

【计算机视觉】复习笔记 (。・`ω´・)

原创 长歌承闲 最后发布于2018-01-13 00:58:43 阅读数 1663 ☆ 收藏

展开

部分归纳来自 @雨诺寒雪, @ZJU_fish1996, @lutos, 感谢大佬们一起造福后人 (。・v・)ノ`

• 计算机视觉

计算机视觉的**首要目标**是用图像创建或恢复现实世界模型，然后认知现实世界
中心任务就是对图像进行理解

• 格式塔法则

格式塔理论的出发点是**形**，任何形都是**知觉**进行了**积极组织或构造**的结果和功能，而不是客体本身就有的

-

1. Law of Proximity

接近的物体容易被感知成同一组

-

2. Law of Similarity

人们将相似的物体感知成同一组的部分

-

3. Law of Common Fate

人们将以相同的速度和（或）方向运动的物体感知成一个组的部分

-

4. Law of Symmetry

互相对称的元素容易被感知为统一的组

-

5. Law of Continuity

人们会把一些离散的物体看成是连续的

-

6. Law of Closure

人们会把不完整的个体看成是一个整体的形状

• Marr视觉表示框架的三个阶段

Primal Sketch:

将输入的原始图像进行处理，抽取图像中诸如**角点**，**边缘**，**纹理**，**线条**，**边界**等基本特征，这些特征的集合称为基元图

-

2.5D Sketch:

在以观测者为中心的坐标系中，由输入图像和基元图恢复场景可见部分的**深度**，**法线方向**，**轮廓**等，这些信息称为二维半图

-

3D Model:

在以物体为中心的坐标系中，由输入图像，基元图，二维半图来恢复，表示和识别三维物体

• 二值图像

• 几何特性

-

1. 尺寸和位置

1.1 面积（零阶矩）：表示前景的像素点个数

$$A = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} B[i, j]$$

1.2 区域中心（一阶矩）：

$$\bar{x} = \frac{\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} jB[i, j]}{A}$$

$$\bar{y} = -\frac{\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} iB[i, j]}{A}$$

2. 方向

某些形状（如圆）是没有方向的。假定物体是长形的，**长轴方向**为物体的方向

2.1 求方向：

实际转化成一个最小化问题，需要找到一条满足以下条件的直线的方向：所有的前景像素点到该直线的距离之和最小

2.2 最小二乘法拟合求解：

• 直线方程 $\rho = x \cos \theta + y \sin \theta$

• 距离 $r^2 = (x \cos \theta + y \sin \theta - \rho)^2$

• 微分求解

$$\sin 2\theta = \pm \frac{b}{\sqrt{b^2 + (a-c)^2}}$$

$$\cos 2\theta = \pm \frac{a-c}{\sqrt{b^2 + (a-c)^2}}$$

$$a = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} (x_{ij} - \bar{x})^2 B[i, j]$$

$$b = 2 \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} (x_{ij} - \bar{x})(y_{ij} - \bar{y}) B[i, j]$$

$$c = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} (y_{ij} - \bar{y})^2 B[i, j]$$

http://blog.csdn.net/Dango_miracle

3. 伸长率，密度集，形态比

3.1 伸长率：图像短轴方向度量与长轴方向度量之比

$$E = \frac{\chi_{\max}}{\chi_{\min}}$$

3.2 密集度：

其中 **p**: 周长，**A**: 面积

$$C = \frac{A}{p^2}$$

给定周长，密集度越高则围成的面积就越大：

圆>正方形>长方形

3.3 形态比: 区域的最小外接矩形的长与宽之比

3.4 欧拉数

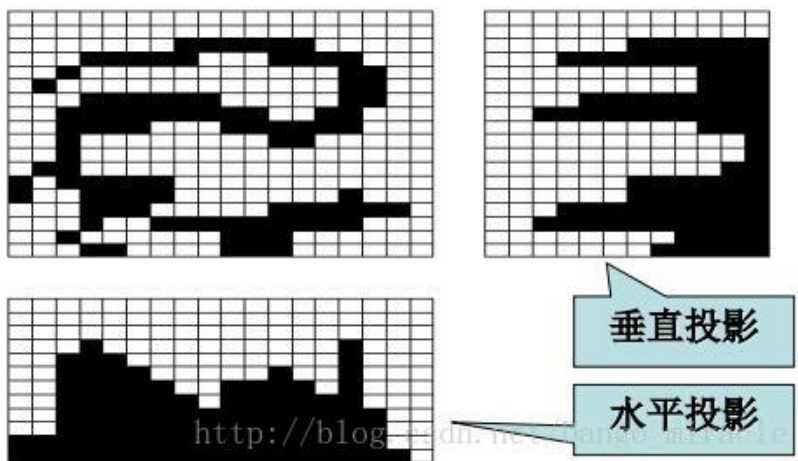
又叫亏格数, 定义为联通分量数减去洞数, 例如:

A B C

$E=1-1=0$ //blog $E=1-2=-1$ Dango $E=1-0=1$

• 投影计算

定义: 给定一条直线, 用垂直该直线的一簇等间距直线将一幅二值图像分割成若干条, 每一条内像素值为1的像素的数量



• 连通区域

概念: 四联通和八联通

联通分量的定义: 联通像素的集合

联通分量标记算法:

递归算法:

1. 扫描图像, 找到没有标记的一个前景点, 分配标记L

2. 递归分配标记L给该点的邻点
3. 如果不存在没标记的点，则停止
4. 返回第1步

序贯算法（四联通）：

1. 从左到右，从上到下扫描图像
2. 如果像素点值为1，则分四种情况：
 - 2.1 如果上面点和左面点有且仅有一个标记，则复制这一标记
 - 2.2 如果两点有相同标记，则复制这一标记
 - 2.3 如果两点有不同标记，则复制上点的标记且将这两个标记输入等价表中作为等价标记
 - 2.4 否则给这一像素点分配一个新的标记并将这一标记输入等价表
3. 如果需要考虑更多的点，则返回2
4. 在等价表的每一等价集中找到最低的标记
5. 扫描图像，用等价表中的最低标记取代每一标记

区域边界跟踪算法：

区域边界跟踪算法

c: 当前点 (在边界上) **b**: 当前点的邻域点 (不在边界上)

n_8	n_7	n_6
n_1	c	n_5
n_2	n_3	n_4

(1)从左到右, 从上到下扫描图像, 求区域S的**起始点**

$$s(k) = (x(k), y(k)), k = 0$$

(2)用**c**表示当前边界上被跟踪的像素点, 置**c**=**s(k)**, 记**c**的**左邻点**为 $b, b \in \bar{S}$

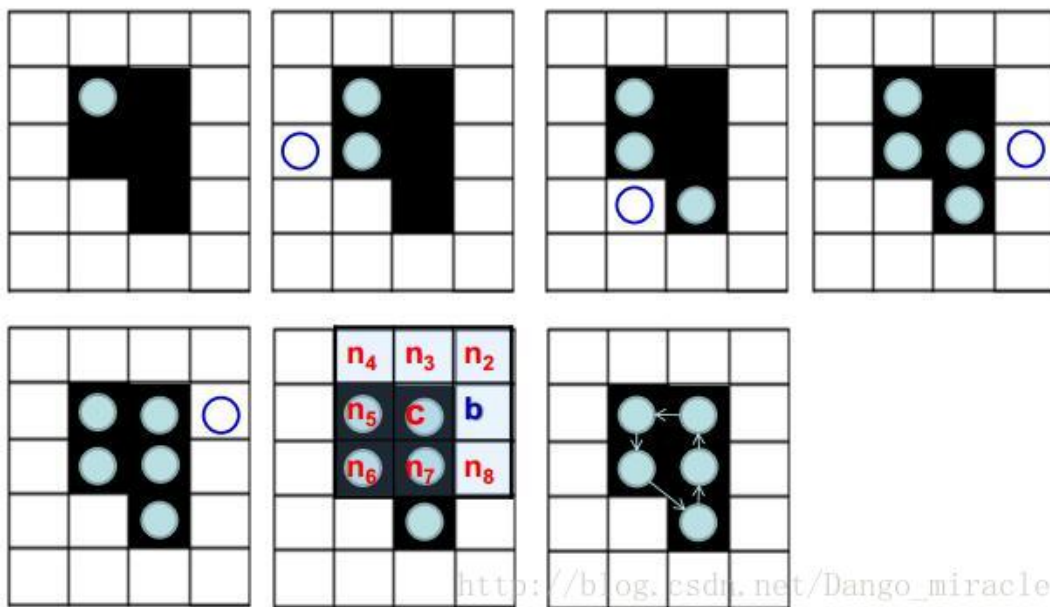
(3)按逆时针方向记从**b**开始的**c**的8个8邻点分别为 $n_1, n_2, \dots, n_8, k = k + 1$

(4)从**b**开始, 沿逆时针方向找到第一个 $n_i \in S$

(5)置 $c = s(k) = n_i, b = n_{i-1}$

保证了**b**不属于S

(6)重复步骤(3),(4),(5), 直到**s(k)=s(0)**



http://blog.csdn.net/Dango_miracle

• 边缘

• 模版卷积

给一个图像与一个模板，会计算卷积结果

• 四种最主要的不连续：

1. 表面法向量的不连续
2. 深度的不连续
3. 表面颜色的不连续
4. 光照的不连续

• 边缘检测的基本思想：

检测灰度不连续的地方

-

• 基于一阶的边缘检测：

一阶导数的极大值为边界点

- **梯度(Gradient):** 是图像对应二维函数的一阶导数

$$G(x, y) = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix}$$

- **梯度的幅值:** isotropic operator anisotropic

$$|G(x, y)| = \sqrt{G_x^2 + G_y^2}$$



$$|G(x, y)| = |G_x| + |G_y|$$

$$|G(x, y)| \approx \max(|G_x|, |G_y|)$$

- **梯度方向:**

$$\alpha(x, y) = \arctan(G_y / G_x)$$

梯度方向为函数最大变化率方向

- **图像中用差分近似偏导数**

实际运算中用差分近似微分:

$$G_x = f[x+1, y] - f[x, y]$$

$$G_y = f[x, y] - f[x, y+1]$$

一般用卷积模板进行计算：

$$G_x = \begin{bmatrix} -1 & 1 \end{bmatrix} \quad G_y = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

http://blog.csdn.net/Dango_miracle

种类有：

1. Roberts交叉算子
2. Sobel算子
3. Prewitt算子：运算较快

• 基于二阶的边缘检测：

图像灰度二阶导数的过零点对应边缘点

种类有：

1. Laplacian算子
2. LoG算子（Marr&Hildreth算子）：高斯滤波 + 拉普拉斯边缘检测

- 为什么要加高斯滤波：平滑去噪和边缘检测是一对矛盾，应用高斯函数的一阶导数，可以在两者之间获得一个较好的平衡

• Canny边缘检测

1. 用高斯滤波器平滑图像
2. 用一阶偏导有限差分计算**梯度幅值和方向**
3. 对梯度幅值进行**非极大值抑制 NMS**：

先将所有梯度角离散为圆周的八个扇区之一，然后去掉不比梯度线方向上的两个相邻点幅值大的点

4. 用**双阈值算法**检测和连接边缘

为什么需要非极大值抑制：

可以进一步消除非边缘的噪点，更重要的是可以细化边缘，将模糊的边界变得清晰

双阈值算法：

(a) 取高低两个阈值(T_2, T_1)作用于新幅值图 $N[i,j]$, 得到两个边缘图: 高阈值和低阈值边缘图。

高阈值图: $N[i,j] > T_2$;

低阈值图: $N[i,j] > T_1$

(b) 连接高阈值边缘图, 出现断点时, 在低阈值边缘图中的8邻点域搜寻边缘点。

阈值太低: 假边缘

阈值太高: 部分轮廓丢失

• 局部特征 Local Feature

• Harris角点检测

原理: 给定包围一个点的窗口, 在任何方向上移动窗口, 强度都会发生较大变化, 即为角点

公式推导:

Harris算法是利用窗口内图像的灰度自相关性进行的, 设定一个窗口并在图像中移动, 计算移动前与移动后窗口所在的区域图像的自相关系数

Change of intensity for the shift $[u, v]$:

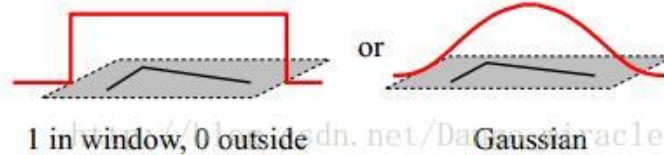
$$E(u, v) = \sum_{x, y} w(x, y) [I(x+u, y+v) - I(x, y)]^2$$

Window
function

Shifted
intensity

Intensity

Window function $w(x, y) =$



- If u, v are small in values,

$$I(x+u, y+v) \approx I(x, y) + uI_x(x, y) + vI_y(x, y)$$

$$\begin{aligned} E(u, v) &= \sum_{x, y} w(x, y) [I(x+u, y+v) - I(x, y)]^2 \\ &= \sum_{x, y} w(x, y) [I(x, y) + uI_x(x, y) + vI_y(x, y) - I(x, y)]^2 \\ &= \sum_{x, y} w(x, y) [uI_x(x, y) + vI_y(x, y)]^2 \\ &= [u, v] \left(\sum_{x, y} w(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \right) \begin{bmatrix} u \\ v \end{bmatrix} \end{aligned}$$

For small shifts $[u, v]$ we have a *bilinear* approximation:

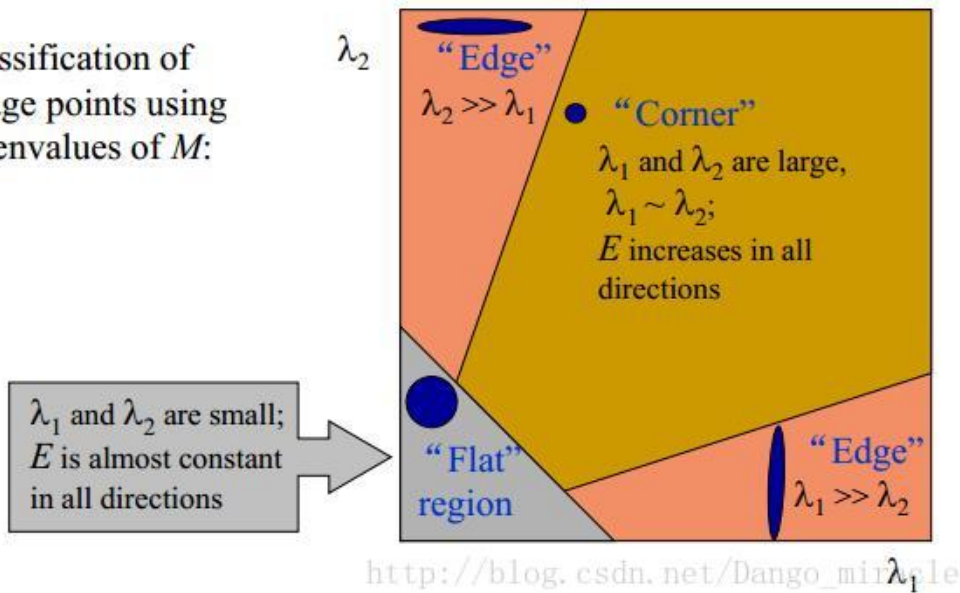
$$E(u, v) \cong [u, v] M \begin{bmatrix} u \\ v \end{bmatrix}$$

where M is a 2×2 matrix computed from image derivatives:

$$M = \sum_{x,y} w(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

Harris Detector: Mathematics

Classification of image points using eigenvalues of M :



λ_1 和 λ_2 是 M 的特征向量

Measure of corner response:

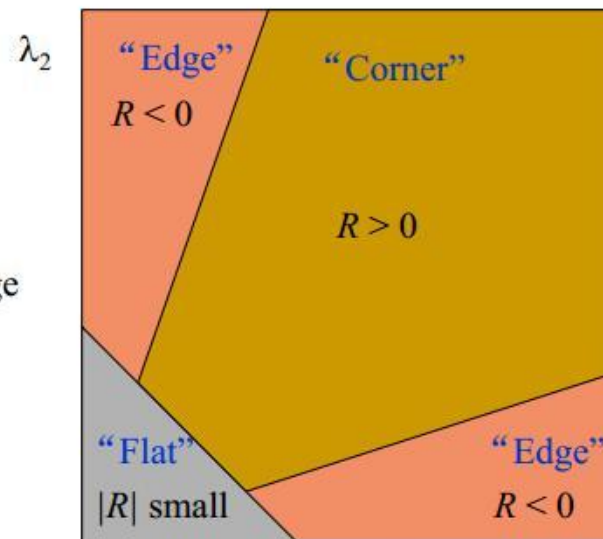
$$R = \det M - k (\text{trace } M)^2$$

$$\det M = \lambda_1 \lambda_2$$

$$\text{trace } M = \lambda_1 + \lambda_2$$

(k = empirical constant, $k = 0.04-0.06$)

- R depends only on eigenvalues of M
- R is large for a **corner**
- R is negative with large magnitude for an **edge**
- $|R|$ is small for a **flat** region



http://blog.csdn.net/Dango_miracle

■ The Algorithm:

- Find points with large corner response function R ($R > \text{threshold}$)
- Take the points of local maxima of R /Dango_miracle

Harris角点有旋转不变性，但没有尺度不变性：旋转不变性是因为我们取的是局部最大。尺度放大以后一个原来的角点会被检测成edge
设计一个尺度不变函数，不同尺度下的图片找到的区域是相同的：

1. 在多尺度检测关键点
2. 然后找上下不同尺度的局部最大值
3. 消除低于阈值的点

• SIFT描述子的计算（得到128维的

计算的基本步骤：

1. 用 16×16 的窗口放在特征点附近 + 将 16×16 分成16个 4×4 的窗口
2. 计算窗口中每个像素的边的方向（梯度角减去 90° ）
3. 丢掉方向能量小的边（使用阈值）
4. 用直方图描述结果 + 将每个小窗口中的所有的方向离散成8个方向，一共 $16 \times 8 = 128$ 个

为什么只使用梯度信息：

因为梯度信息可以表示边缘信息，并且在光照变化时有抵抗能力

如何实现旋转不变的：

因为我们找的是对应位置的参考方向而非绝对方向

• 尺度不变的原理

因为在使用高斯模糊的不同尺度（如图像金字塔）处重新采样图像，仅当在两个尺度之间观察到最大值时才将梯度存储为描述符

• 曲线

• Hough变换直线检测：

Hough变换是基于投票原理的参数估计方法，是一种重要的形状检测技术

基本思想：图像中的每一点对参数组合进行表决，赢得多数票的参数组合为胜者（结果）

步骤：

1. 适当地量化参数空间

2. 假定参数空间每一个单元都是一个累加器，把累加器初始化为0
3. 对图像空间的每一点，在其所满足的参数方程对应的累加器上加1
4. 累加器阵列最大值对应模型的参数

• 图像频域

• 图像的傅里叶变换

理解变换的基本含义：

傅里叶变换讲的是任何信号（如图像信号）都可以表示成一系列正弦信号的叠加，在图像领域就是将图像brightness variation 作为正弦变量。傅里叶变化储存每一个频率的 magnitude和phase信息，前者代表这个频率上有多少信号，后者间接可代表空间信息。

理解图像的低频成分与高频成分：

图像的高频部分是像素值变化剧烈的地方，如图像的边缘和轮廓。低频部分是变化不剧烈的地方，对应大的色块。我们从近处看图像看到的是高频信息，而远处看到的是低频信息

-

• 怎么理解拉普拉斯金字塔的每一层是带通滤波？

拉普拉斯金字塔是通过源图像减去先缩小再放大的一系列图像构成的。下采样的时候丢失了高频信息，而相邻金字塔相减的时候丢失了低频信息，因此只有中间频段的信息保留了下来

• 相机模型

• 理解：景深，光圈，焦距，视场

景深：相机镜头能够取得清晰图像的成像所测定的被摄物体前后范围距离

光圈：镜头中，用来控制光线透过镜头，进入机身内感光面光量的装置

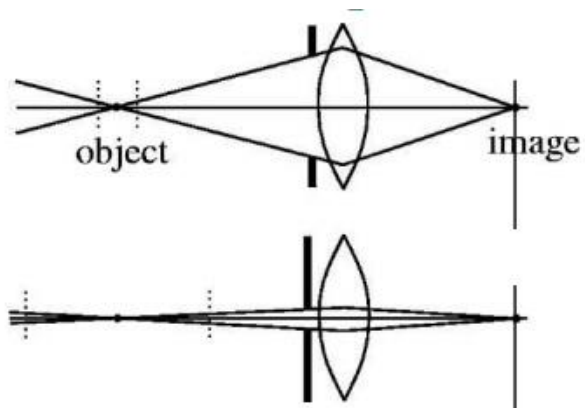
焦距：从镜片中心到底片等成像平面的距离

视场：摄像头能够观察到的最大范围

-

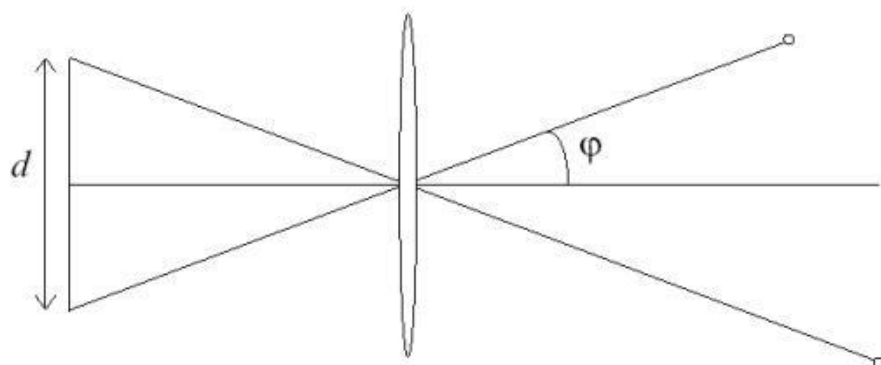
光圈对景深的影响？

较小的光圈对应较大的景深（但是较小的光圈限制了光线的进入，因此需要增加曝光）



焦距对视场的影响？

焦距越短，视场角越大，放大倍率越小，拍摄范围越大，拍摄画面中的人越小



Size of field of view governed by size of the camera retina:

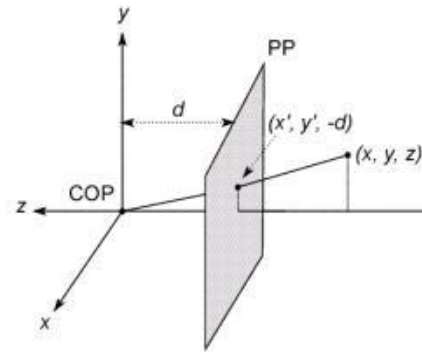
$$\varphi = \tan^{-1}\left(\frac{d}{2f}\right)$$

http://blog.csdn.net/Dango_miracle

• 理想的针孔相机模型

基本投影：

Modeling projection



■ Projection equations

- Compute intersection with PP of ray from (x,y,z) to COP

- Derived using similar triangles (on board)

$$(x, y, z) \rightarrow \left(-d\frac{x}{z}, -d\frac{y}{z}, -d\right)$$

- We get the projection by throwing out the last coordinate:

$$(x, y, z) \rightarrow \left(-d\frac{x}{z}, -d\frac{y}{z}\right) \text{ (Fundamental Equations)}$$

齐次坐标形式下的透视投影公式:

负号表示倒立

Perspective Projection (透视投影)

- Projection is a matrix multiply using homogeneous coordinates:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1/d & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ -z/d \end{bmatrix} \Rightarrow \left(-d\frac{x}{z}, -d\frac{y}{z}\right)$$

divide by third coordinate

- This is known as **perspective projection**

- The matrix is the **projection matrix**

- Can also formulate as a 4x4 or a 3x4 matrix

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1/d & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ -z/d \end{bmatrix} \Rightarrow \left(-d\frac{x}{z}, -d\frac{y}{z}\right)$$

有哪几个内参:

$$(f_x, f_y, c_x, c_y)$$

All the parameters inside

$$\begin{cases} x_{screen} = f_x \left(\frac{Y}{Z} \right) + c_x, \\ y_{screen} = f_y \left(\frac{X}{Z} \right) + c_y \end{cases} \quad f_x = Fs_x, f_y = Fs_y$$

The units

- F : mm
- s_x, s_y : pixel/mm
- f_x, f_y : pixel

http://blog.csdn.net/Dango_miracle

会写内参矩阵:

$$q = MQ, \text{ where } q = \begin{bmatrix} x \\ y \\ w \end{bmatrix}, M = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, Q = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

http://blog.csdn.net/Dango_miracle

对角线和竖着的分别代表什么含义?

分别代表焦距对应的单位像素和中心点的偏移

• 畸变

径向畸变与切向畸变各是什么原因引起的?

径向畸变:

不完美的镜头, 镜片的不理想, 镜头的几何形状, 光圈位置

↑ 概括成透镜几何性质和孔径位置

远离透镜中心的地方比靠近中心的地方更加弯曲

如何解决:

$$x_{\text{corrected}} = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6)$$

$$y_{\text{corrected}} = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6)$$

To model lens distortion

- Use above projection operation instead of standard projection matrix multiplication

切向畸变:

由于CMOS等感光元件摆放倾斜，没有平行于图像平面

越靠近中间，畸变越小

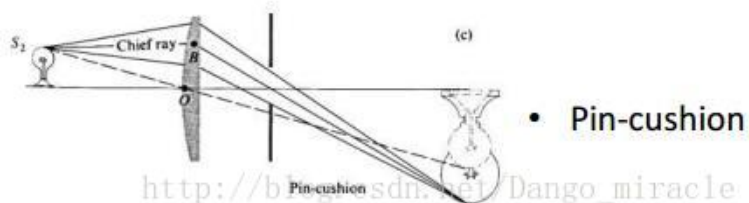
如何解决:

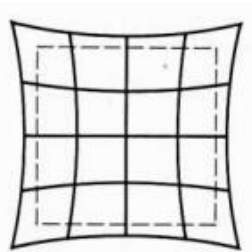
$$x_{\text{corrected}} = x + [2p_1 y + p_2(r^2 + 2x^2)]$$

$$y_{\text{corrected}} = y + [p_1(r^2 + 2y^2) + 2p_2 x]$$

径向畸变常见的有哪两种?

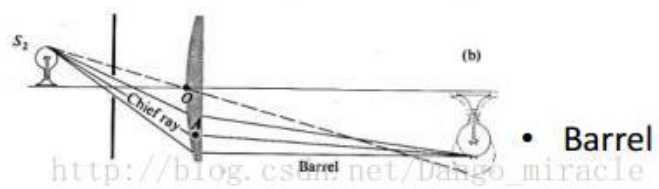
枕形畸变:



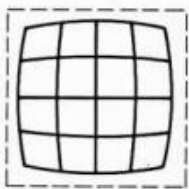


Pin cushion

桶形畸变:



• Barrel



Barrel

畸变参数:

Called **distortion parameters**

$(k_1, k_2, p_1, p_2, k_3)$

- 外参有哪几个？分别代表什么含义？（每拍一张照片就有一个外参）

Called **extrinsic parameters**

$$(\theta, \varphi, \psi, t_x, t_y, t_z)$$

外参用于模型空间到相机空间的变化，三个参数是平移参数，三个参数是旋转参数

$$P' = \begin{bmatrix} \mathbf{R}_{3 \times 3} & \mathbf{t}_{3 \times 1} \\ 0 & 0 & 0 & 1 \end{bmatrix} P$$

$$R = R_z(\theta), R_y(\varphi), R_x(\psi)$$

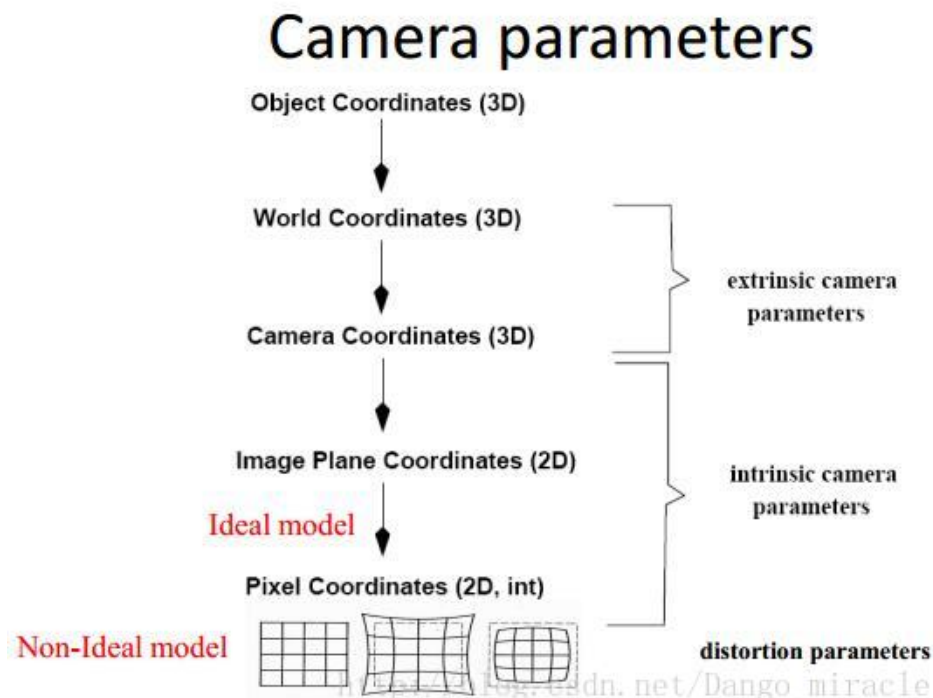
$$\mathbf{t}_{3 \times 1} = (t_x, t_y, t_z)'$$

http://blog.csdn.net/Dango_miracle

- 内参，外参，畸变参数在成像各阶段中的角色（三维物体到真实图像的过程）

1. 第一步是从世界坐标系转为相机坐标系，这一步是从三维点到三维点的转换，包括R, t等参数（相机外参）
2. 第二步是从相机坐标系转为成像平面坐标系（像素坐标系），这一步是三维点到二维点的转换，包括K等参数（相机内参）

3. 最后再用到畸变参数



• 相机标定

• 相机标定需要求解哪些参数?

畸变参数, 内参, 外参

• 基于Pattern/Reference Object的相机标定

已知什么? 求解什么?

已知: 给定标定物体的N个角点, K个视角 (棋盘格子两个点可以得出四个等式)

求解: 所有的参数。N个点K个视角可以列出 $2NK$ 个等式, 会带来 $6*K+4$ 个参数。每次会变的是外参, 而内参和畸变参数是不变的, 所以只需要 $2NK > 6*K+4$ 即可

简述其基本过程哪几个步骤?

1. 获取标定物体网格的角点在坐标系的位置
2. 找到图片的角点
3. 根据图像空间坐标系到世界坐标系列出等式

4. 求解相机参数

即：拍照 - 找特征点 - 特征点对应 - 计算公式和参数

• 立体视觉

• 立体视觉的三角测量基本原理

假设有两个焦距相同的完美相机，假设我们知道3维空间点在2维图像上的具体位置

Triangulation

- **Assume** we have a stereo **rig** (two cameras with the same f in this case) that is **perfectly**
 - Undistorted
 - Aligned -- *frontal parallel*
 - Measured
- **Assume** we know the 2D-positions of a 3D-point in the two images.
- Similar triangles!

http://blog.csdn.net/Dango_miracle

会画视差disparity的那张图：

$$\frac{T - (x^l - x^r)}{Z - f} = \frac{T}{Z} \Rightarrow Z = \frac{fT}{x^l - x^r}$$

Triangulation

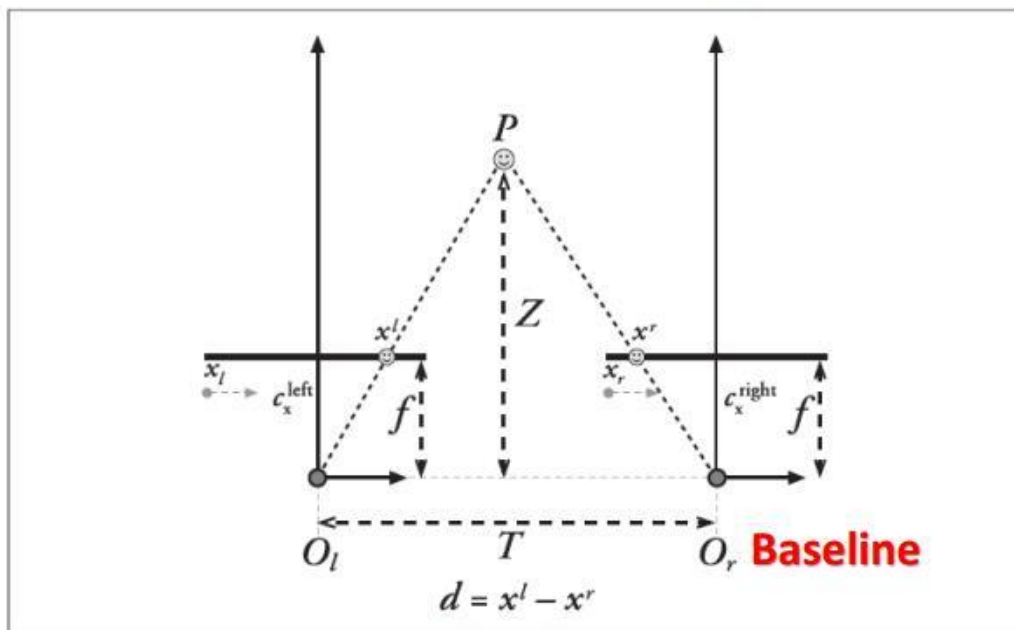


Figure 12-4. With a perfectly undistorted, pinhole camera and known correspondence, the depth Z can be found by similar triangles; the principal rays of the imagers begin at the centers of projection

推导上图公式:

要知道: 相机的位置, 点的对应关系 (真实环境中的点到平面上)

视差: $x^l - x^r$, 表示左右两摄像头成像的距离

Z 的结果误差主要在分母 (视差) 那里。视差小的时候, 视差的误差会对 Z 产生很大的影响。此外 T 越小, 误差越小
 T 越大, 看到的范围越小 (因为是取两眼图像的交叉部分)

• 立体视觉的基本步骤

How to Do Stereo?

- Four major steps:
 - **Undistortion**
 - remove distortions -> undistorted images
 - **Rectification**
 - adjust cameras -> the two images row-aligned
 - **Correspondence**
 - find the same features in the two images -> disparity
 - **Reprojection**
 - triangulation -> a depth map

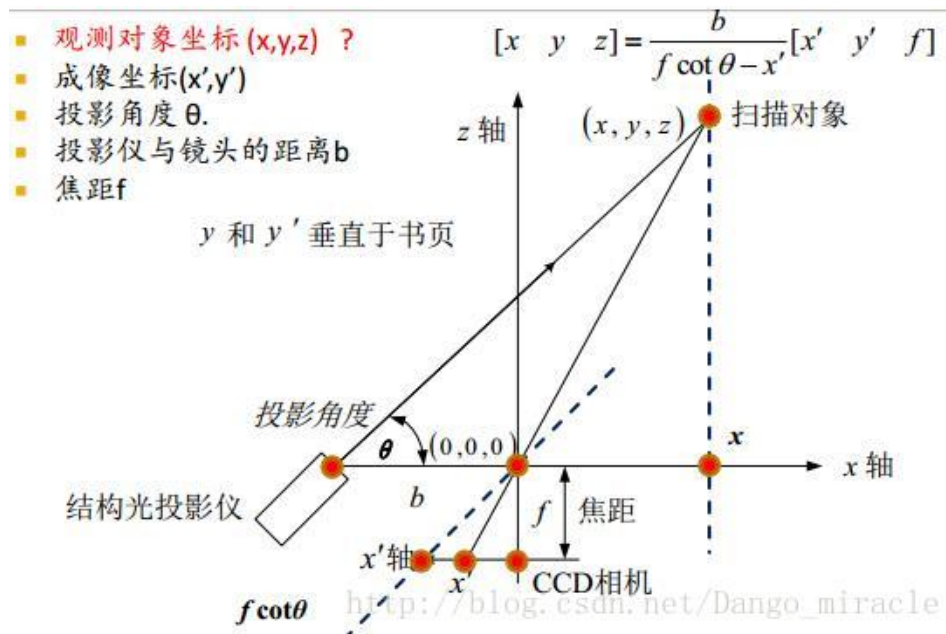
1. 恢复失真，消除畸变
2. 矫正相机，使图像在同一个平面上
3. 在两张图中找到对应的相同特征
4. 三角测量

• 三维数据获取

• 结构光成像系统的构成：

一个结构光三维成像系统主要由三个部分组成：结构光投影仪（一台或多台），CCD相机（一台或多台），以及深度信息重建系统

• 利用结构光获取三维数据的基本原理



公式可以由三角形相似推出来 (公式不要弄混了!)

• 三维获取: Kinect

kinect的原理是单目结构光

-

• ICP算法的作用及其基本步骤

ICP: 迭代最近点方法 (用于多个摄像机的配准问题, 即把多个扫描结果拼接在一起形成对扫描对象的完整描述)

基本步骤:

给定两个三维点集 X 与 Y , 将 Y 配准到 X :

1. 计算 Y 中的每一个点在 X 中的对应最近点
2. 求使上述对应点对的平均距离最小的刚体变换, 获得刚体变换参数 (平移参数和旋转参数)
3. 对 Y 应用上一步求得的刚体变换 (平移和旋转), 更新 Y
4. 如果 X 与 Y 的对应点对平均距离大于阈值, 回到1, 否则停止计算

• 光流

• 光流解决的是什么问题?

给定两张图像 H 和 I , 评估从 H 到 I 的像素运动, 解决的是像素对应的问题

- 光流的三个基本假设是什么?

亮度一致, 空间一致, 移动小

- 一个点的约束公式会推导

Optical flow equation

■ Combining these two equations

$$\begin{aligned}
 0 &= I(x + u, y + v) - H(x, y) \\
 &\approx I(x, y) + I_x u + I_y v - H(x, y) \quad \text{shorthand: } I_x = \frac{\partial I}{\partial x} \\
 &\approx (I(x, y) - H(x, y)) + I_x u + I_y v \\
 &\approx I_t + I_x u + I_y v \\
 &\approx I_t + \nabla I \cdot [u \ v]
 \end{aligned}$$

■ In the limit as u and v go to zero, this becomes exact

$$0 = I_t + \nabla I \cdot \left[\frac{\partial x}{\partial t} \ \frac{\partial y}{\partial t} \right] \quad // \text{blog.csdn.net/Dango_miracle}$$

- 哪些位置的光流比较可靠, 为什么

纹理复杂区域, 梯度比较大且方向不同, 求出来的特征值比较大

• 图像拼接

- 图像拼接的基本步骤

1. 检测关键点
2. 计算SIFT描述子
3. 匹配SIFT描述子
4. 计算变换矩阵, 进行仿射变换
5. RANSAC
6. 图像混合 (边缘平滑过渡)

-

• RANSAC: 随机抽样一致算法

采用迭代的方式从一组包含outliers的数据中估算数学模型的参数。核心思想是**随机性和假设性**，随机性是根据正确数据出现的概率去随机选取抽样数据，假设性是假设选取出的抽样数据都是正确数据。

优点：是大范围模型匹配问题的一个普遍意义上的方法，且运用简单，计算快。

缺点：只能计算outliers不多的情况（投票机制可以解决outliers高的情况）

基本步骤：

1. 随机选择一些点作为样本
2. 计算选出的样本应该使用的变换矩阵
3. 把刚才没有选中的点代入建立的模型，判断有多少点符合，误差是否小于阈值
4. 比较匹配数量是否为当前最优解，如果是，则更新当前最优集，并更新迭代次数
5. 重复多次，如果迭代次数大于K（k由最优的inliers的点集计算得到），则退出，否则迭代次数+1

• 人脸识别

• 主成分分析PCA

基本思想：选择一个新的坐标系统进行线性降维，第一轴是数据集的投影的方差最大的方向，第二轴是数据集投影的方差第二大方向

作用：降维和去噪

优化目标函数的推导：

PCA求解

$$\begin{aligned}
 \text{var}(z_1) &= E(z_1^2) - [E(z_1)]^2 = E\left[\left(\sum_{i=1}^d a_1^i x_i\right)^2\right] - \left(E\left(\sum_{i=1}^d a_1^i x_i\right)\right)^2 \\
 &= \sum_{i,j=1}^d a_1^i a_1^j E(x_i x_j) - \sum_{i,j=1}^d a_1^i a_1^j E(x_i) E(x_j) \\
 &= \sum_{i,j=1}^d a_1^i a_1^j [E(x_i x_j) - E(x_i) E(x_j)] \\
 &= \sum_{i,j=1}^d a_1^i a_1^j S_{ij} \quad \text{令 } S_{ij} = E(x_i x_j) - E(x_i) E(x_j) \\
 &= \mathbf{a}_1^T \mathbf{S} \mathbf{a}_1 \quad \text{S? } \text{cov}(x_i, x_j) = E(x_i x_j) - E(x_i) E(x_j)
 \end{aligned}$$

• Eigenface

是什么：所有人脸图像展开后构成的大矩阵的协方差矩阵的特征向量

人脸识别方法的基本步骤：

1. 获得人脸图像训练集
2. 对人脸图像归一化处理
3. PCA计算获取一组特征向量
4. 将每幅图像投影到特征向量的空间中，得到新的空间的坐标
5. 对输入的待测试图像，归一化后投影到特征人脸空间中，然后用距离度量来描述两幅人脸图像的相似性

将重构用于人脸检测的原理：

人脸投影到特征人脸空间中，保留了主要特征的信息，所以可以恢复人脸本来的样子

• 物体识别

• Visual Recognition

基本任务大概可以分为哪几大类?

1. 图片分类
2. 检测和定位物体/图片分割
3. 估计语义和几何属性
4. 对人类活动和事件进行分类

都有哪些挑战因素?

视角变换

光线变化

尺度变化

物体形变

物体遮挡

背景凌乱

内部类别多样

• 基于词袋的物体分类:

图像的BoW (bag-of-words) 是指什么意思?

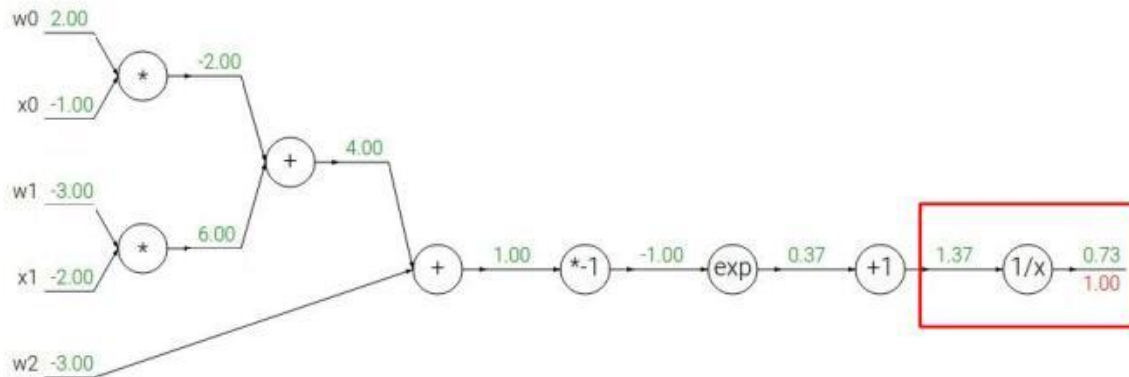
图像中的单词被定义为一个图像块的特征向量, 图像的Bow模型即图像中所有图像块的特征向量得到的直方图

几个基本步骤

1. 特征提取和表示 (grid), 每个特征为一个质点
2. 通过对质点聚类建立字典 (k 聚类), 得到k个聚类中心, 聚类中心就是词袋中的单词, 所有聚类中心就是特征直方图的基
3. 将图片用直方图的基表示出来, 这样就可以得到关于图片的特征直方图。该直方图与单词的顺序没有关系, 而是每个单词在图片中出现的频率
4. 将新的图片获取质点, 然后映射到直方图上进行聚类

• 物体识别-CNN

Another example: $f(w, x) = \frac{1}{1 + e^{-(w_0x_0 + w_1x_1 + w_2x_2)}}$



$f(x) = e^x$	\rightarrow	$\frac{df}{dx} = e^x$		$f(x) = \frac{1}{x}$	\rightarrow	$\frac{df}{dx} = -1/x^2$
$f_a(x) = ax$	\rightarrow	$\frac{df}{dx} = a$		$f_c(x) = c + x$	\rightarrow	$\frac{df}{dx} = 1$

• 卷积神经网络:

卷积层: 提取特征, 用了共享参数

池化层: 缩小 (如max pooling, 每四个像素点只取最小的那个), 让它只专注于关注某一小块

缩小 - 卷积 - 缩小 - 卷积.....

-

卷积神经网络和多层感知机最大的差别是什么?

引入了卷积层和池化层

要会计算卷积层的连接数和权重数:

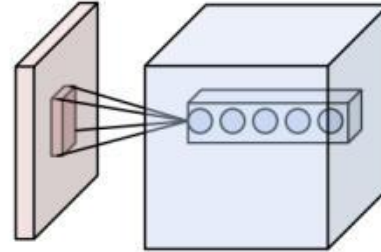
ConvNets: architecture

Examples time:

Input volume: **32x32x3**

Receptive fields: **5x5**, **stride 1**

Number of neurons: **5**



Output volume: $(32 - 5) / 1 + 1 = 28$, so: **28x28x5**

How many weights for each of the 28x28x5

neurons? **5x5x3 = 75** http://blog.csdn.net/Dango_miracle

ConvNets: architecture

There's one more problem...

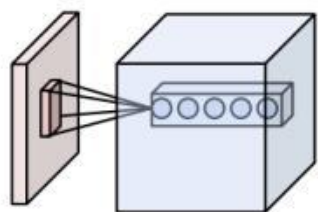
Assume input **[32 x 32 x 3]**

30 neurons with receptive fields **5x5**, applied at **stride 1/pad1**:

=> Output volume: **[32 x 32 x 30]** ($32 \times 32 \times 30 = 30720$ neurons)

Each neuron has $5 \times 5 \times 3 (=75)$ weights

=> Number of weights in such layer: $30720 \times 75 \approx 3 \text{ million} : \backslash$



← Example trained weights

https://blog.csdn.net/Dango_miracle

• 图像分割

基于K means聚类的图像分割

先表示，再聚类

理解用聚类进行图像分割的基本原理

给定一幅图像，能描述如何用K means进行分割的算法基本步骤

mean shift图像分割的思路

空间分布密集程度

*. 最后再看看这门课的老师叫什么名字吧，万一考到了呢`v`)

👍 点赞 5 ☆ 收藏 📄 分享 ...



长歌承闲

发布了23 篇原创文章 · 获赞 34 · 访问量 4万+

私信

关注

©2019 CSDN 皮肤主题: 大白 设计师: CSDN官方博客