Consider responses to the edge:
$$\begin{matrix} 0 & h & h \\ 0 & 0 & h \\ 0 & 0 & 0 \end{matrix}$$

	Prewitt	Kirsch	Robinson-3	Robinson-5
E	$-h$	$-9h$	$-2h$	$-3h$
NE	$-3h$	$-9h$	$-3h$	$-4h$
N	$-h$	$-9h$	$-2h$	$-3h$
NW	$+h$	$-h$	0	0
W	$+3h$	$7h$	$2h$	$3h$
SW	$+3h$	$15h$	$3h$	$4h$
S	$+3h$	$7h$	$2h$	$3h$
SE	$+h$	$-h$	0	0
scale factor	$\frac{1}{5}$	$\frac{1}{15}$	$\frac{1}{3}$	$\frac{1}{4}$

input | output

$$\frac{1}{15}\begin{bmatrix} 5 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & -3 & -3 \end{bmatrix} \quad \begin{matrix} 0 & h & h \\ 0 & 0 & h \\ 0 & 0 & 0 \end{matrix} \quad \rightarrow \quad \frac{-9h}{15}$$

$$\frac{1}{15}\begin{bmatrix} -3 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & 5 & -3 \end{bmatrix} \rightarrow \frac{-9h}{15}$$

$\vdots$

$$\frac{1}{15}\begin{bmatrix} -3 & 5 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & -3 \end{bmatrix} \rightarrow \frac{15h}{15} = h$$

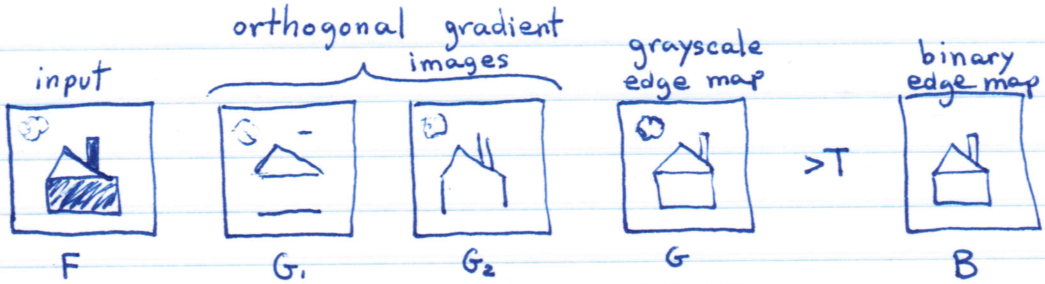
SW edge → obtain orientation directly

↳ largest magnitude

With directional derivatives (compass gradients) we use many filters (usually 8)

$$G(j,k) = \text{MAX}\{\,|G_1(j,k)|, |G_2(j,k)|, \ldots, |G_8(j,k)|\,\}$$

Edge Detection 2

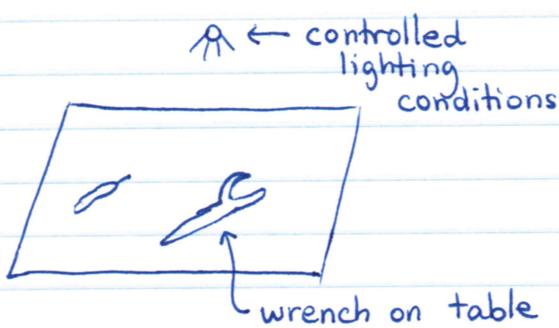


G_1: $\frac{1}{3}\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$ G_2: $\frac{1}{3}\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$ G: SQR, MAX, SUM

Threshold Selection

Noise-free images: threshold selected such that all amplitude discontinuities above certain amount are detected as edges.

Noisy images: Trade-off between missing valid edges and creating noise-induced false edges

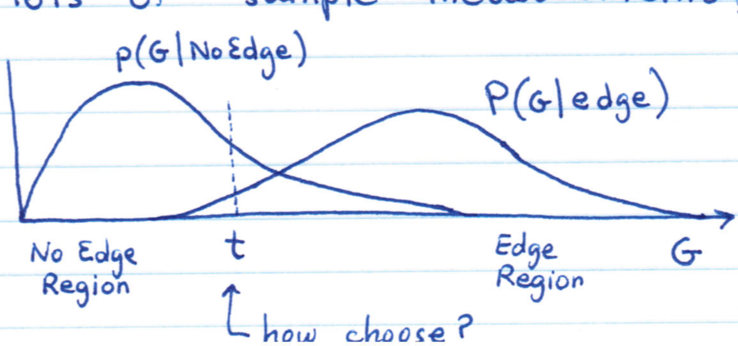


When do we have known types of objects & controlled lighting?

- Automated manufacturing
- Agricultural inspection
- Industrial inspection
- Bank check processing
- Mail sorting, ...

Known: $P(\text{edge}) \approx .05$ $P(\text{non-edge}) \approx .95$

Take lots of sample measurements, get:



← known

Everything above threshold t gets called an edge:

$$p(\text{pixel is called an edge}) = \int_t^{\infty} p(G)\, dG$$

p(called an edge | is an edge)

$$p(\text{correct detection}) = \int_t^{\infty} p(G \mid \text{edge})\, dG = P_D$$

p(called an edge | not edge)

$$p(\text{false detection}) = \int_t^{\infty} p(G \mid \text{no-edge})\, dG = P_F$$

Now, take a new image
At a given pixel, compute G
Want to assign it to most likely class

Call it an edge if

$$p(\text{edge} \mid G) \gtrless p(\text{no-edge} \mid G)$$

Bayes theorem says:

$$p(\text{edge} \mid G) = \frac{p(G \mid \text{edge})\, P(\text{edge})}{P(G)}$$

Here $p(G) = p(G \mid \text{edge})\, P(\text{edge}) + p(G \mid \text{no-edge})\, p(\text{no-edge})$
Substituting:

$$p(G \mid \text{edge})\, P(\text{edge}) \gtrless P(G \mid \text{no-edge})\, P(\text{no-edge})$$

$$\frac{p(G \mid \text{edge})}{p(G \mid \text{no-edge})} \gtrless \frac{p(\text{no-edge})}{p(\text{edge})}$$

known as the Maximum Likelihood Ratio Test

The probability of edge misclassification error is

$$P_E = (1 - P_D)\, p(\text{edge}) + P_F\, p(\text{no-edge})$$

The max likelihood ratio decision rule minimizes this probability.

This rule considers the 2 types of errors to be equally important (false alarms & false dismissals)
Suppose they're not ?

Receiver Operating Characteristic (ROC) curves

10,000 pixels total

True Binary Edge Map

1000 true edge pixels

9000 true non-edge pixels

Candidate Binary Edge Map

Found 1200 points

1200 < 800 correct, 400 not

True Positive Rate (TPR)

$$\text{TPR} = \Pr(\text{called edge} \mid \text{is edge}) = \frac{\Pr(\text{called edge and is edge})}{\Pr(\text{edge})}$$

$$\Pr(\text{called edge and is edge}) = \frac{\#\text{ candidate edge pix which are true ed}}{\text{total }\#\text{ pixels in image}} \rightarrow \frac{800}{10{,}00}$$

$$\Pr(\text{edge}) = \frac{\#\text{ true edge pixels}}{\text{total }\#\text{ pixels in image}} = \frac{1000}{10{,}000}$$

$$\Rightarrow \text{TPR} = \frac{\#\text{ candidate edge pixels which are true edge pix}}{\#\text{ true edge pixels}}$$

$$= \frac{800}{1000}$$

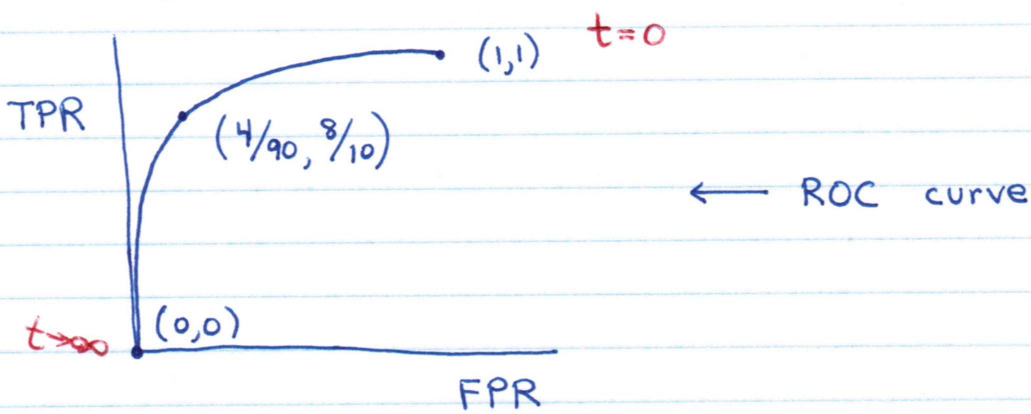
TPR is the same thing as P_D defined earlier

$$P_D = \int_t^{\infty} p(G \mid \text{edge})\, dG$$

Similarly, False Positive Rate (FPR) is P_F

$$\begin{aligned} FPR &= Pr(\text{called edge} \mid \text{not an edge}) \\ &= \frac{Pr(\text{called an edge and not an edge})}{Pr(\text{not an edge})} \\ &= \frac{\#\text{ candidate edge pixels which are NOT true edge pixels}}{\text{total }\#\text{ true non-edge pixels}} \\ &= \frac{400}{9000} \end{aligned}$$

We can plot TPR vs. FPR :



Suppose $t \to \infty$

Nothing gets declared an edge pixel

\# candidate edge pixels $\to 0$

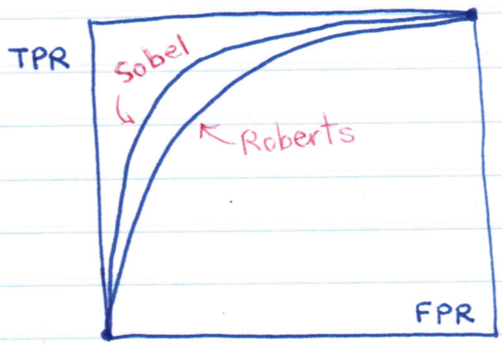
TPR $\to 0$ FPR $\to 0$

Suppose $t \to 0$ All pixels declared edge pixels

\# correct candidate edge pix $\to$ \# true edge pix $\Rightarrow$ TPR $\to 1$

\# incorrect " " " $\to$ \# true non-edge pix $\Rightarrow$ FPR $\to 1$

Suppose we have:

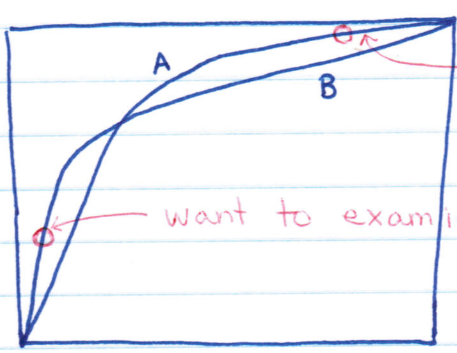


Curves don't cross.

- For a fixed acceptable FPR, can get higher (better) TPR using Sobel
- For a fixed acceptable TPR, can get lower (better) FPR using Sobel

→ Sobel is better, regardless of threshold

Consider instead: curves cross

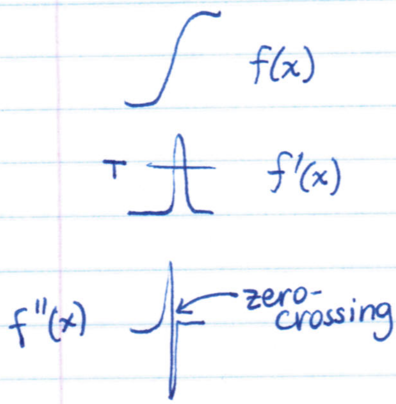


Medical imaging – want high TPR, highlight everything for the radiologist

want to examine very few pixels (only a few widgets, only need a few starting seeds to find edges)

Curves cross, can't say one is universally better. Depends where you need to operate.

Second derivative operators:



In 2 dimensions:

$$\sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \quad \text{isotropic}$$

$$\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad \text{isotropic}$$

Laplacian operator

Discrete approximation?

D	C	B	A
k-2	k-1	↑ k	k+1

$$\text{first difference}(k) = F(k) - F(k-1) = B - C$$

What happens if we define 2nd diff as being the 1st diff of 1st diff?

$$\text{2nd diff}(k) = \text{1st diff}(k) - \text{1st diff}(k-1)$$

$$= F(k) - F(k-1) - [F(k-1) - F(k-2)]$$

$$= F(k) + F(k-2) - 2F(k-1)$$

$$= B + D - 2C$$

This is centered on C! Wrong place!
So we define:

$$\text{2nd diff}(k) = \text{1st diff}(k+1) - \text{1st diff}(k)$$

$$= F(k+1) - F(k) - [F(k) - F(k-1)]$$

$$= F(k+1) + F(k-1) - 2F(k)$$

$$= A + C - 2B$$

filter $\begin{bmatrix} 1 & -2 & 1 \end{bmatrix}$ OR $\begin{bmatrix} -1 & 2 & -1 \end{bmatrix}$

2nd derivative operators

In 2 dimensions:

$$H = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 2 & -1 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & -1 & 0 \\ 0 & 2 & 0 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

digital Laplacian

Often normalized to provide unit gain average of positive and negative

$$H = \frac{1}{4}\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

← zero response to constant value and to ramp

~~This array is~~ 8-neighbor Laplacian:

$$H_2 = \frac{1}{8}\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

← not separable

A separable 8-neighbor Laplacian is

$$H_3 = \frac{1}{8}\begin{bmatrix} -2 & 1 & -2 \\ 1 & 4 & 1 \\ -2 & 1 & -2 \end{bmatrix}$$

comes from $\begin{bmatrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{bmatrix} + \begin{bmatrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{bmatrix}$