计算机视觉——图像预处理

2022年春季

桑 农

- 图像预处理的目的：

纠正几何失真 → 图像坐标变换

提高视觉质量 → 灰度映射

直方图修正

降低噪声干扰 → 空域滤波

第4章　图像预处理

- 4.1 图像坐标变换
- 4.2 灰度映射
- 4.3 直方图修正
- 4.4 空域滤波

4.1 图像坐标变换

- 4.1.1　　基本坐标变换
- 4.1.2　　几何失真校正

4.1.1 基本坐标变换

- 1. 变换的表达

$$\boldsymbol{v}' = \boldsymbol{A}\,\boldsymbol{v}$$

$$\boldsymbol{v} = \begin{bmatrix} x & y & 1 \end{bmatrix}^{\mathrm{T}} \qquad \boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

$$\boldsymbol{v}' = \begin{bmatrix} x' & y' & 1 \end{bmatrix}^{\mathrm{T}}$$

图像上各点的新齐次坐标

= 变换矩阵 × 图像上各点的原齐次坐标

4.1.1　基本坐标变换

- 2.　平移变换

$$T=\begin{bmatrix}1 & 0 & x_0\\ 0 & 1 & y_0\\ 0 & 0 & 1\end{bmatrix}\qquad T^{-1}=\begin{bmatrix}1 & 0 & -x_0\\ 0 & 1 & -y_0\\ 0 & 0 & 1\end{bmatrix}$$

原图

平移变换

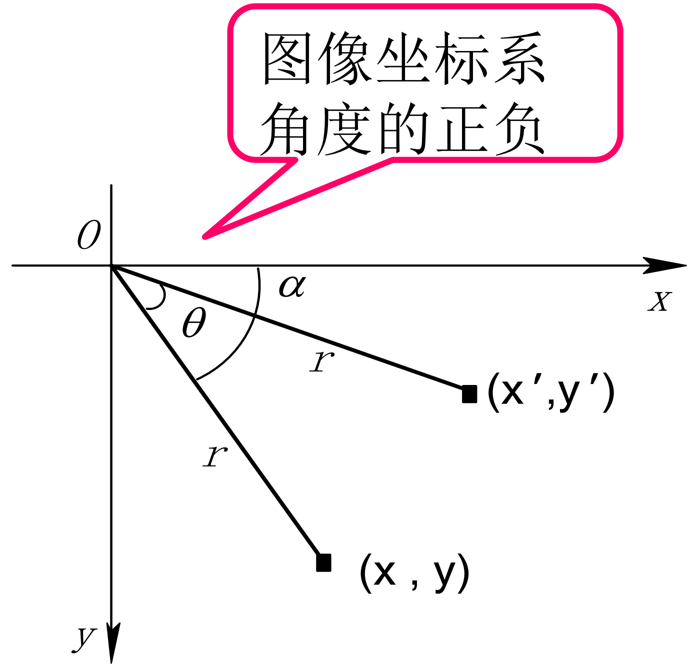
4.1.1 基本坐标变换

- 3. 尺度变换

$$\boldsymbol{S}=\begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \boldsymbol{S}^{-1}=\begin{bmatrix} 1/S_x & 0 & 0 \\ 0 & 1/S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

4.1.1 基本坐标变换

- 4. 旋转变换

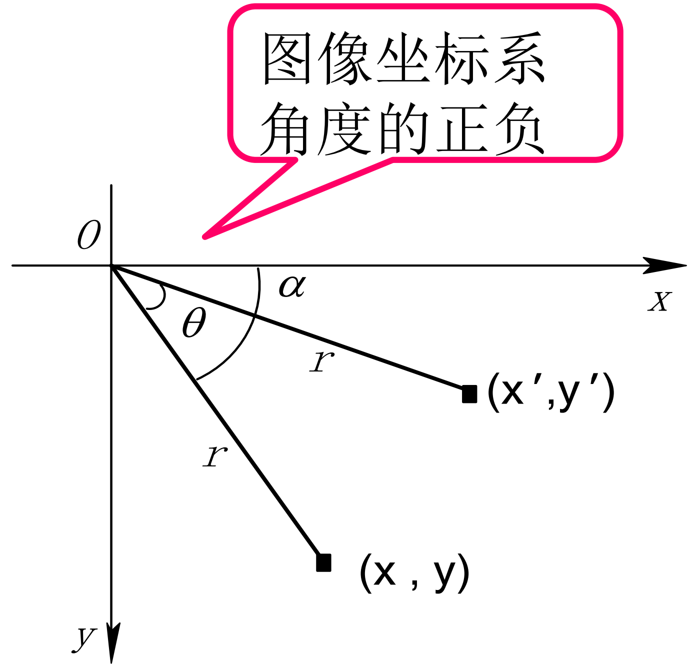


$$
\begin{cases}
x = r\cos\alpha \\
y = r\sin\alpha
\end{cases}
$$

$$
\begin{cases}
x' = r\cos(\alpha-\theta) \\
y' = r\sin(\alpha-\theta)
\end{cases}
$$

4.1.1　基本坐标变换

- 4.　旋转变换

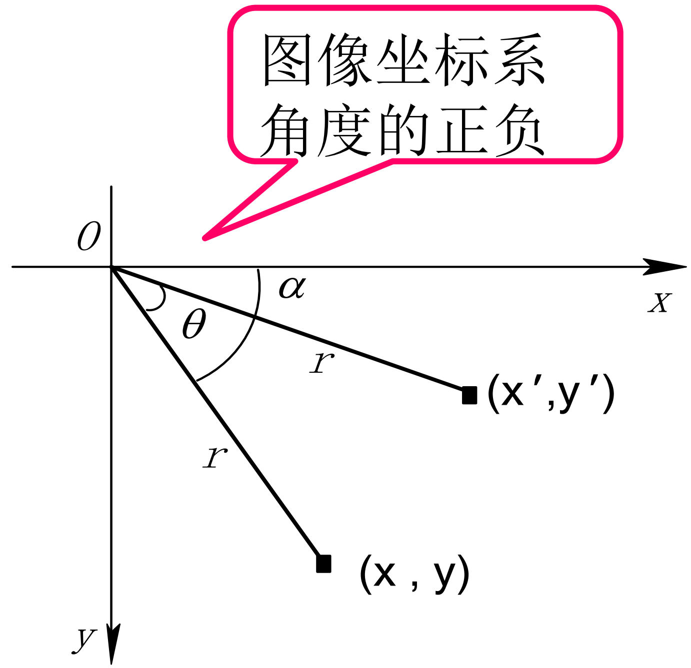


$$\begin{cases} x = r\cos\alpha \\ y = r\sin\alpha \end{cases}$$

$$\begin{cases} x' = r\cos(\alpha-\theta) = r\cos\alpha\ \cos\theta + r\sin\alpha\sin\theta \\ y' = r\sin(\alpha-\theta) = r\sin\alpha\ \cos\theta - r\cos\alpha\sin\theta \end{cases}$$

4.1.1 基本坐标变换

- 4. 旋转变换



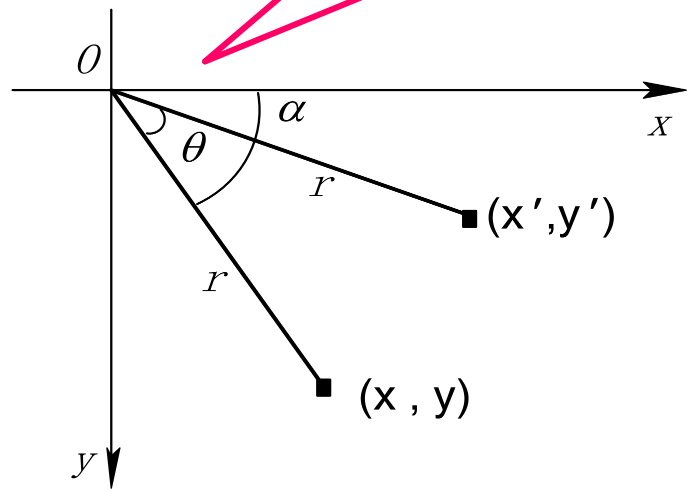
$$\begin{cases} x = r\cos\alpha \\ y = r\sin\alpha \end{cases}$$

$$\begin{cases} x' = r\cos(\alpha-\theta) = r\cos\alpha\ \cos\theta + r\sin\alpha\sin\theta \\ y' = r\sin(\alpha-\theta) = r\sin\alpha\ \cos\theta - r\cos\alpha\sin\theta \end{cases}$$

4.1.1 基本坐标系变换

- 4. 旋转变换

图像坐标系
角度的正负



$$\begin{cases} x = r\cos\alpha \\ y = r\sin\alpha \end{cases}$$

$$\begin{cases} x' = r\cos(\alpha-\theta) = \underline{r\cos\alpha}\ \cos\theta + \underline{r\sin\alpha}\sin\theta = x\cos\theta + y\sin\theta \\ y' = r\sin(\alpha-\theta) = \underline{r\sin\alpha}\ \cos\theta - \underline{r\cos\alpha}\sin\theta = -x\sin\theta + y\cos\theta \end{cases}$$

4.1.1　基本坐标变换

- 4.　旋转变换

$$R=\begin{bmatrix}\cos\theta & \sin\theta & 0\\ -\sin\theta & \cos\theta & 0\\ 0 & 0 & 1\end{bmatrix}\qquad R^{-1}=\begin{bmatrix}\cos\theta & -\sin\theta & 0\\ \sin\theta & \cos\theta & 0\\ 0 & 0 & 1\end{bmatrix}$$

4.1.1 基本坐标变换

- 5. 变换级连

$$v' = R\left[S(Tv)\right] = Av$$

例：平移→旋转→反平移

$$A = T^{-1}RT = \begin{bmatrix} 1 & 0 & -x_0 \\ 0 & 1 & -y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & x_0 \\ 0 & 1 & y_0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos\theta & \sin\theta & x_0\cos\theta + y_0\sin\theta - x_0 \\ -\sin\theta & \cos\theta & -x_0\sin\theta + y_0\cos\theta - x_0 \\ 0 & 0 & 1 \end{bmatrix}$$

4.1.1　基本坐标变换

- 5.　变换级连

平移（50, 50）→ 尺度（1.4, 1.4）→ 旋转30°

原图

组合变换

4.1.1　基本坐标变换

- 6.　基本坐标变换小结

$$T=\begin{bmatrix}1 & 0 & x_0\\ 0 & 1 & y_0\\ 0 & 0 & 1\end{bmatrix}\quad S=\begin{bmatrix}S & 0 & 0\\ 0 & S & 0\\ 0 & 0 & 1\end{bmatrix}\quad R=\begin{bmatrix}\cos\theta & \sin\theta & 0\\ -\sin\theta & \cos\theta & 0\\ 0 & 0 & 1\end{bmatrix}$$

$$A=\begin{bmatrix}a_{11} & a_{12} & a_{13}\\ a_{21} & a_{22} & a_{23}\\ 0 & 0 & 1\end{bmatrix}$$

x和y方向可以有不同的尺度及旋转变换

4.1.1 基本坐标变换

- 7. 其他变换

斜切（扭曲） $\begin{bmatrix} 1 & 0.5 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

原图

斜切变换

4.1.2 几何失真校正

- 在许多实际的图像采集过程中，图像像素之间的空间关系发生变化，即图像产生了几何失真或几何畸变
- 需要通过几何变换来校正失真图像各像素位置，以重新得到像素间原来应有的空间关系
- 对于灰度图像处理考虑空间关系以外，还要考虑灰度关系

4.1.2 几何失真校正

1. 空间变换

$$x' = s(x, y)$$
$$y' = t(x, y)$$

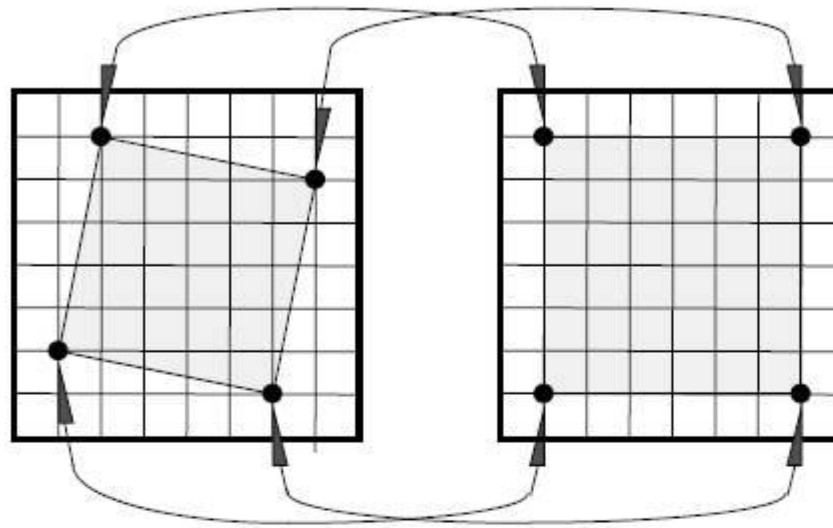


图 4.1.1 失真图和校正图的对应点

$$s(x, y) = k_1 x + k_2 y + k_3$$
$$t(x, y) = k_4 x + k_5 y + k_6$$

线性失真

$$s(x, y) = k_1 x + k_2 y + k_3 xy + k_4$$
$$t(x, y) = k_5 x + k_6 y + k_7 xy + k_8$$

双线性失真

4.1.2 几何失真校正

2. 灰度插值

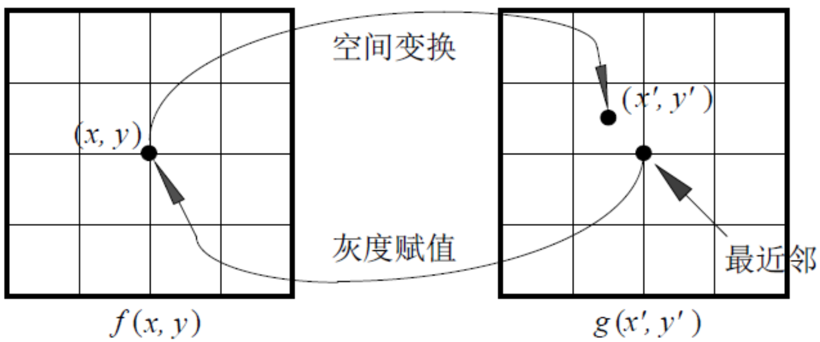


图 4.1.2 灰度插值示意图

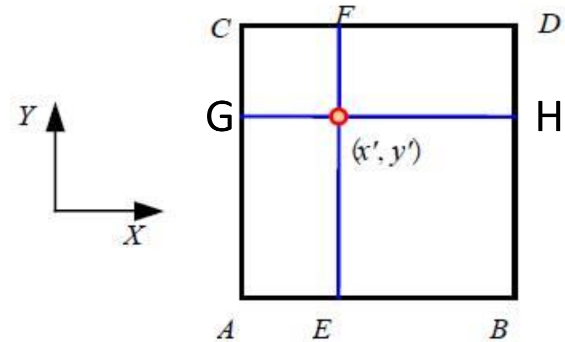


图 4.1.3 双线性插值示意图

$$x' - i = \mathrm{AE} = \mathrm{CF}$$

$$y' - j = \mathrm{AG} = \mathrm{BH}$$

$$\mathrm{AB} = \mathrm{AC} = \mathrm{BD} = \mathrm{CD} = 1$$

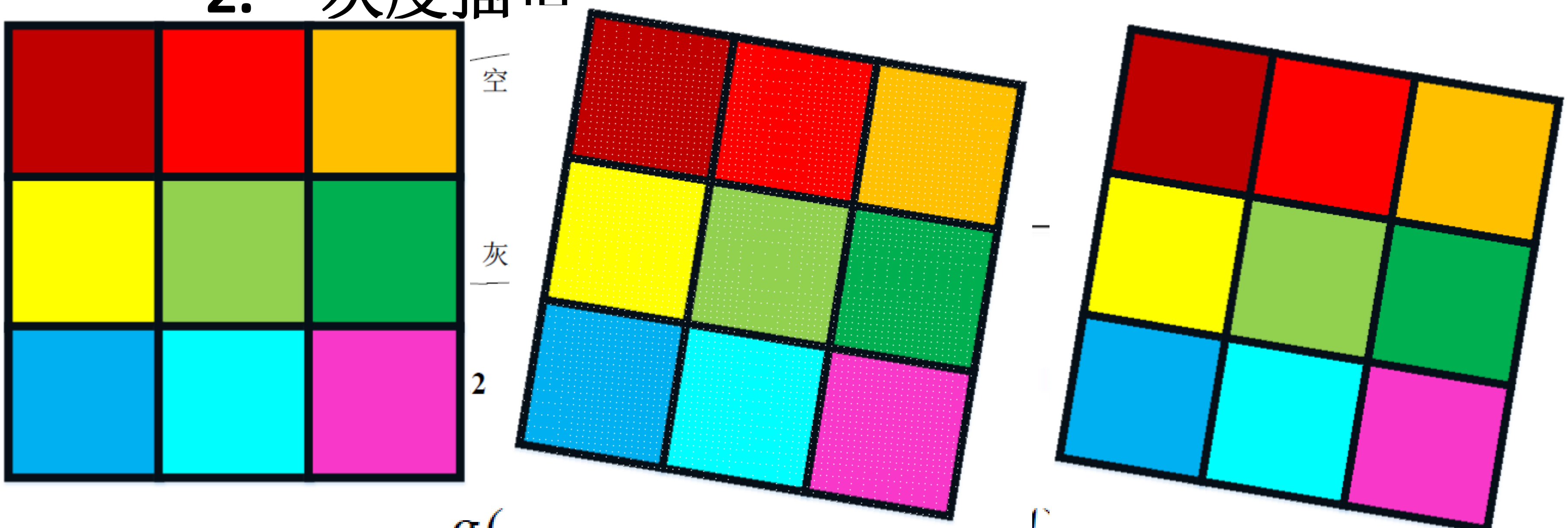
$$g(E) = (x' - i)[g(B) - g(A)] + g(A)$$

$$g(F) = (x' - i)[g(D) - g(C)] + g(C)$$

$$g(x', y') = (y' - j)[g(F) - g(E)] + g(E)$$

4.1.2　几何失真校正

2.　灰度插值



$$g(E)=(x'-i)[g(B)-g(A)]+g(A)$$

$$g(F)=(x'-i)[g(D)-g(C)]+g(C)$$

$$g(x',y')=(y'-j)[g(F)-g(E)]+g(E)$$

4.1.2 几何失真校正

2. 灰度插值

空间

灰度

$$g(E)=(x'-i)[g(B)-g(A)]$$

$$g(F)=(x'-i)[g(D)-g(C)]+g(C)$$

$$g(x',y')=(y'-j)[g(F)-g(E)]+g(E)$$

4.2 灰度映射

- 4.2.1　　灰度映射原理
- 4.2.2　　灰度映射示例

4.2.1 灰度映射原理

基于图像像素的点操作

映射函数

$$t = T(s)$$

灰度级到灰度级的变换，与位置无关

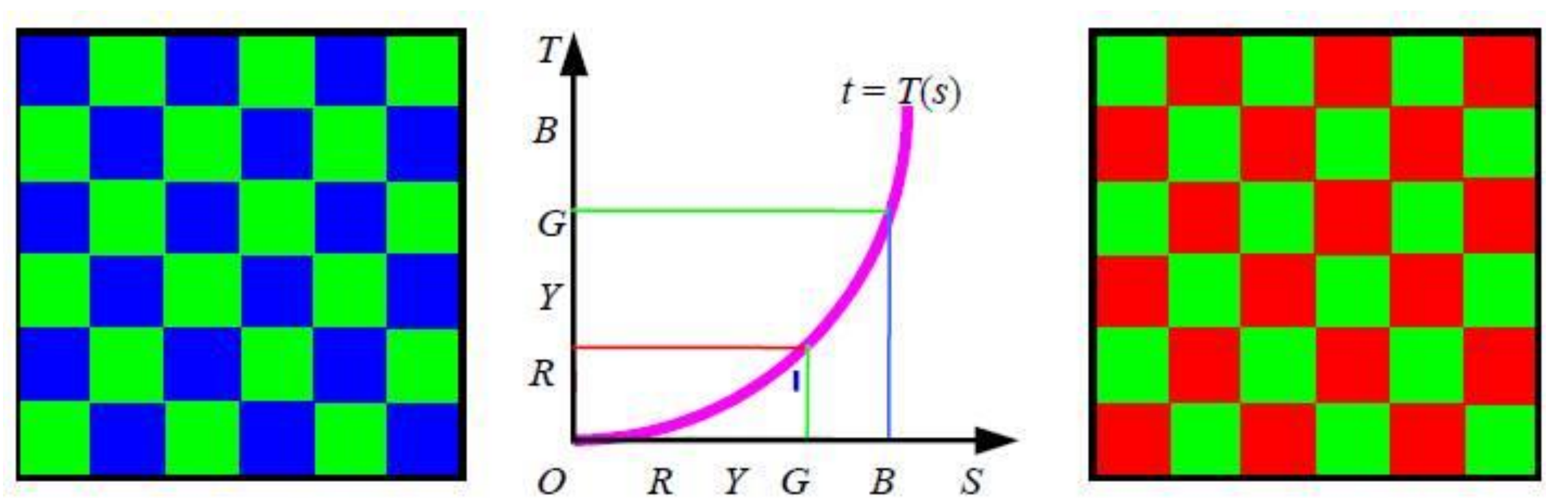


图 4.2.1 灰度映射原理

4.2.2 灰度映射示例

灰度映射技术的关键是根据增强要求设计映射函数

扩展低灰度范围
压缩高灰度范围

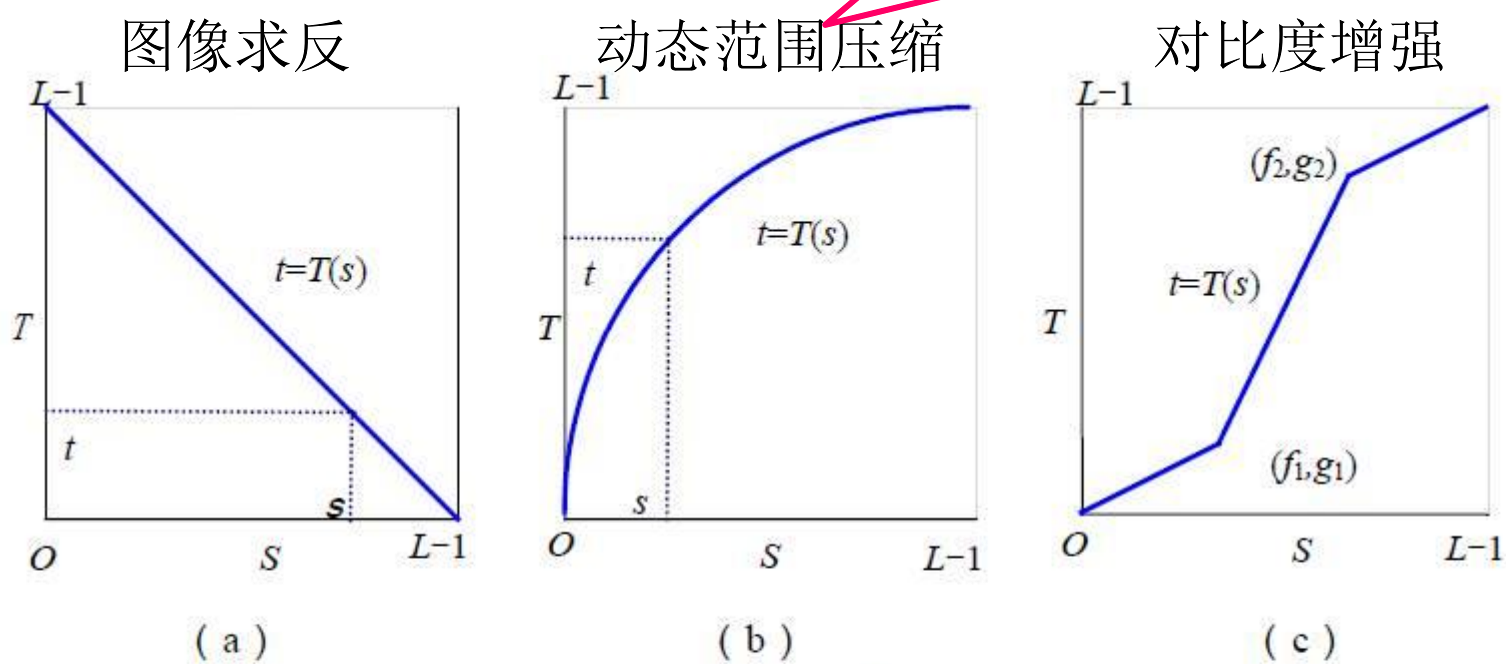
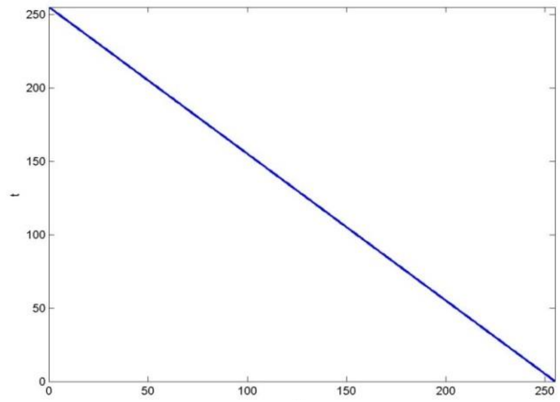


图 4.2.2 若干典型的灰度映射函数的示例

4.2.2 灰度映射示例

1. 图像求反

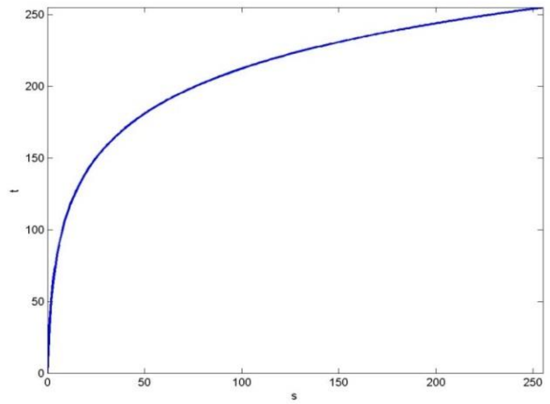


原图

图像求反

4.2.2　　灰度映射示例

2.　动态范围压缩

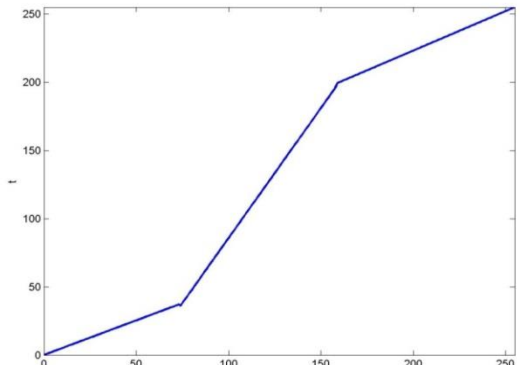


原图

动态范围压缩

4.2.2　　灰度映射示例

3.　对比度增强

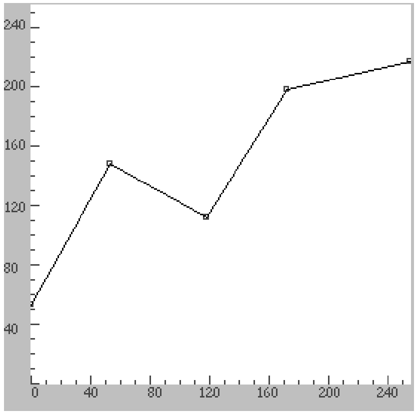


原图

对比度增强

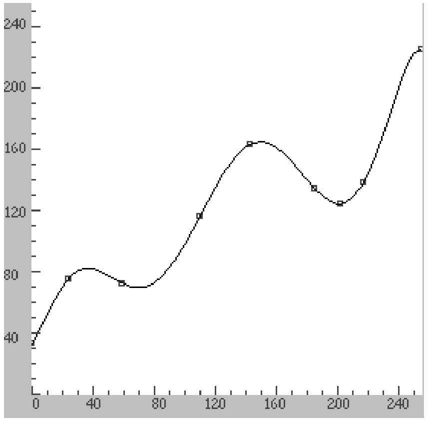
4.2.2　　灰度映射示例

4.　任意变换



4.2.2 灰度映射示例

4. 任意变换



4.2.2　　灰度映射示例

思考：如何判断是压缩还是扩展

图像求反　　　动态范围压缩　　　对比度增强

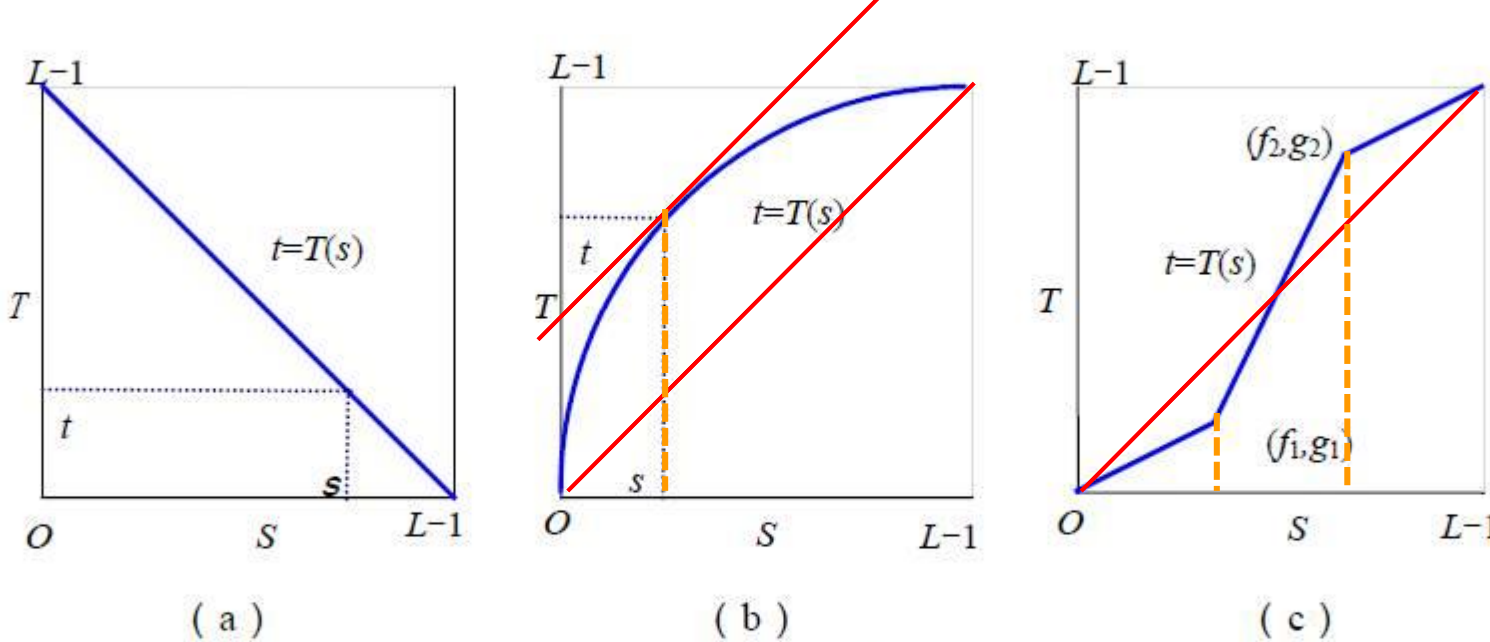


图 4.2.2　若干典型的灰度映射函数的示例

4.2.2　　灰度映射示例

思考：如何判断是压缩还是扩展

对输入图象灰度作线性扩张或压缩，映射函数为一个直线方程，其表达式如下：

$t = a \times s + b;$

其中：

a相当于变换直线的斜率，b相当于截距；

$$b = 0: \begin{cases} a > 1 \text{----对比度扩张} \\ a < 1 \text{----对比度压缩} \\ a = 1 \text{----相当于复制} \end{cases}$$

$b \neq 0$:　灰度偏置

作业

- 练习题：
  - 4.1

4.1 设给定平移量（2, 5），并用 2 和 5 作为放缩因子沿 X 和 Y 轴进行尺度变换，分别计算对图像点（2, 5）先平移变换后尺度变换和先尺度变换后平移变换所得到的结果，并进行比较和讨论。

  - 4.2

4.2 给出实现对一个像素先平移，再旋转，最后尺度变换的变换矩阵。

The end！