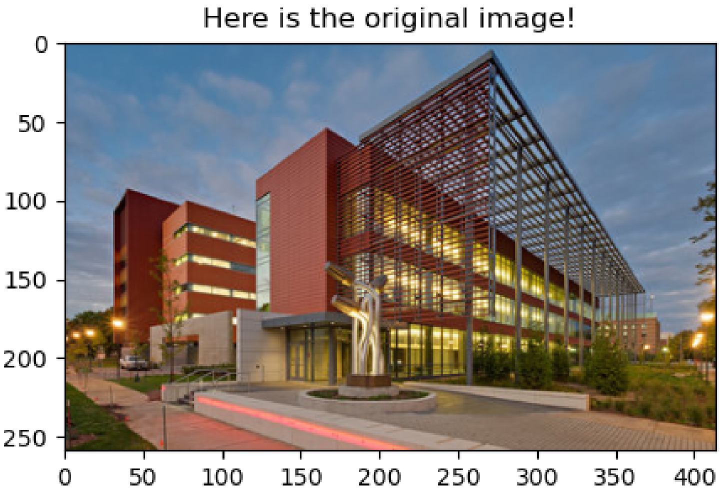
# Lecture 2: Linearity and Shift-Invariance

Mark Hasegawa-Johnson


ECE 401: Signal and Image Analysis

# Outline

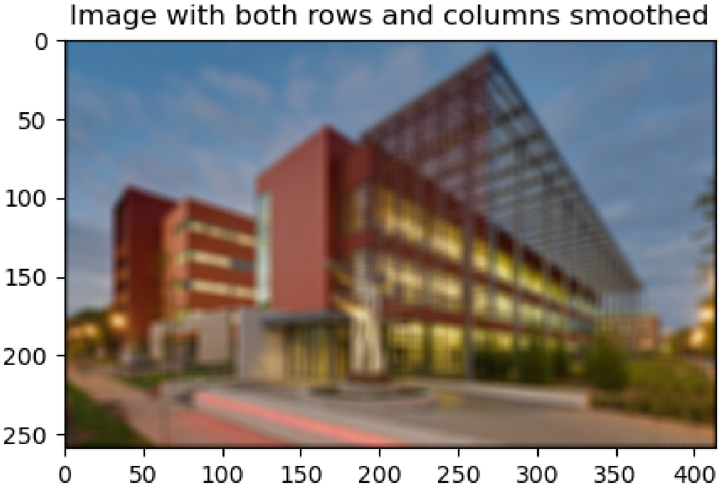
# How do you treat an image as a signal?



# How do you treat an image as a signal?

- An RGB image is a signal in three dimensions: $f[i,j,k] =$ intensity of the signal in the $i^{\text{th}}$ row, $j^{\text{th}}$ column, and $k^{\text{th}}$ color.
- $f[i,j,k]$, for each $(i,j,k)$, is either stored as an integer or a floating point number:
  - Floating point: usually $x \in [0,1]$, so $x = 0$ means dark, $x = 1$ means bright.
  - Integer: usually $x \in \{0, \ldots, 255\}$, so $x = 0$ means dark, $x = 255$ means bright.
- The three color planes are usually:
  - $k = 0$: Red
  - $k = 1$: Blue
  - $k = 2$: Green

# Local averaging



# Local averaging

- "Local averaging" means that we create an output image, $y[i,j,k]$, each of whose pixels is an **average** of nearby pixels in $f[i,j,k]$.
- For example, if we average along the rows:

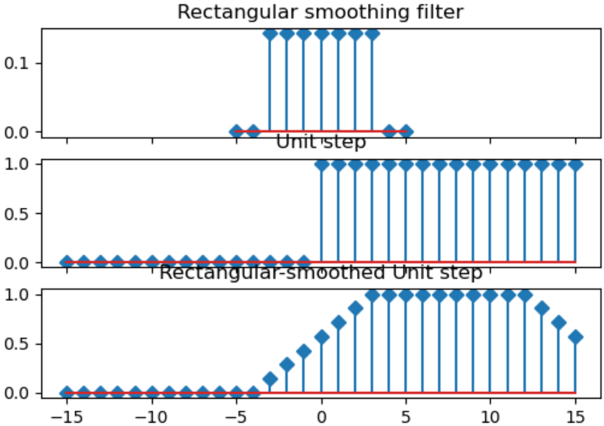
$$y[i,j,k] = \frac{1}{2M+1} \sum_{j'=j-M}^{j+M} f[i,j',k]$$

- If we average along the columns:

$$y[i,j,k] = \frac{1}{2M+1} \sum_{i'=i-M}^{i+M} f[i',j,k]$$

# Local averaging of a unit step

The top row are the averaging weights. If it's a 7-sample local average, $(2M+1)=7$, so the averaging weights are each $\frac{1}{2M+1}=\frac{1}{7}$. The middle row shows the input, $f[n]$. The bottom row shows the output, $y[n]$.



# Outline

# Weighted local averaging

- Suppose we don't want the edges quite so abrupt. We could do that using "weighted local averaging:" each pixel of $y[i,j,k]$ is a **weighted average** of nearby pixels in $f[i,j,k]$, with some averaging weights $g[n]$.
- For example, if we average along the rows:

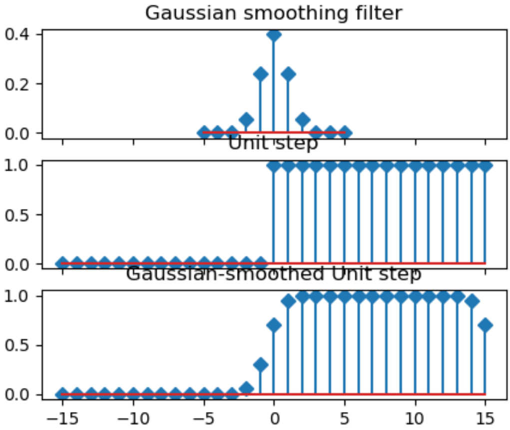
$$y[i,j,k] = \sum_{m=j-M}^{j+M} g[j-m]f[i,m,k]$$

- If we average along the columns:

$$y[i,j,k] = \sum_{i'=i-M}^{i+M} g[i-m]f[m,j,k]$$

# Weighted local averaging of a unit step

The top row are the averaging weights, $g[n]$. The middle row shows the input, $f[n]$. The bottom row shows the output, $y[n]$.



# Outline

# What is a System?

A **system** is anything that takes one signal as input, and generates another signal as output. We can write

$$x[n] \xrightarrow{\mathcal{H}} y[n]$$

which means

$$x[n] \longrightarrow \boxed{\mathcal{H}} \longrightarrow y[n]$$

# Example: Averager

For example, a weighted local averager is a system. Let's call it system $\mathcal{A}$.

$$x[n] \xrightarrow{\mathcal{A}} y[n] = \sum_{m=0}^{6} g[m]x[n-m]$$

# Example: Time-Shift

A time-shift is a system. Let's call it system $\mathcal{T}$.

$$x[n] \xrightarrow{\mathcal{T}} y[n] = x[n-1]$$

# Example: Square

If you calculate the square of a signal, that's also a system. Let's call it system $\mathcal{S}$:

$$x[n] \xrightarrow{\mathcal{S}} y[n] = x^2[n]$$

# Example: Add a Constant

If you add a constant to a signal, that's also a system. Let's call it system $\mathcal{C}$:

$$x[n] \xrightarrow{\mathcal{C}} y[n] = x[n] + 1$$

# Example: Window

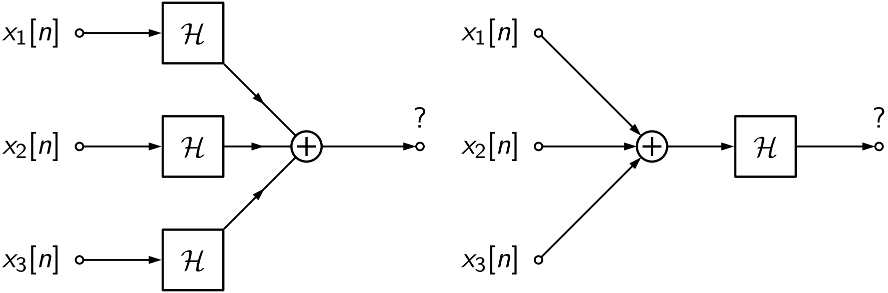
If you chop off all elements of a signal that are before time 0 or after time $N-1$ (for example, because you want to put it into an image), that is a system:

$$x[n] \xrightarrow{\mathcal{W}} y[n] = \begin{cases} x[n] & 0 \le n \le N-1 \\ 0 & \text{otherwise} \end{cases}$$

# Outline

# Linearity

A system is **linear** if these two algorithms compute the same thing:



# Linearity

A system $\mathcal{H}$ is said to be **linear** if and only if, for any $x_1[n]$ and $x_2[n]$,

$$x_1[n] \xrightarrow{\mathcal{H}} y_1[n]$$
$$x_2[n] \xrightarrow{\mathcal{H}} y_2[n]$$

implies that

$$x[n] = x_1[n] + x_2[n] \xrightarrow{\mathcal{H}} y[n] = y_1[n] + y_2[n]$$

In words: a system is **linear** if and only if, for every pair of inputs $x_1[n]$ and $x_2[n]$, (1) adding the inputs and then passing them through the system gives exactly the same effect as (2) passing both inputs through the system, and **then** adding them.

# Special case of linearity: Scaling

Notice, a special case of linearity is the case when $x_1[n] = x_2[n]$:

$$x_1[n] \xrightarrow{\mathcal{H}} y_1[n]$$
$$x_1[n] \xrightarrow{\mathcal{H}} y_1[n]$$

implies that

$$x[n] = 2x_1[n] \xrightarrow{\mathcal{H}} y[n] = 2y_1[n]$$

So if a system is linear, then **scaling the input** also **scales the output**.

# Example: Averager

Let's try it with the weighted averager.

$$x_1[n] \xrightarrow{\mathcal{A}} y_1[n] = \sum_{m=0}^{6} g[m]x_1[n-m]$$

$$x_2[n] \xrightarrow{\mathcal{A}} y_2[n] = \sum_{m=0}^{6} g[m]x_2[n-m]$$

Then:

$$\begin{aligned} x[n] = x_1[n] + x_2[n] &= \sum_{m=0}^{6} g[m]\left(x_1[n-m] + x_2[n-m]\right) \\ &= \left(\sum_{m=0}^{6} g[m]x_1[n-m]\right) + \left(\sum_{m=0}^{6} g[m]x_2[n-m]\right) \\ &= y_1[n] + y_2[n] \end{aligned}$$

. . . so a weighted averager is a **linear system**.

# Example: Square

A squarer is just obviously nonlinear, right? Let's see if that's true:

$$x_1[n] \xrightarrow{\mathcal{S}} y_1[n] = x_1^2[n]$$
$$x_2[n] \xrightarrow{\mathcal{S}} y_2[n] = x_2^2[n]$$

Then:

$$\begin{aligned} x[n] = x_1[n] + x_2[n] \xrightarrow{\mathcal{A}} y[n] &= x^2[n] \\ &= (x_1[n] + x_2[n])^2 \\ &= x_1^2[n] + 2x_1[n]x_2[n] + x_2^2[n] \\ &\neq y_1[n] + y_2[n] \end{aligned}$$

. . . so a squarer is a **nonlinear system**.

# Example: Add a Constant

This one is tricky. Adding a constant seems like it ought to be linear, but it's actually **nonlinear**. A system that would be linear except for the added constant is called an **affine** system.

$$x_1[n] \xrightarrow{\mathcal{C}} y_1[n] = x_1[n] + 1$$
$$x_2[n] \xrightarrow{\mathcal{C}} y_2[n] = x_2[n] + 1$$

Then:

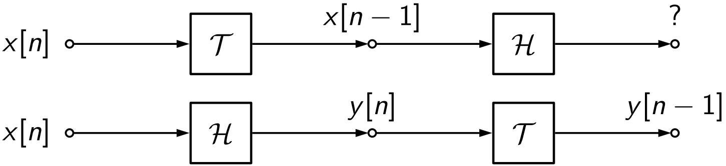
$$\begin{aligned} x[n] = x_1[n] + x_2[n] \xrightarrow{\mathcal{A}} y[n] &= x[n] + 1 \\ &= x_1[n] + x_2[n] + 1 \\ &\neq y_1[n] + y_2[n] \end{aligned}$$

. . . so adding a constant is a **nonlinear system**.

# Outline

# Time Invariance

A system $\mathcal{H}$ is **time-invariant** if these two algorithms compute the same thing (here $\mathcal{T}$ means "time shift"):



# Shift Invariance

**Shift-invariance** generalizes **time-invariance** to cases where the index variable, $n$, might mean something other than time (for example, it might mean spatial location, or row of a matrix, or whatever). A system $\mathcal{H}$ is said to be **shift-invariant** if and only if, for every $x_1[n]$,

$$x_1[n] \stackrel{\mathcal{H}}{\longrightarrow} y_1[n]$$

implies that

$$x[n] = x_1[n-n_0] \stackrel{\mathcal{H}}{\longrightarrow} y[n] = y_1[n-n_0]$$

In words: a system is **shift-invariant** if and only if, for any input $x_1[n]$, (1) shifting the input by some number of samples $n_0$, and then passing it through the system, gives exactly the same result as (2) passing it through the system, and then shifting it.

# Example: Averager

Let's try it with the weighted averager.

$$x_1[n] \xrightarrow{\mathcal{A}} y_1[n] = \sum_{m=0}^{6} g[m]x_1[n-m]$$

Then:

$$\begin{aligned}
x[n] = x_1[n-n_0] \xrightarrow{\mathcal{A}} y[n] &= \sum_{m=0}^{6} g[m]x[n-m] \\
&= \sum_{m=0}^{6} g[m]x_1\left[(n-m)-n_0\right] \\
&= \sum_{m=0}^{6} g[m]x_1\left[(n-n_0)-m\right] \\
&= y_1[n-n_0]
\end{aligned}$$

...so a weighted averager is a **shift-invariant system**.

# Example: Square

Squaring the input is a nonlinear operation, but is it shift-invariant? Let's find out:

$$x_1[n] \xrightarrow{\mathcal{S}} y_1[n] = x_1^2[n]$$

Then:

$$\begin{aligned} x[n] = x_1[n-n_0] \xrightarrow{\mathcal{A}} y[n] &= x^2[n] \\ &= (x_1[n-n_0])^2 \\ &= x_1^2[n-n_0] \\ &= y_1[n-n_0] \end{aligned}$$

...so computing the square is a **shift-invariant system**.

# Example: Windowing

How about windowing, e.g., in order to create an image?

$$x_1[n] \xrightarrow{\mathcal{W}} y_1[n] = \begin{cases} x_1[n] & 0 \le n \le N-1 \\ 0 & \text{otherwise} \end{cases}$$

If we shift the **output**, we get

$$y_1[n-n_0] = \begin{cases} x_1[n-n_0] & n_0 \le n \le N-1+n_0 \\ 0 & \text{otherwise} \end{cases}$$

...but if we shift the **input** ($x[n] = x_1[n-n_0]$), we get

$$y[n] = \begin{cases} x[n] & 0 \le n \le N-1 \\ 0 & \text{otherwise} \end{cases} = \begin{cases} x_1[n-n_0] & 0 \le n \le N-1 \\ 0 & \text{otherwise} \end{cases}$$
$$\ne y_1[n-n_0]$$

...so windowing is a **shift-varying system** (not shift-invariant).

# Quiz

Go to the course web page, try the quiz!

# Outline

# Summary

- A system is **linear** if and only if, for any two inputs $x_1[n]$ and $x_2[n]$ that produce outputs $y_1[n]$ and $y_2[n]$,

$$x[n] = x_1[n] + x_2[n] \xrightarrow{\mathcal{H}} y[n] = y_1[n] + y_2[n]$$

- A system is **shift-invariant** if and only if, for any input $x_1[n]$ that produces output $y_1[n]$,

$$x[n] = x_1[n - n_0] \xrightarrow{\mathcal{H}} y[n] = y_1[n - n_0]$$