计算机视觉基础：第二章 图像采集

Fei Gao

gaofeihifly@163.com

https://fei-hdu.github.io/

杭州電子科技大學
HANGZHOU DIANZI UNIVERSITY

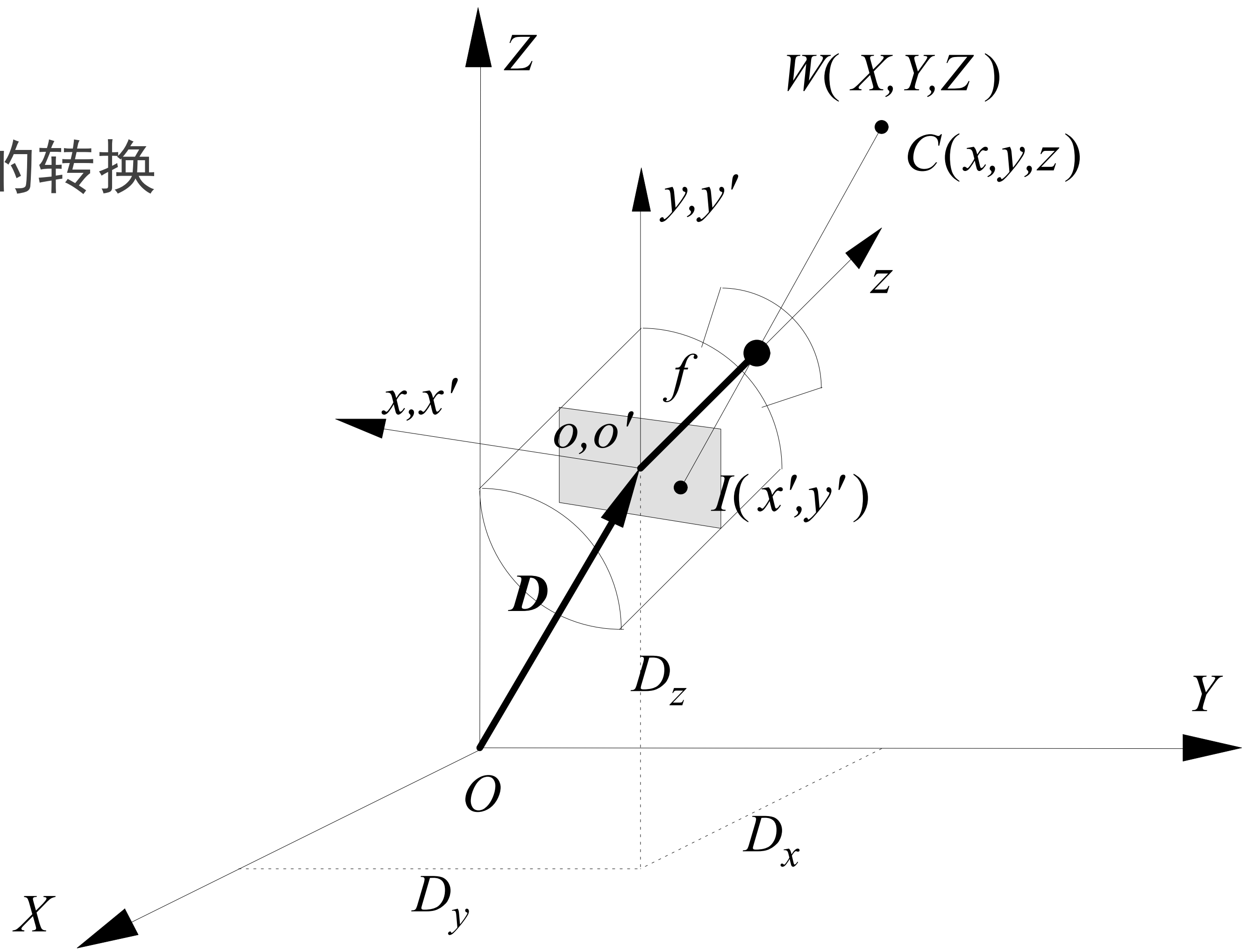
篤學力行 守正求新

第2章　图像采集

目录

2.1 几何成像模型

- 投影成像几何
  - 投影成像涉及在不同坐标系统之间的转换
  - 坐标系统（对3-D空间景物成像时）
    - (1) 世界坐标系统
    - (2) 摄像机坐标系统
    - (3) 像平面坐标系统
    - (4) 计算机图像坐标系统

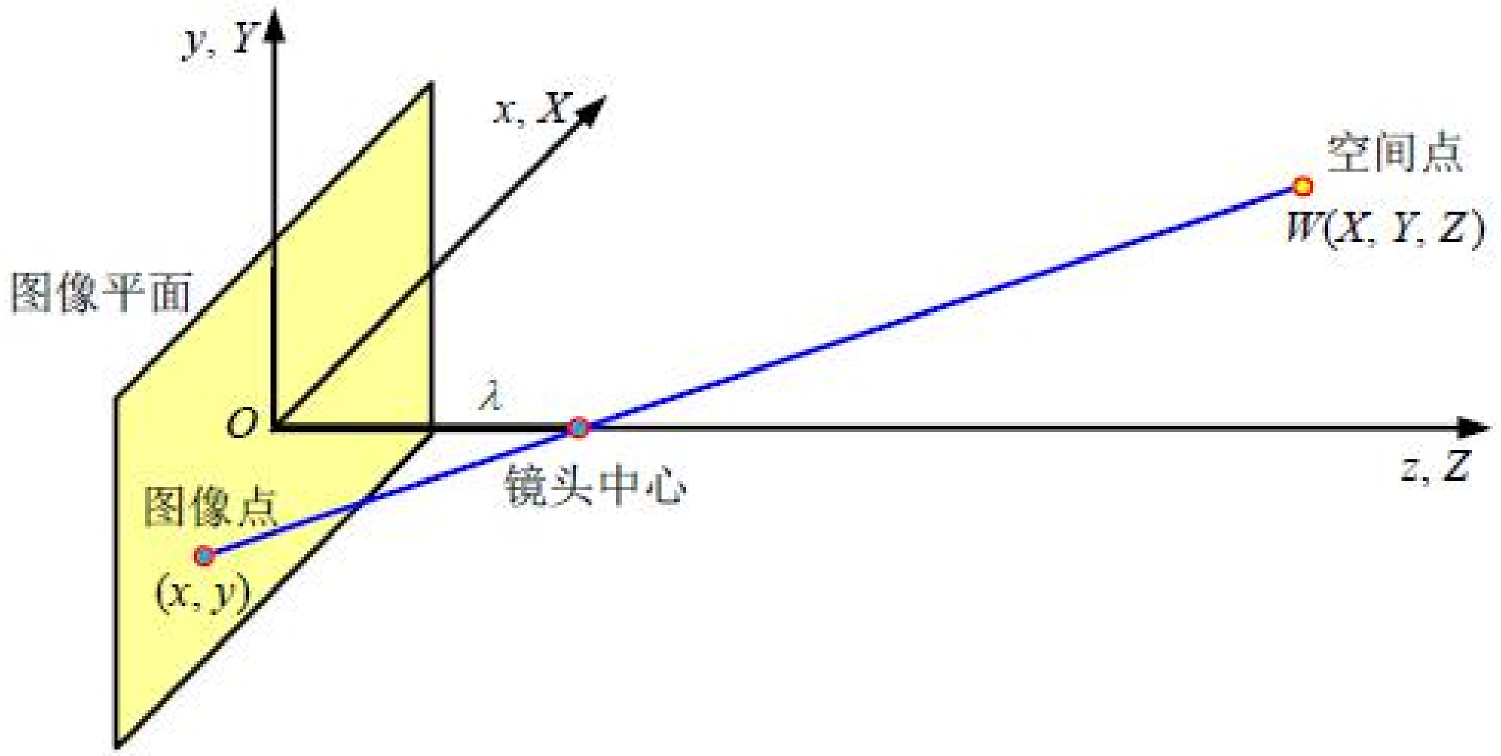


2.1 几何成像模型

- 投影成像几何
  - 齐次坐标
    - 将坐标系统用齐次坐标来表达，就可将坐标系统之间的转换表示成线性矩阵形式
    - 空间一个点的对应笛卡尔坐标XYZ的**齐次坐标**定义为(kX, kY, kZ, k)，其中k是一个任意的非零常数。
    - 将齐次坐标变回笛卡尔坐标可用第4个坐标量去除前3个坐标量得到

2.1 几何成像模型

- 基本成像模型
  - 摄像机坐标系统xyz中的图像平面与xy平面重合而光学轴（由镜头中心给出）沿z轴



2.1 几何成像模型

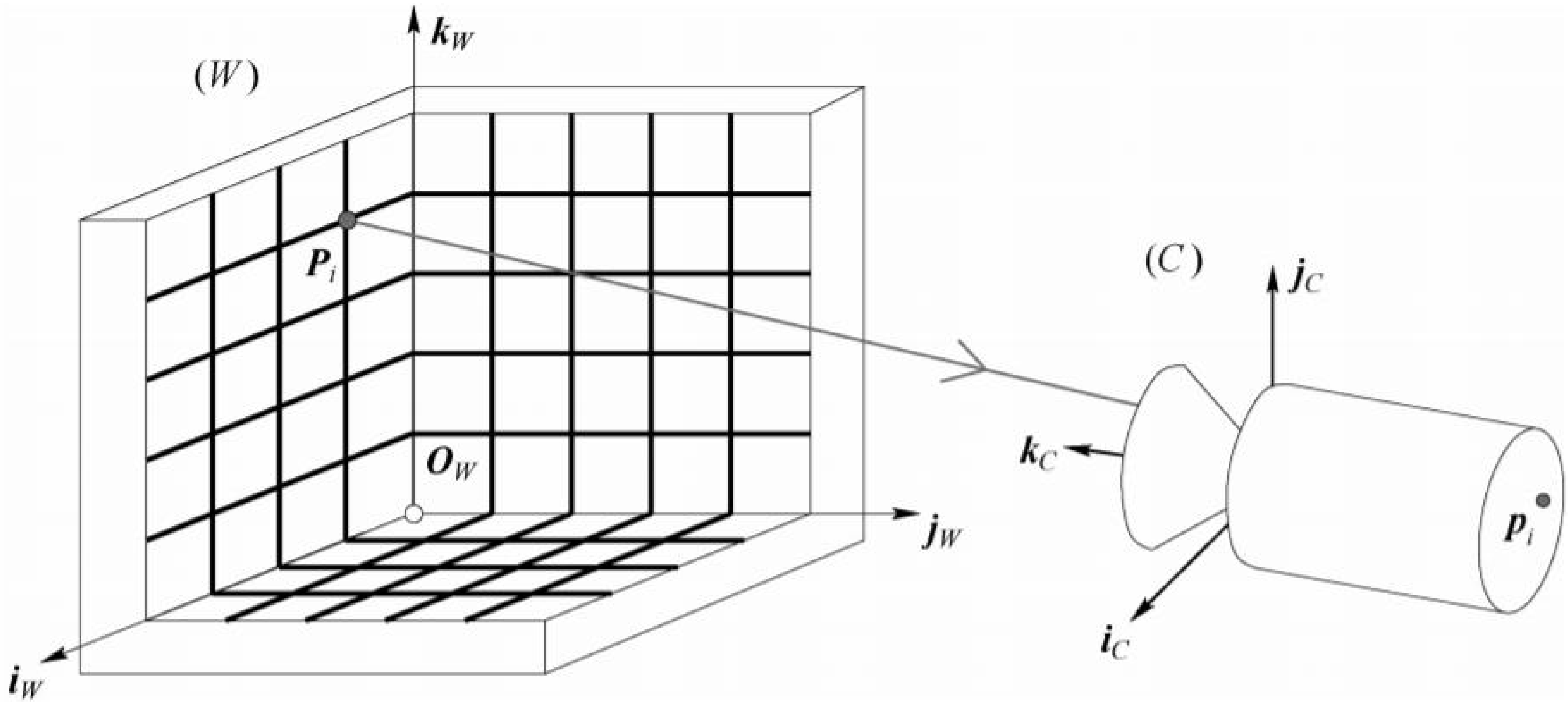


图 1.16　照相机的标定装置：这里采用三个相互垂直的棋盘格平面构成标定框架。可以采用其他样式的框架，也可以采用包括直线在内的其他几何图形

2.1 几何成像模型

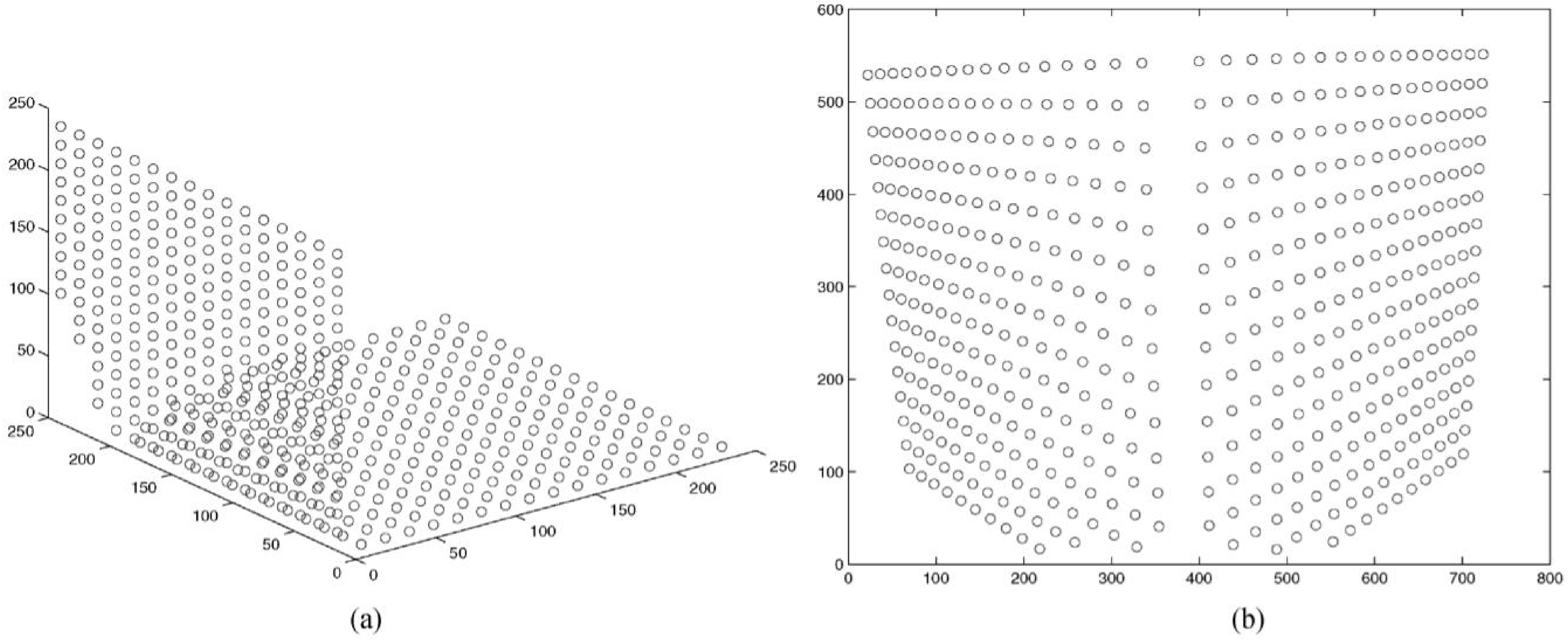
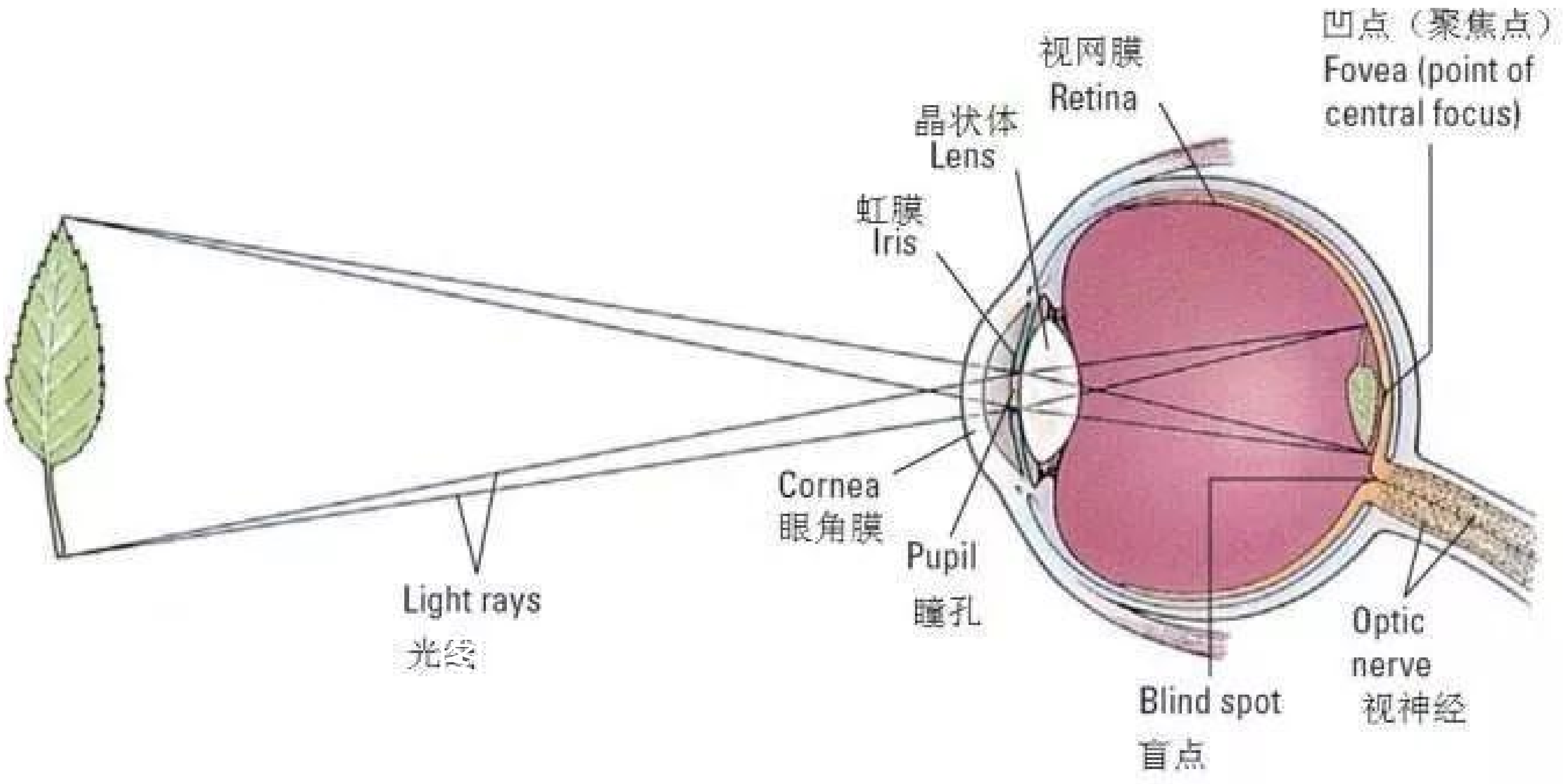


图 1.17　照相机标定数据。(a) 三维的 491 个基准点在标定网格的投影图；(b) 对应的成像点 (Data courtesy of Janne Heikkilä; data copyright © 2000 University of Oulu)

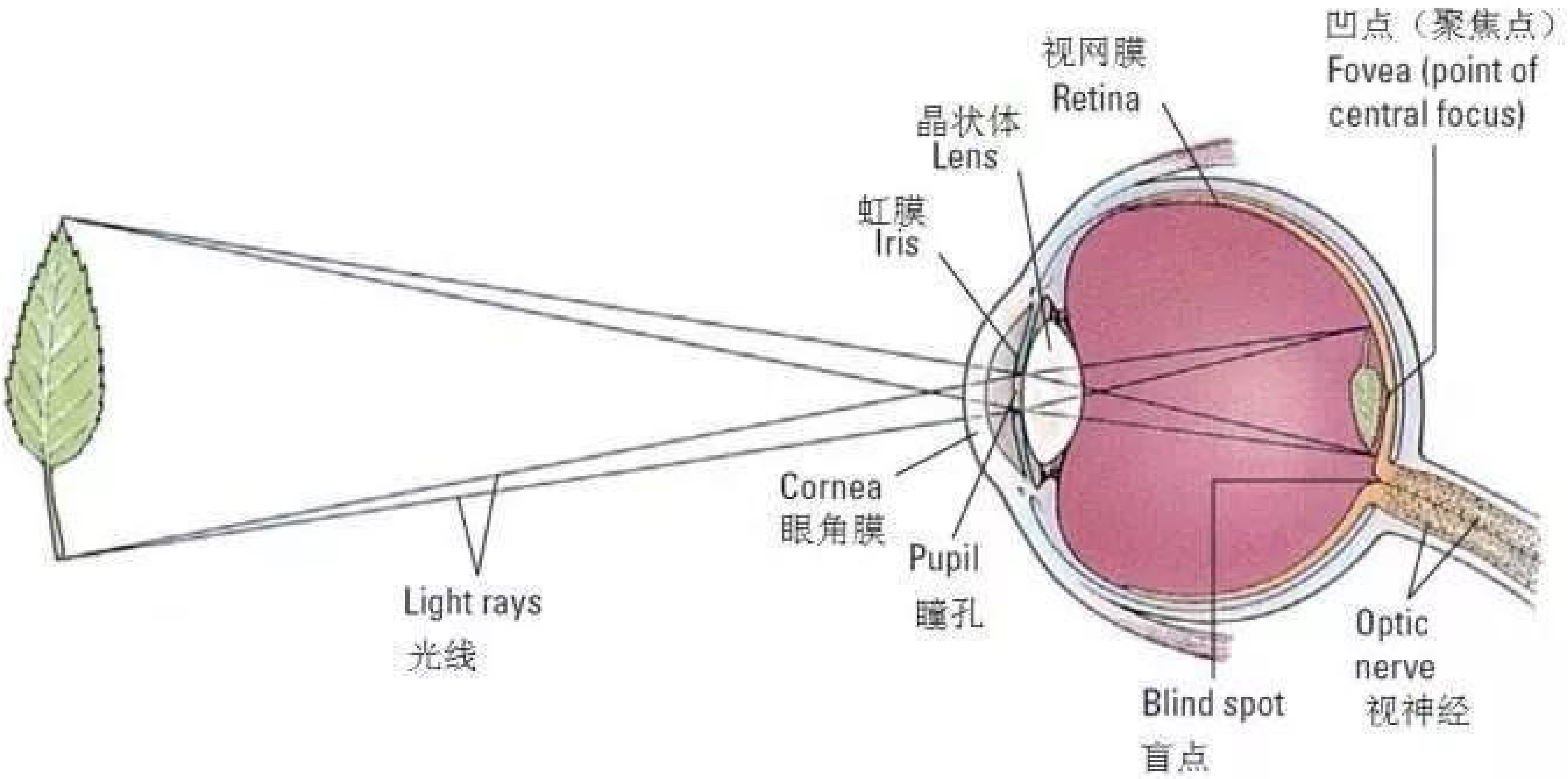
2.1 几何成像模型

- 世界是颠倒的？



2.1 几何成像模型

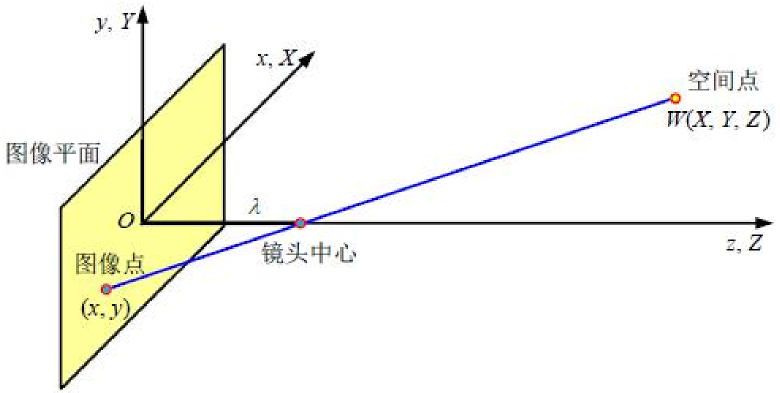
- 世界是颠倒的？



视网膜感受的颠倒信号，是在通过视神经传导到大脑皮层的视觉中枢后，在视觉中枢实现自动翻转的。

2.1 几何成像模型

- **基本模型**
  - **投影变换**
  - 3-D点投影后的图像平面坐标：



$$x = \frac{\lambda X}{\lambda - Z} \qquad y = \frac{\lambda Y}{\lambda - Z}$$

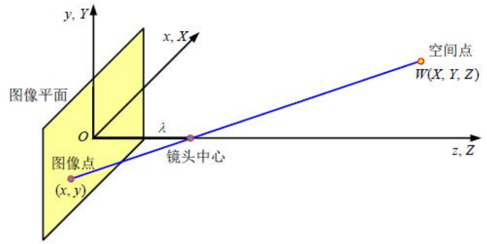
  - 投影变换矩阵

$$\boldsymbol{P} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1/\lambda & 1 \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = P \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{\lambda}{} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1/\lambda & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

2.1 几何成像模型

- **基本模型**



  - 齐次形式的摄像机坐标

$$\boldsymbol{c}_\mathrm{h}=\boldsymbol{P}\boldsymbol{W}_\mathrm{h}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&-1/\lambda&1\end{bmatrix}\begin{bmatrix}kX\\kY\\kZ\\k\end{bmatrix}=\begin{bmatrix}kX\\kY\\kZ\\-kZ/\lambda+k\end{bmatrix}$$

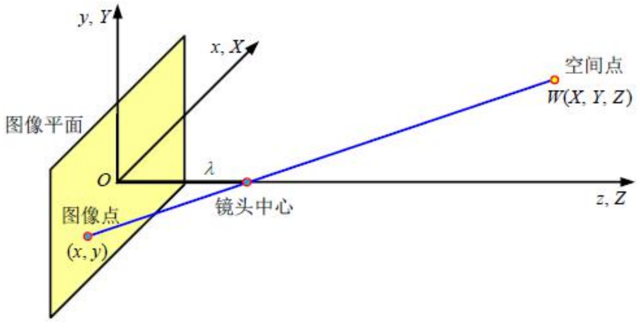
  - 用 c_h 的第4项去除前3项转换成笛卡儿形式

$$\boldsymbol{c}=\begin{bmatrix}x & y & z\end{bmatrix}^\mathrm{T}=\begin{bmatrix}\dfrac{\lambda X}{\lambda-Z} & \dfrac{\lambda Y}{\lambda-Z} & \dfrac{\lambda Z}{\lambda-Z}\end{bmatrix}^\mathrm{T}\qquad e.g.\frac{kX}{-kZ/\lambda+k}=\frac{X}{-Z/\lambda+1}=\frac{\lambda X}{-Z+\lambda}$$

2.1 几何成像模型

- 练习题
- 1.用一个带有50mm焦距镜头的照相机拍摄距离10m外，高2m的物体，该物体的成像尺寸是多少？如果换一个焦距为135mm的镜头，成像尺寸又为多少？

$$x = \frac{\lambda X}{\lambda - Z} \qquad y = \frac{\lambda Y}{\lambda - Z}$$



- 2.该出空间点（-2, -8, 10）经焦距为0.5的镜头投影变换成像后的摄像机坐标。

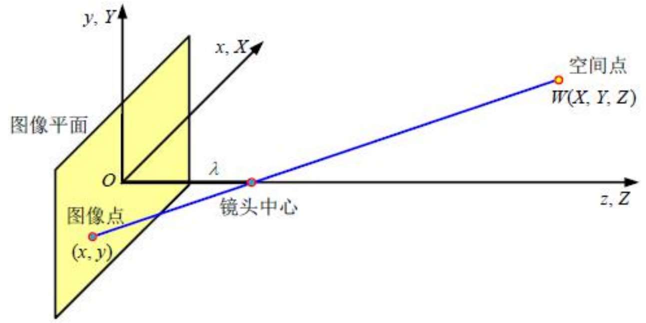
2.1 几何成像模型

- **练习题**
- 1.用一个带有50mm焦距镜头的照相机拍摄距离10m外，高2m的物体，该物体的成像尺寸是多少？如果换一个焦距为135mm的镜头，成像尺寸又为多少？

$$h = \frac{0.05H}{10-0.05} = \frac{0.1}{9.95} \approx 0.01$$

$$h' = \frac{0.135H}{10-0.135} = \frac{0.27}{9.865} \approx 0.03$$

$$x = \frac{\lambda X}{\lambda - Z} \qquad y = \frac{\lambda Y}{\lambda - Z}$$

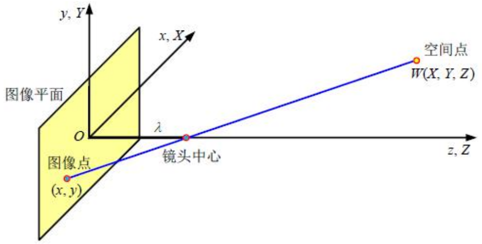


- 2.该出空间点（-2, -8, 10）经焦距为0.5的镜头投影变换成像后的摄像机坐标。

$$x = \frac{-2*0.5}{0.5-10} = \frac{-1}{-9.5} = \frac{2}{19} \approx 0.1$$

2.1 几何成像模型

- **基本模型**
  - **逆投影变换**
    - 根据2-D图像坐标来确定3-D客观景物的坐标



  - 逆投影变换矩阵

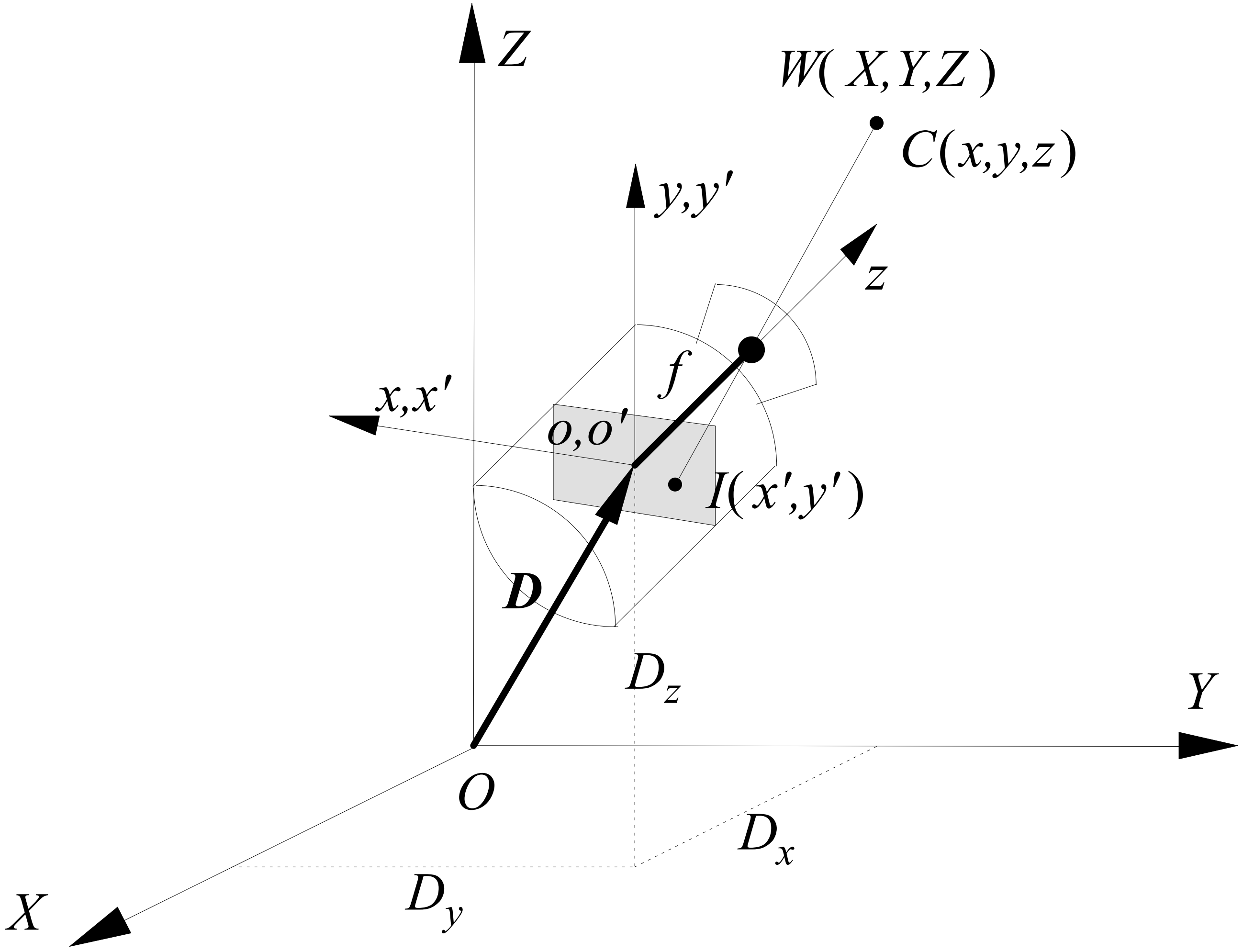
$$P^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \dfrac{1}{\lambda} & 1 \end{bmatrix}$$

  - 要利用逆投影变换将3-D空间点从其图像中恢复出来需要知道该点的至少一个世界坐标

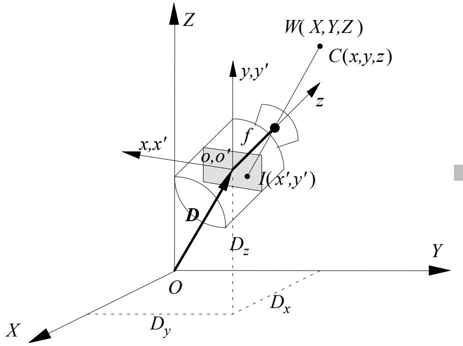
2.1 几何成像模型

• 一般成像模型

- 摄像机坐标系统与世界坐标系
- 但摄像机坐标系统与像平面坐

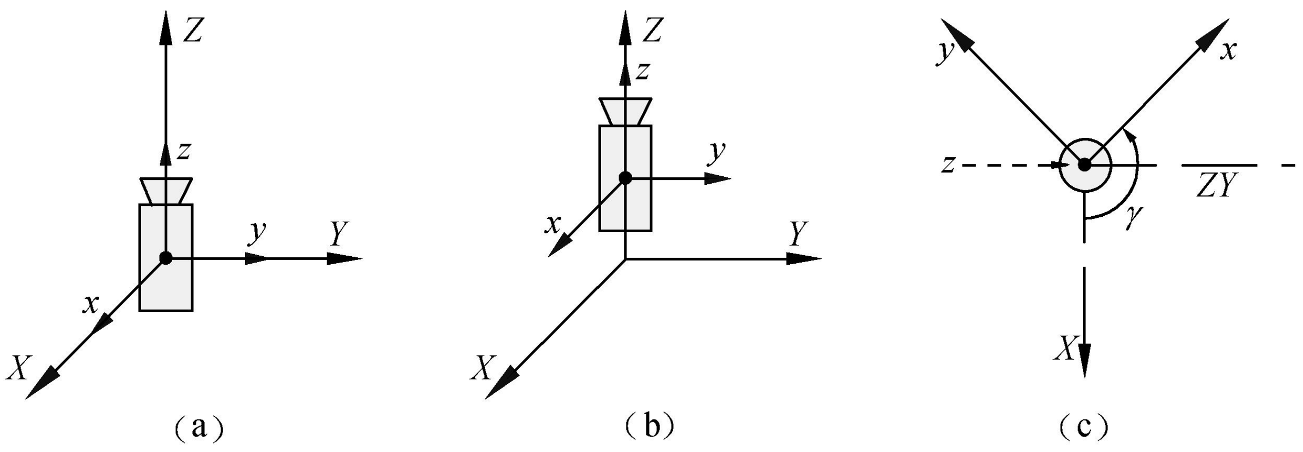


2.1 几何成像模型

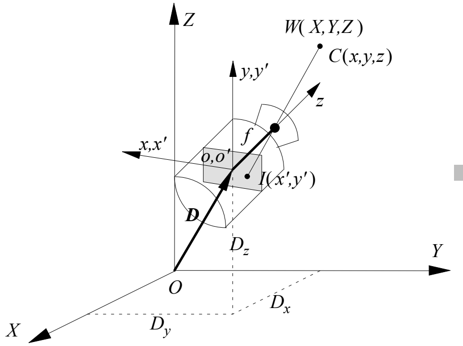


• 一般成像模型

- 上述模型可通过以下一系列步骤转换为世界坐标系统与摄像机坐标系统重合时的摄像机模型：
- ① 将像平面原点按矢量D移出世界坐标系统的原点；
- ② 以某个 γ 角（绕 z 轴）扫视 x 轴；
- ③ 以某个 α 角将 z 轴倾斜（绕 x 轴旋转）



2.1 几何成像模型



- **一般成像模型**
  - 平移矩阵

$$
\boldsymbol{T}=\begin{bmatrix}1 & 0 & 0 & -D_x\\ 0 & 1 & 0 & -D_y\\ 0 & 0 & 1 & -D_z\\ 0 & 0 & 0 & 1\end{bmatrix}
$$

  - 旋转矩阵

$$
\boldsymbol{R}=\boldsymbol{R}_\alpha\boldsymbol{R}_\gamma=\begin{bmatrix}\cos\gamma & \sin\gamma & 0 & 0\\ -\sin\gamma\cos\alpha & \cos\alpha\cos\gamma & \sin\alpha & 0\\ \sin\alpha\sin\gamma & -\sin\alpha\cos\gamma & \cos\alpha & 0\\ 0 & 0 & 0 & 1\end{bmatrix}
$$

2.2 亮度成像模型

- 光度学

  - 光源沿某个方向的亮度是用在该方向上的单位投影面积在单位立体角（其单位是球面度，sr）内发出的光通量来衡量，单位是cd/m2（坎[德拉]每平方米），其中cd是发光强度的单位，1cd = 1 lm/sr

  - 被光线照射的表面上的照度用照射在**单位面积上的光通量**来衡量，单位是 lx（勒[克斯]，也有用lux的），1 lx = 1 lm/m2

2.2 亮度成像模型

- 一个简单的图像亮度成像模型
  - 2-D亮度函数

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

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

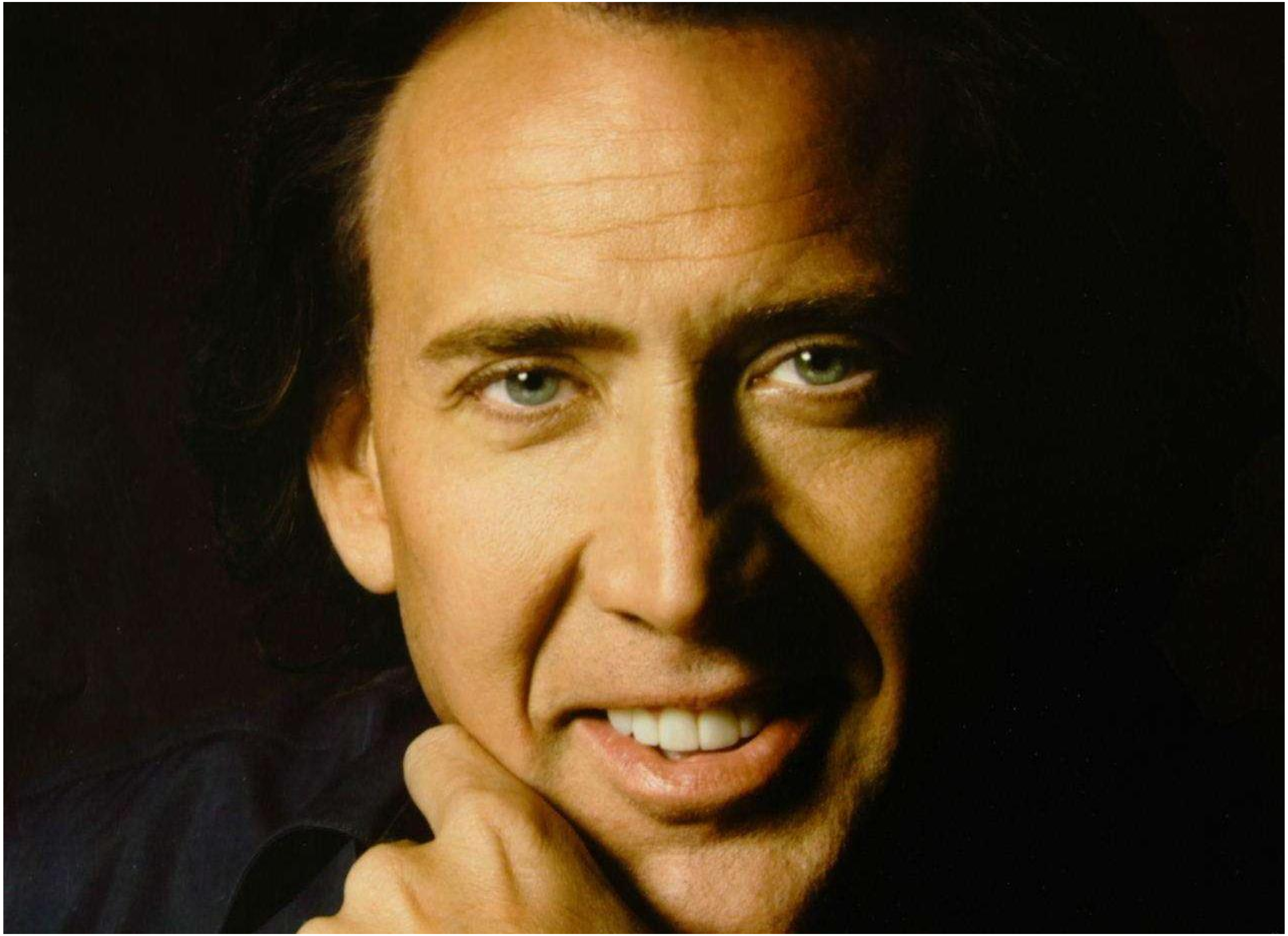
  - **照度**函数

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

  - **反射**函数

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

  - 单色图像 $f(\cdot)$在坐标 (x, y) 处的亮度值称做图像在该点的灰度值：$Gmin \le g \le Gmax$



2.2 亮度成像模型

- 练习题

- 3.如果办公室工作所需的照度为100-1000 lx，设墙面的反射率为0.8，那么在这样的办公室里，对墙面拍得的照片的亮度（l）范围是多少（只考虑数值）？如何将其线性地移到灰度值（g）范围[0, 255]之中？

2.3 采样和量化

- 空间分辨率
  - 图像的尺寸，在成像时采了MN个样

$$M = 2^m \qquad N = 2^n$$

- 幅度分辨率
  - 在成像时量化成了G 个灰度级 $G = 2^k$

- 存储一幅图像所需的位数b（单位是bit）

$$b = M \times N \times k$$

2.3 采样和量化

- 图像质量与采样

2.3 采样和量化

- 图像质量与采样

2.3 采样和量化

- 图像质量与量化

2.4 像素间联系

- 像素邻域
  - 4-邻域，记为$N_4(p)$
  - 对角邻域，记为$N_D(p)$
  - 8-邻域，记为$N_8(p)$

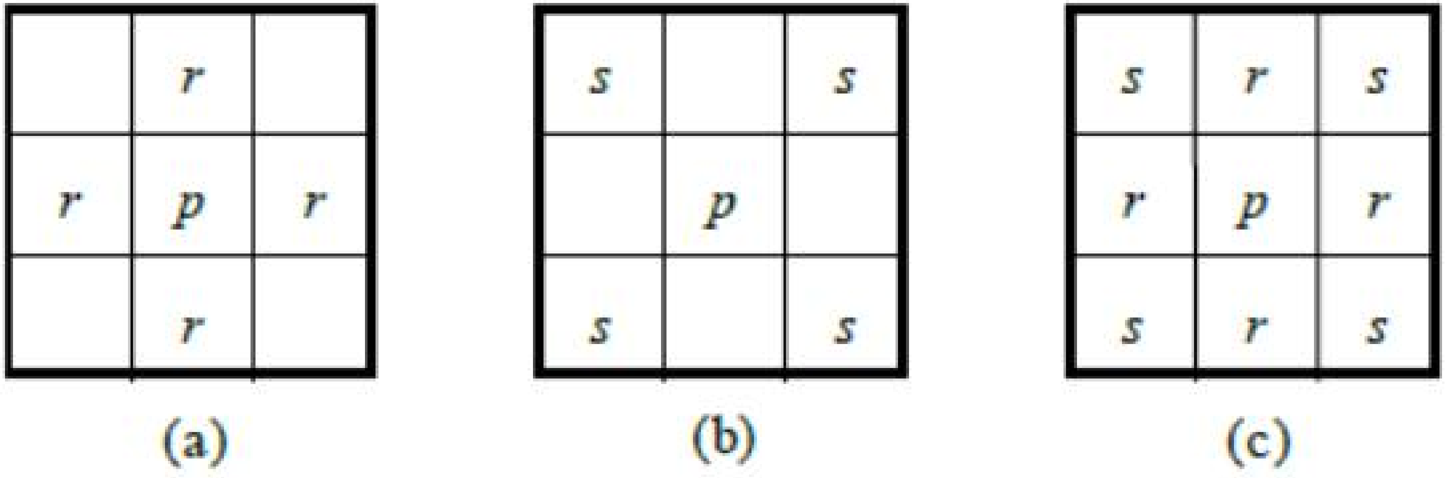
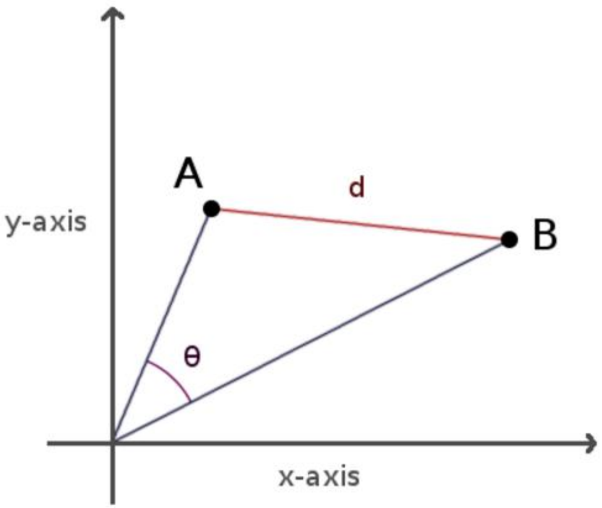


图 2.4.1 像素的邻域

2.4 像素间联系

- 像素间距离
  - 欧氏距离（也是范数为2的距离）

$$D_E(p,q)=[(x-s)^2+(y-t)^2]^{1/2}$$



  - 城区距离（也是范数为1的距离）

$$D_4(p,q)=|x-s|+|y-t|$$

		2		
	2	1	2	
2	1	0	1	2
	2	1	2	
		2		

  - 棋盘距离（也是范数为 ∞ 的距离）

$$D_8(p,q)=\max(|x-s|,|y-t|)$$

2	2	2	2	2
2	1	1	1	2
2	1	0	1	2
2	1	1	1	2
2	2	2	2	2

2.4 像素间联系

- 距离计算示例
  - 在图2.4.3中，两个像素p和q之间的DE距离为5（见图2.4.3(a)），D4距离为7（见图2.4.3(b)），D8距离为4（见图2.4.3(c)）

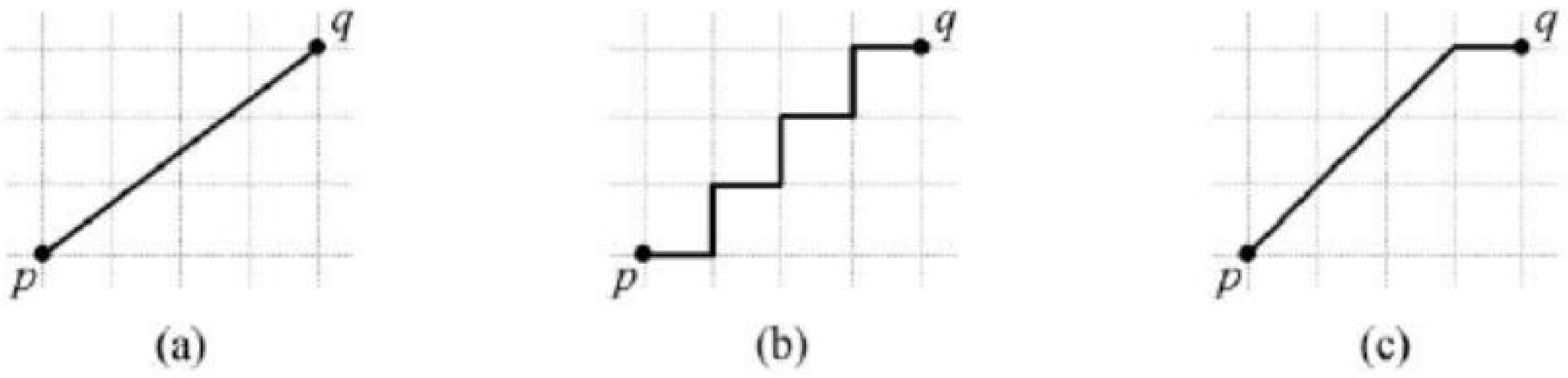
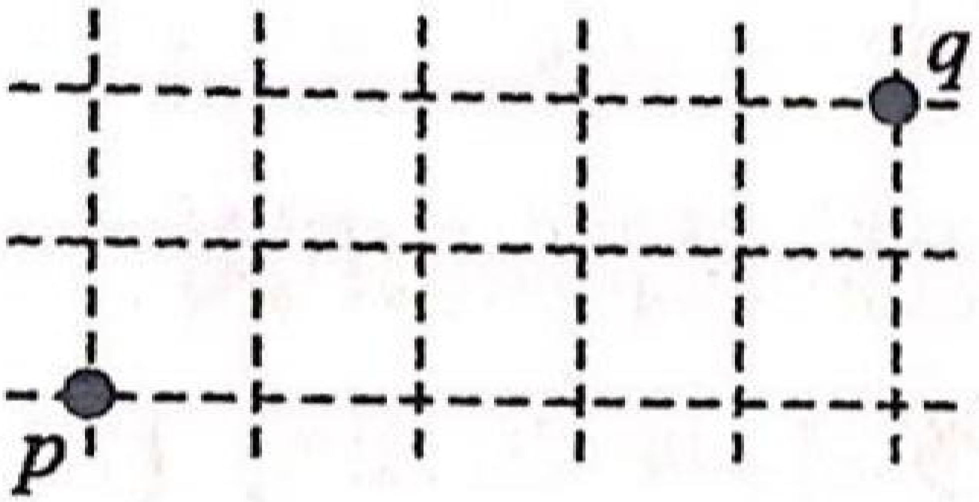


图 2.4.3 像素间距离的计算

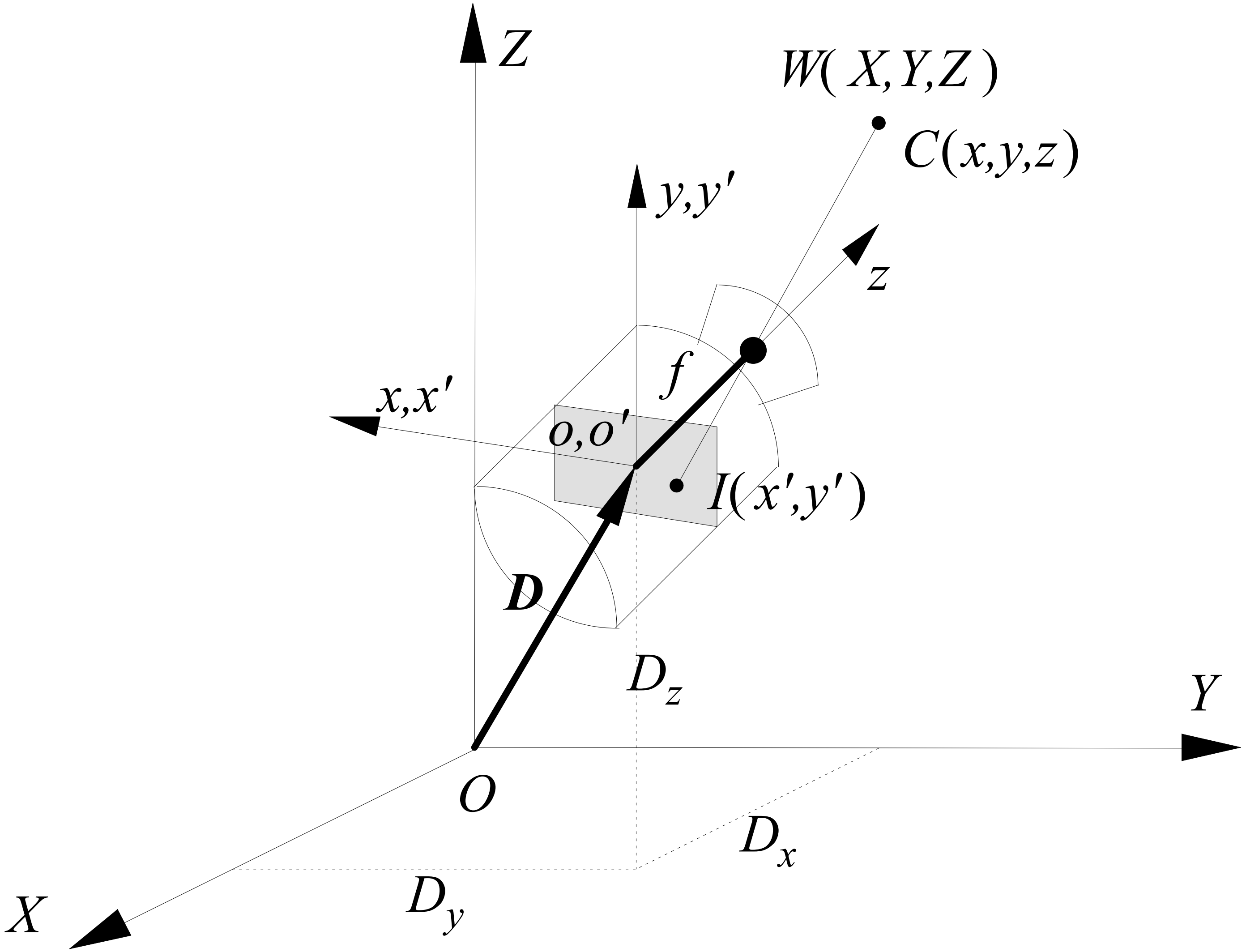
回顾：图像表示

- 5.计算如图所示的两个像素 p 和 q 之间的 DE 距离、D4 距离、D8 距离。



习题

- **练习题**
  - **成像模型？**
  - 坐标系统（对3-D空间景物成像时）
    - (1)　世界坐标系统
    - (2)　摄像机坐标系统
    - (3)　像平面坐标系统
    - (4)　计算机图像坐标系统



- # 练习题

*2.1 用一个带有 50 mm 焦距镜头的照相机拍摄距离 10 m 外，高 2 m 的物体，该物体的成像尺寸为多少？如果换一个焦距为 135 mm 的镜头，成像尺寸又为多少？

2.2 给出空间点(–2, –8, 10)经焦距为 0.5 的镜头投影变换成像后的摄像机坐标。

2.3 设一摄像机如图 2.1.3 所示安置。如果摄像机中心位置为(0, 0, 1)，摄像机的焦距为 0.135 m，扫视角为 135°，倾斜角为 135°，那么图中空间点 $W(1, 1, 0)$的像平面坐标是什么？

2.4 如果办公室工作所需的照度为 100~1 000 lx，设墙面的反射率为 0.8，那么在这样的办公室里，对墙面拍得的照片的亮度(l)范围是多少(只考虑数值)？如何将其线性地移到灰度值(g)范围[0, 255]之中？

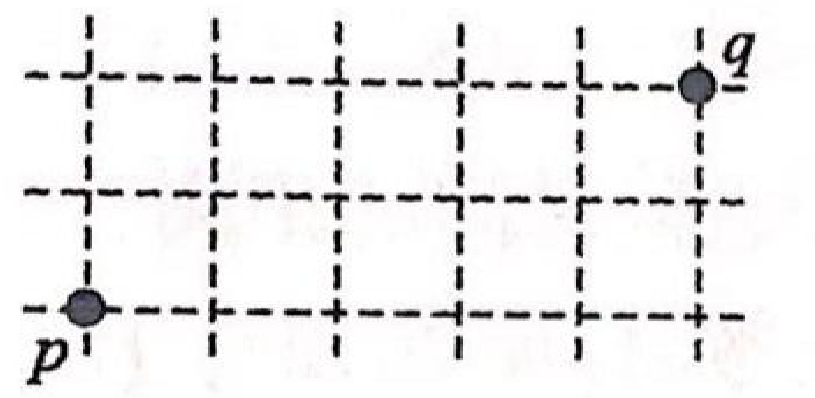
2.5 设图像的长宽比为 16 : 9。

（1）1000 万像素的手机上摄像机的空间分辨率约是多少？

（2）1800 万像素的相机的空间分辨率是多少？它拍的一幅彩色图像需多少个字节来存储？

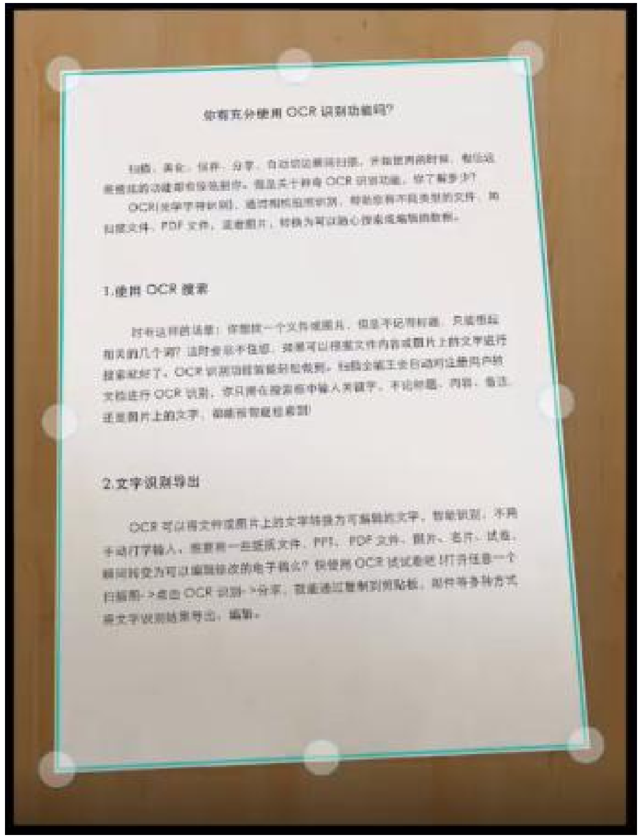
习题

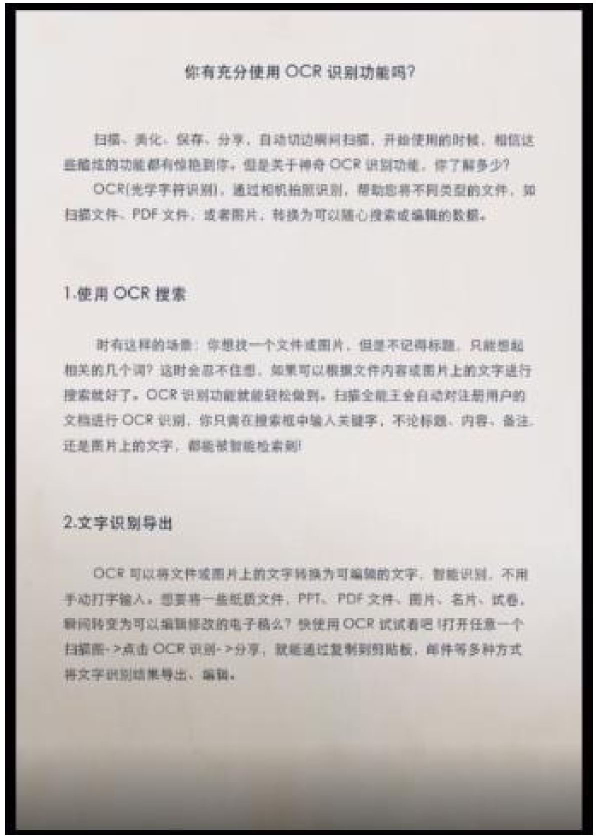
- 练习题
  - 计算如图所示的两个像素 p 和 q 之间的
    - D_E 距离
    - D_4 距离
    - D_8 距离

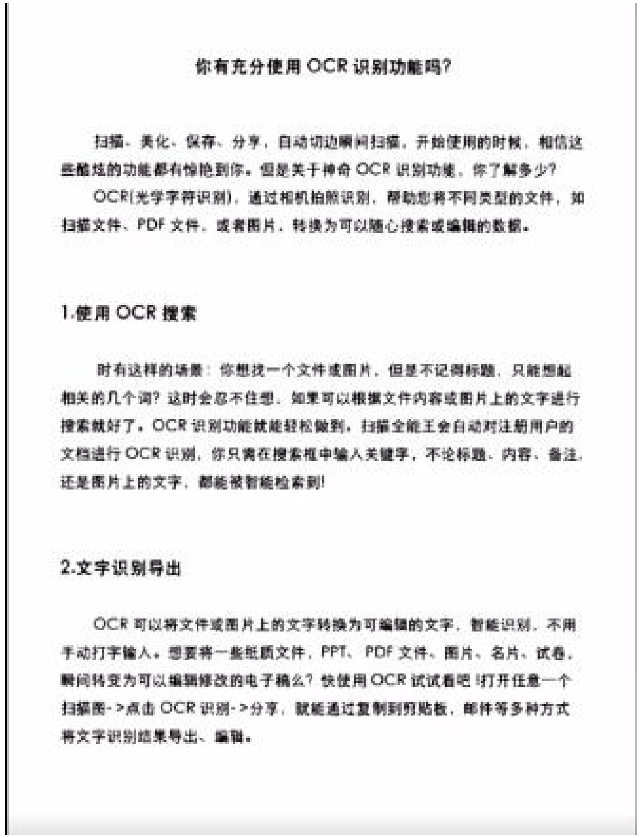


思考

- 如何实现以下功能？







你有充分使用 OCR 识别功能吗？

扫描、美化、保存、分享，自动切边瞬间扫描，开始使用的时候，相信这些酷炫的功能都有惊艳到你。但是关于神奇 OCR 识别功能，你了解多少？

OCR(光学字符识别)，通过相机拍照识别，帮助您将不同类型的文件，如扫描文件、PDF 文件，或者图片，转换为可以随心搜索或编辑的数据。

1.使用 OCR 搜索

时有这样的场景：你想找一个文件或图片，但是不记得标题，只能想起相关的几个词？这时会忍不住想，如果可以根据文件内容或图片上的文字进行搜索就好了。OCR 识别功能就能轻松做到。扫描全能王会自动对注册用户的文档进行 OCR 识别，你只需在搜索框中输入关键字，不论标题、内容、备注，还是图片上的文字，都能被智能检索到!

2.文字识别导出

OCR 可以将文件或图片上的文字转换为可编辑的文字，智能识别，不用手动打字输入。想要将一些纸质文件、PPT、PDF 文件、图片、名片、试卷，瞬间转变为可以编辑修改的电子稿么？快使用 OCR 试试看吧！打开任意一个扫描图->点击 OCR 识别->分享，就能通过复制到剪贴板，邮件等多种方式将文字识别结果导出、编辑。

第2章　图像采集

目录

2.5 图像坐标变换

• 基本坐标变换

- 坐标变换可借助矩阵写为：

平移变换　放缩变换　旋转变换

拉伸变换　剪切变换

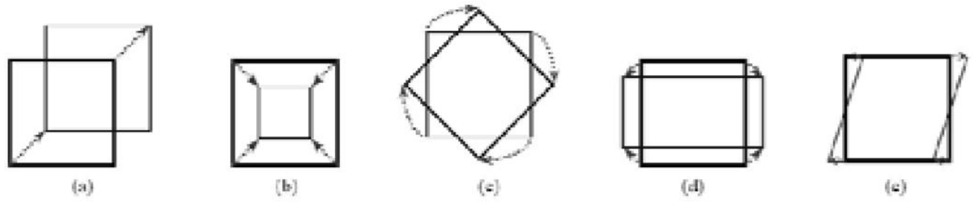


图 2.1.2　五种典型的坐标变换示意

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

$$\boldsymbol{v} = \begin{bmatrix} x & y & 1 \end{bmatrix}^{\mathrm{T}} \qquad \boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \qquad \boldsymbol{v}' = \begin{bmatrix} x' & y' & 1 \end{bmatrix}^{\mathrm{T}}$$

2.5 图像坐标变换

- 基本坐标变换
  - 平移变换矩阵

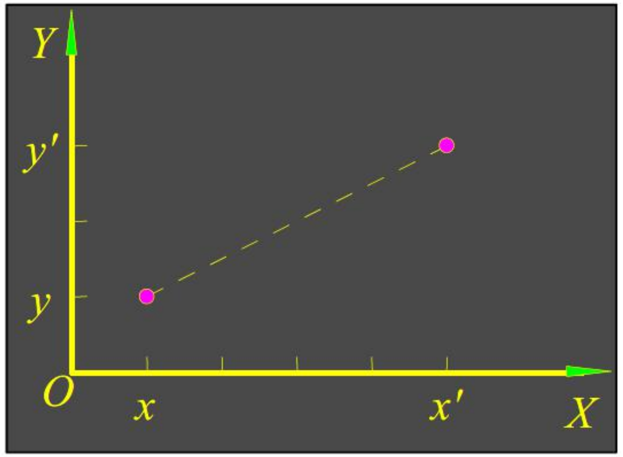
$$T=\begin{bmatrix}1 & 0 & t_x\\ 0 & 1 & t_y\\ 0 & 0 & 1\end{bmatrix}$$

平移变换的逆矩阵

$$T^{-1}=\begin{bmatrix}1 & 0 & -t_x\\ 0 & 1 & -t_y\\ 0 & 0 & 1\end{bmatrix}$$

$$T=\begin{bmatrix}1 & 0 & T_x\\ 0 & 1 & T_y\\ 0 & 0 & 1\end{bmatrix}$$

$$\begin{bmatrix}1 & 0 & 4\\ 0 & 1 & 2\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}1\\ 1\\ 1\end{bmatrix}=\begin{bmatrix}5\\ 3\\ 1\end{bmatrix}$$



2.5 图像坐标变换

- 基本坐标变换
  - 平移变换矩阵

2.5 图像坐标变换

- 基本坐标变换
  - 尺度变换矩阵

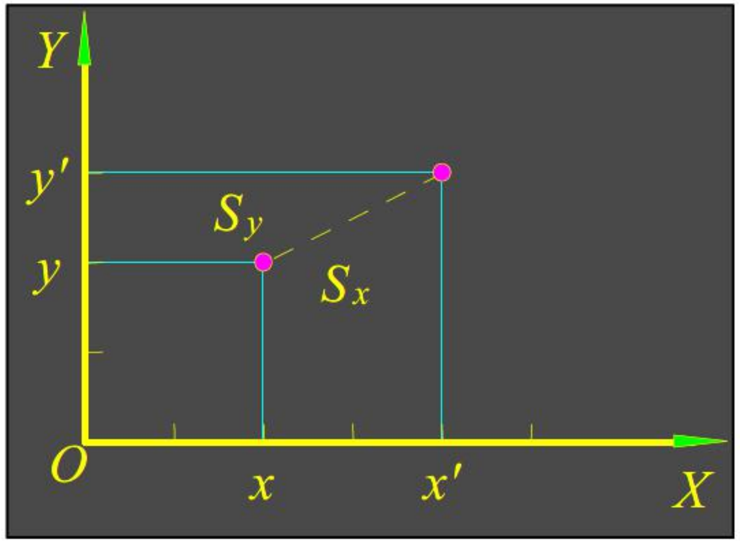
$$S=\begin{bmatrix} s_x & 0 & t_x \\ 0 & s_y & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

尺度变换的逆矩阵

$$S^{-1}=\begin{bmatrix} 1/s_x & 0 & t_x \\ 0 & 1/s_y & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$S=\begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 0 & 0 \\ 0 & 1.5 & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix}=\begin{bmatrix} 4 \\ 3 \\ 1 \end{bmatrix}$$



2.5 图像坐标变换

- 尺度变换（放缩变换）

$$\boldsymbol{S}=\begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}=\begin{bmatrix} 1/2 & 0 & 0 \\ 0 & 1/2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

• 图像镜像

- 镜像变换又分为水平镜像和竖直镜像。
  - 水平镜像即将图像左半部分和右半部分以图像竖直中轴线为中心轴进行对换；
  - 竖直镜像则是将图像上半部分和下半部分以图像水平中轴线为中心轴进行对换.

2.5 图像坐标变换

- 图像镜像

- 水平镜像的变换关系为：

$$[x_1 \quad y_1 \quad 1]=[x_0 \quad y_0 \quad 1]\begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ Width & 0 & 1 \end{pmatrix}=[Width-x_0 \quad y_0 \quad 1]$$

对矩阵求逆得到：$[x_0 \quad y_0 \quad 1]=[x_1 \quad y_1 \quad 1]\begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ Width & 0 & 1 \end{pmatrix}=[Width-x_1 \quad y_1 \quad 1]$

- 竖直镜像变换关系可形式化地描述为：

$$[x_1 \quad y_1 \quad 1]=[x_0 \quad y_0 \quad 1]\begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & Height & 1 \end{pmatrix}=[x_0 \quad Height-y_0 \quad 1]$$

逆运算为：$[x_0 \quad y_0 \quad 1]=[x_1 \quad y_1 \quad 1]\begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & Height & 1 \end{pmatrix}=[x_1 \quad Height-y_1 \quad 1]$

2.5 图像坐标变换

- **转置变换**
  - 图像转置是指将图像像素的x坐标和y坐标互换，图像的大小会随之改变：高度和宽度将互换

2.5 图像坐标变换

- **转置变换**
  - 图像转置是指将图像像素的x坐标和y坐标互换，图像的大小会随之改变：高度和宽度将互换

$$[x_1 \quad y_1 \quad 1]=[x_0 \quad y_0 \quad 1]\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}=[y_0 \quad x_0 \quad 1]$$

显然，转置矩阵 $\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$ 的逆矩阵仍为其自身。

故转置变换的逆变换具有相同的形式。

2.5 图像坐标变换

- 基本坐标变换
  - 旋转变换矩阵

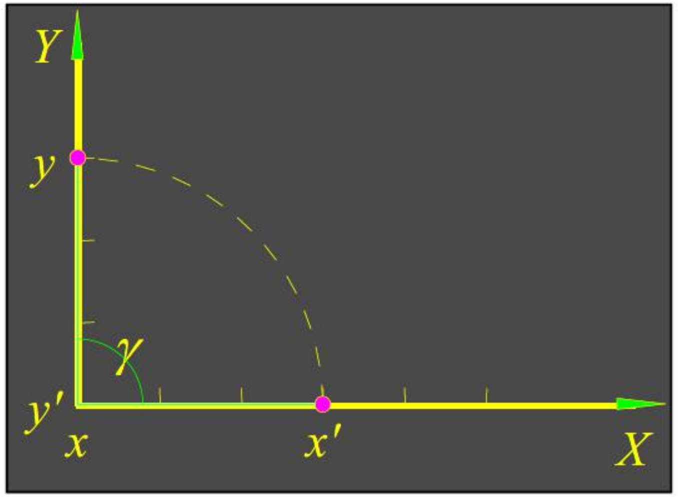
$$
\boldsymbol{R} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}
$$

旋转变换的逆矩阵

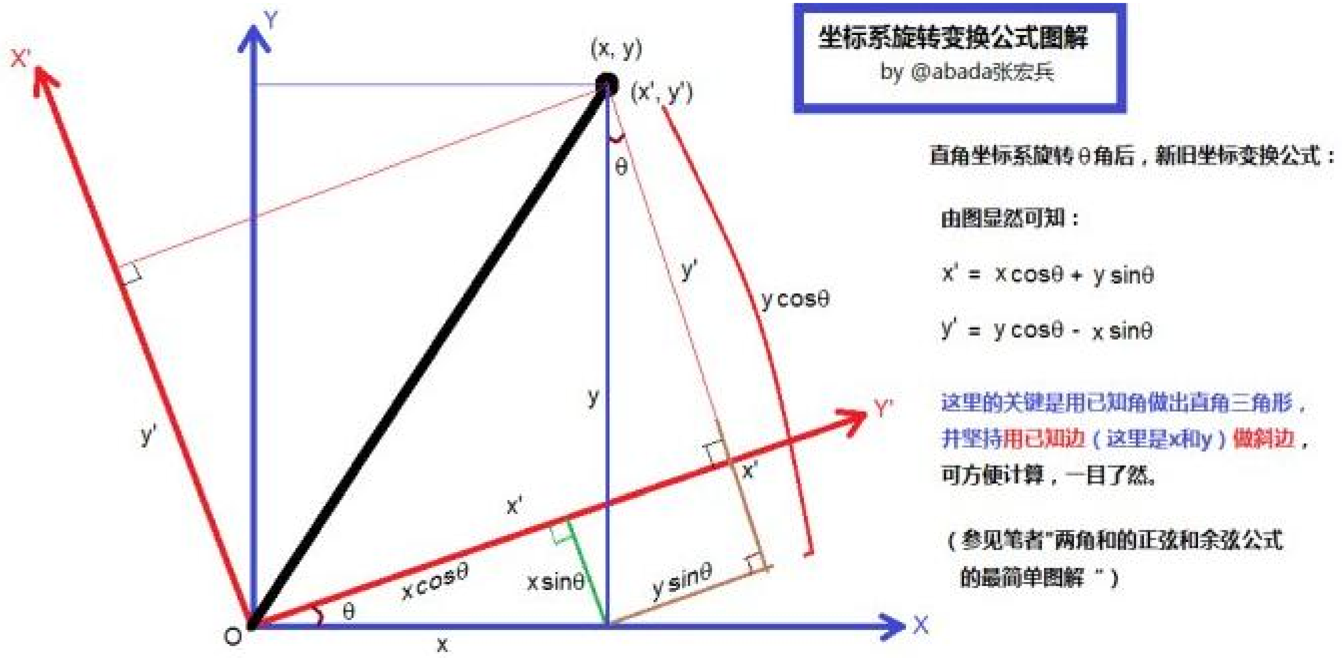
$$
\boldsymbol{R}^{-1} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}
$$

$$
\boldsymbol{R}_{\gamma} = \begin{bmatrix} \cos\gamma & \sin\gamma & 0 \\ -\sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}
$$

$$
\begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 \\ 0 \\ 1 \end{bmatrix}
$$



2.5 图像坐标变换


坐标系旋转变换公式图解
by @abada张宏兵
直角坐标系旋转θ角后，新旧坐标变换公式：
由图显然可知：
x' = x cosθ + y sinθ
y' = y cosθ - x sinθ
这里的关键是用已知角做出直角三角形，
并坚持用已知边（这里是x和y）做斜边，
可方便计算，一目了然。
（参见笔者"两角和的正弦和余弦公式
的最简单图解"）
X'
Y
(x, y)
(x', y')
θ
y'
y cosθ
y
Y'
x'
x'
x cosθ
x sinθ
y sinθ
O
θ
x
X

2.5 图像坐标变换

- 基本坐标变换：旋转变换

```
A = imread('lena.bmp');
B = imrotate(A,30,'nearest','crop');
```

$$\gamma = -45^\circ$$

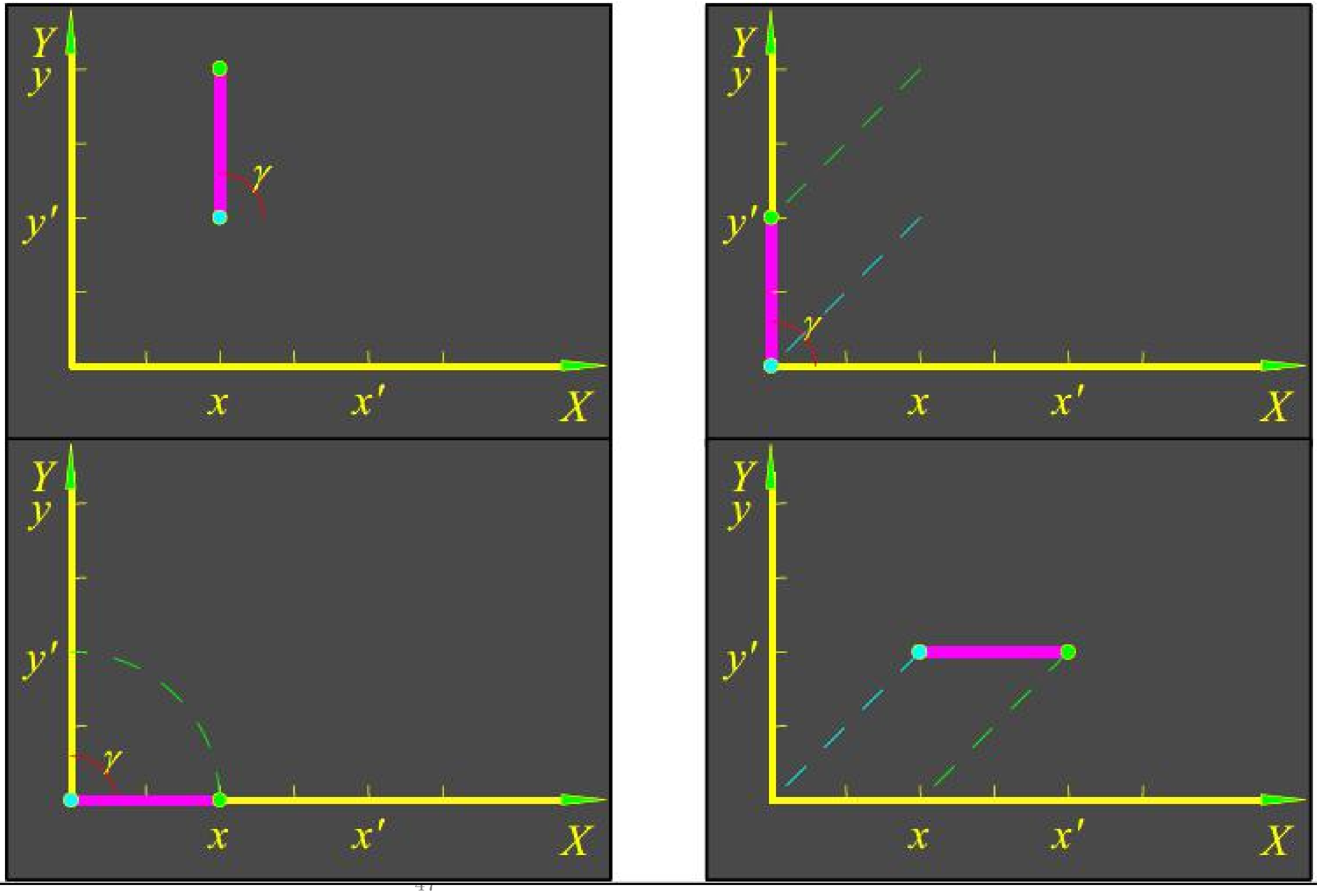
$$R = \begin{bmatrix} \sqrt{2}/2 & -\sqrt{2}/2 & 0 \\ \sqrt{2}/2 & \sqrt{2}/2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

2.5 图像坐标变换

- 基本坐标变换：旋转变换

旋转轴不在原点

绕非原点旋转
=
平移至原点
+
绕原点旋转
+
平移回去



- 基本坐标变换：**变换级联**

对一个坐标为v的点的平移、放缩、绕Z轴旋转变换（级联起来）可表示为：

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

等价于用单个变换矩阵A对点v进行变换

这些矩阵的运算次序一般不可互换

2.5 图像坐标变换

- 变换级联

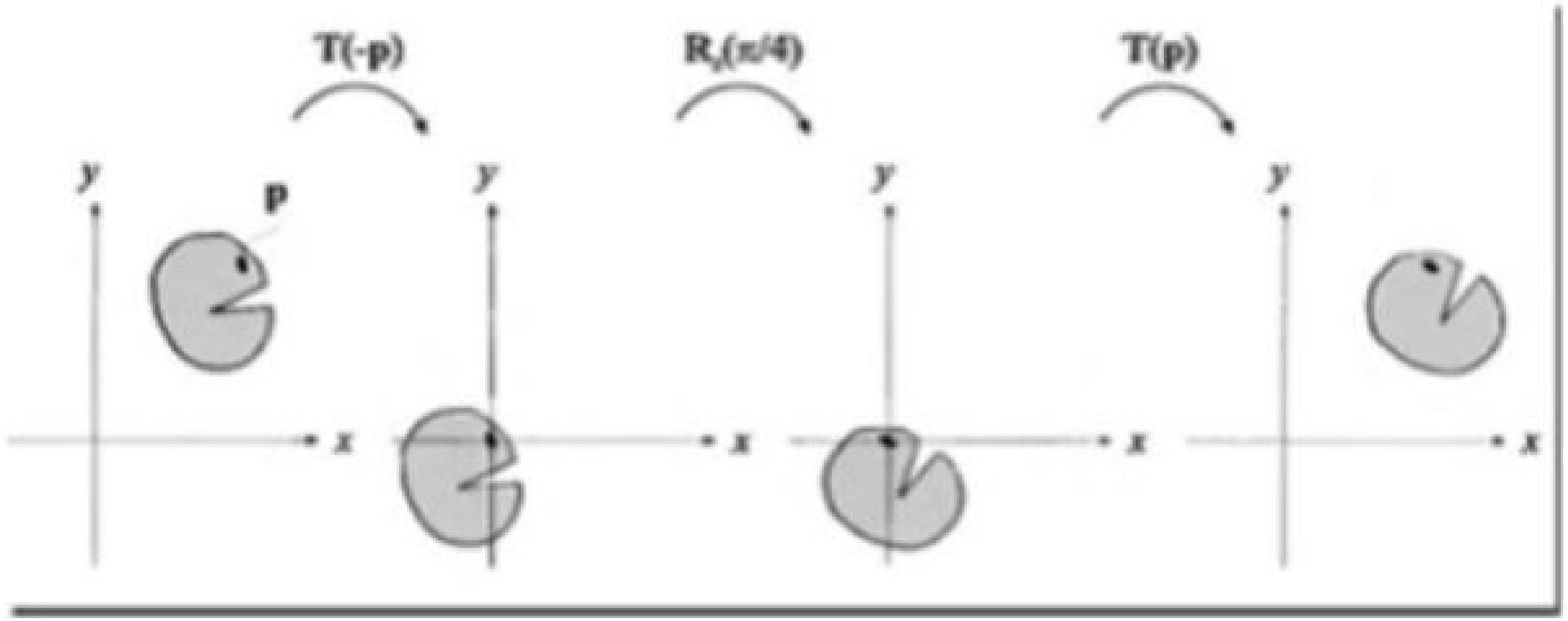


图4.2 绕指定点P旋转的例子。

2.5 图像坐标变换

- 基本坐标变换：变换级联

$$T=\begin{bmatrix}1&0&2\\0&1&4\\0&0&1\end{bmatrix}$$

$$S=\begin{bmatrix}3&0&0\\0&2&0\\0&0&1\end{bmatrix}$$

$$v=[1\quad 2\quad 1]^{\mathrm{T}}$$

$$v'_{ST}=[9\quad 12\quad 1]^{\mathrm{T}}$$

$$v'_{TS}=[5\quad 8\quad 1]^{\mathrm{T}}$$

先平移后放缩 $\neq$ 先放缩后平移

- ## 仿射变换
  - 一个平面上的仿射变换有6个自由度

$$
\boldsymbol{A}=\begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}
$$

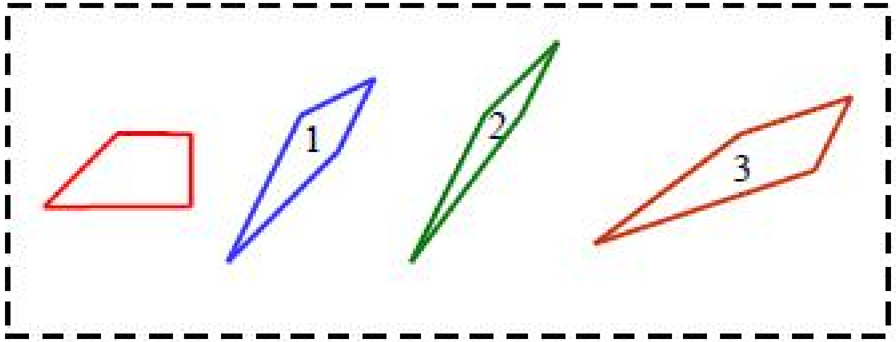


图 2.5.1　对一个多边形图形分别进行三次仿射变换得到的结果

2.5 图像坐标变换

- 仿射变换
  - 仿射变换的一种特例是欧氏变换

$$E=\begin{bmatrix}\cos\theta & \sin\theta & t_x\\ -\sin\theta & \cos\theta & t_y\\ 0 & 0 & 1\end{bmatrix}$$

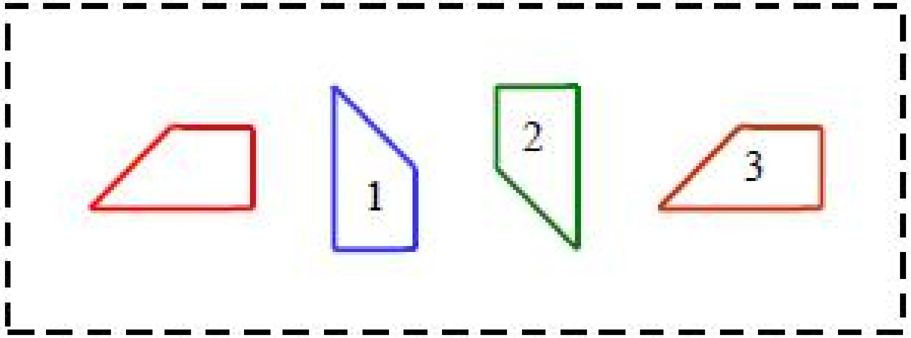


图 2.5.2　对一个多边形图形分别进行三次欧氏变换得到的结果

2.5 图像坐标变换

- 仿射变换
  - **相似变换**也是仿射变换的一种特例

$$
X = \begin{bmatrix} s\cos\theta & s\sin\theta & t_x \\ -s\sin\theta & s\cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix}
$$

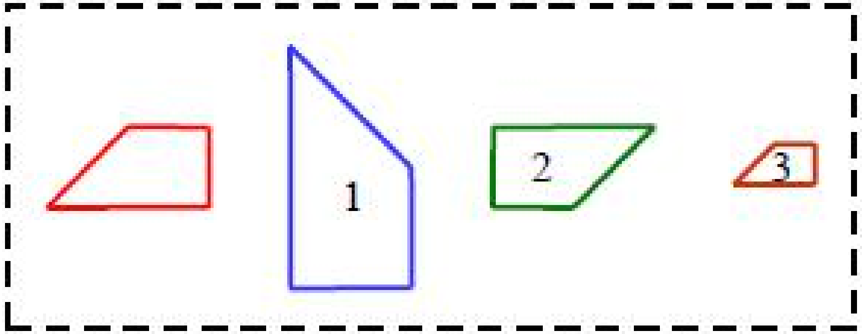
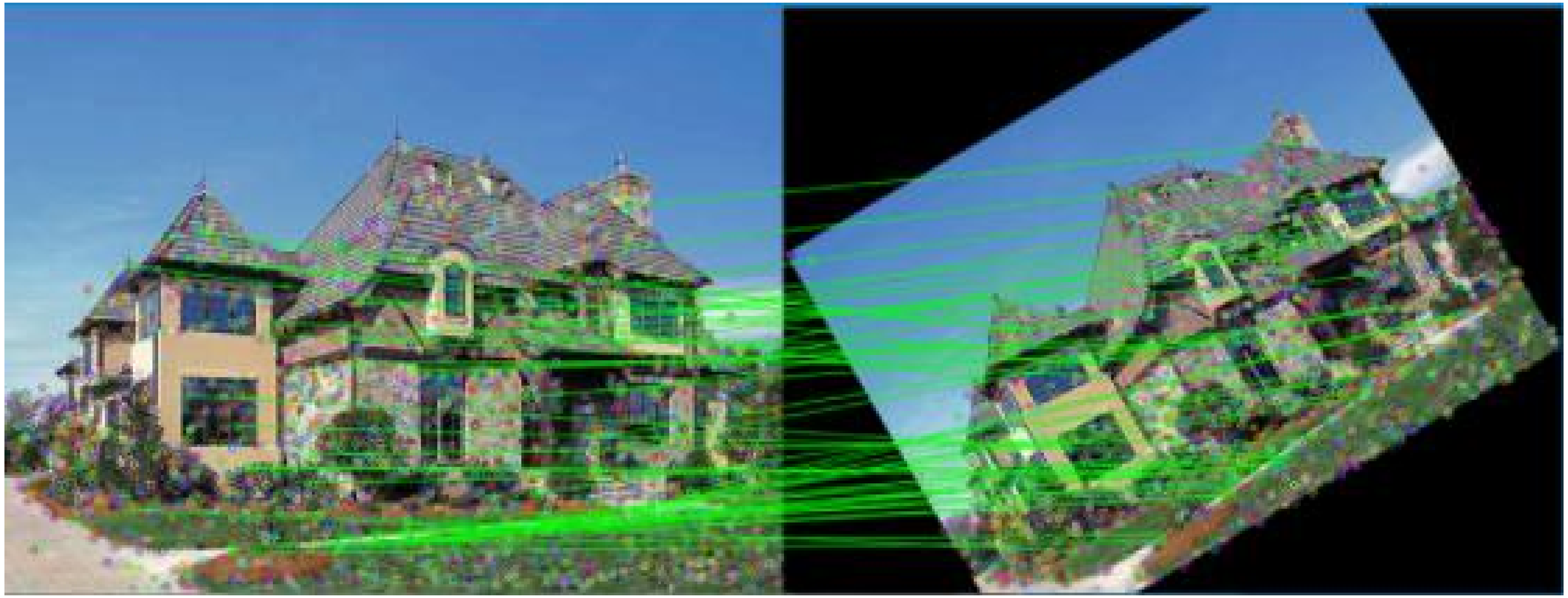


图 2.5.3　对一个多边形图形分别进行三次相似变换得到的结果

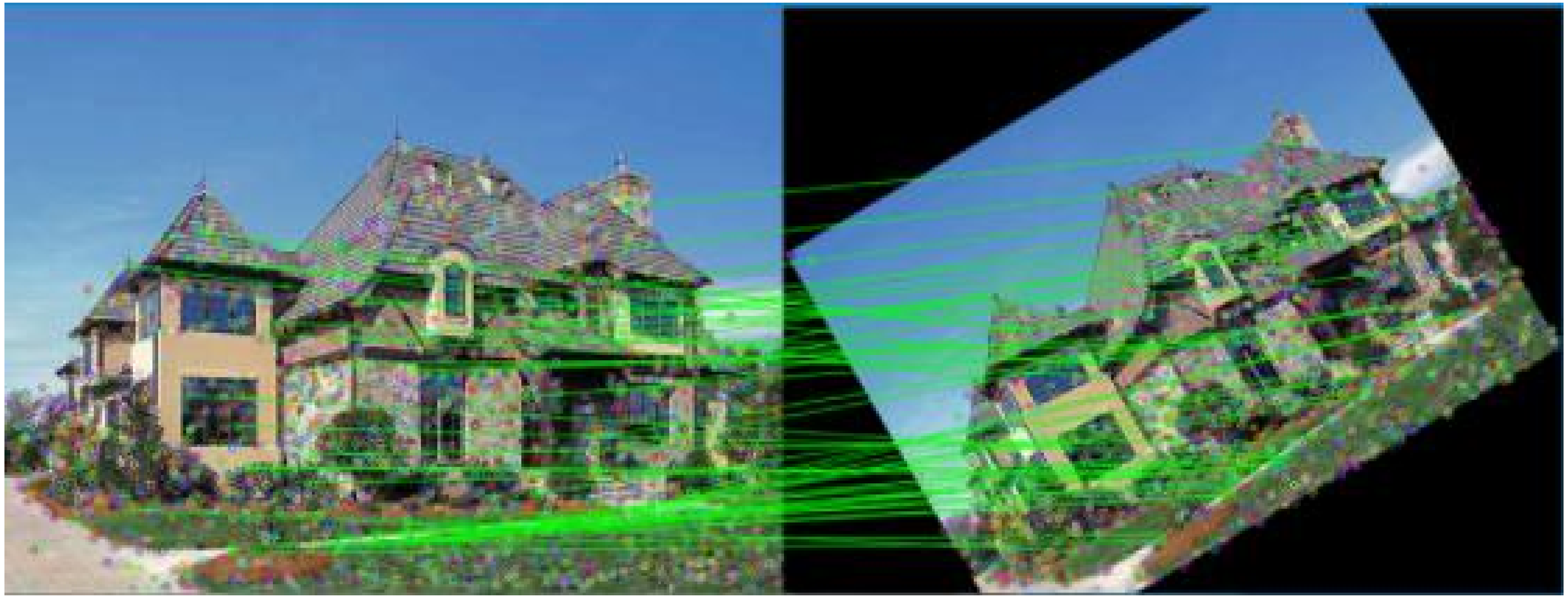
几何失真校正

- 两个步骤：
  - 计算空间变换函数
  - 插值填充



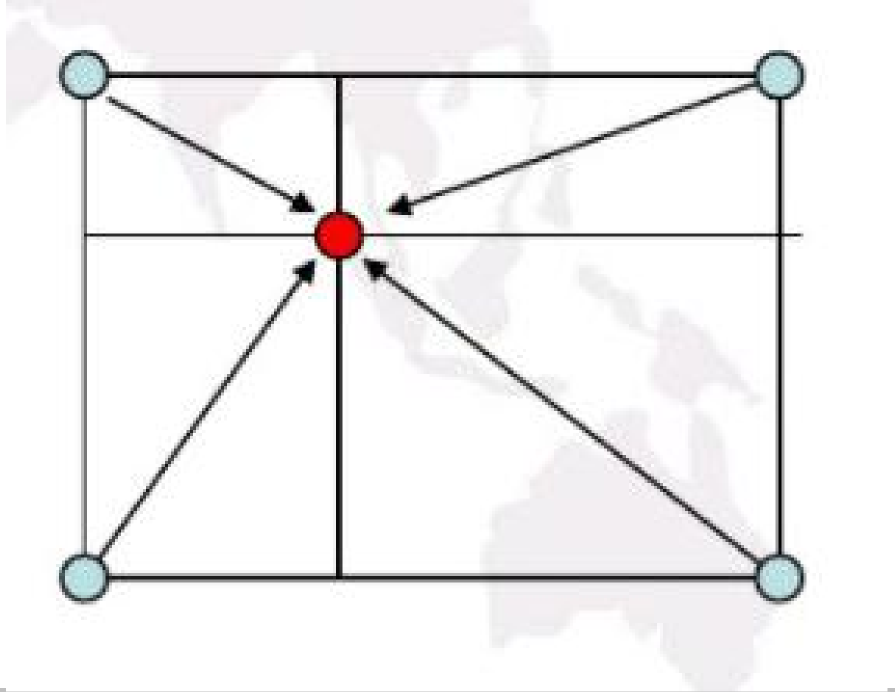
计算空间变换

- 问题：变换矩阵未知，其参数通过若干已知的匹配对应点后求解得到



插值填充

- 问题：为什么需要插值填充？
  - 变换后的坐标 (x', y')可能不是整数，如何取整？
    - 最近邻插值：使用离(x’,y’)最近的像素的值，不够精确
    - 双线性插值：使用(x’,y’)的4邻域的像素值插值计算



2.5 图像坐标变换

- 几何失真校正
  - **空间变换：** 对图像平面上的像素进行重新排列以恢复像素原空间关系

$$x' = s(x, y) \qquad s(x, y) = k_1 x + k_2 y + k_3$$

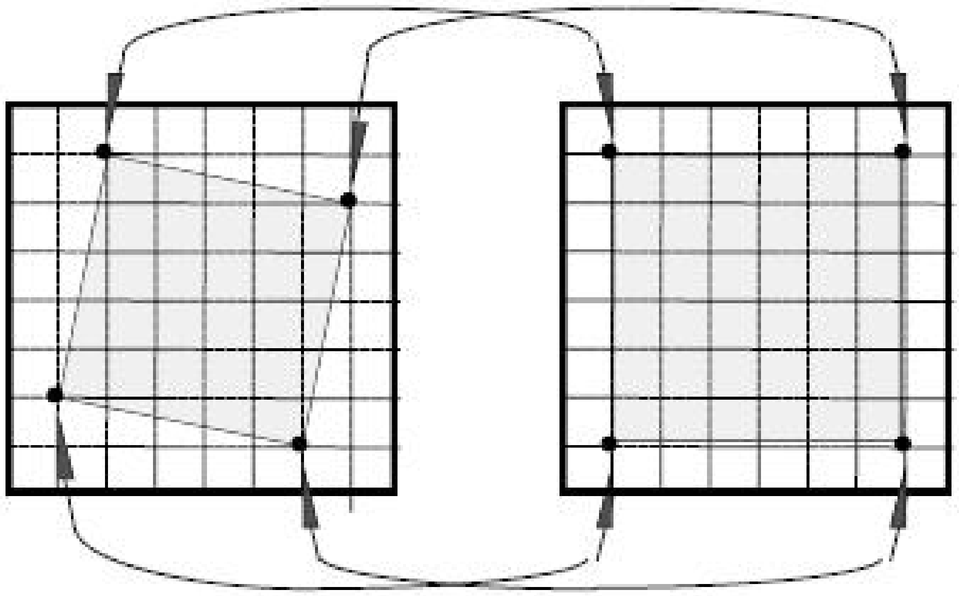
$$y' = t(x, y) \qquad t(x, y) = k_4 x + k_5 y + k_6$$

$$s(x, y) = k_1 + k_2 x + k_3 y + k_4 x^2 + k_5 xy + k_6 y^2$$

$$t(x, y) = k_7 + k_8 x + k_9 y + k_{10} x^2 + k_{11} xy + k_{12} y^2$$

2.5 图像坐标变换

- 几何失真校正
  - 一个在失真图上的四边形区域和在校正图上与其对应的四边形区域的顶点可作为对应点



$$x' = k_1 x + k_2 y + k_3 xy + k_4$$

$$y' = k_5 x + k_6 y + k_7 xy + k_8$$

图 2.5.4　失真图和校正图的对应点

2.5 图像坐标变换

- 几何失真校正
  - **灰度插值（最近邻插值/零阶插值）**：对空间变换后的像素赋予相应的灰度值以恢复原位置的灰度值

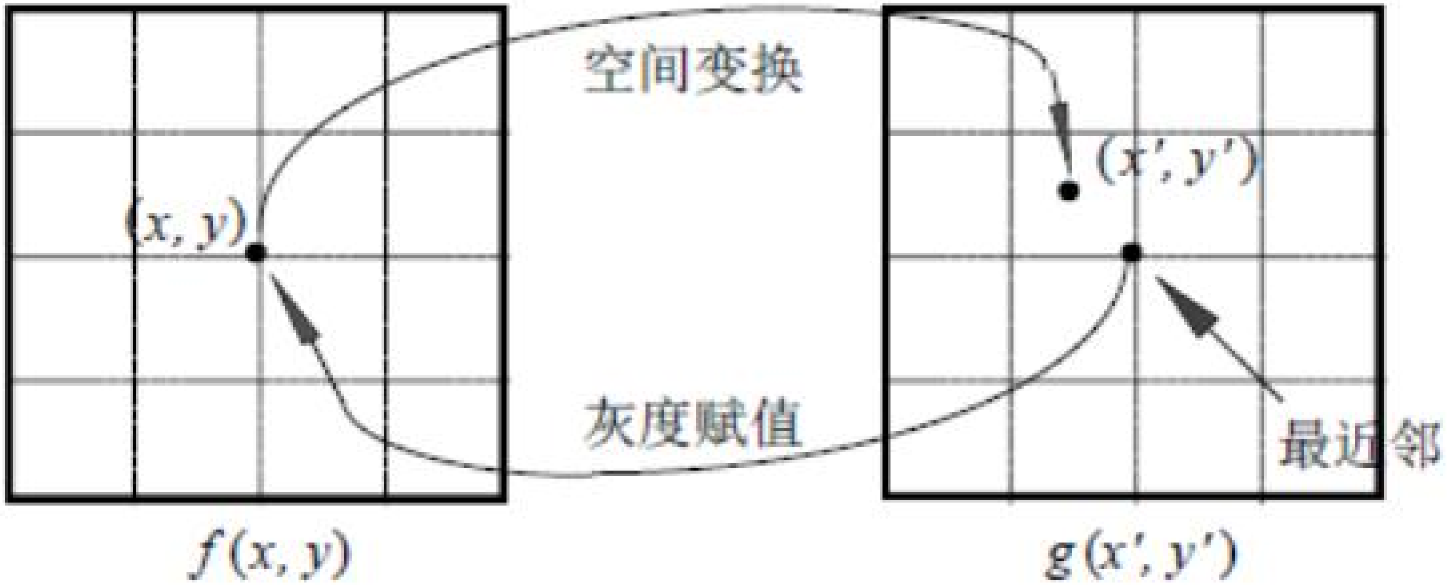
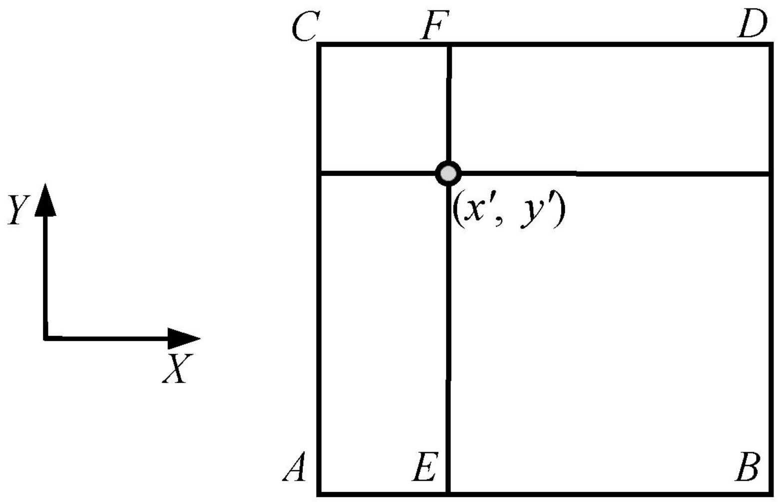


图 2.5.5 灰度插值示意图

- 几何失真校正
  - **双线性插值**：
    - 利用 (x', y')点的4个最近邻像素的灰度值来计算 (x', y')点处的灰度值



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

第2章　图像采集

- 2.1 几何成像模型
- 2.2 亮度成像模型
- 2.3 采样和量化
- 2.4 像素间联系
- 2.5 图像坐标变换和应用