

数字图像处理 笔记 前八周

数字图像处理 笔记 前八周

[Week 2](#)

[Week 3](#)

[Week 4](#)

[Week 5](#)

[Week 6](#)

[Week 7](#)

[Week 8](#)

Week 2

- 如下：

色度图，马蹄形区域的边界和内部代表了所有可见光的颜色值，图中中央的一点W表示标准白光。

色系数的计算方法(归一化)：

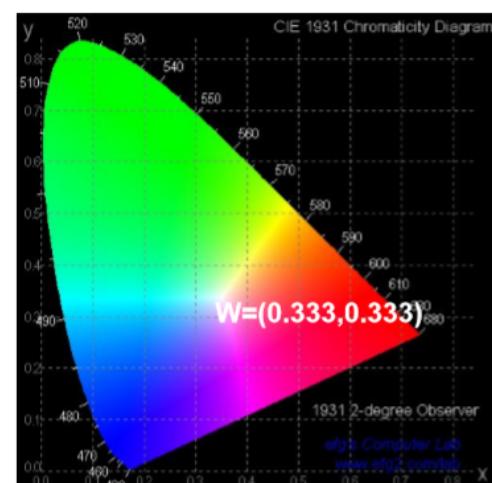
$$x = \frac{X}{X+Y+Z}$$

$$y = \frac{Y}{X+Y+Z}$$

$$z = \frac{Z}{X+Y+Z}$$

其中，色系数x, y分别为色度图上的横、纵坐标值。

Guojin Wang, Tsinghua University
1931 CIE (Commission Internationale de l'Eclairage)
国际照明协会



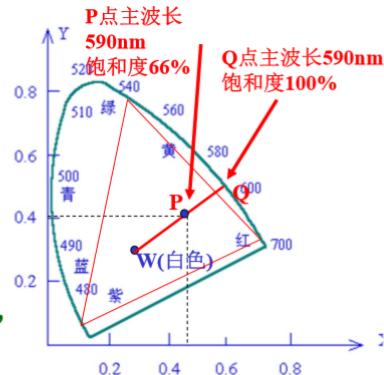
- 色调、亮度：

描述彩色特性的三种（基本）感知量：

彩色 { 亮度：与物体的反射率成正比
色度 { 色调：与光谱中光的波长相联系
 饱和度：与一定色调光的纯度有关
色调和饱和度合起来称为色度
彩色可用亮度和色度共同表示

在色度图中：

- (1) 每点都对应一种颜色
- (2) 边界上的点代表纯彩色，中心点W处纯度为零
- (3) 连接任两端点的直线上的各点表示将这两端点所代表的彩色相加可组成的一种新彩色
- (4) 过W点直线端点的两彩色为互补色
- (5) 三角形包含由三顶点可组成的所有



Guojin Wang, Tsinghua University, 第一章 引言

39

- 彩色光中加入白色光，由于白色光中包含各种色光的分量，故白色光越多，亮度越大。

颜色的基本特性

区分颜色常用3种基本特性量：
辉度，色调，饱和度



钢材颜色随辉度的变化

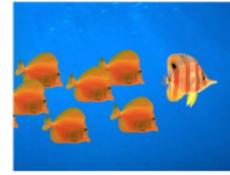
■ 辉度（亮度）

与光的强度直接相关。与物体的反射率成正比，无彩色时只有辉度量。

对彩色光而言，颜色中加入白色越多越明亮

40

■ **色调**: 表示光的颜色, 决定于混合光谱中主要光的波长, 由强度最大的彩色成分决定。



饱和度100%



饱和度50%

■ **饱和度**: 与表示彩色光的鲜艳度, 取决于彩色中的白色光含量。数值为介于0~100%的百分比。随着白光含量的增加, 饱和度逐渐减少。

2.2.1 面向视觉感知的彩色模型

HSI模型:

彩 { 色度 {
 H 表示色调 (hue)
 S 表示饱和度 (saturation)
 I 表示强度 (intensity, 对应成象亮度和图象灰度)

两个基本特点:

- I 分量与图象的彩色信息无关
- H 和 S 分量与人感受彩色的方式紧密相连 (合称色度)

- CMY: 减法运算 涂料的组合
- CMYK: K表示黑色 打印时专门有K, 不需要CMY三者加起来, 比较浪费。
- RGB: 加法运算 光的组合
- 注意亮度对应灰度图

2.2.1 面向视觉感知的彩色模型

HSI模型：

彩 { 色度 $\left\{ \begin{array}{l} H \text{ 表示色调 (hue)} \\ S \text{ 表示饱和度 (saturation)} \\ I \text{ 表示强度 (intensity, 对应成象亮度和图象灰度) } \end{array} \right.$

两个基本特点：

- I 分量与图象的彩色信息无关
- H 和 S 分量与人感受彩色的方式紧密相连（合称色度）

2.2.1 面向硬件设备的彩色模型

CMY模型：

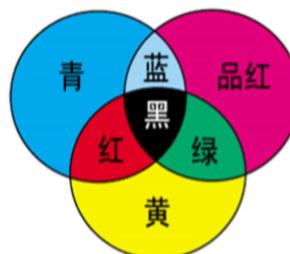
主要用于彩色打印，这三种补色可由从白色中分别减去三种基色而得到

从CMY到RGB的转换为

$$R = 1 - C \quad C = 1 - R$$

$$G = 1 - M \quad M = 1 - G$$

$$B = 1 - Y \quad Y = 1 - B$$



Guojin Wang, Tsinghua University, 第一章 引言

53

Week 3

§ 2.3 图像采样与量化

图像采样与量化的概念

从实际场景中获取的图像，需要在空间和灰度上离散化后，获得数字图像，才能由计算机进行处理

- > 空间坐标的离散化称为**取样**→数字化坐标值
- > 灰度的离散化则称为**量化**→数字化幅度值

取样密度由取样间隔决定，
量化值由灰度级确定

两者决定数字图像的质量

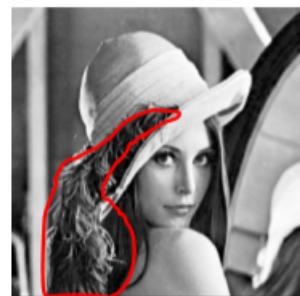
27

2. 非均匀取样和量化

即根据图像的特性进行自适应采样和量化的过程

- (1) 非均匀取样：在比较尖锐的灰度过渡区可采用较密集的采样，在较平滑的区域进行较稀的采样。

→缺点是需要确定区域的边缘

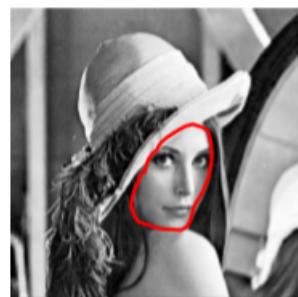


- (2) 非均匀量化：量化时非均匀地分配灰度级。

方法：

计算所有灰度值出现的频率；

若某个范围的灰度值出现频繁，量化灰度在此范围就较密，其他范围则较稀



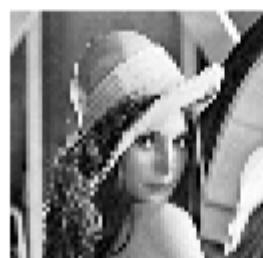
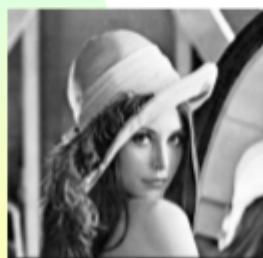
36

空间与灰度分辨率

1. 空间分辨率

取样数目是决定一幅图像空间分辨率的主要参数。

空间分辨率是指图像中可分辨的最小细节

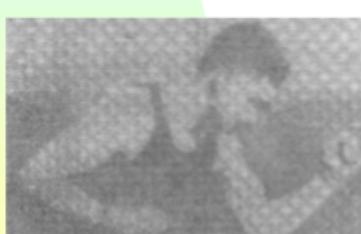


37

2. 灰度级分辨率

灰度级分辨率是指在灰度级别中可分辨的最小变化。黑白只有2个灰度级

出于硬件考虑，通常，灰度级数为2的整数次幂，**大多数场合取 8bit 表示灰度级，宽动态10、12bit**



1921年，电报打印机，5级灰度

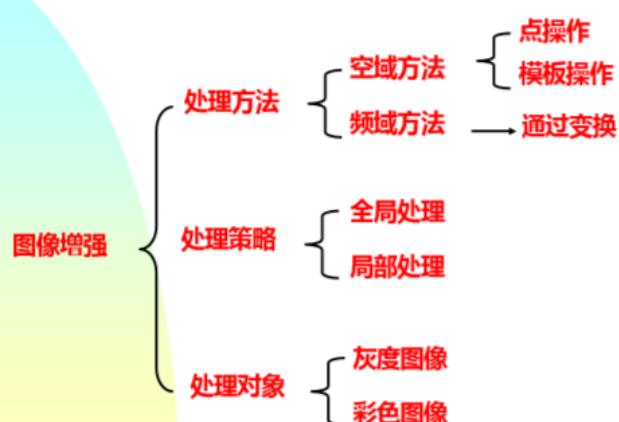


1929年，15级灰度

44

From Gonzalez' book

图像增强方法结构关系图

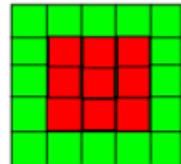
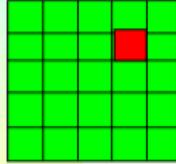


51

点操作

考察空域增强处理 $g(x, y) = EH[f(x, y)]$

EH定义在点(x, y)上 → 点操作
(模板操作的特例)



EH定义在(x, y)邻域上 → 模板操作

56

点操作与邻域无关，所以对于

$$g(x, y) = EH[f(x, y)]$$

$g(x, y)$ 的值只与在(x, y)处的 $f(x, y)$ 有关。

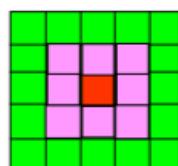
例如，在阈值分割中，
大于阈值的像素值被
扩大到某一个区间，
小于阈值的像素值被
缩小至某个区间



From Gonzalez' book 57

模板操作(masking)

通常，像素具有相关性。
邻域是包含该像素的一块矩
形区域，通常是 $N \times N$ 的正
方形区域。



$g(x, y)$ 的值，不仅取决于 $f(x, y)$ ，还与
邻域内的所有像素值有关。

$$g(x, y) = EH[f(x, y)]$$

58

3.2.1 图像间运算

- {将图像作为运算单元/对象，逐像素进行}

3.3.1.1 图像间算术运算

3.2.1.2 图像间和辑运算

Week 4

- 图像与灰度直方图之间是一个一一对应的关系。灰度直方图丧失了空间信息。

3.2.3 图像灰度映射

- {将 $f(x, y)$ 中的每个象素灰度按映射 E_H 操作，直接变换以得到 $g(x, y)$ }

$$g(x, y) = E_H[f(x, y)]$$

3.2.3.1 灰度映射原理

3.2.3.2 典型灰度映射

3.2.3.2 典型灰度映射

基本灰度变换，是简单的空域变换增强方法。在空间域变换增强中，基本灰度变换有以下几种方法：

- 图像求反、增强对比度、动态范围调整（压缩）、灰度切分、伽马校正



2. 增强对比度

1. 亮度适应和鉴别的概念

- 由于数字图像作为离散的亮点集显示，因此需要考虑人眼对不同亮度的鉴别能力

⇒人眼适应光强度级别范围很宽，从夜视阈值到强闪光

2. 为适应大范围的光强变化，人眼靠改变灵敏度，来调节不同范围内的鉴别能力

→即亮度适应

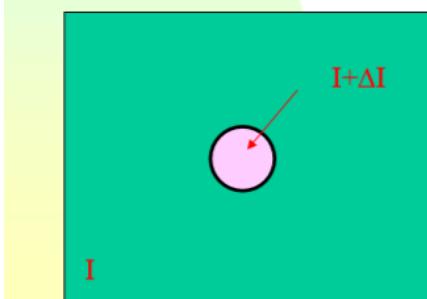
实验指出，主观亮度（即由人的视觉系统感觉到的亮度），是进入眼睛的、光强度的对数函数。

3. 在任何一个适应级，除了亮度适应之外，人眼对光强的鉴别能力也十分重要

■ 人眼鉴别能力实验：

不同背景光强度下的亮度鉴别能力

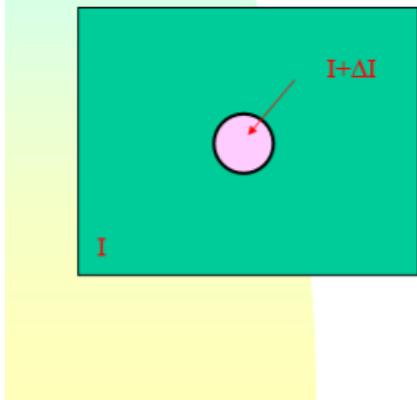
背景强度 I , 变化 ΔI



◆如果 ΔI 不够亮，实验者说“**No**”，表示没有感知圆内区域上的变化

◆当 ΔI 足够强，感知到亮度变化时，说“**yes**”

背景强度 I , 变化 ΔI

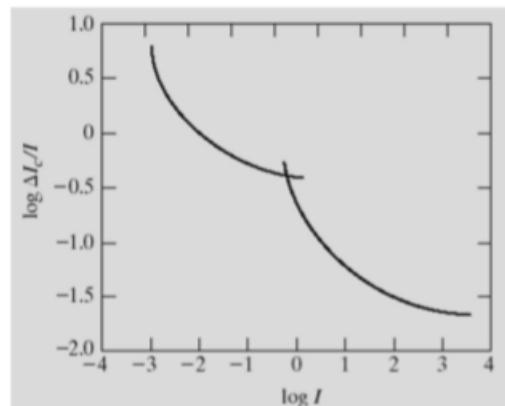


ΔI_c 为对给定背景亮度有 50% 的机会回答 yes 的照度 增量，则量 $\Delta I_c / I$ 称为 韦伯比

18

- ◆ 小 $\Delta I_c / I$ 表示可以区分 强度小的变化，代表较好的 亮度区分
- ◆ 反之，大 $\Delta I_c / I$ 则代表 亮度区分能力差

作为强度函数的典型
韦伯比 $\Delta I_c / I$ 图



From Gonzalez' book

实验表明，当背景亮度低时，亮度鉴别能力低

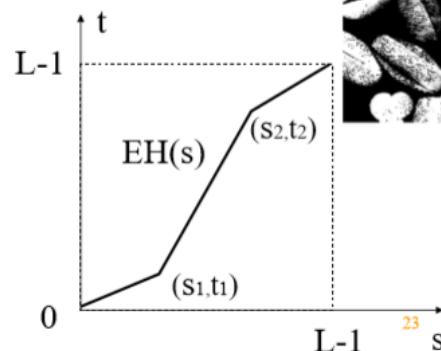
2. 增强对比度

(2) 算法(对比度拉伸, 属分段线性变换)

增强原图像各部分的反差，实际中增加 某2个灰度值区间的动态范围，增强对比度。

对于图示的增强 对比度变换曲线， 不 同区间的动态范围发 生了变化

0→s₁ 减少
s₁→s₂ 增加
s₂→L-1 减少



3. 动态范围调整（压缩）

(2) 动态范围调整算法

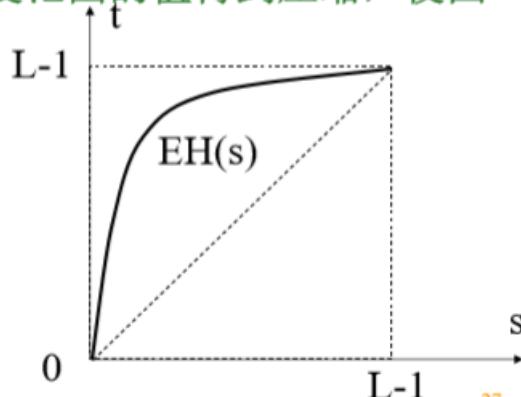


- 将灰度变化的动态范围进行调整→使低灰度范围的值得到扩展，高灰度范围的值得到压缩，使图像分布均匀。

常用的一种方法是对数变换

$$t = c \cdot \log(1+s)$$

其中 c 是比例常数

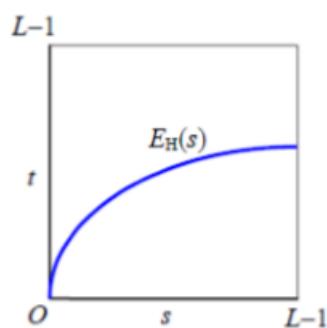


作用是将 $[0, >>L-1]$ 的动态范围调整到 $[0, L-1]$ 27

3. 动态范围调整（压缩）

(3) 动态范围调整实例

-



4. 灰度切分

(1) 灰度切分概念：突出某范围的灰度，使得代表该部分的物体明显显示，如裂纹的增强显示



40

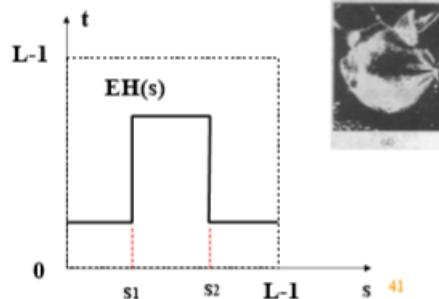
4. 灰度切分

(2) 灰度切分算法：

灰度切分法与增强对比度法近似，目的是将某个灰度值范围突出

如图所示：

将 $s_1 \rightarrow s_2$ 区间的灰度级突出，而将其他区间的灰度值变换为一个低灰度值。



41

5. 伽马变换

$t = Cs^\gamma$ 幂律（伽马）变换 c 和 γ 是正常数

$\gamma < 1$ 提高灰度级，在正比函数上方，使图像变亮
 $\gamma > 1$ 降低灰度级，在正比函数下方，使图像变暗

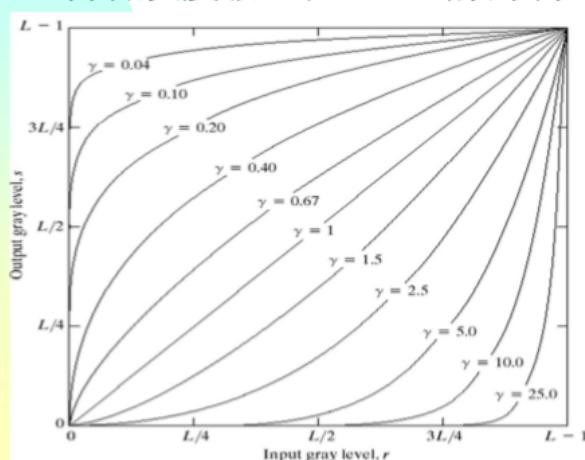


FIGURE 3.6 Plots of the equation $s = cr^\gamma$ for various values of γ ($c = 1$ in all cases).

42

3.2.3 直方图变换

{以概率论为基础，通过改变图像的直方图来改变图像中像素的灰度，以实现图像增强。也常称直方图修正}

3.2.3.1 直方图的概念

3.2.3.2 直方图均衡化

3.2.3.3 直方图规定化

45

3.2.3.1 直方图的概念

表示图像中具有每种灰度级的象素的个数，反映图像中每种灰度出现的频率。

图像的灰度统计直方图，是一个一维的离散函数，可表示为

$$p_s(s_k) = n_k / n \quad k = 0, 1, \dots, L - 1$$

其中， s_k 为图像 $f(x, y)$ 的第 k 级灰度值

n_k 是 $f(x, y)$ 中具有灰度值 s_k 的像素的个数

47

3.2.3.1 直方图的概念

$$p_s(s_k) = n_k / n \quad k = 0, 1, \dots, L - 1$$

n_k 是 $f(x, y)$ 中具有灰度值 s_k 的像素的个数

$p_s(s_k)$ 给出了对灰度 s_k 出现概率的一个估计

直方图表示了一幅图像的整体灰度分布，不包含有空间位置信息的灰度分布

0	1	2	3
1	2	3	0
2	3	0	1
3	0	1	1

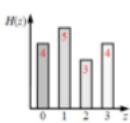


图 2.4.1 图像和直方图

3.2.3.2 直方图均衡化



图像问题：

由于图像传感器受光照影响，经常会导致图像整体偏暗、或整体偏亮现象发生



因此，需要增加像素灰度值的动态范围

→使图像全体对比度得到增强

51

3.2.3.2 直方图均衡化

直方图均衡化原理

借助直方图变换实现（归一的）灰度映射

均衡化（线性化）基本思想

变换原始图像的直方图为均匀分布

==> 灰度层次丰富

使像素灰度值的动态范围最大

==> 增强图像整体对比度（反差）

3.2.3.2 直方图均衡化

直方图均衡化的基本思想

直方图均衡是以累积分布函数变换为基础，将原始图像的直方图，由不均匀变换为均匀分布

一幅图像中灰度级 s_k 出现的概率(直方图)为

$$p_s(s_k) = n_k / n \quad k = 0, 1, \dots, L-1$$

其中 $p_s(s_k)$ 代表第 k 个灰度级出现的概率，
 s_k 常被归一化到 $[0, 1]$ 区间

53

3.2.3.2 直方图均衡化

对一幅原始图像进行直方图均衡化处理时，首先要确定一个变换函数 $t=EH(s)$ ，以完成由非均匀到均匀的变换。

上述变换要求所确定的变换函数 $t=EH(s)$ 需满足2个条件：

- ① 变换函数 $EH(s)$ 在区间 $0 \leq s \leq L-1$ 内是单值单增函数。
- ② 动态范围的一致性，即对 $0 \leq s \leq L-1$ ，有 $0 \leq EH(s) \leq L-1$

54

3.2.3.2 直方图均衡化

当满足变换函数的条件时，其数学模型可以给出如下保证：

- (a) $EH(s)$ 在区间 $0 \leq s \leq L-1$ 内是单值单增函数。

保证原图像各灰度级在变换后仍保持从黑到白(或白到黑)的排列顺序。单值是保证反变换存在。

- (b) 对 $0 \leq s \leq L-1$, 有 $0 \leq EH(s) \leq L-1$
保持变换前后动态范围的一致性。

55

3.2.3.2 直方图均衡化

- 累积直方图——满足条件的增强函数
1-D的离散函数
提供了图像像素灰度值的累积分布情况

$$t_k = EH(s_k) = \sum_{i=0}^k p_s(s_i) = \sum_{i=0}^k \frac{n_i}{n}$$

- (1) t_k 是 k 的单值单增函数
- (2) 灰度取值范围一致, $0 \leq t_k \leq 1$
- (3) 将 s 的分布转换为 t 的均匀分布

56

3.2.3.2 直方图均衡化

$$t_k = EH(s_k) = \sum_{i=0}^k p_s(s_i) = \sum_{i=0}^k \frac{n_i}{n}$$

- 上式给出的变换 称为直方图均衡化

上式的物理含义是：

由原图像从0→至第k个灰度级的概率密度函数的累积和，作为新图像的第k级灰度的灰度值。

3.2.3.2 直方图均衡化

这里需要说明的是

通常，不能证明离散变换能产生均匀概率密度函数的离散值。

- 由于数字图像是离散数据集合，因此



在数字图像处理中，无均匀直方图。

但可以认为，直方图均衡化公式的应用，
有展开输入图像直方图的一般趋势。

58

- 直方图均衡化其实也是一个点到点的映射，新的直方图需要等到所有点都映射完后才能求概率得到。

Week 5

- 注意区别，人眼不经过调整可以适应光强度范围为 10^5 范围，经过调整可以适应的总范围为： 10^{10} ，注意区分。

3.2.3 直方图变换

{以概率论为基础，通过改变图像的直方图来改变图像中像素的灰度，以实现图像增强。也常称直方图修正}

3.2.3.1 直方图的概念

3.2.3.2 直方图均衡化

3.2.3.3 直方图规定化

3.2.3.3 直方图规定化

上述例子表明，直方图均衡化不是万能的，需要有选择地增强某个范围灰度值。

➤解决方案：

采用较为灵活的直方图规定化方法，

有的书亦称为直方图匹配



➤关键技术：

正确选择规定化的函数，使输出图像具有规定的概率密度函数

3.2.3.3 直方图规定化

符号设定：

$p_s(s)$ 是原始图像的灰度级概率密度函数
(原始直方图)；

• $p_u(u)$ 是希望得到的图像的灰度级概率密
度函数 (规定直方图)

现在面临的问题是，



如何建立 $p_s(s)$ 与 $p_u(u)$ 之间的变换函数，
这是直方图规定化处理的关键！

7

借助直方图变换实现规定/特定的灰度映射

• (1) 对原始直方图进行灰度均衡化

$$t_k = EH_s(s_k) = \sum_{i=0}^k p_s(s_i) \quad k = 0, 1, \dots, M-1$$

• (2) 规定所需要的直方图，计算能使规定
直方图均衡化的变换

$$v_k = EH_u(u_k) = \sum_{i=0}^k p_u(u_i) \quad k = 0, 1, \dots, N-1; N \leq M$$

• (3) 求规定直方图的反变换函数

$$v = EH_u(u) \rightarrow u = EH_u^{-1}(v)$$

• (4) 将原始图像直方图中所得到灰度级 t

代入上面的反变换函数中

8

四个步骤

(1) 直方图规定化算法

① 首先，对原始图像的直方图进行均衡化

类似于直方图均衡化方法中的步骤，对原始图像进行均衡化处理

$$t_k = EH_s(s_k) = \sum_{i=0}^k p_s(s_i) \quad k = 0, 1, \dots, M-1$$

概率密度函数的累积和作为新灰度级

s_k 是原始图像的灰度级 设M是原始图像的灰度级数

9

(1) 直方图规定化算法

② 规定希望的直方图

即规定灰度概率密度函数 $p_u(u)$ ，并计算能使该直方图均衡化的变换。

规定直方图的累积变换函数

$$v_k = EH_u(u_k) = \sum_{i=0}^k p_u(u_i) \quad k = 0, 1, \dots, N-1; N \leq M$$

u_k 是规定图像的灰度级 设N是规定图像的灰度级数

考虑 $p_t(t)$ 与 $p_v(v)$ 具有同样均匀概率密度

如果用从原始图像中得到的均匀灰度级 t ,
替代希望得到的图像的均匀灰度级 v

(理论上两者具有相同的灰度级, 通过 t 、 v 作为桥梁,
建立 s 、 u 之间的联系)

$$\begin{aligned} v_k &= EH_u(u_k) \quad \text{寻找近似相等 } t, v \\ t_k &= EH_s(s_k) \quad \text{用 } t \text{ 替代 } v, t \rightarrow v \end{aligned}$$

根据 t 对应的 s ,
建立起 u 、 s 关系

$$\begin{aligned} u &= EH_u^{-1}(v) \\ u &= EH_u^{-1}(t) \end{aligned}$$

12

直方图规范化算法

$$\begin{aligned} v_k &= EH_u(u_k) \quad \text{寻找近似相等 } t, v \\ t_k &= EH_s(s_k) \quad \text{用 } t \text{ 替代 } v, t \rightarrow v \end{aligned}$$

根据 t 对应的 s ,
建立起 u 、 s 关系

由此, 可以得到反变换灰度级, 就是所
希望的概率密度函数 $p_u(u)$ 的灰度级 u

(1) 直方图规定化算法

将上述表达，用数学公式表示，如下式：

用t替代v， $t \rightarrow v$

$$u = EH_u^{-1}(v) = EH_u^{-1}(t)$$

④ 将原始图像直方图中所得到灰度级t代入上面的反变换函数中

$$t = EH_s(s)$$

即可获得新图像灰度级，由此可得到 $s \rightarrow u$ 的对应

14

(1) 直方图规定化算法

即对输入图像所有像素应用下式可得到输出图像

$$u = EH_u^{-1}(t) = EH_u^{-1}[EH_s(s)]$$

求上述反变换函数的解析式较为困难，离散条件下无法实现解析解，因此：

对于离散的数字图像，一般采用一种近似方法：

即找出t与v的近似值，以两者近似相等时的s、u对应，完成从s→u的变换

- 理想条件下认为t和v是相等的，因为二者都是均衡化后的结果，都是均匀分布的。

(2) 直方图规定化实例

③ 用均衡化中的 t_k 进行 EH_u 反变换，求 u 各值

$$u_k = EH^{-1}_u(t_k) = EH^{-1}_u(EH_s(s_k))$$

实际上，即找出 t_k 与 v_k 的近似值，如

$$v_3 = EH_u(u_3) = 0.15 \quad \text{接近于 } t_0 = 0.19$$

$$\text{即 } u_3 = EH^{-1}_u(t_0 = 0.19)$$

此时有

$$t_0 = EH_s(s_0) \quad \rightarrow \text{ 对应 } s_0 \rightarrow u_3 \quad u_3 = \frac{3}{7}$$

24

因为图像只取8个等间隔灰度级，故变换后的 t 值只能选择最靠近的一个灰度级的值

(2) 直方图规定化实例

$t_k = EH_s(s_k)$ 与 $v_k = EH_u(u_k)$ 的近似值对应

v_k	$EH_u(u_k)$
v_0	$(u_0=0)$ 0.
v_1	$(u_1=1/7)$ 0.
v_2	$(u_2=2/7)$ 0.
v_3	$(u_3=3/7)$ 0.15
v_4	$(u_4=4/7)$ 0.35
v_5	$(u_5=5/7)$ 0.65
v_6	$(u_6=6/7)$ 0.85
v_7	$(u_7=1)$ 1

t_k	$EH_s(s_k)$
t_0	$(s_0=0)$ 0.19
t_1	$(s_1=1/7)$ 0.44
t_2	$(s_2=2/7)$ 0.65
t_3	$(s_3=3/7)$ 0.81
t_4	$(s_4=4/7)$ 0.89
t_5	$(s_5=5/7)$ 0.95
t_6	$(s_6=6/7)$ 0.98
t_7	$(s_7=1)$ 1

25

(2) 直方图规定化实例

④ 由 v_k 和 t_k , 建立 u_k 与 s_k 像素数对应关系

原图像

v_k	u_k	n_k
v_0	$(u_0=0)$	0
v_1	$(u_1=1/7)$	0
v_2	$(u_2=2/7)$	0
v_3	$(u_3=3/7)$	
v_4	$(u_4=4/7)$	
v_5	$(u_5=5/7)$	
v_6	$(u_6=6/7)$	985
v_7	$(u_7=1)$	448

t_k	n_k
t_0	$(s_0=0)$ 790
t_1	$(s_1=1/7)$ 1023
t_2	$(s_2=2/7)$ 850
t_3	$(s_3=3/7)$ 656
t_4	$(s_4=4/7)$ 329
t_5	$(s_5=5/7)$ 245
t_6	$(s_6=6/7)$ 122
t_7	$(s_7=1)$ 81

希望 (规定) 图像

(2) 直方图规定化实例

得到的新直方图 ($n=4096$) 结果值 希望值

u_k	n_k	$p_u(u_k)$	$p_u(u_k)$
$u_{0=0}$	0	0.	0.
$u_{1=1/7}$	0	0.	0.
$u_{2=2/7}$	0	0.	0.
$u_{3=3/7}$	790	0.19	0.15
$u_{4=4/7}$	1023	0.25	0.20
$u_{5=5/7}$	850	0.21	0.30
$u_{6=6/7}$	985	0.24	0.20
$u_{7=1}$	448	0.11	0.15

结果与希望有误差 → 离散及多次近似导致; 连续函数准确

- 邻接和通路只是空间概念。连接和连通除了空间概念, 还包括灰度相同或相似。

3.3.1.2 像素的邻域和邻接

两个邻接像素之间边接触: **4-邻接**

4邻域:

像素 $p(x, y)$ 的4邻域是:

$(x+1, y); (x-1, y);$

$(x, y+1); (x, y-1)$

用 $N_4(p)$ 表示像素 p 的4邻域

$(x, y+1)$

$(x-1, y)$

$(x+1, y)$

$(x, y-1)$

P

Guojin Wang, Tsinghua University 第三章 空域图像增强

37

3.3.1.2 像素的邻域和邻接

两个邻接像素之间顶点接触: **对角-邻接**

D邻域:

像素 $p(x, y)$ 的D邻域是:

$(x+1, y+1); (x+1, y-1);$

$(x-1, y+1); (x-1, y-1)$

用 $N_D(p)$ 表示像素 p 的D邻域

$(x-1, y+1)$

$(x+1, y+1)$

P

$(x+1, y-1)$

$(x-1, y-1)$

Guojin Wang, Tsinghua University 第三章 空域图像增强

38

3.3.1.2 像素的邻域和邻接

两个邻接像素之间边接触/顶点接触: **8-邻接**

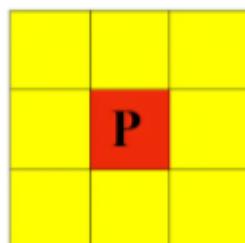
8邻域:

像素 $p(x, y)$ 的8邻域是:

4邻域的点 + D邻域的点

用 $N_8(p)$ 表示像素 p 的8邻域。

$$N_8(p) = N_4(p) + N_D(p)$$



3.3.1.3 像素间的连接和连通

像素间的连接

两个像素之间的一种空间和属性关系

• 两个像素是否连接：

- (1) 是否它们空间接触（邻接）
- (2) 是否它们的灰度值满足某个特定的相似准则（如它们灰度值相等，或同在一个灰度值集合V中取值）

3.3.1.3 像素间的连接和连通

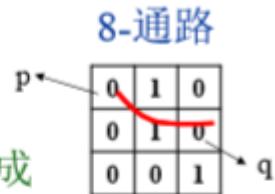
像素间的通路

通路:由一系列依次邻接的像素组成

从具有坐标(x, y)的像素 p 到具有坐标(s, t)的像素 q 的一条通路由一系列具有坐标 $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$ 的独立像素组成

这里 $(x_0, y_0) = (x, y), (x_n, y_n) = (s, t)$, 且 (x_i, y_i) 与 (x_{i-1}, y_{i-1}) 邻接, 其中 $1 \leq i \leq n$, n 为通路长度

4-邻接, 8-邻接 \Leftrightarrow 4-通路, 8-通路

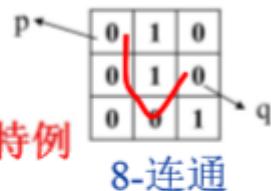


42

3.3.1.3 像素间的连接和连通

像素间的连通

连通是连接的推广, 连接是连通的特例



连通:由一系列依次连接的像素组成

从具有坐标(x, y)的像素 p 到具有坐标(s, t)的像素 q 的一条通路由一系列具有坐标 $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$ 的独立像素组成

这里 $(x_0, y_0) = (x, y), (x_n, y_n) = (s, t)$, 且 (x_i, y_i) 与 (x_{i-1}, y_{i-1}) 连接, 其中 $1 \leq i \leq n$, n 为连通长度

4-连接, 8-连接 \Leftrightarrow 4-连通, 8-连通

43

3.3.1.3 像素间的连接和连通

邻接、连接、通路、连通

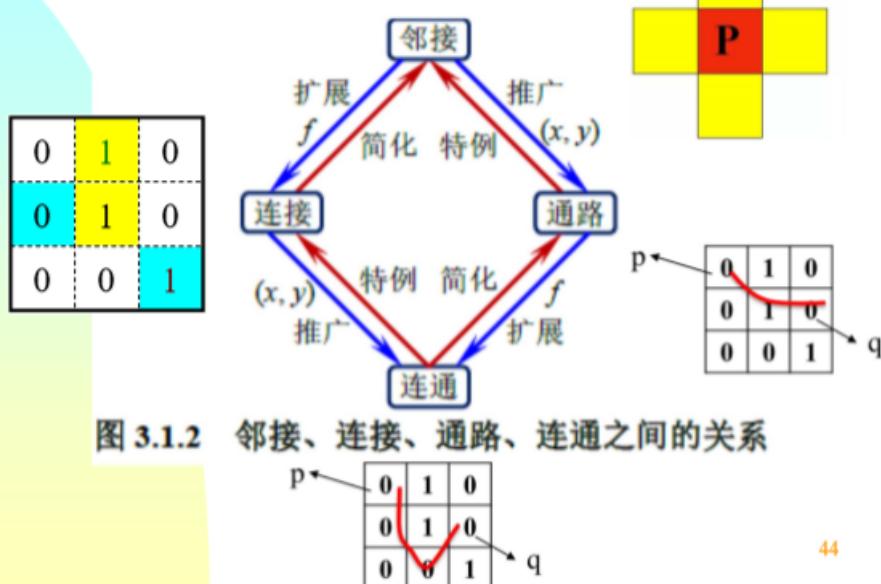


图 3.1.2 邻接、连接、通路、连通之间的关系

44

3.3.1.3 像素间的连接和连通

小结:

邻接: 仅考虑像素间的关系

连接: 还考虑像素的灰度值

通路: 仅借助邻接来定义

连通: 还考虑像素的连接

子集: 一般为满足连通条件

3.3.3 常见噪声

■ 噪声来源

数字图像的噪声主要来源于**图像的获取和传输过程**，如图像传感器的质量、环境条件和图像传输过程中**传输信道的噪声干扰**。

■ 一些重要的噪声类型

- 高斯噪声
- 瑞利噪声
- 伽马（爱尔兰）噪声
- 指数分布噪声
- 均匀分布噪声
- 脉冲噪声（椒盐噪声）



Week 6

- 平滑滤波使得图像反差减少

3.3.4.1 均值滤波器——线性平滑滤波器

作用：

线性平滑滤波器用于模糊处理和减小噪声
→**提取大目标时消除细节，缝隙**

线性平滑滤波器的输出，是包含在滤波模板内像素的简单平均值



因此，也称为均值滤波器，根据其性质，它属于低通滤波器

3.3.4.1 线性平滑滤波器

1. 邻域平均

模板系数都是正的（整数）

保持灰度值范围（所有系数之和为1）

卷积结果
除以总个数

$$g(x, y) = \frac{1}{n^2} \sum_{(s,t) \in N(x,y)} f(s, t)$$

例：

k_4	k_3	k_2
k_5	k_0	k_1
k_6	k_7	k_8

1	1	1
1	1	1
1	1	1

Guojin Wang, Tsinghua University 第三章 空域图像增强

3

3.3.4.1 线性平滑滤波器

2. 加权平均

离中心近的像素应对滤波结果有较大的贡献

$$g(x, y) = \frac{\sum_{(s,t) \in N(x,y)} w(s, t) f(s, t)}{\sum_{(s,t) \in N(x,y)} w(s, t)}$$

模板：中心系数大

周围系数小

➤要除以的系数？

1	2	1
2	4	2
1	2	1

图 3.3.2 一个加权平均模板

3.3.4.1 线性平滑滤波器

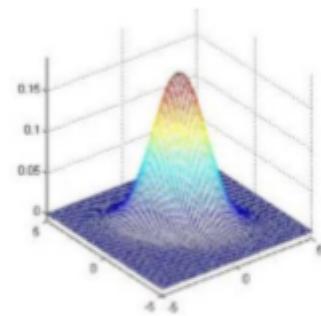
3. 高斯平均

根据高斯分布确定模板系数

模板
分解

$$\frac{1}{273} \begin{bmatrix} 1 & 4 & 7 & 4 & 1 \\ 4 & 16 & 26 & 16 & 4 \\ 7 & 26 & 41 & 26 & 7 \\ 4 & 16 & 26 & 16 & 4 \\ 1 & 4 & 7 & 4 & 1 \end{bmatrix}$$
$$\frac{n \times n}{n + n}$$
$$\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \end{bmatrix}$$

减少计算量



Guojin Wang, Tsinghua University 第三章 空域图像增强

5

3.3.4.1 线性平滑滤波器

3. 高斯平均

对高斯函数在整数位置 $-n, \dots, 0, \dots, +n$ 采样

表 3.3.2 1-D 高斯模板系数

$g(i)$	σ
1	0
1 1	1/4
1 2 1	1/2
1 3 3 1	3/4
1 4 6 4 1	1
1 5 10 10 5 1	5/4

Guojin Wang, Tsinghua University 第三章 空域图像增强

6

3.3.4.1 线性平滑滤波器

4. 边缘保持平滑

为了避免局部平均会模糊不同区域之间的边缘，考虑选择只在不包含边缘的邻域中进行平均

取中心像素的8个不同邻域，对每个邻域都统计其像素的灰度方差，并将灰度方差最小（方差小的邻域中像素灰度级的变化小，包含边缘的概率也小）的那个邻域中的像素均值赋给中心像素

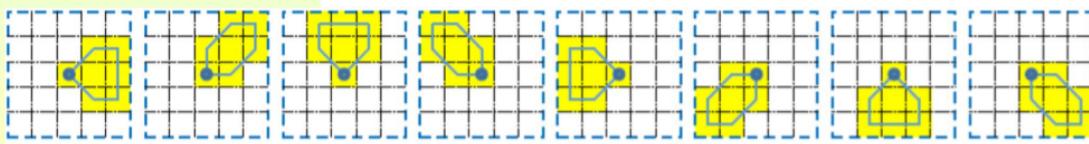


图 3.3.3 八个边缘保持平滑模板

3.3.4.2 中值滤波器——非线性平滑滤波

中值滤波器

Median filter

➤依靠n*n的模板对图像进行中值平滑处理

利用区域的中值进行平滑

➤进行中值滤波的目的：

让与周围像素灰度值相差较大的像素改
取与周围灰度值接近的值

→消除孤立噪声能力强，

→不是简单取均值，产生模糊较少，保持细节

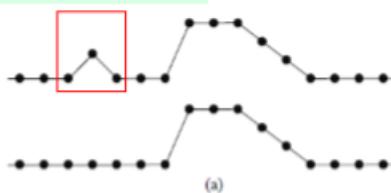
9

3.3.4.2 非线性平滑滤波

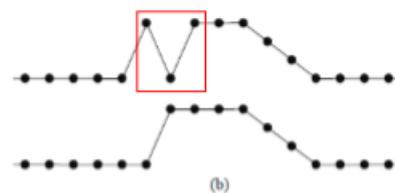
中值滤波原理

设模板尺寸为M, $M = 2r + 1$, r为模板半径

$$g_j = \text{median}[f_{j-r}, f_{j-r+1}, \dots, f_j, \dots, f_{j+r}]$$



消除孤立的脉冲而
不对边缘产生影响



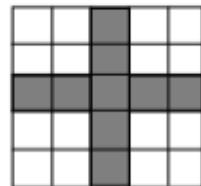
接近边缘的孤立脉
冲会使边缘偏移

图 3.4.2 1-D 中值滤波示例

3.3.4.2 非线性平滑滤波

(1) 中值滤波器 实现步骤:

- A、模板在图中滑动，中心与像素重合
- B、读取模板下各对应像素的灰度值
- C、将灰度值由小→大排列
- D、找出排列中的中间值
- E、将中间值赋给中心点



11

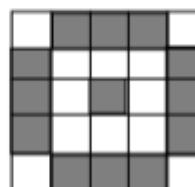
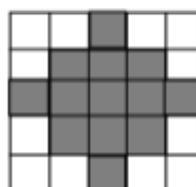
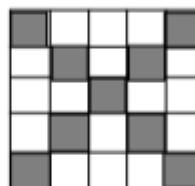
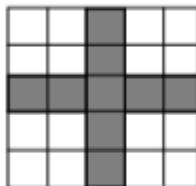
3.3.4.2 非线性平滑滤波

(2) 中值滤波器模板

4邻域和8邻域的延伸

中值滤波器的消噪效果，与
模板尺寸、
参与运算的像素数
模板排列形状
均有密切关系

具体选取什么样形状的滤波模
板要根据图像内容而定



$D_4\text{距离} = |x-s| + |y-t| \leq 2$ 与中心像素的欧氏距离在[2,2.5]

3.3.5 锐化滤波器

基于微分的图像锐化增强

均值产生钝化的效果，而均值与积分相似，
由此而联想到，微分能不能产生相反的效果，即
锐化的效果？结论是肯定的。

• 锐化滤波器的分类

- ✓ 一阶微分滤波器—梯度算子
- ✓ 二阶微分滤波器—拉普拉斯算子

3.3.5.1 梯度算子——非线性锐化滤波

1. 基于梯度的锐化滤波

利用微分可以锐化图像（积分平滑图像）

梯度：对应一阶导数

最常用的微分矢量

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} \end{bmatrix}^T = [G_X \quad G_Y]^T$$

-1		1
-1		1
-1		1

1	1	1
-1	-1	-1

图 3.4.7 两个差分模板

(需要用2个模板分别沿 X 和 Y 方向计算)

3.3.5.1 梯度算子

1. 基于梯度的锐化滤波

在实际滤波中，常常只使用梯度矢量的幅度
(即矢量的模)

- 非线性组合模板输出值（矢量转标量）
幅度以2为范数/模计算（对应欧氏距离）

$$|\nabla f_{(2)}| = \text{mag}(\nabla f) = [G_X^2 + G_Y^2]^{1/2}$$

以1为范数（城区距离） 以 ∞ 为范数（棋盘距离）

$$|\nabla f_{(1)}| = |G_X| + |G_Y| \quad |\nabla f_{(\infty)}| = \max\{|G_X|, |G_Y|\}$$

3.3.5.1 梯度算子

(1) 梯度算子

$$f(x, y) \text{ 在 } (x, y) \text{ 的梯度定义} \quad \nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

向量的模
称为梯度 $|\nabla f| = \sqrt{G_x^2 + G_y^2}$

梯度向量的分量本身是线性算子，但它的模不是线性的，因为用到了平方和开方运算

一阶导数可以用于检测图像中的一个点是否是边缘的点

二阶导数的符号可以用于判断一个边缘像素是在边缘亮的一边还是暗的一边

3.3.5.1 梯度算子

(2) 近似处理

$$|\nabla f| = \sqrt{G_x^2 + G_y^2}$$

此算式的计算量很大，可以对梯度采用数字近似



$$|\nabla f| \approx |G_x| + |G_y|$$

一阶微分最简单的近似处理

$$G_x = z_8 - z_5 \quad G_y = z_6 - z_5$$

z1	z2	z3
z4	z5	z6
z7	z8	z9

注意图像处理中有时候将竖直轴设为x轴。

•

◆ 一阶微分算子和二阶微分算子均可以作为图像边缘检测算子。

21
From Gonzalez' book

3.3.5.2 拉普拉斯算子——线性锐化滤波器

2. 基于二阶微分的图像增强-拉普拉斯算子

均值产生钝化的效果，而均值与积分相似，由此而联想到，微分能不能产生相反的效果，即锐化的效果？结论是肯定的。

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$
$$\frac{\partial^2 f}{\partial x^2} = 2f(x, y) - f(x+1, y) - f(x-1, y)$$
$$\frac{\partial^2 f}{\partial y^2} = 2f(x, y) - f(x, y+1) - f(x, y-1)$$

$$\nabla^2 f(x, y) = 4f(x, y) - f(x+1, y) - f(x-1, y) - f(x, y+1) - f(x, y-1)$$

Guojin Wang, Tsinghua University 第三章 空域图像增强

22

3.3.5.2 拉普拉斯算子

2. 拉普拉斯算子

既可考虑4-邻域，也可考虑8-邻域

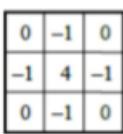
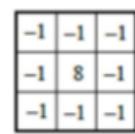
	
(a)	(b)

图 3.3.4 两种拉普拉斯算子模板

✓ 系数之和均为零

✓ 尺度变换到 [0, L-1]

拉普拉斯算子可增强图像中的灰度不连续边缘，减弱灰度值缓慢变化区域的对比度，将结果叠加到原始图像上，就可以得到锐化后的图像

3.3.5.3 最大最小锐化变换——非线性锐化滤波

3. 最大-最小锐化变换

最大值滤波和最小值滤波结合

将一个模板覆盖区域里的中心像素值与该区域里的最大值和最小值进行比较，然后将中心像素值用与其较接近的极值（最大或最小值）替换

$$S[f(x, y)] = \begin{cases} g_{\max}(x, y) & \text{如果 } g_{\max}(x, y) - f(x, y) \leq f(x, y) - g_{\min}(x, y) \\ g_{\min}(x, y) & \text{其他} \end{cases}$$

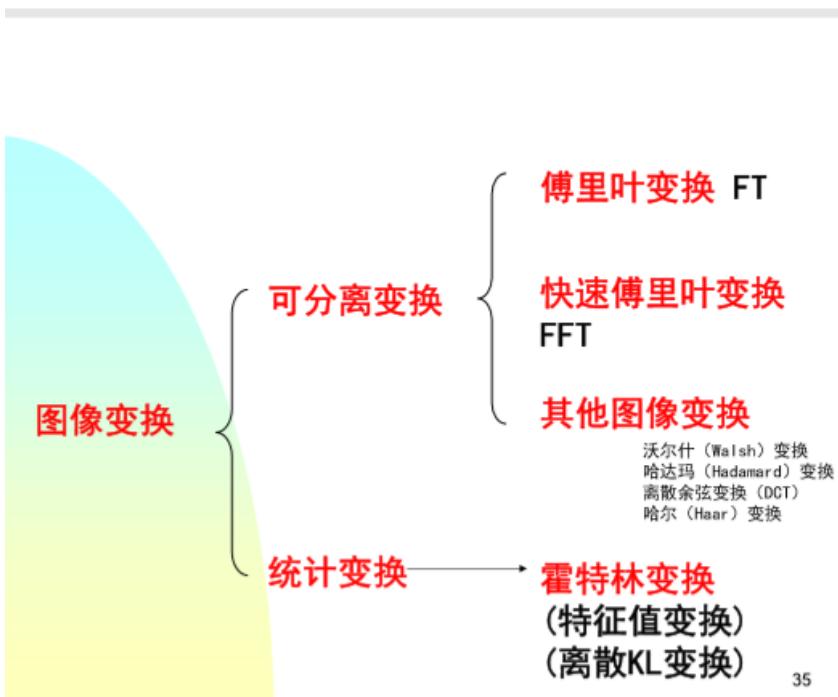
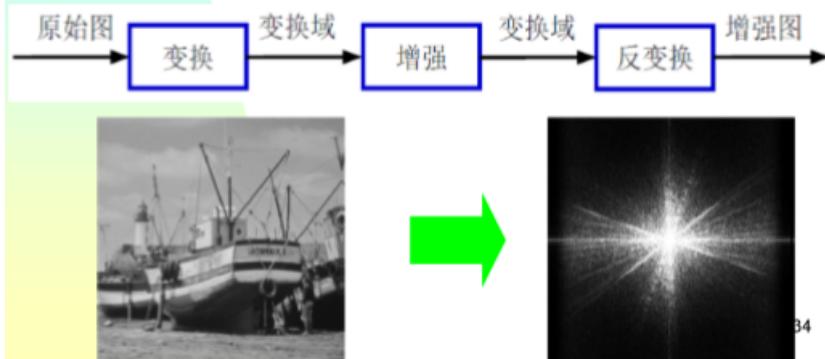
迭代进行： $S^{n+1}[f(x, y)] = S\{S^n[f(x, y)]\}$

3.4 局部增强

- 全局增强：把一幅图像作为一个整体
- 局部增强：对图像某些局部区域的细节进行增强
- 用全局增强不能保证在所关心的局部区域得到所需要的增强效果
- 局部增强多了一个选择局部区域的步骤
- 直接利用局部信息以达到局部增强目的

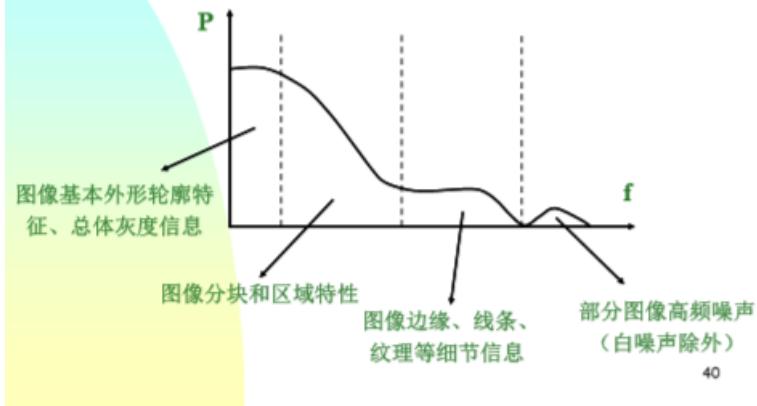
4.1 频域图像增强概述

对图像的加工，除了可以在图像空间直接进行外，也可以考虑将图像变换到其他空间，在其他空间进行加工。这些利用变换后空间的方法称为基于变换域的方法。



4.1 频域图像增强概述

图像信号的频域模型



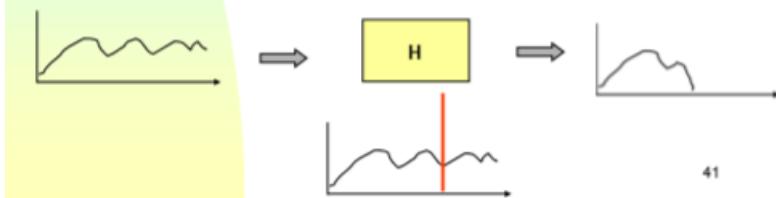
4.1 频域图像增强概述

频域增强原理

在频域空间，图像的信息表现为不同频率分量的组合

如果能让一定范围内的频域分量受到抑制，而让其他分量不受影响，就可以改变输出图像的频率分布

进而达到所需要的图像增强的目的



4.1 频域图像增强概述

频域图像增强，主要借助滤波器实现

不同滤波器，根据其滤除的频率和保留的频率不同，从而实现不同的增强效果

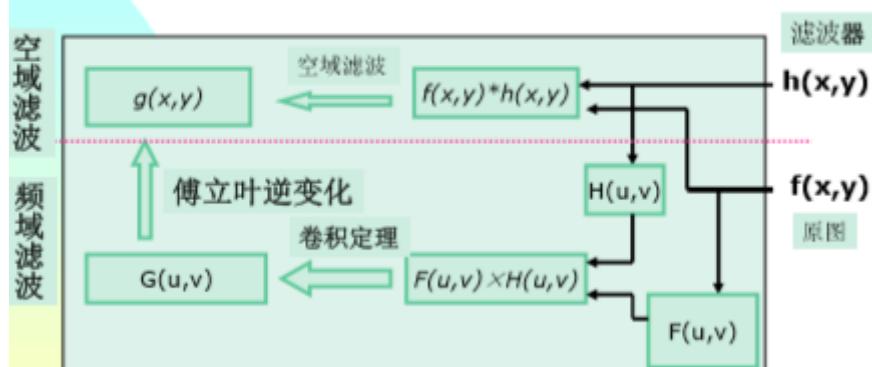


滤除高频、保留低频：消除噪声、平滑

滤除低频、保留高频：获得边缘、增强

42

频域增强与空域增强的关系



43

•

频域增强与空域增强的关系

在频域中分析图像的频率成分与图像的视觉效果间的对应关系比较直观

有些在图像空间比较难以表述的图像增强任务可以比较简单的在频域中表述，所以在频域中设计滤波器比较直接。

将图像转换到频域空间的傅里叶变换有快速算法，有些增强任务在频域可以更快地实现。

$$\begin{aligned} g(x, y) &= h(x, y) * f(x, y) && \text{邻域模板处理} \\ G(u, v) &= H(u, v)F(u, v) && \text{在频域计算} \\ g(x, y) &= F^{-1}[H(u, v)F(u, v)] && \text{反变换到空域} \end{aligned}$$

44

频域增强与空域增强的关系

图像频谱给出图像全局的特性，所以频域增强不是对逐个像素进行

空域增强中无论使用点操作还是模板操作，每次都只是基于图像中部分像素进行的

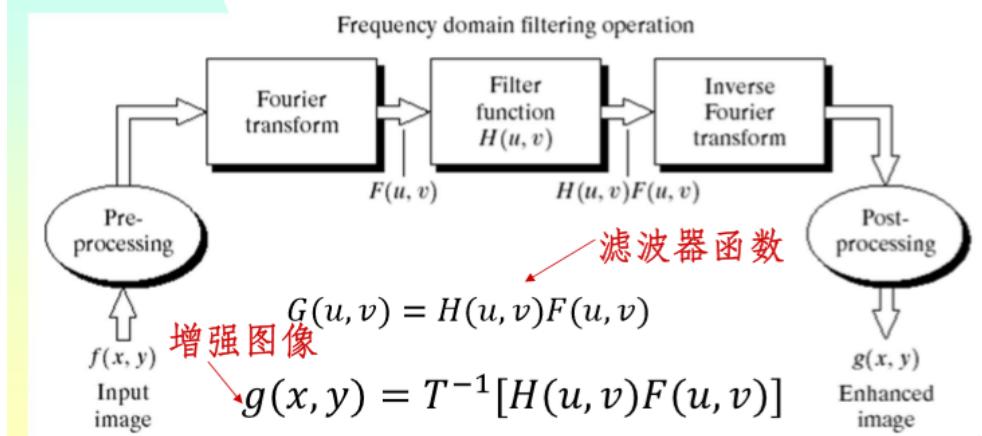
在硬件实现上，如果两个域内滤波器具有相同尺寸，借助快速傅里叶变换在频域进行滤波一般效率更高

但是，在空域常可以使用较小的滤波器来取得与在频域使用较大滤波器相似的滤波效果，计算量也有可能反而小

45

在频域中进行增强的操作主要步骤：

- 计算所需增强图像的傅里叶变换
- 与一个根据需要设计的滤波器函数相乘
- 再将获得的结果进行傅里叶反变换
→ 得到增强的图像



46

FIGURE 4.5 Basic steps for filtering in the frequency domain.

4.2.1 2-D傅里叶变换

2-D变换核

对于 $M \times N$ 二维图像

正变换: $f(x, y) \Rightarrow F(u, v)$ 正向变换核

$$F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi(ux/M+vy/N)}$$

反变换: $f(x, y) \Leftarrow F(u, v)$ 反向变换核

$$f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) e^{j2\pi(ux/M+vy/N)}$$

$$\begin{cases} u = 0, 1, \dots, M-1 \\ v = 0, 1, \dots, N-1 \end{cases}$$

4.2.1 2-D傅里叶变换

对于 $N \times N$ 二维图像

2-D变换核

正变换: $f(x, y) \Rightarrow F(u, v)$ 正向变换核

$$F(u, v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi(ux+vy)/N}$$

反变换: $f(x, y) \Leftarrow F(u, v)$ 反向变换核

$$f(x, y) = \frac{1}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u, v) e^{j2\pi(ux+vy)/N}$$

4.2.1 2-D傅里叶变换

分离性

1个2-D变换核可分解成2个1-D变换核

$$\frac{1}{N} \exp[-j2\pi(ux+vy)/N] = \frac{1}{\sqrt{N}} \exp[-j2\pi ux/N] \frac{1}{\sqrt{N}} \exp[-j2\pi vy/N]$$

$$\frac{1}{N} \exp[j2\pi(ux+vy)/N] = \frac{1}{\sqrt{N}} \exp[j2\pi ux/N] \frac{1}{\sqrt{N}} \exp[j2\pi vy/N]$$

对称性

分离后的两部分函数形式一样

52

4.2.1 2-D傅里叶变换

正变换分解

列变换 $F(x, v) = \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} f(x, y) \exp[-j2\pi vy/N] \quad x, v = 0, 1, \dots, N-1$

行变换 $F(u, v) = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} F(x, v) \exp[-j2\pi ux/N] \quad u, v = 0, 1, \dots, N-1$

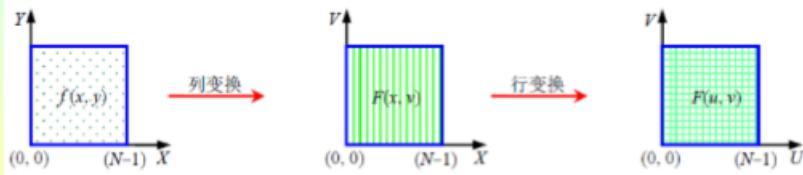


图 4.2.3 由 2 步 1-D 变换计算 2-D 变换

53

4.2.2 傅里叶变换定理

1. 平移定理

平移性质表明利用指数项相乘可以进行
变换结果中心的移动

$$\begin{aligned} & \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} F(u, v) \exp[-j2\pi(au + bv)/N] \exp[j2\pi(xu + yv)/N] \\ &= \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} F(u, v) \exp[j2\pi[(x-a)u + (y-b)v]/N] \\ &= f(x - a, y - b) \end{aligned}$$

反向变换核

56

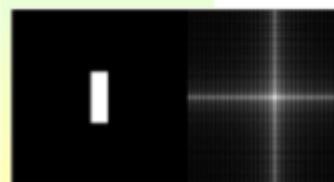
4.2.2 傅里叶变换定理

1. 平移定理

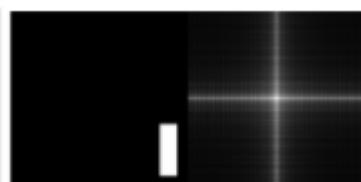
$$f(x, y) \Leftrightarrow F(u, v)$$

$$F(u - c, v - d) \Leftrightarrow f(x, y) \exp[j2\pi(cx + dy)/N]$$

矩形居中

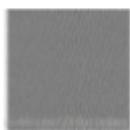
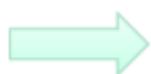


矩形移动至右下

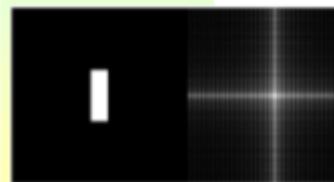


4.2.2 傅里叶变换定理

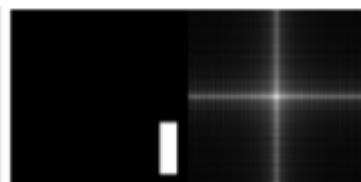
1. 平移定理



矩形居中



矩形移动至右下



2. 旋转定理

$$f(x, y) \Leftrightarrow F(u, v)$$

$f(x, y)$ 旋转角度 θ_0 , $F(u, v)$ 也将转过相同的角度
 $F(u, v)$ 旋转角度 θ_0 , $f(x, y)$ 也将转过相同的角度

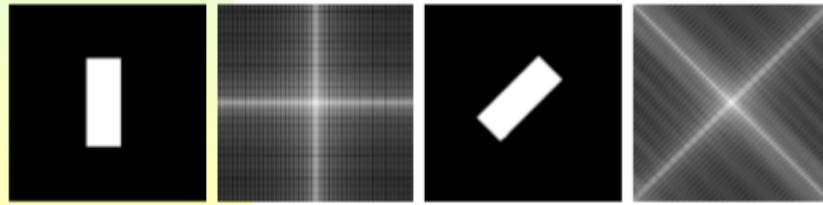
$$\begin{aligned}x &= r\cos\theta \\y &= r\sin\theta\end{aligned}$$

$$f(r, \theta) \Leftrightarrow F(\omega, \varphi)$$

$$f(r, \theta + \theta_0) \Leftrightarrow F(\omega, \varphi + \theta_0)$$

极坐标

$$\begin{aligned}u &= \omega\cos\varphi \\v &= \omega\sin\varphi\end{aligned}$$



4.2.2 傅里叶变换定理

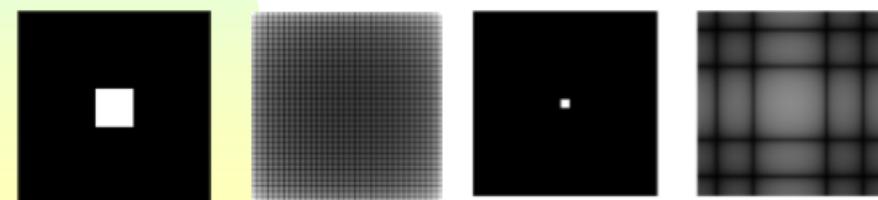
3. 尺度定理

$$f(x, y) \Leftrightarrow F(u, v)$$

(相似定理) : 尺度(放缩)变化时的性质

$$af(x, y) \Leftrightarrow aF(u, v)$$

$$f(ax, by) \Leftrightarrow \frac{1}{|ab|} F\left(\frac{u}{a}, \frac{v}{b}\right)$$



Week 7

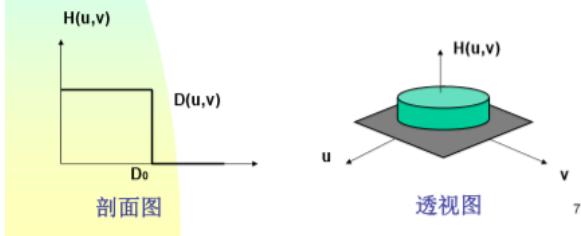
- 看清2D频谱图，离原点越远表示频率越高。

4.3.1 低通滤波

1. 理想低通滤波器

图像中的细节、边缘和噪声对应傅里叶变换中高频部分

→ 可选择合适 $H(u, v)$ 减弱高频分量



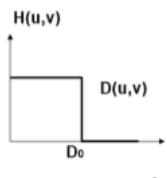
4.3.1 低通滤波

2D 理想低通滤波器传递函数，满足

$$H(u, v) = \begin{cases} 1 & \text{若 } D(u, v) \leq D_0 \\ 0 & \text{若 } D(u, v) > D_0 \end{cases}$$

其中截断频率 $D_0 > 0$

$$D(u, v) = \sqrt{u^2 + v^2}$$



8

4.3.1 低通滤波

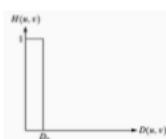
理想低通滤波器不能用硬件来实现，这是因为实际的元器件，无法实现 $H(u, v)$ 从 1→0 的陡峭突变。

理想低通滤波器在滤除噪声的同时，产生不良后果

→ 图像变得模糊，而且，



D_0 越低，滤除噪声越彻底，高频分量损失越严重，图像越模糊。

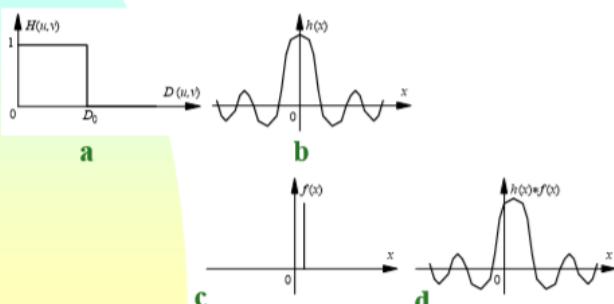


10

4.3.1 低通滤波

理想低通滤波的模糊

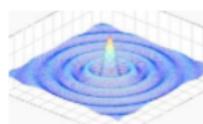
理想低通滤波产生“振铃”现象



4.3.1 低通滤波

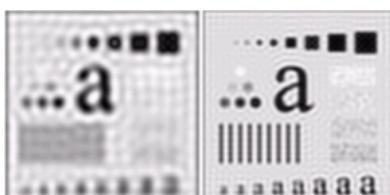
理想低通滤波的模糊

理想低通滤波所产生的“振铃”现象 2-D
图像上表现为一系列同心圆环
圆环半径反比于截断频率



理想低通滤波

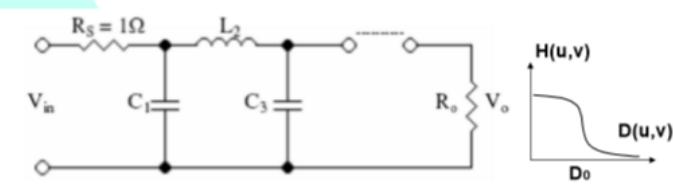
器在滤除噪声的同时，随所选截止频率 D_0 的不同，会发生不同程度的振铃现象。



4.3.1 低通滤波

2. 巴特沃斯低通滤波器

由于实际应用中理想低通难于物理实现



N 阶巴特沃斯低通滤波器传递函数定义

$$H(u, v) = \frac{1}{1 + [D(u, v)/D_0]^{2n}}$$

15

4.3.1 低通滤波

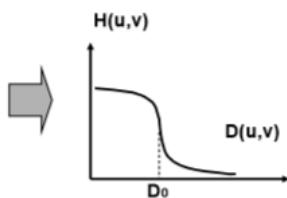
2. 巴特沃斯低通滤波器

巴特沃斯滤波器是一种通频带之频率响应曲线平坦无涟波的信号处理滤波器。它也被称作最大平坦滤波器。

4.3.1 低通滤波

1 阶巴特沃斯低通滤波器

高频低频间的过渡较光滑，可以有效减少振铃效应



巴特沃斯低通滤波器的高低频之间，没有明显的截止点。



通常取使 H 最大值下降到一个%时的频率为截止频率 D_0 ，如 50%， $1/\sqrt{2}$

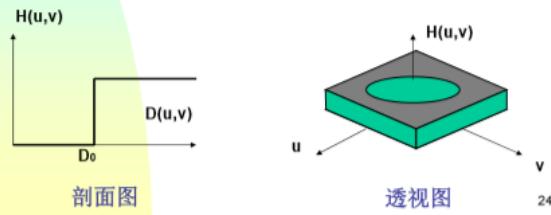
19

- 随着阶数增加，巴特沃斯低通滤波器也会产生振铃现象，因为它越来越接近理想低通滤波器。

4.3.2 高通滤波

1. 理想高通滤波器

图像中的边缘对应傅里叶变换中高频部分
→ 锐化图像可选择高通滤波。



4.3.2 高通滤波

2D 理想高通滤波器传递函数满足

$$H(u,v) = \begin{cases} 0 & \text{若 } D(u,v) \leq D_0 \\ 1 & \text{若 } D(u,v) > D_0 \end{cases}$$

但类似理想低通の場合，



理想高通滤波器也无法用实际器件实现。 25

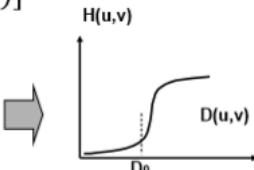
4.3.2 高通滤波

2. 巴特沃斯高通滤波器

n 阶巴特沃斯高通滤波器传递函数定义

$$H(u,v) = \frac{1}{1 + [D_0/D(u,v)]^{2n}}$$

1 阶巴特沃斯高通滤波器，高低频间连续过渡

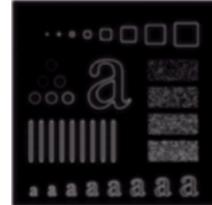


27

高频加强滤波

滤波后的图像有一个共同特点：

它们背景的平均强度减小到接近黑色。



这归结为在这些图像上所用的高通滤波器除去了傅里叶变换的零频率成分。

为获得好的显示效果，解决办法是把一定比例的原始图像加到过滤后的结果中去。

31

高频加强滤波

有时，用一幅图像的高频成分强调增强的作用是有益的。这种情况下，在高通滤渡器函数前简单地乘以一个常数，再增加一个偏移可使零频率不被滤波器除掉。

这种处理叫做**高频加强**，为如下传递函数：

$$H_{\text{hfe}}(u, v) = a + bH_{\text{hp}}(u, v)$$

这里，**a≥0且b>a**，**a**的典型值在**0.25**到**0.5**之间，**b**的典型值在**1.5**到**2.0**之间。

32

- 高通滤波后的结果特征较弱，但它显示出微弱的主要图像边缘。

4.4 带通和带阻滤波

带通和带阻滤波器，是通过或抑制某个区间频率成分的一种滤波方法

1. 带阻滤波器

$$H(u, v) = \begin{cases} 1 & \text{当 } D(u, v) < D_0 - \frac{W}{2} \\ 0 & D_0 - \frac{W}{2} \leq D(u, v) \leq D_0 + \frac{W}{2} \\ 1 & D(u, v) > D_0 + \frac{W}{2} \end{cases}$$

W为带宽, D₀为半径

38

4.4 带通和带阻滤波

带阻滤波器透视图

→主要用于消除周期性噪声。

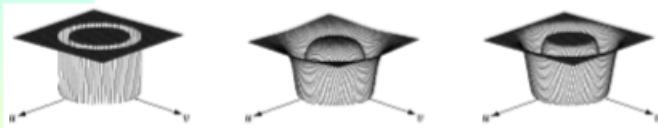


FIGURE 5.15 From left to right, perspective plots of ideal, Butterworth (of order 1), and Gaussian bandreject filters.

理想滤波器

巴特沃斯(1阶)滤波器

高斯带阻滤波器

4.4 带通和带阻滤波

N 阶巴特沃斯带阻滤波器：

$$H(u, v) = \frac{1}{1 + [\frac{D(u, v)W}{D^2(u, v) - D_0^2}]^{2n}}$$

W为带宽
D₀为半径

2. 带通滤波器：

执行与带阻滤波器相反的操作，帮助提取噪声模式

$$H_{bp}(u, v) = 1 - H_{br}(u, v)$$

带阻滤波器传递函数

40

4.5.1 亮度成象模型

图像成象模型

2-D亮度函数: $f(x, y)$

亮度是能量的量度, 一定不为零且为有限值

$$0 < f(x, y) < \infty$$

(1) 入射到可见场景上的光量

(2) 场景中目标对入射光反射的比率

照度成分 $i(x, y)$ 和反射成分 $r(x, y)$

4.5.1 亮度成象模型

图像成象模型

$f(x, y)$ 与 $i(x, y)$ 和 $r(x, y)$ 都成正比

$$f(x, y) = i(x, y)r(x, y)$$

$i(x, y)$ 的值是由光源决定的

$$0 < i(x, y) < \infty$$

$r(x, y)$ 的值是由场景中的目标特性所决定的

$$0 < r(x, y) < 1$$

4.5.2 同态滤波增强

同态滤波

同时将图像亮度范围进行压缩
和将图像对比度进行增强

这个概念在
1960年由Thomas
Stockham, Alan
V. Oppenheim和
Ronald W.
Schafer在麻省理工学院提出

- 利用频域增强思路

将乘性转换为加性，处理后再转换回来

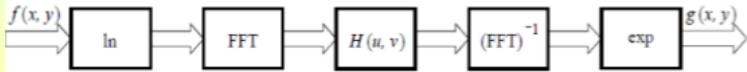


图 4.5.1 同态滤波流程图

4.5.2 同态滤波增强

(1) 两边取对数：

$$\ln f(x, y) = \ln i(x, y) + \ln r(x, y)$$

(2) 两边取傅氏变换： $F(u, v) = I(u, v) + R(u, v)$

(3) 用一个频域函数 $H(u, v)$ 处理 $F(u, v)$ ：

$$H(u, v)F(u, v) = H(u, v)I(u, v) + H(u, v)R(u, v)$$

(4) 反变换到空域： $h_f(x, y) = h_i(x, y) + h_r(x, y)$

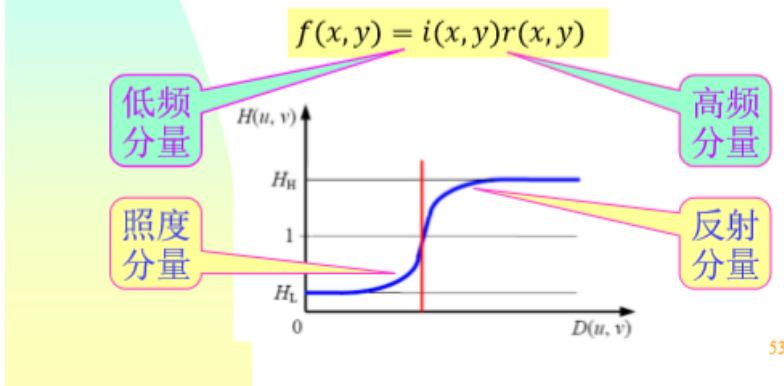
(5) 两边取指数：

$$g(x, y) = \exp|h_f(x, y)| = \exp|h_i(x, y)| \cdot \exp|h_r(x, y)|$$

4.5.2 同态滤波增强

图像成像模型

- $f(x, y)$ 与 $i(x, y)$ 和 $r(x, y)$ 都成正比

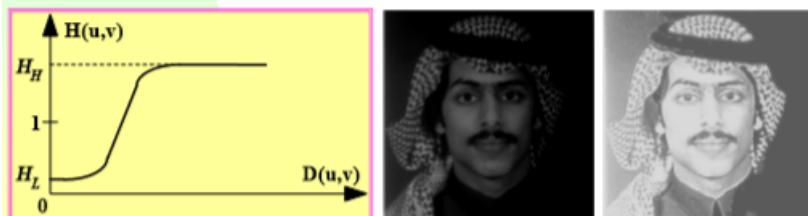


53

4.5.2 同态滤波增强

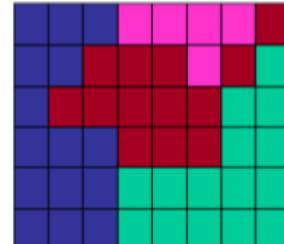
特点：能消除乘性噪声，还能同时压缩图像的整体动态范围和增加图像中相邻区域间的对比度

典型曲线 效果示例 ($H_L=0.5$, $H_H=2.0$)



Week 8

- **连通组元**: 对 S 中任1个像素 p , 所有与 p 相连通且又在 S 中的像素的集合, 合起来称为 S 中的1个连通组元。
 - 图像中相同连通组元中的2个像素互相连通, 不同连通组元中的各个像素互不连通。
- **区域**: 一个连通集合称为图像的一个区域
- **边界(轮廓)**: 是区域的一个子集, 它将该区域与其他区域分离开, 组成区域的边界像素本身属于该区域而在其邻域中有不属于该区域的像素。



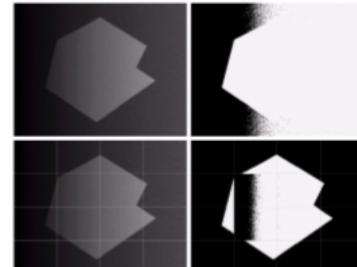
- 图像分割: 是从图像处理到图像分析的一个关键技术。

传统图像分割的两类方法

- 边缘方法: 这种方法假设图像分割结果的某个子区域, 在原来图像中一定会有边缘存在。
- 区域方法: 这种方法假设图像分割结果的某个子区域一定具有相同的性质, 而不同区域的像素没有共同性质。



边缘分割方法结果



区域分割方法结果

实际应用中，这两种方法各有优缺点，可以考虑把两者结合起来，进行图像分割。

目前，随着计算机处理能力的提高，很多方法不断涌现，如：

- 基于彩色分量分割
- 纹理图像分割
- 运动目标分割
- 深度学习像素分割等



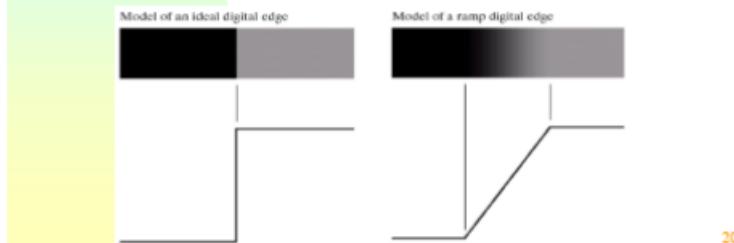
边缘检测

- 边缘像素是指图像中灰度突变的那些像素
- 边缘（或边缘线段）是连接边缘像素的集合
- 因此对于边缘检测，最为有效的检测方法是检测灰度级的变化。

颜色变化也由灰度变化表现。

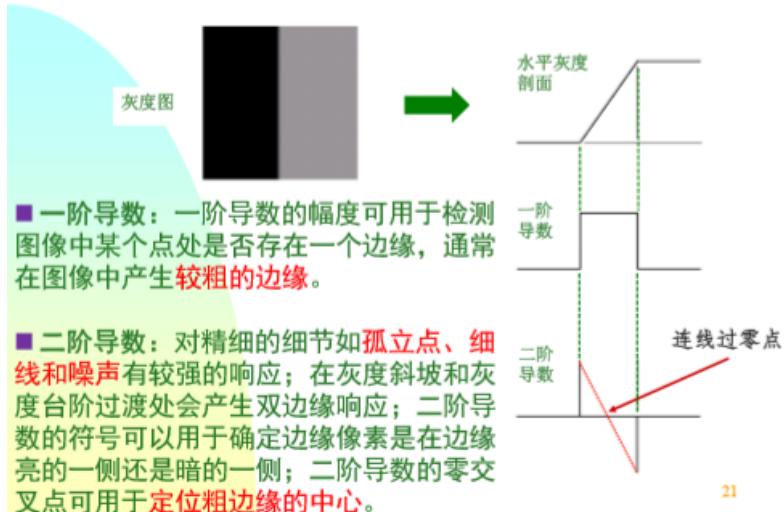
边缘检测的原理

- 理想边缘所具有的模型特性：
每个像素都处在灰度跃变的垂直台阶上
- 由于噪声等干扰获得的实际边缘的模型特性：
边缘模型为具有“类斜面”的剖面，边缘为一组彼此相连的点集→边缘宽度取决于斜坡长度



20

边缘检测的方法



21

- 决定一个值是否有效的选择方法，是使用阈值，导数响应阈值。

→如果一个点的二维一阶导数 大于 指定阈值阈值
→则 此点为图像中的边缘点



一组相连的
边缘点定义为
一条边缘。

23

- Robert算子定位比较精确，但由于不包括平滑，所以对于噪声比较敏感。
- 一阶、二阶梯度算子对噪声都敏感，在检测边缘前最好进行去噪处理。
-

二阶梯度算子

■ Laplace算子

对噪声敏感，常产生双像素宽的边缘，很少直接用于边缘检测。可用于孤立点、细线的检测，及已知边缘像素后确定该像素是在亮的一侧还是暗的一侧。改进版本包括Marr-Hildreth算子等。



Laplace算子检测孤立点

将二阶导数近似为：

$$\nabla^2 f(x, y) = f(x + 1, y) + f(x - 1, y) + f(x, y + 1) \\ + f(x, y - 1) - 4f(x, y)$$

将上式扩展为包括四个对角点，模板为：

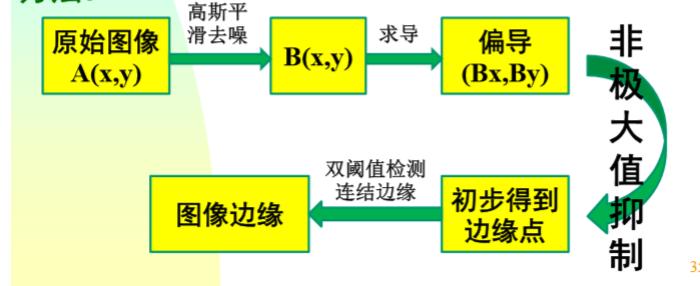
1	1	1
1	-8	1
1	1	1

如果在某点处该模板的响应的绝对值超过了阈值T，则认为此时模板中心位置为孤立点。将孤立点标注为1，其余点标注为0得到二值化输出图像。

$$g(x, y) = \begin{cases} 1 & \text{if } |R(x, y)| \geq T \\ 0 & \text{otherwise} \end{cases}$$

■ Canny边缘检测算子

Canny边缘检测算子是John F. Canny于1986年提出的一个多级边缘检测算法。属于先平滑后求导数的方法。



Canny边缘检测算法步骤

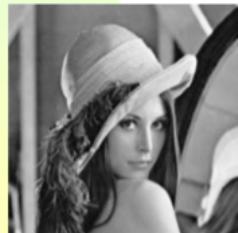
step1：用高斯滤波器平滑图象；

令 $f(x, y)$ 表示输入图像， $G(x, y)$ 表示高斯函数：

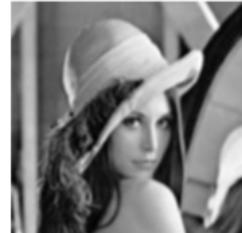
$$G(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}}$$

用 G 和 f 的卷积形成平滑后的图像 $f_s(x, y)$ ：

$$f_s(x, y) = G(x, y) \star f(x, y)$$



原图



高斯滤波

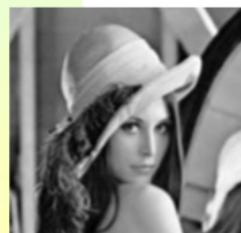
Canny边缘检测算法步骤

step2：计算梯度幅值图像和角度图像；

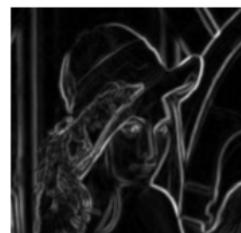
如前所述，幅值和角度分别采用下式计算：

$$\text{幅值 } M(x, y) = \sqrt{g_x^2 + g_y^2}$$

$$\text{边缘法线的方向 } \alpha(x, y) = \tan^{-1} \left[\frac{g_y}{g_x} \right]$$



高斯滤波



幅值

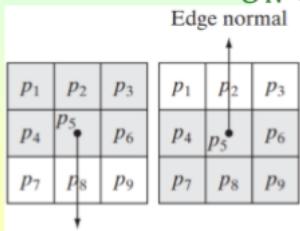
Canny边缘检测算法步骤

step3：对梯度幅值进行非极大值抑制，细化边缘；

令 d_1, d_2, d_3, d_4 表示 3×3 区域的四个基本边缘方向：

1. 找到最接近 $\alpha(x, y)$ 的方向 d_k ；
2. 若 $M(x, y)$ 至少小于沿 d_k 的两个邻居之一，则令 $g_N(x, y) = 0$ (抑制); 否则 $g_N(x, y) = M(x, y)$ 。

其中 $g_N(x, y)$ 为非极大值抑制后的图像。



水平边缘法线的
两个可能方向



幅值



NMS

40

• Canny边缘检测算法步骤

step4：用双阈值算法检测和连接边缘。

2. g_{NH} 中的非零值都包含在 g_{NL} 中，因此要从 g_{NL} 中删除所有来自 g_{NH} 的非零像素。

$$g_{NL}(x, y) = g_{NL}(x, y) - g_{NH}(x, y)$$

3. g_{NL} 和 g_{NH} 的非零像素可以看做是”弱”和”强”边缘，认为 g_{NH} 中的所有强像素均为有效边缘。

4. g_{NH} 中边缘通常会存在缝隙，需要进行连接



$g_{NL} = g_{NL} - g_{NH}$



g_{NH}

42

Canny边缘检测算法步骤

step4: 用双阈值算法检测和连接边缘。

1. 在 g_{NH} 中定位下一个未被访问的边缘像素 p ;
2. 将 g_{NL} 中将所有与 p 满足八连通的弱像素标记为有效边缘像素;
3. 若 g_{NH} 中所有的非零像素已被访问, 则跳到4, 否则返回1;
4. 将 g_{NL} 中未标记为有效边缘像素的所有像素置零



43

- ■ Canny的目标是找到一个最优的边缘检测算法
 - 好的检测 – 算法能够尽可能多地标识出图像中的实际边缘。
 - 好的定位 – 标识出的边缘要与实际图像中的实际边缘尽可能接近。
 - 最小响应 – 图像中的边缘只能标识一次，并且可能存在的图像噪声不应标识为边缘。

边缘连接

理论上，边缘检测应该仅产生处在边缘上的一组像素点

由于噪声、不均匀照明产生的边缘间断、以及其他虚假亮度间断产生的影响，使得到的一组像素很少能完整地描绘一条边缘



因此，需要采用连接处理，将边缘像素点组合成有意义的边缘

48

局部处理

连接边缘点最简单的方法之一，是分析每一个边缘点 (x, y) 的一个邻域 $(3 \times 3, 5 \times 5)$ 内像素的梯度特性（如强度、方向），

然后

- 依据事先预定的准则，
- 连接相似的边缘点，
- 形成由共同满足准则的像素组成的一条边缘

49

•

确定边缘点相似性的两个主要性质：

- (1) 用于生成边缘点的梯度算子的强度
- (2) 梯度向量的方向

对梯度算子强度，(1) 的性质由 ∇f 值给定，即当

$$|\nabla f(x, y) - \nabla f(x_0, y_0)| \leq E \quad E \text{为非负阈值}$$

则在 (x, y) 的邻域内，坐标为 (x_0, y_0) 的边缘点在幅度上相似于位于 (x, y) 的像素点

50

对梯度向量的方向，(2) 的性质由下式给定，即当

$$|\alpha(x, y) - \alpha(x_0, y_0)| < A \quad A \text{为非负角阈值}$$

其中 $\alpha(x, y) = \arctan\left(\frac{G_y}{G_x}\right)$

则在 (x, y) 的邻域内坐标为 (x_0, y_0) 的边缘点在角度上相似于位于 (x, y) 的像素点

且 (x, y) 处的边缘方向垂直于此点处的梯度向量方向

51

如果梯度大小和梯度方向准则得到满足，

则在 (x, y) 的邻域内的点 (x_0, y_0) 就与位于 (x, y) 的像素点连接



在图像中每个位置重复这一操作，邻域的中心从一个像素转移到另一个像素，并记录下两个相连接的点，

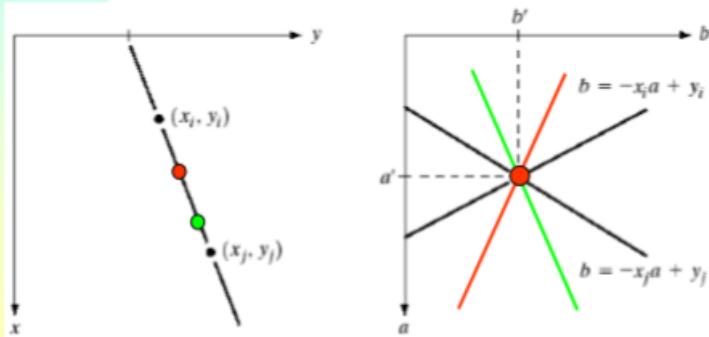
→由此即获得有意义的边缘

- Hough 不仅适用于检测直线，也适用于形如 $g(v, c) = 0$ 的任何函数，如：圆等。

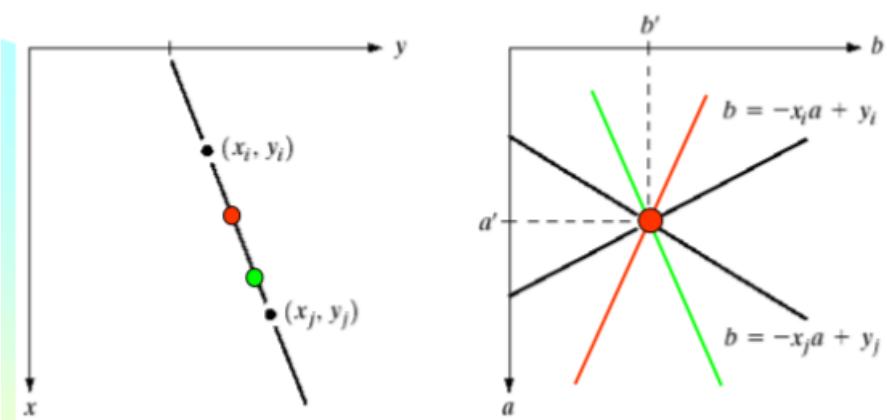
•

由上分析可得结论：

xy平面内直线 $y_i = ax_i + b$ 上所有的点，在参数空间中都有一条过点 (a', b') 的直线



60



若把所有直线切成很小的小段，则落在 (a', b') 位置的小段应最多

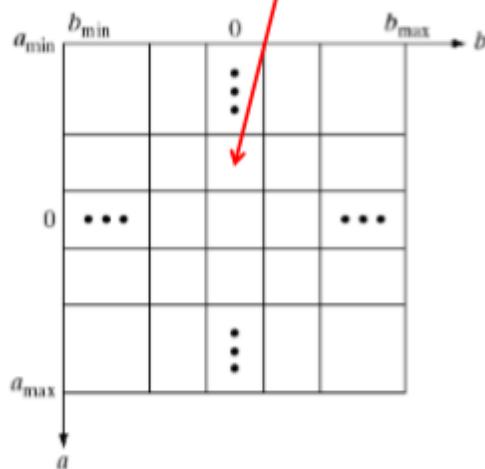
累加的最终结果， $A(i, j)$ 中的每一个值 Q ，都对应 xy 平面上的一条直线

$$y = a_i x + b_j$$



Q 的值，确定该条直线上的边缘点的数目；

Q 较大的值，确定由较多共线点确定的一条直线

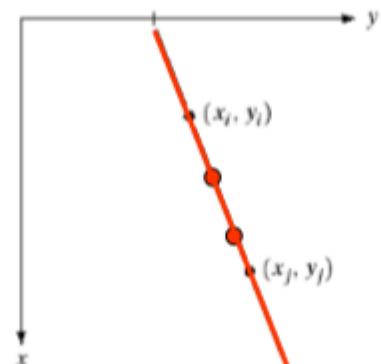
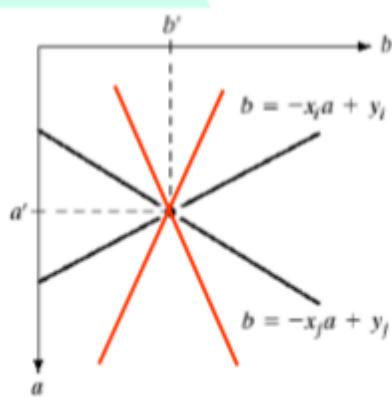


64

Q 较大的值，确定由较多共线点确定的直线



定义一个数目阈值，即可求出一条直线



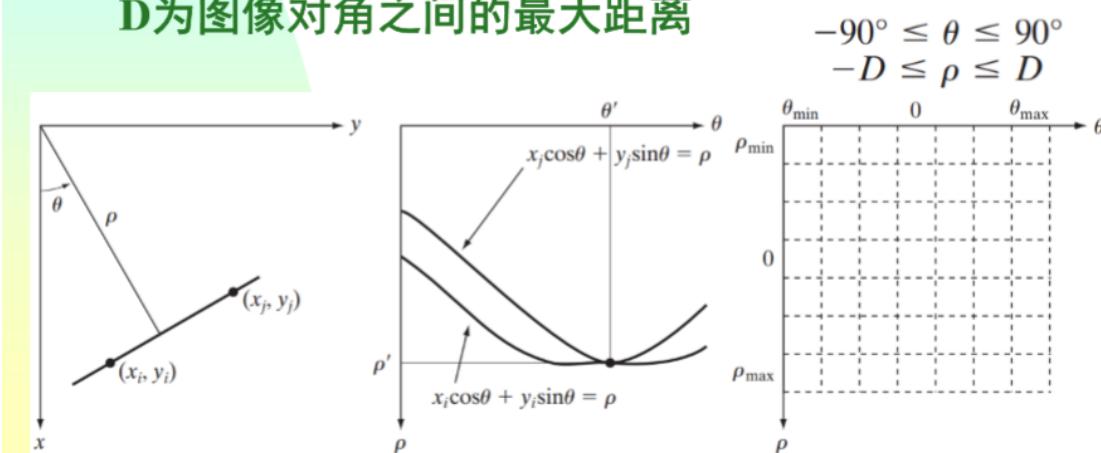
65

当采用 $y = ax + b$ 的直线方程时，如果直线接近竖直方向，则 a, b 的值都可能趋向无穷大而是计算量大增（因为累加器尺寸将会很大）。此时可以采用直线的极坐标系表示：

$$x \cos \theta + y \sin \theta = \rho$$

此时在 $\rho\theta$ 参数空间构造累加数组

D 为图像对角之间的最大距离



- 2020数字图像处理交流群 (123)



郭雅宁

老师我想问下是不是边缘检测最后的输出结果一般都是二值图像呀，就是都需要在计算出梯度幅值的基础上进行阈值的判断呢，因为好像matlab自带的edge函数输出是二值图像



王子浩+2016011219

边缘检测输出不是边缘和非边缘吗？



郭雅宁

我不知道使用sobel等模板计算出的图像算不算是边缘检测还是只是锐化滤波？



王贵锦

需要阈值得到边缘点



郭雅宁

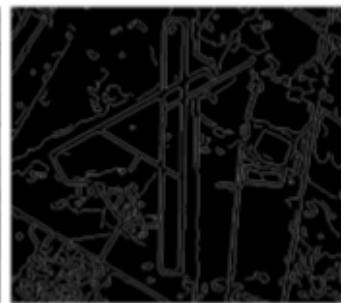
明白了，谢谢老师~

实例：提取机场跑道的两条边

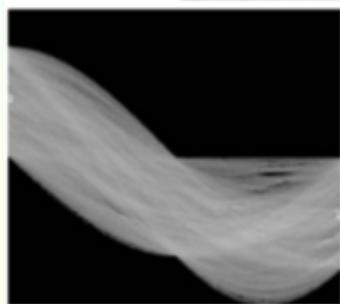
机场航拍图



canny算子检测



霍夫参数空间



提取的跑道线



叠加到原图

无人机航拍图



canny算子检测



叠加到原图



叠加到原图

