



数字图像处理学

第8章 图像分析

(第二讲)



8.2 描绘

当一幅图像被分割或确定之后，通常希望用一系列符号或某种规则来具体的描述该图像的特征，以便在进一步的识别、分析或分类中有利于区分不同性质的图像。同时，也可以减少图像区域中的原始数据量。

描绘子：把表征图像特征的一系列符号叫做描绘子。

要 求：描绘子的基本要求是它们对图像的大小、旋转、平移等变化不敏感。

也就是说，只要图像内容不变，仅仅产生几何变化，描绘图像的描绘子将是唯一的。



8.2.1 区域描绘

8.2.2 关系描绘

8.2.3 相似性描绘

8.2.4 霍夫变换



➤ 傅立叶描绘子

当一个区域边界上的点已被确定时，可以从这些点中提取信息。这些信息就可以用来鉴别不同区域的形状。假如一个区域上有 M 个点可利用，可以把这个区域看作是在复平面内，纵坐标为虚轴，横坐标为实轴。如图8—11所示。

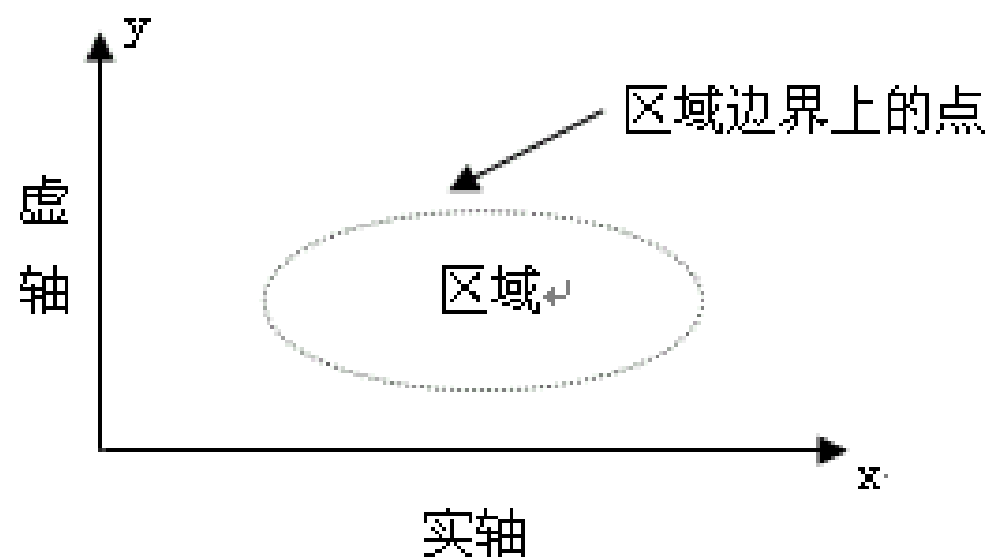


图 8—11 在复平面上区域边界的表示



在边界上要分析每一个点的坐标 (x,y)

可以用一复数来表示，即： $x + jy$ 。从边界上任一点开始，沿此边界跟踪一周就可以得到一个复数序列。这个复数序列叫做傅立叶描绘子（FD）。



因为 DFT 是可逆的线性变换，因此，在这个过程中没有信息的增益或损失。对于形状的这种频域表示作些简单的处理就可以避免对于位置、大小及方向的依赖性。当给定了任意的 FD ，用若干步骤可以使之归一化，从而不必考虑其原始形状的大小、位置及方向。



关于归一化问题可直接从 DFT 的性质中得出结论。例如，要改变轮廓大小，只要把 FD 分量乘一个常数就行了。由于傅立叶变换是线性的，它的反变换也会被乘以同样的常数。

又如，把轮廓旋转一个角度，只要把每一个坐标乘以 $\exp(j\theta)$ 就可以使其旋转 θ 角。由***DFT***的性质，在空域旋转了 θ 角度，那么在频域中也会旋转 θ 角度。关于轮廓起始点的移动，由***DFT***的周期性可以看到，在空域中有限的数字序列实际上代表周期函数的一个周期。

DFT的系数就是这个周期函数的傅立叶表示式的系数。当轮廓的起点在空域中移动时，就相当于在频域中把第 K 次频率系数乘以 $\exp(jkT)$ ，这里 T 是周期的一部分，这部分即为起始点移动的部分。实际上这就是傅立叶变换的平移性质所导致的结果。当 T 从 $0 \sim 2\pi$ 变化时，则起点将把整个轮廓点经历一次。



给定一任意轮廓的FD后，归一化就可以执行一系列步骤，使轮廓有一个标准的大小、方向和起点。

在实际执行上还要考虑到如下一些问题：

1)、如果取样不均匀将会给问题带来困难，因此，在理论上采用均匀间隔取样；



2)、其次是 FFT 的算法要求阵列长度为 2 的整数次幂，这样在采用 FFT 之前，应调整表达式的长度。为作到这一点，首先计算出轮廓的周长，然后用所希望的长度（当然应是 2 的整数次幂）去除，然后从一个起始点去追踪，所希望的 2 的幂次可以是大于序列长度的最小的 2 的幂次。



实际上，形状分析是从取样图片中取出的轮廓。这个轮廓的周长近似等于轮廓的实际周长。

如果原始图像的取样密度足够高的话，那么序列将是轮廓的很好的近似。



例如，图8—12所示的等边直角三角形，如果用 4 个取向的链码来追迹则会比较粗糙，如果用 8 个取向的链码来追迹，就得到精确得多的结果。由图8—12可见，追踪后的轮廓长度是直角边长度的 4 倍。如图8—13所示，用 8 个方向来追踪，其结果更接近实际轮廓的长度。

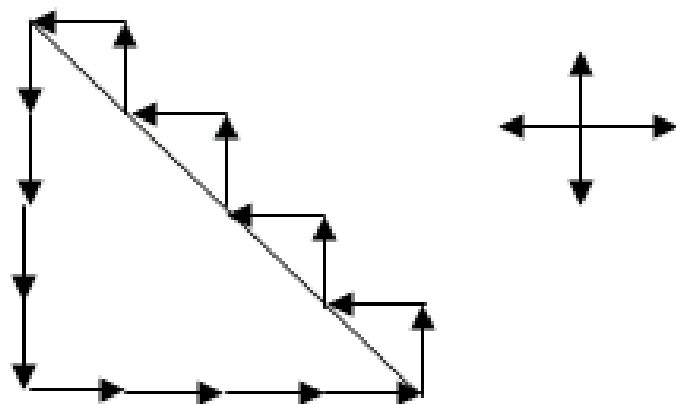


图 8—12 4个取向的链码追迹

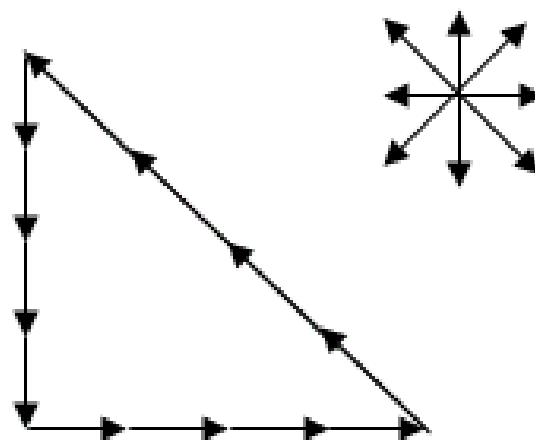


图 8—13 8方向追迹



在归一化中，为了克服噪声和量化误差带来的扰动，应选择最大幅度系数做为归一化系数。

图8—14是飞机侧影的描绘结果。这些结果是如下得到的：计算边界的NFD（应用512点）；保留最低频率的32个点而把其他的点位置0；求修改了的512阵列的傅立叶反变换，得到原始数据的近似。

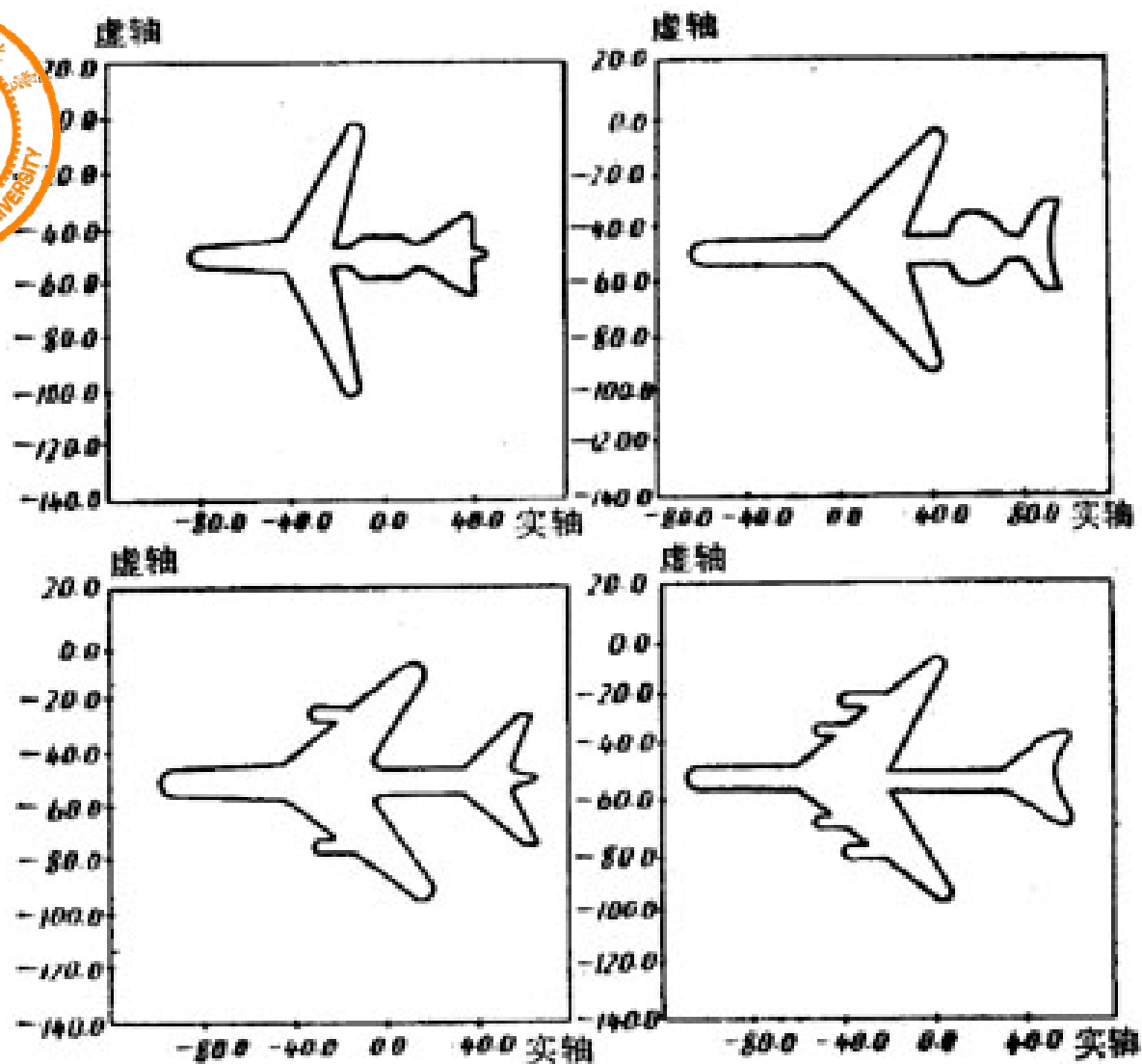


图8—14 采用傅立叶描绘子得到的外形



➤ 矩描绘子

采用傅立叶描绘子是以边界上的集合点（可用的）为基础。有时，一个区域以内部点的形式给出，那么，可用另外一种描绘子来描述。它对于图像的变换、旋转和大小变化都是恒定的，这就是矩描绘子。

设 $f(x,y)$ 是一个二维函数，可用下式来表示
($p + q$) 阶矩

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y) dx dy$$

(8—48)

式中 $p, q = 0, 1, 2, \dots$ 。

中心矩由下式表示

$$\mu_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x - \bar{x})^p (y - \bar{y})^q f(x, y) dx dy$$

式中

$$\bar{x} = \frac{m_{10}}{m_{00}}, \bar{y} = \frac{m_{01}}{m_{00}} \quad (8-49)$$

对于数字图像来说，式(8—49)可表示为下式

$$\mu_{pq} = \sum \sum (x - \bar{x})^p (y - \bar{y})^q f(x, y)$$

(8—50)

m_{00}, m_{10}, m_{01} 如下

$$m_{00} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) dx dy \quad (8-51)$$

$$m_{10} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x f(x, y) dx dy \quad (8-52)$$

$$m_{01} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} y f(x, y) dx dy \quad (8-53)$$



$$m_{00} = \sum \sum f(x, y)$$

$$m_{10} = \sum \sum xf(x, y)$$

$$m_{01} = \sum \sum yf(x, y)$$



如果假定所给图像 $f(x, y)$ 在每一点 (x, y) 处的灰度级是在 (x, y) 点的“质量”，那么就可以定义 $f(x, y)$ 的重心点 (\bar{x}, \bar{y}) ，

其中：



$$\bar{x} = \frac{\int \int_{-\infty}^{\infty} x f(x, y) dx dy}{\int \int_{-\infty}^{\infty} f(x, y) dx dy} \quad (8-54)$$

$$\bar{y} = \frac{\int \int_{-\infty}^{\infty} y f(x, y) dx dy}{\int \int_{-\infty}^{\infty} f(x, y) dx dy} \quad (8-55)$$

因此有 $\bar{x} = \frac{m_{10}}{m_{00}} \quad \bar{y} = \frac{m_{01}}{m_{00}} \quad (8-56)$

由上边各式可得到三阶中心矩如下：

$$\begin{aligned} \mu_{10} &= \sum_x \sum_y (x - \bar{x})^1 (y - \bar{y})^0 f(x, y) \\ &= m_{10} - \frac{m_{10}}{m_{00}} (m_{00}) = 0 \end{aligned} \quad (7-57)$$

$$\begin{aligned}
\mu_{11} &= \sum_x \sum_y (x - \bar{x})^1 (y - \bar{y})^1 f(x, y) \\
&= m_{11} - \frac{m_{10}m_{01}}{m_{00}}
\end{aligned}
\tag{8—58}$$

$$\begin{aligned}
\mu_{20} &= \sum_x \sum_y (x - \bar{x})^2 (y - \bar{y})^0 f(x, y) \\
&= m_{20} - \frac{2m_{10}^2}{m_{00}} + \frac{m_{10}^2}{m_{00}} = m_{20} - \frac{m_{10}^2}{m_{00}}
\end{aligned}
\tag{8—59}$$

$$\begin{aligned}
\mu_{02} &= \sum_x \sum_y (x - \bar{x})^0 (y - \bar{y})^2 f(x, y) \\
&= m_{02} - \frac{m_{01}^2}{m_{00}}
\end{aligned}
\tag{8—60}$$

$$\begin{aligned}
\mu_{30} &= \sum_x \sum_y (x - \bar{x})^3 (y - \bar{y})^0 f(x, y) \\
&= m_{30} - 3\bar{x}m_{20} + 2m_{10}\bar{x}^2
\end{aligned}
\tag{8—61}$$



$$\begin{aligned}\mu_{12} &= \sum_x \sum_y (x - \bar{x})^1 (y - \bar{y})^2 f(x, y) \\ &= m_{12} - 2\bar{y}m_{11} - \bar{x}m_{02} + 2\bar{y}^2 m_{10}\end{aligned}\quad (8-62)$$

$$\begin{aligned}\mu_{21} &= \sum_x \sum_y (x - \bar{x})^2 (y - \bar{y})^1 f(x, y) \\ &= m_{21} - 2\bar{x}m_{11} - \bar{y}m_{20} + 2\bar{x}^2 m_{01}\end{aligned}\quad (8-63)$$



$$\begin{aligned}\mu_{03} &= \sum_x \sum_y (x - \bar{x})^0 (y - \bar{y})^3 f(x, y) \\ &= m_{03} - 3\bar{y}m_{02} + 2\bar{y}^2 m_{01}\end{aligned}$$

(8—64)

概括起来有如下一些结果：

$$\mu_{00} = m_{00}$$

$$\mu_{01} = 0$$

$$\mu_{10} = 0$$

$$\mu_{20} = m_{20} - \bar{x}m_{10}, \mu_{02} = m_{02} - \bar{y}m_{01}$$

$$\mu_{11} = m_{11} - \bar{y}m_{10}$$

$$\mu_{30} = m_{30} - 3\bar{x}m_{20} + 2m_{10}\bar{x}^2$$



$$\mu_{12} = m_{12} - 2\bar{y}m_{11} - \bar{x}m_{02} + 2\bar{y}^2m_{10}$$

$$\mu_{21} = m_{21} - 2\bar{x}m_{11} - \bar{y}m_{20} + 2\bar{x}^2m_{01}$$

$$\mu_{03} = m_{03} - 3\bar{y}m_{02} + 2\bar{y}^2m_{01}$$



定义归一化中心矩为：

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}} \quad (8-65)$$

$$\gamma = \frac{p+q}{2} \quad (8-66)$$

利用第二阶和第三阶矩可导出七个不变矩组：

$$\phi_1 = \eta_{20} + \eta_{02}$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} + \eta_{03})^2$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$



$$\begin{aligned}\varphi_5 = & (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12}) \left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + 3\eta_{03})^2 \right] \\ & + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) \left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2 \right]\end{aligned}$$

(8—67)



$$\varphi_6 = (\eta_{20} - \eta_{02}) \left[(\eta_{30} + \eta_{12})^2 - (\eta_{12} + \eta_{03})^2 \right] \\ + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$$

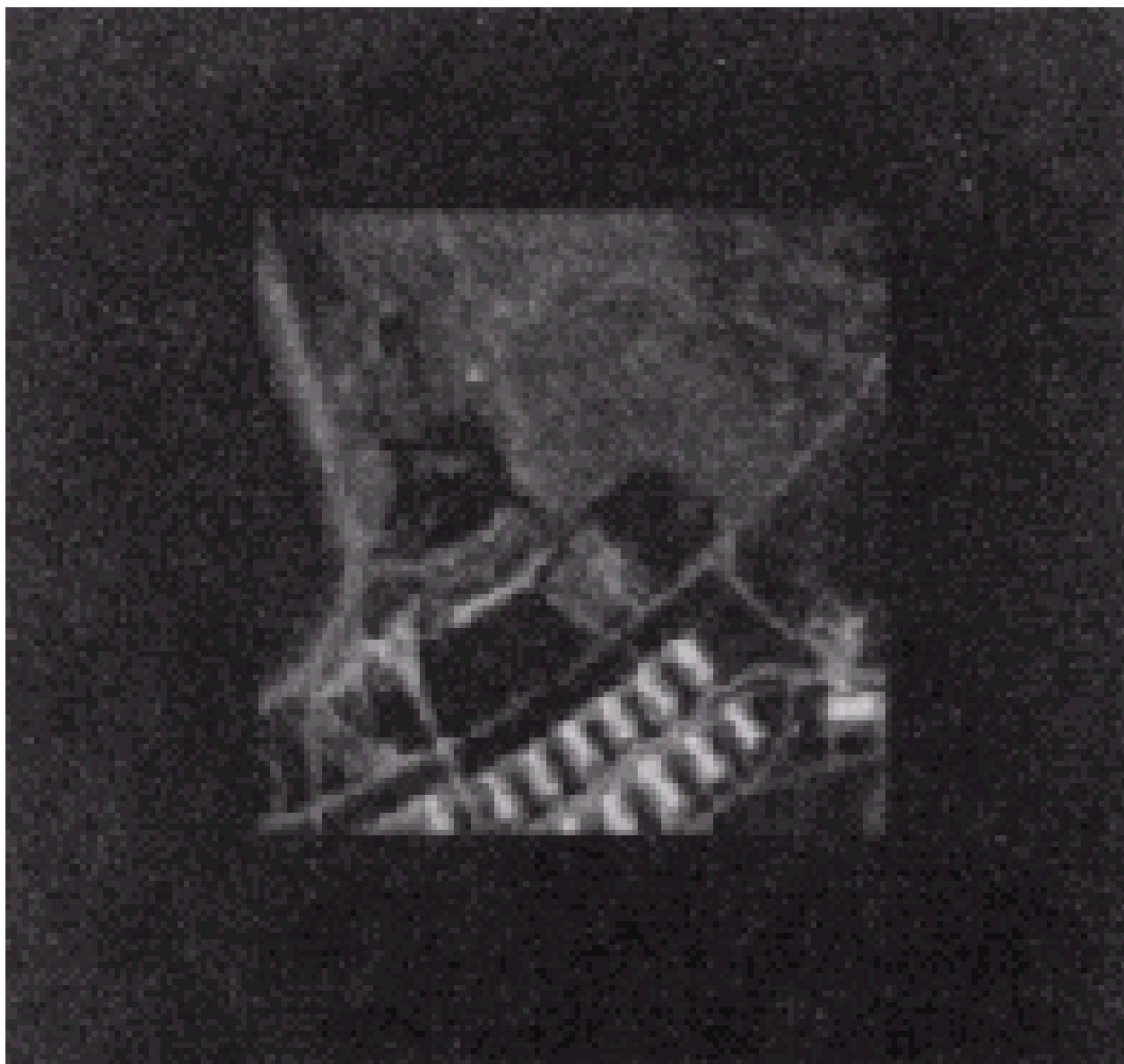


$$\begin{aligned}\varphi_7 = & (3\eta_{12} - \eta_{30})(\eta_{30} + \eta_{12}) \left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2 \right] \\ & + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) \left[3(\eta_{30} + \eta_{12})^2 - (\eta_{12} + \eta_{03})^2 \right]\end{aligned}$$



Hu[1962]已经证明了这个矩组对于
平移、旋转和大小比例变化都是不变的，
因此用它们可以描绘一幅给定的图像。

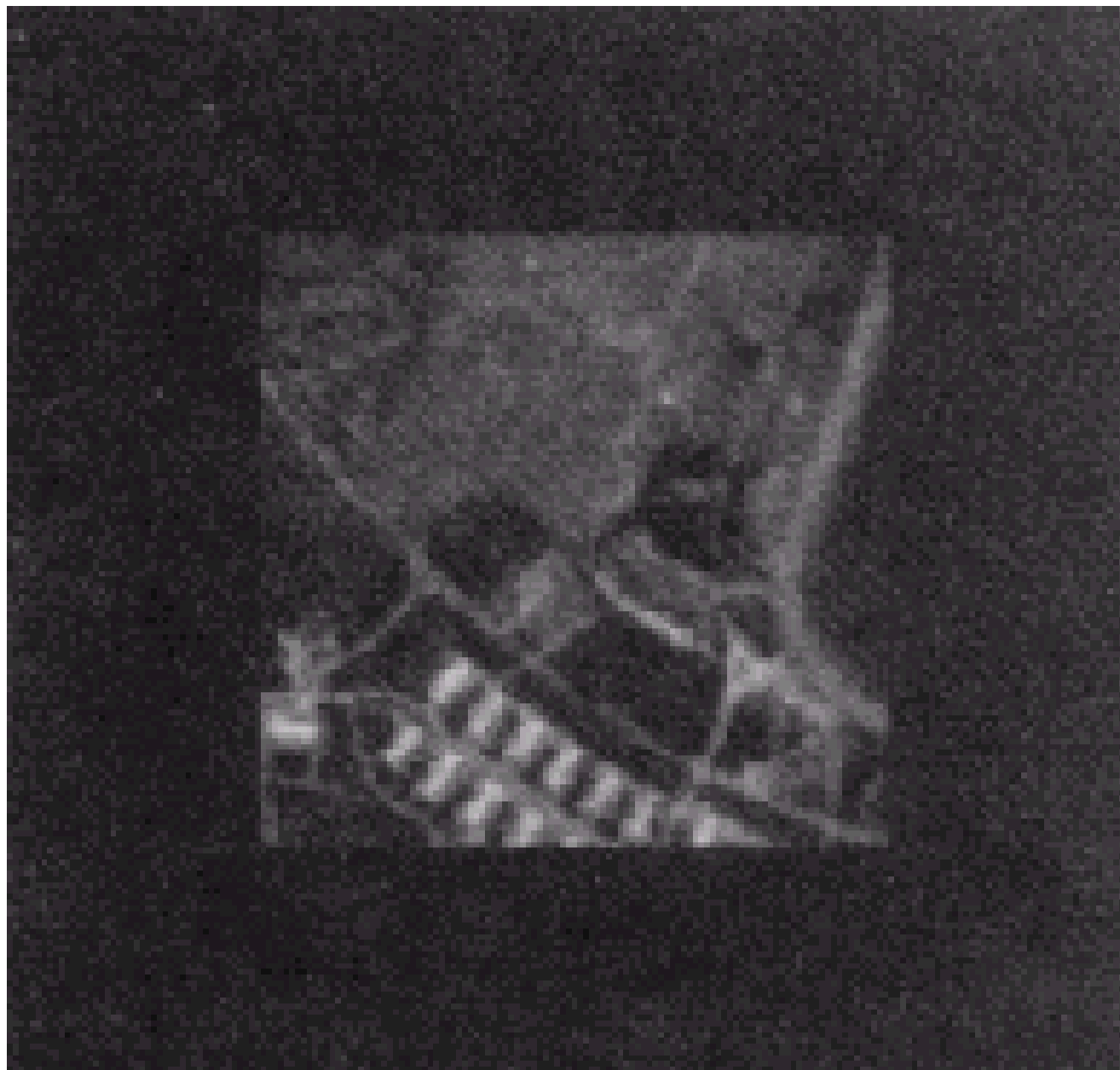
原像



缩小
一倍



镜像



旋转
2度



旋转
45度



图像的矩不变量

不变量	原值	一半尺寸	映像	旋转 2 度	旋转 45 度
φ_1	6.249	6.226	6.919	6.253	6.318
φ_2	17.180	16.954	19.955	17.270	16.803
φ_3	22.655	23.531	26.689	22.836	19.724
φ_4	22.919	24.236	26.901	23.130	20.437
φ_5	45.749	48.349	53.724	46.136	40.525
φ_6	31.830	32.916	37.134	32.068	29.315
φ_7	45.569	48.343	53.590	46.017	40.470



➤ 拓扑描绘子

拓扑学是研究图形性质的理论。拓扑特性可用于描绘图像平面区域。有些图形只要不撕裂或连结，其拓扑性质并不受形变的影响。



图8—15是带有两个孔的图形，如果把区域中孔洞数做为拓扑描绘子，显然这个性质不受伸长或旋转变换的影响，但是，如果撕裂或折叠时孔洞数就要变化了。

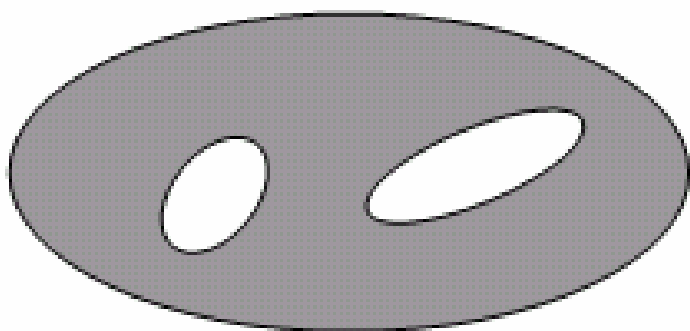


图 8—15 有两个孔的图形

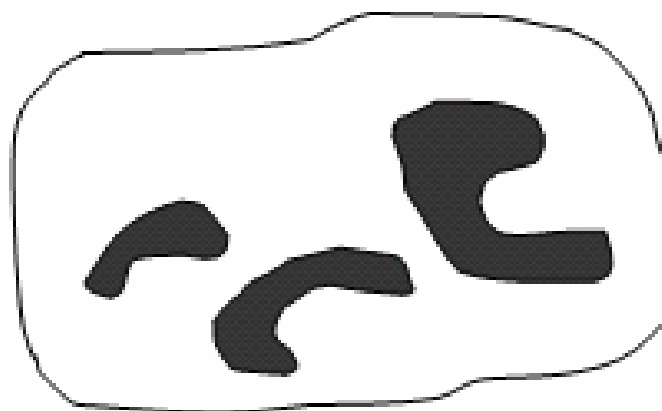


图 8—16 有三个连结部分的图形



区域描绘的另一种有用的拓扑特性是连接部分的个数。一个集合的连接部分就是它的最大子集，在这个子集中的任何两点都可以用一条完全在子集中的曲线加以连接。图8—16所示的图形就有三个连接部分。



如果一幅图像的孔洞数为 H ，连接部分为 C ，
则欧拉数的定义如下式所示。

欧拉数：

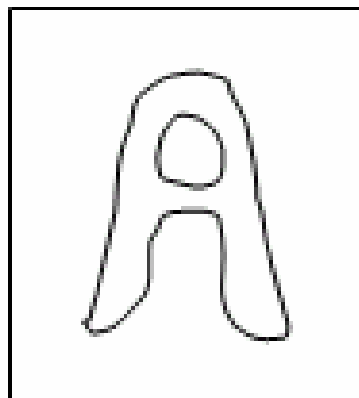
$$E = C - H \quad (8-68)$$

也是拓扑特性之一。

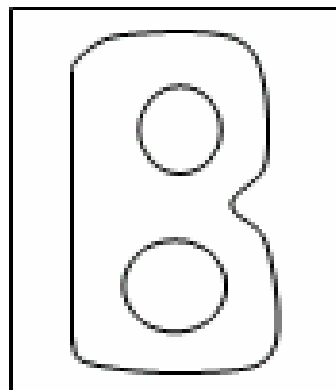


如图8—17 (a) 所示图形有一个连接部分和一个孔，
所以它的欧拉数为0；而图 (b) 有一个连接部分和
二个孔，所以它的欧拉数为-1。

$$E = C - H$$



(a)



(b)

图8—17 具有欧拉数为0和-1的图形

由直线表示的区域，按照欧拉数有一个简单的描述。如图8—18所示的多角网络，把这样的网络内部区域分成面和孔。如果设顶点数为 W ，边缘数为 Q ，面数为 F ，将得到下列关系，这个关系称为欧拉公式。

$$W - Q + F = C - H \quad (8—69)$$

即： $W - Q + F = C - H = E$

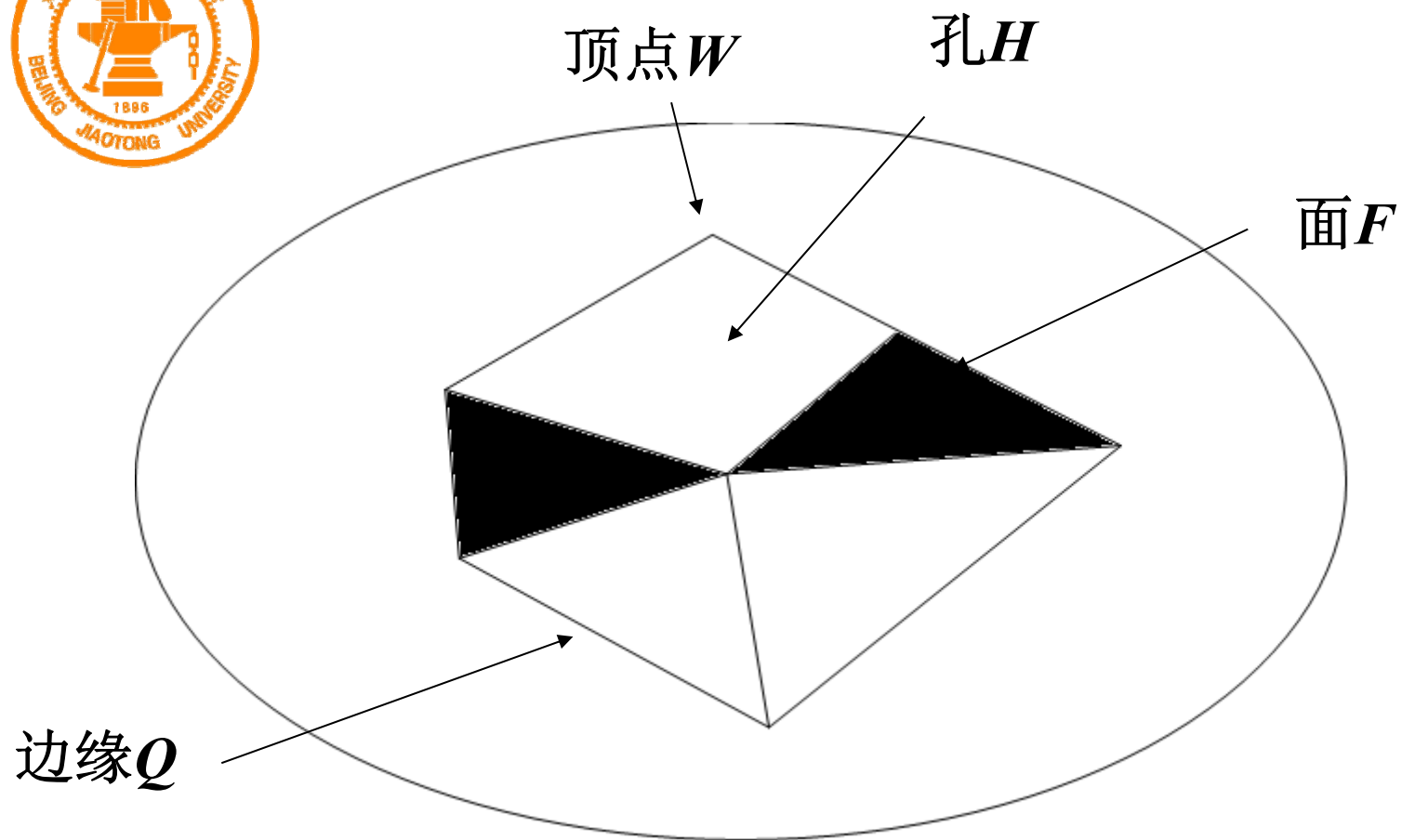


图8—18 包含多角网络的区域

在图8—18的多角网络中，有 7 个顶点、11 条边、
2 个面、1 个连接区、3 个孔、因此，由式（7—69）
可得到：

$$7-11+2=1-3=-2。$$

拓扑的概念通常在图像中确定特征区域很有用。



8.2.1 区域描绘

8.2.2 关系描绘

8.2.3 相似性描绘

8.2.4 霍夫变换



如果图像已经被分割为区域或部分，则图像描绘的下一步任务就是如何把这些元素组织成为有意义的关系结构。结构描绘一般是以文法概念为基础的。



例如，从一幅图像中已分割出图8—19所示的阶梯形结构，要用某种方法来描绘它，首先要定义一些基本元素，然后再定义一个重写规则就可以描绘出此阶梯形结构。图中(a)是阶梯结构；(b)是基本元素；(c)是编码结构。

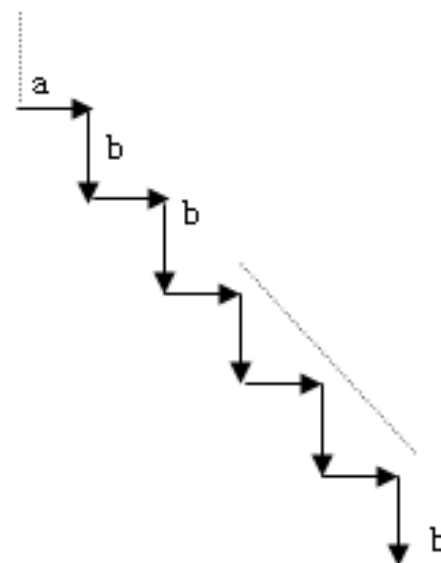
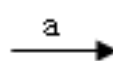
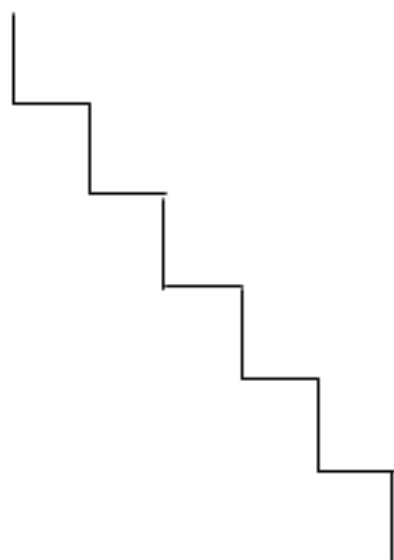


图8—19 阶梯形结构之描绘



在描绘过程中规定基本元素为 a 和 b ，重写规则如下：

$$S \rightarrow aA$$

$$A \rightarrow bS$$

$$A \rightarrow b$$



这里 S 和 A 是变量，元素 a 和 b 是常量。第一个规则说明 S 可以用基本元素 a 和变量 A 来代替，变量 A 可以用 b 和 S 来代替，也可以用 b 来代替。



如果用 bS 来代替 A ，则可以重复第一个规则的步骤。如果用 b 来代替，则步骤终止。这里假定都用 S 为起始点，第一个元素后面总是 b 。由上例可见只需三条重写规则就可以产生无穷多的相似结构。



➤ 串文法和语言

图8—19说明的编码结构是由符号的连接串组成的。这种符号串可以引用形式语言的概念来处理。



形式语言起源于1950年。诺姆、乔姆斯基用数学模型研究了文法，其目的是研究一种计算机文法，然后用这个文法去研究自然语言，以便计算机在翻译和解答问题的过程中解释自然语言。



关于形式语言的研究和应用已渗透到编译设计、
计算机语言、自动机理论及模式识别和图像处理
领域中了。

首先，介绍一些最基本的定义：

定义： V 为任何有限的符号集合；

在 V 范围内的一个句子、一串字符或字都是由集合中的符号组成的任何有限长度的串。

例如，给定 $V = \{0, 1\}$ ，则

$\{0, 1, 00, 01, 11, 000, 001, \dots\}$ 都是有效的句子。



定义：没有符号的句子为空句子，用 λ 来代表。

用 V^* 代表由 V 中的符号组成的所有句子集合，其中包括空句子。

V^+ 代表 $(V^* - \lambda)$ 的句子集合。

例如，字母 $V = \{a, b\}$

$$V^* = \{\lambda, a, b, ab, aa, ba, \dots\}$$

$$V^+ = \{a, b, ab, aa, ba, \dots\}$$

语言是在字母范围内句子的任意集合。



形式语言理论主要研究文法及其性质。

串文法：

串文法（或叫简单文法）是四元的，即：

$$G = (V_N, V_T, P, S)$$



其中：

V_N 为非终端符集合（变量）；

V_T 为终端符的集合（常量）；

P 为产生式或重写规则集合；

S 为起始符或根符号。



假定 S 属于集合 V_N , 并且 V_N 和 V_T 是不相交的集合, 字母 V 是 V_N 和 V_T 的合集。

由形式串文法 G 产生的语言记作 $L(G)$,
这个语言就代表了一个模式。

由字符产生的语言 $L(G)$ 满足两个条件:

- ① 每一串只由终端符组成;
- ② 每一串都由 S 开始并用由 P 决定的产生式来生成。

例如，有文法

$$G = (V_N, V_T, P, S)$$

$$V_N = \{S\}$$

$$V_T = \{a, b\}$$

$$P = \{S \rightarrow aSb, \quad S \rightarrow ab\}$$



如果把第一个产生式用 $m-1$ 次，然后再用第二个产生式，由此可产生下列语言：

$$S \rightarrow aSb \rightarrow aaSbb \rightarrow a^3 Sb^3$$

$$\rightarrow a^{m-1} Sb^{m-1} \rightarrow a^m b^m$$



显然，这个文法产生的语言看作仅仅由这种形式的串组成，特定的串长取决于 m 。 m 可以是任意整数，可以把 $L(G)$ 表达为下面的形式：

$$L(G) = \{a^m b^m \mid m \geq 1\}$$

这个简单的文法可以用来产生无限多串组成的语言。

➤ 文法的类型

在讨论文法类型之前，为了便于叙述，规定所使用的符号如下：

非终端符 V_N 用大写英文字母表示，

如： S, A, B, C, \dots ;



终端符 V_T 用小写英文字母表示,

如: a, b, c, \dots ;



终端字符串用英文字母表后边的小写字母来表示，如 v, w, x, y, \dots ;

终端和非终端的混合字符串用小写的希腊字母来表示，如 $\alpha, \beta, \gamma, \delta \dots$ 。



我们把产生式的一般形式为 $P: \alpha \rightarrow \beta$
的文法叫短语结构文法，一般根据加于产生式的约
束条件而把文法分成如下几类：



(1) 无约束文法

这种文法的产生式为 $P: \alpha \rightarrow \beta$,

其中箭头的左右端可以是任意形式的链。

(2) 上下文有关的文法

这种文法的 产 生 式 为:

$$P: \alpha_1 A \alpha_2 \rightarrow \alpha_1 \beta \alpha_2$$

其中:

$$A \in V_N;$$

$$\alpha_1, \alpha_2, \beta \in V^*; \beta \neq \lambda$$



这种文法规定只有当 A 出现在 α_1 和 α_2 串的前后关系 $\alpha_1 A \alpha_2$ 中时，才允许用串 β 来代替非终端符 A 。

(3) 上下文无关的文法

此文法的产生式为 $P: A \rightarrow \beta$

其中 $A \in V_N$, $\beta \in V^*$ 。这个文法并不考虑出现 A 的上下文就可以用串 β 去代替 A 变量。

(4) 正则文法

这种文法也称为有限状态文法。

它的产生式为: $P: A \rightarrow aB$

或 $P: A \rightarrow a$

这里 $A, B \in V_N$, $a \in V_T$, A, B, a 均是单个符号。



还可以选择另外一种产生式，即

$P: A \rightarrow Ba$ 和 $P: A \rightarrow a$ 。

当然二种产生式中只能选一种，而不可同时选用。

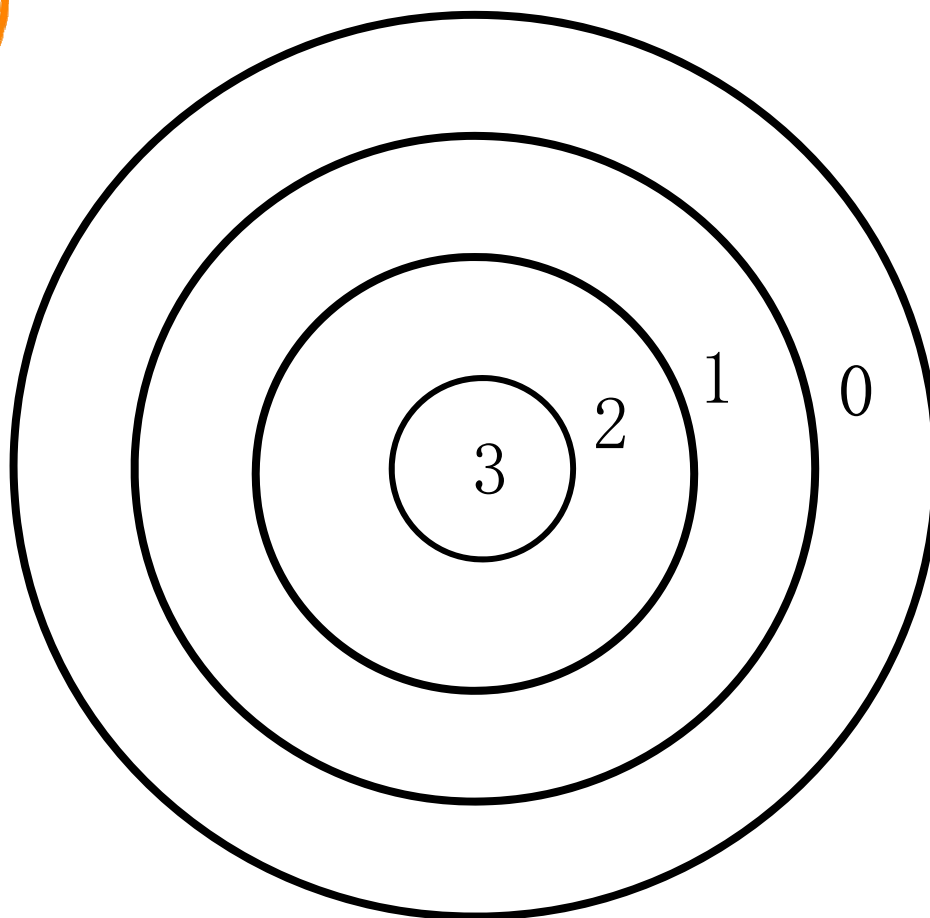


上述四类文法有时依次称为 0 型， 1 型， 2 型和 3 型文法。

由它们产生的语言分别称为 0 类型语言， 1 类型语言， 2 类型语言及 3 类型语言。



由上述分类可以看出，所有的正则文法都是上下文无关的，所有上下文无关的文法都是上下文有关的，所有上下文有关的文法都是无约束的。对上述四类文法下面分别举例说明。



例 1 : 无约束文法

$$G = (V_N, V_T, P, S)$$

$$V_N = \{A, B, S\}$$

$$V_T = \{a, b, c\}$$



$P:$ ① $S \rightarrow aAbc$

② $Ab \rightarrow bA$

③ $Ac \rightarrow Bbcc$

④ $bB \rightarrow Bb$

⑤ $aB \rightarrow aaA$

⑥ $aB \rightarrow a$

$$\begin{aligned}
S &\xrightarrow{\textcircled{1}} aAbc \xrightarrow{\textcircled{2}} abAc \xrightarrow{\textcircled{3}} abBbcc \\
&\xrightarrow{\textcircled{4}} aBbbcc \xrightarrow{\textcircled{5}} aaAbbcc \xrightarrow{\textcircled{2}} aabAbcc \\
&\xrightarrow{\textcircled{2}} aabbAcc \xrightarrow{\textcircled{3}} a^2b^2Bbcc \xrightarrow{\textcircled{4}} a^2b Bbbccc \\
&\xrightarrow{\textcircled{4}} a^2Bbbbccc \xrightarrow{\textcircled{6}} a^2b^3c^3
\end{aligned}$$

所以有 $L(G) = (a^n b^{n+1} c^{n+1} \mid_{n \geq 0})$

例 2 : 上下文有关的文法

$$G = (V_N, V_T, P, S)$$

$$V_N = \{A, B, S\}$$

$$V_T = \{a, b, c\}$$



- P:
- ① $S \rightarrow abc$
 - ② $S \rightarrow aAbc$
 - ③ $Ab \rightarrow bA$
 - ④ $Ac \rightarrow Bbcc$
 - ⑤ $bB \rightarrow Bb$
 - ⑥ $aB \rightarrow aaA$
 - ⑦ $aB \rightarrow aa$

$$\begin{aligned}
S &\xrightarrow{2} aAbc \xrightarrow{3} abAc \xrightarrow{4} abBbcc \\
&\xrightarrow{5} aBbbcc \xrightarrow{7} aabbcc \longrightarrow a^2b^2c^2
\end{aligned}$$

所以 $L(G) = (a^n b^n c^n \mid_{n \geq 1})$

例 3 : 上下文无关的文法

$$G = (V_N, V_T, P, S)$$

$$V_N = \{S\}$$

$$V_T = \{a, b, c\}$$



$$P: \begin{array}{l} \textcircled{1} \quad S \rightarrow ab \\ \textcircled{2} \quad S \rightarrow aSb \end{array}$$

$$\begin{array}{l} S \xrightarrow{\textcircled{2}} aSb \xrightarrow{\textcircled{2}} aaSbb \xrightarrow{\textcircled{1}} \\ aaabbb \longrightarrow a^3b^3 \end{array}$$

所以
$$L(G) = (a^n b^n \mid_{n \geq 1})$$

例 4 : 正则文法

$$G = (V_N, V_T, P, S)$$

$$V_N = \{S\}$$

$$V_T = \{a, b\}$$



P: ① $S \rightarrow a$

② $S \rightarrow b$

③ $S \rightarrow aS$

④ $S \rightarrow bS$

⑤ $A \rightarrow aA$

⑥ $A \rightarrow a$



$$S \xrightarrow{\textcircled{3}} aS \xrightarrow{\textcircled{3}} aaS \xrightarrow{\textcircled{3}} aaaS \xrightarrow{\textcircled{1}} a^4$$

$$S \xrightarrow{\textcircled{4}} bS \xrightarrow{\textcircled{4}} bbS \xrightarrow{\textcircled{2}} b^3$$

$$S \xrightarrow{\textcircled{3}} aS \xrightarrow{\textcircled{4}} abS \longrightarrow \cdots \longrightarrow a^n b^m$$

所以 $L(G) = (a^n b^n \mid_{m,n \geq 0})$

➤ 位置算子的运用

前述的字符串是一维结构，而图像是二维结构，因此，在用字符串来描绘图像时就需要建立一种相应的方法，把二维的位置关系缩减为一维形式。



串文法在图像描绘中大多数是从物体中抽取的联接线段为基础的。这种方法如图8—20所示。这是用有特定方向和长度的线段把结果编码。

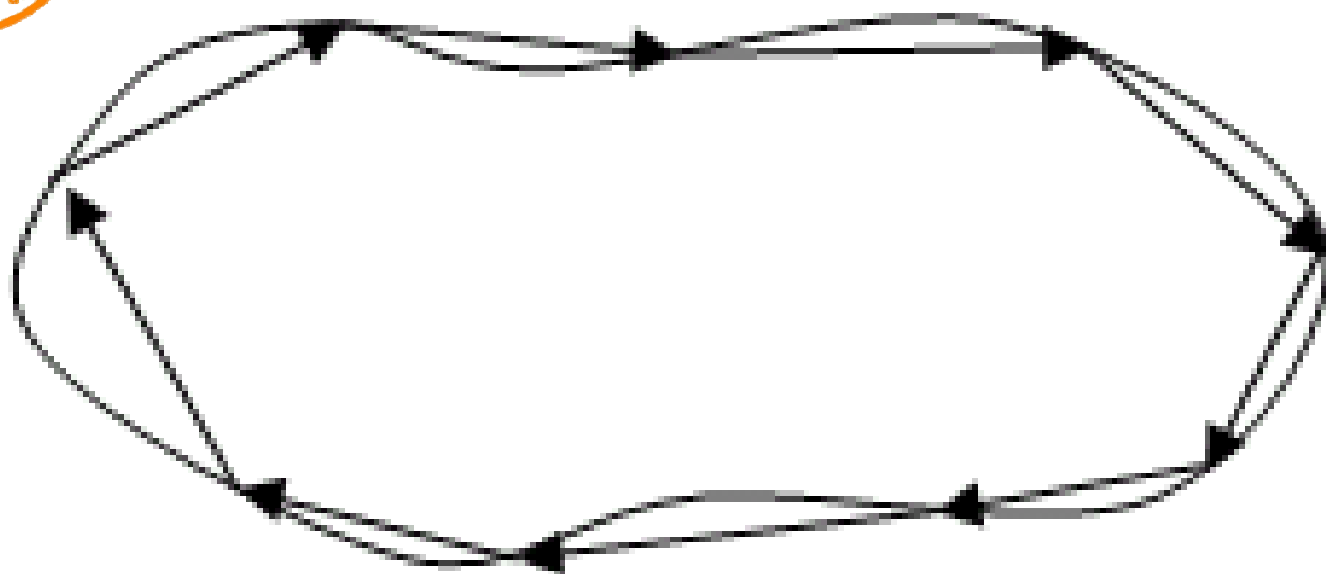
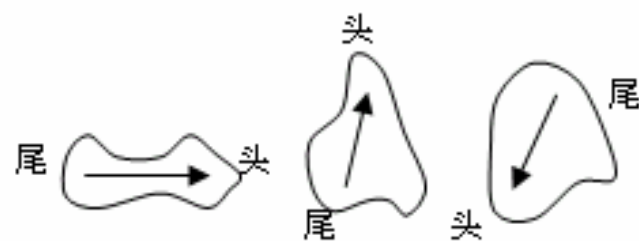


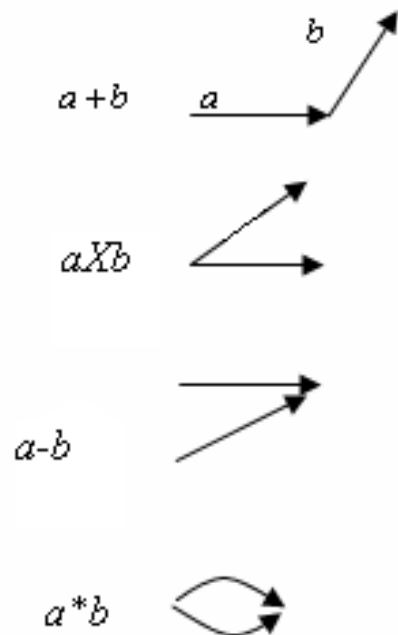
图8—20 用有方向的线段描绘区域边界



另外一种方法如图8—21所示。利用有向线段并用已定义的一些运算来描绘图像的某些部分。图中（a）是用有向线段来表示某些区域，图（b）是定义的某些运算。下面用一个具体例子来说明这些概念。



(a)



(b)

图8—21 另一种描绘方法

例 1 : 图片描绘语言

$$G = (V_N, V_T, P, S)$$

$$V_N = \{S, A_1, A_2, A_3, A_4, A_5\}$$



$$V_T = \{a \nearrow, b \searrow, c \rightarrow, d \downarrow\}$$

P: ① $S \rightarrow d + A_1$

② $A_1 \rightarrow c + A_2$

③ $A_2 \rightarrow d * A_3$

④ $A_3 \rightarrow a + A_4$

⑤ $A_4 \rightarrow b * A_5$

⑥ $A_5 \rightarrow c$



这里 $(\sim d)$ 表示与基本单元 d 方向相反的像元，而 a, b, c, d 的方向如 V_T 中所定义的那样。应用产生式产生一个像元 d ，其后，跟随一个尚未定义的变量 A_1 。



但是， A_1 分量所表示的结构的尾在这点上将和 d 的头相连， 这是由规定的算法 “+” 所决定的。变量 A_1 又可分解为 $C+A_2$ ， 当然 A_2 尚未定义。



同样 A_2 又可分解为 $\sim d * A_3$ 。应用前三条产生式得到的结果如图8—22中(a), (b), (c)所示。算子 “ $*$ ” 定义为尾到尾、头到头的连接方式。使用全部产生式所得到的最后结果示于图8—22 (f) 中。



这种文法只能产生一个结构。如果在产生式的规则中引入递归规则（变量有代替自己的能力），则这种PDL文法所产生的结构可扩展到各种结果。

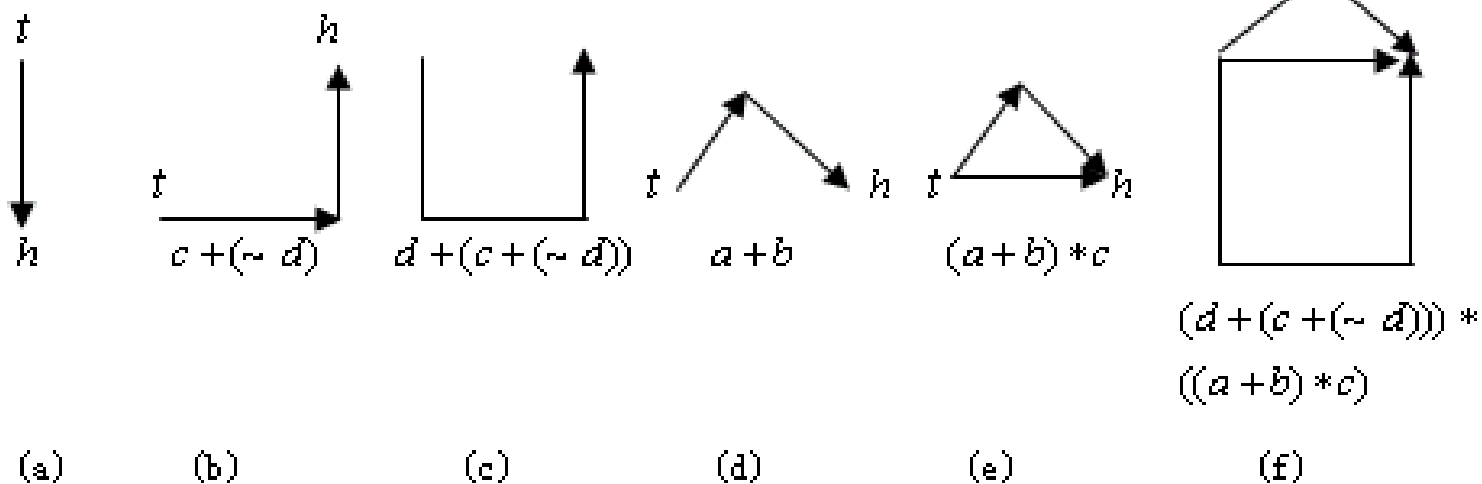


图8—22 PDL结构组成步骤

例 2：利用例 1 的条件，定义下列产生式规则

$$\textcircled{1} \quad S \rightarrow d + A_1$$

$$\textcircled{2} \quad A_1 \rightarrow c + A_1$$

$$\textcircled{3} \quad A_1 \rightarrow \sim d * A_2$$

$$\textcircled{4} \quad A_2 \rightarrow a + A_2$$

$$\textcircled{5} \quad A_2 \rightarrow b * A_2$$

$$\textcircled{6} \quad A_2 \rightarrow c$$

如果顺次应用这些产生式就会产生图8—22中(f)的结果。

例 3：染色体特征的描绘

描绘染色体的文法所用的基本像元如图8—23(a)所示。这些基本像元是对染色体沿边界顺时针方向循迹而检测出来的。典型的半中期和末期染色体的形状如图8—23(b)所示，描绘文法如下：



$$G = (V_N, V_T, P, S)$$

$$V_T = \{a, b, c, d, e\}$$

$$V_N = \{S, T, A, B, C, D, E, F\}$$



P: (1) $S \rightarrow C \cdot C$

(2) $T \rightarrow A \cdot C$

(3) $C \rightarrow B \cdot C$

(4) $C \rightarrow C \cdot B$

(5) $C \rightarrow F \cdot D$

(6) $C \rightarrow E \cdot F$

(7) $E \rightarrow F \cdot c$

(8) $D \rightarrow c \cdot F$

(9) $A \rightarrow b \cdot A$



$$(10) \quad A \rightarrow A \cdot b$$

$$(11) \quad A \rightarrow e$$

$$(12) \quad B \rightarrow b \cdot B$$

$$(13) \quad B \rightarrow B \cdot b$$

$$(14) \quad B \rightarrow b$$

$$(15) \quad B \rightarrow a$$

$$(16) \quad F \rightarrow b \cdot F$$

$$(17) \quad F \rightarrow F \cdot b$$

$$(18) \quad F \rightarrow a$$



其中算子 “ \cdot ” 用来描绘在按顺时针方向
追迹边界时在产生式中各项的可连接性。在
这个文法中， S 和 T 均为起始符， S 可产生相
应于半中期染色体的结构， T 可以产生相应
于末期染色体的结构。

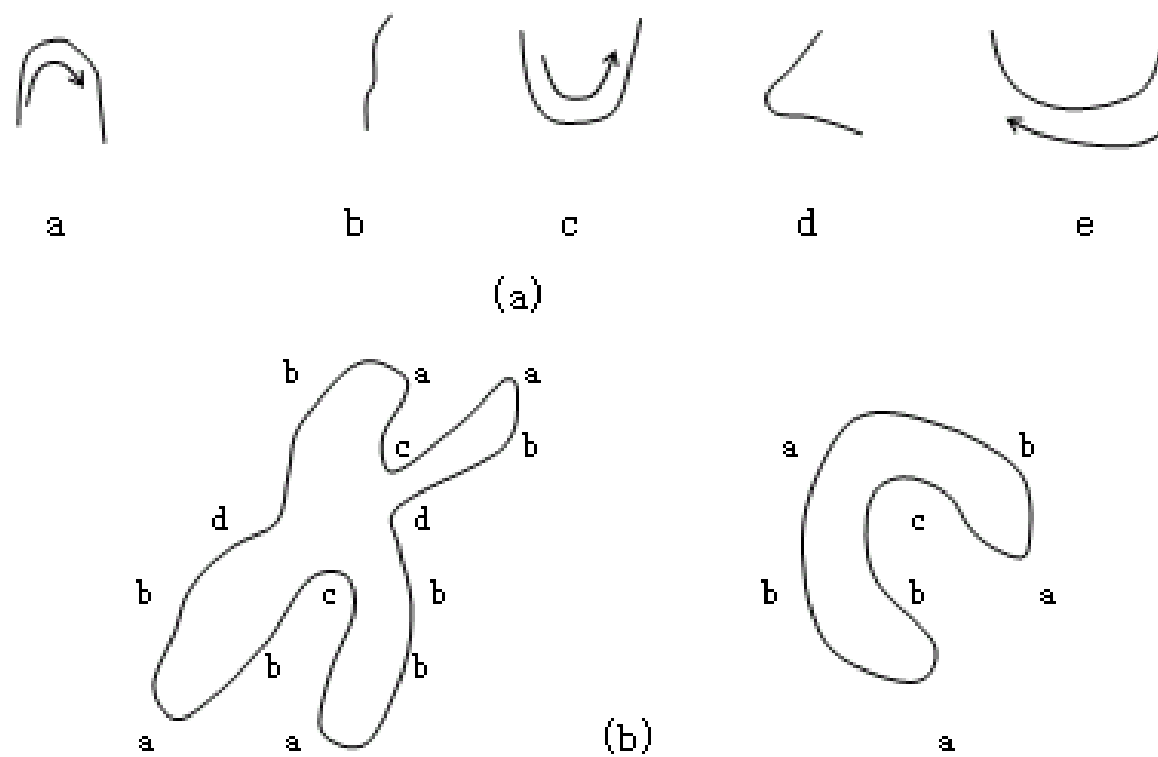


图 8—23 用边界轨迹描述染色体

➤ 高维文法

串文法适用于那些图像元素的连接可以用从头到尾或用其他连续形式的图像元素的描绘。在这里我们考虑一种更普遍的文法描绘途径，它有能力描绘更高级的图像元素。

(1) 树文法

高维文法之一是所谓树文法。树文法中所定义的树是一个或一个以上的节点的集合。其中有一个唯一的指定的节点为根；剩下的节点划分为 m 个不相交的集合，这些集合为 $T_1, T_2, T_3, \dots, T_m$ ，把 T_m 叫做 T 的子树；



树尖是树的根干部节点的集合，取从左到右的次序。图8—24是一树图，其中 s 是根(root)， x, y 为树尖。

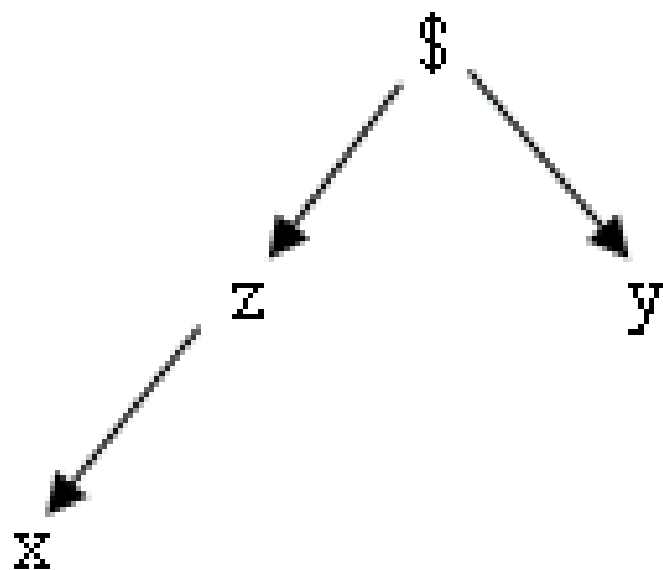


图8—24 树图

一般来讲，在树图中有两类重要信息。

一是关于节点的信息；

（节点是用一组字描述并存储）

另一个便是节点与其邻点的有关信息。

（节点与邻点的有关信息是以对其邻点的指示符的集合的形式存储。）



第一类信息用于识别模式的像元，

第二类信息定义像元和其他子结构间的物理关系。



例如，把图8—25所示的关系用图(b)所示的树来表示。图8—25 (b) 中，\$表示根，在\$中包含着*a*和*c*两部分，因此，从根放射出两个分支。第二级，*a*中包含*b*，*c*中包含*d*和*e*，在*e*中又包含*f*。这样就构成了一个树图，其中*b*，*d*，*f*是树的叶子。

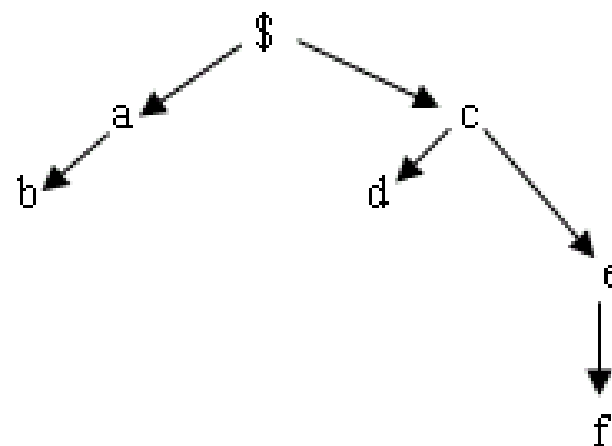
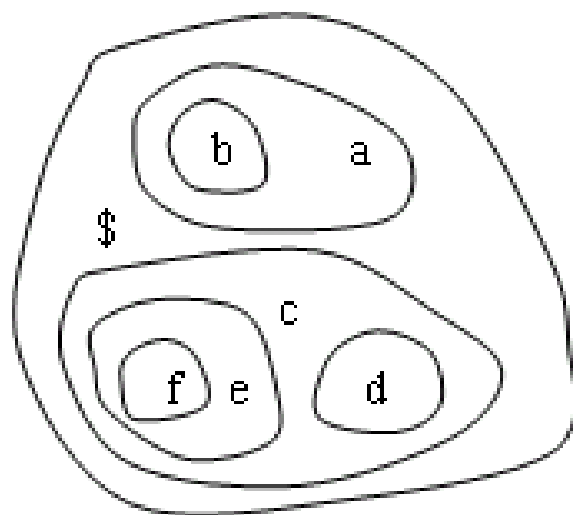


图8—25 用树来表示简单的组合区域

树文法为五元式，即

$$G = (V_N, V_T, P, r, S)$$

其中： V_N 为非终端符； V_T 为终端符； S 为起始符； P 为产生式集合；

r 为秩函数，它指出一个节点直接下降的数目。

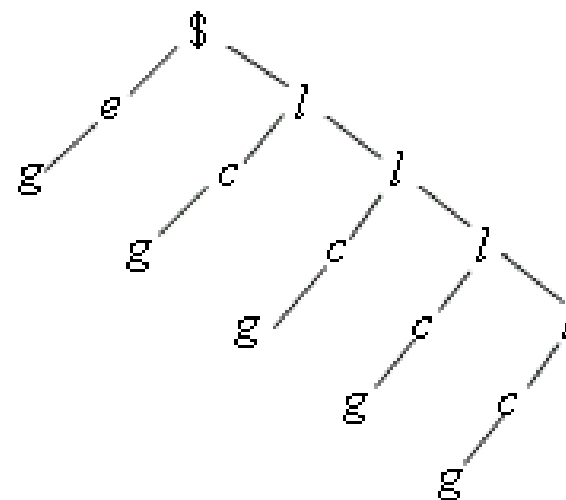
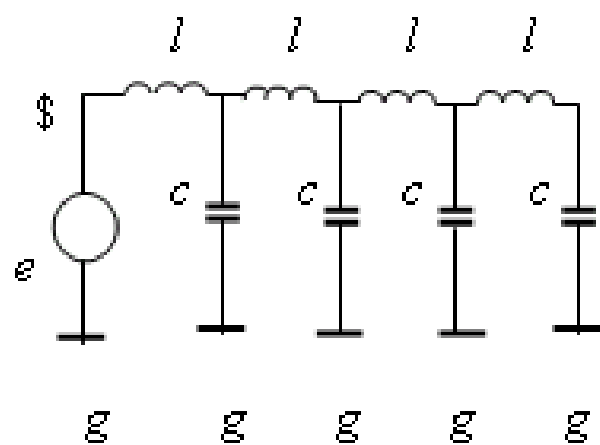


图 8—26 图形的树表示



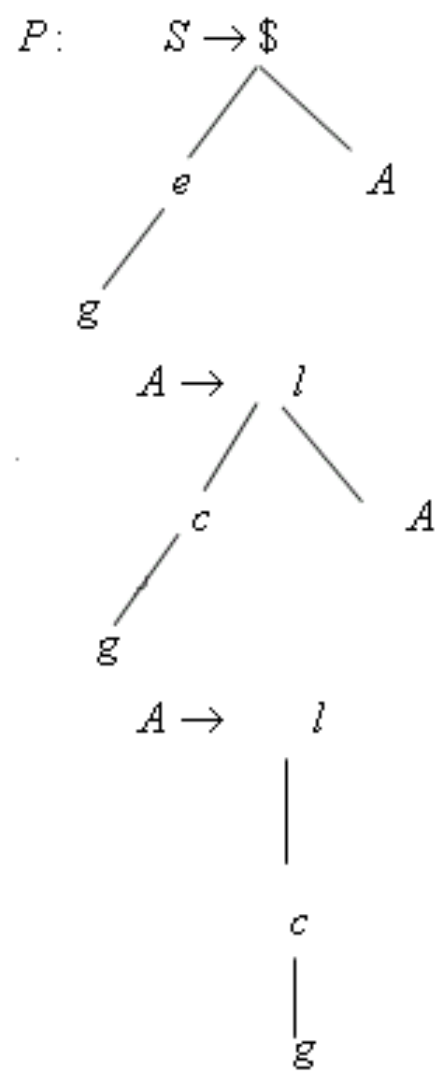
例：利用树文法把图8—26(a)的电路结构表示成树的形式。这个树图是把 L - C 网络的最左边的节点定义为根。其树文法如下（电路中的 L 用 l ， C 用 c ）：



$$G = (V_N, V_T, P, r, S)$$

$$V_N = \{S, A\}$$

$$V_T = \{e, g, l, c, \$\}$$



在这种情况下，秩函数

$$r(e) = 1$$

$$r(g) = 0$$

$$r(l) = \{2, 1\}$$

$$r(c) = 1$$

$$r(\$) = \{2, 1\}$$

利用三条产生式并利用 A 上定义的递归性就能产生无限数量的结构。



8.2.1 区域描绘

8.2.2 关系描绘

8.2.3 相似性描绘

8.2.4 霍夫变换



图像描绘的另外一种途径可借助于与已知描绘子的相似程度来进行，这种方法可以在任何复杂的程度上建立相应的相似性测度。它可以比较两个简单的像素，也可以比较两个或两个以上的景物。



1、距离测度

前面研究过的某些方法可以用来做为两幅图像区域之间进行比较的准则。例如，以矩做为描绘子，假如两个区域的矩分别为 X_1 和 X_2 。把它们写成向量式如下：



$$X_1 = \{x_1, x_2, x_3, \cdots \cdots x_n\}$$

$$X_2 = \{x'_1, x'_2, x'_3 \cdots \cdots x'_n\}$$

(8—70)



此时, X_1 和 X_2 之间的距离可定义如下:

$$\begin{aligned} D(X_1, X_2) &= |X_1 - X_2| \\ &= \sqrt{(X_1 - X_2)'(X_1 - X_2)} \end{aligned}$$

(8—71)



采用距离这一测度可以测量两个描绘子之间的相似性。如果已知描绘子用 $X_1, X_2, X_3, \dots, X_L$ 表示，未知描绘子用 X 表示，可以计算 X 与已知描绘子的距离 $D(X, X_i)$ ，如果

$$D(X, X_i) < D(X, X_j) \quad (8-72)$$



就可以判定 X 更接近第 i 个描绘子。式中

$j=1,2,3,\dots,L$ ，并且。这个方法原则上可

用于各种描绘子，只要它们能够用一矢量来

表示就可以。

2、相关性

当给定一幅大小为 $M \times N$ 的数字图像 $f(x,y)$,
要确定它是否包含一个区域, 该区域与某个大
小为 $J \times K$ 中的某个区域 $w(x,y)$ 相类似, 其
中 $J < M$, $K < N$ 。



解决这样问题常用的方法之一是求 $f(x,y)$ 和 $w(x,y)$ 之间的相关性。两个函数之间的相关的定义由下式表示：

$$R(m,n) = \sum_x \sum_y f(x,y)w(x+m,y+n)$$



为计算方便有的文献写成卷积形式

$$R(m, n) = \sum_x \sum_y f(x, y) w(x - m, y - n) \quad (8-73)$$

其中

$$m = 0, 1, 2, 3, \dots, M - 1$$

$$n = 1, 2, 3 \dots N - 1$$

具体检测步骤如下：

对于 $f(x,y)$ 中的任意值 $w(x,y)$ 用式(8—73)可求得一个 R 值，在 m,n 变化时， $w(x,y)$ 沿着图像移动，这时可得到 $R(m,n)$ 。求出 $R(m,n)$ 的最大值就说明 $f(x,y)$ 和 $w(x,y)$ 在此处最相似。



但是在 m, n 接近边缘时，其精度较差。这个误差量正比于 $w(x, y)$ 的大小。上述步骤可由图8—30加以形象地说明。

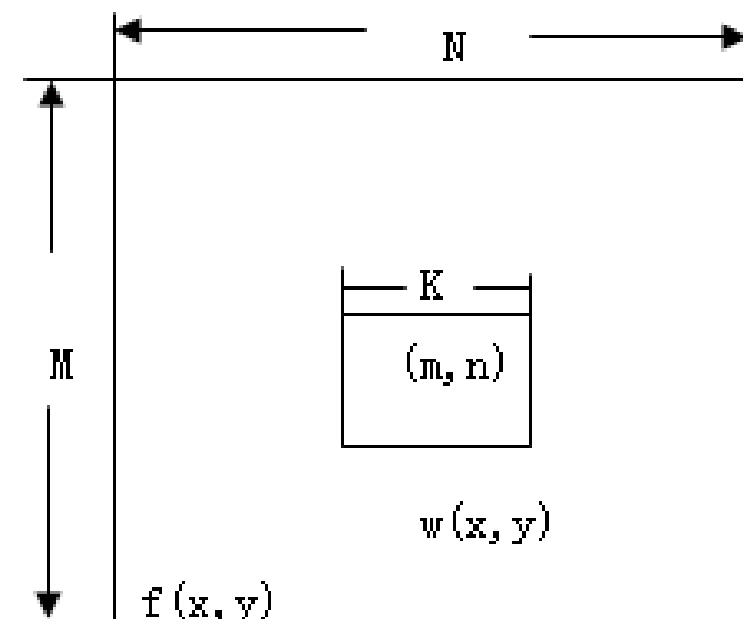


图 8—30 在给定点 (m, n) 上求 $f(x, y)$ 和 $w(x, y)$ 的相关性步骤



这里提到的相关检测法与前述的样板匹配法颇为相似。在这个意义下，样板就是 $w(x,y)$ 。相关检测法与样板匹配法的主要区别是 $w(x,y)$ 一般是一幅子图像。

适合于图像特性的更复杂的相关定义

可由下式表示

$$r(m, n) = \frac{\sum_x \sum_y f(x, y) w(x + m, y + n)}{\left[\sum_x \sum_y f^2(x, y) \right]^{\frac{1}{2}} \left[\sum_x \sum_y w^2(x + m, y + n) \right]^{\frac{1}{2}}}$$

(8—74)

或采用如下形式：

$$r(m, n) = \frac{\sum_x \sum_y f(x, y) w(x - m, y - n)}{\left[\sum_x \sum_y f^2(x, y) \right]^{\frac{1}{2}} \left[\sum_x \sum_y w^2(x - m, y - n) \right]^{\frac{1}{2}}}$$



式中引入了归一化因子。归一化因子的计算是在 $w(x,y)$ 被划定的整个面积上进行的，因此它是作为位移函数而变化的。

相关性的计算可通过FFT算法在频域进行，这样比直接在空域计算更有效。

3、结构相似性

一般来讲，结构相似性的描绘比起距离测度与相关性更难于公式化，因此，应用起来也就有更高的难度。可以用作相似测度的典型的结构描绘子是线段的长度、线段之间的角度、亮度特性、区域的面积、在一幅图像中一个区域相对于另外一个区域的位置等等。



例如，对一幅图像可用出现于图像中的物体以及这些物体间的关系来描述图像，用于描述的一部分性质可能是下列的一种或几种。

(1) 亮度：黑、灰、白、亮、暗、均匀、有阴影
等等；

(2) 颜色：红、橙、黄、绿、青等等；

(3) 结构：平滑、粒状、斑驳的、有条纹的等
等；

(4) 大小：长度、面积、体积、高度、宽度、深
度、大小、高、矮、宽、窄等等；

(5) 取向：水平、垂直、倾斜；

(6) 形状：实心的、中空的、密集的、参差不齐的、伸长的等等；

上述的几种只反映了一个侧面，但是，就是这些，如果要从一幅图像中将它们抽取出来也是相当难办的。



基于结构分量之间关系的相似性测度，可以用某些文法将其公式化。假定有两类物体，可以分别用两种文法 G_1 和 G_2 来产生它们。



给定一个特定的物体，如果它能用 G_1 来产生而不能用 G_2 来产生，那么就可以说这个物体更接近第一类。



如果用两种文法都能产生，就不能分辨给定的物体了。所以，这种方法可以把文法近似于给定结构的紧密程度作为相似性的测度。



8.2.1 区域描绘

8.2.2 关系描绘

8.2.3 相似性描绘

8.2.4 霍夫变换



霍夫变换是一种线描述方法。它可以将笛卡尔坐标空间的线变换为极坐标空间中的点。图8—31是 x, y 坐标系中的一条直线。如果用 ρ 代表直线距原点的法线距离， θ 为该法线与 x 轴的夹角，则可用如下参数方程来表示该直线。



这一直线的霍夫变换:

$$\rho = x \cos \theta + y \sin \theta \quad (8-75)$$

在极坐标域中便是如图所示的一个点。

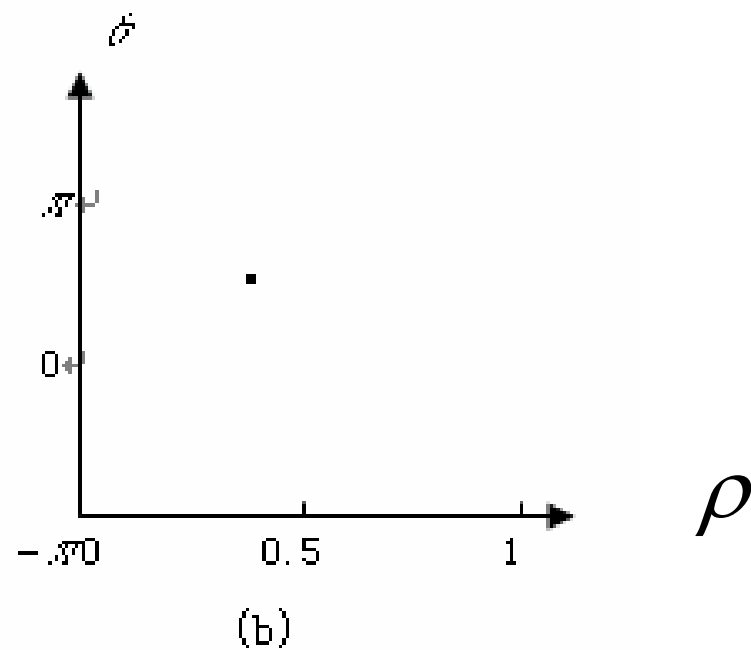
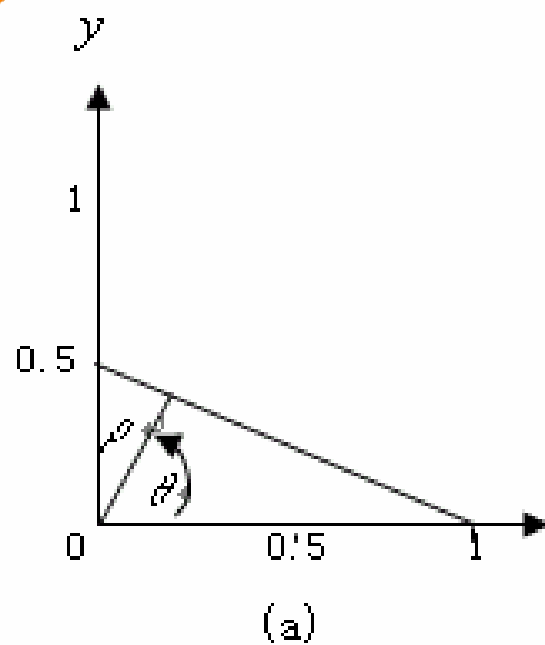
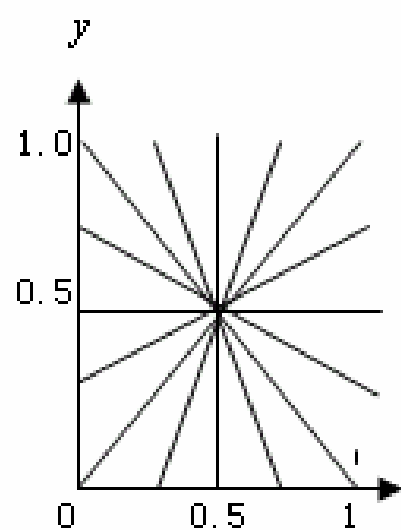


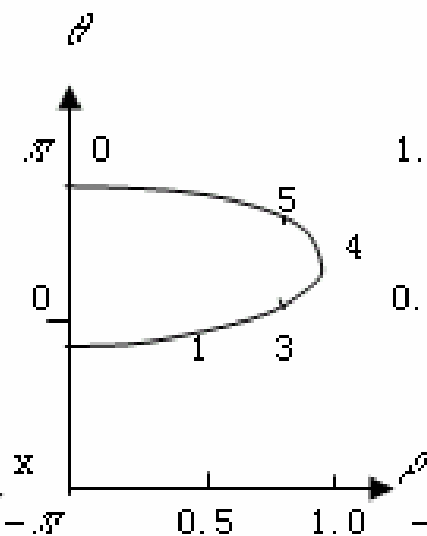
图 8—31 Hough 变换的原理



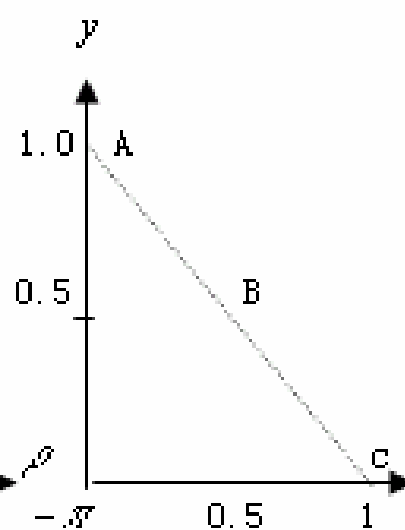
由图8—31(b)所示的一个点。由图8—31(c)、(d)、
(e)、(f)所示，在 (x,y) 坐标系中通过公共点的一簇
直线，映射到 (ρ,θ) 坐标系中便是一个点集。



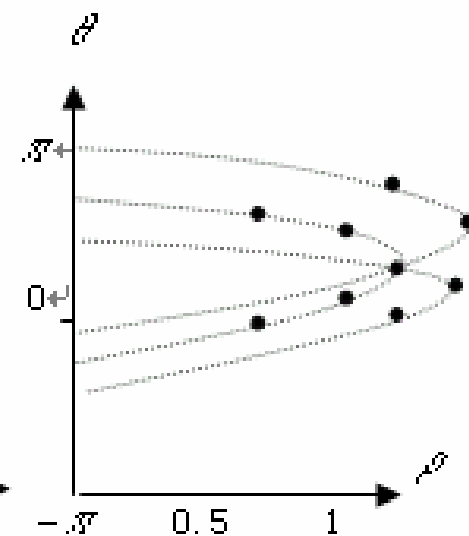
(c)



(d)



(e)



(f)

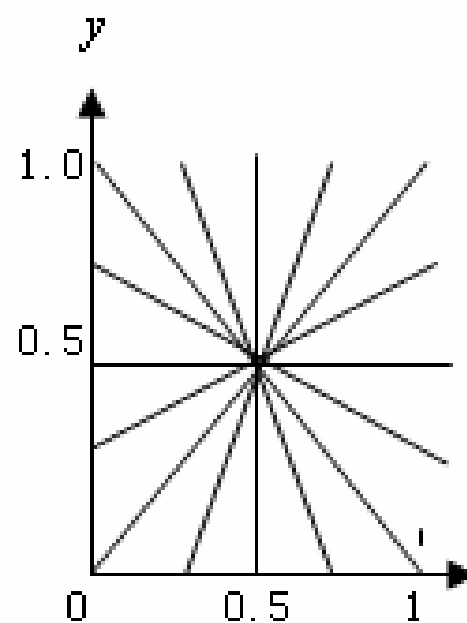


在 (x,y) 坐标系中共线的点映射到 (ρ,θ) 坐标系便成为共点的一簇曲线。由此可见，霍夫变换使不同坐标系中的线和点建立了一种对应关系。

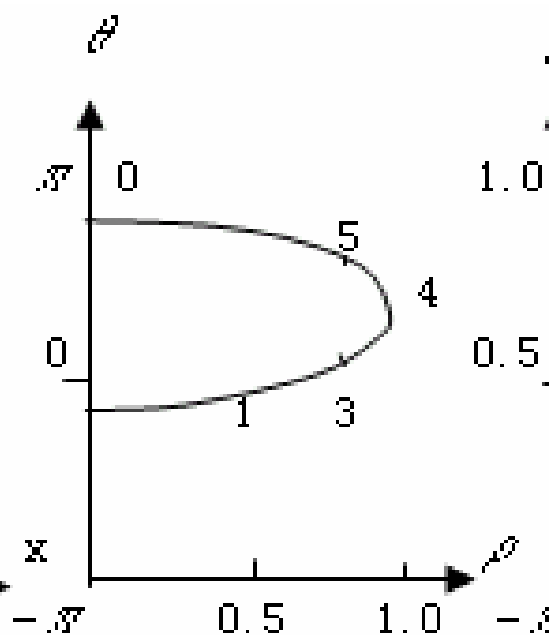


综上所述，可总结霍夫变换的几点性质如下：

(1) (x,y) 域中通过公共点的一簇直线，对应于变换域 (ρ,θ) 中的一条正弦曲线。

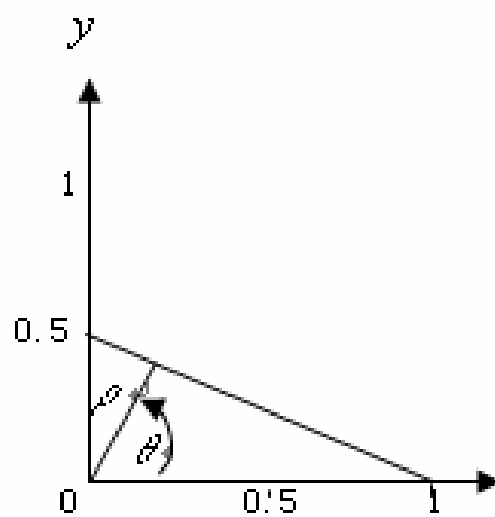


(c)

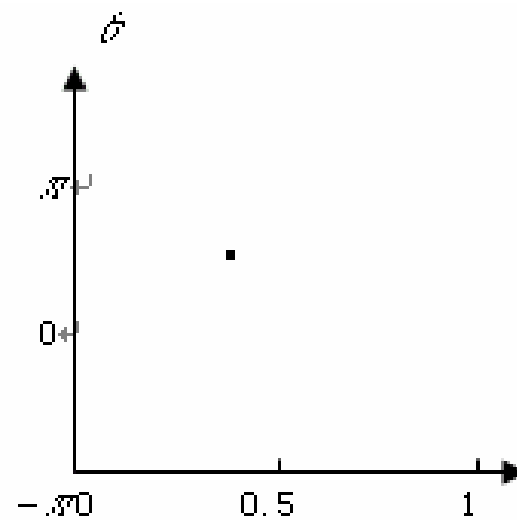


(d)

(2) 变换域中的一点对应于 (x,y) 域中的一条直线。



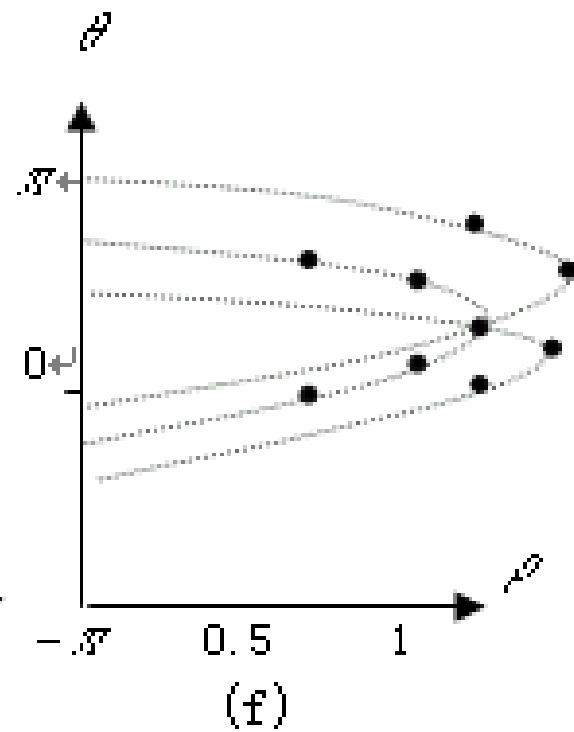
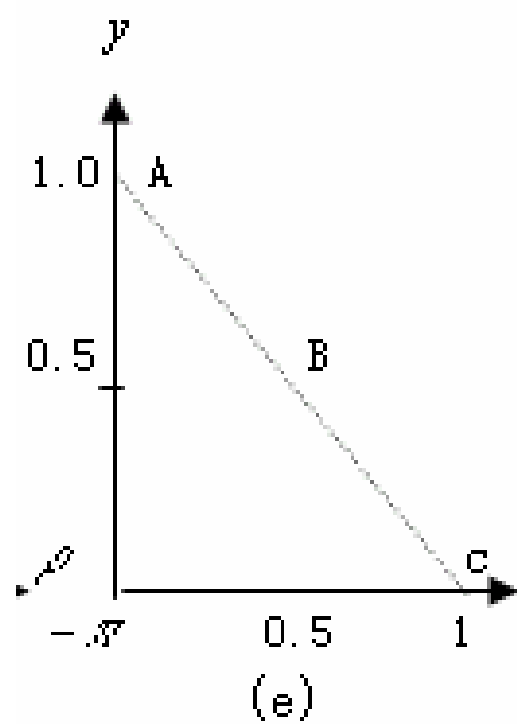
(a)



(b)



(3) (x, y) 域中一条直线上的 n 个点对应于变换域中经过一个公共点的 n 条曲线。这条性质可证明如下：



证明:

设 (x,y) 平面中的 n 个点

$$(x_1, y_1), (x_2, y_2), (x_2, y_2) \cdots (x_n, y_n)$$

共一条直线, 则有:

$$y_i = ax_i + b \quad i = 1, 2, 3, \cdots, n$$

由*Hough*变换的定义可知，变换域的曲线为：

$$\rho_i = x_i \cos\theta_i + y_i \sin\theta_i$$

将 $y_i = ax_i + b$ 代入上式，有：

$$\begin{aligned}
\rho_i &= x_i \cos \theta_i + y_i \sin \theta_i \\
&= x_i \cos \theta_i + (ax_i + b) \sin \theta_i \\
&= x_i (\cos \theta_i + a \sin \theta_i) + b \sin \theta_i
\end{aligned}$$

由此可知，无论 x_i 为何值，曲线都将通过

$$\cos \theta_i + a \sin \theta_i = 0 \quad \text{这一点。}$$

也就是

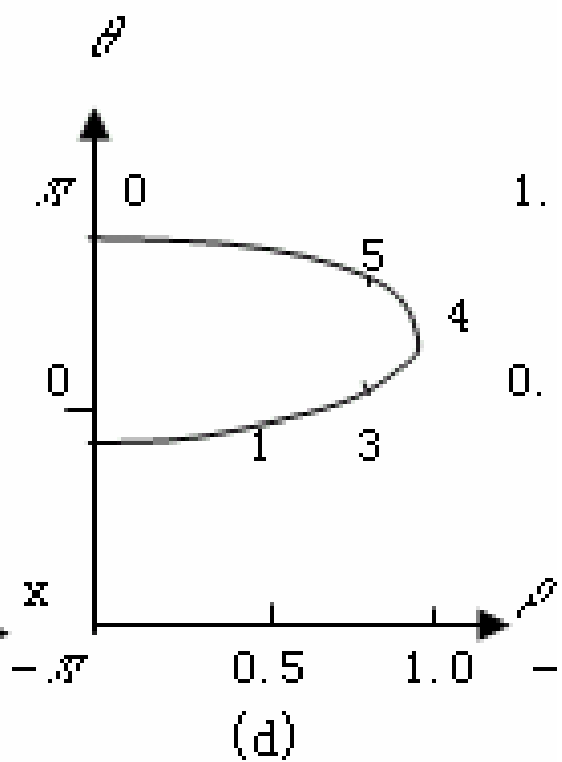
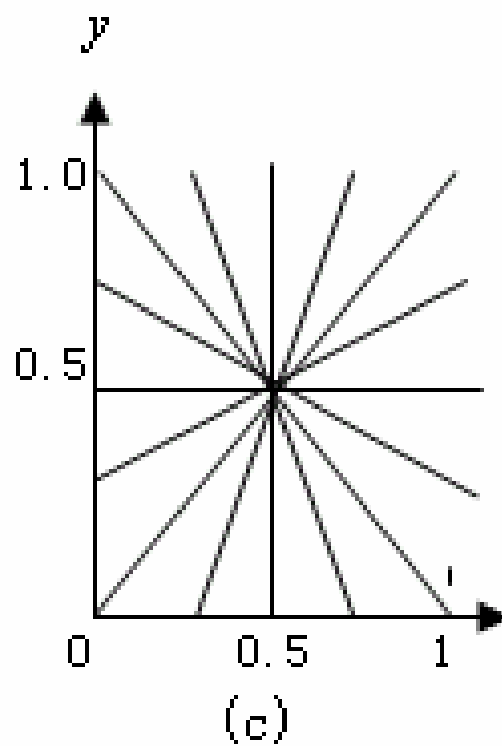
$$\left\{ \theta = -\arctg \frac{1}{a}, \rho = b \sin(-\arctg \frac{1}{a}) \right\} \quad \text{这一点。}$$



(4) 变换域中一条曲线上的 n 点对应于 (x,y)

域中过一公共点的 n 条直线。这条性质

可证明如下：



证明：

假设变换域中有 n 点

$$(\rho_1, \theta_1), (\rho_2, \theta_2), (\rho_3, \theta_3) \cdots \cdots (\rho_n, \theta_n)$$

在同一曲线上，则有：

$$\rho_i = a \cos \theta_i + b \sin \theta_i \qquad i = 1, 2, 3, \cdots n$$

对应于 (x,y) 域的直线可导出如下:

因为 $\rho_i = x_i \cos \theta_i + y_i \sin \theta_i$

所以 $y_i = -x_i \operatorname{ctg} \theta_i + \frac{1}{\sin \theta_i} \cdot \rho_i$

$$= -x_i \operatorname{ctg} \theta_i + \frac{1}{\sin \theta_i} (a \cos \theta_i + b \sin \theta_i)$$

$$= (-x_i + a) \operatorname{ctg} \theta_i + b$$



由此可见，不管 θ_i 为何值，直线都
经过 $\{x_i = a, y_i = b\}$ 这一点。

霍夫变换的应用可用如下方法实现：

在 (x, y) 域中的每一离散数据点变换为 (ρ, θ) 域中的曲线。将 θ 和 ρ 分成许多小段，每一个 θ 段和每一 ρ 小段构成一个小单元 $(\Delta\theta, \Delta\rho)$ 。对应于每一个小单元可设一累加器。



在 (x, y) 域中可能落在直线上的每一点对应变换域中的一条曲线

$$\rho = x_i \cos \theta + y_i \sin \theta$$



分别使 θ 等于 $0, \Delta\theta, 2\Delta\theta, 3\Delta\theta\ldots$, 便可求出相应的 ρ 值, 并分别计算落在各小单元中的次数, 待全部 (x, y) 域内数据点变换完后, 可对小单元进行检测.



这样，落入次数较多的单元，说明此点为较多曲线的公共点，而这些曲线对应的 (x, y) 平面上的点可以认为是共线的。



检测出 (x, y) 平面上 n 点后，将曲线交点坐标

(ρ_0, θ_0) 代入

$$\rho_0 = x \cos \theta_0 + y \sin \theta_0$$

便可得到逼近 n 点的直线方程。

在这种实现中，变换域小单元
($\Delta\rho$, $\Delta\theta$)的大小直接影响 (x, y) 域
中逼近直线的精度。霍夫变换的另外一个实用
弱点是未考虑点的相邻性，有时得到的最佳逼
近直线可能会由于邻近的点的影响而产生扭曲。



作为霍夫变换的推广，可看到如下一些结果。例如，有一曲线方程为：

$$Ax^2 + By^2 = C \quad (8-76)$$

显然，在椭圆上的每一点都满足式(8—76)。

在此式中 x, y 是变量， A, B, C 是系数。



如果把式(8—76)写成式(8—77)的形式，即：

$$x^2 A + y^2 B = C \quad (8—77)$$



这里，把 A, B, C 看成变量，把 x^2, y^2 看成系数，那么，在 (x, y) 域中的任何一点将对应于变换域中的一个曲面。 (x, y) 域中椭圆上的 n 点将对应于变换域中 n 个有共同交点的 n 个曲面。这一推广可用于圆的检测。