纵横排列象素对应的已读取的词符—插图的轮廓线,以 t 为自变量,以 X 、 Y 为因变量存储对每个把抽取出的轮廓线的2维坐标 (X, Y) 连续起来的群;设自变量为 t ,把因变量为 x 、 y ,用以 t 为自变量,以 x 、 y 为因变量的分段多项式近似上述各轮廓点串的 X 、 Y 坐标;根据上述近似结果,求出 x 、 y 空间各群的点串的每一点的曲率;根据各群的曲率数据抽取正圆;根据点串的曲率数据抽取空间不可微点作为连接点;当按直线、圆弧的顺序近似同一点串群的相邻连接点之间不能得到指定的近似精度时用以 t 为自变量,以 x 、 y 为因变量的分段多项式进行近似;通过对各点串群存储近似上述连接点的坐标和相邻连接点之间的函数的参数,输入词符—插图;另一方面,输出词符—插图时,对各点串群输出存储的连接点的坐标和相邻连接点之间的函数的参数,然后再生词符—插图的轮廓线;通过使再生的轮廓线内的象素和外部象素对应不同的值,再生出词符—插图;最后把再生的词符—插图作为词符—插图输出。

13. 一种词符—插图数据输入输出方法,其特征在于:

光学读取词符—插图数据,使这些数据与有限个纵横排列的象素对应,并存储这些数据;抽取与纵横排列象素对应的已读取的词符—插图的轮廓线;以 t 为自变量,以 X 、 Y 为因变量存储对每个把抽取出的轮廓线的2维坐标 (X, Y) 连续起来的群;设自变量

为 t , 把因变量为 x, y , 用以 t 为自变量, X, Y 为因变量的分段多项式近似上述各群的轮廓线点串的 X, Y 坐标; 根据上述近似结果, 求出 x, y 空间各群的点串的每一点的曲率; 根据各群的曲率数据抽取正圆; 根据点串的曲率数据抽取空间不可微点作为临时连接点; 把位于临时连接点近旁的其它候补连接点根据相关关系曲线概率地进行曲线拟合, 求出最佳连接点; 从最佳连接点中找出并除去即使没有也能保证近似精度的不必要的连接点; 当按直线、圆弧的顺序近似同一点串群的相邻连接点之间不能得到指定的近似精度时, 就用以 t 为自变量, 以 x, y 为因变量的分段多项式进行近似, 并在增加分段多项式的因次数的同时, 反复进行最小二乘近似直到近似精度达到指定的数值后, 再利用直线、圆弧、分段多项式近似相邻连接点之间的连线; 通过对各点串群存储上述近似连接点的坐标和相邻连接点的函数的参数, 输入词符一插图; 另一方面, 输出词符一插图时, 对各点串群输出存储的近似连接点的坐标和相邻连接点之间的函数的参数; 然后再生词符一插图的轮廓线; 通过使再生的轮廓线内的象素和外部象素对应不同的值, 再生出词符一插图, 最后, 把再生的词符一插图作为词符一插图输出。

14. 一种词符一插图的输入输出方法, 其特征在于:

光学读取词符一插图数据, 使这些数据与有限个纵横排列的象素对应并存储这些数据; 以 t 为自变量, 以 X, Y 为因变量存储对每个把抽取出的轮廓线的 2 维坐标 (X, Y) 连续起来的群; 求出

x, y 空间各群的点串的每一点的离散曲率;根据各群的曲率数据抽取正圆;根据点串的曲率数据抽取出空间不可微点作为连接点;当按直线、圆弧的顺序近似同一点串群的相邻连接之间不能得到指定的近似精度时,就用以 t 为自变量,以 x, y 为因变量的分段多项式进行近似,并在增加分段多项式的因次数的同时,反复进行最小二乘近似直到近似精度达到指定的数值后,再用直线、圆弧、分段多项式近似相邻连接点之间;通过对各点串群存储近似上述连接点的坐标和相邻连接点的函数的参数,输入词符—插图;另一方面,输出词符—插图时,对各点串群输出存储的近似连接点的坐标和相邻连接点之间的函数的参数;然后再生词符—插图的轮廓线;通过使再生的轮廓线内的象素和外部象素对应不同的值,再生出词符—插图;最后把再生的词符—插图作为词符—插图再生。

15. 一种词符—插图数据的输入输出方法,其特征在于:

光学读取词符—插图数据,使这些数据与有限个纵横排列的象素对应并存储这些数据;抽取与纵横排列象素对应的已读取的词符—插图的轮廓线;以 t 为自变量, X, Y 为因变量存储对每个把抽取出的轮廓线的 2 维坐标 (X, Y) 连续起来的群;设自变量为 t , 因变量为 x, y , 用以 t 为自变量 t, x, y 为因变量的分段多项式近似上述各群的轮廓点串的 X, Y 坐标;根据上述近似结果,求出 x, y 空间各群的点串的每一点的曲率;根据各群的曲率数据抽取正

圆；根据点串的曲率数据抽取空间不可微点作为连接点；当按直线、圆弧的顺序近似同一点串群的相邻连接点之间不能得到指定的精度时，用以 t 为自变量，以 x 、 y 因变量的分段多项式进行近似；通过对各点串群存储近似上述连接点的坐标和相邻连接点之间的函数的参数，输入词符一插图；另一方面，输出词符一插图时，对各点串群输出存储的近似连接点的坐标和相邻连接点之间的函数的参数；平移这些连接点的坐标，或者将连接点的坐标和圆弧的曲率、分段多项式的系数等乘以 x 方向、 y 方向的系数，通过求和进行平移、放大缩小、旋转及不同方向的放大缩小等等之后，再生词符一插图的轮廓线；通过使再生的轮廓线内的象素和外部象素对应不不的值，再生出词符一插图，最后，把再生的词符一插图作为词符一插图输出。

16. 一种词符一插图数据输入输出方法，其特征在于：

光学读取词符一插图数据，使这些数据与有限个纵横排列的象素对应；抽取与纵横排列象素对应的已读取的词符一插图的轮廓线；以 t 为自变量，以 X 、 Y 为因变量存储对每个把已抽取出的轮廓线的 2 维坐标 (X, Y) 连续起来的群；求出 x 、 y 空间各群的点串的每一点的离散曲率；根据各群的曲率数据抽取正圆，根据点串的曲率数据抽取空间不可微点作为连接点；当按直线、圆弧的顺序近似同一点串群的相邻连接点之间不能得到指定的精度时，就用以 t 为自变量，以 x 、 y 为因变量的分段多项式进行近似，

并在增加分段多项式的因次数的同时,进行最小二乘近似直到近似精度达到指定的数值后,再用直线、圆弧、分段多项式近似相邻连接点之间的连线;通过对各点串群存储近似上述连接点的坐标和相邻连接点之间的函数的参数,输入词符—插图数据;另一方面,输出词符—插图时,对各点串群输出存储的近似连接点的坐标和相邻连接点之间的函数的参数,平移这些连接点的坐标或将连接点的坐标和圆弧的曲率,分段多项式的系数等乘以 x 方向、 y 方向的系数,通过求和,进行平移、放大缩小、旋转、不同方向的放大缩小等某种操作之后,再生词符—插图的轮廓线;通过使再生的轮廓线内的象素和外部象素对应不同的值,再生出词符—插图,最后,把再生的词符—插图作为词符—插图输出。

说 明 书

文字数据、词符—插图数据的输入

输出装置及其方法

本发明是通过图象读取装置以二值数据得到文字字体(铅印体或毛笔体等)、词符—插图等(为简单起见,以下简称为词符—插图)数据后,在不丢失文字、词符—插图的特征的情况下,去除噪声,进行压缩、存储,并根据压缩过的数据重现文字、词符—插图的技术。特别是利用根据母字把文字字体或根据原图把词符—插图自动地变为数字数据,以及以任意的尺寸在任意的位置进行再生,能够简单地在印刷机器、文字处理器、微机中使用的技术。

文字字体是规定形状文字的集合,要想将其自由地缩小或放大后加以利用,就需要以抽象的数据而不是文字的二维形状进行存储。由于包括汉字在内的文字的数量很多,所以,如果每一个文字的数据量很大的话,不但输入数据费时间,还要求存储装置的容量要很大。还有,输出数据也费时间,故而很不经济。最好是每一个文字的数据量尽可能少。

以往,把每个文字的数据存储到存储装置中的方法有两种:

①. 把文字数据作为二值数据串进行编码,这种方法,是把文字作

为黑白二值图象，通过微分等方法抽取出轮廓线，求出轮廓线的连续方向后，编成链式字符，存储始点和字符串。

②. 抽取出文字字体的轮廓线，用直线或圆弧进行近似，存储始点的坐标和直线、圆弧等的函数。由于这种方法能够压缩数据，因而输入、输出很便利。

本发明属于后一种方法。抽取出文字字体的轮廓线后，编成链式字符时，由于根据图象最小单位，即象素，存储轮廓点的坐标，所以，放大时，再生文字的质量低劣。因为是将存储的数据乘上一定的乘数后来决定放大文字的坐标的，故轮廓点不光滑，成为锯齿状。因此，用链式字符法不能再生任意倍率的良好放大文字。

轮廓线可以视为直线与圆弧等的组合，但是，有时也有一部分是其它形状。这些能用某些函数进行表示。考虑到数据处理的简易性，数据量越少越好。为了正确地表现文字字体，必须求出能够正确地近似轮廓线的数据。因此，希望有既能尽量减少数据量又有尽量正确地近似轮廓线的方法。

文字字体是二维信息。关于这方面的先有技术比较少。作为关于一维信息的数据压缩方法，有特愿昭 63-211948 号专利。这是用二次曲线近似一条连续的曲线，根据近似式求出各点的曲率后，求微分不连续点。微分不连续点相当于连接点。接着，用二次函数近似相邻微分不连续点间的数据后，进行压缩。这个二次函数可以由简单的参数指定。于是，通过指定微分不连续点和二次函数串的

参数，能够近似一维数据。

词符一插图是任意形状和尺寸的图形、花纹或它们的集合。虽然也有3维的情况，但这里以2维情况为对象。和文字不同，形状不是预先确定的，而是根据设计人员的创意创造出来的。有多种多样的词符一插图，但在本说明书中，参照图17所示的乐谱音符、图18的半导体二极管、图44的圆形图表、图45的香烟以及图51—图57的例子等等进行说明。

原图是以画在纸等材料上的状态直接保存的。原图只有1份。要想再生只有用复印机进行光学复制。想放大或缩小时，就用复印机进行光学放大或缩小。

还有，想把词符一插图插入某种文件中指定的位置时，需要用复印机复印，然后剪切下来，再用浆糊粘到所希望的位置。这全部是手工作业。这样，由于是不定形的形状、花纹的集合，所以，难于进行词符一插图的记录、再生、放大、缩小等处理。

因为是设计人员创造出来的，故词符一插图的数量极多，而且又由于一个符号反复使用次数不多，所以，词符一插图处理的自动化还没有达到完全理想的程度。除用复印机复制外也有用印刷机大量复制再生的。然而，这是复印原画后制的版，所以和光学复制相同。

因此，能够称为关于词符一插图自动处理的先有技术还不存在。

以往，不存在能够以较少的数据存储文字字体并能再生任意大小文字的技术。前述方法是自变量为时间，因变量是位移时使用的方法，所以，数据是一维的信息，例如，处理像声音这样的变量的情况。

前述方法不能用于文字字体这样的二维信息的数据压缩。以横轴为 x ，纵轴为 y 时，可以把 y 作为 x 的函数表示文字字体，所以看起来似乎可以把这个方法扩展到 2 维使用。然而，事实并不如此。由于文字字体的轮廓线是闭合的二维信息，因此，若取独立变量为 x ， y 就成为 x 的多值函数，所以不能简单使用前述方法。

还有，前述方法的对象不是二维信息，所以不具有轮廓线和连接点这样明确的意义。变量的变化是随机的，微分不连续点的意义也不明确，所以，没有考虑这些点的决定是否存在误差。

然而，对于文字的情况，不仅包含有大量的直线成分，还有很多圆弧部分。当然，变量不会是随机变化的。

另外，为了把前述方法应用于二维信息，可以考虑某一独立变量，把轮廓线的 x 、 y 坐标视为是相对于该独立变量的因变量。但，这时不明确什么量表示曲率。处理二维信息时首先出现的是连接点，但是并不是仅作为微分不连续点而求出的，必须考虑噪声的影响。有时，表观上噪声也以连接点的形式出现。因此，必须去除这些噪声，求出正确的连接点。

本发明第一个目的是提供一种文字字体数据输入输出装置。该

装置能读取文字字体、自动地进行数据压缩和存储，并能以任意尺寸的文字、在任意的位置再生文字字体。其目标如下：

- (1). 节省造字的劳力；
- (2). 能进行高品质的文字处理；
- (3). 减少字体的数据量。

以往，尚不存在不用复印机而能再生词符一插图的技术。除了直接保存原图之外，不存在其它的记录方法。况且，缩小、放大等处理只有依赖复印机。这样的方法根本不能自动地处理词符一插图。用人工方法进行处理不仅费时，而且需要熟练的操作技术。另外，反复复制后就会掺入噪声降低画面质量，放大后轮廓线就变得模糊不清，不能得到漂亮的放大图。

于是，人们迫切期望能有一种不是保存原图进行光学处理，而能以数字数据记录词符一插图，并能以任意的尺寸、在任意的位置漂亮地再生词符一插图的方法。

为满足这一要求，本发明的第2个目的是要提供一种词符一插图的输入输出装置。该装置能用光学方法读取词符一插图、自动地进行数据压缩和存储，并能以任意的尺寸在任意的位置再生忠实于原图的词符一插图。其目标如下：

- (1). 节省造词符一插图的劳力；
- (2). 能进行高品质的词符一插图处理；
- (3). 减少存储词符一插图的数据量。

本发明的文字数据、词符—插图数据输入输出装置的特征在于包含：

文字、词符—插图读取装置，用于以光学方式读取文字字体数据、词符—插图数据，并对应于纵横方向上有限个排列的象素存储这些数据；

轮廓线抽取机构，用于对应纵横排列的象素抽出已读取的文字的轮廓线；

轮廓点串存储机构，用于对每个使抽出的轮廓线的二维坐标 (x, y) 连续的群进行存储；

数据近似机构，用于把自变量记为 t ，把因变量记为 x, y ，用以 t 为自变量、以 x 和 y 为因变量的2次分段多项式近似上述每个群的轮廓点串的 x, y 坐标，反复进行最小二乘法近似，直到近似精度达到指定的范围，从而求出每个轮廓线点中的每个群的近似多项式；

曲率运算机构，用于从上述近似结果求 x, y 空间的每个群的点串中各点的曲率；

正圆抽取机构，用于从每个群的曲率数据中抽取出正圆；

临时连接点位置抽取机构，用于从点串的曲率数据中把空间不可微点作为临时连接点抽取出来；

最佳连接点抽取机构，用于把位于临时连接点近旁的其它候补连接点根据相关关系进行概率拟合，求出最佳连接点；

不必要的连接点去除机构，用于从最佳连接点中找出即使去掉也可以保证近似精度的那些不必要的连接点；

数据近似机构，用于当按直线、圆弧的次序近似同一点串群内相邻连接点的间隔而不能得到指定的近似精度时，用以 t 为自变量、以 x 、 y 为因变量的2次分段多项式近似，并增加2次分段多项式的因次数，同时反复进行最小二乘法近似，直到近似精度达到指定的数值后，在相邻连接点间用直线、圆弧、2次多项式进行近似；

压缩数据存储机构，用于存储点串的每个群中近似上述连接点的坐标和相邻连接点的函数的参量；

压缩数据输出机构，用于输出存储的用来近似点串的每个群中的连接点的坐标和相邻连接点的函数的参量；

压缩数据输入机构，用于输入压缩数据；

轮廓线再生机构，用于根据输入的压缩数据得到近似点串的每个群的连接点的坐标和相邻连接点函数的参量，从而再生轮廓线；

文字、词符—插图再生机构，用于使与再生的轮廓线的内部象素与外部象素不同的值对应；

再生数据输出机构，用于把再生的文字字体和词符—插图作为文字、词符—插图输出。

图1是表示本发明的文字字体、词符—插图数据输入输出装置

的总体结构图。

图 2 是在读取了文字、词符—插图的图象上对于象素定义的 x, y 坐标系的示意图。

图 3 是一个象素周围的链形码示意图。

图 4 是在图象数据中用于搜索轮廓点的扫描方向示意图。

图 5 是在已知一个轮廓点时用于寻找下一个轮廓点的搜索方向示意图。

图 6 是存在黑象素和白象素时某轮廓点和下一个轮廓点的示意图。

图 7 是在“口”字型的图象数据中抽出外侧轮廓线和内侧轮廓线，并使 2 值数据与象素对应的状态示意图。

图 8 是表示轮廓点串抽取装置结构的流程图。

图 9 是为近似轮廓点串求曲率的数据近似机构的流程图。

图 10 是数据近似机构的一部分，表示在用分段多项式近似全部轮廓点串时决定系数的过程的流程图。

图 11 是用于计算前段所述近似的轮廓点串的各点处的曲率的流程图。

图 12 是表示根据用分段多项式所近似的数据抽取正圆的机构的流程图。

图 13 是表示连接点抽取方法的概略的流程图。

图 14 是表示根据前面计算的曲率抽取连接点的位置的机构的

流程图。

图 15 是用于制作文字字体的原文字“あ”、“ホ”的示例图。

图 16 是求“あ”和“ホ”的轮廓点串并进而抽出连接点的示例图。

图 17 是作为词符—插图的一例的乐谱的 16 分音符图。包括原图(a)和表示其轮廓点串抽取结果和连接点抽取结果的图(b)。“x”号表示连接点。

图 18 是作为词符—插图的一例的半导体二极管图。包括原图(a)和表示其轮廓点串抽取结果和连接点抽取结果的图(b)。

图 19 是最佳连接点抽取机构概略的流程图。

图 20 是根据曲率所选择的临时连接点及其近旁的候补连接点编号的示意图。

图 21 是在根据曲率所选择的临时连接点附近有近旁连接点时输出近旁的各点的编号示意图。

图 22 是在有近旁连接点时决定相宜系数的流程图。

图 23 是连接根据曲率所选择的临时连接点附近的候补连接点 C、输出近旁的候补连接点 B、临时连接点的下一个连接点 D 与输出近旁的连接点的前一个连接点 A 的近似曲线和拐点的数目的示意图。

图 24 是在根据曲率所选择的临时连接点附近不存在近旁连接点时求相宜系数的流程图。

图 25 是在临时连接点附近不存在近旁连接点时，该临时连接点的候补连接点 B 、前后临时连接点的候补连接点 A 、 C 和连接它们的曲线、以及表明这些点的编号的示意图。

图 26 是说明根据曲率所选择的临时连接点附近的候补连接点 C 、输入近旁的近旁连接点附近的候补连接点 B 和该临时连接点前面一个临时连接点的候补连接点 D 的定义和符号的说明图。

图 27 是用于决定一个临时连接点的输入近旁及输出近旁的近旁连接点的流程图。

图 28 是表示候补连接点内哪个为最佳连接点的概率变量的定义和反复运算的流程图。

图 29 是其它最佳系数的给出方法的流程图。

图 30 是对以临时连接点为中心的 $2\alpha+1$ 个候补连接点给出的初始概率的示意图。

图 31 是对全部临时连接点进行概率变量的计算并把给出最大概率的候补连接点作为最佳连接点存储到最佳连接点存储装置的示意图。

图 32 是以“ β ”为例，简单地回顾迄今为止的操作程序的示意图。

图 33 是以变形二极管的图为例简单地回顾迄今为止的操作程序的示意图。

图 34 是说明从最佳连接点串去除不必要的连接点求出最终连

接点的操作流程图。

图 35 是在多个直线连接点连续存在时，应判断是否应去除的对象的中间连接点和两端连接点的符号的示意图。

图 36 是在图 35 的配置下从中间的连接点向连接两端的连接点的直线所做的垂线的垂足长度小于某个阈值时去除中间的连接点的示意图。

图 37 是在多个圆弧连接点连续存在时应判断是否应去除的中间连接点和两端连接点的符号的示意图。

图 38 是在图 37 中从第 1 个连接点开始的圆弧的半径、中心，以及从第 2 个连接点开始的圆弧的半径、中心的示意图。

图 39 是根据轮廓点串、最终连接点、正圆等的数据来近似数据，并利用近似函数表示轮廓点串的操作说明图。

图 40 是“体”字的原图象和利用本发明的方法将其进行数据压缩后再生的图象。

图 41 是“语”字的原图象和处用本发明的方法将其进行数据压缩后再生的图象。

图 42 是“明”字的原图象和利用本发明的方法将其进行数据压缩后再生的图象。

图 43 是乐谱的 16 分音符的词符—插图的原图象 (a) 和利用本发明的方法进行数据压缩后再生的图象 (b)。

图 44 是圆形图表的词符—插图的原图象 (a) 和利用本发明的

方法进行数据压缩后再生的图象(b)。

图 45 是香烟的词符—插图的原图象(a)和利用本发明的方法进行数据压缩后再生的图象(b)。

图 46 是 2 次光滑函数和 0 次、1 次、3 次光滑函数(*fluency*)的概略图。

图 47 是采用 2 次光滑函数、而基函数的数目较少时,即使不能充分地近似数据,只要提高因次数,就能提高近似程度的事例的概略图。

图 48 是以“和”字为例,用分段多项式近似全部轮廓点串的过程的示意图。

图 49 是以香烟的词符—插图为例,用分段多项式近似全部轮廓点串的过程的示意图。

图 50 是半导体二极管的词符—插图的原画象(a)和利用本发明的方法进行数据压缩后再生的图象(b)。

图 51 是含有文字且自由曲线较多的词符—插图的例子。

图 52 是图 51 的词符—插图的缩小图。

图 53 是自由曲线较多的词符—插图的例子。

图 54 是根据本发明的方法把图 53 存储、变形后输出的示例图。

图 55 是包含装饰文字和图形的词符—插图的例子,(a)是缩小图,(b)是放大图。用本发明的装置存储其中的任一个就能够简

单地得到另一个图形。

图 56 是利用本发明的方法把蜗牛的词符一插图印到明信片的某位置上的示例图。可以任意决定、变更尺寸和位置。

图 57 是利用本发明的方法把山岳的词符一插图印到明信片的某位置上的示例图。可以任意决定、变更尺寸和位置。

图 1 中把总体结构示为一览表的形式，在这里，预先写出所有的机构：

- A. 图象存储装置 1
- B. 轮廓点串抽取装置
- C. 轮廓点串存储装置
- D. 数据近似机构 A
- E. 曲率运算机构
- E'. 近似曲率存储装置
- F. 正圆抽取机构
- G. 正圆存储装置
- H. 连接点位置抽取机构
- I. 连接点位置存储装置
- J. 最佳连接点抽取机构
- K. 最佳连接点存储装置
- L. 连接点去除机构
- M. 最终连接点存储装置

- N. 数据近似机构 B
- O. 压缩数据输出机构
- P. 压缩数据存储装置
- R. 轮廓再生机构
- S. 文字、词符—插图再生机构
- T. 再生数据输出机构
- U. 图象存储装置 2。

作为例子，对图 32 所示的“ぼ”字简单地说明操作顺序。图 32 后面还要再一次说明。首先用图象扫描器读取写在纸上的“ぼ”，这是文字的光学读取、输入。当抽出轮廓线的集合即轮廓点串时，就成为黑框白底字。在本例中有 5 个轮廓点串群。左边的纵线有 1 个，中央与“よ”相似的部分有 2 个，右边肩部的圆圈有内外 2 个。使用参变量表示方法，把 X, Y 坐标作为 t 的函数。在每一个轮廓点串群中，用分段多项式进行全部近似。在分段的多项式中成为连续函数，所以在各轮廓点串中可以进行 2 阶微分求曲率。曲率一定的轮廓点串是正圆，这样就作为正圆被分离出来。在本例中，右上肩的“O”就被去除了。在本例中剩下的还有 3 个轮廓点串群。曲率大的位置是连接点。在第 4 步，对轮廓点串的交点和弯曲点等处标上“x”号。这是连接点。也有后来追加轮廓点串的情况。

根据连接点分割出轮廓点串，若曲率很大，也可以使用所求出的最初的连接点(称为临时连接点)作为连接点。对于文字字体的

情况，因为从开始就是以漂亮地书写的文字为对象的，所以噪声难以混入，故即使只根据曲率也能够正确地求出连接点。这样，计算就极其单纯化了。

有时，要对连接点进一步反复推敲。这时，便考虑与邻近连接点的配合来修正连接点。这些作为临时连接点的候补连接点将在后面进行说明。另外，修正连接点的位置后，对去除不必要的连接点也是有效的。也有再追加新连接点的情况。这样，就能有效地去除在进行光学读取等操作时所产生的噪声。

当确定了连接点时，就可以用直线、圆弧、自由曲线近似连接点之间的连线。近似按直线、圆弧、自由曲线的顺序进行。由于是文字字体，所以直线部分较多。特别是对于汉字的情况，直线部分的比率极高，仅次于直线的是圆弧，对于汉字的情况，可以认为比率高低的顺序是直线、圆弧、自由曲线。不过，近似的顺序为直线、圆弧、自由曲线的理由并不是因为其出现频度为这个顺序。

不是由于频度，而是为了提高所记录的数据的品质来按这个顺序近似的。如果从开始就以自由曲线进行近似，则直线和圆弧也就相应地被用2次曲线进行近似。2次曲线的集合不能正确地表现直线和圆弧。存在噪声时，近似出的直线、圆弧更加变形。由于确实存在直线和圆弧部分，故为确实地抽出它们而按直线、圆弧、自由曲线的顺序进行近似。对于直线，进行近似所需要的时间短，数据量也少。圆弧的近似也比较简单，只要给出圆心的坐标、半径、圆心

角就可以了。

既不能用直线，也不能用圆弧进行近似时，就用自由曲线进行近似。这时，把连接点之间分割为 n 个小区间用分段多项式进行近似。由于增加小区间分割数能够提高近似程度所以近似达到所希望的精度。

这样，由于可以得到连接点和直线、圆弧、自由曲线的参量，所以可以把这些参量作为文字字体的数据进行存储。随各文字的笔划数和复杂程度不同，1 个文字大约用 300—500 字节的数据就够了。如果是黑白图象的原图，则需要构成画面的全部象素数的数据。例如，假设有纵横 256 个象素，则也有 $8k$ 个字节，而按照本发明就能大幅度地压缩数据。反过来，读出这些数据后，又可以连接点为基准再生出直线、圆弧、自由曲线。并且通过计算，能够以任意的尺寸、在任意的位置再生。

作为词符—插图的例子，简单地说明图 33 所示的半导体二极管的词符—插图的操作顺序。图 33 还要在后面再次说明。这是为说明正圆的抽取，在右上肩画入了一个多余的圆。

首先，用图象扫描器读取画在纸上的半导体二极管的图形，这是词符—插图的光学读取和输入。当抽取出轮廓线的集合即轮廓点串时，就成为黑框白底的词符—插图。在本例中，作为轮廓点串的群有 5 个。右上肩的圆有内外 2 条轮廓线，二极管部分有外侧轮廓线和内外上、下轮廓线，合计是 5 条。

用参变量表示,把 x, y 坐标作为 t 的函数。在各个轮廓点串群中,全部用分段多项式进行近似。由于在分段多项式中为连续函数,所以在各轮廓点串中可以做 2 阶微分求曲率。曲率一定的轮廓点串是正圆,这样就作为正圆被分离出来。在本例中,右上肩的“O”就被去除。

在本例中,余下的还有 3 个轮廓点串群,曲率大的位置是连接点。在第 4 步,对轮廓点串的交点、拐点等处标上“x”号,这就是连接点。也有在后来追加轮廓点串的情况。

利用连接点分割出轮廓点串。若曲率很大,也可以使用所求出的最初的连接点(称为临时连接点)。对于词符一插图的情况,若从开始就使用漂亮地描画的词符一插图,由于噪声难以混入,故即使只根据曲率也能够正确地求出连接点。这样,计算就极其单纯化了。但是,按照本发明,即使根据包含噪声的原图也能够得到非常漂亮的复制图形。

以后的顺序与文字字体的情况相同。

实施例:

A. 图象存储装置 1

这是用光学方法读取写在纸等介质上的文字字体,词符一插图,并以分解为各象素的信息进行存储的装置。文字、词符一插图即使着色也可以,但是,所需要的是黑白 2 值图象。例如,将着色部分作为黑色,将不构成文字、词符一插图的部分作为白色形成 2

值图象，把它们按各个象素进行存储。

例如，用图象扫描器以 256×256 个点的精度输入。当然，点的数目是任意的，点数越多，作为文字、词符—插图所存储的内容的品质就越高，但是点数太多时，计算时间和存储容量就增大，所以，使用适当点数的图象读取装置就行了。对每个点(象素)区别出是白象素或黑象素后暂时存储起来。以后，有时把一个象素称为点。另外，把一系列连续的黑象素串称为点串。为了表示点，使用由画面上象素的横向符号 x 和纵向符号 y 约成的坐标 (x, y) 进行表示。对坐标变量标以各种下标以示区别。

B. 轮廓点串抽取装置

轮廓点串抽取装置是进行求所读取的文字、词符—插图的轮廓线的操作的装置。下面先介绍其大致的流程。

- ①. 从左上方到右下方扫描图象数据，并寻找还未被找到的轮廓点(示于图 4)。
- ②. 环绕轮廓线一周搜索下一个轮廓点。轮廓点用 2 标记。
- ③. 检查图象数据直至右下方的数据为止。

首先根据定义说明进行的这一操作。图象用象素的集合进行表示。象素用画面内的坐标进行指示。象素的坐标以行(横)方向的坐标为 x ，以列(纵)方向的坐标为 y 。左上方的坐标为 $x=0, y=0$ 。若横向有 x 个象素，纵向有 y 个的象素，则 x 的取值为 $0 \sim X-1$ ， y 的取值为 $0 \sim Y-1$ 。坐标系如图 2 所示，是指向右下方的。图象本

身是输入的文字、词符—插图，所以只取黑、白 2 个值。对每个象素进行黑白的指定。

二值图象输入：

按如下方式定义特性函数 g_{xy} ：

(i) 在 $0 \leq x \leq X, 0 \leq y < Y$ 内，坐标 (x, y) 上存在点(黑象素)时，

$$g_{xy} = 1 \quad (1)$$

(ii) 其它情况(在该点上不存在黑象素)

$$g_{xy} = 0 \quad (2)$$

轮廓点串的表现：

所谓轮廓点串是存在于黑象素群周围的黑象素的左右上下倾斜的连续点的串。不仅在外围，在内部也有轮廓点串。把闭合的一个点串称为轮廓点串。设轮廓点串的总数为 U 。对 U 个轮廓点串标以从 0 到 $U-1$ 的序号。第 u 个轮廓点串的轮廓点的总数用 $N(u)$ 表示。在一个轮廓点串中，对连续的点标以序号 k 。 k 是 $0 \sim N(u) - 1$ 的整数。

把第 u 个轮廓点串的第 k 个轮廓点的坐标用 (x_k^u, y_k^u) 表示。

全部轮廓点用

$$\{ (x_k^u, y_k^u) \}_{k=0}^{N(u)-1}, u=0^{U-1} \quad (3)$$

表示。 $k=0^{N(u)-1}$ 表示点串序号 k 能取从 0 到 $N(u)-1$ 的值。 $N(u)-1$ 含有括号, 而括号在打字时不能用 $1/4$ 角的格式, 所以作为变量的标注时, 去掉括号写为 $N(u)-1$ 。即 $N(u)-1=N(u)-1$ 。由于是标注, 故应该按上下方式写, 但这在打字时不可能, 所以分为左下和右上进行标注。 $u=0^{U-1}$ 表示轮廓点串群的序号 u 取 $0 \sim U-1$ 的值。另外, 轮廓点串的序号 u 应在变量的右肩加上括号, 但因为括号在打字时不能用 $1/4$ 角的格式故省略。实际上如图所示的那样是带有括号的。不是变量的 u 次方。这对于参变量 t 、自变量 x, y 等都是相同的。 u 是群序号, 直接写在变量的右肩上, 但实际上是带有括号的。

链形码:

如图 3 所示, 以某象素为中心, 周围 8 个点的象素按顺时针标以序号 0—7。这是为了表示轮廓点串连续而使用的。另外, 也用于调查连续的状态。设 C 为链形码, 定义如下的函数 $\alpha(c)$ 和 $\beta(c)$ 。函数 α

(c) 为

$$c=3, 4, 5 \text{ 时, } \alpha(c)=-1 \quad (4)$$

$$c=2, 6 \text{ 时, } \alpha(c)=0 \quad (5)$$

$$c=0, 1, 7 \text{ 时, } \alpha(c)=1 \quad (6)$$

函数 $\beta(c)$ 为

$$c=5, 6, 7 \text{ 时, } \beta(c)=-1 \quad (7)$$

$$c=0, 4 \text{ 时, } \beta(c)=0 \quad (8)$$

$$c=1,2,3 \text{ 时, } \beta(c)=1 \quad (9)$$

根据这样的定义进行轮廓线的抽取。如前述,像图 4 所示那样把整个图像从左上方到右下方进行扫描寻找出黑象素。轮廓线是二值图象的 $g_{xy}=1$ 部分的外周的象素,在循环 1 周进行搜索的同时,就把点串的坐标存储起来。环绕 1 周的方向规定为:在其区域的外侧环绕时,按顺时针方向旋转,在其内侧环绕时按逆时针方向旋转。图 5 示出了环绕轮廓点的顺序。当然也可以规定按与上述方向相反的方向环绕,但这里讨论的是按上述规定环绕的。

假定已知道了某轮廓点串中的某轮廓点。为了求出下一个轮廓点,就以此轮廓点为中心,从前一次的轮廓点开始按顺时针方向旋转搜索 8 个相邻象素。图 6 示出其一例。设标以“☆”符号的是最新的轮廓点。按顺时针方向旋转搜索下个轮廓点,但直到序号 1,2,3,4,5 全是白象素 $g_{xy}=0$,所以不符合条件。第 6 个象素是黑象素,因 $g_{xy}=1$,所以是轮廓点。这样,由于求出了下一个轮廓点,所以就以第 6 个象素为中心来求其下一个轮廓点。

下面,用同样的顺序可以求出一个个轮廓点。因是根据邻近 8 个象素求下一个轮廓点的,所以能够得到连续的轮廓点串。轮廓点把 g_{xy} 的值取为 2。由此能够与内部的黑象素 $g_{xy}=1$ 相区别。另外,由于 $g_{xy}=1$ 的象素应该局限在画面上,所以如果逐次求轮廓点,就会回到最初的轮廓点。即轮廓点串形成环状。设这样的轮廓点串总共有 U 个,轮廓点串的序号用 u 表示,某轮廓点串 u 部的点的序号用 k 表

示。

确定了1个轮廓点串时,就重新开始图4的扫描,搜索下一个轮廓点。知道了这个轮廓点后,就按同样的顺序确定轮廓点的坐标。有多个轮廓点串时,当然是从左上方开始进行这些处理并顺序标以0、1、 \dots (U-1)的轮廓点串序号。对于内部有孔的情况如图5所示,可以通过若干次的扫描检出某些点。虽然进行图6所示的相邻点的搜索,但是,有孔时就如图5所示的那样按逆时针方向旋转找出轮廓点。这样,不论是外部的轮廓点串还是内部的轮廓点串,探索到后就存储点串的坐标。这些点串的 g_{xy} 的值成为2。

由于轮廓点串搜索是对于 $g_{xy}=1$ 的象素进行的,所以在外周和内周的宽度仅为1个象素时要产生一点问题。如图7所示。(a)是仅为1个象素宽度的环状的图形。 $g_{xy}=1$ 的象素分布为正方形。若宽度大于2个象素,就可以按前述的顺序毫无问题地求出外部轮廓点串、和内部轮廓点串。但是,如果宽度是1个象素,由于在先确定了外周的轮廓点串后, $g_{xy}=1$ 的象素是轮廓点,故变为 $g_{xy}=2$ 。因为轮廓点的搜索是对于 $g_{xy}=1$ 进行的,因此 $g_{xy}=2$ 就不能再成为轮廓点串。于是,不去识别内周的轮廓点串。因此,就不考虑孔的存在。由于这样是不行的,所以,作为对策,规定当下一个轮廓点位于现轮廓点的下方时就不进行赋值($g_{xy} \rightarrow 2$),该点仍保留为 $g_{xy}=1$ 。在图7(b)中,右边中间的值1就表示这一点。因为数值为1,所以,就找到了作为内轮廓点的出发点。从该点出发能够抽出内轮廓点串。结果,可

以既不忽略内部的孔又能找出内外轮廓点串。图 7(c) 示出了这一结果。

结果确定了轮廓点串，由轮廓点串所包围的内部的像素就不是为了规定这个区域最早所必须的像素了。因此，内部的像素就不必存储。这样，就可以先减少应存储的数据量。

C. 轮廓点串存储装置

轮廓点串存储装置是存储在前段所求的轮廓点串的装置。以 $(x_k^u, y_k^u)_{k=0}^{N^u-1} \quad u=0^{U-1}$ 的形式进行存储。如前所述， U 是总点串数， u 是点串的序号。点串 u 是的点的数目是 $N(u)$ ， k 是点的序号。用 $k=0^{N^u-1} \quad u=0^{U-1}$ 表示上述规定。 (x_k^u, y_k^u) 是第 u 个点串的第 k 个点的 x 、 y 坐标。再重复说明一下，把 $N(u)-1$ 以标注的形式写为 $Nu-1$ 。

D. 数据近似机构 A

数据近似机构有 2 个。这是最初的 1 个，为区别起见标以 A。这是假定求出的轮廓点串的曲率大时，为了求连接点所必须的。为了求曲率，也可以在离散的点内采用若干个点作为数据，根据这些数据来求。本发明采用这样的离散曲率也可以实施。但是，这里用连续函数近似数据后，将其进行 2 阶微分求曲率。因此，此处的近似是数据近似。

因为是求曲率，所以，通常的方法是把 x 作为自变量， y 作为因变量，求 y 对 x 的二阶微分。但是这样一来，对 x, y 的处理就变成非对称的，这是所不希望的。本发明不采用这样的方法。

这里, 设自变量为 t , 因变量为 x, y , 用以 t 为自变量, x 和 y 为因变量的 2 次分段多项式近似前述每个连续群的轮廓线的点串的 x, y 坐标, 反复运用最小二乘法近似, 直到近似精度达到指定的范围, 求出轮廓线点串的每个群的近似多项式。这样做不是要得到最终的数据而是求连接点。

如前所述, 把群 u 的点串 k 的坐标写为 (x_k^u, y_k^u) 。根据前述轮廓线点串存储机构可以得到这个坐标。图 9 示出了这一操作。先从轮廓点串存储装置读入轮廓点串全部的 (x_k^u, y_k^u) , 把它们分解为参变量。即对于各点使共同的参变量 t 与 2 个 (x_k^u, y_k^u) 对应, 也可以附加标注, 成为 t_k^u 。虽然是二维信息, 但是, 为了把它用于一维问题而使用了参变量。图 9 的第 2 步示出了这一过程。

通过使用参变量, (t_k^u, x_k^u) 和 (t_k^u, y_k^u) 这 2 个坐标的组合就和各轮廓点串的各点对应。以后, 由于对 2 个变量的处理相同, 故仅对 1 个进行说明。

按照以二维光滑函数 (Fluency functions) 的函数 $\{\psi_m\}$ 为基底的线性组合给出近似群 u 中的轮廓点串 (t_k^u, x_k^u) 的 t 的函数 $S_x(t)$ 。根据 $S_x(t)$ 来近似群 u 中 t 的函数 x 。同样, 利用 $S_y(t)$ 也可以近似群 u 中的 y 。近似函数是否合适, 可以通过评价最小二乘法的误差是否在规定的范围内来确定。

应该注意的是, 这里是利用 $S_x(t), S_y(t)$ 近似轮廓点串群 u 的整个闭合曲线的。是用 1 个函数 $S_x(t)$ 近似全体而不是中途具有的连接

点。这是因为还没有决定连接点。如前所述,为了求曲率,有不利用近似的更简便的方法,那就是直接用轮廓点串的数据求离散曲率的方法。为了实施本发明,也可以利用这样的离散曲率,但是,这里对此不作说明,而说明近似函数 $S_x(t)$ 、 $S_y(t)$ 的生成。

将 $S_x(t)$ 以非周期 m 次光滑函数 ψ_k 为基底展开,则

$$S_x(t) = \sum_{k=-m}^{M+m} C_k^x \phi_k(t) \quad (10)$$

所谓光滑函数是本发明者命名的函数名。次数 m 对应于多项式的因次。 M 是维数。一般,设 m 次光滑函数的定义域为 $[0, T]$, 参量为 k , 把这个参量作为角标表示。 C_k^x 是线性组合系数。 ψ_k 本身是在 k 附近有值的多项式。

$$\phi_k(t) = 3(T/M)^{-m} \sum_{q=0}^{m+1} (-1)^q \{t - (k+q)(T/M)\}^{m+q} / \{q!(m+1-q)!\} \quad (11)$$

式中,

$$k = -m, -m+1, \dots, 0, 1, 2, \dots, m+M$$

这里,标在 m 次下方的“+”号表示括号内的值为负时是 0,括号内的值为正时是 m 次方的意思,即定义如下:

$$(t-a)^m_+ = (t-a)^m \quad t > a \quad (12)$$

$$0 \quad t \leq a \quad (13)$$

基底函数 ψ_k 是在分段序号 $k \sim k+m+1$ 内具有有限值、而在其两侧为 0 的山峰形函数。这是把 $\{t - (k+q)(T/M)\}^{m+q}$ 从 0 上升的 m 次函数沿横向一个坐标一个坐标地移动(一个 q 一个 q 地增加)、重

迭组合而成的形状。 $t > (k+m+1)(T/M)$ 时,必须恒等于0。根据这一条件,重迭系数由 $(-1)^q / \{q! (m+1-q)!\}$ 决定。使区域的大小 T 与轮廓点串群的点数 $N(u)$ 相等是比较简单,但也可以按某种比例进行定义。这样,利用光滑函数就可以近似轮廓点串。但是,由于具有 T/M 间隔的分割点有很多,故即使没有连接点也能够近似。为提高近似程度,可以提高光滑函数的因次 m 。但是,如果因次数很高,计算次数就要增加,从而会使处理时间加长。

本发明者的主张是:表示众多自然界物理量变化的函数可以用1次、2次光滑函数的线性组合表示。光滑函数并不是完备、正交、规范化函数。如果采用 m 值取到无穷大的函数并把它们进行线性组合,就可以表现任意的函数,这是毫无疑义的。然而,本发明者所说的并非如此,而是利用很小因次数的光滑函数描述自然界物理量的变化的情况。这里只采用 $m=2$ 。这样就可以使文字、词符一插图等的轮廓线不会表现得过分不足。当然,本发明可以用 $m=3$ 以上的光滑函数构成。

如果采用最适宜的函数系并利用它们的线性组合来描述物理量的变化,则可用最少的函数得到最佳的近似。如果函数系不佳,则必须以较多的函数为基底展开线性组合式。这样,既不能得到良好的近似,而且最终的数据也增多,从而导致存储装置的负担增大。还有,读出这些数据进行利用也不容易。故应选择最佳函数系。本发明者认为 $m=2$ 为最佳。

本发明者在这里采用 $m=2$ 的光滑函数。图 46(a) 示出了 2 次光滑函数的概略。这是跨越 3 个区间的 2 次曲线。两端的上升边和下降边是 2 次函数。中央的点为最大值, 其邻域也是 2 次函数。

0 次光滑函数如图 46(b) 所示, 是仅在 1 个区间内取一定值的箱形函数。

1 次光滑函数如图 46(c) 所示, 是跨越 2 个区间的三角形函数。

3 次光滑函数如图 46(e) 所示, 是涉及 4 个区间的光滑函数。

一般 m 次光滑函数是跨越 $(m+1)$ 个区间、中部具有极大值中部的平滑 ($m \geq 2$) 的函数。两端有以 m 次方的上升边和下降边。中部的函数形状仍是 m 次方。基底 ψ_k 的参量 k 每增加 1, 就从原来的位置向右平移 1 个单位。如果用 m 次分段多项式近似在某区间的某种变化, 则在该区间具有值的函数的基底有 $(m+1)$ 个。虽然可以采用 $m=3$ 以上的函数, 但这里取 $m=2$ 。即, 在 3 个区间上有值, 极大值和端点处的 2 次函数。上式中 $m=2$ 时, 则

$$S_x(t) = \sum_{k=-2}^{M+2} C_k \psi_k(t) \quad (14)$$

$$\psi_k(t) = 3(T/M)^{-2} \sum_{q=0}^3 (-1)^q \{t - (k+q)(T/M)\}^2 / \{q!(3-q)!\} \quad (15)$$

基底函数是 4 个从 0 开始上升的 2 次函数的重迭组合。这 4 个函数以 $\{t - (k+q)(T/M)\}^2$ 所示的向横向逐次移动 T/M 。是小区间的数目从 k 到 $k+3$ 有值的函数。由于大于 $k+4$ 时恒等于 0, 故重迭系数成为 $(-1)^q / \{q!(3-q)!\}$ 。基数函数的数目为 $M+5$ 个。 M 是全

区间的分割数,将其称为近似的因次数。不能把它与光滑函数的次数 m 混同,用图 47 说明这一点。图(a)示出数据点。数据的区域是 $0 \sim T$ 。图(b)示出基底函数为 3 个、因次 M 为 1 的情况。这时,由于基底函数太少,所以系数的量也少,因而给不出良好的近似。然而,如果增加因次 M ,就如(C)所示,无论多么复杂的变化也能近似。近似的程度可以用在多大程度上接近原轮廓点串 (x_k^u, y_k^u) 来判断。根据小二乘法评价,就是使

$$Q = \sum \{ S_x(t_k^u) - x_k^u \}^2 + \{ S_y(t_k^u) - y_k^u \}^2 \quad (16)$$

为最小。求和的范围是轮廓点串群 u 的全部点。这里,由于仅求曲率,所以精度不必很高。在后面说明的第 2 次的数据近似 B 中也是进行同样的近似。那时,精度将更高。

图 10 示出这一部分的流程。这是决定系数 C_h ,它的阶段是 M 。规了某个 M 时,根据(10)式,系数 C_h^x 就唯一地决定了。然而,该系数并不限于满足小二乘法的限制。这时,使因次 M 增加 1。因而,当达到所希望的近似范围时,就用这个值确定因次 M 中的系数 C_h 。

(10)式左边的 $S_x(t)$ 在 $t=t_k$ 时应成为 $S_x(t)=x_k^u$ 。而且,把这个值代入到(10)式中,则

$$x_k^u = \sum_{h=-2}^{M+2} C_h^x \phi_h(t_k^u) \quad (17)$$

把上式乘以 $\psi_w(t_k^u)$,并对点串 k 求和,则

$$\sum_{k=0}^{N-1} \psi_w(t_k^u) x_k^u = \sum_{k=0}^{N-1} \sum_{h=-2}^{M+2} C_h^x \phi_h(t_k^u) \psi_w(t_k^u) = \sum_{h=-2}^{M+2} \sum_{k=0}^{N-1} \phi_h(t_k^u) \psi_w(t_k^u) C_h^x \quad (18)$$

这里，从第2式到第3式的变化是交换了求和的顺序。由于是有限数列，故可以自由地变更求和次序。

该式的左边 $\sum_{k=0}^{N-1} \phi_w(t_k^u) x_k^u$ 可以认为是标注为 $W(W=-2, -1, 0, 1, 2)$ 的 $M+5$ 维的列向量。右项的前项可以认为是光滑函数的乘积之和 $\sum_{k=0}^{N-1} \phi_h(t_k^u) \phi_w(t_k^u)$ 即 $M+5$ 列、 $M+5$ 行的正方形矩阵。另外，右边的后项 C_h^x 是 $M+5$ 列的行向量。由于在文字上方不能添加“ \rightarrow ”符号，所以决定以下把表示向量的符号“ \rightarrow ”加在文字之后。用

$$\text{向量} = c_{x \rightarrow} = {}^T \{C_{-2}^x, C_{-1}^x, \dots, C_{m+2}^x\} \quad (19)$$

$$\text{向量} = b_{x \rightarrow} = {}^T \{ \sum_{k=0}^{N-1} \phi_{-2}(t_k^u) x_k^u, \sum_{k=0}^{N-1} \phi_{-1}(t_k^u) x_k^u, \dots, \sum_{k=0}^{N-1} \phi_{m+2}(t_k^u) x_k^u \} \quad (20)$$

定义向量 $b_{x \rightarrow}$ 、向量 $c_{x \rightarrow}$ 。T 表示转置矩阵。

矩阵 G 定义为

$$G = \{G_{hw}\} = \{ \sum_{k=0}^{N-1} \phi_h(t_k^u) \phi_w(t_k^u) \} \quad (21)$$

向量之间下式成立：

$$b_{x \rightarrow} = G c_{x \rightarrow} \quad (22)$$

这是求向量 $c_{x \rightarrow}$ 的公式，但是，以

$$c_{x \rightarrow} = G^{-1} b_{x \rightarrow} \quad (23)$$

的形式可以求关于某个因次数 M 的系数 C_h 。 G^{-1} 是矩阵 G 的逆矩阵。这是对于 x 的计算。对于 y 同样

$$S_y(t) = \sum_{h=-2}^{M+2} C_h^y \phi_h(t) \quad (24)$$

$$y_k^u = \sum_{h=-2}^{M+2} C_h^y \phi_h(t_k^u) \quad (25)$$

$$\sum_{k=0}^{N-1} \phi_w(t_k^u) y_k^u = \sum_{k=0}^{N-1} \sum_{h=-2}^{M+2} C_h^y \phi_h(t_k^u) \phi_w(t_k^u) = \sum_{h=-2}^{M+2} \sum_{k=0}^{N-1} \phi_h(t_k^u) \phi_w(t_k^u) C_h^y \quad (26)$$

也成立,用

$$\text{向量 } c_y \rightarrow = {}^t \{C_{-2}^y, C_{-1}^y, \dots, C_{m+2}^y\} \quad (26')$$

$$\text{向量 } b_y \rightarrow = {}^t \{ \sum_{k=0}^{N-1} \phi_{-2}(t_k^u) x_k^u, \sum_{k=0}^{N-1} \phi_{-1}(t_k^u) x_k^u, \dots, \sum_{k=0}^{N-1} \phi_{M+2}(t_k^u) x_k^u \} \quad (27)$$

来定义向量 $b \rightarrow, c \rightarrow$ 。和(23)式相同,对于 y 分量也可以以

$$c_y \rightarrow = G^{-1} b_y \rightarrow \quad (28)$$

的形式求关于某因次数 M 的数 C_h^y 。 G^{-1} 是矩阵 G 的逆矩阵,和用于 x 的计算相同。图 10 说明了这一情况。最初是对群序号 $u=0$ 进行的。另外,近似次数 M 从 1 开始。这样,对于轮廓点串 $u=0$ 可以求出近似 (x_k^u, y_k^u) 的 $S_x(t), S_y(t)$ 的系数 C_h^x, C_h^y 。计算它们与 (x_k^u, y_k^u) 点之差的平方和 Q , 则为

$$Q = \sum_{k=0}^{N-1} \{ S_x(t_k^u) - x_k^u \}^2 + \{ S_y(t_k^u) - y_k^u \}^2 \quad (29)$$

把 Q 值与预先确定的阈值 ϵ 相比较。 ϵ 是充分小的确定的正定数。在此取为 1 个象素的距离。如果 $Q > \epsilon$, 则近似就不够充分。因此,这时,就把次数 M 从 1 增加到 2。于是 C_h 数增加 1。

在此种状态下,反复同样的过程,如果还是 $Q > \epsilon$ 。就进一步把 M

加1。然后,反复同样的过程,提高近似程度,直至使 $Q < \varepsilon$ 。这样就确定了 M 和 $\{C_h^x\}$ 、 $\{C_h^y\}$ 。从而可以求近似函数 $S_x(t)$ 、 $S_y(t)$ 。当完成对轮廓点串群 $u=0$ 的计算时,对 $u=1$ 的群进行计算。以下,同样,当对某个 u 值决定了系数时,就对下一个 $u+1$ 的轮廓点串进行同样的计算。并且反复进行计算直至 $u=U-1$ 群为止。

下面,根据图48再次说明文字字体的近似机构的过程。设画面上输入“和”字。如果进行轮廓点串抽取,能得到3个轮廓点串群。左边的轮廓点串为 $u=0$,右边的“口”部分具有外轮廓点串 $u=1$ 和内轮廓点串 $u=2$ 。从画面左上方的点 $(0,0)$ 开始向右方向逐行扫描像素的值。把最先遇到轮廓线点记为 (t^0, x^0) 、 (t^0, y^0) ,以 t 为参量求轮廓点串的坐标。作为 t 的函数能够得到轮廓点串坐标 $x(t)$ 、 $y(t)$ 。若用2次光滑函数进行近似,则首先从 $M=1$ 出发。这是把全区间 $[0,1]$ 作为1个区间处理。3个基底函数 ψ_{-2} 、 ψ_{-1} 、 ψ 。在这个区间有值。适当地选择这些函数系数的3个值,能得到中间所示的近似曲线、放大后来看,可知与原数据的误差相当大。根据误差的平方和评价误差,近似不充分时,使 M 增加1。在下一步,示出了 $m=2$ 的例子。基底函数增加为4。最后一步示出了 $M=10$ 的情况。基底函数的个数为12。随着 M 的增大,能够精密地近似复杂的曲线。这里所说明的是用于求曲率的近似,而在后面所述的数据压缩时的近似中,对各个区间进行这种操作。

以上是数据近似机构A。

下面根据图 49 进一步说明词符—插图情况的近似机构的过程。设画面上输入香烟的概略图的图形。如果抽取轮廓点串,能得到 2 个轮廓点串群。左边长方形的轮廓点串为 $u=1$, 右边烟的部分具有轮廓点串 $u=0$ 。从画面左上方的点 $(0,0)$ 开始向右方逐行扫描象素的值。把最初遇到轮廓线的点记为 (t_0^0, x_0^0) 、 (t_0^0, y_0^0) , 以 t 为参量求轮廓点串的坐标。能够得到作为 t 的函数的轮廓点串坐标 $x(t)$ 、 $y(t)$ 。

其后的过程与图 48 文字字体的场合相同。

E. 曲率演算机构

由于对所有的轮廓点串群求出了近似函数, 故通过对这些函数求 2 阶微分, 可以求出各轮廓点串群和各点的曲率。图 11 示出了曲率计算的流程。轮廓点串群 u 的第 k 个点 (x_k^u, y_k^u) 的曲率 $K(t_k^u)$ 可以利用公式

$$K(t_k^u) = \{ S_x'(t_k^u) S_y''(t_k^u) - S_x''(t_k^u) S_y'(t_k^u) \} / \{ S_x'(t_k^u)^2 + S_y'(t_k^u)^2 \}^{3/2} \quad (30)$$

进行计算。从最初 $u=0$ 的轮廓点串群的 $k=0$ 的点开始计算。对每个点进行这一计算。即, 当完成对第 k 点的计算时, 接着就对第 $k+1$ 点进行同样的计算。当一个轮廓点串群的计算结束后, 就移到下一个轮廓点串。并且, 对所有轮廓点串的全部点求曲率。

E'. 近似曲率存储装置

这是把在前一步所求的曲率 $K(t_k^u)$ 按每个点(轮廓点串群 u 、第 k 点)进行存储的装置。

F. 正圆抽出机构

这是根据近似曲率判断某个轮廓点串是否为正圆、并抽出正圆的机构。文字、(词符—插图中有不少正圆部分。就是利用字文、词符—插图的这种性质。然而,这里所说的正圆是对轮廓线而言的,所以,指的是孤立的正圆。对于正圆部分与其它的直线,曲线交差接触的情况,不能抽出正圆。对于图 33 的情况,通过抽出正圆可以去除加在右肩上的圆。对于图 44 的圆形图表的情况,亦因为最外周的圆是孤立的圆,所以通过正圆抽取被抽出后,以后就可以从近似计算中除去。

如果抽出正圆就有如下优点。其一,原来是正圆的图形,即使因噪声产生一些畸变,由于也按正圆变为数据化处理,所以能够非常正确地决定完全除去噪声的形状。还有,由于圆只用半径和圆心的坐标就能够指定,故在数据压缩这一点上是极其有效的。

所谓正圆是其轮廓点串中各点的曲率全部相等的图形。利用这个性质可以抽出正圆。图 12 中说明了其流程。首先读入关于曲率 $K(t_k^u)$ 的数据,然后,检查各轮廓点串群 u 是否构成圆。求关于轮廓点串群 u 的平均曲率 k^u 。这就是简单地对曲率求平均值。群序号 u 应该加括号,但是,因为括号不能加在标注上,所以只写作 u 。在图面上添标有括号。

$$K^u = \{ \sum_{k=0}^{N-1} K(t_k^u) \} / N \quad (31)$$

这就是平均曲率。预先确定曲率的上限 ε_1 ,若平均曲率大于 ε_1 ,就认

为该轮廓点串 u 群不是正圆。

$$|K^u - 1| > \varepsilon_1 \quad (32)$$

是包含曲率极大点的情况，与不是圆又不能微分的点和拐点等相对应。这时，将群 u 表为不是正圆的表示 $\text{Circle}(u)=0$ 。

与此相反，若平均曲率小于 ε_1 ，则群 u 有可能是正圆。因此，从 $k=0$ 的点开始求点附近的曲率 $K(t_k^u)$ 与平均曲率 K^u 之差的绝对值，并检查这个值是否小于一定的阈值 ε_2 。如果

$$|K^u - K(t_k^u)| < \varepsilon_2 \quad (33)$$

则有可能是圆的一部分。于是，对下一个点 $k=1$ 的曲率同样也求与平均曲率之差的绝对值并与 ε_2 相比较。这样，依次对群 u 的所有点进行比较。只要这个不等式成立，这些点就位于圆弧上。若在某点曲率和平均曲率之差大于 ε_2 ，则该点就不再位于圆弧上，包含该点的轮廓点串也就不形成正圆。这时，就对该群标以不是正圆的表示 $\text{Circle}(u)=0$ 。

如果群内所有点的曲率与平均曲率之差都小于 ε_2 ，则该轮廓点串就是正圆。因此，就标以是正圆的表示 $\text{Circle}(u)=1$ 。对于所有的轮廓点串群从 $0 \sim U-1$ 进行这样正圆的抽取。最后，总起来从群 $u=0$ 开始对群检查是否 $\text{Circle}(u)=0$ 。如果所有的轮廓点串群 u 都是 $\text{Circle}(u)=0$ ，则在这些文字字体、词符—插图中设有一个正圆。若有 $\text{Circle}(u)=1$ ，则该群是正圆，可以求出该圆的圆心坐标 (X, Y) 和半径 r ，并把它们输入正圆存储装置 G 存储起来。

G. 正圆存储装置

这是用于在上一阶段所求出的正圆的圆心坐标和半径 r 的装置。由此,可用 3 个值记述群 u 的数据。由于以文字字体、词符一插图对象,所以全部轮廓点串是闭合曲线。对于单重正圆的情况,由于是内部整体用黑象素涂满的圆,所以是孤立的圆点。对于 2 重正圆的情况,与 2 重圆之间用黑象素涂满的圆圈(はひふへほ)等相对应。

H. 连接点位置抽取机构

所谓连接点,就是直线和直线、曲线和曲线、直线和曲线等的连接处。由于是不同斜率的线相接故称之为连接点。对文字字体、词符一插图进行函数近似时,连接点起着极其重要的作用。因此,对于连接点的意义。连接点的作用、连接点的抽取等等必须具有明确的观念。本发明通过正确适当地决定连接点,可以使文字字体、词符一插保持高品质,同时使数据量减到最少。

图 13 示出连接点抽取过程的概略情况。输入文字,词符一插图后,把它们作为图象,形成 2 值图象后,抽取构成外周的轮廓点串。由此,对得到的各轮廓点串群抽出连接点。

首先,抽取已知道的连接点。这可以按曲率大的点决定。这是临时连接点,还不是最佳的连接点。还有,仅以这样的点不能找出所有的连接点。反之,也包含有不必要的连接点。

因此,必须对连接点的指定做修正。然后进行直角部分连接点

的抽取、这将在后面叙述,这是为了在文字、词符一插图的二条直线交差时使交差点的接续平滑。所谓直线的连接点就是始点和终点,是细线,是在曲率大的条件下追加不显现的连接点情况。对于文字的情况几乎不必要,但对于词符一插图,常常是必要的。

反之,由于有不必要的连接点,所以必须除去这些点。其中之一就是位于直线中的连接点。如果连接点位于直线的中间,则当到直线的距离比较小时,即使除去该点也能够保持为直线。因为它是由噪声产生的连接点,所以除去。另一种是圆弧的连接点。本来应是一个圆弧的曲线,有时会因为连接点多而分离为二个圆弧来表现。这也要除去中间的连接点将圆弧合并成一个圆弧。

并且,当确定了连接点时,就用分段多项式近似各区间。这种分段多项式的近似和前面在数据近似机构A中所述的相同,但是,前面所述的近似区间遍及所有轮廓点串群。现在并不是那样,而是在每个连接点进行分段多项式近似。

首先,对临时连接点的抽取进行说明。在连接点处,由于斜率变化,故曲率也大。在接着进行的函数近似时,连接点是决定函数的定义域的。连接点和连接点之间是直线或斜率平滑地变化的曲线。连接点间的近似函数可以用简单的形式表现。因此,只要求出连接点的坐标和连接点间的近似函数,就可以求出该文字字体的全部数据。

但是,若连接点的选择不合适,连接点之间过于分离,则由于相邻连接点之间不能用简单形状的近似函数表现,反而增加了数据

量。反之,若连接点过多,则由于连接点坐标的数据增加,故不能充分进行数据压缩。因此,连接点的选择是很重要的。根据图 14 的流程,说明连接点位置的抽取方法。前面已对全部轮廓点串求出了近似曲率,故可以利用它们。

从近似曲率存储装置 E' 读入曲率,该曲率是 $\{K(t_k^u)\}$ 。根据这些数据顺序检查 u 群中所有点 $k=0\sim N-1$ 的曲率是否大于某个值 δ 。即,

$$|K(t_k^u)| > \delta \quad (34)$$

所谓连接点,就是斜率是不连续的,并且是曲率大到某种程度的点,所以利用这种方法可以求出。 δ 的值可以根据不同目的适当决定。

而且,由于构成正圆的轮廓点串已被除去,所以,在圆弧的一部分用这种演算不会出现大曲率的点(不可能是连接点)。

因此,把曲率超过 δ 的点作为临时连接点。并且,把这个坐标 (x_k^u, y_k^u) 存储到连接点位置存储装置 I 中。设临时连接点的序号为 i。因此,第 i 个临时连接点可以写为 $d_i(x_i^u, y_i^u)$ 。轮廓点串中的序号 k 置换为连接点序号 i。对于各点 k 进行这样的比较和连接点的抽取。对群 u 进行这样的处理后,对群 u+1 进行同样的处理。这样就能得到各群内所有临时连接点的坐标。

下面,直觉地说明连接点。图 15 是平假名“あ”和片假名“ホ”的原文字。把它们输入到图象处理装置进行轮廓点串抽取,抽出连接

点的图形示于图 16。标“x”号的地方是连接点。为了无差错地抽取这些连接点,之后要进行一些计算。

另外,图 17 是乐谱音符的例子。左右有 2 条竖线,上面有 2 条倾斜地连接竖线的线。竖线的下方连接着 2 个倾斜的椭圆。(a)是原图,是黑白图形。(b)是进行轮廓点串抽取、进而抽取出连接点后的图形。标有“x”号的点是连接点。在这个例中,有外侧轮廓线 1 和内侧轮廓线 2 条轮廓线。

外轮廓线 1 具有连接点 ハ~ヌ。ハ~ニ、ニ~ホ、へ~ト、ト~チ、チ~リ、ヌ~ハ 是直线。ホ~へ 和 リ~ヌ 是自由曲线。内轮廓线 2 中,ル~ヲ,カ~ワ 是直线。ヲ~ワ、カ~ル 是圆弧或自由曲线。这个例中直线部分多,能够把直线按直线的形式正确地抽取出来。当然也包括自由曲线部分。

图 18 是半导体二极管的略图。大的圆形中有三角和线段,有从这些线段中央横向贯穿的直线。轮廓线有外侧圆形的轮廓线 3 和内侧轮廓线 4、5 共有 3 条。外轮廓线 3 的连接点有 8 个,仅对上半部分标注了符号。ツ~ネ 是短线段,ネ~ナ 是半圆弧,ナ~ラ 是短线段,ツ~フ 是短圆弧或自由曲线、内轮廓线 4、5 是对称的,所以对 4 进行说明。ム~ウ 是线段,ウ~中 是线段,中~ノ 似乎也是线段,但在ノ 的附近为自由曲线。ノ~才 是线段,才~ク 也是线段,ク~ヤ 是线段,ヤ~マ 是线段,マ~ム 是半圆弧。这个图形直线部分也较多,其次圆弧也较多。在几何学中线段规定了两端,直线是没有两端的图

形,但在本说明书中把线段和半直线也称为直线。

当然不是说无论什么样的词符一插图都是直线多,也有圆弧多的图形(图 45、图 56)、自由曲线多的图形(图 51、图 53、图 55、图 57)和多种多样的图形。词符一插图是由设计者的自由想象创造出来的,所以无法决定其倾向。

1. 连接点位置存储装置

这是存储由前述操作求出的临时连接点的序号和坐标 $d_i(x_i^u, y_i^u)$

的装置。称为临时连接点是正确的。关于连接点,有前述以大曲率所选择的临时连接点和后面所述的候补连接点以及最佳连接点 3 种。不能将它们混同。

J. 最佳连接点抽取机构

前面所求的是仅以曲率的大小判断的临时接合点。这些有的并不是实际的连接点。在光学读取文字字体。词符一插图的图象中,由于透镜及玻璃对光的反射和散射以及外界干扰光等存在噪声,所以实际上有时 2 条直线、2 条曲线的连接点的曲率会变小。因此,按照前面的处理,有时可能没有以连接点留下来。相反,由于噪声的影响,尽管是直线、曲线的中间点,有时也因曲率变大而视为连接点。

因而,必须找出最佳连接点。最佳连接点应该位于用前述方法求出的候补连接点附近。所谓最佳连接点,如前所述是直线和直线的交点及拐点等,所以,是文字、词符一插图所固有的。由于噪声的影响,仅用曲率的大小不可能正确地找到它们。

图 19 示出了最佳连接点抽出机构的流程。从连接点位置存储装置 I 读入临时连接点 $d_i(x_i^y, y_i^u)_{i=0}^{l-1}$ 。这里, u 是轮廓点串的群序号, i 是该群中临时连接点的序号, l 是该群中临时连接点数。称为临时连接点是因为这不是最终的连接点。本发明的显著特征也在于此, 对于连接点的位置反复推敲这一点, 在本发明中是很重要的。

考虑临时连接点附近的几个称为候补连接点的点。设它们为 λ^i 。加上坐标写为 $\lambda^i(x_{i0}, y_{i0})$ 。 i 是临时连接点的序号, 应该放在括号内作为上标, 但因括号不能形成 $1/4$ 角, 所以省略。在图面正确地书写为 (i) 的标注。省略了群 u 的表示。当然是群 u 中第 i 个临时连接点附近的候补连接点。设候补连接点的数为 0 。这对任何临时连接点都是相同的数。图 20 示出了这种表示方法。

中央的 x_i 是根据曲率所选择的临时连接点, 其前后 α 个是候补连接点。不过, 这里写为 x_i 的就是 $d_i(x_i^u, y_i^u)$, 包括 x 坐标和 y 坐标 2 个坐标的意思。虽然可以写为 d_i , 但这里为了清楚地表示是位置坐标的意思, 所以简单地写为 x_i 。后面出现的附近连接点的表记也是同样的。

候补连接点有 $o=0, 1, \dots, 0-1$ 个。这由 $d_{i+\alpha}$ 或 $d_{i-\alpha}$ 求出的。这是把临时连接点前后 α 个轮廓点串作为候补连接点的结果。由于临时连接点也和它对应, 故 $0=2\alpha+1$ 。其中存在一个最佳连接点。根据临时连接点与最佳连接点偏离的大小的评价来决定, 偏离大时必须增大 α 。但是, 如果增大 α , 后面的计算量就要增加。因此, 应该兼顾两

方面决定适当的数。

接着,根据这个数抽出近旁连接点 $\{d_i(x_i'', y_i'')\}$ 。近旁连接点是位于以临时连接点 x_i 为中心、半径为 ρ 的圆内的临时连接点,是由连续的轮廓点串所联系的点以及位于由包含这个临时连接点的轮廓点串所包围的轮廓点串内的点。例如,对于“ ”这一文字的情况,右下角的○的内侧轮廓点串也形成外侧临时连接点的近旁连接点,对于图 33 的词符一插图的情况,如最下段所示的包含在○内部的临时连接点 ψ 、 γ 、 τ 成为近旁连接点。

设对近旁连接点标的序号为 j 。成为先前考察对象的临时连接点 x_i 以 i 为序号,以示区别。所谓位于半径 ρ 内,可以用条件

$$\{(x_i - x_j)^2 + (y_i - y_j)^2\}^{1/2} < \rho \quad (35)$$

表示。需要“由轮廓点串联系着”这个条件,是因为最佳连接点的抽出,在线与线的交差处为了使连接点之间光滑地相联系。没有由轮廓点串联系着的连接点之间的相对关系,无论如何都可以。另外,由轮廓点串(黑象素的连结)联系的这一条件可以利用 x_j 与 x_i 是否在一个轮廓点串群中简单地进行检查。

近旁连接点应该正确地写为 $x_j(x_j'', y_j'')$,但由于太冗长,故以后简单地写为 x_j 。这不单纯只是 x 坐标,也包含 y 坐标的意思。

设近旁连接点的数为 J 。 J 有时为 0 ,有时大于 1 。 $J=0$ 时,没有近旁连接点。 J 大于 1 时,有几个近旁连接点。对于两种情况的处理不同。下面,分为 $J \geq 1$ 和 $J < 1$ (即 $J=0$)的两种情况进行说明。

$J \geq 1$ 时,进一步分为两种情况。轮廓点串的跟踪,在外部有轮廓点串时,顺时针旋转,位于内部时,逆时针旋转。所以,有的近旁连接点 x_j 位于成为考察对象的临时连接点 x_i 之后,有的近旁连接点 x_j 位于 x_i 之前。必须区别二者。

位于用曲率选择的临时连接点 x_i 之前的近旁连接点 x_j 称为输出近旁,如图 21 所示。位于临时连接点 x_i 之后的近旁连接点 x_j 称为输入近旁。即,位于从临时连接点 x_i 向轮廓点串扫描方向发出的向量走出去的前端的近旁连接点是输出近旁。相反,从近旁连接点向扫描方向发出的向量进来的临时连接点是输入近旁。关于输出近旁和输入近旁的判断在后面进行说明。

情况 1. ($J \geq 1$, 输出近旁的情况)

首先,根据图 21 考虑输出近旁的近旁连接点。可以有几种考虑方法。这里,考虑 4 个临时连接点的近旁。并且,根据联系这 4 个点的曲线为最光滑的这一条件,决定临时连接点附近的候补连接点 λ^i 。($x_{\lambda_0}, y_{\lambda_0}$) 中的最佳点。

图 21 中,最初根据曲率所选定的临时连接点是 x_i (C 的附近)。近旁连接点是 x_j (B 的附近) 对于输出近旁的情况,考虑临时连接点的下一个临时连接点 x_{i+1} (D 的附近) 和近旁连接点 x_j 之前的一个临时连接点 x_{j-1} (A 的附近)。连接这些点,使通过这 4 个点的曲线最光滑,这 4 个连接点内,两侧的 A、D 固定,仅移动中间的 B、C 求适宜系数。B、C 点具有 2α 个候补连接点。于是,进行 $(2\alpha+)^2$ 个组合的

分段多项式近似。其中,最光滑地联结4个点的近似给出最高的适宜系数。

有几种方法表现光滑这一条件,但是,此处采用根据曲线的拐点最少这一条件进行评价。以逐次近似为例,用 t 表示近似的次数。

曲线的端点 A 是 x_{j-1} 中某个候补连接点 $\lambda^{j-1}(x_{j-1}, y_{j-1})$ 。不过,这里候补连接点的角标是 l' ,但因“'”号不能用于 $1/4$ 角格式,故在此写为 l 。实际上是 l' 。

中间点 B 是近旁连接点 x_j 的某个候补连接点 $\lambda^i(x_{j1}, y_{j1})$ 。候补连接点的序号用 l 表示。 l 是一个参数,改变 l 要进行若干次相同的计算。

中间点 c 是原来的临时连接点 x_i 中某个候补连接点 $\lambda_0^i(x_{i0}, y_{i0})$ 。候补连接点的序号用 o 表示。 o 是一个参数,改变它要反复若干次相同的计算。

曲线的另一个端点 D 是 x_{i+1} 中某个候补连接点 $\lambda_0^{i+1}(x_{i0}, y_{i0})$ 。不过,这里候补连接点的角标是 o' ,但是由于“'”号不能用于 $1/4$ 角格式,故在此仅写为 o 。实际上是如图 21 的 o' 。

以上是计算的顺序,但是,由于两端的 A、D 点是固定点(在第 t 次近似中),所以,对于中间点 B 的候补连接点 $X_i^i(x_{i1}, y_{i1})$ 、中间点 c 的候补连接点 $\lambda_0^i(x_{i0}, y_{i0})$ 逐次改变参数 l 、 o 反复进行计算。在这个计算中把作为连接点的适宜性视为概率函数进行计算。概率函数具有由曲线光滑性决定的系数 $r_{ij}(ol)$ 。可以把这个系数称为适宜

系数。i 和 j 是临时连接点 x_i 和近旁连接点 x_j 的符号。

o 和 l 是这些临时连接点所属的候补连接点的序号。由于系数是对每个参数 o、l 决定的值，因而必须有这些角标。在曲线 ABCD 足够光滑时为 1，不光滑时小于 1。所以可以成为决定曲线适宜性的参数。

图 22 是表示存在近旁连接点时 ($J \geq 1$) 的适宜系数 $r_{ij}(ol)$ 决定法的流程图。首先指定参数的范围。

i 从 0 变到 $l-1$ ，对于根据曲率所求的临时连接点 x_i 就是从 0 到 $l-1$ 反复进行这种计算。

o 从 0 变到 $2x$ ，就是在作为对象的临时连接点 x_i 中有 $2\alpha+1$ 个候补连接点，故对它们一个一个地进行计算。

i 从 0 变到 $J-1$ ；就是有多个近旁连接点 x_j 时，要全部对它们反复进行这种计算。因为 J 大于 1，所以必须如此处理。

l 从 0 变到 2α ，就是在邻近连接点 x_j 中有 $2\alpha+1$ 个候补连接点，故对它们一个一个地进行计算。

虽然对 i、o、j、l 各参数进行这样的计算，但这里，书写的顺序不同，并不是越靠前面的参数构成的循环越大。但是，根据以后的说明，可以唯一地给这些参数的变化。

首先，根据与数据近似相同的方法，用光滑函数近似点 A、B、C、D。即，用分段多项式进行近似。近似函数的次数为 2 次就够了。在前述的公式中，取 $m=2$ ，则

$$S_x(t) = \sum_{k=-2}^{M+2} C_k x \phi_k(t) \quad (36)$$

$$S_y(t) = \sum_{k=-2}^{M+2} \{-M+2 C_k x \phi_k(t)\} \quad (37)$$

$$\phi_k(t) = 3(T/M)^{-2} \sum_{q=0}^3 (-1)^q \{t - (k+q)(T/M)\}^2 / \{q!(3-q)!\} \quad (38)$$

$$k = -2, -1, 0, 1, 2, \dots, M+2 \quad (39)$$

这里,参量 t 是表示连接量的 t 。与前面所讲的表示近似次数的 t 不同。由于字符数不够,故用同一个 t 表现不同的量。

最初的数据近似时,对全部轮廓点串确定范围 T ,对全部轮廓点串给出一个近似。所以,必须参照所有轮廓点串的坐标数据。

然而,此处不是那样,只要联结 4 点 ABCD 就可以了。只需要 4 点的坐标,不需要 4 点间众多的轮廓点串的坐标。由于只要联结 4 点就可以了,故用分段多项式近似 AD。区间 T 与曲线 AD 对应。虽然进行图 10 所示的计算,但因点数明显减少,故这种近似很简单。用最小二乘法确定系数时也相同。利用这样的近似可以作通过 4 点的近似曲线 ABCD。

下面,计算曲线 ABCD 的各轮廓点处的曲率。由于有近似函数 $S_x(t)$ 、 $S_y(t)$,所以有

$$K(t_w^u) = \{S_x'(t_w^u) S_y''(t_w^u) - S_x''(t_w^u) S_y'(t_w^u)\} / \{S_x'(t_w^u)^2 + S_y'(t_w^u)^2\}^{3/2} \quad (40)$$

式中, $w=A, B, C, D$

这不用遍及整个曲线 AD 求曲率,只计算 4 点就行,因而很简单。

下面,根据曲率的反转次数求拐点数 n_c 。拐点数当然小于 3。图 23 是曲线的大致形状和拐点数 n_c 的示例图。用“X”号表示拐点。(a)是拐点数为 3 的情况,曲折明显。(b)是拐点数为 2 的情况,(c)是

拐点数为 1 的情况, (d) 是拐点数为 0 的情况。这是最光滑地联结 4 点的情况。采用这个形式是所希望的。这时, 规定 $r_{ij}(ol)$ 为 1。例如, 用拐点数 n_c , 可以按下式决定:

$$r_{ij}(ol) = \min \{ 1/c_2 (n_c - c_1), 1 \} \quad (41)$$

(c_1, c_2 为常数)

在本例中, 拐点数为 3 时, $r_{ij}(ol) = 1/3$ 。这是取 $c_1 = -1$ 、取 $c_2 = 3/4$ 的结果。于是, $n_c = 2$ 时为 $4/9$, $n_c = 1$ 时为 $2/3$, $n_c = 0$ 时为 1。当然可以适当决定这些常数。1 和 min 的演算是为了防止在拐点数过少时计算值过大, 但是, 并不是说 $r_{ij}(ol)$ 的上限必须为 1。也可以仅用公式 $r_{ij}(ol) = 1/c_2(n_c - c_1)$ 计算。

重复若干次这种计算。用 t 表现近似计算次数。这是求概率的计算。关于近似计算将在后面将叙述。这里, 仅就适宜系数 $r_{ij}(kl)$ 决定进行了说明。

情况 2, ($J < 1$ 时, 没有近旁连接点)

图 24 示出了 $J > 1$, 即不存在近旁连接点时的处理流程。图 25 示出了这种情况下连接点位置的例子。这是临时连接点 X_i 之前的临时连接点 X_{i-1} 和其后的临时连接点 X_{i+1} 的 3 点联结的问题。图 24 的流程中开始也定义了参数的范围。

i 从 $0 \sim I-1$, 是临时连接点的序号, 对所有的临时连接点进行演算。 k 是临时连接点 X_i 的候补连接点 $\lambda_k^i (X_{ik}, y_{ik})$ 的序号。1 是 X_i

之前的临时连接点 X_{i-1} 的候补连接点 $\lambda_1^{(i)}(X_{i-1}, y_{i-1})$ 的序号。对所有的候补连接点进行计算。j 是 $i-1 \sim i-1$, 即, j 仅是 $j=i-1$ 。

设各个临时连接点的临时连接点 X_{i-1}, X_i, X_{i+1} 的候补连接点为 A、B、C。点 A 是 $\lambda_1^{(i)}(X_{i-1}, y_{i-1})$, l 是参数。点 B 是 $\lambda_k^{(i)}(X_{ik}, y_{ik})$, 参数是 k。点 C 是 X_{i+1} 的候补连接点 $\lambda_h^{(i)}(X_{ih}, y_{ih})$, 表示 $\max_h (p_{i+1h}^{i-1})$, \max_h 表示参数 h 中的最大者。

用光滑函数插值这 3 个点。虽然用图 10 所示的方法进行, 但只有 3 个点, 故极其简单。然而这也要提高近似程度, 用 t 表示近似次数。并且, 近似到最小二乘法的误差达到 ϵ_{\max} 。这时, 设近似的因次数为 M。 $r_{ij}(ol)$ 由 $\min\{1/(c_3m + c_4\epsilon), 1\}$ 决定。 ϵ 是最终误差, M 是最终因次。

即, 要求出能以尽可能低的因次, 很小的误差进行近似的点。这样的点可以在分别属于各临时连接点的候补连接点内给出最光滑的曲线。t 是近似次数, 从 0 开始上升。与前面说明的 $J \geq 1$ 时相同, t 从 0—4, 在 4 次停止近似。

情况 3, ($J \geq 1$, 输入近旁的情况)

图 21 中所示的是 $J \geq 1$ 时对于临时连接点 X_i 的近旁连接点 X_j 为输出近旁的情况。 $J \geq 1$ 时, 近旁连接点 X_j 为输入近旁的情况也进行同样的计算, 但这时, 4 个点的取法稍有不同。

图 26 地出了输入近旁情况下对于近旁连接点 X_j 取 4 个点的方法。近旁连接点 X_j 位于以 X_i 为中心的半径 ρ 内。而且对于扫描方

向,该点位于原来的临时连接点 X_i 的前面。4 个点作为对象的临时连接点 X_i 之前的临时连接点 X_{i-1} 和近旁连接点 X_j 的下一个临时连接点 X_{j+1} 。从位于这些点附近的 2α 个候补连接点中取出 4 个点。

A 是 X_{i+1} 的候补连接点 $\lambda_1^i(X_{j+1}, y_{j+1})$ 之一。这里的 1 是加“1”的,但由于它能用于 $1/4$ 角的格式,故省去“1”号。该点是用前面的概率计算给出的最大值决定的,是固定值。

B 是近旁连接点 X_j 的候补连接点 $\lambda_1^j(X_{11}, y_{11})$ 之一。

C 是作为对象的临时连接点 X_i 的候补连接点 $\lambda_0^i(X_{i0}, y_{i0})$ 之一。

D 是 X_{i-1} 的候补连接点 $\lambda_0^{i-1}(X_{i-10}, y_{i-10})$ 之一。该点也是利用前面的概率计算中给出的最大值决定的,是固定值。如图所示,实际上是 o' ,但是由于“1”不能形成 $1/4$ 角,故这里省略。

以上确定了 4 个点。以后的计算与前面说明过的情况 1 相同。

[输出近旁和输入近旁决定方法]

对于作为对象的临时连接点 X_i 的近旁连接点 X_j , 必须区别输出近旁和输入近旁。通过沿轮廓点串扫描方向的 2 个临时连接点的向量指向近旁连接点时,称为输出近旁。相反,通过沿轮廓点串扫描方向的 2 个的近旁连接点的向量指向临时连接点时称为输入近旁。

这是难于理解的概念。连接点具有 2 个近旁连接点,必须与二者光滑地接续。为此,必须区别这两个近旁连接点。

图 27 是决定输入、输出近旁连接点的流程。输入全部候补连接点 $\lambda_0^i(X_{i0}, y_{i0})$ 。

i 是计算对象即临时连接点 X_i 的序号。对于一个轮廓点串群 u 有 l 个序号 ($0 \sim l-1$), 从 $i=0$ 开始。 j 是近旁连接点的序号, 当然不是从一开始就知道是否为近旁的。开始, 全部为临时连接点 X_j 。 o 是包含在临时连接点附近 $2a$ 个之内的候补连接点的序号。图中, o 上加有上横线, 但在说明书中未加横线, 省略了仅固定临时连接点 X_i 的 i , 决定近旁连接点。

$d_j(x_j^u, y_j^u)$ 是第 j 个临时连接点的坐标, 为简单起见写为 d_j 。检查 d_j 是否在以 $\lambda_0^i(X_{i0}, y_{i0})$ 。(为简单起见, 写为 λ_0^i) 为中心的半径 ρ 内。这里(35)式中已示出的条件, 现在再次记述如下:

$$\{ (x_{i0} - x_j)^2 + (y_{i0} - y_j)^2 \}^{1/2} < \rho \quad (42)$$

(35) 直接使用了临时连接点的坐标 X_i , 这里使用 X_i 的候补连接点的坐标。如果 d_j 位于半径 ρ 内, 则该点有可能是近旁连接点。其次, 检查 d_j 和 λ_0^i 是否位于同一个轮廓点串。如果位于不同的轮廓点串, 它们就不是近旁连接点。

如果位于同一个轮廓点串, d_j 就是 λ_0^i 所属的临时连接点 X_i 的近旁连接点。用 d_j 表示 X_i 。在作为对象的临时连接点 X_i , 可以用 d_{i-b} 表示某一常数 b 前面的点的坐标(不一定是候补连接点)。向量 d_{i-b} 的方向在临时连接点沿扫描方向。但 b 在图中是 β , 由于 β 不能成为 $1/4$ 角格式, 故在此写为 b 。 b 应该改读为 β 。另外, 由于在文字上面不能加 \rightarrow 号故在这里写为标量。

一方面,在近旁连接点 X_j 可以把位于其后一定数 b 的点的坐标
 写为 d_{j+b} 。向量 $d_j d_{j+b}$ 的方向在近旁连接点之后沿扫描的方向。计
 算这些向量的标积。标积很小时,这些向量几乎正交。即,如果

$$\text{向量 } d_i d_{i+b} \cdot \text{向量 } d_j d_{j+b} < \epsilon \quad (43)$$

则两者几乎正交。式中, ϵ 是适当小的数值。这时, d_j 是输入近旁。
 于是,把它作为 d_j^{input} (i) 进行存储。

另一方面,在近旁连接点 X_j ,可以把位于其前面一定数 b 的点
 的坐标写为 d_{j-b} 。向量 $d_j d_{j-b}$ 的方向在近旁连接点之前沿扫描的方
 向。计算这些向量的标积。标积很小时,这些向量几乎正交。即,如果

$$\text{向量 } d_j d_{j+b} \cdot \text{向量 } d_{j-b} d_j < \epsilon \quad (44)$$

二者几乎正交。式中, ϵ 是适当小的数值。这时, d_j 是输入近旁。于
 是,把它作为 d_j^{output} (i) 进行存储。

以上所说明的是利用向量标积的情况,也可以利用向量的矢
 积。使用矢积时,可以找到平行的向量,即

$$|\text{向量 } d_{i-b} d_i \times \text{向量 } d_{j-b} d_j| < \epsilon \quad (45)$$

$$|\text{向量 } d_i d_{i+b} \times \text{向量 } d_j d_{j+b}| < \epsilon \quad (46)$$

等等。另外,若用以上方法还不能知道输出近旁和输入近旁的区别,
 则计算标积

$$\text{向量 } d_{i-b} d_i \cdot \text{向量 } d_{j-b} d_j < \epsilon \quad (47)$$

$$\text{向量 } d_i d_{i+b} \cdot \text{向量 } d_j d_{j+b} < \epsilon \quad (48)$$

确定这些标积的正负后,就能区别输入、输出近旁。

[概率的计算]

为了求最佳连接点,对候补连接点进行以下的计算,以选择最佳点。图 28 是表示这一计算的流程。输入的是

$$\lambda_0^i (X_{i0}, Y_{i0}), d_j^{\text{input}}(i), d_j^{\text{output}}(i) \quad (49)$$

$\lambda_0^i (X_{i0}, Y_{i0})$ 是根据曲率所选定的临时连接点 X_i 附近的候补连接点。 0 是候补连接点的序号。 i 的意思是关于 X_i 的量。 $d_0^{\text{input}}(i)$ 是临时连接点 X_i 的近旁连接点中的输入近旁连接点的位置坐标。 $d_j^{\text{output}}(i)$ 是 X_i 的近旁连接点中的输出近旁连接点的位置坐标。

开始时先输入这些值。用 t 表示计算概率的次数。这不是当用最小二乘法决定的误差小到某种程度时中止计算,而是从开始就预先决定次数。例如取 4 次 ($t=0-4$)。当然根据目的不同也可以取 5 次以上。

定义概率变量 $p^t(0)$ 。这是位于临时连接点 X_i 附近的 $2\alpha+1$ (包括 X_i 自身) 个候补连接点内 0 的第 t 次概率。不是 t 次方而是第 t 次,所以应该加括号。但由于不能把括号加在标注上,所以只写为 t 。这里说的概率是作为最佳连接点的概率。在 $2\alpha+1$ 个候补连接点内,选择它的概率最高的点。每增加概率计算的次数 t ,对于连接点,就提高了更适宜的概率变量。这样,就可以决定概率变量。当然,可能有几种定义。在这里,按如下决定。考虑互补概率变量 $Q_i^t(0)$ 。这个变量也给出候补连接点 0 的第 t 次概率,这是输入近旁的贡献 $q_i^{\text{input}}(0)$

与输出近旁的贡献 $q_i^{input}(o)$ 之和。这些值也计入了 t 次,但此处省略。

$$Q_i^t(o) = q_i^{input}(o) + q_i^{output}(o) \quad (50)$$

$q_i^{input}(o)$ 和 $q_i^{output}(o)$ 定义如下:

$$q_i^{input}(o) = \max \{ r_{ij}(ol) \times P_j^t(l) \} \quad (51)$$

$$(\lambda_i^j \in d_j^{input}(i)) \quad (52)$$

式中, \max 的计算,在候补连接点 λ_i^j 属于 $d_j^{input}(i)$ 的候补连接点这一条件下取最大值。这里, l 是变化的参数。适宜系数 $r_{ij}(ol)$ 已在前面给出。同样,

$$q_i^{output}(o) = \max \{ r_{ij}(ol) \times P_j^t(l) \} \quad (53)$$

$$(\lambda_i^j \in d_j^{output}(i)) \quad (54)$$

\max 的计算,在候补连接点 λ_i^j 属于 $d_j^{output}(i)$ 的候补连接点这一条件下取最大值。

第 $t+1$ 次的概率变量是用第 t 次的概率变量、互补概率量等按如下定义的。

$$P_i^{t+1}(o) = P_i^t(o) Q_i^t(o) / \{ \sum P_i^t(o) Q_i^t(o) \} \quad (55)$$

(\sum 是对 λ_i^j 进行求和)。

在 $t=0-4$ 进行这个计算。但是,初始($t=0$)的概率变量不知道时不能计算。虽然初始概率不管怎样赋值都可以,但是此处赋予尽可能

接近 X_i 的较大值。因为一般认为最佳连接点位于临时连接点 X_i 的附近,如果提高 X_i 的初始概率,就可能提高概率变量的收敛速度。

图 30 中说明了初始值的设定。候补连接点 λ_0^i 有 $2\alpha+1$ 个。从 0 开始。 $o=\alpha$ 时,就是 X_i 本身。初始值 $p_i'(o)$ 在 $o=\alpha$ 时取最大值 k 。 $o=\alpha-1, \alpha+1$ 时取 $(k-2)$ 。以后,随着远离 α ,每次减少 2。在 $o=0, 2\alpha$ 时,初始值是 1。使用初始值时,初始概率 $p_i^{(0)}(o)$ 为

$$P_i^{(0)}(o) = P_i'(o) / \sum P_i'(o) \quad (56)$$

\sum 的运算对 o 进行求和。这样,由于给出了初始概率,所以,和适宜系数 $r_{ij}(k_0)$ 组合后,就可能计算 $t=1$ 时的 $q_i^{\text{input}}(o)$ 和 $q_i^{\text{output}}(o)$ 。作为它们的和,可以计算互补概率变量 $Q_i'(o)$,以及进而计算第 $t+1$ 次的概率变量 $p_i^t(o)$ 。反复进行这一计算直到 $t=4$ 。在 $t=4$ 时,具有最高概率变量的候补连接点就是最佳连接点。这里的输出是 $p_i^4(o)$

$i=0^{1-1}, o=0^{0-1}$, 示于图 31。

这样,就能根据位于由概率求出的临时连接点附近的候补连接点求出最佳连接点。开始时,先求临时连接点,进而把范围扩大到该点附近的轮廓点中,作为候补连接点,从中求出最佳连接点,这就是本发明的显著特征之一。这是基于本发明者发现,曲率极大的点作为连接点,不一定是最佳的。

[适当系数的其它定义]

在前述方法中,最佳连接点的条件是连接 4 个点的曲线的拐点

最少，即最光滑地接续。除了计算拐点外也可以直接采用以曲线的曲率为对象的方法：图 29 是表示这种根据其它条件求候补连接点适宜系数方法的流程图。

开始的 $J > 1$ 是存在近旁连接点的条件。这时，考虑输入近旁和输出近旁，考虑过 4 个点 A、B、C、D 的曲线（图 21，图 26），但是，这时不是求拐点而是求曲率。 K_{input} 是在输入近旁联结 4 点的曲线的最大曲率。所希望的是曲率变小的曲线。 K_{output} 是在输出近旁联结 4 点的曲线的最大曲率。这也是较小为好。为此取二者之和，求给出最小值的点。因此，定义适宜系数如下：

$$r_{ij}(0) = \min \{ r (K_{input} + K_{output})^{-1}, 1 \} \quad (57)$$

r 是常数。它是在最佳时曲率之和几乎等于 r 这一要求决定的。

图 31 表示反复计算概率变量，当 $t=4$ 时中止反复，求出所得到的概率变量 $P_i^4(0)$ 为最大的候补连接点的序号 o ，并把它作为前述临时连接点 x_i 所属的候补连接点内的最佳连接点 λ_o^i ，然后，把它输入最佳连接点存储装置 K 中。对轮廓点串群的所有临时连接点进行这样的操作，可以决定各类最佳连接点。

K. 最佳连接点存储装置

这是存储利用上述操作求出的全部最佳连接点的装置。如图 31 的最后一步所示，把最佳连接点作为 (x_i^u, y_i^u) 进行存储。 u 是轮廓点串的序号。 i 是群 u 中的连接点的序号。前面对临时连接点标以 X_i

的序号(也代表 y 坐标),虽然序号 i 相同,但没有关系。因为一个临时连接点对应一个最佳连接点,并且最佳连接点与前面所求的临时连接点相同,或者极为接近。

按照顺序说明了至此的操作,但是为了易于了解步骤,这里以图 32 所示的“**は**”字为例,简单回顾前面的操作。图 32 中有黑体的“**は**”字。用图象扫描器读入,作为图象。在经过图象处理后,进行轮廓点串抽出。用黑框白底字表示。右肩上的圆圈利用正圆抽出除去,并用正圆存储装置进行存储。

余下的变为“**は**”。作为轮廓点串有独立的 3 个群。遍及它们的全长进行分段多项式近似,并且,计算曲率后求临时连接点。这些画在第 3 步中。临时连接点出现在线段的接头处。

在第 4 步用“**×**”号表示临时连接点。由于利用曲率进行选择,所以“**は**”的右下部分有时出现临时连接点。这时,要将该点追加为连接点。接着,求候补连接点。进而在“**は**”的下部的交差点部分抽出近旁连接点。根据以上的说明,至此为止的过程是清楚的。

其次,以图 33 所示的词符—插图为例,同样简单地回顾前面的操作。这是在图 18、图 50 的半导体二极管图形的右肩添加了多余的“**o**”的图形。这是为了说明正圆的处理。用图象扫描器读入这一原图,形成图象。经过图象处理后,进行轮廓点串的抽取。用黑框白底的词符—插图表示这一图形。有 5 个轮廓点串。右肩上的圆圈利用正圆抽取除去,存入正圆存储机构。由于去除了圆圈,所以,减少了 2

个内外同芯的轮廓点串。

余下的是由圆和三角、线段组合的图形。作为轮廓点串有3个独立的群。遍及轮廓点串的全长进行分段多项式近似。并且,计算曲率后求临时连接点。这些画在第3步中。临时连接点出现在线段的接头处。

在第4步中,用“×”号表示临时连接点。由于利用曲率进行选择,所以有时在本来应该成为连接点的地方2个线段也以钝角交叉时,不出现临时连接点。这时,要将该点追加为临时连接点。在本例中,由于利用曲率计算全部出现临时连接点,所以不需要追加连接点。接着,求候补连接点。然后,以圆和横直线的交差点、纵线段和直线的交差点等作为一定半径内的临时连接点,抽取近旁连接点。根据以上的说明,至此为止的过程是清楚的。

L. 连接点去除机构

在这里,是除去前面抽出的连接点内不必要的连接点。图34是表示该过程的流程图。从最佳连接点存储装置读入最终连接点,是 $\{\lambda, i\} = \{(x_i^u, y_i^u)\}_{i=0}^{I-1}$ u 是群序号, i 是群中连接点的序号。然后,除去直线近似中冗长的连接点。

进而,除去在圆弧近似中冗长的连接点。对于每个轮廓点串利用坐标 (x_i^u, y_i^u) 和序号 i 把最终连接点存储到存储装置中。

前面所求的连接点是用分段多项式近似轮廓线、求曲率、求临时连接点,再从候补连接点中求出的最终连接点。因此,应该几乎没有

不必要连接点,但是由于图象噪声的影响等,可能存在不必要的连接点。这就是直线的连接点和圆弧的连接点。

[直线的连接点的除去]

图 35 的轮廓点串的例子示出了在线段中有连接点的情况。利用 $(X_{i_4}^B, Y_{i_4}^B)$ 表现这个点的坐标。顺次评价这个连接点。存在直线的连接点大于 2 的连续区间时,在该区间内取 3 个连续的点串设 $i_4 = ns, ns+1, ns+2$ 。用直线连接两侧的 (X_{ns}^B, Y_{ns}^B) 和 (X_{ns+2}^B, Y_{ns+2}^B) 。设从中间的连接点 (X_{ns+1}^B, Y_{ns+1}^B) 向这条直线所做的垂线长度为 L_{ns+1} 。这个长度上预先确定的常数 K_7 时,

$$L_{ns+1} < K_7 \quad (58)$$

则除去这个连接点。图 36 示出了除去该连接点后连接两侧连接点的状态。

反之,垂线的足 L_{ns+1} 大于常数 K_7 时,

$$L_{ns+1} \geq K_7 \quad (59)$$

则保留该连接点。对于所有的连接点都进行这样的评价。

[圆弧的连接点除去]

在前面抽出的圆弧的连接点中,有时多个连接点连续。这时,即使除去中间的连接点,也能保证数据的品质时,就除去该连接点。判断除去的可能性按如下方法进行。如图 37 所示,用 $(X_{i_4}^B, Y_{i_4}^B)$ 表示圆弧的连接点。 i_4 是连续的序号。

如图 37 那样,设连续存在 3 个连接点 (X_{ns}^B, Y_{ns}^B) 、 $(X_{ns+1}^B,$

Y_{ns+1}^B)、 (X_{ns+2}^B, Y_{ns+2}^B) 。这时,考虑从 (X_{ns}^B, Y_{ns}^B) 开始的圆弧(μ)和从 (X_{ns+1}^B, Y_{ns+1}^B) 、开始的圆弧(ξ)。如图 38 所示,设两圆弧的半径为 r_{ns} 、 r_{ns+1} 。设它们的中心坐标分别为 (X_{ns}, Y_{ns}) 、 (X_{ns+1}, Y_{ns+1}) 。圆弧 μ, ξ 满足如下条件时,可以看作是单一的圆弧。

$$|r_{ns+1} - r_{ns}| < K_8 \quad (60)$$

$$\{(X_{ns+1} - X_{ns})^2 + (Y_{ns+1} - Y_{ns})^2\}^{1/2} < K_9 \quad (61)$$

即,如果中心座标几乎为同一点,且半径几乎相同时,则 2 个圆弧 μ, ξ 视为同一圆弧的一部分。这时,中间的连接点 (X_{ns+1}^B, Y_{ns+1}^B) 、可以作为不必要的点而除去。否则,就保留该连接点。

对全部圆弧的连接点顺次进行这种判断、除去、保留的作业。虽然应适当地决定半径差、中心差的阈值,但是例如定为 $K_8=1, K_9=2$ 就可能了。

利用以上的操作,可以除去直线和圆弧上的不必要的连接点。这对于压缩数据是有效的。不仅如此,而且由于还把应该是直线和圆弧的图形修正为所应有的形状,所以在提高数据的品质方面具有积极的意义。这样选定的连接点称为最终连接点。

M. 最终连接点存储装置

这是存储除去不必要的连接点后得到的最终连接点的装置。如

图 34 的最后一步所示,坐标 (x_i^u, y_i^u) 。但是,由于在图面中,为了与最佳连接点区别加有“'”号。由于“'”号不能形成 1/4 角格式,故在说明书中省去。实际上是加有“'”号的。

N. 数据近似机构

这是根据前面得到的轮廓点串、最终连接点、正圆等数据来近似数据的机构。是本发明的核心部分。从各个存储装置中输入的数据,如图 39 所示,分别是:

轮廓点串存储装置…轮廓点串 $\{ (x_k^u, y_k^u) \}_{k=0}^{N^u-1} =_0^{U-1}$

最终连接点存储装置…最终连接点 $\{ (x_i^u, y_i^u) \}_{i=0}^{I^u-1}$

正圆存储装置…圆 Circle(u)

轮廓点串的标就 u 是轮廓点串群的序号。 k 是该轮廓点串群内的轮廓点串的序号。 $N(u)$ (写为 N_u) 是群 u 内的轮廓点串数。 i 是群 u 内的最终连接点。只写为 i , 但实际上如图 39 所示是 i' 。由于“'”号不能形成 1/4 角格式,故在此省略。虽然可以得到最终连接点,但是把相邻的 2 个连接点之间(连接点间)进行直线、圆弧、自由曲线近似。在近似中,用参变量 t 把 x 分量用 $S_x(t)$ 表示, y 分量用 $S_y(t)$ 表现。

这和最初用参变量 t 表现整个轮廓点串的方法是相同的。但是现在区域成为连接点之间,所以 t 的范围和 t 与 $S_x(t)$ 、 $S_y(t)$ 的对应关系就与前面的不同。还有,从某连接点开始的区间是直线区间。还是圆弧区间或自由曲线区间,求出各点的曲率就知道了。

在存储了这些信息时,则从开始就能区别是直线还是圆弧或自由曲线。即,如果该区间的所有点的曲率为0,则这就是直线。另外,如果该区间的曲率为一定值,则这就是圆弧。进而,如果曲率变化,则这就是自由曲线。若能像这样区别区间,则决定近似计算的参数就比较简单了。

但是,反过来,即使没有存储曲率计算的结果,在这个阶段也能检查直线、圆弧及自由曲线的区别。区别之后也可以进行近似。

不论哪种情况,本发明都能适用。下面,对前者的情况进行说明。

[直线区间的近似]

对从直线的连接点开始的区间的近似进行说明。在连接点抽取阶段,已判断为直线。这时,x方向的近似曲线 $S_x(t)$ 成为联结始点 x_1 和终点 x_{n3} 的一次函数。y方向的近似曲线 $S_y(t)$ 成为联结始点y和终点 y_{n3} 的一次函数;这时,对于参变量x、y都成为一次函数。

参变量t与 $S_x(t)$ 、 $S_y(t)$ 的比例常数变为参数。但是,没有必要存储这个比例常数。这是因为,若是直线区间,则只要知道始点 (x_1, y_1) 和终点 (x_{n3}, y_{n3}) 就可以在这两点之间引直线。另外,由于终点 (x_{n3}, y_{n3}) 作为下一个区间的始点给出,故没有必要进行存储。

[圆弧区间的近似]

对从圆弧的连接点开始的区间的近似进行说明。这个区间在连接点抽取阶段,已判断为圆弧。表示圆弧的近似曲线 $S_x(t)$ 、 $S_y(t)$ 可用如下三角函数的线性组合表示。设观测区间为 $t \in [0, T]$,则 $S_x(t)$ 、 S_y

(t)可以表示为

$$s_x(t) = A_x \cos(2\pi t / (T/n_{arc})) + B_x \sin(2\pi t / (T/n_{arc})) + C_x \quad (62)$$

$$s_y(t) = A_y \cos(2\pi t / (T/n_{arc})) + B_y \sin(2\pi t / (T/n_{arc})) + C_y \quad (63)$$

n_{arc} 是圆弧与整个圆之比。即，是以360度除圆心角得到的值。例如，对于四分之一圆的情况， n_{arc} 等于1/4。所以， $2\pi n_{arc}$ 是该圆弧的圆心角。变量 $2\pi t / (T/n_{arc})$ 是从圆弧的始点到与参数 t 对应的点的圆心角。 (C_x, C_y) 是圆弧的中心的坐标。这时，如果关系式

$$A_x^2 + B_x^2 = A_y^2 + B_y^2 \quad (64)$$

$$B_y / A_y = B_x / A_x \quad (65)$$

成立，则近似函数就是圆弧。这时，规定圆弧的参数是函数各个系数 $A_x, B_x, A_y, B_y, C_x, C_y, n_{arc}$ 。如果从开始就知道该区间是圆弧，则根据始点、终点的坐标、曲率、中间一点的坐标可以唯一地决定这些参数。

[自由曲线的近似]

对既不是从直线的连接点，也不是从圆弧的连接点开始的区间自由曲线近似。虽然可用参变量 t 表示，但轮廓点串用 (x_{i3}^u, y_{i3}^u) 表示。使其与 t 相对应，可用参变量 t 表示为 $(t_{i3}^u, x_{i3}^u), (t_{i3}^u, y_{i3}^u)$ 。前面轮廓点串的下标是 k ，但是，由于这里用 k 作为区分区间的序号，故

代替 k 用 i_3 作为轮廓点串的序号。并且, 设轮廓点串的总数为 n_3 。

以 2 次光滑函数为底展开 $S_x(t)$ 、 $S_y(t)$ 。关于光滑函数如图 46 所示。它是只在 3 个小区间有值的函数。把区间取为 $(0, T)$, 则 2 次光滑函数 ϕ_{k3} 的 M 维函数系为

$$\phi_{k3}(t) = 3(T/M)^{-2} \sum_{q=0}^3 (-1)^q (t - \xi_{k+q})^2_+ / \{ (q! (3 - q)!) \} \quad (66)$$

$$k = -2, -1, 0, 1, 2, \dots, M+2$$

以该函数为基底, 利用系数, $S_x(t)$ 、 $S_y(t)$ 可以表示为

$$S_x(t_{i3}) = \sum_{k=-2}^{M+2} C_k^x \phi_{k3}(t_{i3}) \quad (67)$$

$$S_y(t_{i3}) = \sum_{k=-2}^{M+2} C_k^y \phi_{k3}(t_{i3}) \quad (68)$$

这里, 定义

$$t > \xi_{k+q} \quad \text{时} \quad (t - \xi_{k+q})^2_+ = (t - \xi_{k+q})^2 \quad (69)$$

$$t \leq \xi_{k+q} \quad \text{时} \quad (t - \xi_{k+q})^2_+ = 0 \quad (70)$$

ξ_{k+q} 是将区间 T 作 M 等分时的小区问。

$$\xi_{k+q} = (k+q) T/M \quad (71)$$

系数 C_k^x 、 C_k^y 通过使各轮廓点串的值 (x_{i3}^u, y_{i3}^u) 和 $S_x(t_{i3})$ 、 $S_y(t_{i3})$ 的值近似来决定。利用最小二乘法决定系数的值。平方误差 Q 定义为

$$Q = \sum_{i3=1}^{n3} | x_{i3}^u - S_x(t_{i3}) |^2 + \sum_{i3=1}^{n3} | y_{i3}^u - S_y(t_{i3}) |^2 \quad (72)$$

为了使其为最小值, 利用下式决定:

$$\sum_{k=-2}^{M+2} C_k^x \{ \sum_{i3=1}^{n3} \phi_{k3}(t_{i3}) \phi_{m3}(t_{i3}) \} = \sum_{i3=1}^{n3} X_{i3} \phi_{m3}(t_{i3}) \quad (73)$$

$$\sum_{k=-2}^{M+2} C_k^y \{ \sum_{i3=1}^{n3} \phi_{k3}(t_{i3}) \phi_{m3}(t_{i3}) \} = \sum_{i3=1}^{n3} Y_{i3} \phi_{m3}(t_{i3}) \quad (74)$$

$$m = -2, -1, 0, 1, \dots, M+2 \quad (75)$$

求解上面的关系式，可以求出系数 C_k^x, C_k^y 。为了评价该系数是否适当，按下式求各点误差的最大值。

$$\varepsilon = \max_{i3=0}^{n3} \{ (X_{i3}^u - s_x(t_{i3}))^2 - (Y_{i3}^u - s_y(t_{i3}))^2 \}^{1/2} \quad (76)$$

其中，在 max 的演算中， $i3$ 的范围从 0 到 $n3-1$ ，求这个范围内的最大值。若 ε 小于某常数，就可以近似。例如，根据是否 $\varepsilon < 0.9$ 进行判断。如果 ε 大于某一常数，就把因次数 M 加 1，进行同样的计算。提高因次数，就增加了区间数，所以可以提高近似精度。进行同样的误差计算，则在某时达到 $\varepsilon < 0.9$ 。这时，近似计算即告结束。

O. 压缩数据输出机构

这是利用前面的程序把文字字体、词符—插图的轮廓线分离为直线(线段)、正圆、圆弧、自由曲线。它们有始点、终点，还有斜率、圆心、半径等等参数。根据种类不同，应存储的数据也不同。

对直线数据的情况，把表示直线的图、直线的始点坐标作为数据进行存储。终点坐标作为下一个区间的始点给出，所以这里不需要存储。直线的斜率是定义直线的参数，但通过连结始点和下一个连接点的坐标，能够得到直线，所以也不需要斜率，也不需要存储。

对于正圆数据情况,可以从正圆存储装置 G 直接得到数据。这些数据已用第 1 次的数据近似机构 A 选择出来。对于正圆,把表示正圆的图、圆心坐标、圆半径作为数据进行存储。

作为圆弧数据,把表示圆弧的图、圆弧始点坐标、圆弧的分割长度(圆弧长/周长)、轮廓点数和函数的系数进行存储。

作为自由曲线的数据,存储函数的因次,轮廓点数,轮廓点串变化的中点及函数的系数。

P. 压缩数据存储装置

这是存储从压缩数据输出机构输出的直线、正圆、圆弧、自由曲线等数据的装置。存储这些数据后,在适当的时候输出。这是将数据压缩并进行存储的装置。后面将说明根据存储的数据再生文字的装置。

存储在压缩数据存储装置 P 中的数据结构示于表 1。

下面,对数据的大小进行说明。连接点间是直线时,用于表示直线的图需要 1 字节,表示线段的始点需要 2 个字节(x 坐标和 y 坐标),共需要 3 个字节。连接点间是圆弧时,表示圆弧的图需要 1 个字节,表示圆弧的始点需要 2 个字节,表示圆弧的圆心角需要 4 个字节,表示轮廓点串数需要 1 个字节,表示近似函数的系数(有 6 个)需要 12 个字节,总共需要 20 个字节。连接点间是自由曲线时,表示函数的因次 M 需要 1 个字节,表示轮廓点数需要 1 个字节,表示轮廓点的变化中心需要 2 个字节,表示近似函数的系数需要 2M 个字节,

合计为 $4+2M$ 个字节。

下面说明的轮廓再生机构 R、文字、词符—插图再生机构 S、再生数据输出机构 T 构成用于以任意的大小再生文字字体、词符—插图的再生数据生成装置。

R. 轮廓再生机构

这是根据存储的压缩数据再生应成为文字字体、词符—插图的骨架的轮廓线的机构。轮廓线有直线、正圆、圆弧和自由曲线等情况。

[直线的再生] 直线的再生通过用直线从始点的坐标连结到下一个区间的连接点的坐标来进行。不需要关于直线的斜率的数据。

[正圆的再生] 正圆的再生根据圆心坐标和半径的数据，通过描述以圆心坐标为圆心给定半径的圆进行。

[圆弧的再生] 圆弧的再生通过把所存储的各数据 ($A_x, B_x \dots$) 代入下式进行。

$$S_x(t) = A_x \cos \{2\pi t / (T/n_{arc})\} + B_x \sin \{2\pi t / (T/n_{arc})\} + C_x \quad (77)$$

$$S_y(t) = A_y \cos \{2\pi t / (T/n_{arc})\} + B_y \sin \{2\pi t / (T/n_{arc})\} + C_y \quad (78)$$

通过使参数 t 在 $(0 \sim T)$ 的区间内变化，可以根据 $S_x(t)$ 、 $S_y(t)$ 得到 x 、 y 坐标。

[自由曲线的再生]

标本点 t_i 处于区间 $((L-1)(T/M), L(T/M))$ 内时 $(1 \leq L \leq M)$, 利用 $p = L - t_i \times M/T$, 各标本点 t_i 的近似函数基底 ψ_{k3} 的值可以表示为

$$\phi_{k3}(t_i) = 0.5 p^2 \quad k = L \quad (79)$$

$$\phi_{k3}(t_i) = p(1-p) + 0.5 \quad k = L+1 \quad (80)$$

$$\phi_{k3}(t_i) = 1 - \phi_{L3}(t_i) - \phi_{L+13}(t_i) \quad k = L+2 \quad (81)$$

$$\phi_{k3}(t_i) = 0 \quad k \leq L-1, L+3 \leq k \quad (82)$$

式中, L 是小于因次数 M 的自然数。由于是同一性质的数, 所以应写为 M' , 但是, 由于“'”号不能成为 $1/4$ 角格式, 故用 L 表示。

利用这样的基底 ψ_{k3} , 各标点的近似函数值 $S(t_i)$ 可以由下式求出。

$$S(t_i) = \sum_{k=L}^{L+2} C_k \phi_{k3}(t_i) \quad (83)$$

S. 文字、词符—插图再生机构

这是再生文字形状、词符—插图形状的机构。由于得到了轮廓线, 所以把轮廓线所包围的部分作为黑象素, 形成黑白 2 值图象后, 再生为文字形状、词符—插图的形状。或者相反, 也可以把轮廓所包围的部分作为白象素, 把其余的部分作为黑象素。进而, 还可以把轮廓线所包围的部分作为某种彩色象素, 把其它部分作为另一种彩

色象素。总之,只要能区别轮廓线的内外就行。

T. 再生数据输出机构

为了根据函数的系数再生图象,例如利用能将再生图象的尺寸从1mm的正方形到A₃版指定的设计编辑器。可用有300DPI精度的激光打印机或具有600DPI精度的激光打印机印制再生图象。此外,还可以用3000DPI的电植机(印刷机器)或400DPI的可读数据对应的打印机、激光剪辑机进行输出。

用函数系数的形式存储文字字体、词符—插图后,可以以任意的信率放大和缩小,另外,坐标也可任意指定其中心。因此,可以在任意的位置以任意的尺寸输出任意的图案、词符—插图。

上述机构,可以通过装配含有根据用C语言编制的程序进行浮点运算的数值运算处理器的PC-9801DA来实现。

为确认本发明的效果,用本发明对铅印体的“明”、“体”、黑体字的“语”以及JIS第1标准的文字进行了压缩。其结果示于表2。另外,在图40—图42还示出了“明”、“体”、“语”3个文字的原图象和根据本发明的再生图象(用300DPI的激光打印机进行打印)。从这些图可知,能够非常忠实地再生。

此外,还用本发明对图43的乐谱音符、图44的圆形图表、图45的香烟的词符—插图进行了压缩,其结果于表3。并对这些乐谱音符、圆形图表、香烟及图50的半导体二极管等,分别在各图的下半部示出了根据本发明的数据压缩存储后再生的图象。它们都是优质

的再生图象,几乎找不出与原图象的区别。由此可知,根据本发明的再生图象(用 300DPI 的激光打印机进行打印)能够不降低品质并且忠实地再生图象。

表 2、表 3 示出了用本发明把各文字、词符—插图数据压缩后的数据量(字节)、压缩率(%)和压缩时间(秒)。

这里,所谓压缩率是用原图象的数据量(约 8k 字节)去除按本发明所压缩的字体的数据量(字节),再乘 100,以 % 表示的数值、因为原图象是用 256×256 象素的光学读取装置读取的,所以有约 8k 字节的数据量,但是,用本发明的方法进行压缩时,对于文字字体,将减少到 300—500 字节,对于词符—插图将减少到 100—300 字节的数据量。

压缩率约为 2—6%,可以归结为极少量的数据。另外,每 1 个文字或 1 个词符—插图的压缩时间约为数秒,时间极短。这里所说的压缩时间是 CPU 时间,也包括收入到单字的数据的分割合成处理时间和绘图时间。

这样,便可缩短处理时间,减少数据。特别是印刷体的文字,多数由直线部分构成,由于在第 1 阶段作为已知的连接点抽出,故在第 2 阶段进行反复近似所求的连接点数就很少。

U. 图象存储装置 2

这是直接存储再生文字字体、词符—插图的装置。包括直接存储用打印机打印出的文字的所有的机构装置。也可以使用录像机进

行存储。也包括打印的纸在内。也可以没有特别的图象存储置。

本发明读取基本的文字字体、词符一插图后,可以压缩为少量数据进行存储。对于文字字体,每1个文字的处理时间很短。因而,即使创造由几千字构成的汉字的新字体也能够在规定时间内把它们作为数据进行存储。不仅存储简单,再生也很简单。使用存储数据能够再生任意大小的文字。

另外,对于词符一插图,同样也能以光学读取词符一插图,并压缩为少量数据进行存储,没有必要再以画在纸上的原图的形式进行保存。由于直线以直线形式抽出进行存储,圆弧和圆以几何学上的正确的圆弧和圆抽出进行存储,所以,能够除去包括在原图象中和读取时混入的噪声,使画质提高。此外,本发明能通过除去连接点,可以提高词符一插图的品质。并且,每1个词符一插图的处理时间很短。因而,即使创造众多的新词符一插图也能在规定时间内把它们作为数学的数据进行存储。

不仅存储简单,而且再生也很简单。使用存储的数据,能在任意的位置再生任意大小的词符一插图。图51和图52是放大和缩小的例子。若用光学方法放大,则无论如何也增加噪声,且轮廓线模糊不清。而本发明在计算的基础上进行放大,所以直线作为直线进行再生,圆弧作为圆弧进行再生。自由曲线也一样。由于这是在计算的基础上进行放大,所以,线段不会凹凸不平,也不会模糊不清。

图53和图54示出图形的放大、缩小、变形、回转、移动等是自

由的。图 54 中, (a) 是沿纵方向拉伸, (b) 是沿纵方向压缩, (c) 是沿斜方向拉伸, (d) 是旋转 45 度, (e)、(f) 是缩小且旋转的例子。图 55 示出了同一标记的放大和缩小, 并且位置也是任意的。这是自由曲线较多的例子, 但是, 这样的图形也能定义连接点, 应用本发明。图 56、图 57 示出了在明信片上印刷图形的情况。能够把原图以任意的大小, 印制到明信片的任意位置。考虑与文字的均衡分布, 可以插入到适当的部位。不需要以前那种用剪刀剪下原图, 再用浆糊贴到原图上的麻烦手续了。

以往, 文字、词符—插图的放大、缩小并不容易, 除了利用光学的方法进行放大和缩小外, 没有别的办法。若把图象进行光学放大、缩小、移动等过程后, 会混入噪声, 降低品质。另外, 不仅自由度低, 而且费时间。反复进行几次复制后品质就降低得很显著。

由于本发明是以连接点和分段多项式系数存储数据的, 所以, 可以通过计算进行文字、词符—插图尺寸的放大、缩小、平行移动等等、因此, 可以迅速、自由地进行文字、词符—插图的放大、缩小和移动等。通过这些操作, 也不会使文字、词符—插图的品质恶化。本发明通过除去连接点还可以进一步提高文字字体、词符—插图的品质。

通过整体坐标的平行移动可以完成位置的移动。将作为基准的点的坐标、圆弧半径、分段多项式的系数 $\{C_n^x\}$ 、 $\{C_n^y\}$ 乘以倍率, 通过计算可以完成扩大和缩小。由于数据量少, 所以计算可以迅速进行。

另外，图 54 所示的旋转也可以只通过计算完成。将作为基准的坐标和分段多项式的系数 $\{C_h^x\}$ 、 $\{C_h^y\}$ 乘以与回转角对应的系数，就可以求出新的系数。

通过计算，也可以给出图 54 所示的沿不同方向的放大和缩小。这里所谓不同的方向是放大和缩小的系数随方向而不同。如同圆和椭圆，正方形和长方形之间的变换一样。将 x 方向的系数 $\{C_h^x\}$ 和 y 方向的系数 $\{C_h^y\}$ 乘以不同的乘数后，只要进行前面所述的旋转即可。

这样，利用系数 $\{C_h^x\}$ 、 $\{C_h^y\}$ 的一次变换，就可以简单地进行移动、旋转、放大、缩小和不同方向的放大、缩小等的仿射变换。因为这些都是根据计算进行的。所以，不会因变形而掺入噪声。另外，由于数据量少，所以，短时间的计算就可以完成。对于和由形状所决定的文字不同的词符—插图的情况，这样的变形常常是有用的。

在本发明中，图形要素是以连接点和圆弧中心、半径、自由曲线的系数等集合表现的，所以，上述计算量少，时间也短。而且，利用这样的计算不会产生噪声。即，通过这些操作，词符—插图的品质不会恶化。

表 1
根据本发明作成的数据结构

	内 容	大小 (字节)
连接点间是直线 时	<ul style="list-style-type: none"> 1. 直线图 1. 线分始点的坐标 	<p>每个连接点间 1</p> <p>每个连接点间 2</p>
连接点间是圆弧 时	<ul style="list-style-type: none"> 1. 圆弧图 1. 圆弧始点坐标 1. 各连接点间的函数中心角的系数 n_{arc} ($1/4 \leq n_{arc} \leq 1$) 1. 存在于各连接点间的 1 个节点区间中的轮廓点数 1. 近似函数的系数 (6 个) 	<p>每个连接点间 1</p> <p>每个连接点间 2</p> <p>每个连接点间 4</p> <p>每个连接点间 1</p> <p>每个连接点间 1 2</p>
连接点间是自由 曲线时	<ul style="list-style-type: none"> 1. 各连接点间的函数因 次数 M ($M \geq 3$) 1. 存在于各连接点间的 1 个节点区间中的轮廓点数 1. 在各连接点间的轮廓 点列的变动中心 (μ_x, μ_y) 1. 近似函数的系数 C_x, C_y 	<p>每个连接点间 1</p> <p>每个连接点间 1</p> <p>每个连接点间 2</p> <p>每个连接点间 2 M</p>

表 2

用本发明的数据平均量、压缩率、压缩时间
(文字字体数据)

	数据量 (字节)	压缩率 (%)	压缩时间 (秒)
铅印体“明”	3 1 7	3.87	2.67
铅印体“体”	4 5 2	5.52	1.51
黑体“语”	1 8 4	2.25	4.01
JIS平均第1水平 (铅印体)	3 6 3	4.43	3.21

表 3

用本发明的平均数据量、压缩率、压缩时间(标识、图符)

	数据量 (字节)	压缩率 (%)	压缩时间 (秒)
乐谱音符	1 3 5	1.65	12.23
圆形图表	2 8 4	3.47	8.66
香烟	2 6 4	3.22	8.48
50种平均	2 5 1	3.06	11.06

说 明 书 附 图

图 1

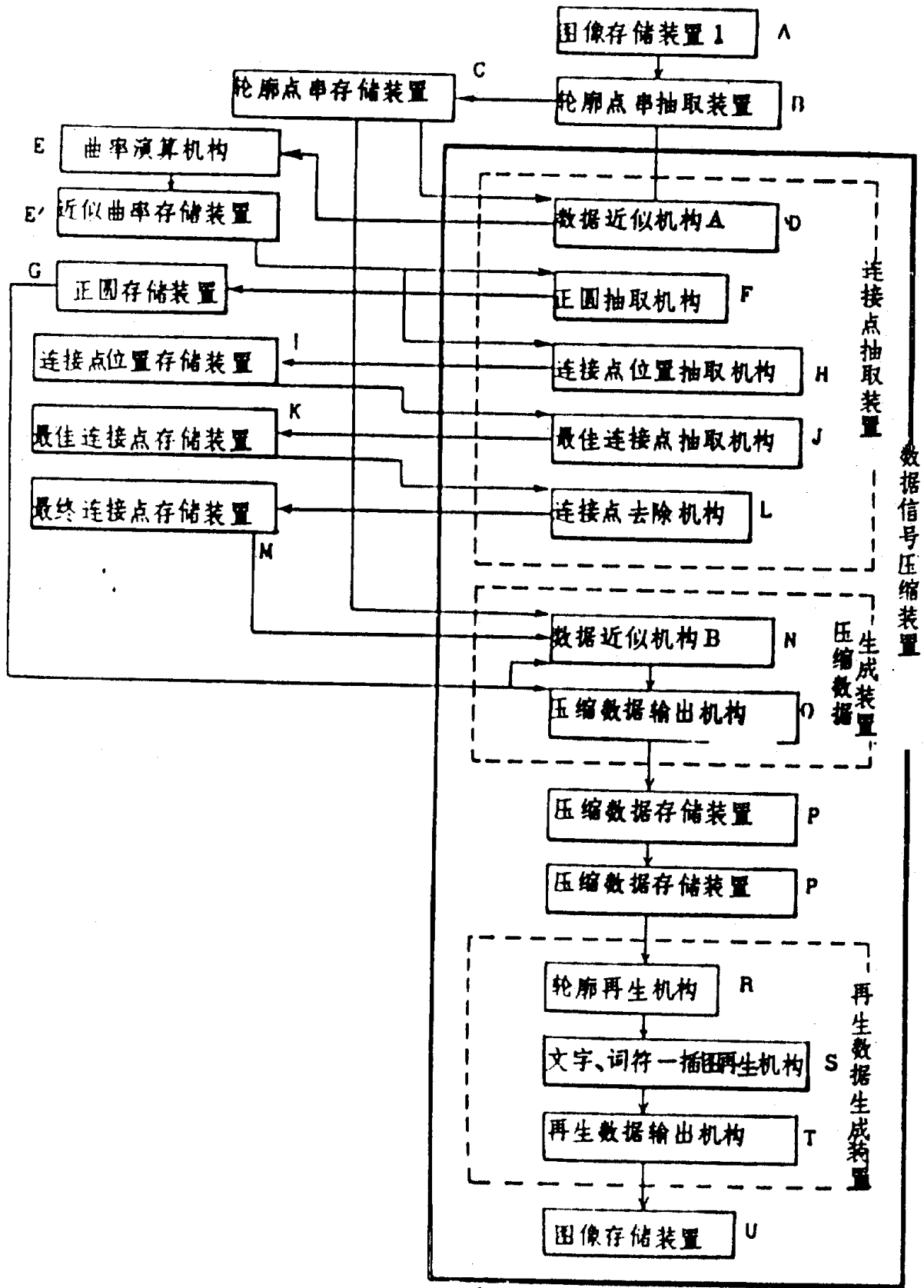


图 2

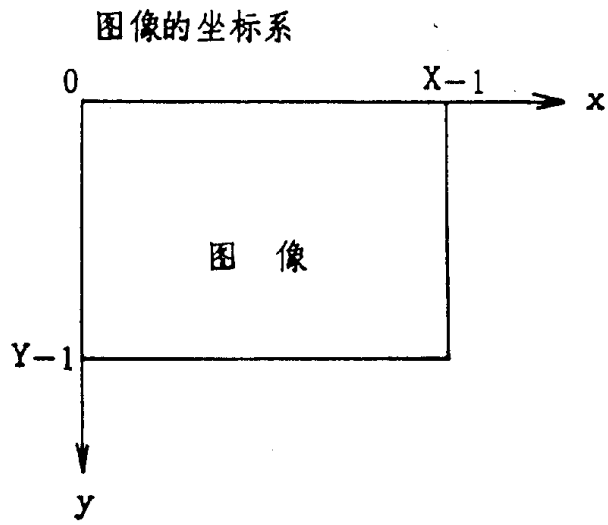


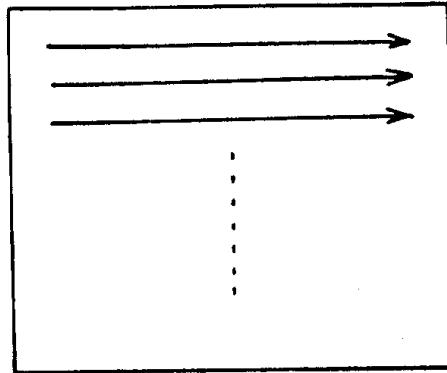
图 3

链形码

5	6	7
4		0
3	2	1

图 4

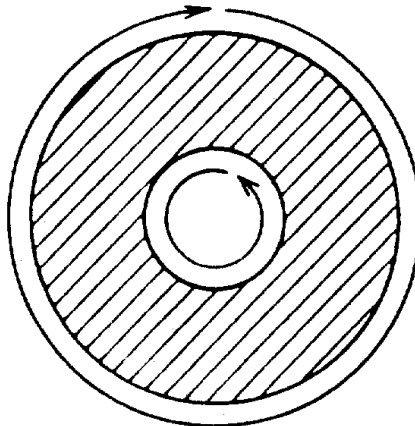
搜索尚未找出的轮廓点的方法



从左上到右下扫描数据

图 5

轮廓点的搜索方向



外侧顺时针旋转

内侧逆时针旋转

图 6

轮廓点的搜索例

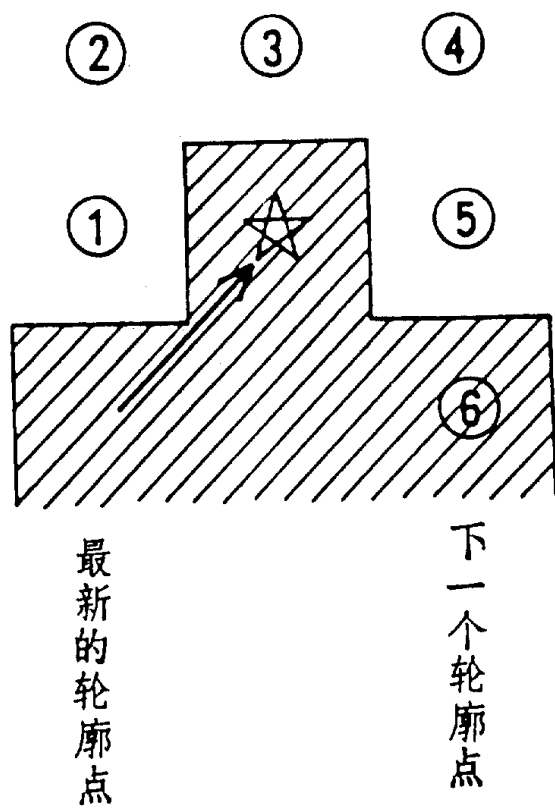
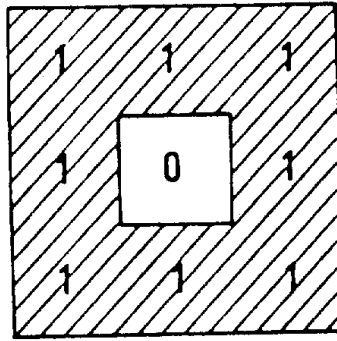
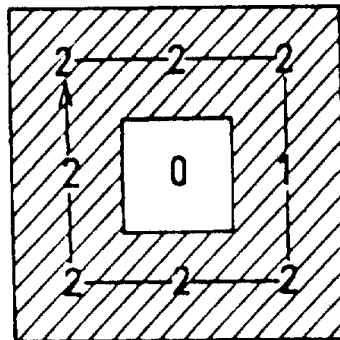


图 7

(a) 画象数据
2 值图像



(b) 抽取出外侧轮廓线的位置



(c) 抽取出内侧轮廓线的位置

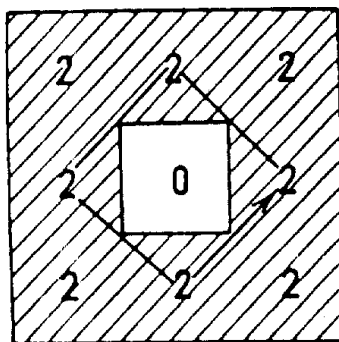


图 8

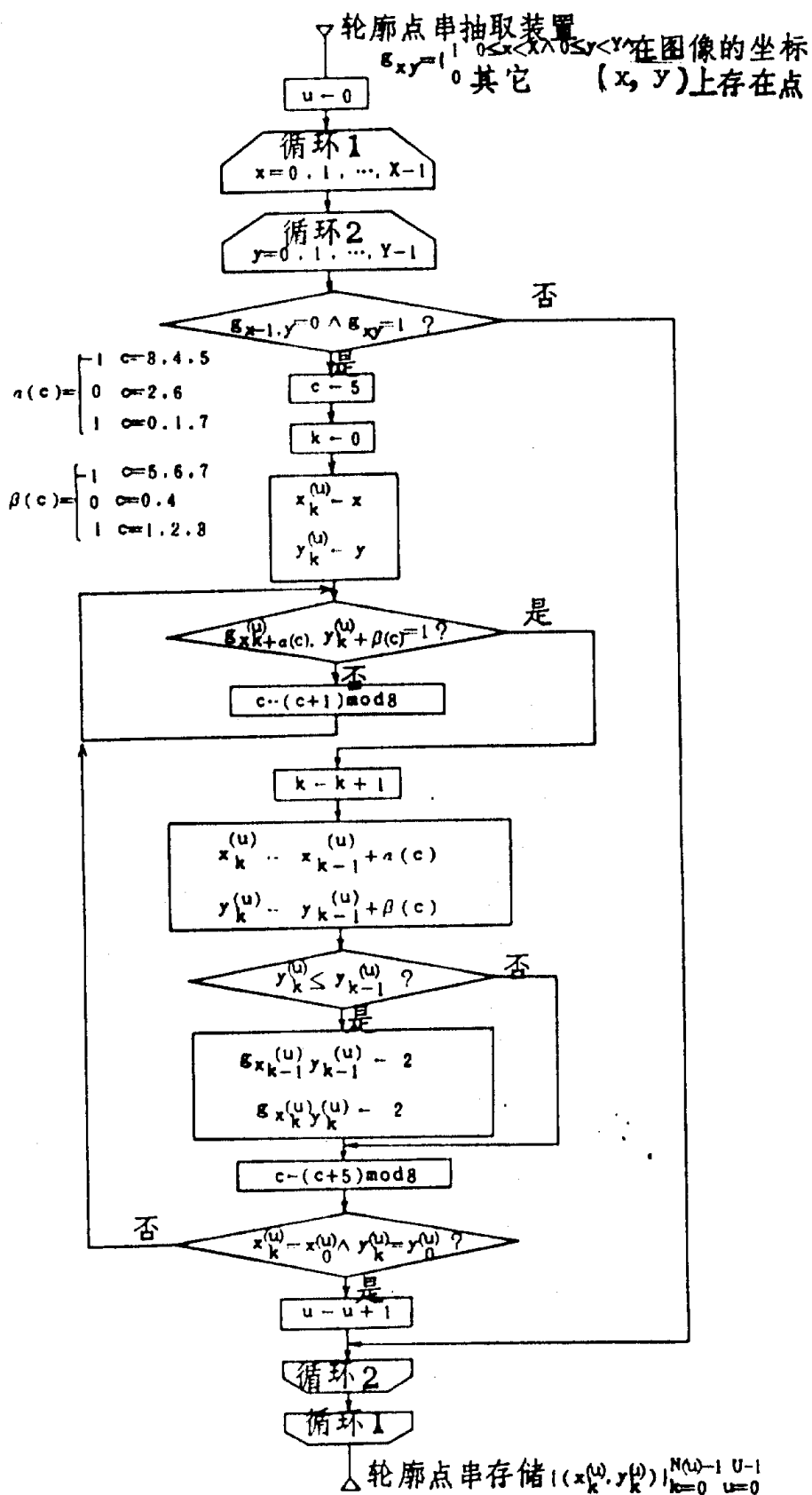


图 9

数据近似机构 A

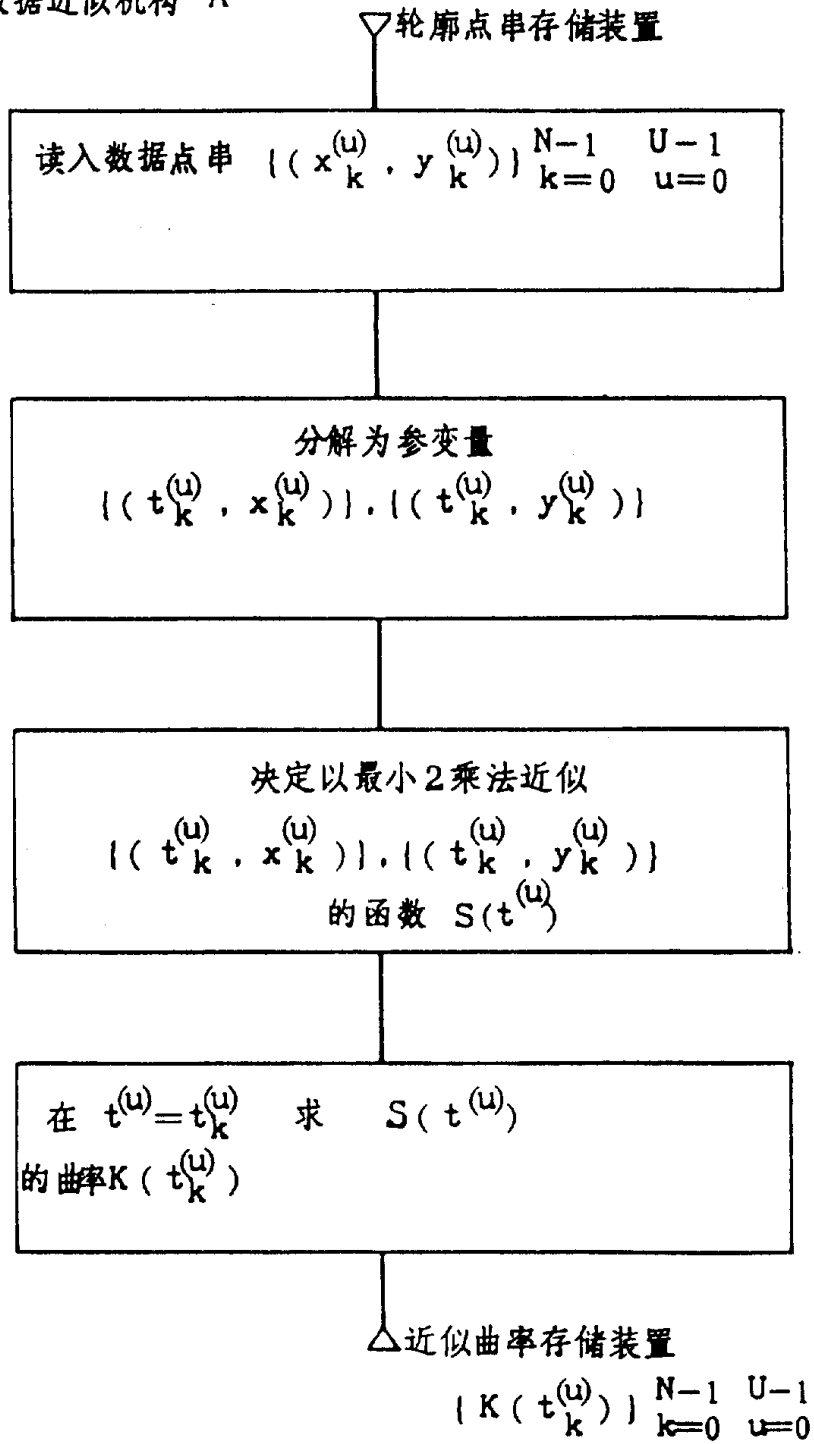


图 10

数据近似机构一部分

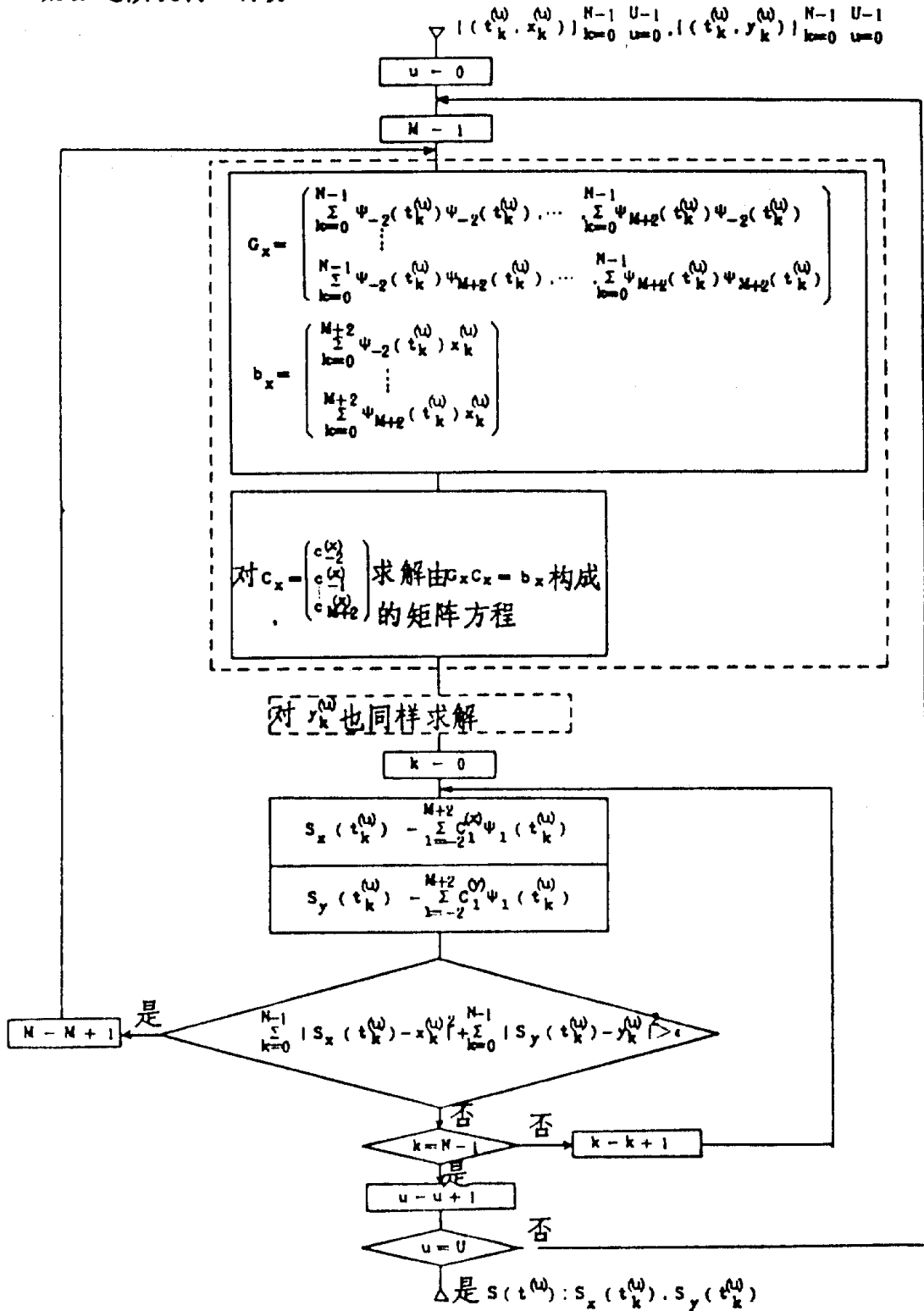


图 11

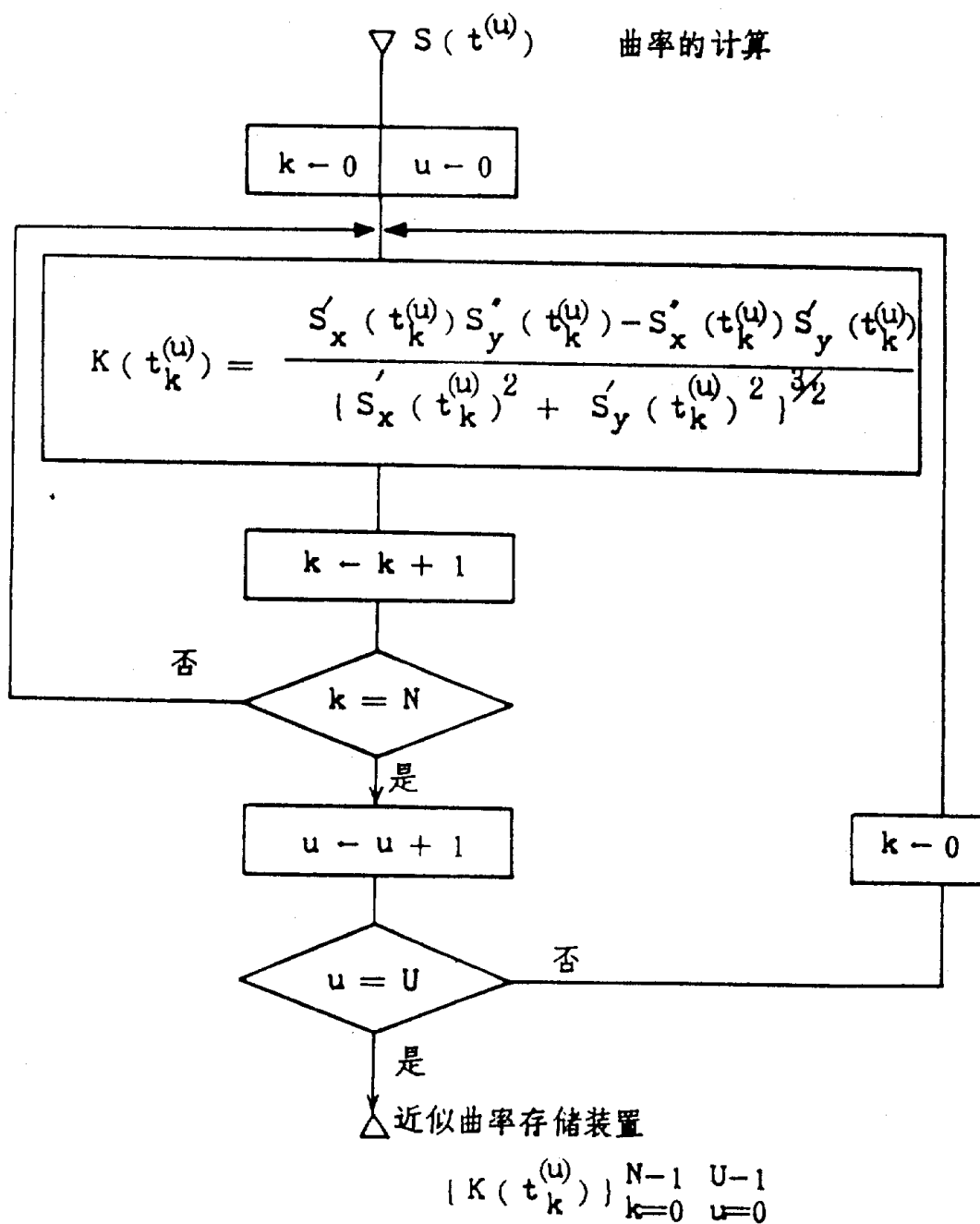


图 12

正圆抽取机构

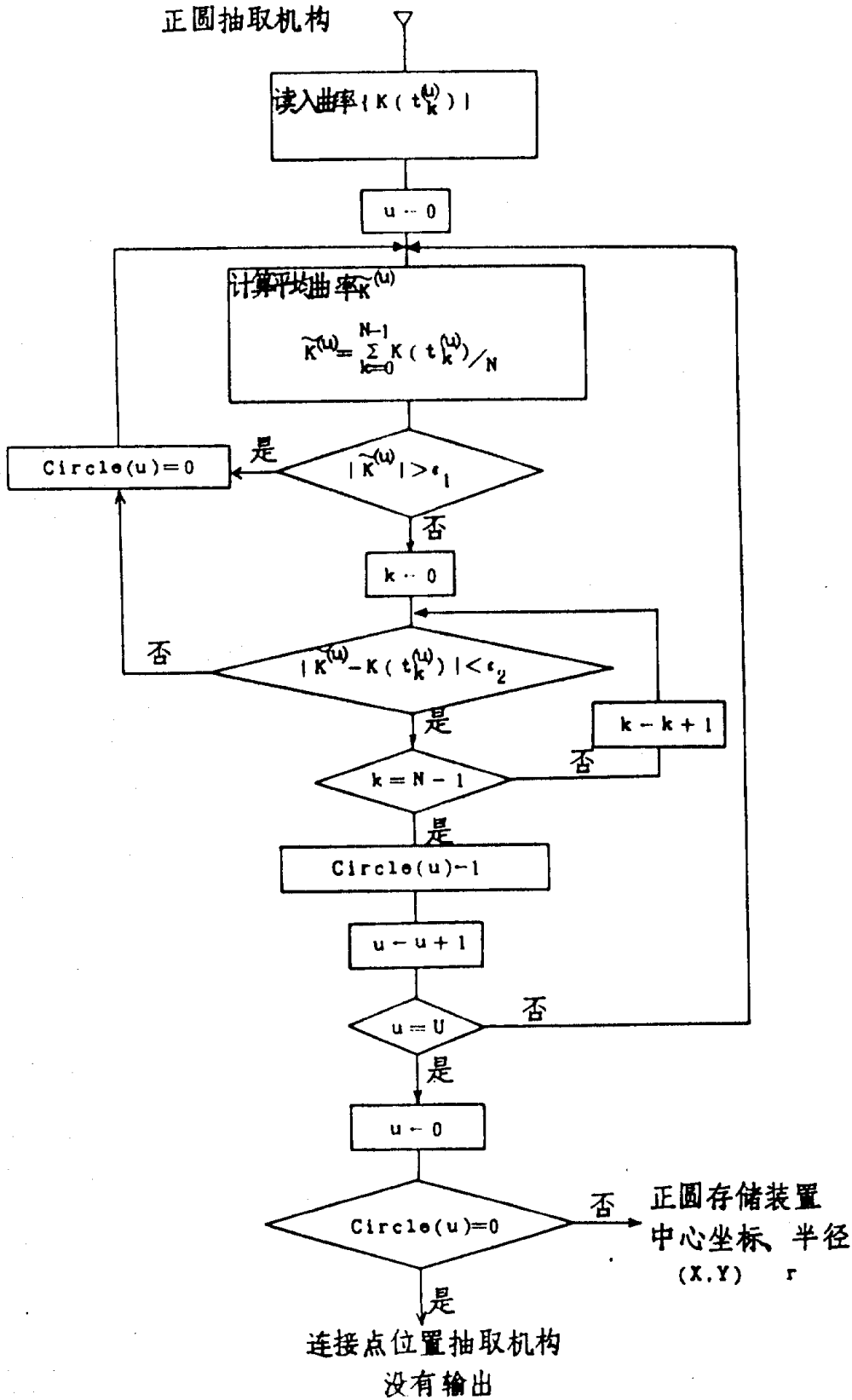


图 13

连接点抽取方法的过程

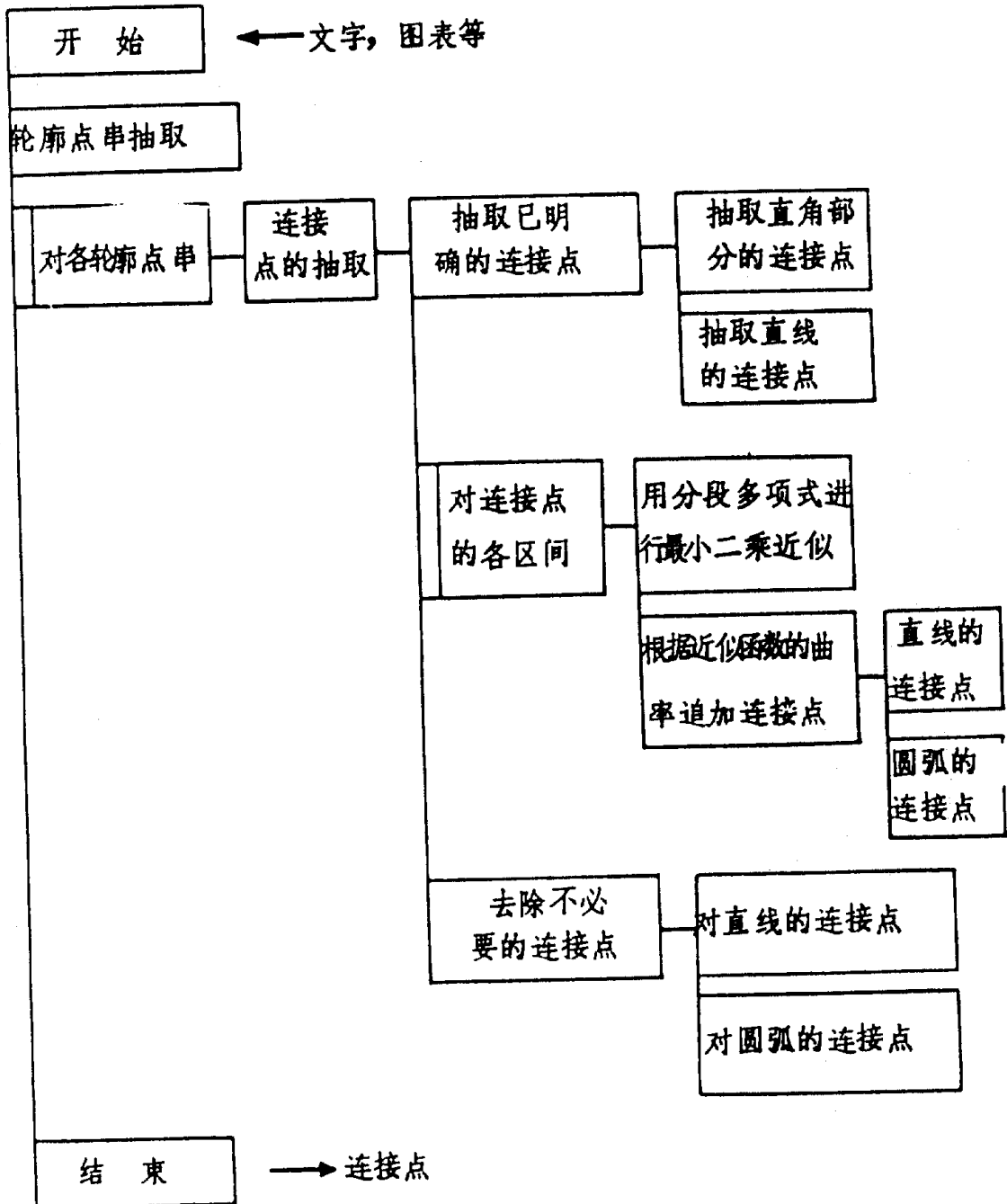


图 14

连接点抽取机构

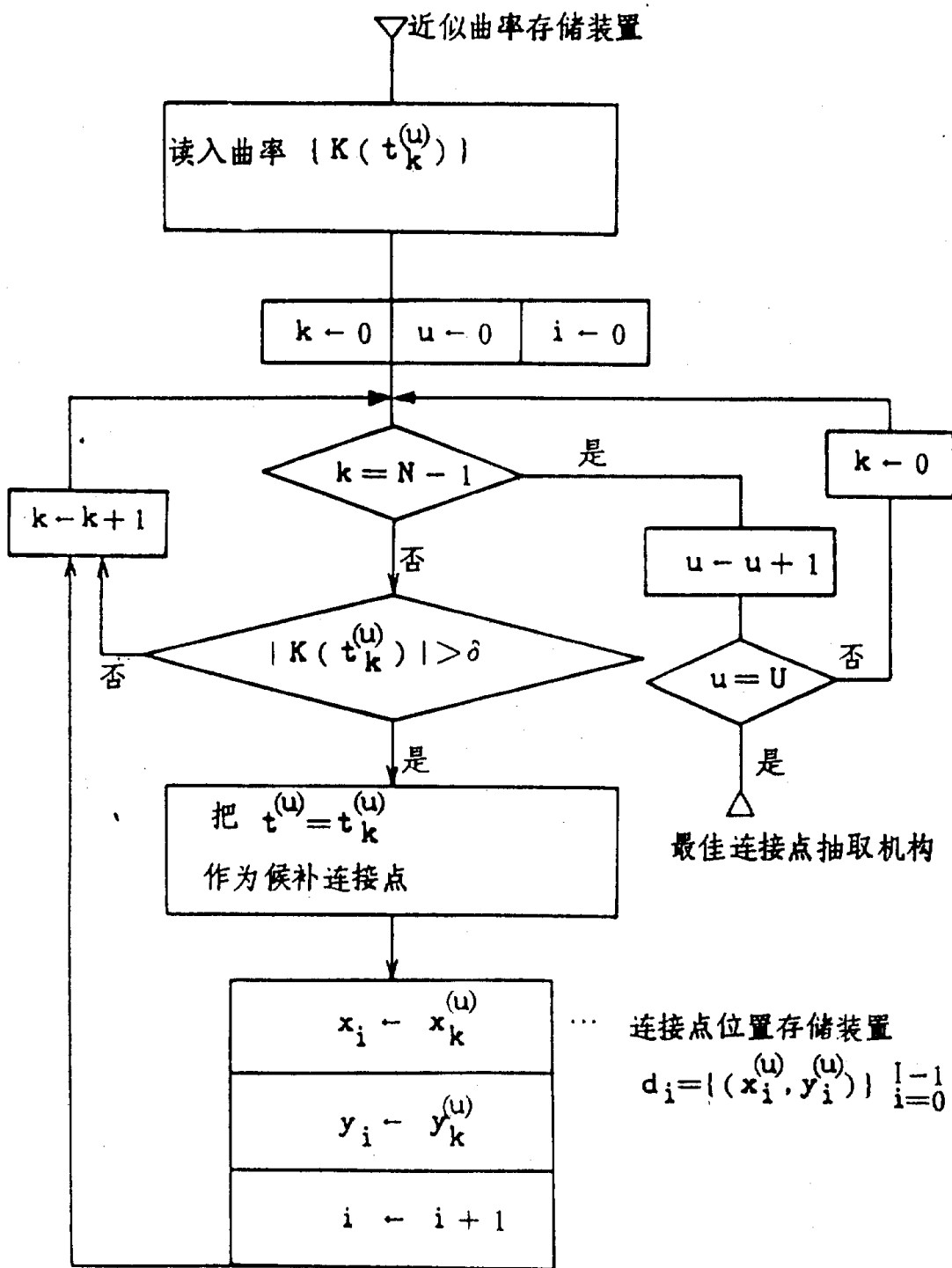
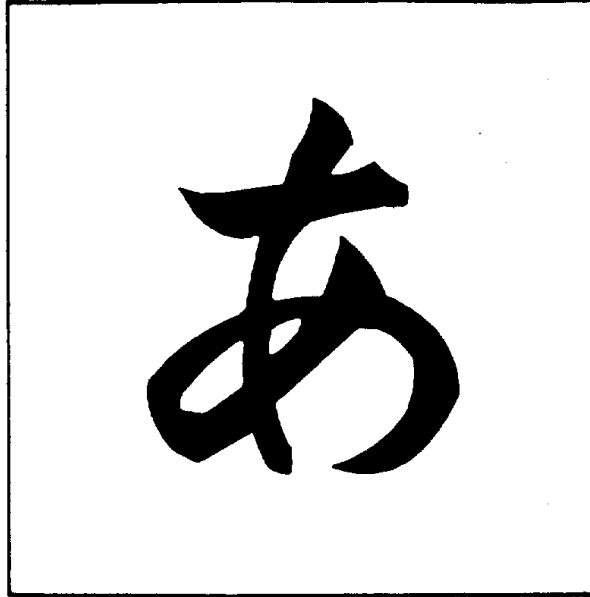


图 15

(a) 原文字



(b) 原文字

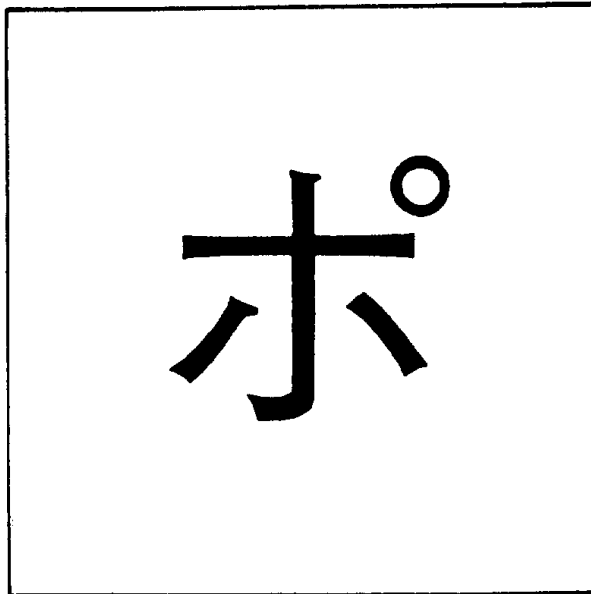
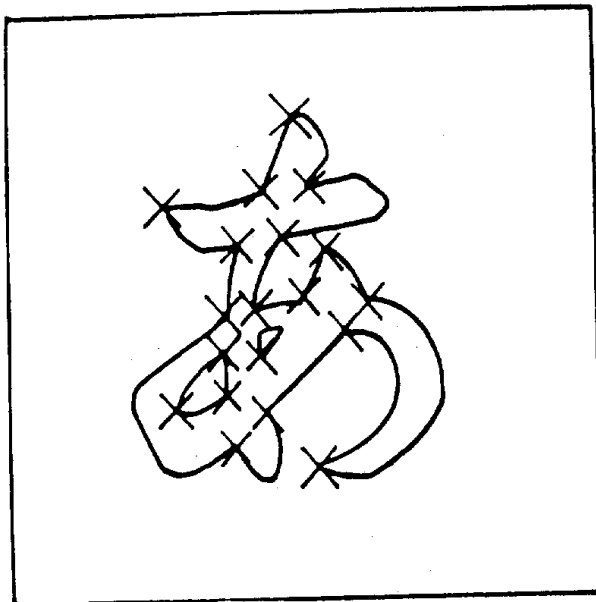


图 16

(a)



× 连接点

(b)

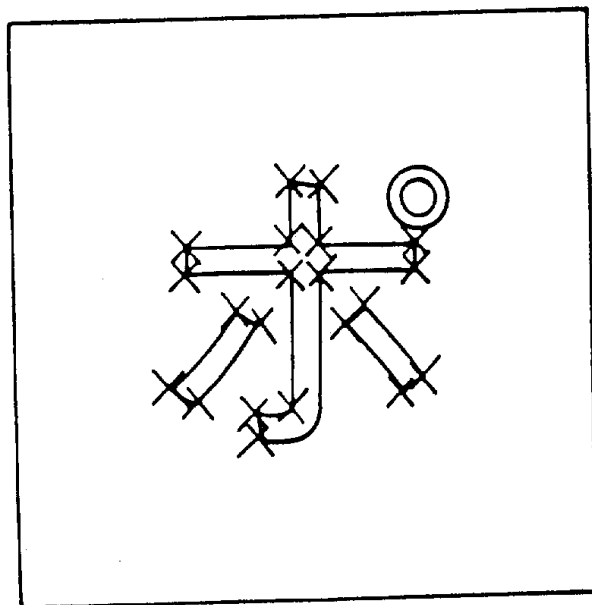


图 17

(a) 原图像



(b) 轮廓点抽取。连接点抽取

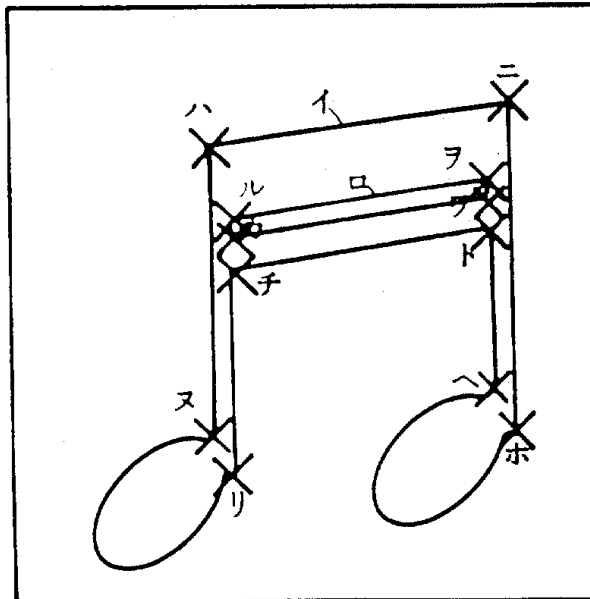
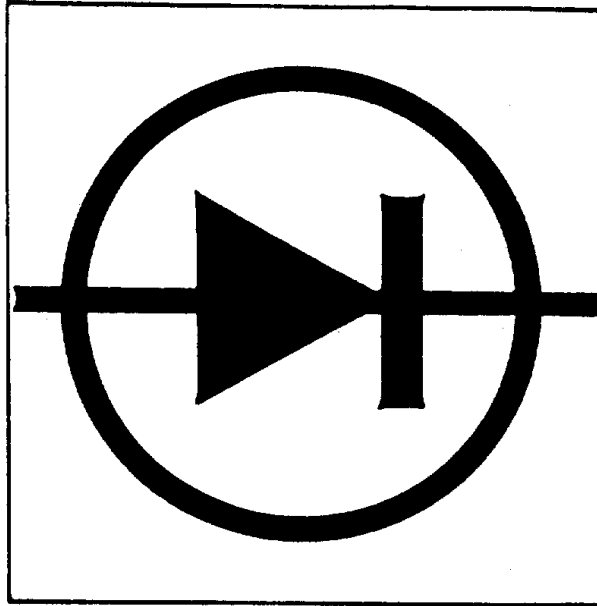


图 18

(a) 原图像



(b) 轮廓点抽取。连接点抽取

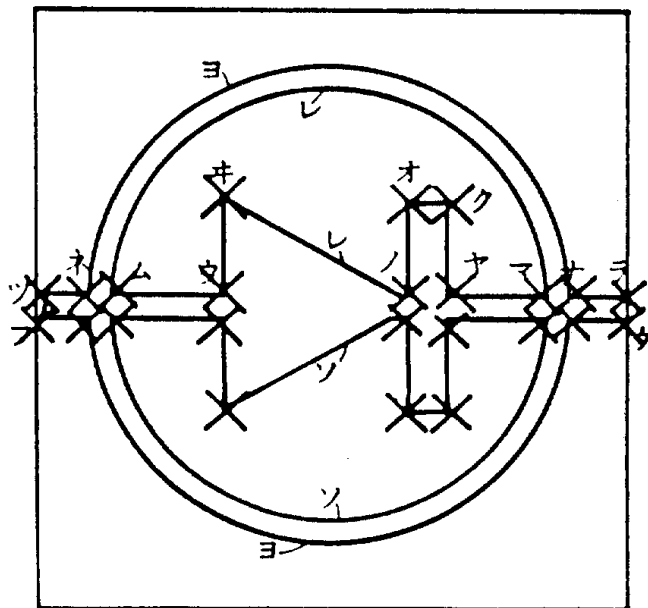


图 19

最佳连接点抽取机构 J

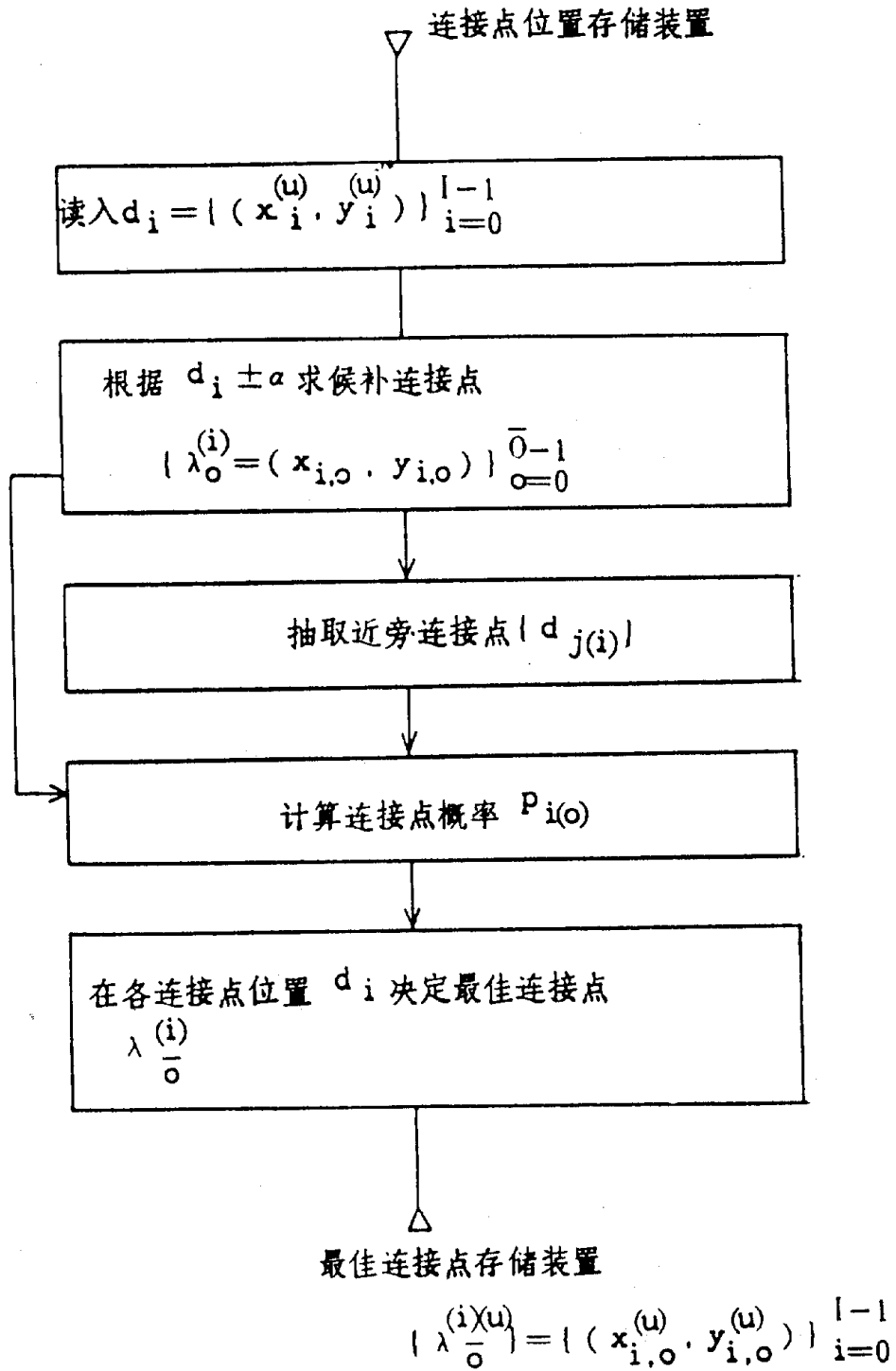


图 20

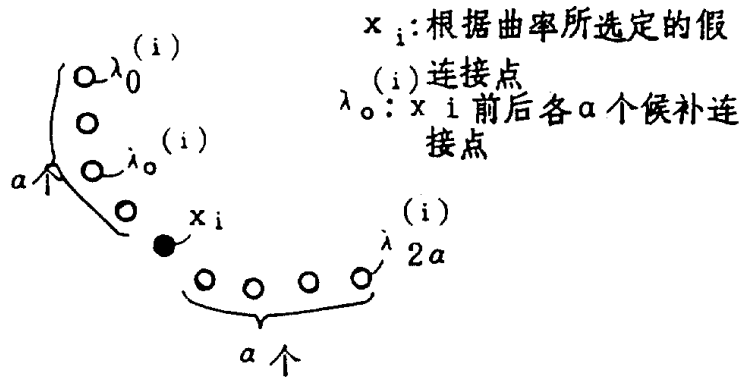


图 21

输出近旁

情况 1 $J \geq 1$ 存在近旁连接点

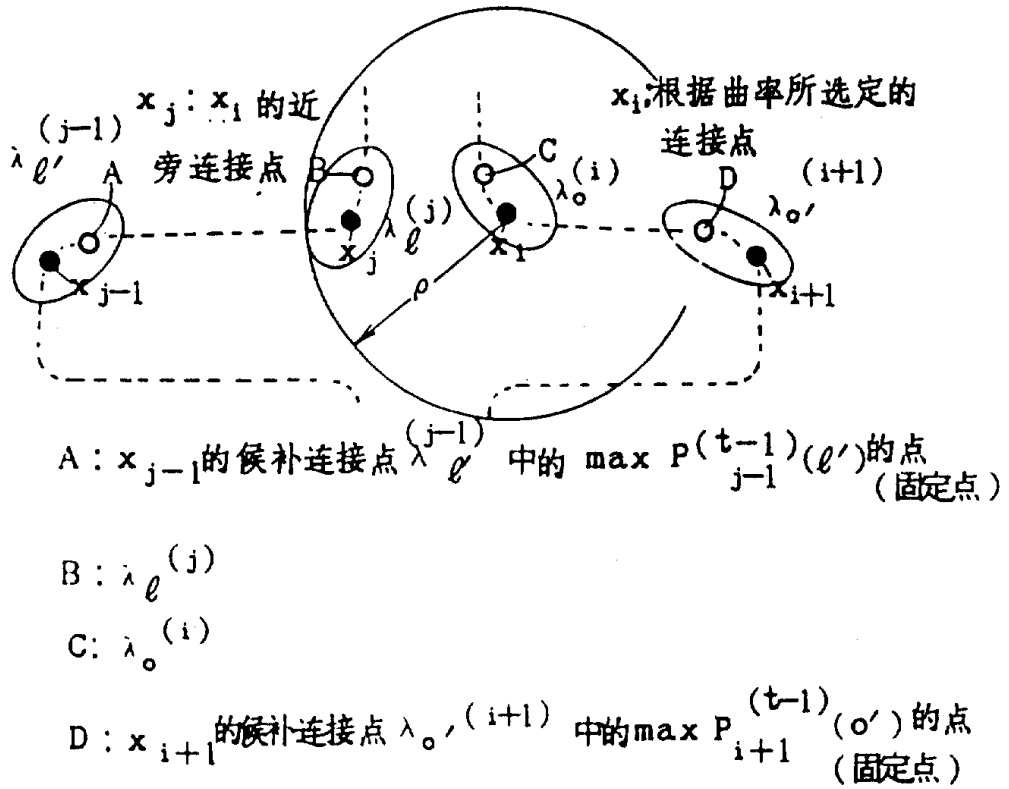


图 22

情况 1 (存在近旁连接点时)

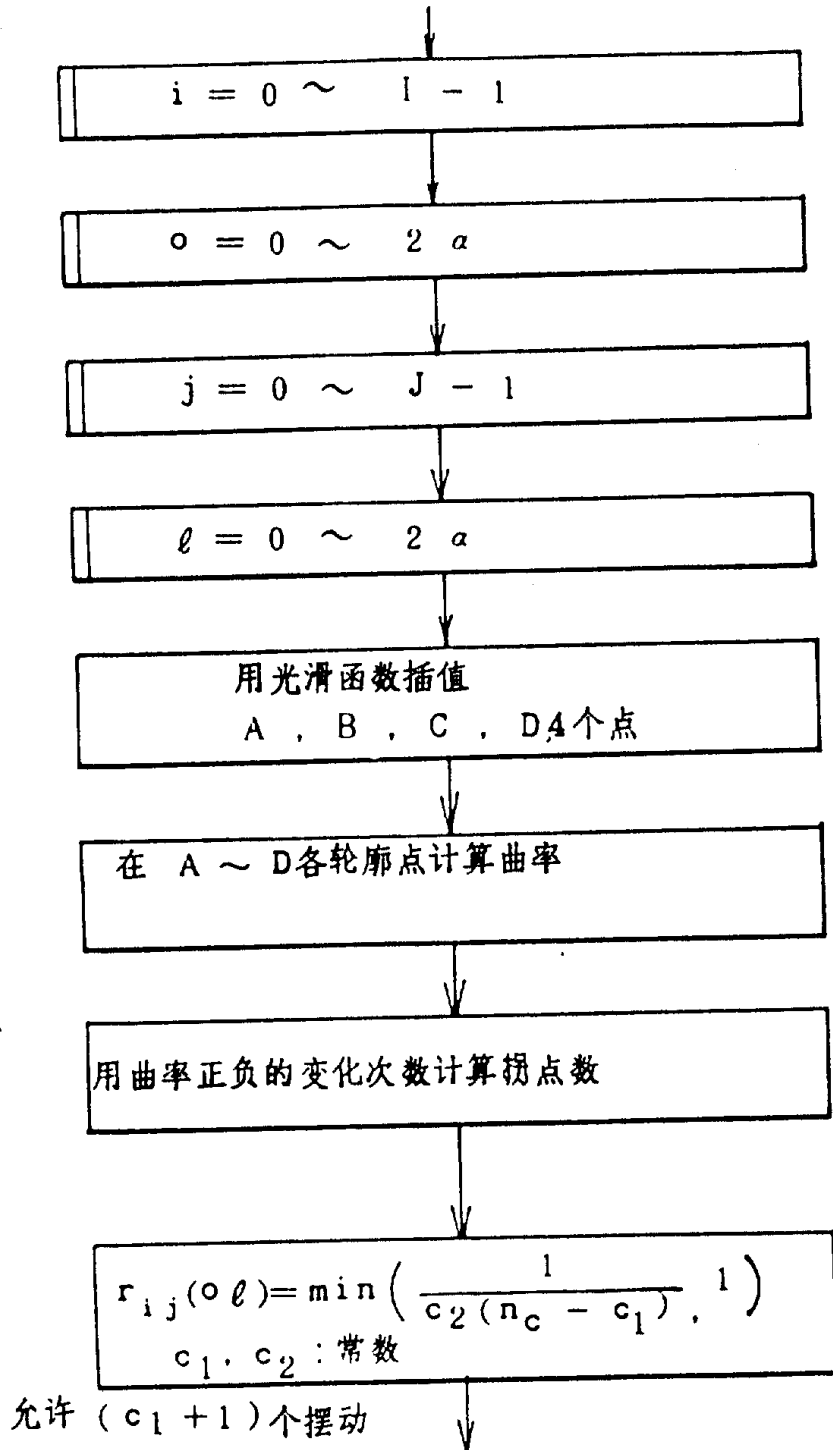
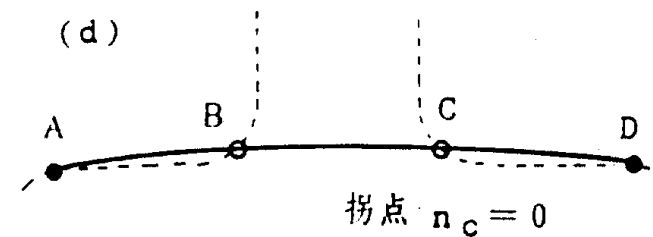
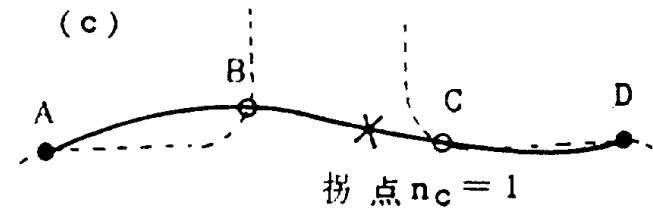
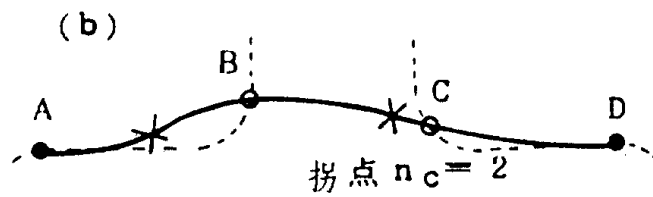
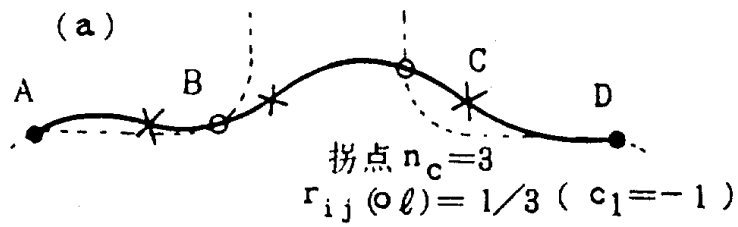


图 23



$r_{ij}(o\ell) = 1$

A: x_{j-1} 的候选连接点 $\lambda_{\ell'}^{(j-1)}$ 中的 $\max P_{j-1}^{(t-1)}(\ell)$ 的点 (固定点)

B: $\lambda_{\ell}^{(j)}$

C: $\lambda_{\sigma}^{(i)}$

D: x_{i+1} 的候选连接点 $\lambda_{\sigma'}^{(i+1)}$ 中的 $\max P_{i+1}^{(t-1)}(\sigma)$ 的点 (固定点)

图 24

$J < 1$ (不存在近旁连接点的情况)

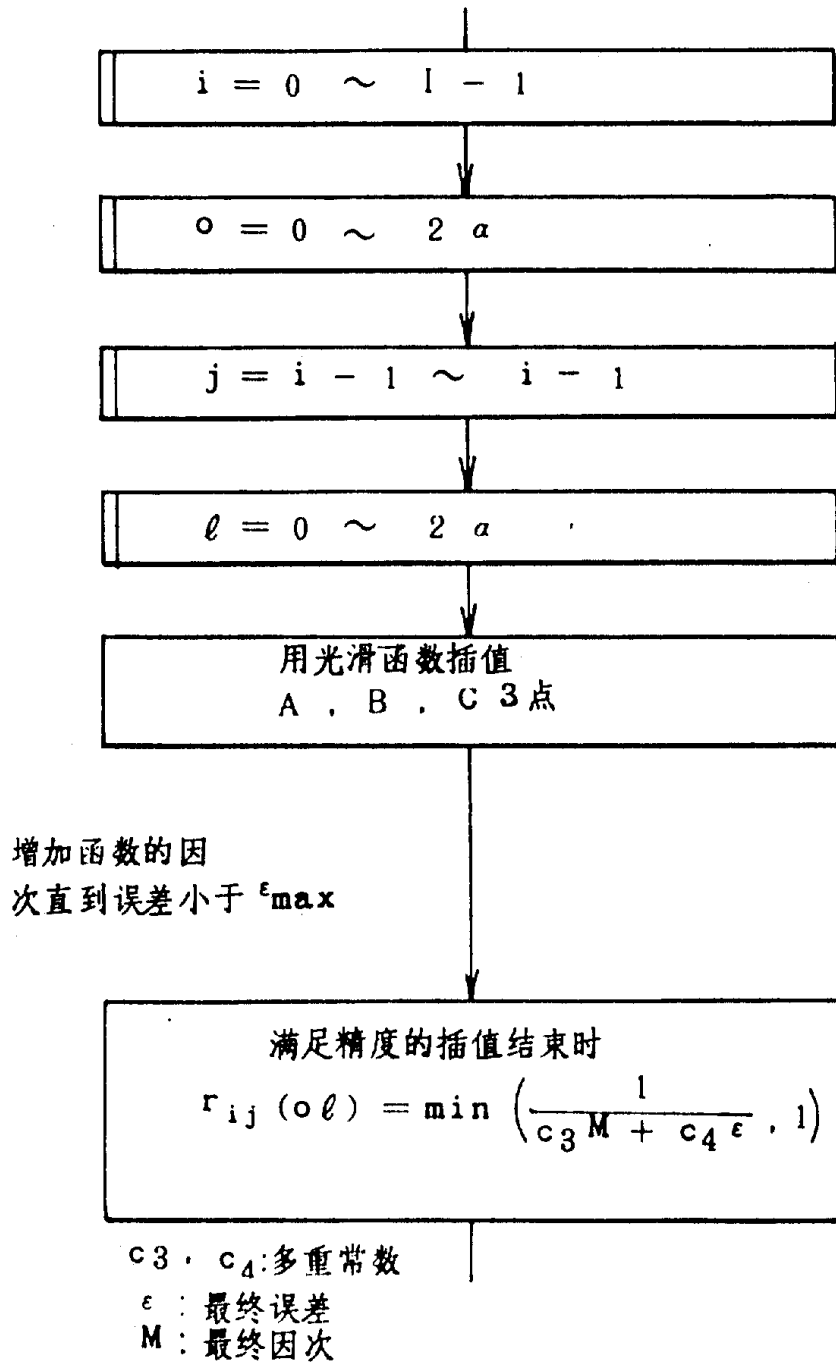
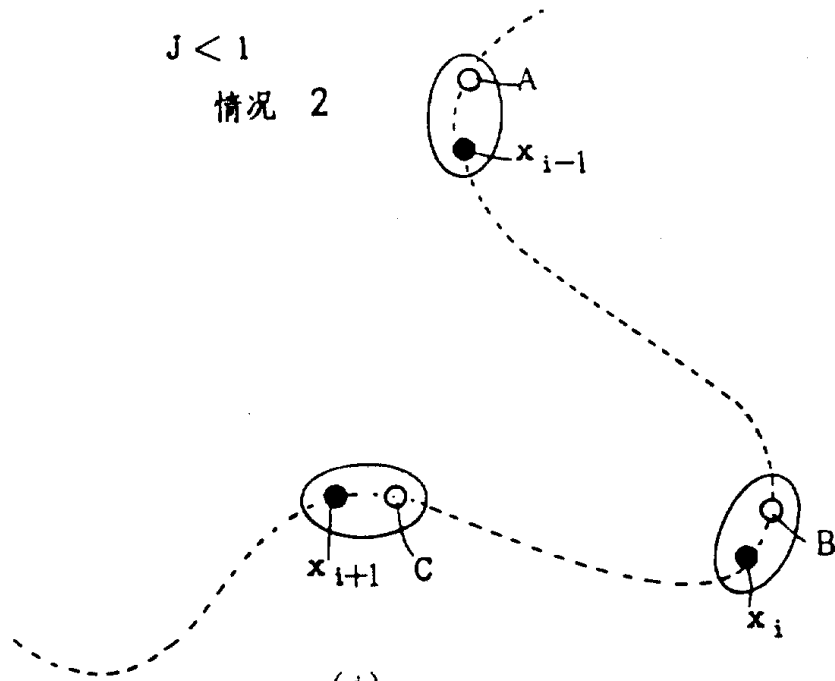


图 25

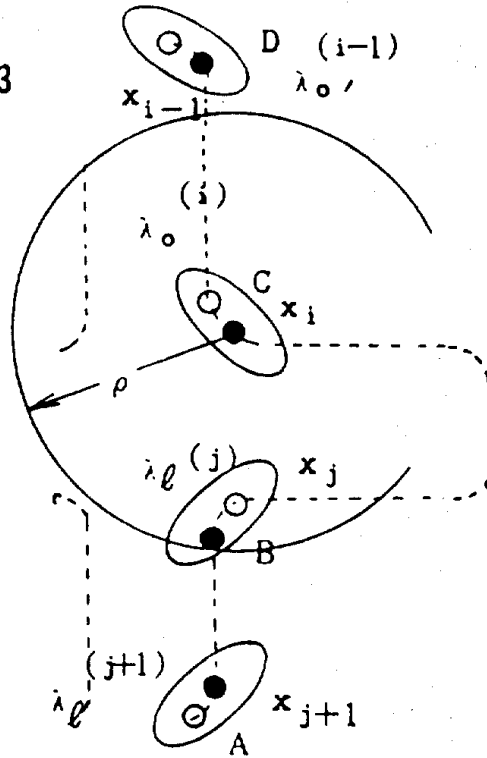


(j)
A : λ_{ℓ}
B : $\lambda_o^{(i)}$
C : x_{i+1} 的候补连接点 $\lambda_h^{(i+1)}$ 中的
 $\max_h P_{i+1}^{(t-1)}(h)$ 的点

图 26

输入近旁

$J \geq 1$
情况 3



- A: x_{j+1} 中候选连接点 $\lambda_l^{(j+1)}$ 中的 $\max_{l'} P_{j+1}^{(t-1)}(l')$ 的点 (固定)
- B: $\lambda_l^{(j)}$
- C: $\lambda_0^{(i)}$
- D: x_{i-1} 的候选连接点 $\lambda_{0'}^{(i-1)}$ 中的 $\max_{i-1} P^{(t-1)}(0')$ 的点 (固定)

图 27

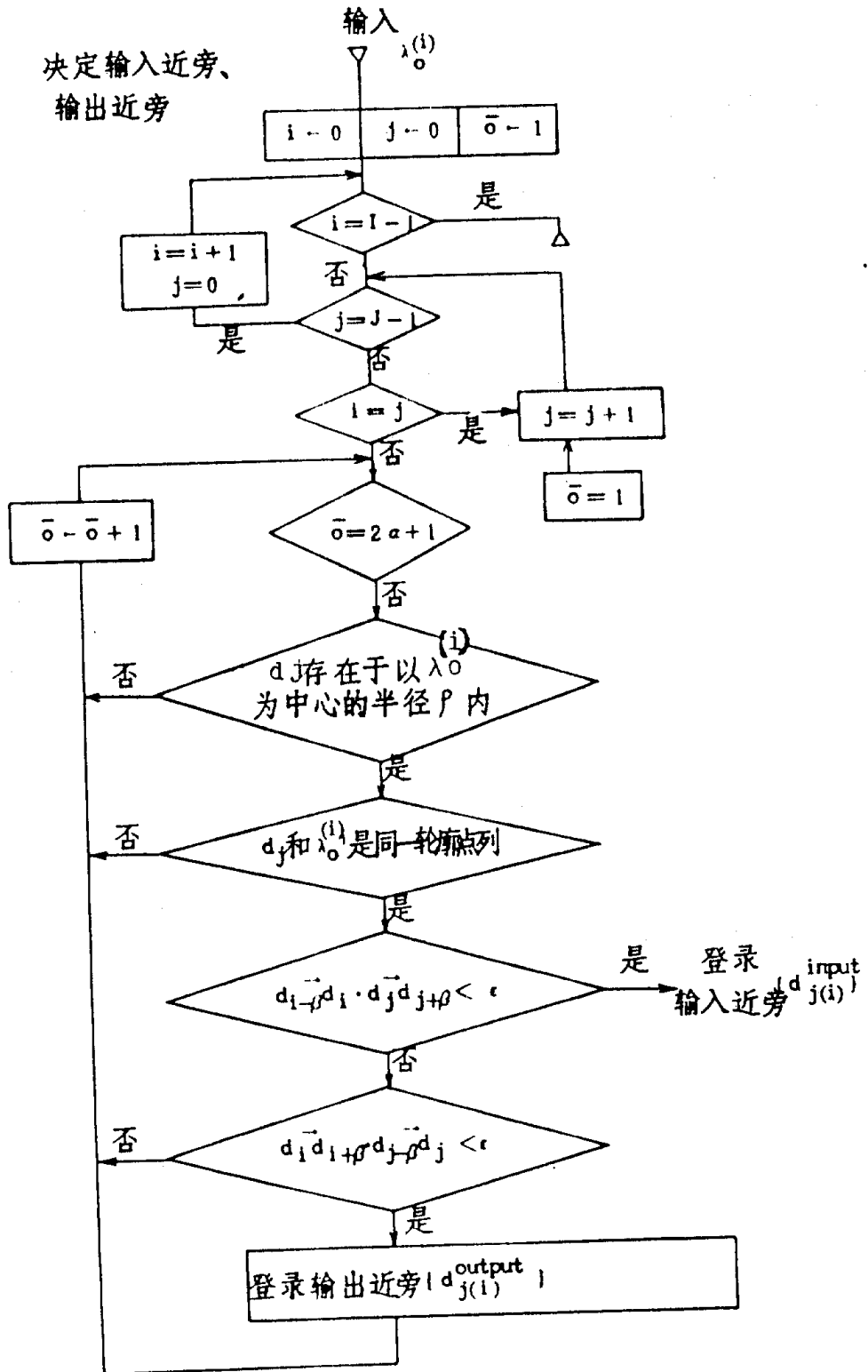


图 28

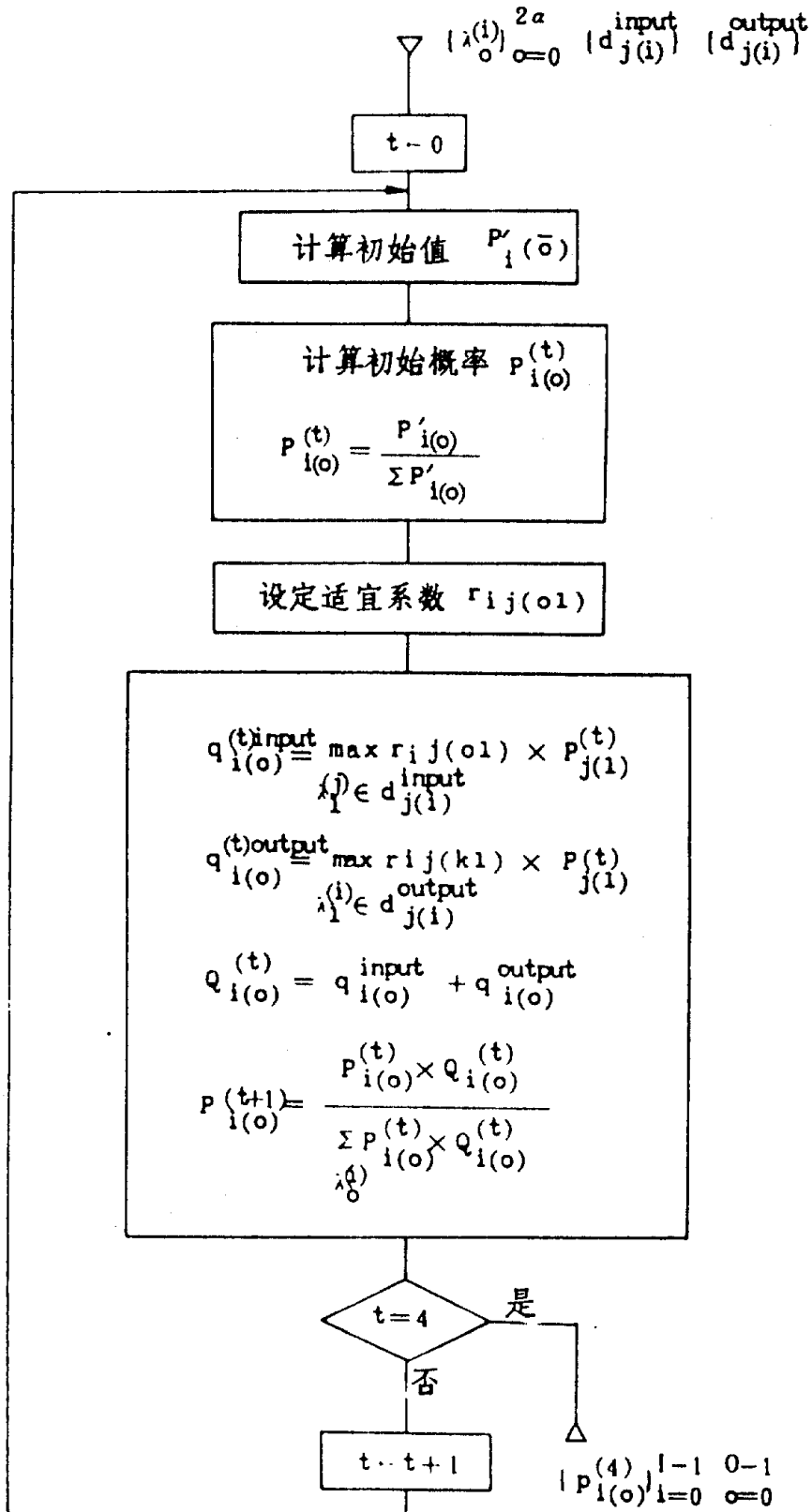


图 29

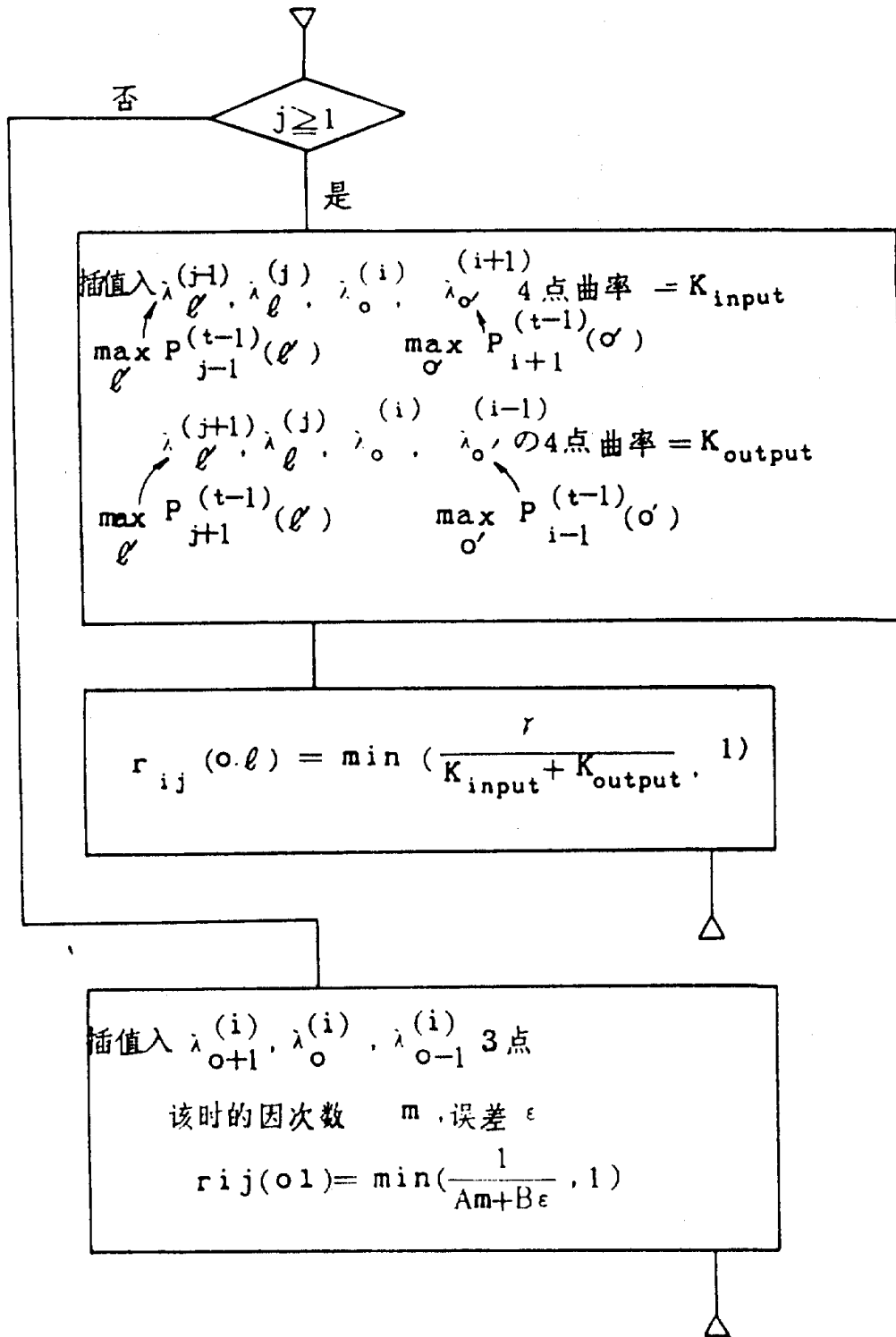
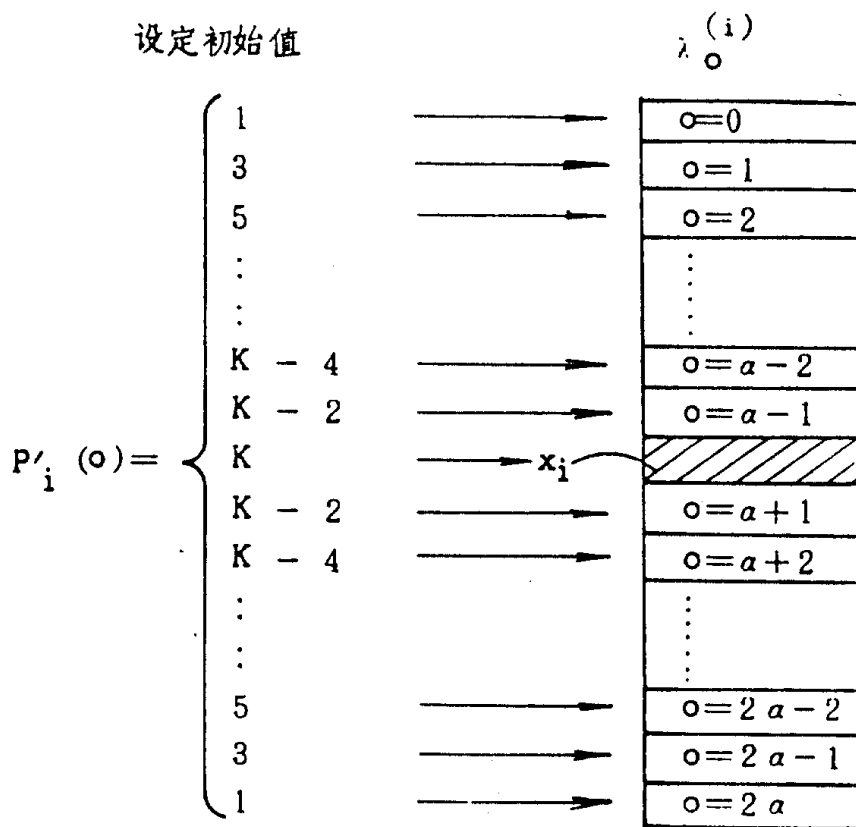


图 30



$$P_i^{(0)}(o) = \frac{P'_i(o)}{\sum_o P'_i(o)}$$

$$P_i^{(t+1)}(o) = \frac{P_i^{(t)}(o) \times Q_i^{(t)}(o)}{\sum_{\lambda_o^{(i)}} P_i^{(t)}(o) \times Q_i^{(t)}(o)}$$

$$Q_i^{(t)}(o) = q_i^{\text{input}}(o) + q_i^{\text{output}}(o)$$

$$q_i^{(t)}(o) = \max_{j} r_{ij}(o, \ell) \times P_j^{(t)}(\ell)$$

图 31

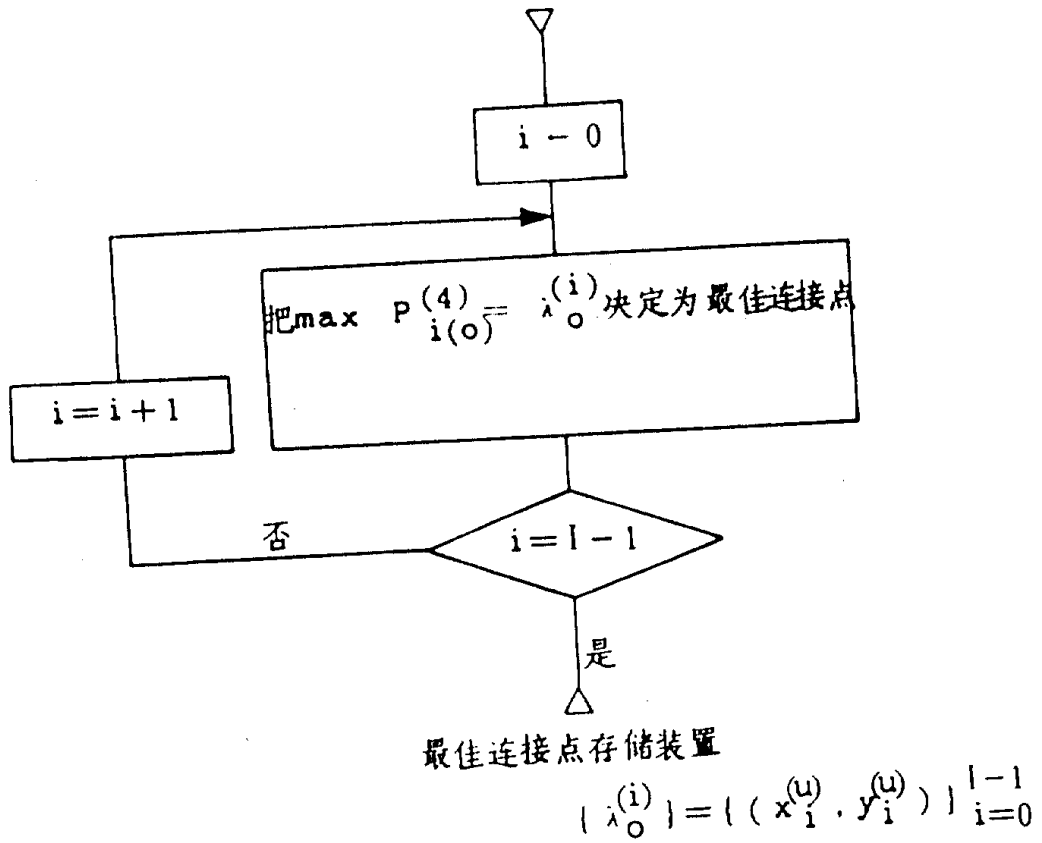


图 32

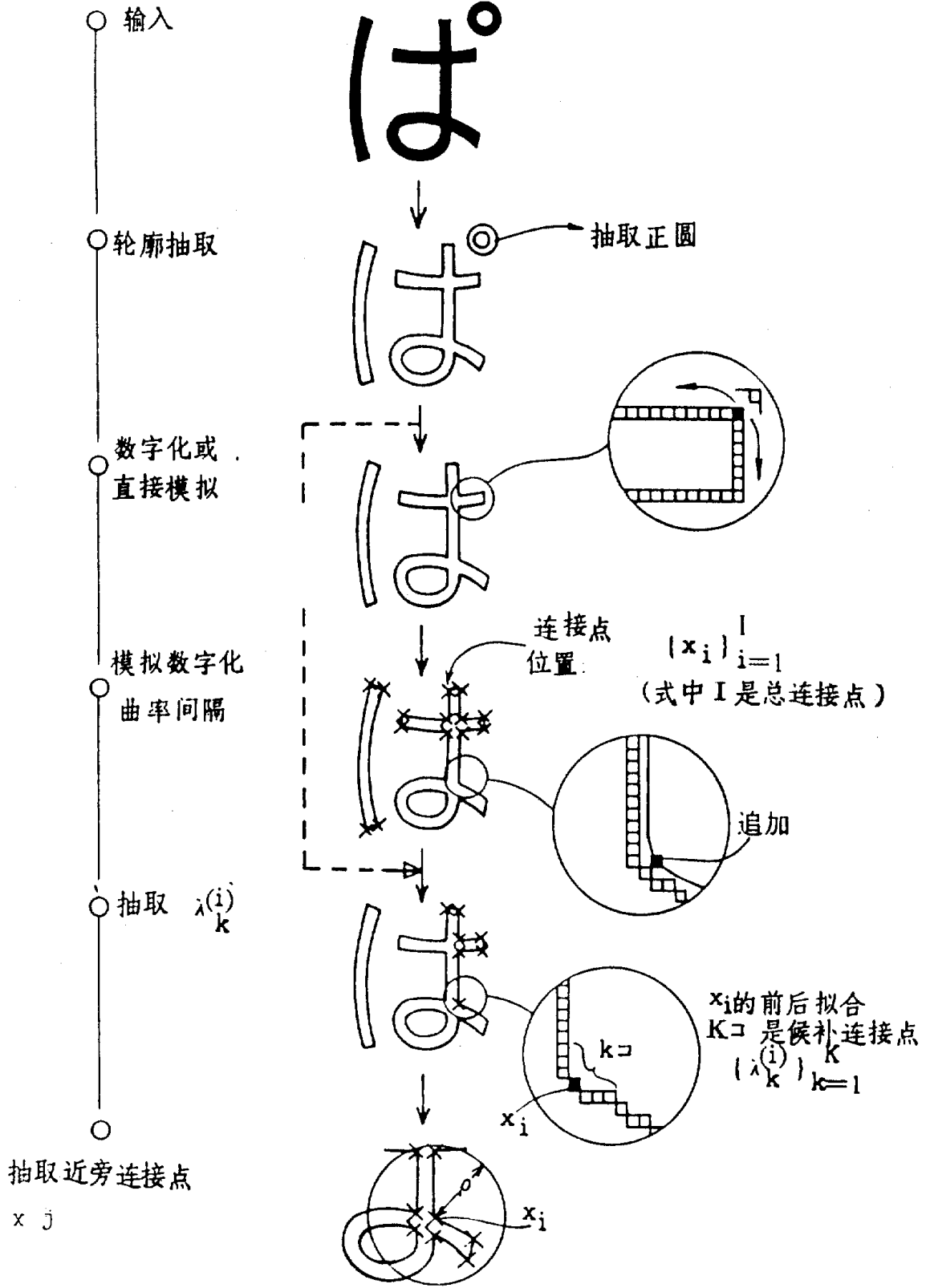


图 33

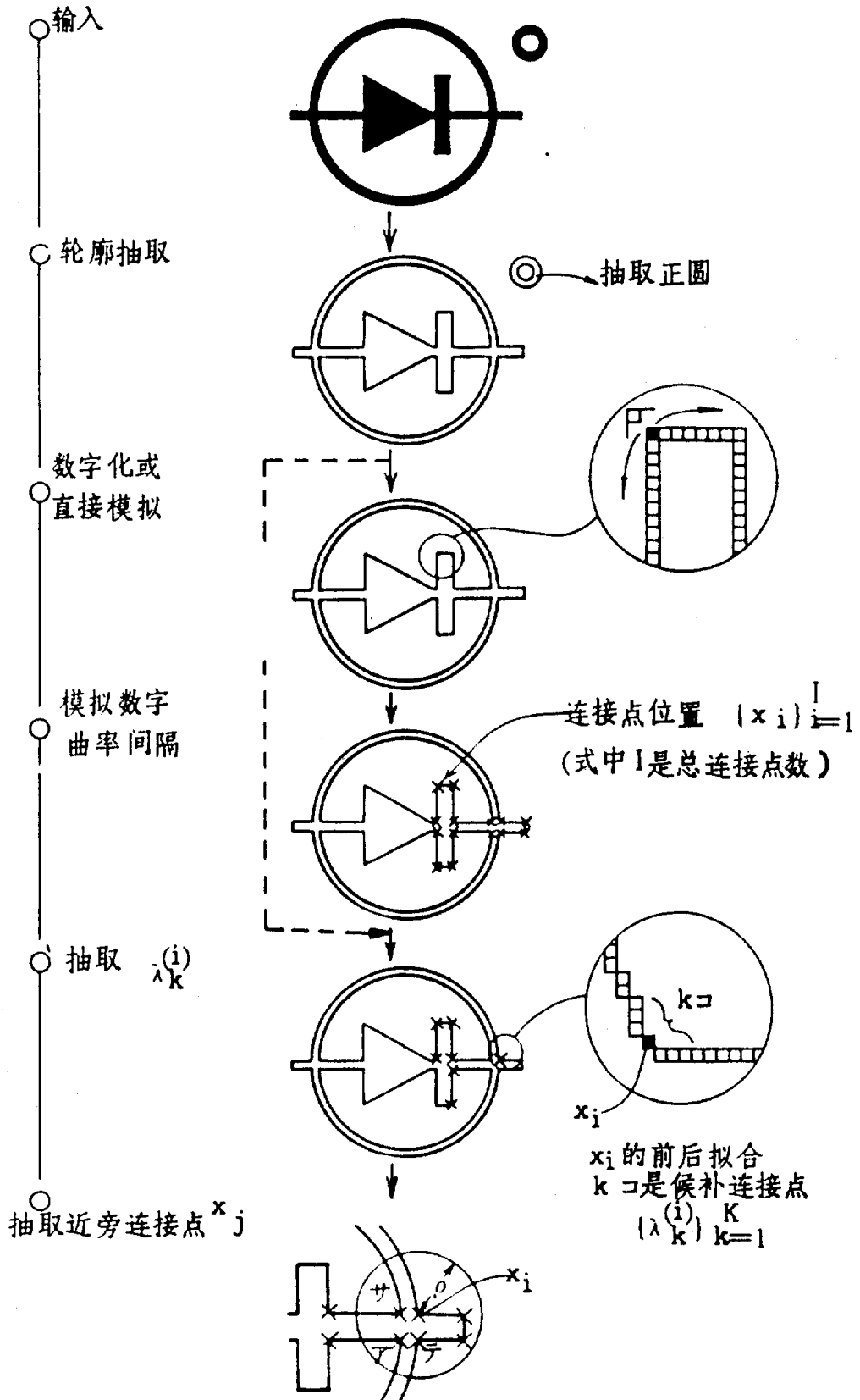


图 34

最佳连接点存储装置

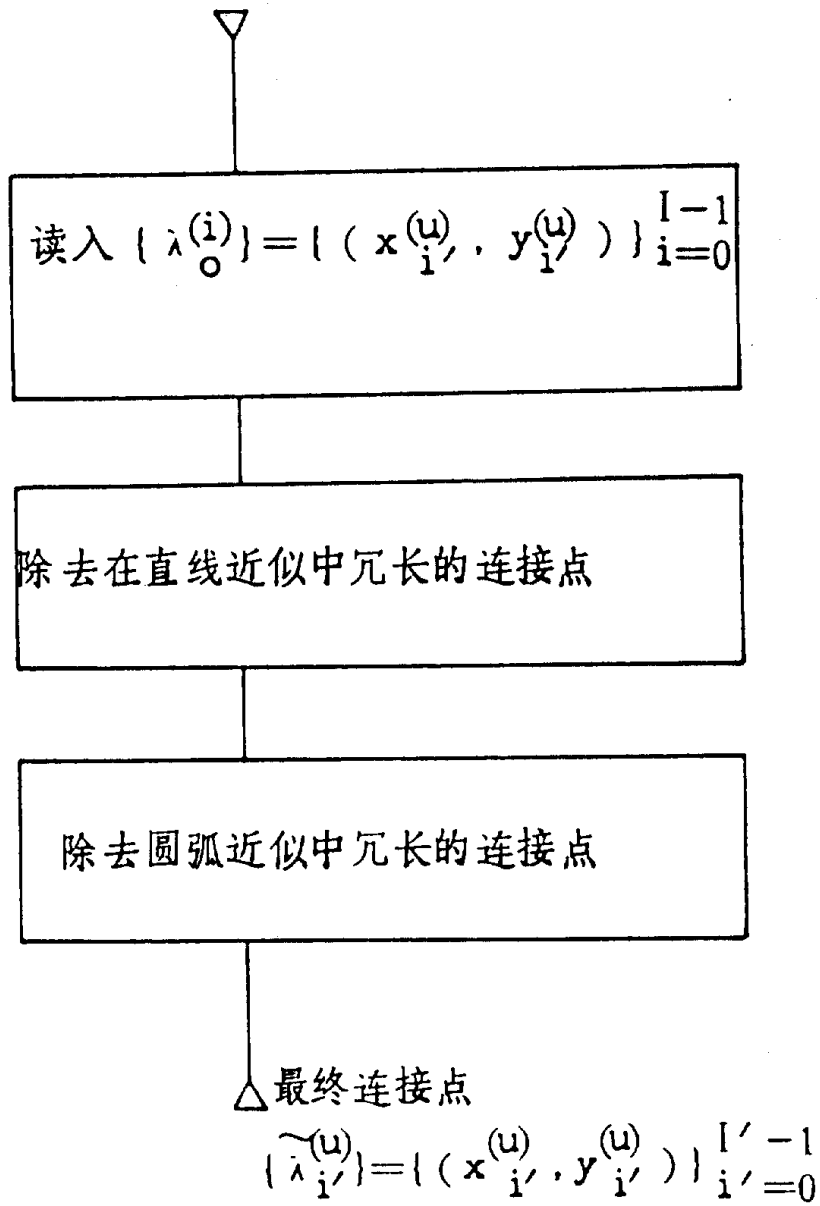


图 35

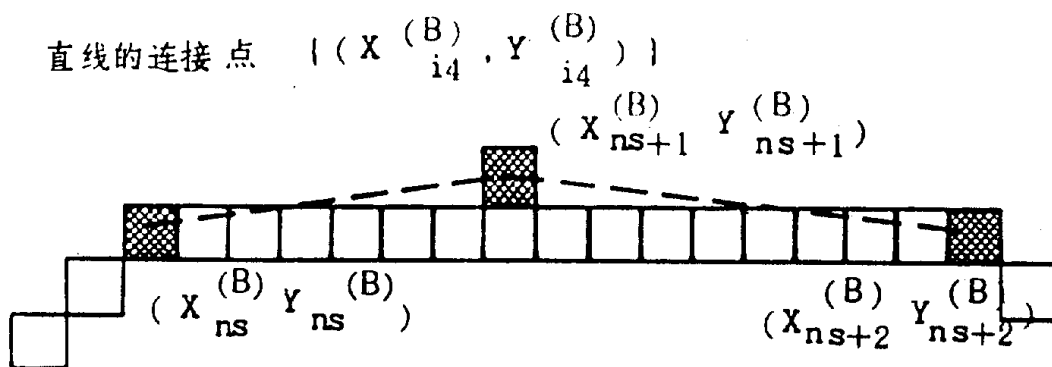


图 36

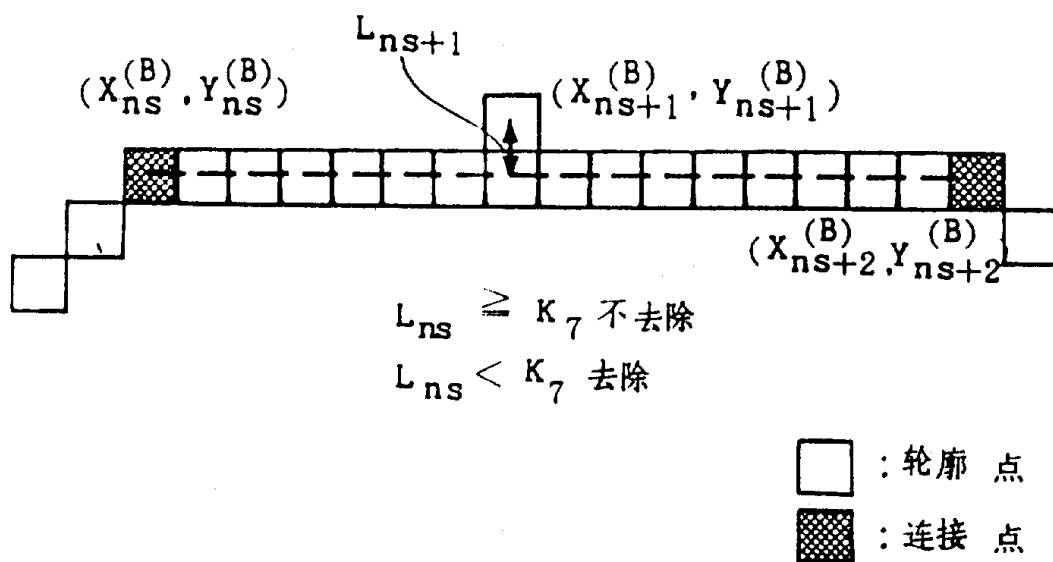


图 37

圆弧的连接点

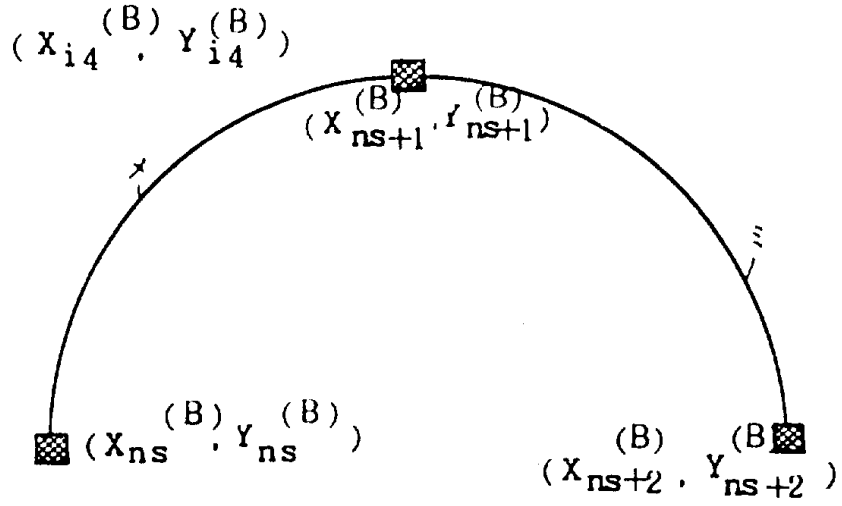
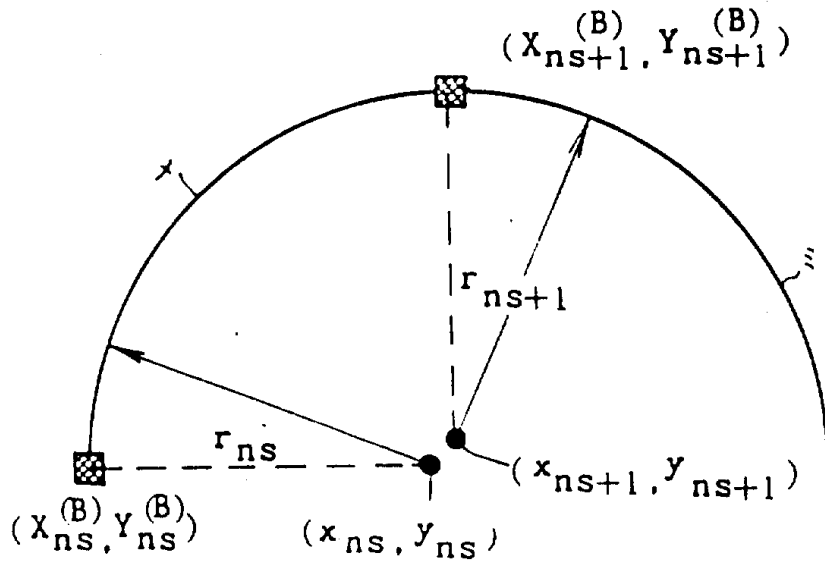


图 38



☒ : 连接点

图 39

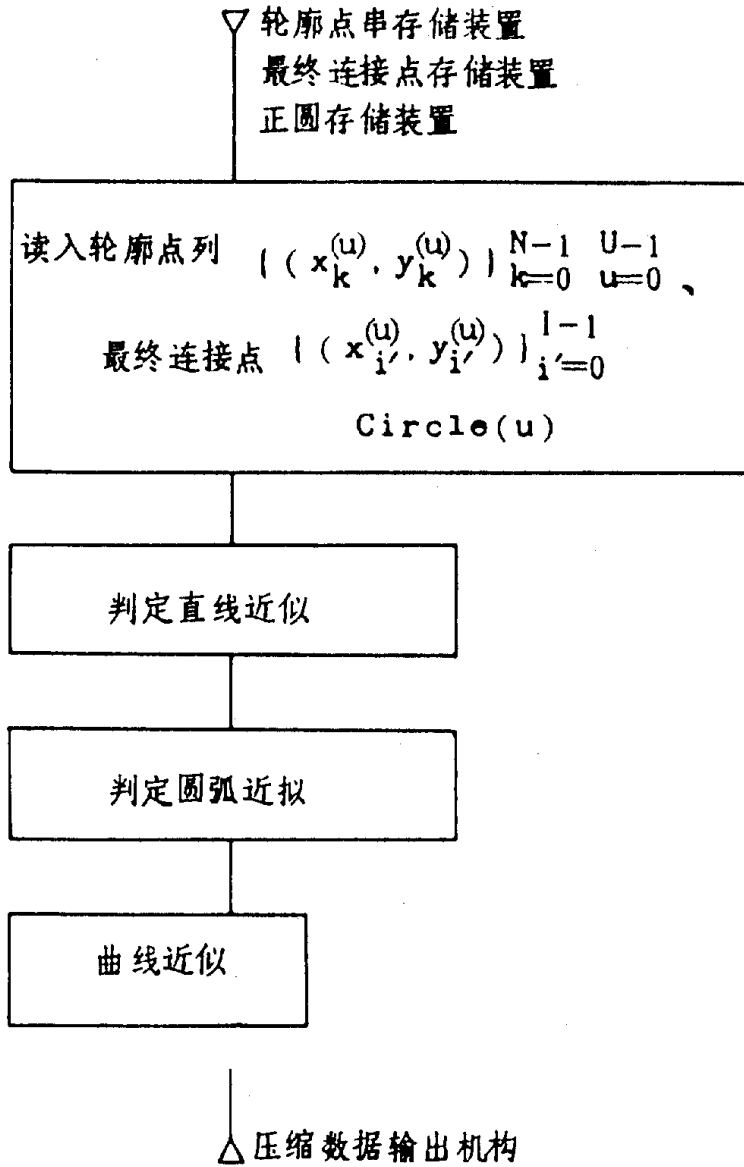


图 40

(a) 原 图 像

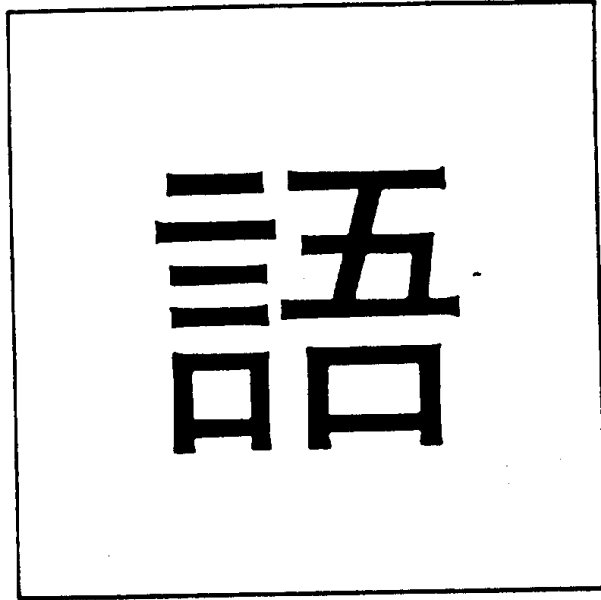


(b) 再 生 图 像



图41

(a) 原 图 像



(b) 再 生 图 像

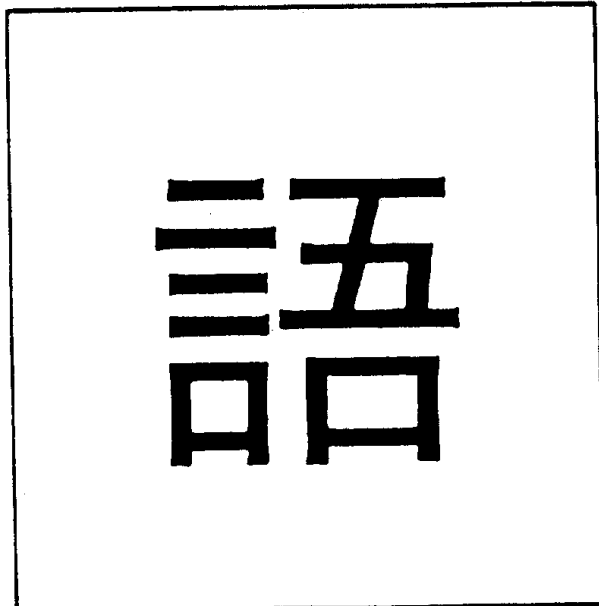


图42

(a) 原 图 像



(b) 再 生 图 像



图43
乐谱音符

(a) 原图像



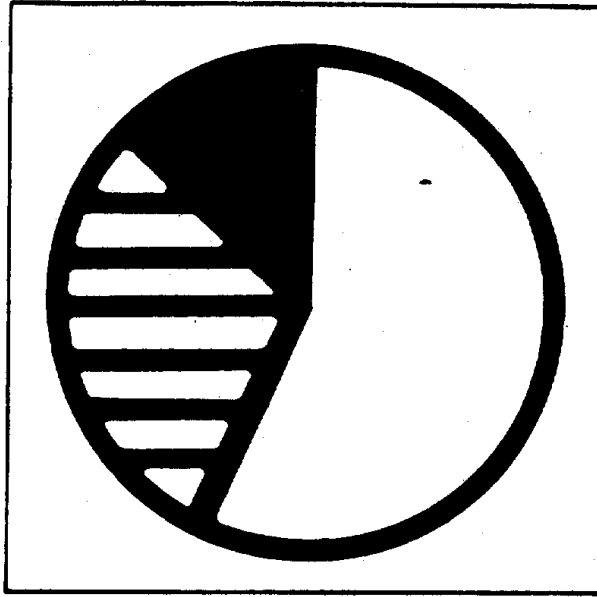
(b) 再生图像



图44

圆形图表

(a) 原图像



(b) 再生图像

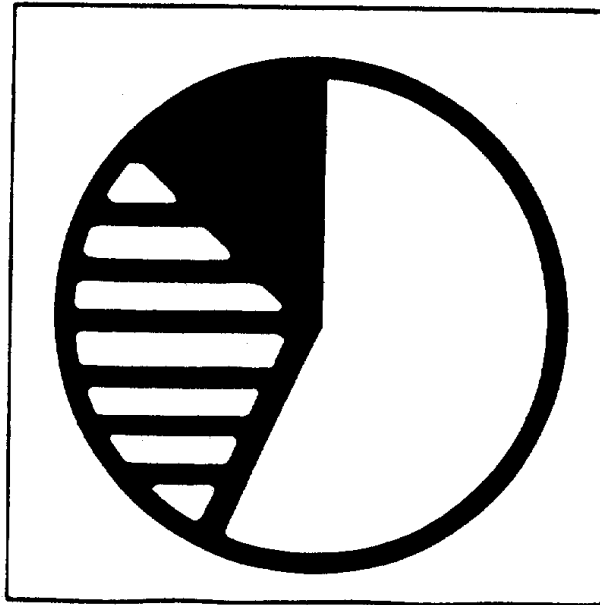
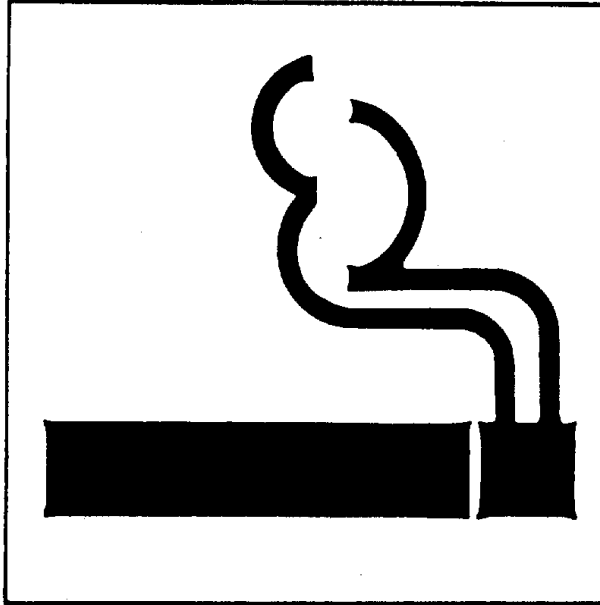


图45
香烟

(a) 原图像



(b) 再生图像

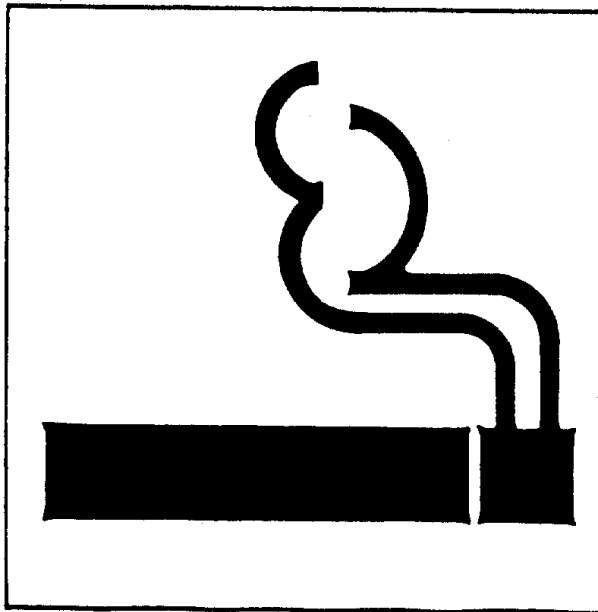
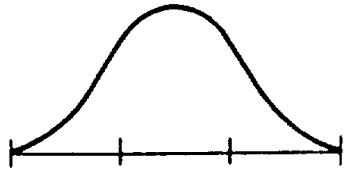


图46

(a)

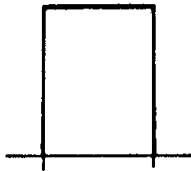
称为 2次光滑 的函数，其函数基底是



(b)

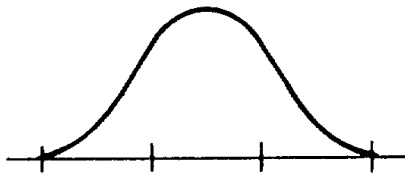
附带说明

0次



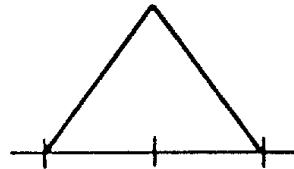
(d)

2次



(c)

1次



(e)

3次 ...

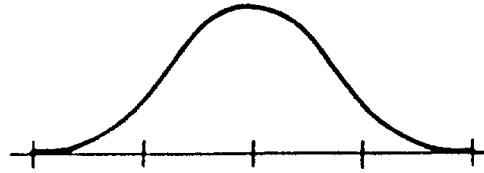
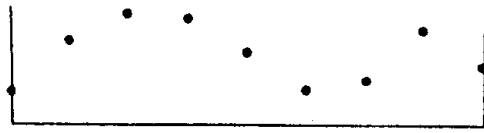
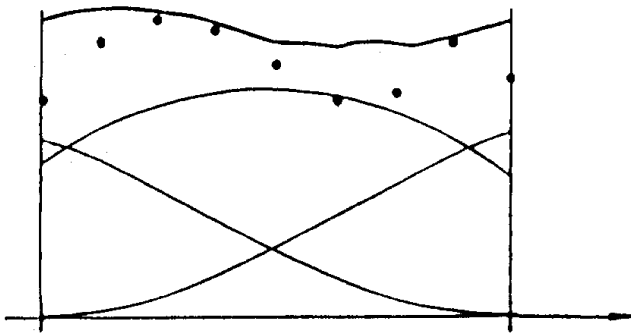


图47

(a)



(b)



(c)

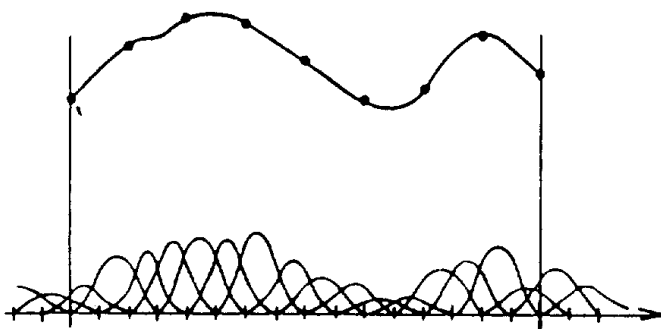


图49

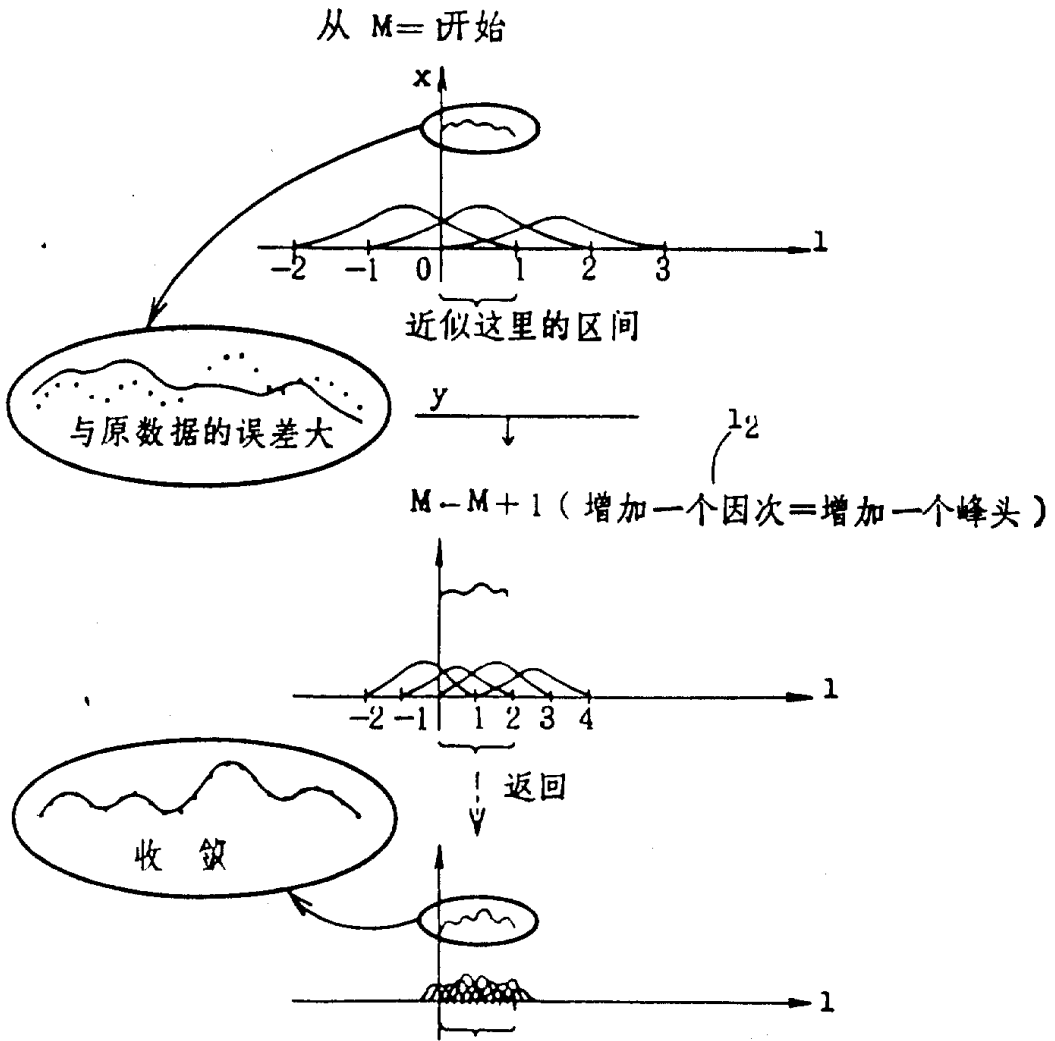
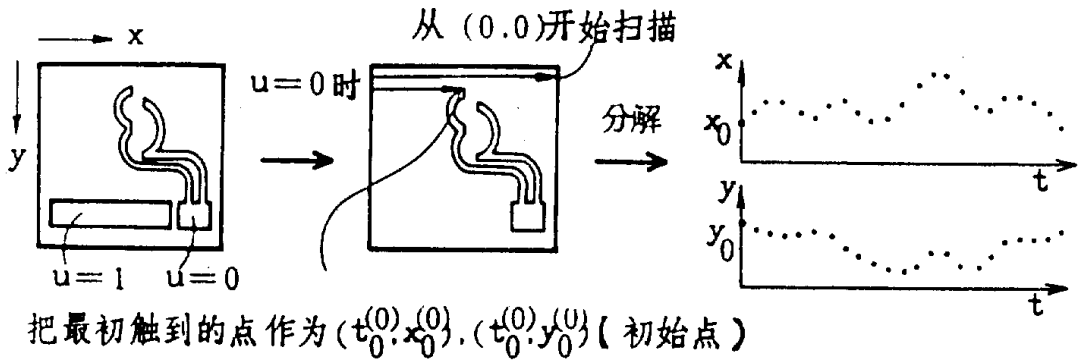
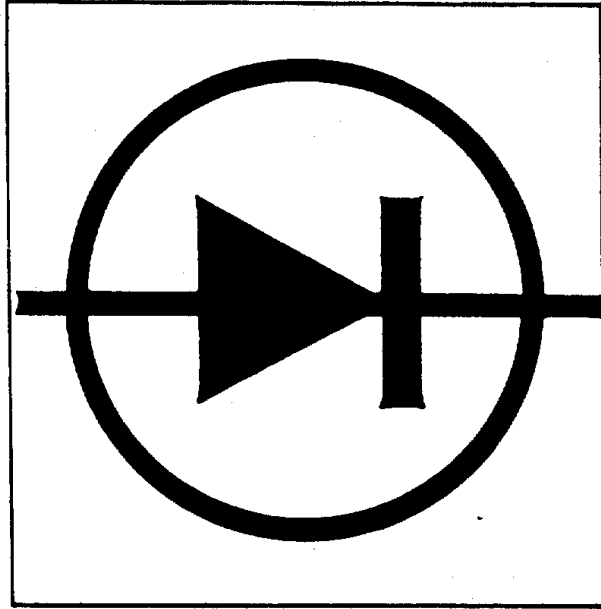


图50

二极管

(a) 原图像



(b) 再生图像

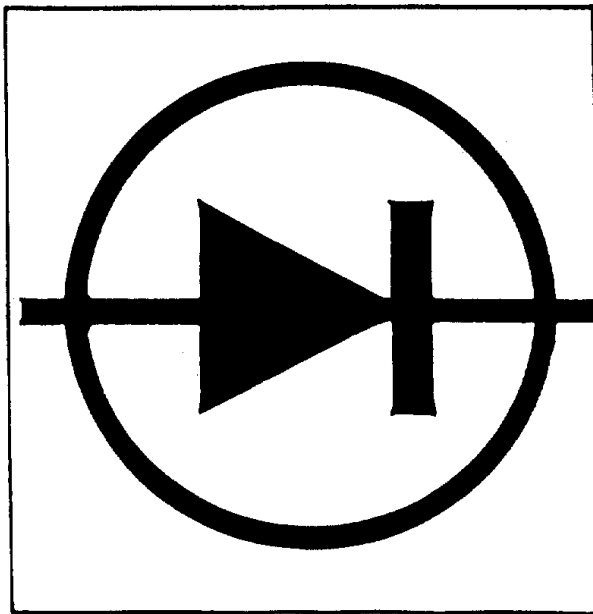


图51



图52



图53



图54

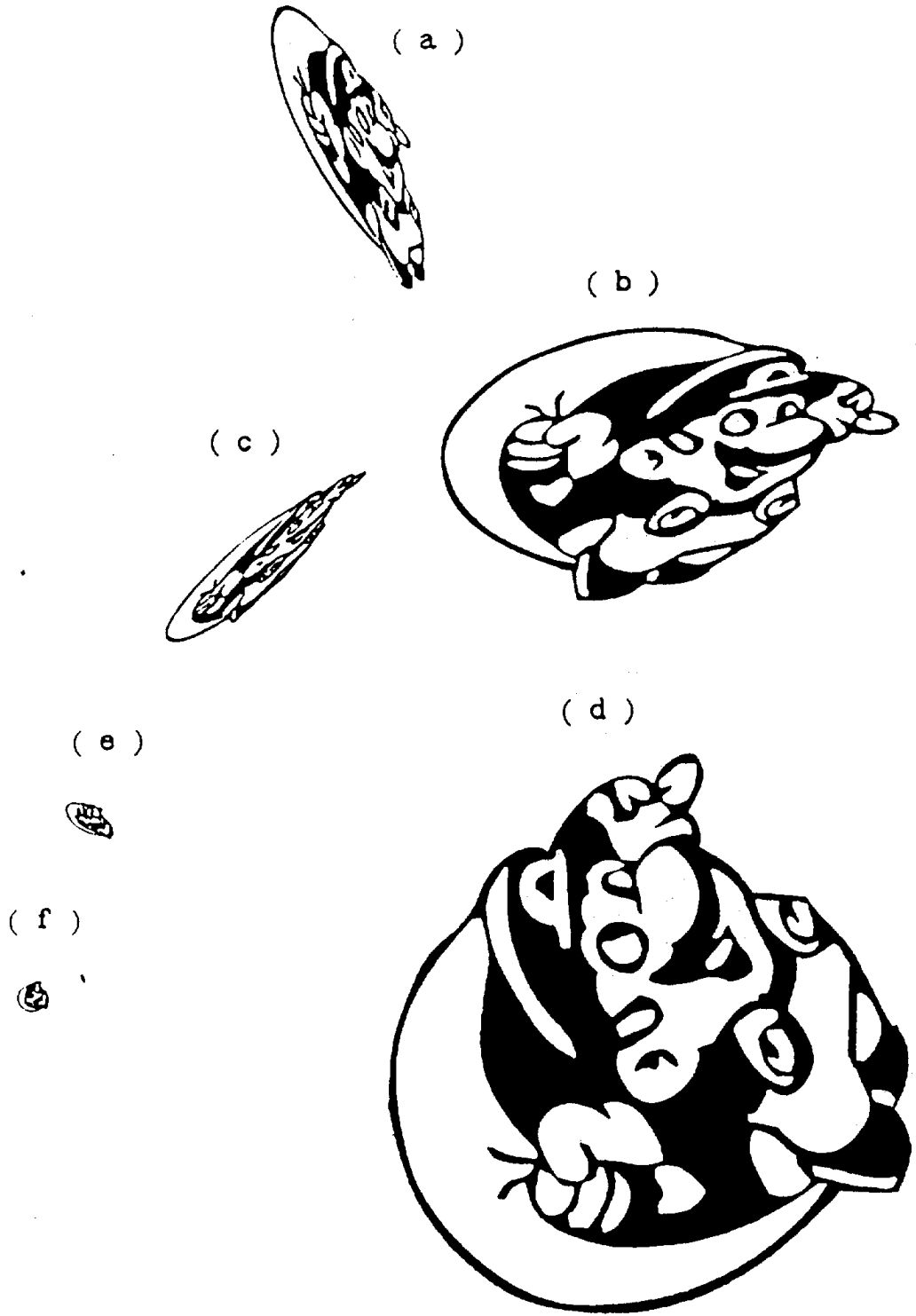


图 55

(a)



(b)



图56

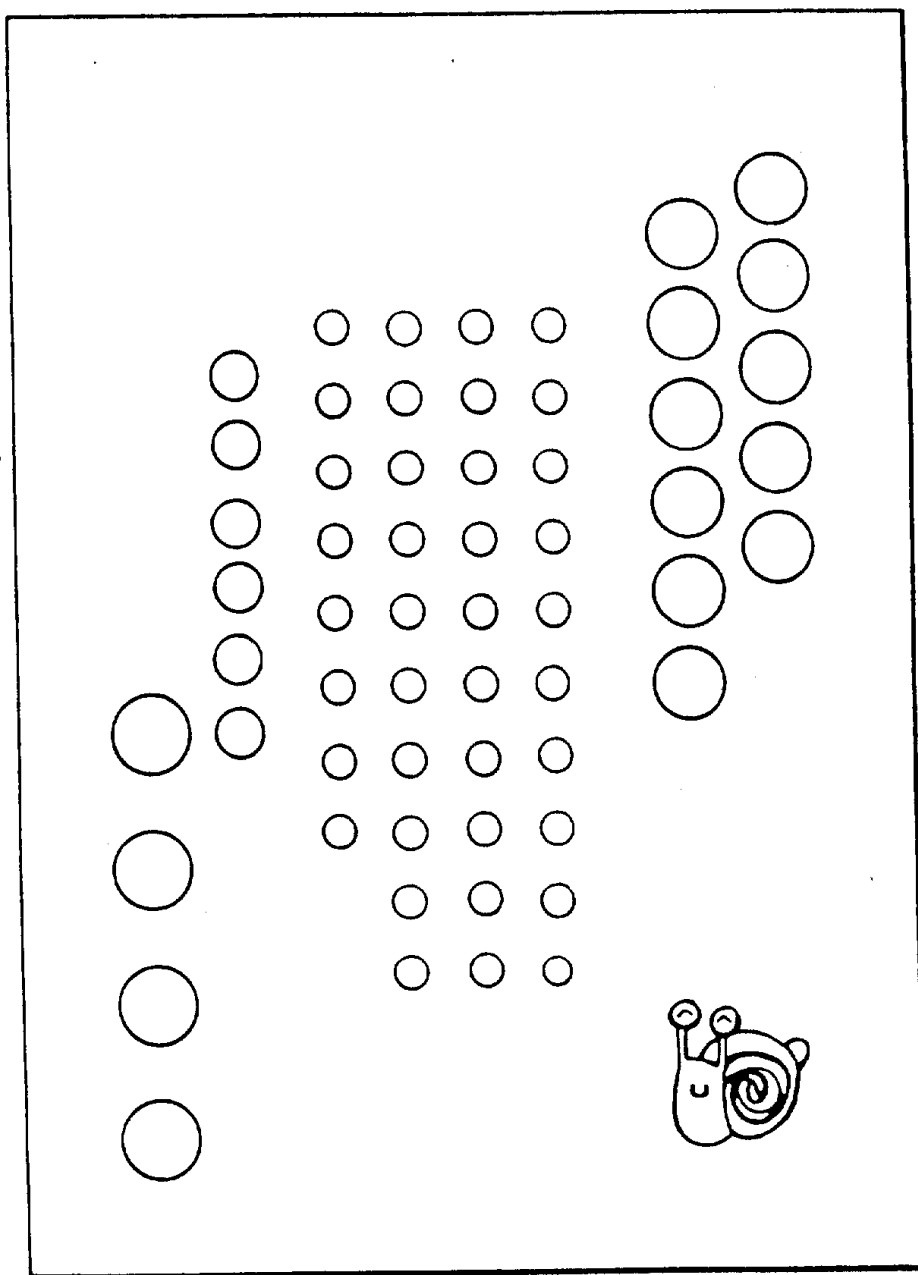


图57

