Lecture 5: Image Filtering (continued)

(unless otherwise stated slides are taken or adopted from Bob Woodham, Jim Little, Leon Sigal and Fred Tung)
Menu for Today (January 20, 2021)

Topics:
— Gaussian and Pillbox filters
— Separability
— The Convolution Theorem
— Fourier space representations

Readings:
— Today’s Lecture: none
— Next Lecture: [Optional] Forsyth & Ponce (2nd ed.) 4.4

Assignment 1: Image Filtering and Hybrid Images due January 29

We will have our first quiz sometime next week (on Canvas).

Format: Quiz is 1-2 minute per question (total time < 10 min).

Can be started within 24 hour window.
Today’s “fun” Example: Rolling Shutter
Today’s “fun” Example: Rolling Shutter

Rolling shutter effect
Lecture 4: Re-cap

— The correlation of \( F(X, Y) \) and \( I(X, Y) \) is:

\[
I'(X, Y) = \sum_{i=-k}^{k} \sum_{j=-k}^{k} F(I, J) I(X + i, Y + j)
\]

— Visual interpretation: Superimpose the filter \( F \) on the image \( I \) at \((X, Y)\), perform an element-wise multiply, and sum up the values

— Convolution is like correlation except filter “flipped”
  if \( F(X, Y) = F(-X, -Y) \) then correlation = convolution.
Lecture 4: Re-cap

Ways to handle boundaries

- **Ignore/discard.** Make the computation undefined for top/bottom $k$ rows and left/right-most $k$ columns
- **Pad with zeros.** Return zero whenever a value of $I$ is required beyond the image bounds
- **Assume periodicity.** Top row wraps around to the bottom row; leftmost column wraps around to rightmost column.
- **Reflect at boundary.** Left pixel is found to right. Up to down, and vice versa

Simple examples of filtering:
- copy, shift, smoothing, sharpening

Linear filter properties:
- superposition, scaling, shift invariance

Characterization Theorem: Any linear, shift-invariant operation can be expressed as a convolution
Example 5: Smoothing with a Box Filter

Gonzales & Woods (3rd ed.) Figure 3.3
Smoothing

Smoothing with a box \textit{doesn’t model lens defocus} well

— Smoothing with a box filter depends on direction

— Image in which the center point is 1 and every other point is 0
Smoothing

Smoothing with a box **doesn’t model lens defocus** well
— Smoothing with a box filter depends on direction
— Image in which the center point is 1 and every other point is 0

Smoothing with a (circular) **pillbox** is a better model for defocus (in geometric optics)

The **Gaussian** is a good general smoothing model
— for phenomena (that are the sum of other small effects)
— whenever the Central Limit Theorem applies
Example 6: Smoothing with a Gaussian

Idea: Weight contributions of pixels by spatial proximity (nearness)

2D Gaussian (continuous case):

\[ G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp\left( -\frac{x^2 + y^2}{2\sigma^2} \right) \]

Forsyth & Ponce (2nd ed.)
Figure 4.2
Example 6: Smoothing with a Gaussian

**Idea:** Weight contributions of pixels by spatial proximity (nearness)

2D **Gaussian** (continuous case):

\[
G_{\sigma}(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)
\]

Forsyth & Ponce (2nd ed.)

Figure 4.2
**Example 6: Smoothing with a Gaussian**

Quantized a truncated **3x3 Gaussian** filter:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_\sigma(-1, 1)$</td>
<td>$G_\sigma(0, 1)$</td>
<td>$G_\sigma(1, 1)$</td>
</tr>
<tr>
<td>$G_\sigma(-1, 0)$</td>
<td></td>
<td>$G_\sigma(0, 0)$</td>
</tr>
<tr>
<td>$G_\sigma(-1, -1)$</td>
<td>$G_\sigma(0, -1)$</td>
<td></td>
</tr>
</tbody>
</table>
Example 6: Smoothing with a Gaussian

Quantized a truncated 3x3 Gaussian filter:

<table>
<thead>
<tr>
<th>$G_\sigma(-1, 1) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}}$</th>
<th>$G_\sigma(0, 1) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}}$</th>
<th>$G_\sigma(1, 1) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_\sigma(-1, 0) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}}$</td>
<td>$G_\sigma(0, 0) = \frac{1}{2\pi\sigma^2}$</td>
<td>$G_\sigma(1, 0) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}}$</td>
</tr>
<tr>
<td>$G_\sigma(-1, -1) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}}$</td>
<td>$G_\sigma(0, -1) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}}$</td>
<td>$G_\sigma(1, -1) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}}$</td>
</tr>
</tbody>
</table>
Example 6: Smoothing with a Gaussian

Quantized and truncated 3x3 Gaussian filter:

\[
\begin{align*}
G_\sigma(-1, 1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}} \\
G_\sigma(0, 1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} \\
G_\sigma(1, 1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}} \\
G_\sigma(-1, 0) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} \\
G_\sigma(0, 0) &= \frac{1}{2\pi\sigma^2} \\
G_\sigma(1, 0) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} \\
G_\sigma(-1, -1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}} \\
G_\sigma(0, -1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} \\
G_\sigma(1, -1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}}
\end{align*}
\]

With \( \sigma = 1 \):

\[
\begin{array}{ccc}
0.059 & 0.097 & 0.059 \\
0.097 & 0.159 & 0.097 \\
0.059 & 0.097 & 0.059 \\
\end{array}
\]
Example 6: Smoothing with a Gaussian

Quantized and truncated 3x3 Gaussian filter:

<table>
<thead>
<tr>
<th>$G_{\sigma}(-1, 1)$</th>
<th>$G_{\sigma}(0, 1)$</th>
<th>$G_{\sigma}(1, 1)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}}$</td>
<td>$\frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}}$</td>
<td>$\frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$G_{\sigma}(-1, 0)$</th>
<th>$G_{\sigma}(0, 0)$</th>
<th>$G_{\sigma}(1, 0)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}}$</td>
<td>$\frac{1}{2\pi\sigma^2}$</td>
<td>$\frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$G_{\sigma}(-1, -1)$</th>
<th>$G_{\sigma}(0, -1)$</th>
<th>$G_{\sigma}(1, -1)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}}$</td>
<td>$\frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}}$</td>
<td>$\frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}}$</td>
</tr>
</tbody>
</table>

With $\sigma = 1$:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.059</td>
<td>0.097</td>
<td>0.059</td>
</tr>
<tr>
<td>0.097</td>
<td>0.159</td>
<td>0.097</td>
</tr>
<tr>
<td>0.059</td>
<td>0.097</td>
<td>0.059</td>
</tr>
</tbody>
</table>

What happens if $\sigma$ is larger?
Example 6: Smoothing with a Gaussian

Quantized an truncated \textbf{3x3 Gaussian} filter:

\[
\begin{align*}
G_\sigma(-1, 1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}} \\
G_\sigma(0, 1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} \\
G_\sigma(1, 1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}} \\
G_\sigma(-1, 0) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} \\
G_\sigma(0, 0) &= \frac{1}{2\pi\sigma^2} \\
G_\sigma(1, 0) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} \\
G_\sigma(-1, -1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}} \\
G_\sigma(0, -1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} \\
G_\sigma(1, -1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}}
\end{align*}
\]

With $\sigma = 1$:

\[
\begin{array}{c|c|c}
\uparrow & \uparrow & \uparrow \\
\uparrow & \downarrow & \uparrow \\
\uparrow & \uparrow & \uparrow \\
\end{array}
\]

What happens if $\sigma$ is larger?

— More blur
### Example 6: Smoothing with a Gaussian

Quantized and truncated 3x3 Gaussian filter:

\[
\begin{align*}
G_\sigma(-1, 1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} & G_\sigma(0, 1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} & G_\sigma(1, 1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} \\
G_\sigma(-1, 0) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} & G_\sigma(0, 0) &= \frac{1}{2\pi\sigma^2} & G_\sigma(1, 0) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} \\
G_\sigma(-1, -1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} & G_\sigma(0, -1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} & G_\sigma(1, -1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}}
\end{align*}
\]

With \( \sigma = 1 \):

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.059</td>
<td>0.097</td>
<td>0.059</td>
</tr>
<tr>
<td>0.097</td>
<td>0.159</td>
<td>0.097</td>
</tr>
<tr>
<td>0.059</td>
<td>0.097</td>
<td>0.059</td>
</tr>
</tbody>
</table>

What happens if \( \sigma \) is larger?

What happens if \( \sigma \) is smaller?
Example 6: Smoothing with a Gaussian

Quantized an truncated 3x3 Gaussian filter:

<table>
<thead>
<tr>
<th>$G_\sigma(-1,1)$</th>
<th>$G_\sigma(0,1)$</th>
<th>$G_\sigma(1,1)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}}$</td>
<td>$\frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}}$</td>
<td>$\frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$G_\sigma(-1,0)$</th>
<th>$G_\sigma(0,0)$</th>
<th>$G_\sigma(1,0)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}}$</td>
<td>$\frac{1}{2\pi\sigma^2}$</td>
<td>$\frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$G_\sigma(-1,-1)$</th>
<th>$G_\sigma(0,-1)$</th>
<th>$G_\sigma(1,-1)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}}$</td>
<td>$\frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}}$</td>
<td>$\frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}}$</td>
</tr>
</tbody>
</table>

With $\sigma = 1$:

What happens if $\sigma$ is larger?

What happens if $\sigma$ is smaller?

— Less blur
Example 6: Smoothing with a Gaussian

Forsyth & Ponce (2nd ed.) Figure 4.1 (left and right)
Box vs. Gaussian Filter

original

7x7 Gaussian

7x7 box

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)
Fun: How to get shadow effect?

University of British Columbia

Adopted from: Ioannis (Yannis) Gkioulkas (CMU)
Fun: How to get shadow effect?

University of British Columbia

Blur with a Gaussian kernel, then compose the blurred image with the original (with some offset)

Adopted from: Ioannis (Yannis) Gkioulekas (CMU)
Example 6: Smoothing with a Gaussian

Quantized and truncated 3x3 Gaussian filter:

\[
\begin{array}{ccc}
G_\sigma(-1, 1) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} & G_\sigma(0, 1) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} & G_\sigma(1, 1) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} \\
G_\sigma(-1, 0) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} & G_\sigma(0, 0) = \frac{1}{2\pi\sigma^2} & G_\sigma(1, 0) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} \\
G_\sigma(-1, -1) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} & G_\sigma(0, -1) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} & G_\sigma(1, -1) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} \\
\end{array}
\]

With \( \sigma = 1 \):

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.059</td>
<td>0.097</td>
<td>0.059</td>
</tr>
<tr>
<td>0.097</td>
<td>0.159</td>
<td>0.097</td>
</tr>
<tr>
<td>0.059</td>
<td>0.097</td>
<td>0.059</td>
</tr>
</tbody>
</table>

What is the problem with this filter?
Example 6: Smoothing with a Gaussian

Quantized an truncated 3x3 Gaussian filter:

\[
\begin{align*}
G_\sigma(-1, 1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}} \\
G_\sigma(-1, 0) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} \\
G_\sigma(-1, -1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}}
\end{align*}
\]

\[
\begin{align*}
G_\sigma(0, 1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} \\
G_\sigma(0, 0) &= \frac{1}{2\pi\sigma^2} \\
G_\sigma(0, -1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} \\
G_\sigma(1, 1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}} \\
G_\sigma(1, 0) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{1}{2\sigma^2}} \\
G_\sigma(1, -1) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{2}{2\sigma^2}}
\end{align*}
\]

With \( \sigma = 1 \):

<table>
<thead>
<tr>
<th>( G_\sigma(-1, 1) )</th>
<th>( G_\sigma(-1, 0) )</th>
<th>( G_\sigma(-1, -1) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.059</td>
<td>0.097</td>
<td>0.059</td>
</tr>
<tr>
<td>0.097</td>
<td>0.159</td>
<td>0.097</td>
</tr>
<tr>
<td>0.059</td>
<td>0.097</td>
<td>0.059</td>
</tr>
</tbody>
</table>

What is the problem with this filter?

- does not sum to 1
- truncated too much
Gaussian: Area Under the Curve
Example 6: Smoothing with a Gaussian

With $\sigma = 1$:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.059</td>
<td>0.097</td>
<td>0.059</td>
</tr>
<tr>
<td>0.097</td>
<td>0.159</td>
<td>0.097</td>
</tr>
<tr>
<td>0.059</td>
<td>0.097</td>
<td>0.059</td>
</tr>
</tbody>
</table>

Better version of the Gaussian filter:

— sums to 1 (normalized)
— captures $\pm 2\sigma$

In general, you want the Gaussian filter to capture $\pm 3\sigma$ for $\sigma = 1 \Rightarrow 7\times7$ filter
Lets talk about efficiency
Efficient Implementation: **Separability**

A 2D function of $x$ and $y$ is **separable** if it can be written as the product of two functions, one a function only of $x$ and the other a function only of $y$.

Both the 2D box filter and the 2D Gaussian filter are separable.

Both can be implemented as two 1D convolutions:

— First, convolve each row with a 1D filter
— Then, convolve each column with a 1D filter
— Aside: or vice versa

The **2D Gaussian** is the only (non-trivial) 2D function that is both separable and rotationally invariant.
Separability: Box Filter Example

Standard (3x3)

\[
F(X, Y) = F(X)F(Y)
\]

filter

$\begin{bmatrix}
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
\end{bmatrix}$

$\begin{bmatrix}
0 & 10 & 20 & 30 & 30 & 30 & 20 & 10 \\
0 & 20 & 40 & 60 & 60 & 60 & 40 & 20 \\
0 & 30 & 50 & 80 & 80 & 90 & 60 & 30 \\
0 & 30 & 50 & 80 & 80 & 90 & 60 & 30 \\
0 & 20 & 30 & 50 & 50 & 60 & 40 & 20 \\
0 & 10 & 20 & 30 & 30 & 30 & 20 & 10 \\
10 & 10 & 10 & 10 & 0 & 0 & 0 & 0 \\
10 & 30 & 10 & 10 & 0 & 0 & 0 & 0 \\
\end{bmatrix}$
# Separability: Box Filter Example

<table>
<thead>
<tr>
<th>Standard (3x3)</th>
<th>$F(X, Y) = F(X)F(Y)$</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Standard Filter" /></td>
<td><img src="image" alt="Separable Filter" /></td>
</tr>
</tbody>
</table>

## I(X, Y)

<table>
<thead>
<tr>
<th>0 0 0 0 0 0 0 0 0 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>0 0 0 90 90 90 90 90 0 0</td>
</tr>
<tr>
<td>0 0 0 90 90 90 90 90 0 0</td>
</tr>
<tr>
<td>0 0 0 90 90 90 90 90 0 0</td>
</tr>
<tr>
<td>0 0 0 90 90 90 90 90 0 0</td>
</tr>
<tr>
<td>0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>0 0 90 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>0 0 0 0 0 0 0 0 0 0</td>
</tr>
</tbody>
</table>

## Separable

<table>
<thead>
<tr>
<th>0 0 0 0 0 0 0 0 0 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>0 0 0 90 90 90 90 90 0 0</td>
</tr>
<tr>
<td>0 0 0 90 90 90 90 90 0 0</td>
</tr>
<tr>
<td>0 0 0 90 90 90 90 90 0 0</td>
</tr>
<tr>
<td>0 0 0 90 90 90 90 90 0 0</td>
</tr>
<tr>
<td>0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>0 0 90 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>0 0 0 0 0 0 0 0 0 0</td>
</tr>
</tbody>
</table>

## F(X)

<table>
<thead>
<tr>
<th>0 0 0 0 0 0 0 0 0 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>0 0 0 30 60 90 90 90 60 30</td>
</tr>
<tr>
<td>0 0 0 30 60 90 90 90 60 30</td>
</tr>
<tr>
<td>0 0 0 30 60 90 90 90 60 30</td>
</tr>
<tr>
<td>0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>30 30 30 30 0 0 0 0</td>
</tr>
<tr>
<td>0 0 0 0 0 0 0 0 0</td>
</tr>
</tbody>
</table>

## $F(X, Y)$

<table>
<thead>
<tr>
<th>0 0 0 0 0 0 0 0 0 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>0 0 0 30 60 90 90 90 60 30</td>
</tr>
<tr>
<td>0 0 0 30 60 90 90 90 60 30</td>
</tr>
<tr>
<td>0 0 0 30 60 90 90 90 60 30</td>
</tr>
<tr>
<td>0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>30 30 30 30 0 0 0 0</td>
</tr>
<tr>
<td>0 0 0 0 0 0 0 0 0</td>
</tr>
</tbody>
</table>
Separability: Box Filter Example

\[ F(X, Y) = F(X)F(Y) \]

<table>
<thead>
<tr>
<th>Standard (3x3)</th>
<th>output</th>
<th>I'(X, Y)</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Standard Filter" /></td>
<td><img src="image" alt="Output Image" /></td>
<td><img src="image" alt="Output Image" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>image</th>
<th>I(X, Y)</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Image" /></td>
<td><img src="image" alt="Image" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Separable</th>
<th>F(X)</th>
<th>F(Y)</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Separable Filter" /></td>
<td><img src="image" alt="Separable Filter" /></td>
<td><img src="image" alt="Separable Filter" /></td>
</tr>
</tbody>
</table>
Separability: How do you know if filter is separable?

If a 2D filter can be expressed as an outer product of two 1D filters
Efficient Implementation: \textbf{Separability}

For example, recall the 2D \textbf{Gaussian}:

\[
G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)
\]

The 2D Gaussian can be expressed as a product of two functions, one a function of \(x\) and another a function of \(y\).
Efficient Implementation: **Separability**

For example, recall the 2D **Gaussian**:

\[
G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)
\]

\[
= \left(\frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{x^2}{2\sigma^2}\right)\right) \left(\frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{y^2}{2\sigma^2}\right)\right)
\]

function of \(x\)  function of \(y\)

The 2D Gaussian can be expressed as a product of two functions, one a function of \(x\) and another a function of \(y\)
Efficient Implementation: **Separability**

For example, recall the 2D **Gaussian**:

$$G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$$

$$= \left(\frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{x^2}{2\sigma^2}\right)\right) \left(\frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{y^2}{2\sigma^2}\right)\right)$$

function of $x$  
function of $y$

The 2D Gaussian can be expressed as a product of two functions, one a function of $x$ and another a function of $y$. In this case the two functions are (identical) 1D Gaussians
Efficient Implementation: **Separability**

Naive implementation of 2D **Gaussian**:

At each pixel, \((X, Y)\) there are \(m \times m\) multiplications

There are \(n \times n\) pixels in \((X, Y)\)

**Total:** \(m^2 \times n^2\) multiplications
Efficient Implementation: **Separability**

Naive implementation of 2D **Gaussian**:

At each pixel, \((X, Y)\), there are \(m \times m\) multiplications.

There are \(n \times n\) pixels in \((X, Y)\).

**Total:** \(m^2 \times n^2\) multiplications.

Separable 2D **Gaussian**:
Efficient Implementation: **Separability**

Naive implementation of 2D **Gaussian**:  
At each pixel, \((X, Y)\), there are \(m \times m\) multiplications  
There are \(n \times n\) pixels in \((X, Y)\)  
**Total:** \(m^2 \times n^2\) multiplications

Separable 2D **Gaussian**:  
At each pixel, \((X, Y)\), there are \(2m\) multiplications  
There are \(n \times n\) pixels in \((X, Y)\)  
**Total:** \(2m \times n^2\) multiplications
Example 7: Smoothing with a Pillbox

Let the radius (i.e., half diameter) of the filter be $r$.

In a continuous domain, a 2D (circular) pillbox filter, $f(x, y)$, is defined as:

$$f(x, y) = \frac{1}{\pi r^2} \begin{cases} 1 & \text{if } x^2 + y^2 \leq r^2 \\ 0 & \text{otherwise} \end{cases}$$

The scaling constant, $\frac{1}{\pi r^2}$, ensures that the area of the filter is one.
Example 7: Smoothing with a Pillbox

Recall that the 2D Gaussian is the only (non trivial) 2D function that is both separable and rotationally invariant.

A 2D pillbox is rotationally invariant but not separable.

There are occasions when we want to convolve an image with a 2D pillbox. Thus, it worth exploring possibilities for efficient implementation.
Example 7: Smoothing with a Pillbox

A 2D box filter can be expressed as the sum of a 2D pillbox and some “extra corner bits”
Example 7: Smoothing with a Pillbox

Therefore, a 2D pillbox filter can be expressed as the difference of a 2D box filter and those same “extra corner bits”
Example 7: Smoothing with a Pillbox

Implementing convolution with a 2D pillbox filter as the difference between convolution with a box filter and convolution with the “extra corner bits” filter allows us to take advantage of the separability of a box filter.

Further, we can postpone scaling the output to a single, final step so that convolution involves filters containing all 0’s and 1’s — This means the required convolutions can be implemented without any multiplication at all.
Example 7: Smoothing with a Pillbox

Original

11 x 11 Pillbox
Let \( z \) be the product of two numbers, \( x \) and \( y \), that is,

\[
z = xy
\]
Speeding Up **Convolution** (The Convolution Theorem)

Let $z$ be the product of two numbers, $x$ and $y$, that is,

$$z = xy$$

Taking logarithms of both sides, one obtains

$$\ln z = \ln x + \ln y$$
Speeding Up **Convolution** (The Convolution Theorem)

Let $z$ be the product of two numbers, $x$ and $y$, that is,

$$z = xy$$

Taking logarithms of both sides, one obtains

$$\ln z = \ln x + \ln y$$

Therefore.

$$z = \exp^{\ln z} = \exp^{(\ln x + \ln y)}$$
Speeding Up **Convolution** (The Convolution Theorem)

Let \( z \) be the product of two numbers, \( x \) and \( y \), that is,

\[
z = xy
\]

Taking logarithms of both sides, one obtains

\[
\ln z = \ln x + \ln y
\]

Therefore,

\[
z = \exp \ln z = \exp (\ln x \ln y)
\]

**Interpretation:** At the expense of two \( \ln() \) and one \( \exp() \) computations, multiplication is reduced to addition.
Speeding Up Rotation

Another analogy: **2D rotation of a point by an angle** $\alpha$ about the origin.

The standard approach, in Euclidean coordinates, involves a matrix multiplication:

$$
\begin{bmatrix}
  x' \\
  y'
\end{bmatrix} =
\begin{bmatrix}
  \cos \alpha & -\sin \alpha \\
  \sin \alpha & \cos \alpha
\end{bmatrix}
\begin{bmatrix}
  x \\
  y
\end{bmatrix}
$$

Suppose we transform to polar coordinates:

$$(x, y) \rightarrow (\rho, \theta) \rightarrow (\rho, \theta + \alpha) \rightarrow (x', y')$$

Rotation becomes addition, at expense of one polar coordinate transform and one inverse polar coordinate transform.
Speeding Up **Convolution** (The Convolution Theorem)

Similarly, some image processing operations become cheaper in a transform domain.

Gonzales & Woods (3rd ed.) Figure 2.39
Speeding Up **Convolution** (The Convolution Theorem)

**Convolution Theorem:**

Let $i'(x, y) = f(x, y) \otimes i(x, y)$

then $\mathcal{I}'(w_x, w_y) = \mathcal{F}(w_x, w_y) \mathcal{I}(w_x, w_y)$

where $\mathcal{I}'(w_x, w_y)$, $\mathcal{F}(w_x, w_y)$, and $\mathcal{I}(w_x, w_y)$ are Fourier transforms of $i'(x, y)$, $f(x, y)$ and $i(x, y)$

At the expense of two Fourier transforms and one inverse Fourier transform, convolution can be reduced to (complex) multiplication
Let's take a detour …
What follows is for fun
(you will **NOT** be tested on this)
Existential **Choice**
Fourier Transform (you will **NOT** be tested on this)

Basic building block:

\[ A \sin(\omega x + \phi) \]

Fourier’s claim: Add enough of these to get **any** periodic signal you want!

**Slide Credit:** Ioannis (Yannis) Gkioulekas (CMU)
Fourier Transform (you will NOT be tested on this)

Basic building block:

\[ A \sin(\omega x + \phi) \]

amplitude

sinusoid

angular frequency

variable

phase

Fourier’s claim: Add enough of these to get any periodic signal you want!

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)
Fourier Transform (you will NOT be tested on this)

How would you generate this function?

\[ = \, ? \, + \, ? \]
Fourier Transform (you will **NOT** be tested on this)

How would you generate this function?

\[ \text{sin}(2\pi x) = \text{?} \]
Fourier Transform (you will **NOT** be tested on this)

How would you generate this function?

\[
\begin{align*}
\text{function} &= \sin(2\pi x) + \frac{1}{3}\sin(2\pi 3x)
\end{align*}
\]
Fourier Transform (you will **NOT** be tested on this)

How would you generate this function?

\[ f(x) = \sin(2\pi x) + \frac{1}{3} \sin(2\pi 3x) = \sin(2\pi x) + \frac{1}{3} \sin(2\pi 3x) \]
Fourier Transform (you will **NOT** be tested on this)

How would you generate this function?

\[ \approx \]  

square wave

**Slide Credit**: Ioannis (Yannis) Gkioulekas (CMU)
Fourier Transform (you will **NOT** be tested on this)

How would you generate this function?
**Fourier Transform** (you will NOT be tested on this)

How would you generate this function?

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)
Fourier Transform (you will **NOT** be tested on this)

How would you generate this function?

slide credit: Ioannis (Yannis) Gkioulekas (CMU)
Fourier Transform (you will NOT be tested on this)

How would you generate this function?

\[ \text{square wave} \approx \text{Fourier Transform} \]

How would you express this mathematically?

**Slide Credit**: Ioannis (Yannis) Gkioulekas (CMU)
**Fourier Transform (you will NOT be tested on this)**

How would you generate this function?

\[
\text{square wave} = A \sum_{k=1}^{\infty} \frac{1}{k} \sin(2\pi k x)
\]

infinite sum of sine waves

*Slide Credit: Ioannis (Yannis) Gkioulkas (CMU)*
Fourier Transform (you will **NOT** be tested on this)

Basic building block:

\[ A \sin(\omega x + \phi) \]

Fourier’s claim: Add enough of these to get any periodic signal you want!
Fourier Transform (you will **NOT** be tested on this)

**Image from:** Numerical Simulation and Fractal Analysis of Mesoscopic Scale Failure in Shale Using Digital Images
What are “frequencies” in an image?

Spatial frequency

| $f = 4$ | $f = 5$ | $f = 6$ | $f = 7$ | $f = 8$ | $f = 9$ | $f = 10$ |
**Fourier Transform** (you will NOT be tested on this)

What are “frequencies” in an image?

**Amplitude** (magnitude) of Fourier transform (phase does not show desirable correlations with image structure)
Fourier Transform (you will NOT be tested on this)

What are “frequencies” in an image?

Amplitude (magnitude) of Fourier transform (phase does not show desirable correlations with image structure)

Observation: low frequencies close to the center
Fourier Transform (you will NOT be tested on this)

What are “frequencies” in an image?

Spatial frequency

Θ=30°

Θ=150°
What are “frequencies” in an image?
Fourier Transform (you will **NOT** be tested on this)

Fourier Transform (you will NOT be tested on this)

First (lowest) frequency, a.k.a. average

Fourier Transform (you will NOT be tested on this)

+ Second frequency

Fourier Transform (you will **NOT** be tested on this)

+ Third frequency

Fourier Transform (you will **NOT** be tested on this)

+ 50% of frequencies

Fourier Transform (you will NOT be tested on this)

Fourier Transform (you will NOT be tested on this)
**Fourier Transform** (you will NOT be tested on this)

Forsyth & Ponce (2nd ed.) Figure 4.6
Fourier Transform (you will NOT be tested on this)

Forsyth & Ponce (2nd ed.) Figure 4.6
Fourier Transform (you will **NOT** be tested on this)

Experiment: Where do you see the stripes?
**Fourier Transform** (you will **NOT** be tested on this)

The diagram shows the Campbell-Robson contrast sensitivity curve. It indicates that our eyes are sensitive to mid-range frequencies.
What preceded was for fun
(you will NOT be tested on it)
Fourier Transform

Preview of Part 3 of your homework

Gala Contemplating the Mediterranean Sea Which at Twenty Meters Becomes the Portrait of Abraham Lincoln (Homage to Rothko)

Salvador Dali, 1976

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)
Fourier Transform

Preview of Part 3 of your homework

Low-pass filtered version

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)
Fourier Transform

Preview of Part 3 of your homework

High-pass filtered version

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)
Aside: You will not be tested on this ...

Piazza: What is a low-pass filter?
Speeding Up **Convolution** (The Convolution Theorem)

**Convolution Theorem:**

Let \( i'(x, y) = f(x, y) \otimes i(x, y) \)

then \( \mathcal{I}'(w_x, w_y) = \mathcal{F}(w_x, w_y) \mathcal{I}(w_x, w_y) \)

where \( \mathcal{I}'(w_x, w_y) \mathcal{F}(w_x, w_y) \) and \( \mathcal{I}(w_x, w_y) \) are Fourier transforms of \( i'(x, y), f(x, y) \) and \( i(x, y) \)

At the expense of two **Fourier** transforms and one inverse Fourier transform, convolution can be reduced to (complex) multiplication
**Speeding Up Convolution** (The Convolution Theorem)

**General** implementation of convolution:

At each pixel, \((X, Y)\), there are \(m \times m\) multiplications.

There are \(n \times n\) pixels in \((X, Y)\).

**Total**: \(m^2 \times n^2\) multiplications.

**Convolution** in FFT space:

Cost of FFT/IFFT for image: \(O(n^2 \log n)\)

Cost of FFT/IFFT for filter: \(O(m^2 \log m)\)

Cost of convolution: \(O(n^2)\)