Lecture 8: Sampling 2

( unless otherwise stated slides are taken or adopted from Bob Woodham, Jim Little, Leon Sigal and Fred Tung )
Menu for Today (September 24)

Topics:
- Sampling (continued)
- Aliasing
- Color Filter Arrays
- Bayer patterns

Readings:
- Today's Lecture: Forsyth & Ponce (2nd ed.) 4.5
- Next Lecture: Forsyth & Ponce (2nd ed.) 4.6, 4.7

Reminders:
- Assignment 1: Image Filtering and Hybrid Images due September 27
Today’s “fun” Example: Optical Illusions

Hermann Grid Illusion
Today’s “fun” Example: Nudging

Aerial view of the white stripes at the lake shore drive in Chicago.
Lecture 7: Re-cap

In the continuous case, images are functions of two spatial variables, $x$ and $y$.

The discrete case is obtained from the continuous case via sampling (i.e. tessellation, quantization).

If a signal is **bandlimited** then it is possible to design a sampling strategy such that the sampled signal captures the underlying continuous signal exactly.
The challenge to intuition is the fact that music (in the 1D case) and images (in the 2D case) can be represented as linear combinations of individual sine waves of differing frequencies and phases (remember discussion on Fourier Transforms).

A fundamental result (**Sampling Theorem**) is:

For bandlimited signals, if you sample regularly at or above twice the maximum frequency (called the **Nyquist rate**), then you can reconstruct the original signal exactly.
**Sampling Theory (informal)**

Exact reconstruction requires constraint on the rate at which $i(x,y)$ can change between samples

— “rate of change” means derivative
— the formal concept is **bandlimited signal**
— “bandlimit” and “constraint on derivative” are linked

Think of music

— bandlimited if it has some maximum **temporal frequency**
— the upper limit of human hearing is about 20 kHz

Think of imaging systems. Resolving power is measured in

— “line pairs per mm” (for a bar test pattern)
— “cycles per mm” (for a sine wave test pattern)

An image is bandlimited if it has some maximum **spatial frequency**
**Example**: A Simple Sine Wave

How do we discretize the signal?

![Sine Wave Diagram]

*Slide Credit*: Ioannis (Yannis) Gkioulkas (CMU)
**Example:** A Simple Sine Wave

How do we discretize the signal?
Example: A Simple Sine Wave

How do we discretize the signal?

How many samples should I take?
Can I take as many samples as I want?

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)
Example: A Simple Sine Wave

How do we discretize the signal?

How many samples should I take?
Can I take as few samples as I want?

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)
**Example:** A Simple Sine Wave

How do we discretize the signal?

Signal can be confused with one at lower frequency

*Slide Credit:* Ioannis (Yannis) Gkioulekas (CMU)
**Example:** A Simple Sine Wave

How do we discretize the signal?

A signal can be confused with one at lower frequency

---

*Slide Credit:* Ioannis (Yannis) Gkioulekas (CMU)
**Example**: A Simple Sine Wave

How do we discretize the signal?

A signal can always be confused with one at higher frequency

*Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)*
Undersampling = Aliasing
The challenge to intuition is the fact that music (in the 1D case) and images (in the 2D case) can be represented as linear combinations of individual sine waves of differing frequencies and phases (remember discussion on FTs).

A fundamental result (Sampling Theorem) is:

For bandlimited signals, if you sample regularly at or above twice the maximum frequency (called the Nyquist rate), then you can reconstruct the original signal exactly.
Sampling Theory (informal)

Question: For a bandlimited signal, what if you oversample (i.e., sample at greater than the Nyquist rate)
**Sampling Theory (informal)**

**Question:** For a bandlimited signal, what if you **oversample** (i.e., sample at greater than the Nyquist rate)

**Answer:** Nothing bad happens! Samples are redundant and there are wasted bits
**Sampling Theory (informal)**

**Question:** For a bandlimited signal, what if you **oversample** (i.e., sample at greater than the Nyquist rate)

**Answer:** Nothing bad happens! Samples are redundant and there are wasted bits.

**Question:** For a bandlimited signal, what if you **undersample** (i.e., sample at less than the Nyquist rate)
Sampling Theory (informal)

**Question:** For a bandlimited signal, what if you **oversample** (i.e., sample at greater than the Nyquist rate)

**Answer:** Nothing bad happens! Samples are redundant and there are wasted bits.

**Question:** For a bandlimited signal, what if you **undersample** (i.e., sample at less than the Nyquist rate)

**Answer:** Two bad things happen! Things are missing (i.e., things that should be there aren’t). There are artifacts (i.e., things that shouldn’t be there are).
Sampling Theory (informal)

Forsyth & Ponce (2nd ed.) Figure 4.7
Sampling Theory (informal)

Forsyth & Ponce (2nd ed.) Figure 4.12
Reducing Aliasing Artifacts

1. **Oversampling** — sample more than you think you need and average (i.e., area sampling)
Aliasing

- aliasing artifacts
- anti-aliasing by oversampling

*Slide Credit:* Ioannis (Yannis) Gkioulekas (CMU)
Reducing Aliasing Artifacts

1. **Oversampling** — sample more than you think you need and average (i.e., area sampling)

2. **Smoothing** before sampling. Why?
**Aliasing** in Photographs

This is also known as “moire”

*Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)*
**Temporal Aliasing**

Imagine a spoked wheel moving to the right (rotating clockwise). Mark wheel with dot so we can see what’s happening.

If camera shutter is only open for a fraction of a frame time (frame time = 1/30 sec. for video, 1/24 sec. for film):

Without dot, wheel appears to be rotating slowly backwards! (counterclockwise)
Temporal Aliasing

Wagon wheel effect

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)
Temporal Aliasing

Slide Credit: Ioannis (Yannis) Gkioulkas (CMU)
Sometimes **undersampling** is unavoidable, and there is a trade-off between “things missing” and “artifacts.”

— **Medical imaging**: usually try to maximize information content, tolerate some artifacts

— **Computer graphics**: usually try to minimize artifacts, tolerate some information missing
— Images also can be considered a function of time. Then, we write $i(x, y, t)$ where $x$ and $y$ are spatial variables and $t$ is a temporal variable.

— To make the dependence of brightness on wavelength explicit, we can instead write $i(x, y, t, \lambda)$ where $x$, $y$ and $t$ are as above and where $\lambda$ is a spectral variable.

— More commonly, we think of “color” already as discrete and write

$$i_R(x, y)$$
$$i_G(x, y)$$
$$i_B(x, y)$$

for specific colour channels, R, G and B.
**Color** is an Artifact of Human Perception

“Color” is **not** an objective physical property of light (electromagnetic radiation). Instead, light is characterized by its wavelength.

What we call “color” is how we subjectively perceive a very small range of these wavelengths.

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)
Color Filter Arrays (CFA)

Slide Credit: Ioannis (Yannis) Gkioulkas (CMU)
Color Filter Arrays (CFA)
Color Filters

Two design choices:
— What spectral sensitivity functions $f(\lambda)$ to use for each color filter?
— How to spatially arrange (“mosaic”) different color filters?
Color Filters

Two design choices:
— What spectral sensitivity functions $f(\lambda)$ to use for each color filter?
— How to spatially arrange ("mosaic") different color filters?

Canon 50D

Generally do not match human sensitivity

$f(\lambda)$

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)
Color Filters

Two design choices:
— What spectral sensitivity functions $f(\lambda)$ to use for each color filter?
— How to spatially arrange (“mosaic”) different color filters?

Generally do not match human sensitivity

Slide Credit: Ioannis (Yannis) Gkioulakes (CMU)
Color Filters

Two design choices:
— What spectral sensitivity functions \( f(\lambda) \) to use for each color filter?
— How to spatially arrange ("mosaic") different color filters?

Canon 50D

Why more green pixels?

Generally do not match human sensitivity

\( f(\lambda) \)

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)
Different Color Filter Arrays (CFAs)

Finding the “best” CFA mosaic is an active research area.

How would you go about designing your own CFA? What criteria would you consider?

**Slide Credit:** Ioannis (Yannis) Gkioulekas (CMU)
Many Different Spectral Sensitivity Functions

Each camera has its more or less unique, and most of the time secret, SSF

Same scene captured using 3 different cameras with identical settings

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)
RAW Bayer Image

After all of this, what does an image look like?

lots of noise

mosaicing artifacts

— Kind of disappointing
— We call this the RAW image

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)
CFA Demosicing

Produce full RGB image from mosaiced sensor output

Any ideas on how to do this?
CFA Demosaicing

Produce full RGB image from mosaiced sensor output

Interpolate from neighbors:
— Bilinear interpolation (needs 4 neighbors)
— Bicubic interpolation (needs more neighbors, may overblur)
— Edge-aware interpolation

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)
Demosaicing by Bilinear Interpolation

Bilinear interpolation: Simply average your 4 neighbors.

\[
\begin{align*}
G_? &= \frac{G_1 + G_2 + G_3 + G_4}{4}
\end{align*}
\]

Neighborhood changes for different channels:

Slide Credit: Ioannis (Yannis) Gkioulkas (CMU)
The sequence of image processing operations applied by the camera’s image signal processor (ISP) to convert a RAW image into a “conventional” image.

**Image Processing Pipeline**

- **RAW image** (mosaiced, linear, 12-bit)
- **analog front-end**
- **denoising**
- **CFA demosaicing**
- **white balance**
- **color transforms**
- **tone reproduction**
- **compression**
- **final RGB image** (non-linear, 8-bit)

*Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)*
Summary

In the continuous case, images are functions of two spatial variables, x and y.

The discrete case is obtained from the continuous case via sampling (i.e. tessellation, quantization).

If a signal is bandlimited then it is possible to design a sampling strategy such that the sampled signal captures the underlying continuous signal exactly.

Adequate sampling may not always be practical. In such cases there is a trade-off between “things missing” and “artifacts”.
— Different applications make the trade-off differently
Lecture 8: Template Matching

( unless otherwise stated slides are taken or adopted from Bob Woodham, Jim Little, Leon Sigal and Fred Tung )

Image Credit: https://docs.adaptive-vision.com/4.7/studio/machine_vision_guide/TemplateMatching.html
How can we find a part of one image that matches another?

or,

How can we find instances of a pattern in an image?
Template Matching

How can we find a part of one image that matches another?

or,

How can we find instances of a pattern in an image?

**Key Idea:** Use the pattern as a template
Template Matching

Scene

Template (mask)

A toy example

Slide Credit: Kristen Grauman
**Template Matching**

We can think of convolution/correlation as comparing a template (the filter) with each local image patch.

— Consider the filter and image patch as vectors.

— Applying a filter at an image location can be interpreted as computing the dot product between the filter and the local image patch.
Template Matching

We can think of convolution/correlation as comparing a template (the filter) with each local image patch.

— Consider the filter and image patch as vectors.
— Applying a filter at an image location can be interpreted as computing the dot product between the filter and the local image patch.

Template

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
**Template Matching**

We can think of convolution/correlation as comparing a template (the filter) with each local image patch.

— Consider the filter and image patch as vectors.

— Applying a filter at an image location can be interpreted as computing the dot product between the filter and the local image patch.

<table>
<thead>
<tr>
<th>Image Patch 1</th>
<th>Template</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0</td>
<td>0 0 0</td>
</tr>
<tr>
<td>0 1 0</td>
<td>0 1 0</td>
</tr>
<tr>
<td>0 1 1</td>
<td>0 1 1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Image Patch 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 0 1</td>
</tr>
<tr>
<td>0 1 0</td>
</tr>
<tr>
<td>0 0 0</td>
</tr>
</tbody>
</table>
**Template Matching**

We can think of convolution/correlation as comparing a template (the filter) with each local image patch.

— Consider the filter and image patch as vectors.

— Applying a filter at an image location can be interpreted as computing the dot product between the filter and the local image patch.

<table>
<thead>
<tr>
<th>Image Patch 1</th>
<th>Template</th>
<th>Image Patch 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0</td>
<td>0 0 0</td>
<td>0 0 0</td>
</tr>
<tr>
<td>0 1 0</td>
<td>0 1 0</td>
<td>0 1 0</td>
</tr>
<tr>
<td>0 1 1</td>
<td>0 1 1</td>
<td>0 1 1</td>
</tr>
</tbody>
</table>

| 1 0 1         | 0 0 0    | 0 0 0         |
| 0 1 0         | 0 1 0    | 0 1 0         |
| 0 0 0         | 0 1 1    | 0 0 0         |
**Template Matching**

We can think of convolution/correlation as comparing a template (the filter) with each local image patch.

— Consider the filter and image patch as vectors.

— Applying a filter at an image location can be interpreted as computing the dot product between the filter and the local image patch.

\[
\begin{array}{ccc}
0 & 0 & 0 \\
1 & 0 & 0 \\
0 & 1 & 1 \\
\end{array}
\quad \quad \quad
\begin{array}{ccc}
0 & 0 & 0 \\
0 & 1 & 0 \\
0 & 1 & 1 \\
\end{array}
\quad \quad \quad
\begin{array}{ccc}
0 & 0 & 0 \\
0 & 1 & 0 \\
0 & 1 & 1 \\
\end{array}
\]

\[
= 3
\]

\[
\begin{array}{ccc}
1 & 0 & 1 \\
0 & 1 & 0 \\
0 & 0 & 0 \\
\end{array}
\quad \quad \quad
\begin{array}{ccc}
0 & 0 & 0 \\
0 & 1 & 0 \\
0 & 1 & 1 \\
\end{array}
\quad \quad \quad
\begin{array}{ccc}
0 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 0 \\
\end{array}
\]

\[
= 1
\]
**Template Matching**

We can think of convolution/correlation as comparing a template (the filter) with each local image patch.

— Consider the filter and image patch as vectors.

— Applying a filter at an image location can be interpreted as computing the dot product between the filter and the local image patch.

<table>
<thead>
<tr>
<th>Image Patch 1</th>
<th>Template</th>
<th>Image Patch 2</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0</td>
<td>0 0 0</td>
<td>1 0 1</td>
<td>3</td>
</tr>
<tr>
<td>0 1 0</td>
<td>0 1 0</td>
<td>0 1 0</td>
<td>1 × 255</td>
</tr>
<tr>
<td>0 1 1</td>
<td>0 1 1</td>
<td>0 0 0</td>
<td>0</td>
</tr>
</tbody>
</table>

\[
\begin{align*}
0 & 0 & 0 \\
0 & 1 & 0 \\
0 & 1 & 1 \\
\end{align*}
\times 255

\[
\begin{align*}
0 & 0 & 0 \\
0 & 1 & 0 \\
0 & 1 & 1 \\
\end{align*}
= 3
\]

\[
\begin{align*}
0 & 0 & 0 \\
0 & 1 & 0 \\
0 & 1 & 1 \\
\end{align*}
= 1 \times 255
\]
**Template Matching**

We can think of convolution/correlation as comparing a template (the filter) with each local image patch.

— Consider the filter and image patch as vectors.
— Applying a filter at an image location can be interpreted as computing the dot product between the filter and the local image patch.

The dot product may be large simply because the image region is bright.

We need to normalize the result in some way.
Template Matching

Let \( \mathbf{a} \) and \( \mathbf{b} \) be vectors. Let \( \theta \) be the angle between them. We know

\[
\cos \theta = \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{a}| |\mathbf{b}|} = \frac{\mathbf{a} \cdot \mathbf{b}}{\sqrt{(\mathbf{a} \cdot \mathbf{a})(\mathbf{b} \cdot \mathbf{b})}} = \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{a}| |\mathbf{b}|}
\]

where \( \cdot \) is dot product and \( | | \) is vector magnitude

Correlation is a dot product.

Correlation measures similarity between the filter and each local image region

**Normalized correlation** varies between \(-1\) and \(1\)

Normalized correlation attains the value \(1\) when the filter and image region are identical (up to a scale factor)

67
Template Matching

Detected template

Correlation map

Slide Credit: Kristen Grauman
**Template Matching**

**Linear filtering** the entire image computes the entire set of dot products, one for each possible alignment of the filter and the image.

Important Insight:

— filters look like the pattern they are intended to find
— filters find patterns they look like

Linear filtering is sometimes referred to as **template matching**