Filtering

An audio recording is a 1-dimensional **signal**: *amplitude(t)*.

A picture is a 2-dimensional **signal**: *intensity(x,y)* or *color(x,y)*.

Signals can be **continuous** (analog) or **discrete** (digital).

We will most often talk about images that are discrete both in space (**sampled** in *x,y*) and in intensity (**quantized** in value); i.e. raster images.

Filtering is a process that takes a signal as input and produces a signal as output.

Linear, Shift-Invariant Filters

Many useful filters are **linear, shift-invariant filters.**
In 1-D, these have the form:

$$b[x] = \sum_{t=-\infty}^{+\infty} a[t]h[x-t]$$

where $a[x]$ = input signal
$b[x]$ = output signal
$h[x]$ = filter
x takes on only integer values

This operation between sequences a and h is called **convolution**, and is written **b = a⊗h** for short. Note that convolution is commutative, i.e. **a⊗h=h⊗a**.

2-D is similar, but with a double-summation:

$$b[x,y] = \sum_{u=-\infty}^{+\infty} \sum_{v=-\infty}^{+\infty} a[u,v]h[x-u,y-v]$$

Blurring Filters

A simple blurring effect can be achieved with a 3x3 filter centered around a pixel:

$$\frac{1}{9}\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

More blurring is achieved with a wider $n \times n$ filter:

$$\frac{1}{n^2}\begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{pmatrix}$$

Derivatives of an Image

Discrete approximations to the derivative:

horizontal 1st derivative:

$$\frac{\partial}{\partial x} \Rightarrow \begin{pmatrix} -1 & 1 \end{pmatrix} \quad \text{or} \quad \frac{1}{2}\begin{pmatrix} -1 & 0 & 1 \end{pmatrix}$$

vertical 1st derivative:

$$\frac{\partial}{\partial y} \Rightarrow \begin{pmatrix} 1 \\ -1 \end{pmatrix} \quad \text{or} \quad \frac{1}{2}\begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}$$

2nd derivative (**Laplacian**):

$$\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \Rightarrow \begin{pmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{pmatrix}$$

Edge Filter

To find edges, use approximation to the **magnitude of the gradient** of the image.
Gradient and its magnitude:

$$\nabla a = \begin{pmatrix} \frac{\partial a}{\partial x} & \frac{\partial a}{\partial y} \end{pmatrix}, \qquad |\nabla a| = \sqrt{\left(\frac{\partial a}{\partial x}\right)^2 + \left(\frac{\partial a}{\partial y}\right)^2}$$

Sobel edge filter uses these weights:

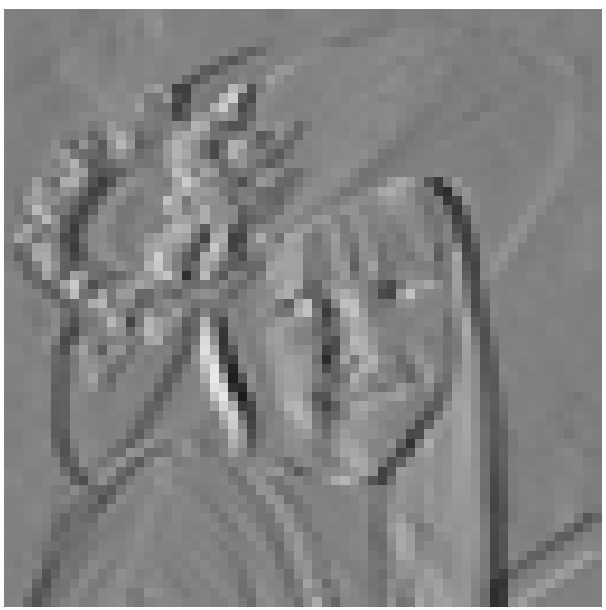
$$\frac{\partial}{\partial x} \Rightarrow \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}, \qquad \frac{\partial}{\partial y} \Rightarrow \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}$$

This is a nonlinear filter because of the sqrt and square operations.

original, 64x64 pixels

3x3 blur

5x5 blur



horizontal derivative

vertical derivative

Laplacian


sharpened (original + c*Lap.)


edge filter