

邊緣偵測

1. Laplacian: 一種二階導數運算式，在邊緣處產生出一個零交叉，

如:4 相連、8 相連, Laplacian 運算式是透過明暗變化來偵測的，不具備檢測方向性，容易受外部干擾。

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

$$\begin{aligned}\nabla^2 f(x, y) &= [f(x+1, y) - f(x, y)] - [f(x, y) - f(x-1, y)] \\ &\quad + [f(x, y+1) - f(x, y)] - [f(x, y) - f(x, y-1)] \\ &= f(x+1, y) + f(x, y+1) + f(x-1, y) + f(x, y-1) - 4f(x, y)\end{aligned}$$

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

2. sobel：利用離散型差分算式來計算圖像函數梯度近似值，是透過 2

個 3X3 矩陣來對原圖進行運算，計算出兩個方向的灰度差分，分別是在橫向及縱向的灰度，定位精度不高，得到的邊緣比較粗，

$$\mathbf{G}_x = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix} * \mathbf{A} \quad \text{and} \quad \mathbf{G}_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * \mathbf{A}$$

適用於精度要求不高的場合。

3. Prewitt:與 sobel 相近，利用像素点上下左右鄰近像素的灰度差，在

邊緣處達到極值來檢測邊緣。Prewitt 梯度算法就是先求平均，再

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad G_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

求差分來求梯度。

4. Canny: 先由高斯平滑模板做運算，讓一個像素噪聲，再透過 4 個 mask 進行檢測，分別為水平、垂直、對角，求出亮度及方向，最後在設定大於多少及小於多少不是邊緣。

實作

分別對 opencv 範例、藥丸、自己的照片、課本房子照片進行邊緣偵測







