Convolution and Inverse Filtering

\[
F[k_x, k_y] = \frac{1}{N_x N_y} \sum_{n_x=0}^{N_x-1} \sum_{n_y=0}^{N_y-1} f[n_x, n_y] e^{-j\left(\frac{2\pi k_x}{N_x} n_x + \frac{2\pi k_y}{N_y} n_y\right)}
\]

\[
f[n_x, n_y] = \sum_{k_x=0}^{N_x-1} \sum_{k_y=0}^{N_y-1} F[k_x, k_y] e^{j\left(\frac{2\pi k_x}{N_x} n_x + \frac{2\pi k_y}{N_y} n_y\right)}
\]

Applications of the 2D DFT in convolution and inverse filtering.

April 30, 2020
Implementing Convolution with the DFT (1D)

One of the most important applications of the DFT is in computing the responses of image processing systems – i.e., in computing convolutions.

- If a system is linear and time invariant, then its response to any input \( x[n] \) is \( (x * h)[n] \) where \( h[n] \) is the unit-sample response of the system.

\[
\begin{align*}
  x[n] \quad &\xrightarrow{\text{DTFT}}\quad X(\Omega) \\
  h[n] \quad &\xrightarrow{\text{DTFT}}\quad H(\Omega) \\
\end{align*}
\]

\[
(h * x)[n] \quad \iff \quad H(\Omega)X(\Omega)
\]

- **Convolution** follows directly from linearity and time-invariance.

Convolution can be implemented in the frequency domain.

\[
\begin{align*}
  x[n] \quad &\xrightarrow{\text{DFT}}\quad X[k] \\
  h[n] \quad &\xrightarrow{\text{DFT}}\quad H[k] \\
\end{align*}
\]

\[
(h \otimes x)[n] \quad \iff \quad H[k]X[k]
\]

Using the DFT speeds computation, but makes convolution “circular.”
Comparing Regular and Circular Convolution (1D)

Convolve $h[n]$ with $x[n]$ given below.

\[(h \ast x)[n] = \sum_{m=-\infty}^{\infty} h[m]x[n-m] = x[n] + x[n-1] + x[n-2]\]
Comparing Regular and Circular Convolution (1D)

Convolve $h[n]$ with $x[n]$ given below.

\[
(h \ast x)[n] = \sum_{m=-\infty}^{\infty} h[m]x[n-m] = x[n] + x[n-1] + x[n-2]
\]
Comparing Regular and Circular Convolution (1D)

Convolve $h[n]$ with $x[n]$ given below.

\[
(h \ast x)[n] = \sum_{m=-\infty}^{\infty} h[m]x[n - m] = x[n] + x[n-1] + x[n-2]
\]
Comparing Regular and Circular Convolution (1D)

Convolve $h[n]$ with $x[n]$ given below.

\[(h * x)[n] = \sum_{m=-\infty}^{\infty} h[m]x[n - m] = x[n] + x[n-1] + x[n-2]\]
Comparing Regular and Circular Convolution (1D)

Convolve $h[n]$ with $x[n]$ given below.

\[(h \ast x)[n] = \sum_{m=-\infty}^{\infty} h[m]x[n - m] = x[n] + x[n-1] + x[n-2] \]
Comparing Regular and Circular Convolution (1D)

Convolve $h[n]$ with $x[n]$ given below.

\[
(h * x)[n] = \sum_{m=-\infty}^{\infty} h[m]x[n - m] = x[n] + x[n-1] + x[n-2]
\]
Comparing Regular and Circular Convolution (1D)

Convolve $h[n]$ with $x[n]$ given below.

\[(h \ast x)[n] = \sum_{m=-\infty}^{\infty} h[m]x[n - m] = x[n] + x[n-1] + x[n-2]\]
Comparing Regular and Circular Convolution (1D)

Convolve $h[n]$ with $x[n]$ given below.

\[
(h \ast x)[n] = \sum_{m=-\infty}^{\infty} h[m]x[n - m] = x[n] + x[n-1] + x[n-2]
\]
Comparing Regular and Circular Convolution (1D)

Convolve \( h[n] \) with \( x[n] \) given below.

\[
(h \ast x)[n] = \sum_{m=-\infty}^{\infty} h[m]x[n - m] = x[n] + x[n-1] + x[n-2]
\]
Comparing Regular and Circular Convolution (1D)

Convolve $h[n]$ with $x[n]$ given below.

\[(h \ast x)[n] = \sum_{m=-\infty}^{\infty} h[m]x[n-m] = x[n] + x[n-1] + x[n-2]\]
Comparing Regular and Circular Convolution (1D)

Convolve $h[n]$ with $x[n]$ given below.

$$(h * x)[n] = \sum_{m=-\infty}^{\infty} h[m]x[n - m] = x[n] + x[n-1] + x[n-2]$$
Comparing Regular and Circular Convolution (1D)

Convolve $h[n]$ with $x[n]$ given below.

$$\sum_{m=-\infty}^{\infty} h[m]x[n-m] = x[n] + x[n-1] + x[n-2]$$
Comparing Regular and Circular Convolution (1D)

Convolve $h[n]$ with $x[n]$ given below.

\[
(h * x)[n] = \sum_{m=-\infty}^{\infty} h[m]x[n - m] = x[n] + x[n-1] + x[n-2]
\]
Comparing Regular and Circular Convolution (1D)

Convolve $h[n]$ with $x[n]$ given below.

\[
(h \ast x)[n] = \sum_{m=-\infty}^{\infty} h[m] x[n - m] = x[n] + x[n-1] + x[n-2]
\]
Comparing Regular and Circular Convolution (1D)

Convolve $h[n]$ with $x[n]$ given below.

\[
(h \ast x)[n] = \sum_{m=-\infty}^{\infty} h[m]x[n - m] = x[n] + x[n-1] + x[n-2]
\]
Comparing Regular and Circular Convolution (1D)

Convolve $h[n]$ with $x[n]$ given below.

\[(h \ast x)[n] = \sum_{m=-\infty}^{\infty} h[m]x[n - m] = x[n] + x[n-1] + x[n-2]\]
Comparing Regular and Circular Convolution (1D)

Find the circular convolution of $h[n]$ with $x[n]$ using DFT (length $N = 5$).

Circular convolution of $h[n]$ with $x[n]$ is equivalent to conventional convolution of $h[n]$ with a periodically extended version of $x[n]$.

$$(h \odot x)[n] = (h \ast x_p)[n]$$

where $x_p[n] = \sum_{m=-\infty}^{\infty} x[n + mN]$
Comparing Regular and Circular Convolution (1D)

Find the circular convolution of $h[n]$ with $x[n]$ using DFT (length $N = 5$).

Circular convolution of $h[n]$ with $x[n]$ is equivalent to conventional convolution of $h[n]$ with a periodically extended version of $x[n]$.

$$(h \odot x)[n] = (h \ast x_p)[n] \quad \text{where} \quad x_p[n] = \sum_{m=-\infty}^{\infty} x[n + mN]$$
Comparing Regular and Circular Convolution (1D)

Find the circular convolution of $h[n]$ with $x[n]$ using DFT (length $N = 5$).

Circular convolution of $h[n]$ with $x[n]$ is equivalent to conventional convolution of $h[n]$ with a periodically extended version of $x[n]$.

$$(h \odot x)[n] = (h * x_p)[n] \quad \text{where} \quad x_p[n] = \sum_{m=-\infty}^{\infty} x[n + mN]$$

$$(h \odot x)[n] = (h * x_p)[n]$$
Comparing Regular and Circular Convolution (1D)

Find the circular convolution of $h[n]$ with $x[n]$ using DFT (length $N = 5$).

Circular convolution of $h[n]$ with $x[n]$ is equivalent to conventional convolution of $h[n]$ with a periodically extended version of $x[n]$.

$$(h \circledast x)[n] = (h * x_p)[n]$$

where

$$x_p[n] = \sum_{m=-\infty}^{\infty} x[n + mN]$$
Comparing Regular and Circular Convolution (1D)

Find the circular convolution of $h[n]$ with $x[n]$ using DFT (length $N = 5$).

Circular convolution of $h[n]$ with $x[n]$ is equivalent to conventional convolution of $h[n]$ with a periodically extended version of $x[n]$.

$$ (h \odot x)[n] = (h \ast x_p)[n] \quad \text{where} \quad x_p[n] = \sum_{m=-\infty}^{\infty} x[n + mN] $$
Comparing Regular and Circular Convolution (1D)

Find the circular convolution of $h[n]$ with $x[n]$ using DFT (length $N = 5$).

Circular convolution of $h[n]$ with $x[n]$ is equivalent to conventional convolution of $h[n]$ with a periodically extended version of $x[n]$.

$$(h \otimes x)[n] = (h \ast x_p)[n] \quad \text{where} \quad x_p[n] = \sum_{m=-\infty}^{\infty} x[n + mN]$$

$$(h \otimes x)[n] = (h \ast x_p)[n]$$
Comparing Regular and Circular Convolution (1D)

Find the circular convolution of $h[n]$ with $x[n]$ using DFT (length $N = 5$).

Circular convolution of $h[n]$ with $x[n]$ is equivalent to conventional convolution of $h[n]$ with a periodically extended version of $x[n]$.

$$(h \odot x)[n] = (h \ast x_p)[n] \quad \text{where} \quad x_p[n] = \sum_{m=\infty}^{\infty} x[n + mN]$$
Comparing Regular and Circular Convolution (1D)

Find the circular convolution of $h[n]$ with $x[n]$ using DFT (length $N = 5$).

Circular convolution of $h[n]$ with $x[n]$ is equivalent to conventional convolution of $h[n]$ with a periodically extended version of $x[n]$.

\[
(h \ast x)[n] = (h \ast x_p)[n] \quad \text{where} \quad x_p[n] = \sum_{m=-\infty}^{\infty} x[n + mN]
\]
Comparing Regular and Circular Convolution (1D)

Find the circular convolution of $h[n]$ with $x[n]$ using DFT (length $N = 5$).

Circular convolution of $h[n]$ with $x[n]$ is equivalent to conventional convolution of $h[n]$ with a periodically extended version of $x[n]$.

\[
(h \odot x)[n] = (h \ast x_p)[n] \quad \text{where} \quad x_p[n] = \sum_{m=-\infty}^{\infty} x[n + mN]
\]
Comparing Regular and Circular Convolution (1D)

Find the circular convolution of $h[n]$ with $x[n]$ using DFT (length $N = 5$).

Circular convolution of $h[n]$ with $x[n]$ is equivalent to conventional convolution of $h[n]$ with a periodically extended version of $x[n]$.

$$ (h \otimes x)[n] = (h \ast x_p)[n] \quad \text{where} \quad x_p[n] = \sum_{m=-\infty}^{\infty} x[n + mN] $$
Comparing Regular and Circular Convolution (1D)

Find the circular convolution of $h[n]$ with $x[n]$ using DFT (length $N = 5$).

Circular convolution of $h[n]$ with $x[n]$ is equivalent to conventional convolution of $h[n]$ with a periodically extended version of $x[n]$.

\[(h \star x)[n] = (h \ast x_p)[n]\] where \[x_p[n] = \sum_{m=\infty}^{\infty} x[n + mN]\]
Comparing Regular and Circular Convolution (1D)

Find the circular convolution of $h[n]$ with $x[n]$ using DFT (length $N = 5$).

Circular convolution of $h[n]$ with $x[n]$ is equivalent to conventional convolution of $h[n]$ with a periodically extended version of $x[n]$.

$$ (h \ast x)[n] = (h \ast x_p)[n] \quad \text{where} \quad x_p[n] = \sum_{m=-\infty}^{\infty} x[n + mN] $$
Comparing Regular and Circular Convolution (1D)

Find the circular convolution of $h[n]$ with $x[n]$ using DFT (length $N = 5$).

Circular convolution of $h[n]$ with $x[n]$ is equivalent to conventional convolution of $h[n]$ with a periodically extended version of $x[n]$.

$$(h \ast x)[n] = (h \ast x_p)[n] \quad \text{where} \quad x_p[n] = \sum_{m=-\infty}^{\infty} x[n + mN]$$
Comparing Regular and Circular Convolution (1D)

The result of circular convolution $h[n]$ and $x[n]$ is an aliased version of the conventional convolution of $h[n]$ and $x[n]$

$$(h \circledast x)[n] = \sum_{m=-\infty}^{\infty} (h \ast x)[n+mN]$$

where $N$ is the length of the DFT analysis window (here $N = 5$).
2D Convolution

2D convolution is similar, but both input and unit-sample response are 2D. If a system is linear and shift-invariant, its response to input $f[r, c]$ is a superposition of shifted and scaled versions of unit-sample response $h[r, c]$.

$$(h \ast f)[r, c] = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} h[m, n] f[m - r, n - c]$$

$$= \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f[m, n] h[m - r, n - c]$$
2D Convolution

Graphical representation of 2D convolution.
Comparison of Conventional and Circular 2D Convolution

**Conventional convolution:** convolve in space or implement with DTFT

\[
\begin{array}{ccc}
\times & * & = \\
\begin{array}{c}
. \\
. \\
. \\
\hline
O \bar{I} \\
. \\
\end{array} & \begin{array}{c}
\cdot \\
\end{array} & \begin{array}{c}
\cdot \\
\cdot \\
\cdot \\
\cdot \\
\end{array}
\end{array}
\]

**Circular convolution:** implement with DFT

\[
\begin{array}{ccc}
\otimes & \odot & = \\
\begin{array}{c}
. \\
. \\
. \\
\hline
O \bar{I} \\
. \\
\end{array} & \begin{array}{c}
\cdot \\
\end{array} & \begin{array}{c}
\cdot \\
\cdot \\
\cdot \\
\cdot \\
\end{array}
\end{array}
\]

Circular convolution wraps vertically, horizontally, and diagonally.
Comparison of Conventional and Circular 2D Convolution

Circular convolution in 2D is equivalent to conventional 2D convolution with a periodically extended input.
Comparison of Conventional and Circular 2D Convolution

Circular convolution in 2D is also equivalent to aliasing the result of conventional convolution.

\[(h \bigcircledast x)[r, c] = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} (h \ast x)[r + mR, c + nC]\]
Comparison of Conventional and Circular 2D Convolution

The output of conventional convolution can be bigger than the input, while that of circular convolution aliases to the same size as the input.
Filtering in 2D

Last time, we looked at a few examples of 2D filtering or real images.

We used DFTs to lowpass filter and highpass filter this image.
2D Filtering

Lowpass filtered.
High-Pass Filtering

Highpass filtered.
An Interesting Optical Illusion

A *hybrid image* is created by combining the low frequencies from one image (left) with the high frequencies of another (right).
Hybrid Image

Look at this image with your eyes about a foot away from the screen. Then look again from a distance of six feet. Any difference?
Filtering and Inverse Filtering

An important area of research in image processing is in **inverse filtering**, (also called deconvolution). The idea is to undo the effect of prior filtering.

\[
\begin{align*}
  f[r, c] & \rightarrow h[r, c] \rightarrow g[r, c] \rightarrow h_i[r, c] \rightarrow \hat{f}[r, c]
\end{align*}
\]

We have already seen examples, as in telescopy (Hubble Space Telescope).

- \(f[r, c]\) represents the unknown image of a distant galaxy and
- \(h[r, c]\) represents distortions in the optics of the telescope.

Goal: design an inverse filter \(h_i[r, c]\) so that \(\hat{f}[r, c]\) approximates \(f[r, c]\).
Inverse Filtering

One simple approach is to filter by the inverse of $H[k_r, k_c]$.

\[
\begin{align*}
\hat{F}[k_r, k_c] &= H_i[k_r, k_c] \times G[k_r, k_c] = H_i[k_r, k_c] \times (H[k_r, k_c] \times F[k_r, k_c]) \\
\end{align*}
\]

If $H_i[k_r, k_c] \times H[k_r, k_c] = 1$ then $\hat{F}[k_r, k_c] = F[k_r, k_c]$!

Letting $H_i[k_r, k_c] = \frac{1}{H[k_r, k_c]}$ is called inverse filtering.

Of course, this only works if $H[k_r, k_c] \neq 0$ for any $k_r, k_c$. 
Motion Blur

Camera images are blurred by motion of the target.
The resulting motion blur can be modelled as the convolution.
Modelling Motion Blur

Assume that streaks in this image resulted from the blurring. There is an isolated streak near the point $r=120, c=250$ (approximate 19x6 pixels).
Inverse Filtering

Make an image $h$ to represent the presumed blurring function.

Let $H$ represent the DFT of $h$, and filter the blurred image with $\frac{1}{H}$. 
Inverse Filtering

Here is the resulting inverse filtered image – not at all what we want.

What went wrong?
Inverse Filtering

This image shows the magnitude of $H$ (DFT of blur function).
Inverse Filtering

This image shows the magnitude of \( \frac{1}{H} \).

What causes the bright spots? Why are they a problem?
Deblurring

The bright spots in $\frac{1}{H}$ come from points in $H$ with values near zero.

\[
\begin{align*}
X \rightarrow H \rightarrow Y \rightarrow G = \frac{1}{H} \rightarrow \hat{X}
\end{align*}
\]

Such bright spots dominate the result. Try limiting their magnitudes.

**Method 1:**
Start with $G = \frac{1}{H}$, but limit the magnitude of every point in $G$ to 4:

```python
for kr in range(R):
    for kc in range(C):
        G[kr,kc] = 1/H[kr,kc]
        if abs(G[kr,kc]) > 4:
            G[kr,kc] *= 4/abs(G[kr,kc])
```
Deblurring

This deblurring filter works better: easy to read license number.

But there are many artifacts.
Deblurring

The form of the previous deblurring function is a bit arbitrary.

\[ H G = \frac{1}{H} X Y \hat{X} \]

Method 2:
Here is a frequently used alternative (a “Weiner filter”):

\[ G = \frac{1}{H} \frac{|H|^2}{|H|^2 + C} \]

where \( C = 0.004 \) (chosen by trial and error).
Deblurring

Alternative deblurring function.

But there are still artifacts.
**Edge Effects**

Much of the ringing results from circular convolution. Window edges in original image to reduce step change due to periodic extension.
Comparison

Method 1 with and without windowing.
Comparison

Method 2 with and without windowing.
Conclusions

In general, inverse filtering worked well. It allowed a clear view of the license plate which was otherwise not legible.

**Problems with inverse filtering.** Inverting $H[k_r, k_c]$ doesn’t work well if $H[k_r, k_c]$ is near zero. Fortunately, there were only a few such points. Arbitrarily limiting the values of such points results in useful deblurring.

**Problems with circular convolution.** Circular convolution introduces enormous artifacts if the left and right (or top and bottom) edges differ in brightness. These artifacts can be reduced by windowing.

**Remaining problems.** The resulting images still suffer from ringing – presumably because of sharp discontinuities in the frequency representation of blurring.
Summary

Today we looked at a number of examples of how the DFT can be used to compute responses to image processing systems.

The enormous advantage of the DFT is in **computation speed**. However, the convolutions that result using DFT methods are **circular**.

An important area of research in image processing is in **inverse filtering**. We looked as an example of inverse filtering to reduce **motion blur**. The method was ultimately limited by Gibb’s phenomenon combined with discontinuities introduced by circular convolution.