

Convolution & Hough Transform

Convolution & Hough Transform

Convolution

- 1.continuous formula
- 2.discrete formula
- 3.example in cv: smoothing
- 4.sum:
5. Why convolution in deep learning?
- 6.卷积的意义

霍夫变换

- 1.Params space
- 2.算法原理
- 3.算法优化

Convolution

1.continuous formula

$$(f * g)(n) = \int_{-\infty}^{\infty} f(\tau)g(n - \tau)d\tau$$

2.discrete formula

$$(f * g)(n) = \sum_{\tau=-\infty}^{\infty} f(\tau)g(n - \tau)$$

Notice that: $n = \tau + (n - \tau)$

3.example in cv: smoothing

$$f = \begin{bmatrix} a_{0,0} & a_{0,1} & a_{0,2} \\ a_{1,0} & a_{1,1} & a_{1,2} \\ a_{2,0} & a_{2,1} & a_{2,2} \end{bmatrix} g = \begin{bmatrix} b_{0,0} & b_{0,1} & b_{0,2} \\ b_{1,0} & b_{1,1} & b_{1,2} \\ b_{2,0} & b_{2,1} & b_{2,2} \end{bmatrix}$$

rotate matrix g 180°, do simple multiplication, get the new $c_{1,1}$.(g can be simple mean or Gaussian mean)

4.sum:

- 卷积可以理解为瞬时行为的持续性后果。
- 可以理解为先将g翻转，然后滑动叠加。
- cv中作为滤波器(卷积和)

5. Why convolution in deep learning?

- Params sharing: unchanged convolution kernel
- sparsity of connections: output depends only on a small number of inputs(size of convolution kernel)
- translation invariance

6.卷积的意义

- 物理意义可以是：瞬时行为的持续性后果，与Bayes类似，即此时的结果依赖之前的输出\假设
- 卷积的傅里叶变换是函数傅里叶变换的乘积：

$$\text{时域: } F[f(\tau) * g(\tau)] = F(\omega) \cdot G(\omega) \quad \text{频域: } F[f(\tau) * g(\tau)] = \frac{1}{2\pi} F(\omega) * G(\omega)$$

具有对称性

霍夫变换

1.Params space

直线方程 $y = kx + b$ 经极坐标转换后 ($k = \frac{-\cos\theta}{\sin\theta}, b = \frac{r}{\sin\theta}$) , 得到:

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

对于点 (x_0, y_0) 的某个参数 (r_0, θ_0) , 表示通过 (x_0, y_0) 的一条直线。

则 $r = x_0\cos\theta + y_0\sin\theta$ 表示为通过 (x_0, y_0) 的所有直线, 且为正弦函数

$$r = x_0\cos\theta + y_0\sin\theta = \sqrt{x_0^2 + y_0^2} \sin(\theta + \phi), \tan\phi = \frac{y}{x}$$

若点 (x_1, y_1) 的参数方程 $r = \sqrt{x_1^2 + y_1^2} \sin(\theta + \phi), \tan\phi = \frac{y}{x}$ 与 (x_0, y_0) 的参数方程相交于 (r_0, θ_0) , 则两点间的直线参数为 (r_0, θ_0) 。

据此可推广, 若找出圆、矩形的平面图形, 至少需要三点 (不共线) 的参数方程相交

2.算法原理

(图片需要预处理: 抑制噪声、灰度等)

霍夫变换通过accumulator (矩阵) 来确定位置参数。accumulator维数等于未知参数的数量 (每一行表示一个参数) 。

因此, 对于直线, 累加器维度为2, 对于圆 (平面图形) , 维度为3.

3.算法优化

- probabilistic Hough transform:
随机选取点集进行计算 (直线检测足够) , 但要相应降低threshold
- Hough gradient direction: 对于平面图形, 将累加器降成2维。