Image Processing & Vision

Lecture 03: Template Matching & Sampling

Hak Gu Kim

hakgukim@cau.ac.kr

Immersive Reality & Intelligent Systems Lab (IRIS LAB)

Graduate School of Advanced Imaging Science, Multimedia & Film (GSAIM)

Chung-Ang University (CAU)

20 Mar. 2023

Topics

- Template Matching
- Sampling Theory
- Color Filter Arrays

***Note:** Many of these slides in this course were adapted from Computer Vision at CMU (16-385) and UBC (CPSC425)

Topics

- Template Matching

- Sampling Theory

- Color Filter Arrays

***Note:** Many of these slides in this course were adapted from Computer Vision at CMU (16-385) and UBC (CPSC425)

Template Matching

- **Correlation** can be considered as a measurement for comparing a template (the filter) with each local image patch
  - Apply a filter at an image location can be interpreted as computing the dot product between the filter and the local image patch

0	0	0
0	1	0
0	1	1

Image Patch 1

1	0	1
0	1	0
0	0	0

Image Patch 2

0	0	0
0	1	0
0	1	1

Template

0	0	0
0	1	0
0	1	1

$= 3$

0	0	0
0	1	0
0	0	0

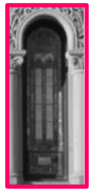
$= 1$

Template

Image

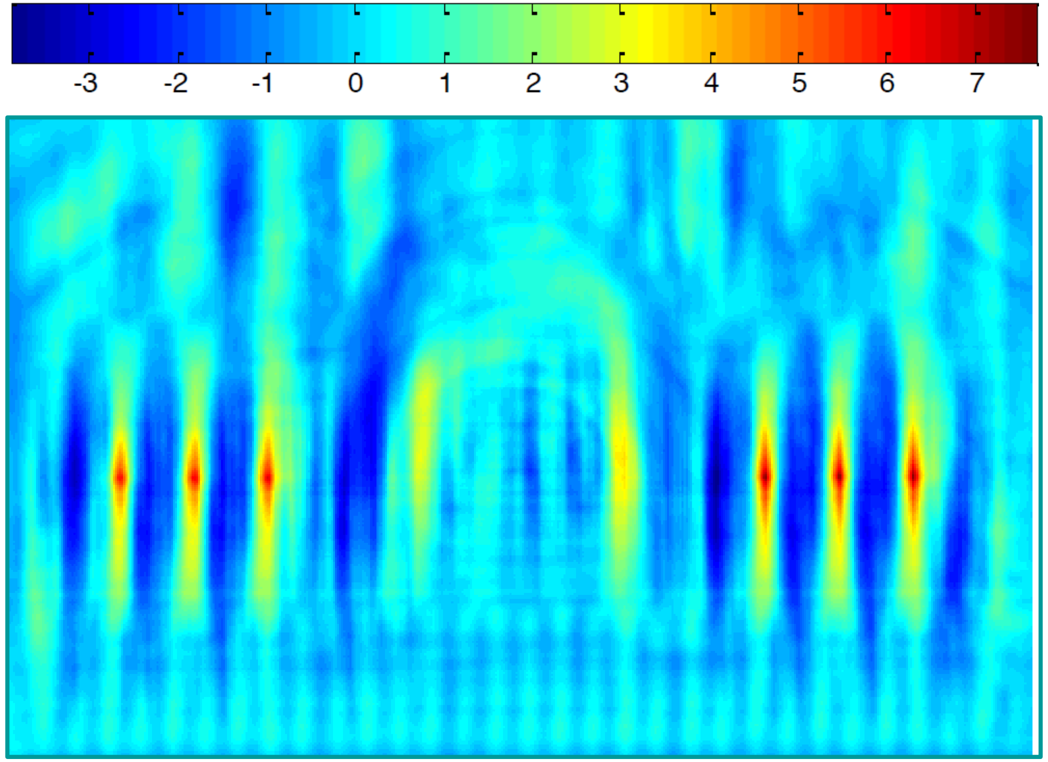
Template Matching: Graphical Understanding

- Template matching is a technique in digital image processing **for finding small parts of an image** which match a template image



Template

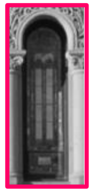
Image



Correlation map

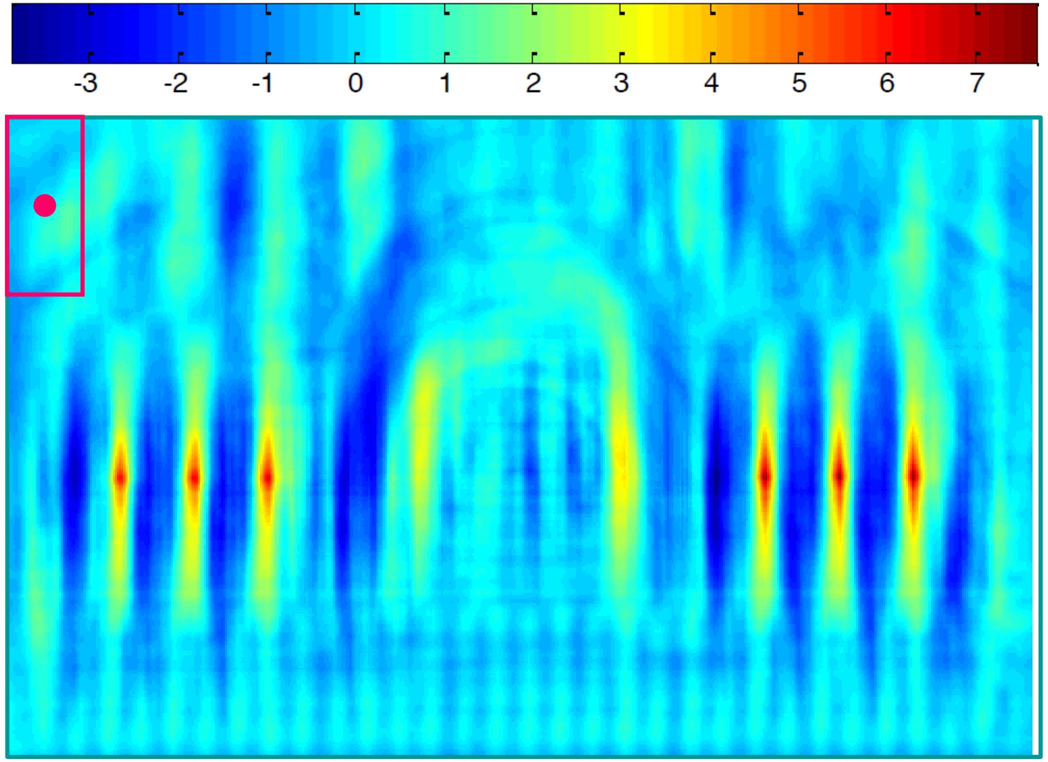
Template Matching: Graphical Understanding

- Template matching is a technique in digital image processing **for finding small parts of an image** which match a template image



Template

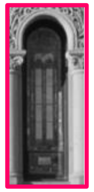
Image



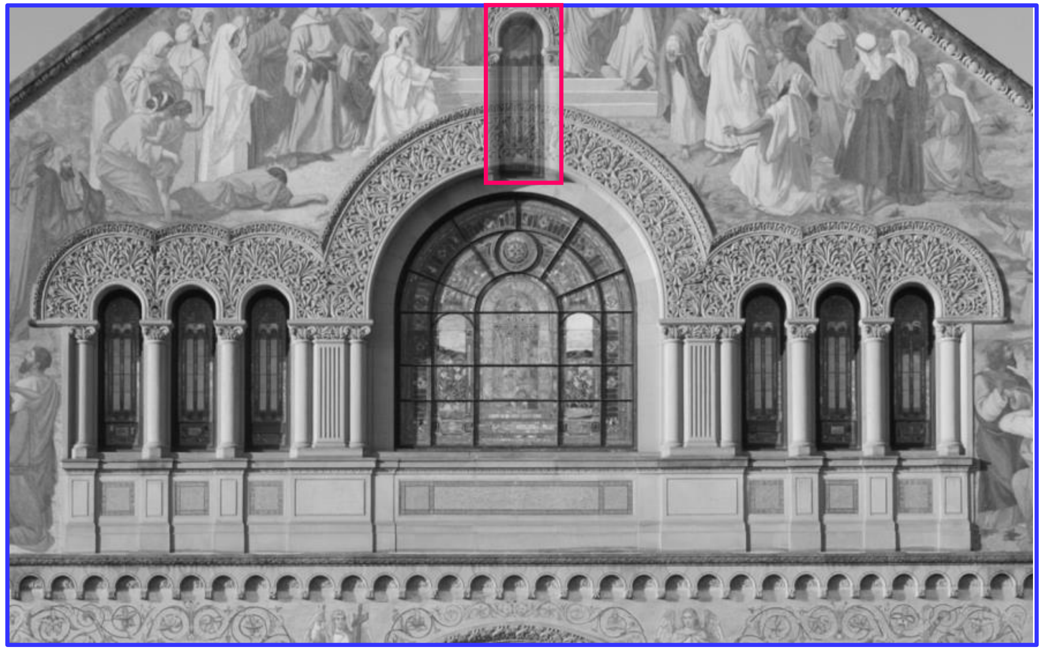
Correlation map

Template Matching: Graphical Understanding

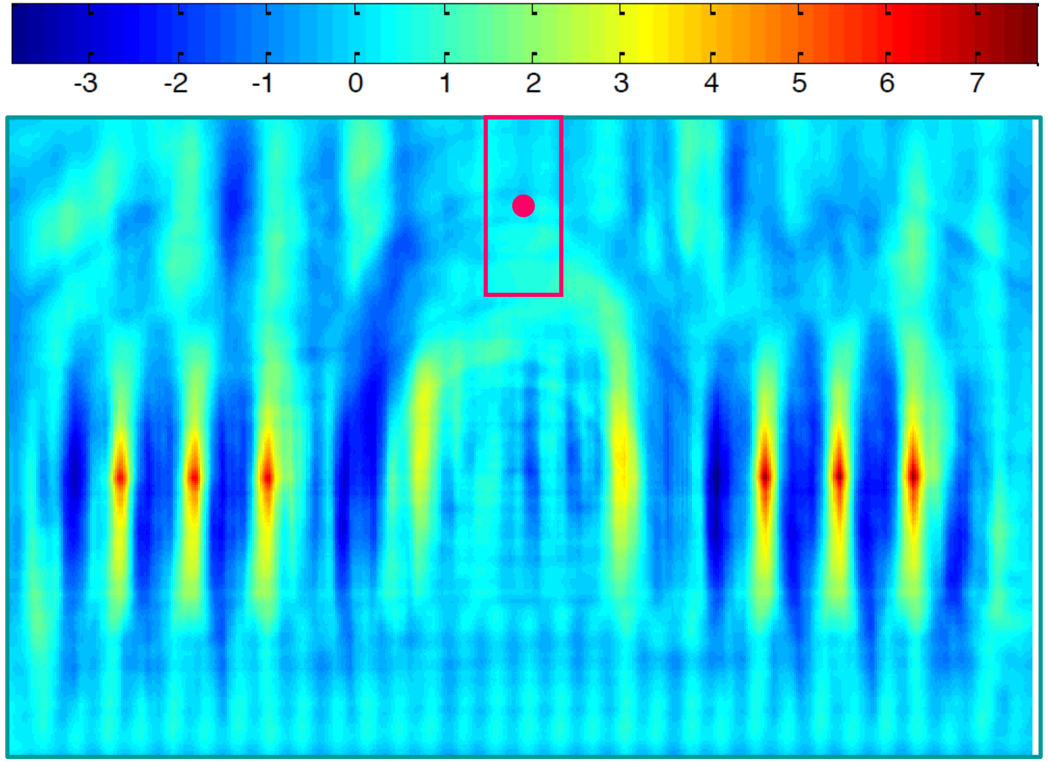
- Template matching is a technique in digital image processing **for finding small parts of an image** which match a template image



Template



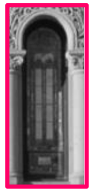
Image



Correlation map

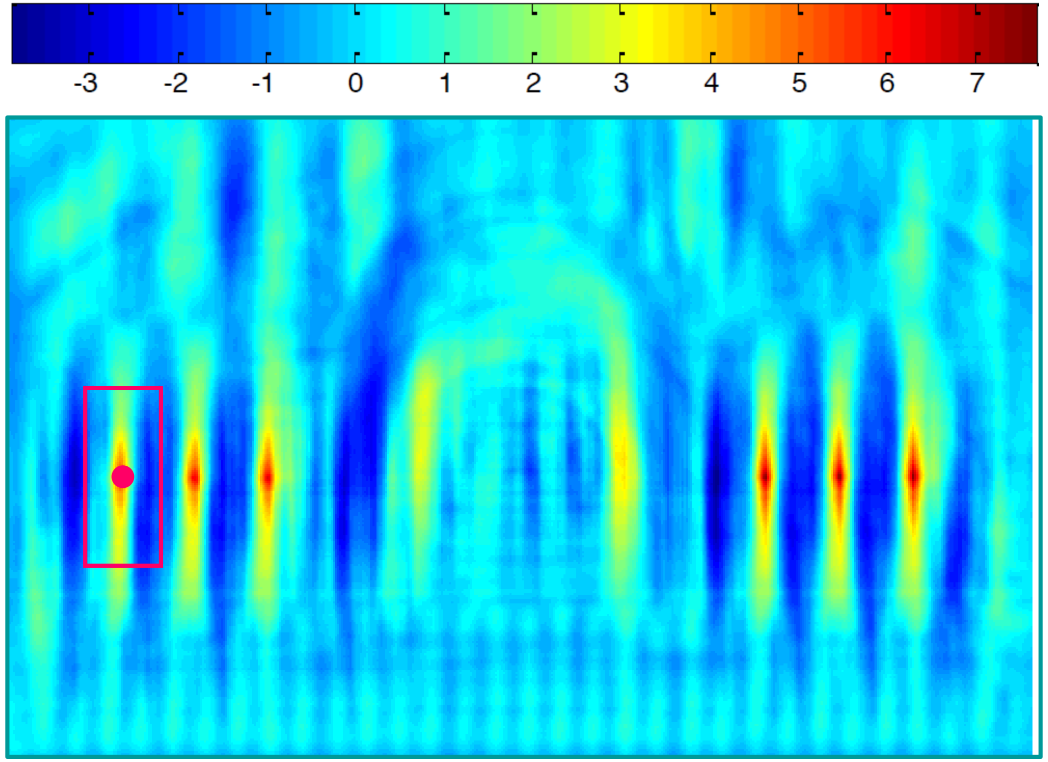
Template Matching: Graphical Understanding

- Template matching is a technique in digital image processing **for finding small parts of an image** which match a template image



Template

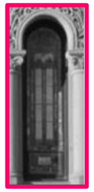
Image



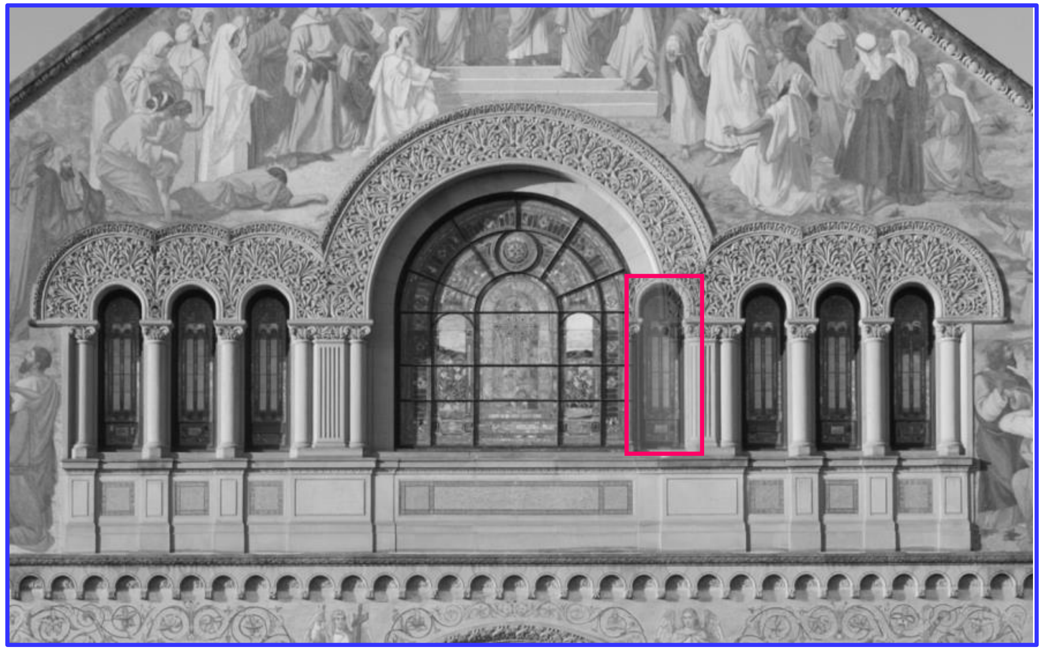
Correlation map

Template Matching: Graphical Understanding

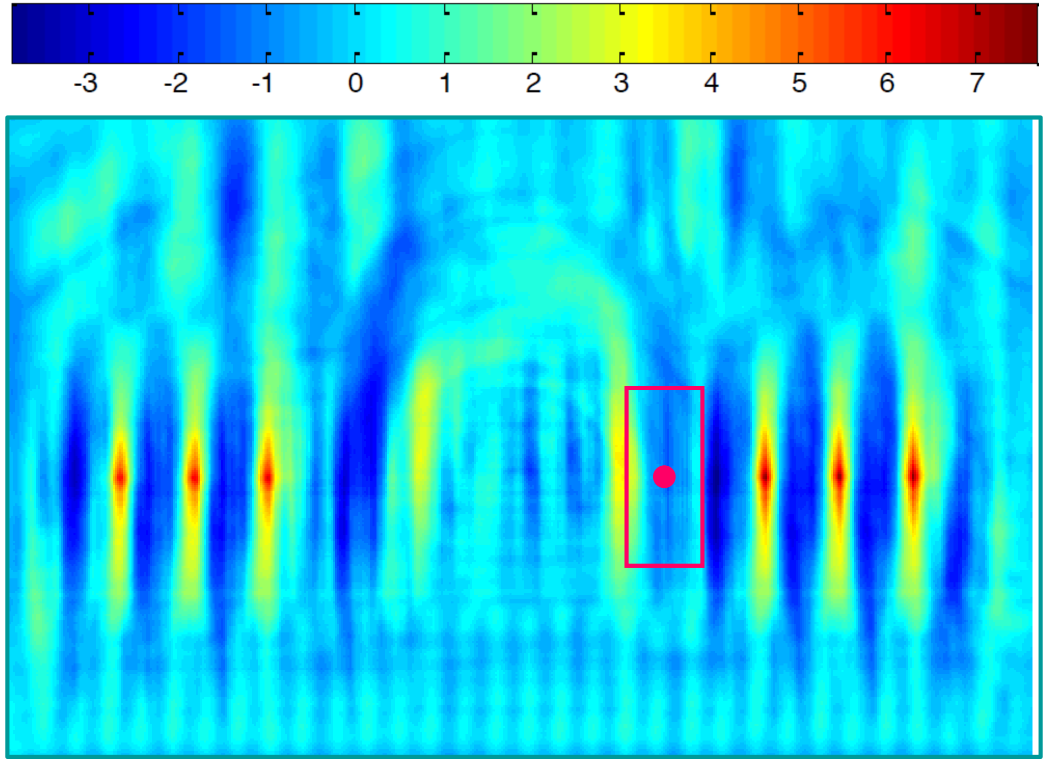
- Template matching is a technique in digital image processing **for finding small parts of an image** which match a template image



Template



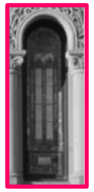
Image



Correlation map

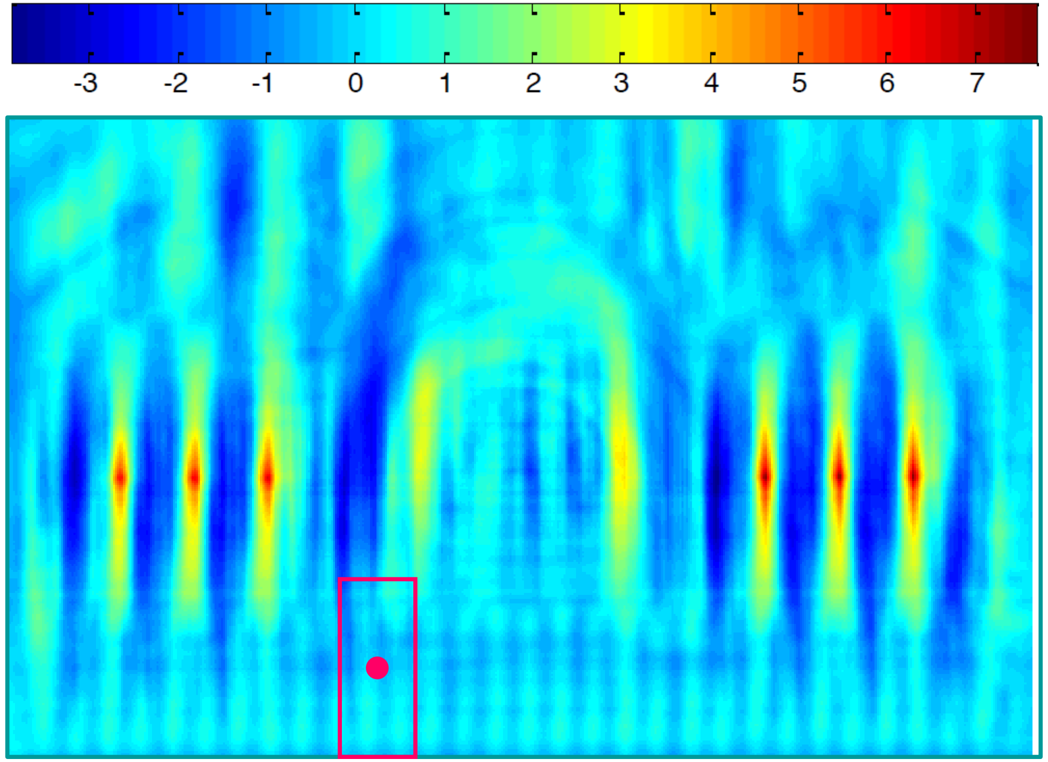
Template Matching: Graphical Understanding

- Template matching is a technique in digital image processing **for finding small parts of an image** which match a template image



Template

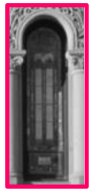
Image



Correlation map

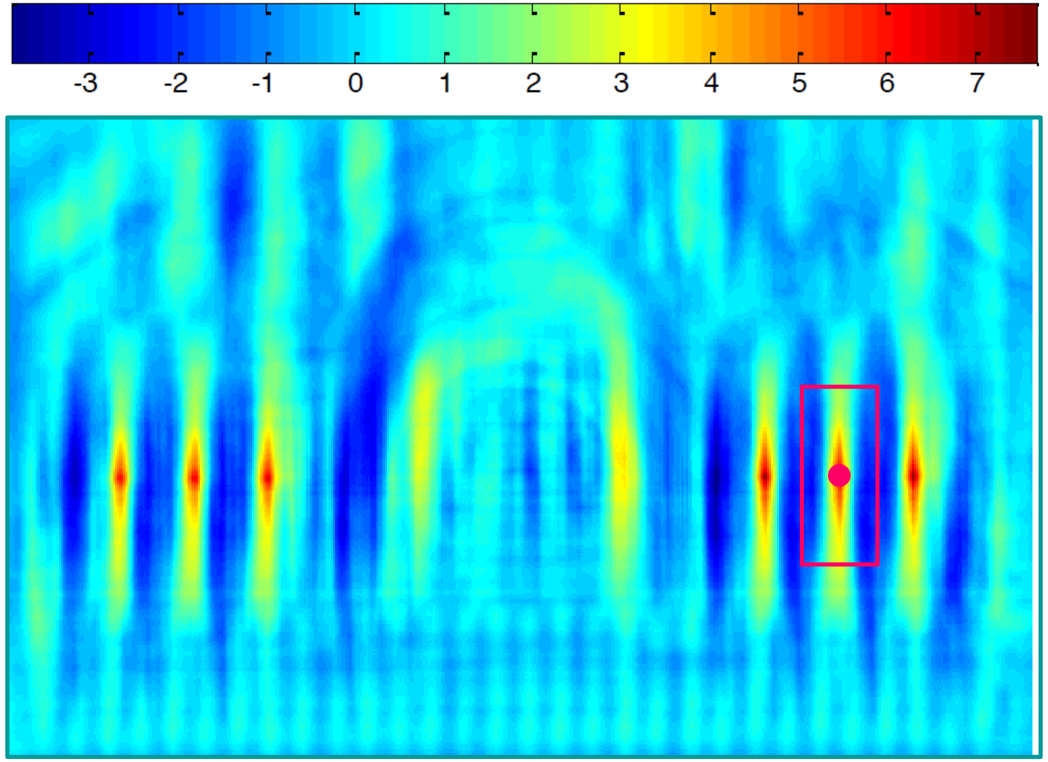
Template Matching: Graphical Understanding

- Template matching is a technique in digital image processing **for finding small parts of an image** which match a template image



Template

Image



Correlation map

Template Matching

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

Template Matching

- Let $\mathbf{a}$ and $\mathbf{b}$ be vectors. Let θ be the angle between them

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

- **Correlation** is **a dot product** operation and **measures the similarity** between the filter and each local image region

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

— NCC attains the value 1 when the filter and image region are identical

Template Matching

- Let $\mathbf{a}$ and $\mathbf{b}$ be vectors. Let θ be the angle between them

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

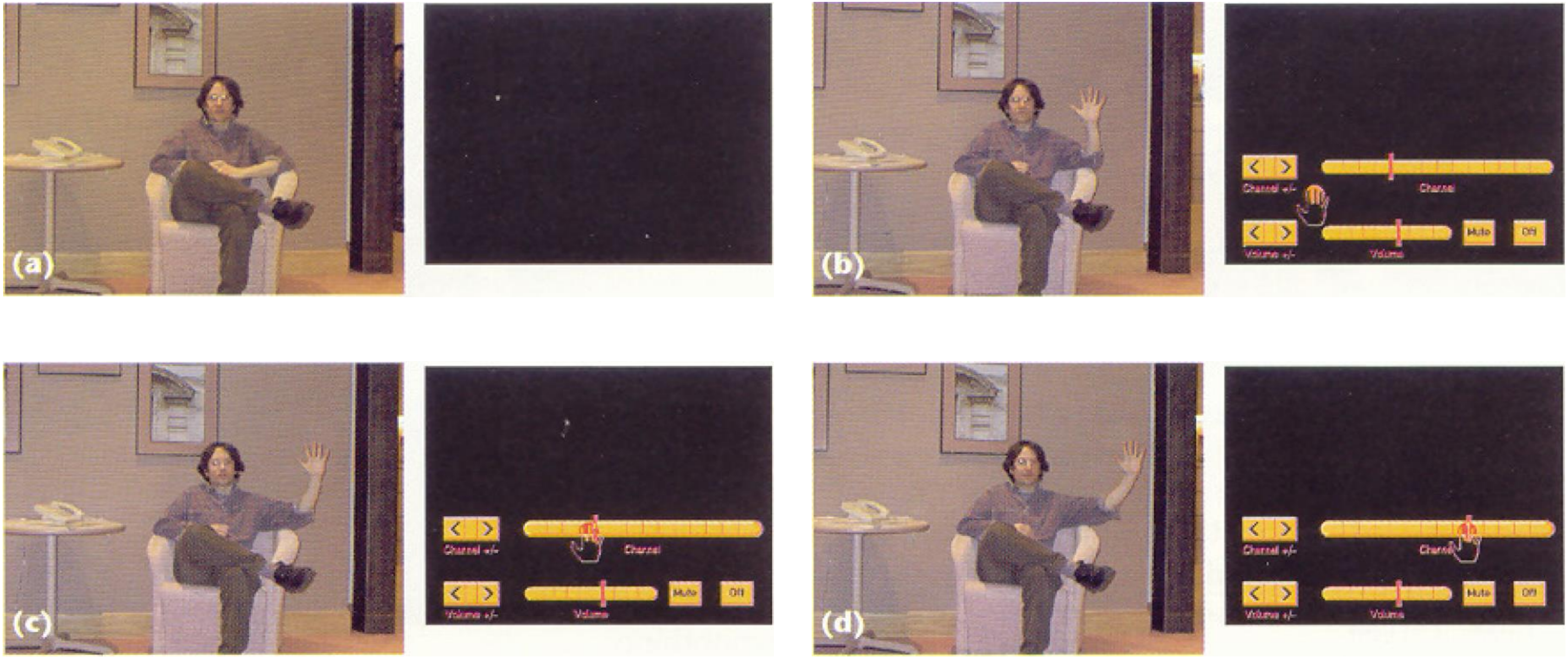
① Normalize the template / filter $(\mathbf{b})$ in the beginning

② Compute norm of $|\mathbf{a}|$ by convolving squared image with a filter of all 1's of equal size to the template and square-rooting the response

③ Compute dot product by convolution of image $(\mathbf{a})$ with normalized filter

④ Compute NCC by dividing element-wise result in step ③ by result in Step ②

Template Matching: Example



W. Freeman et al., Computer Vision for Interactive Computer Graphics, **IEEE Computer Graphics and Applications, 1998**

Template Matching: Example

Template

Image

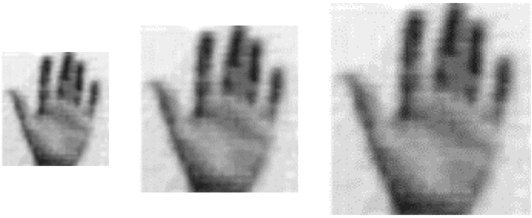
Correlation map

W. Freeman et al., Computer Vision for Interactive Computer Graphics, **IEEE Computer Graphics and Applications, 1998**

Difficulties of Template Matching

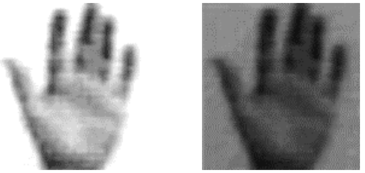
- **Failure Cases** of Template Matching

  — Different scales

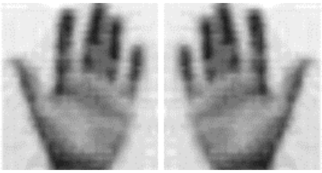


  — Rotations

  — Illumination condition



  — Asymmetry



  — Occlusions

  — Different perspective

  — Motion / blur

Summary: Template Matching

- Good:

— Works well in presence of noise

— Relatively easy to compute

- Bad:

— Sensitive to spatial scale change

— Sensitive to 2D rotation

— Sensitive to conditions of illumination

— Sensitive to viewing direction and pose in 3D worlds

Goals of **Scaled Representations**

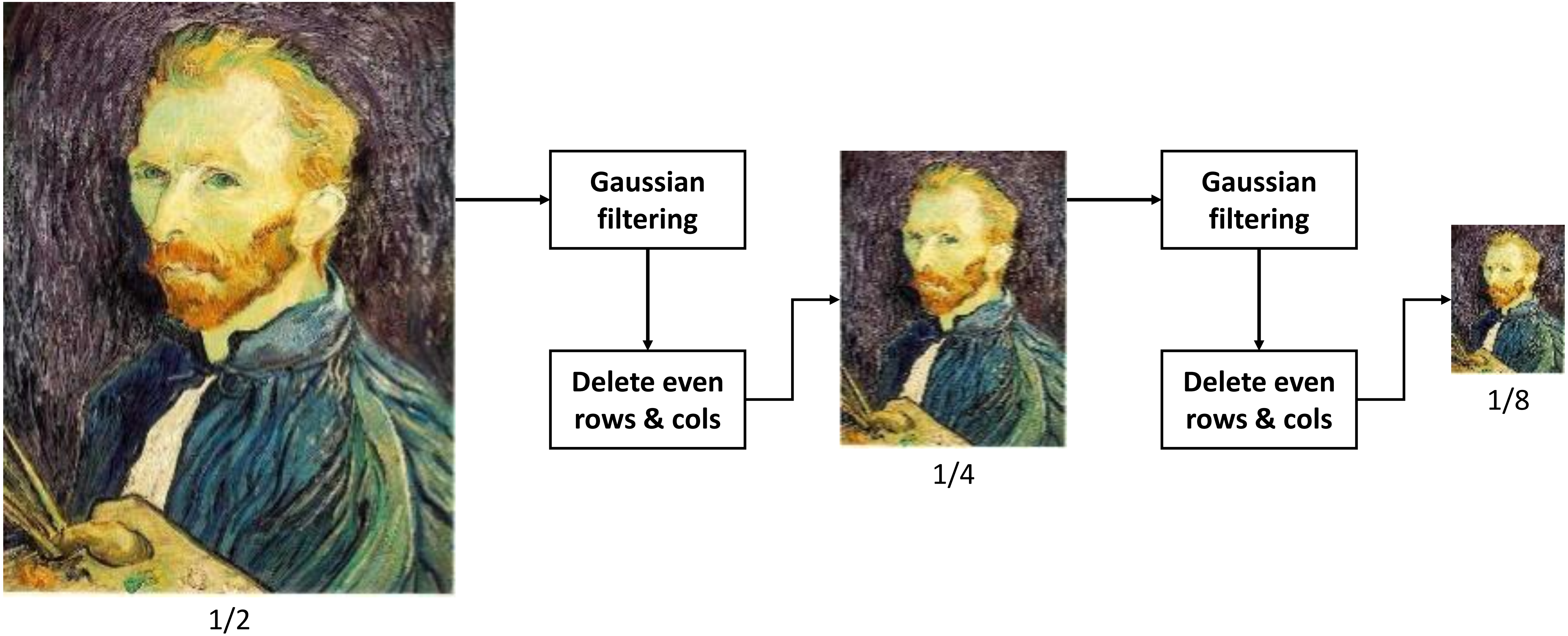
- To find template matches **at all scales**
— Template size constant, image scale varies
— Finding hands or faces when we don't know what size they are in the image

- **Efficient search** for image-to-image correspondences
— Look first at coarse scales, refine at finer scales
— Much less cost (but may miss best match)

- To examine **all levels of detail**
— Find edges with different amounts of blur
— Find textures with different spatial frequencies

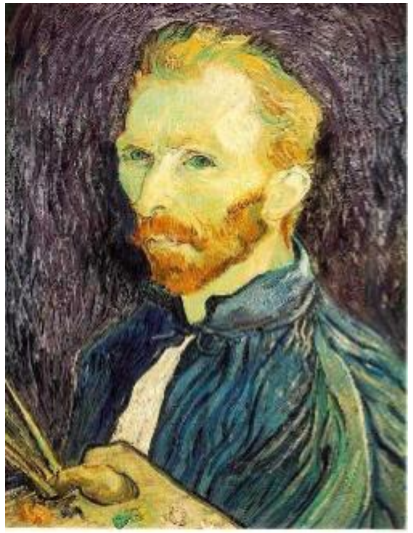
Template Matching: Sub-sampling

- Apply a smoothing filter first, then throw away half the rows and columns

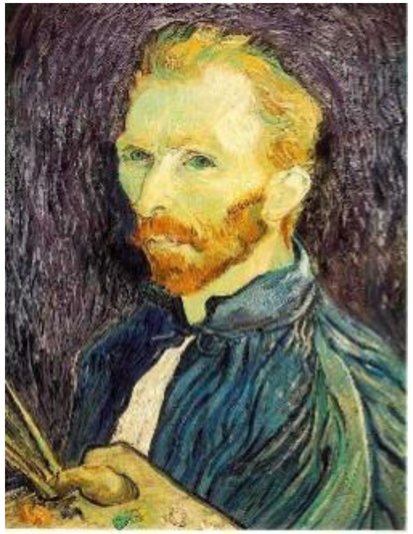


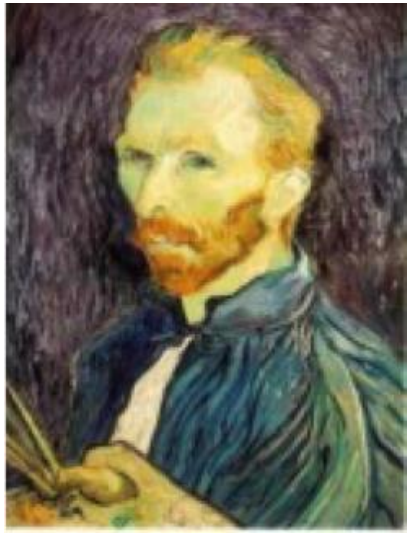
Template Matching: Sub-sampling

Sub-sampling
without pre-filtering



Sub-sampling
with Gaussian pre-filtering





1/2 Sampling | 1/4 Sampling (2x zoom) | 1/8 Sampling (4x zoom)

Image Pyramid for Scaled Representations

- An **image pyramid** is a collection of representations of an image.
  Typically, each layer of the pyramid is half the width and half the height of the previous layer

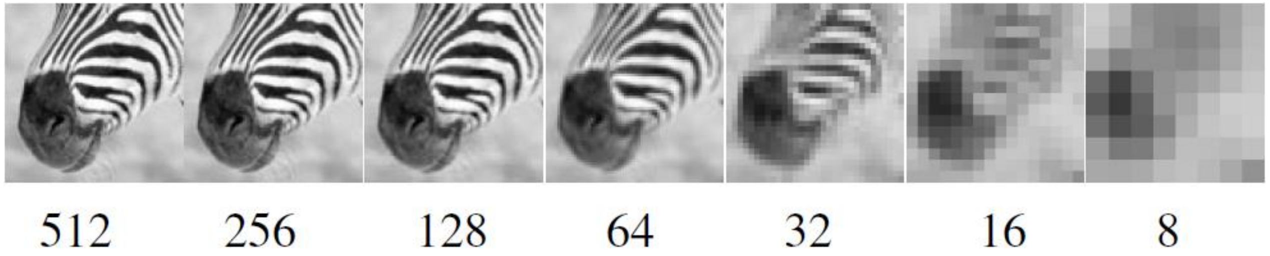
- In a **Gaussian pyramid**, each layer is smoothed by a Gaussian filter and resampled to get the next layer

Gaussian Pyramid: Example

- The details get smoothed out as we move to higher levels

- Mostly large uniform regions in the original image are preserved at the higher levels

- It is not possible to reconstruct the original image form the image at the higher level



Topics

- Template Matching
- Sampling Theory
- Color Filter Arrays

***Note:** Many of these slides in this course were adapted from Computer Vision at CMU (16-385) and UBC (CPSC425)

What is an Image?

- Images are a **discrete**, or **sampled**, representation of a continuous world

- Up to now provided a **physical characterization**

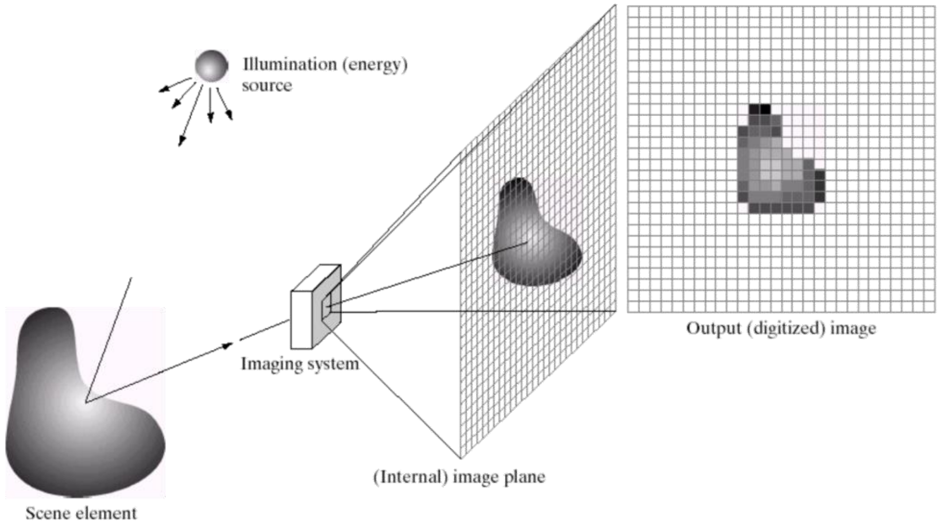
— Image formation as a problem in physics/optics

— We also talked about simple image processing algorithms on image arrays

- Now provide a **mathematical characterization**

— To understand how to represent images digitally

— To understand how to compute with images

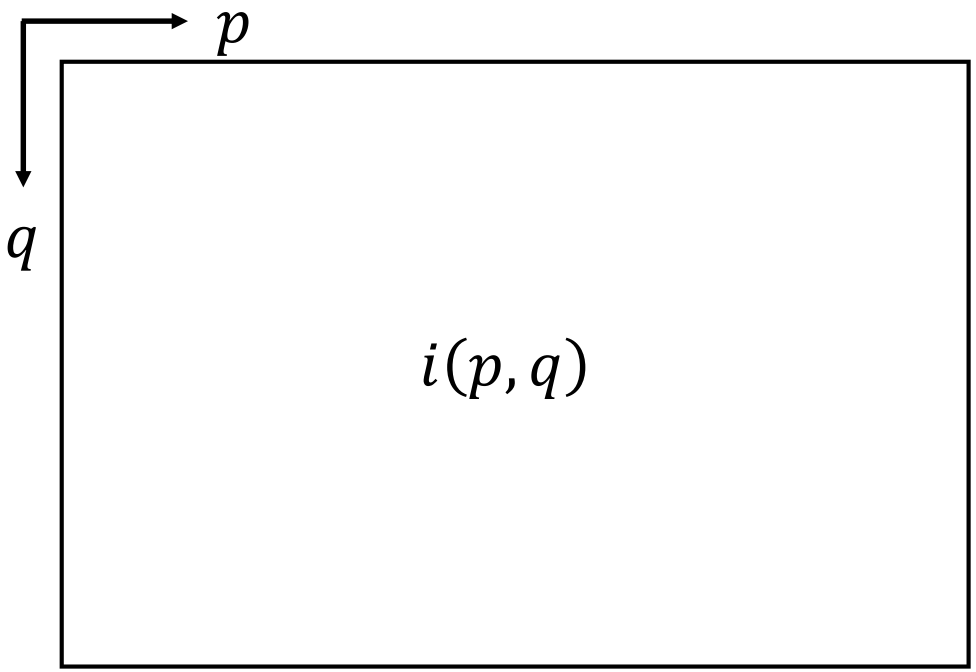


Continuous Case

- **Image** suggests a 2D surface whose appearance varies from point–to–point
— The surface typically is **a plane** (but might be curved, e.g., as is with an eye)

- Appearance can be **Grayscale** (Black and White) or **Color**
— In **Grayscale**, variation in appearance can be described by a single parameter corresponding to **the amount of light** reaching the image at a given point in a given time

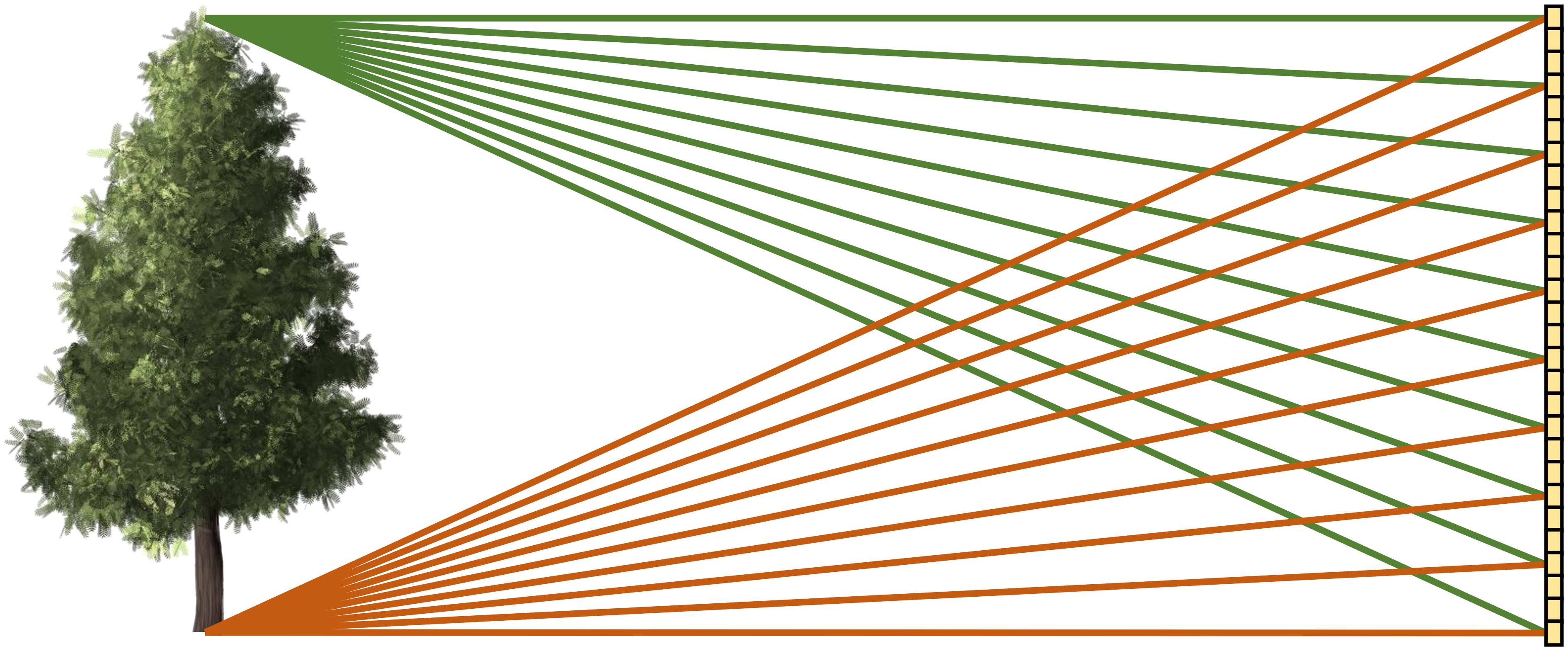
Continuous Case



- Denote the image as a function, $i(p,q)$, where p and q are spatial variables
- $i(p,q)$ is a **real-valued function** of **real spatial variables**
- $i(p,q)$ is **bounded** above and below: $0 \leq i(p,q) \leq M$

Continuous Case: Pinhole Camera

- Principal of Pinhole Camera

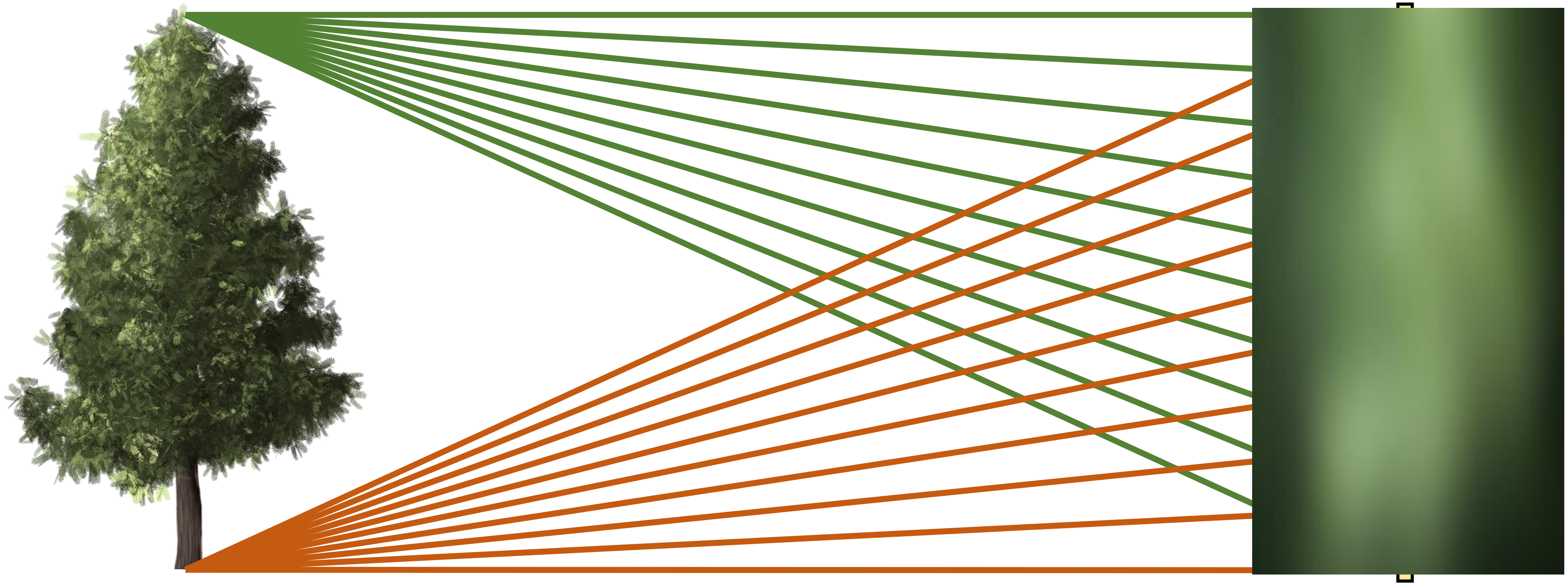


Object in Real World

Image Sensor

Continuous Case: Pinhole Camera

- Principal of Pinhole Camera

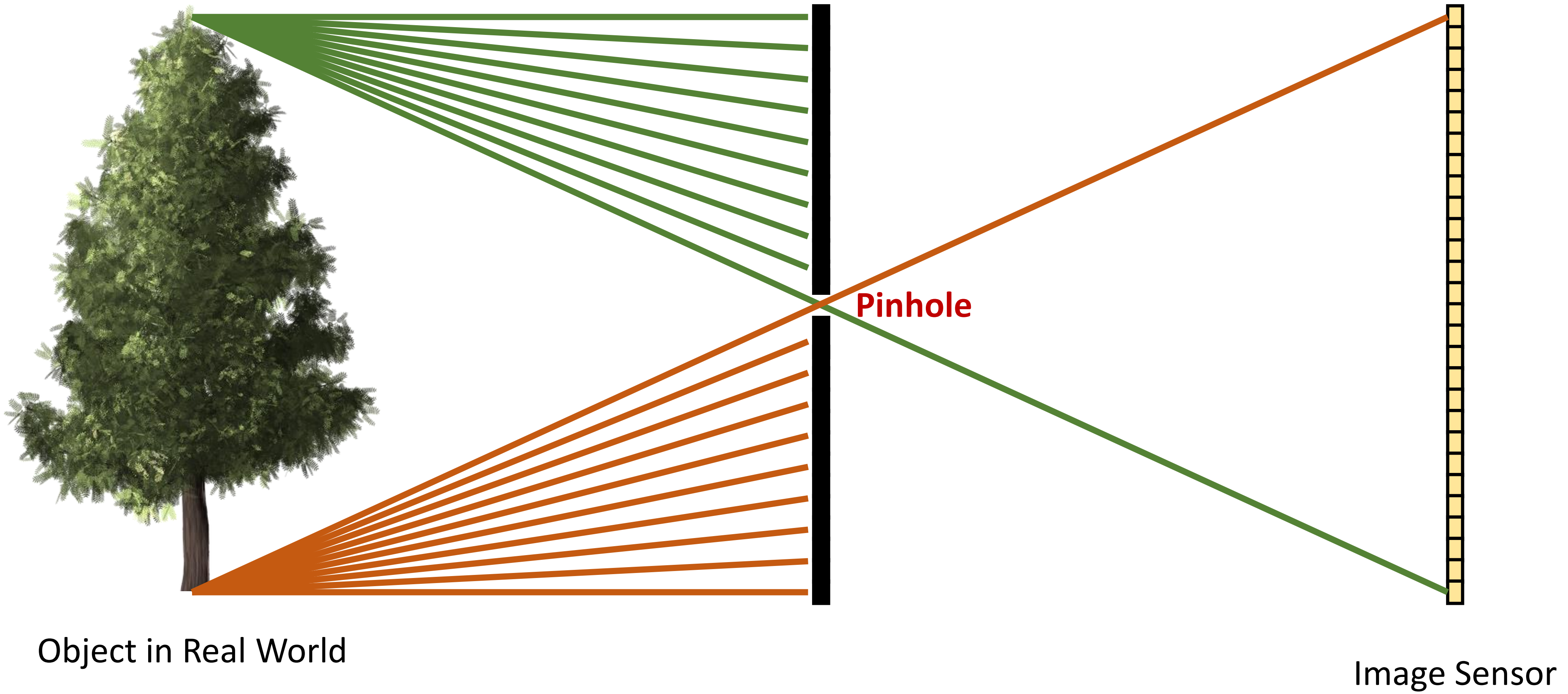


Object in Real World

Image Sensor

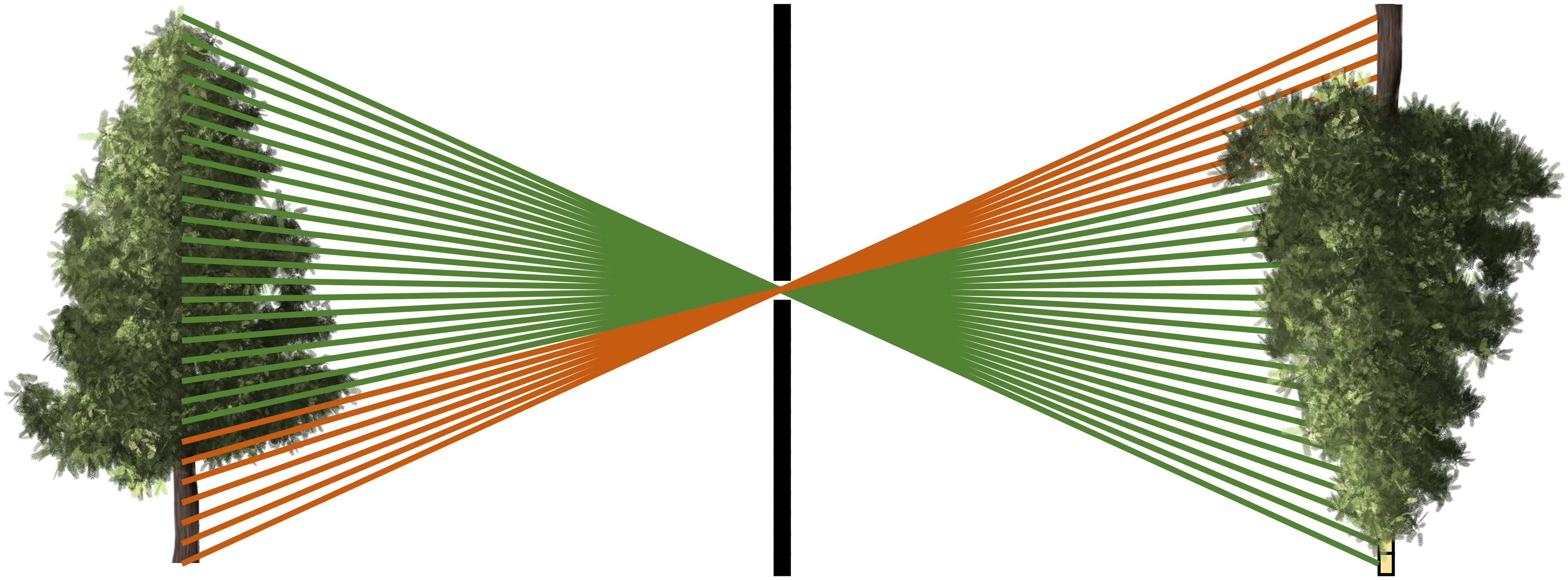
Continuous Case: Pinhole Camera

- Principal of Pinhole Camera



Continuous Case: Pinhole Camera

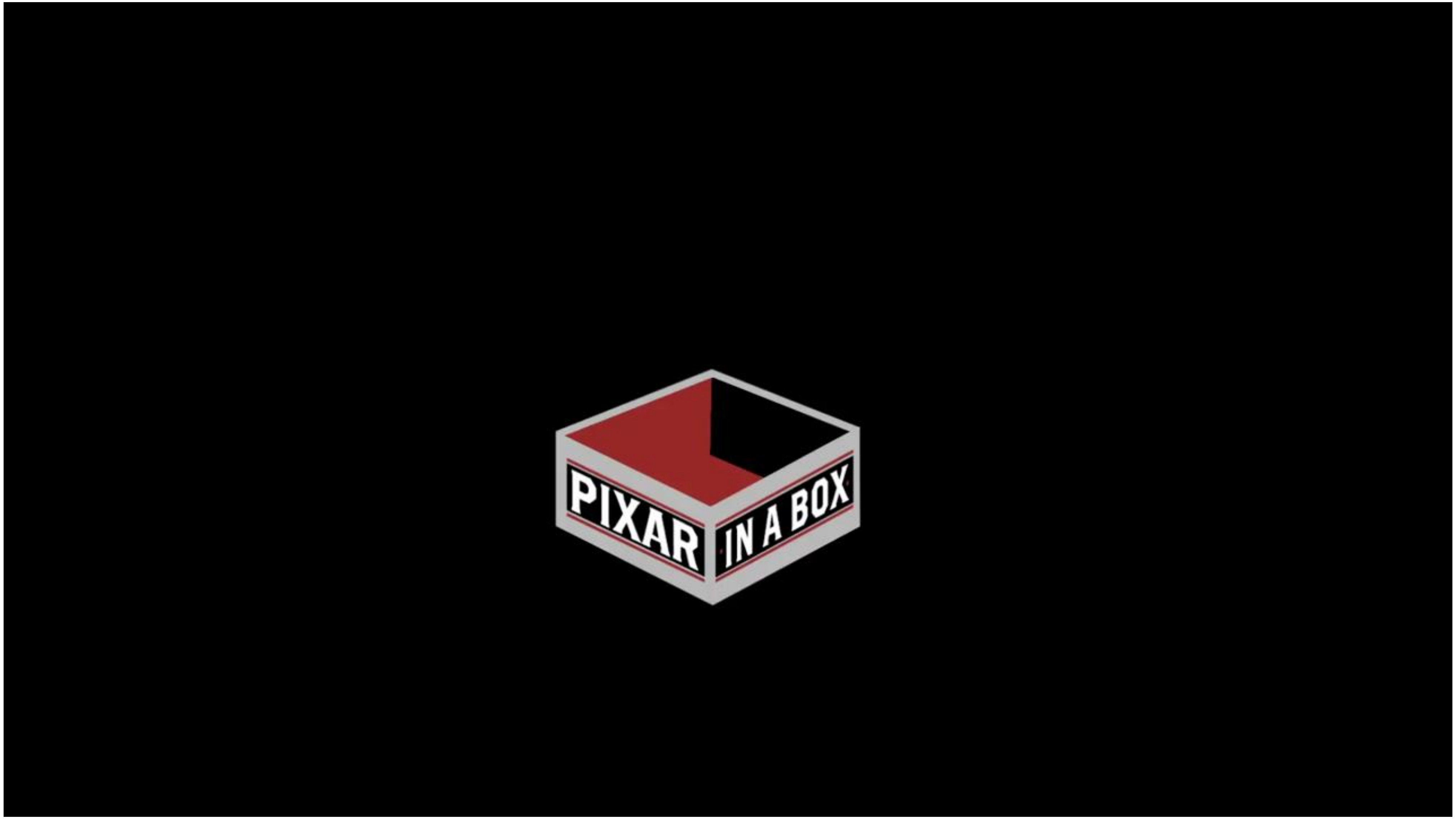
- Principal of Pinhole Camera



Object in Real World

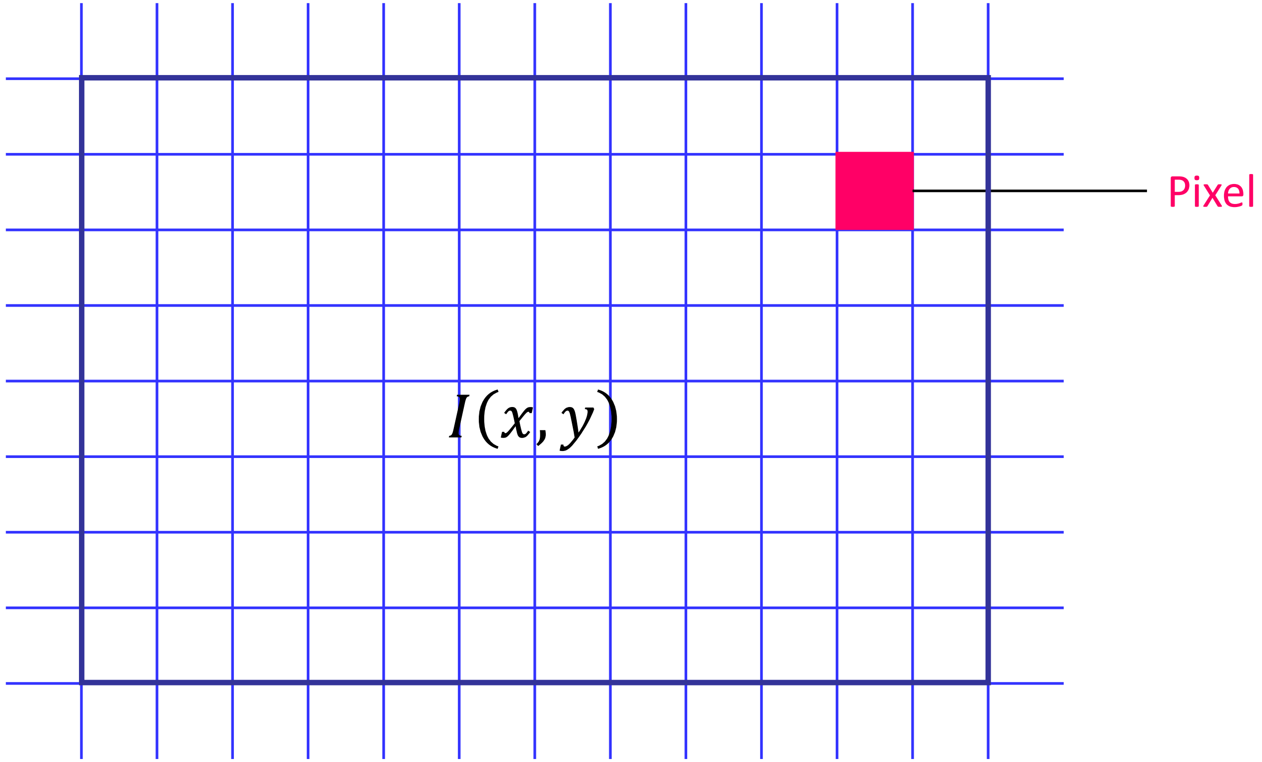
Image Sensor

Continuous Case: Pinhole Camera



https://www.youtube.com/watch?v=jhBC39xZVnw

Discrete Case



- Denote the discrete image as $I(x, y)$
- Each grid cell is called a picture element (**pixel**)
- Sample the underlying continuous image according to the **tessellation**

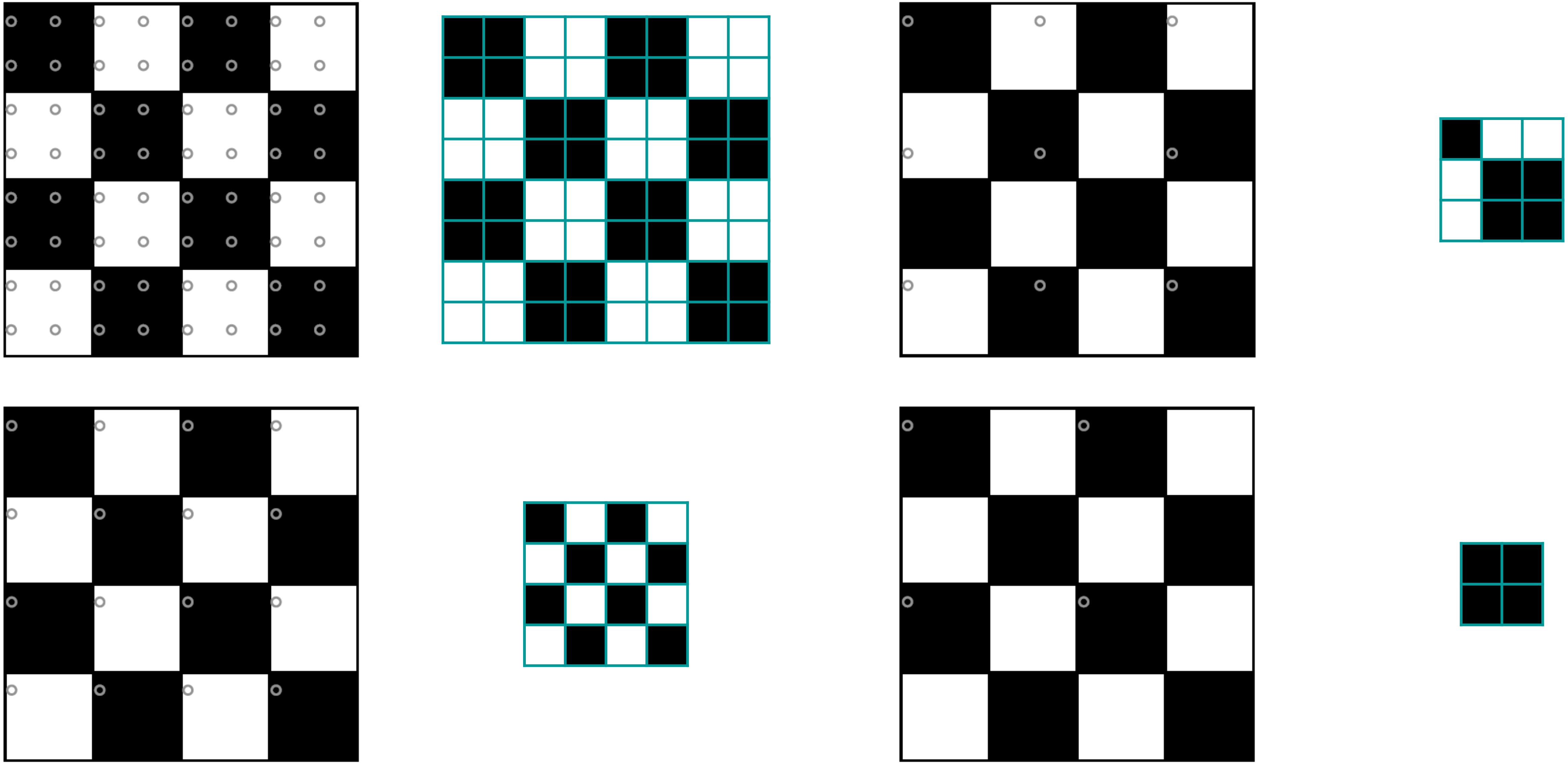
Discrete Case: **Quantization**

- To make values of $I(x, y)$ discrete, we need to divide the range $[0, M]$ into a finite number of equivalence classes: **Quantization**
- The values are called **gray-levels**
- Suppose n bits-per-pixel are available. One can divide the range $[0, M]$ into evenly spaced intervals as follows:

$$i(p,q) \longrightarrow \left\lfloor \frac{i(p,q)}{M}(2^n - 1) + 0.5 \right\rfloor$$

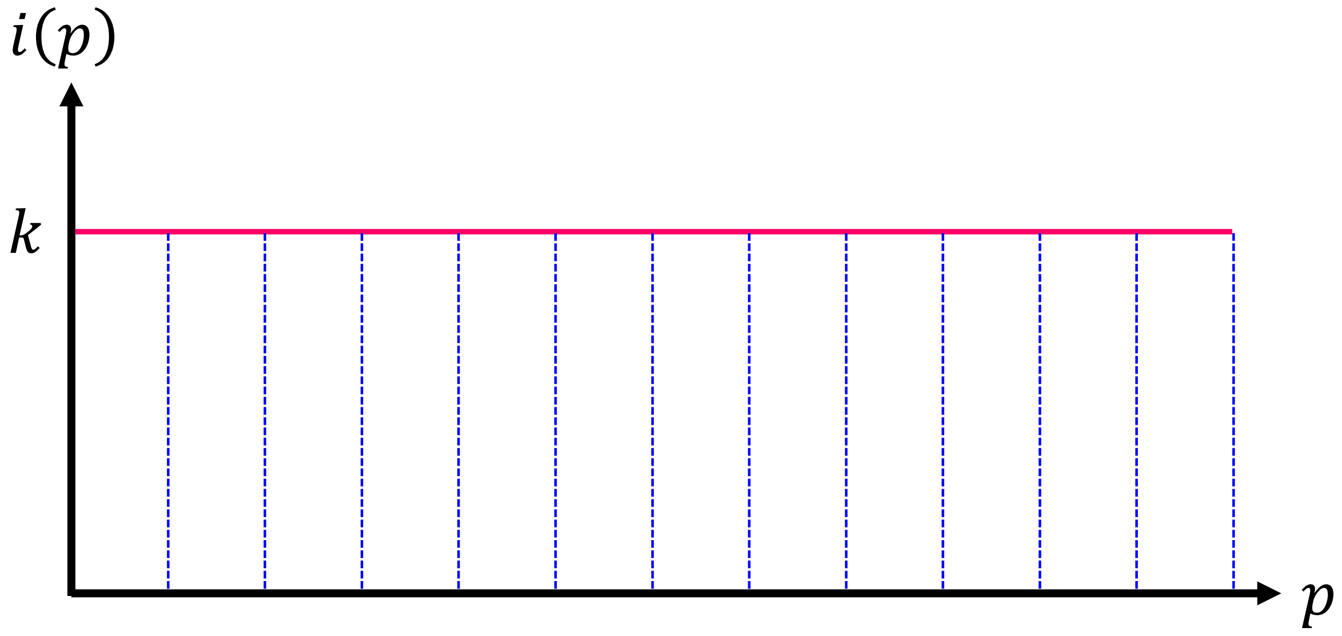
  - where $\lfloor \quad \rfloor$ is floor (i.e., greatest integer less than or equal to)
- Typically $n = 8$ resulting in grey-levels in the range $[0, 255]$

Discrete Case: **Sampling**



Sampling Theory

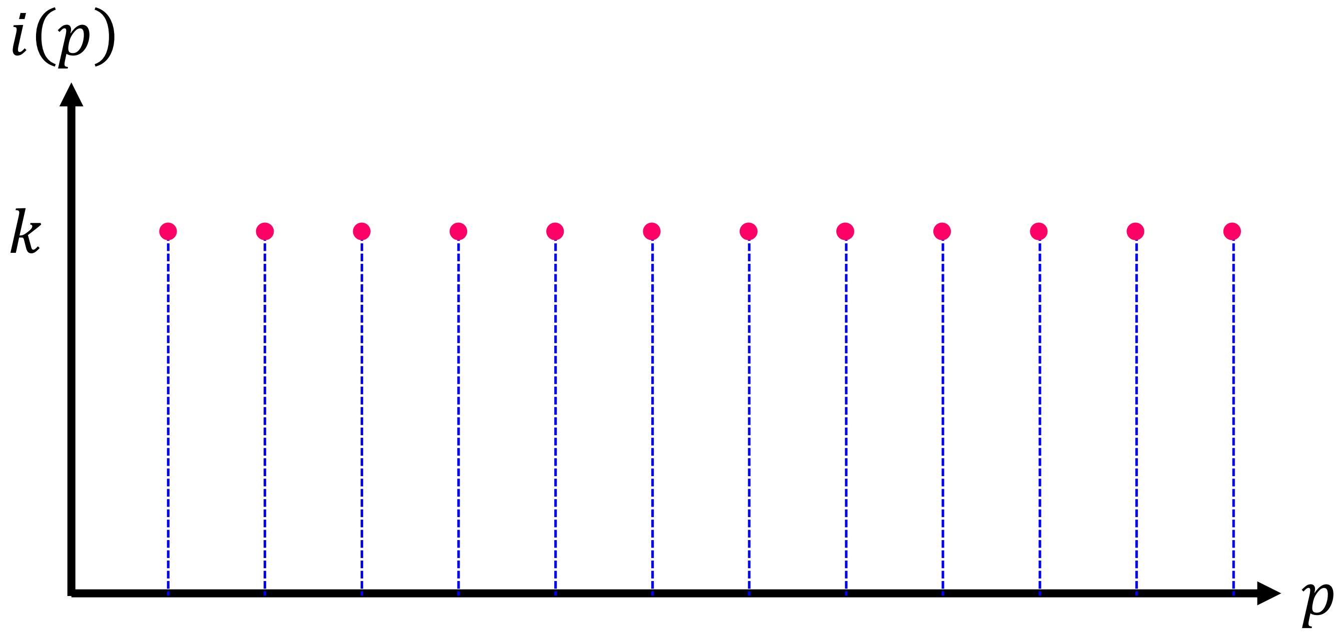
- **Case 0:** Suppose $i(p, q) = k$ (with k being one of our gray levels)



- **Note:** we use equidistant sampling at integer values for convenience

Sampling Theory

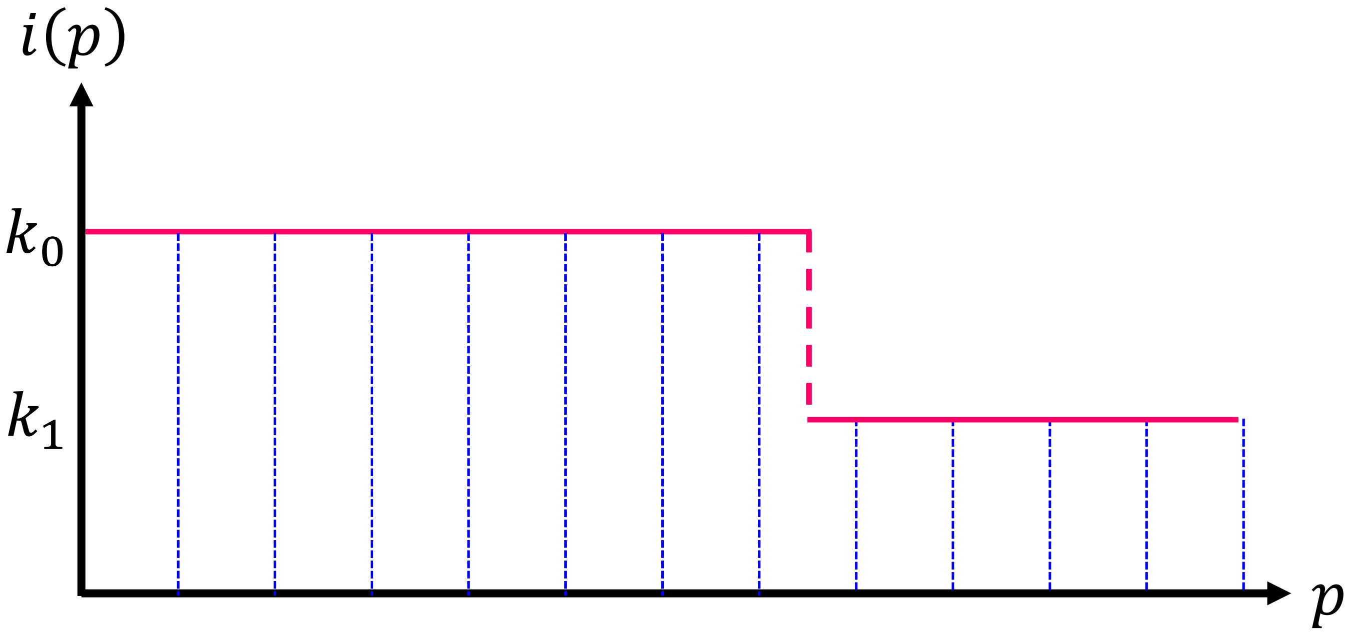
- **Case 0:** Suppose $i(p, q) = k$ (with k being one of our gray levels)



- $I(x, y) = k$. Any standard interpolation function would give $i(p, q) = k$ for non-integer p and q (irrespective of how coarse the sampling is)

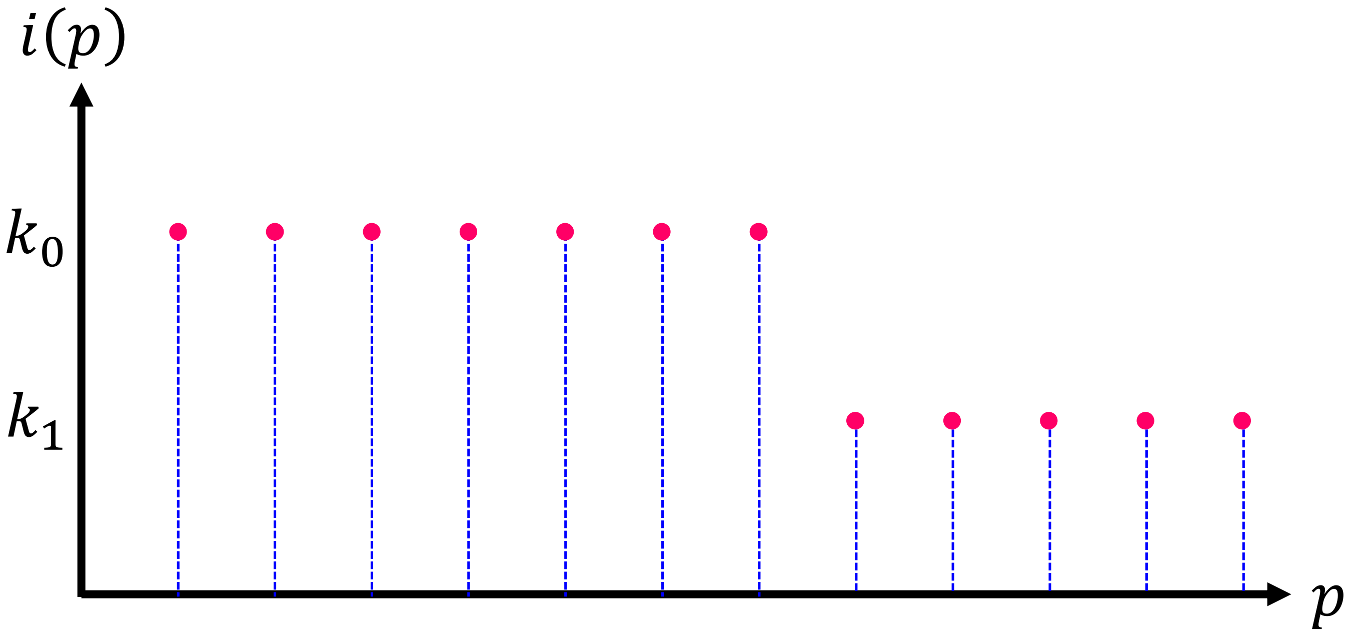
Sampling Theory

- **Case 1:** Suppose $i(p, q)$ has a discontinuity not falling precisely at integer



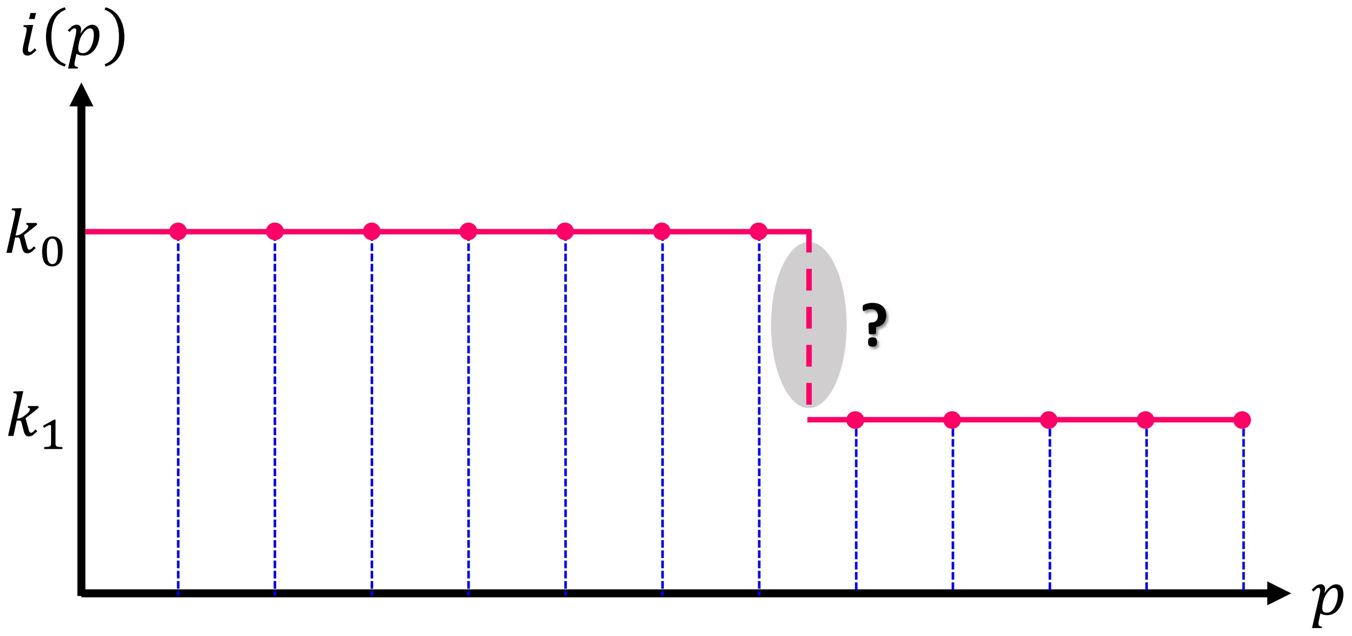
Sampling Theory

- **Case 1:** Suppose $i(p,q)$ has a discontinuity not falling precisely at integer



Sampling Theory

- **Case 1:** Suppose $i(p, q)$ has a discontinuity not falling precisely at integer

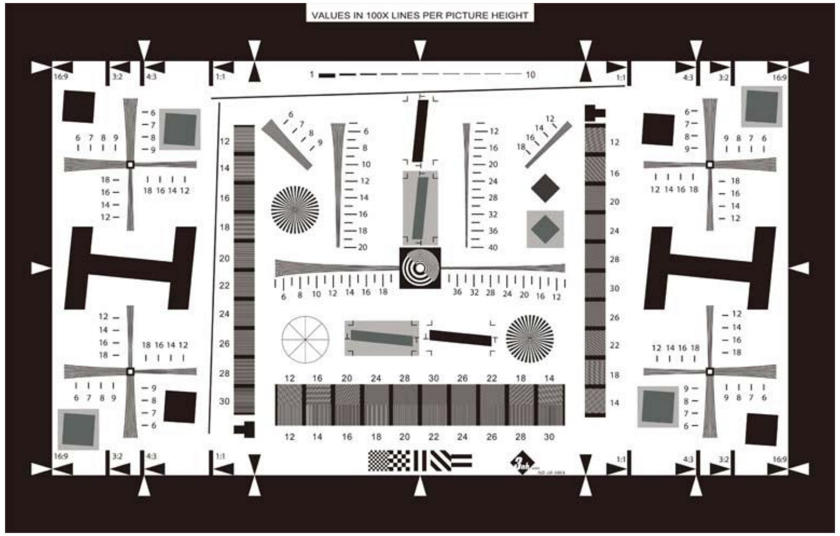


- Original signal, $i(p, q)$, cannot be reconstructed exactly because we can never know exactly where the discontinuity lies

Sampling Theory

- Exact reconstruction requires constraint on the rate at which $i(p, q)$ can change between samples
  - "**Rate of change**" means **derivative**
  - The formal concept is **bandlimited signal**
  - "Bandlimited" and "constraint on derivative" are linked

- Think of Imaging Systems
  - Line pairs per *mm* for a bar test pattern
  - Cycles per *mm* for a sine wave test pattern

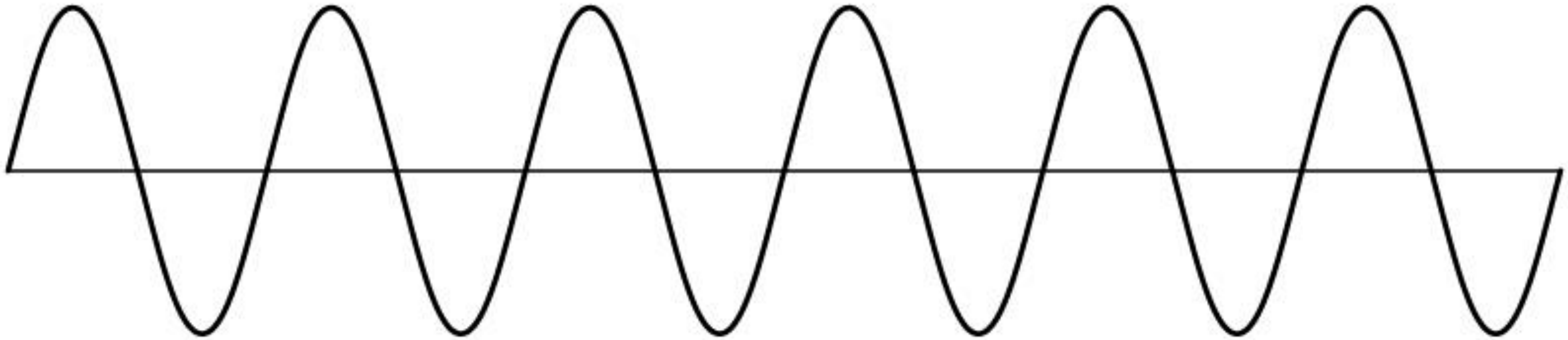


ISO 12233 chart

- An image is bandlimited if it has some maximum **spatial frequency**

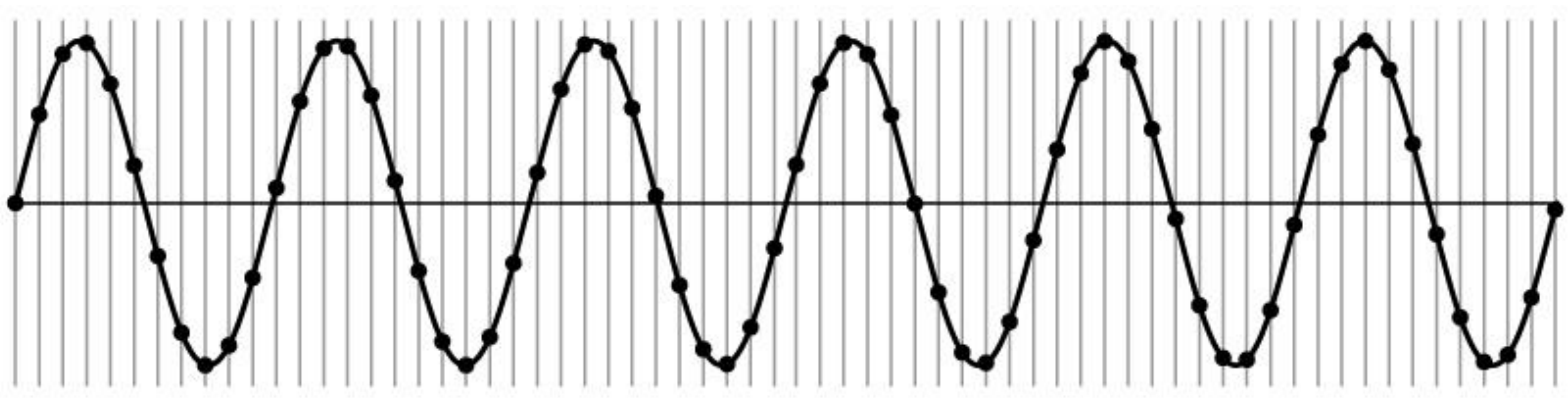
Sampling Example: A Simple Sine Wave

- How do we discretize the signal?



Sampling Example: A Simple Sine Wave

- How do we discretize the signal?

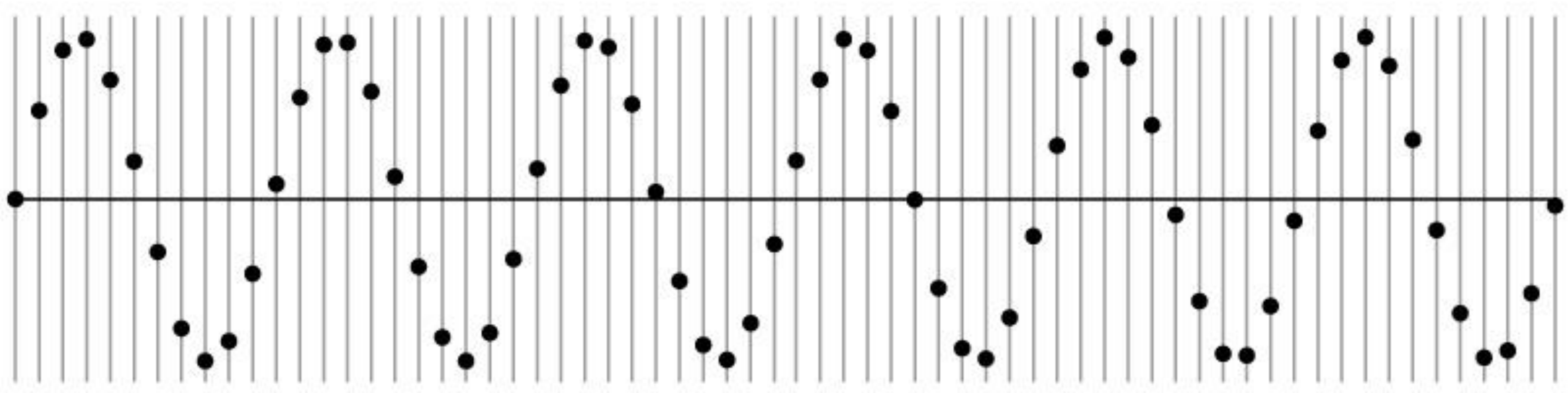


Sampling Example: A Simple Sine Wave

- How do we discretize the signal?

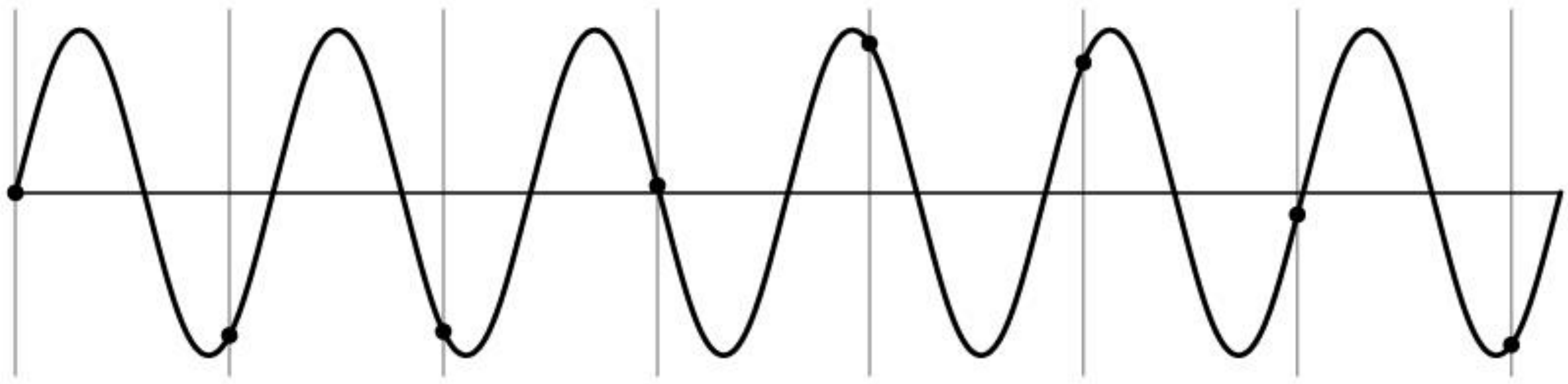
— How many samples should I take?

— Can I take as few samples as I want?



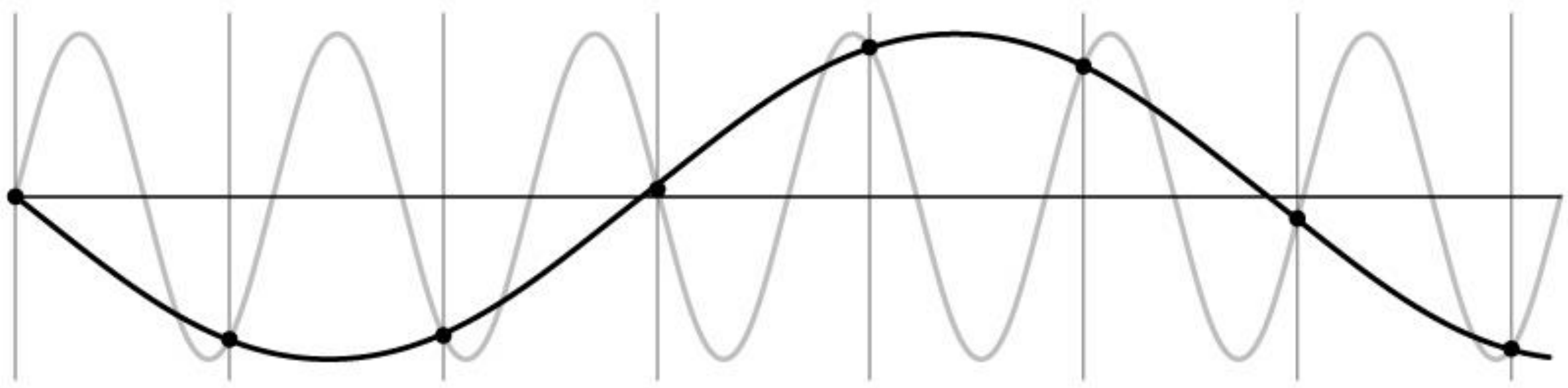
Sampling Example: A Simple Sine Wave

- Signal can be confused with one at **lower** frequency



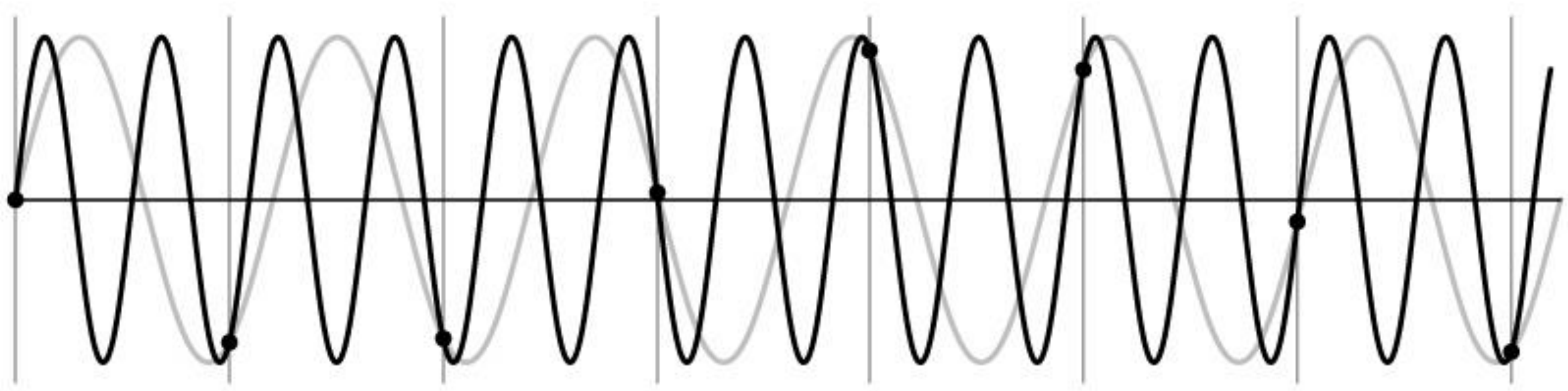
Sampling Example: A Simple Sine Wave

- Signal can be confused with one at **lower** frequency
- We call it **undersampling**



Sampling Example: A Simple Sine Wave

- Signal can always be confused with one at **higher** frequency
- We call it **undersampling**



Sampling Theory

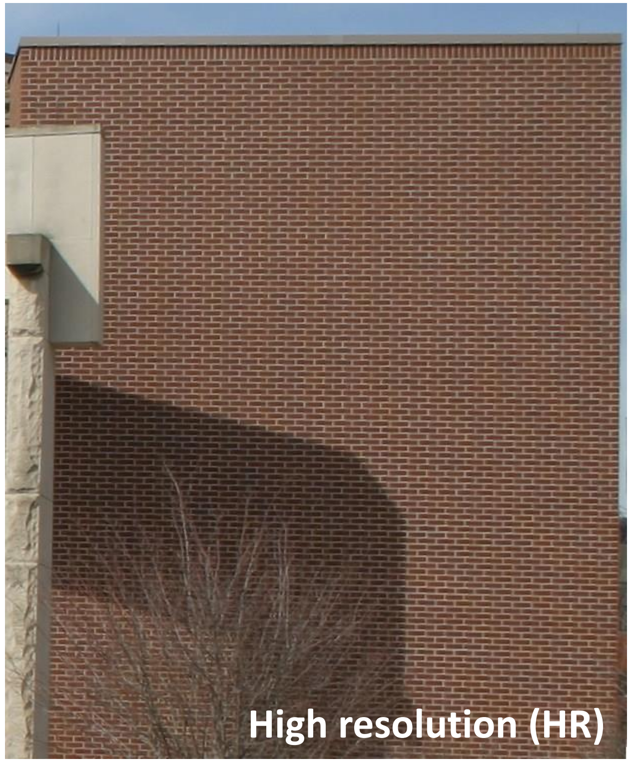
- The challenge to intuition is the fact that music (in the 1D case) and images (in the 2D case) can be represented as linear combinations of individual sine waves of differing frequencies and phases

- **Sampling Theorem:**

— 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

Spatial Aliasing

- Sampling involves **a loss of information**
- High spatial frequency components of the original signal appear as low spatial frequency components in the sampled signal – An effect known as **aliasing**



High resolution (HR)

Low resolution (LR)

Moire pattern

https://en.wikipedia.org/wiki/Aliasing

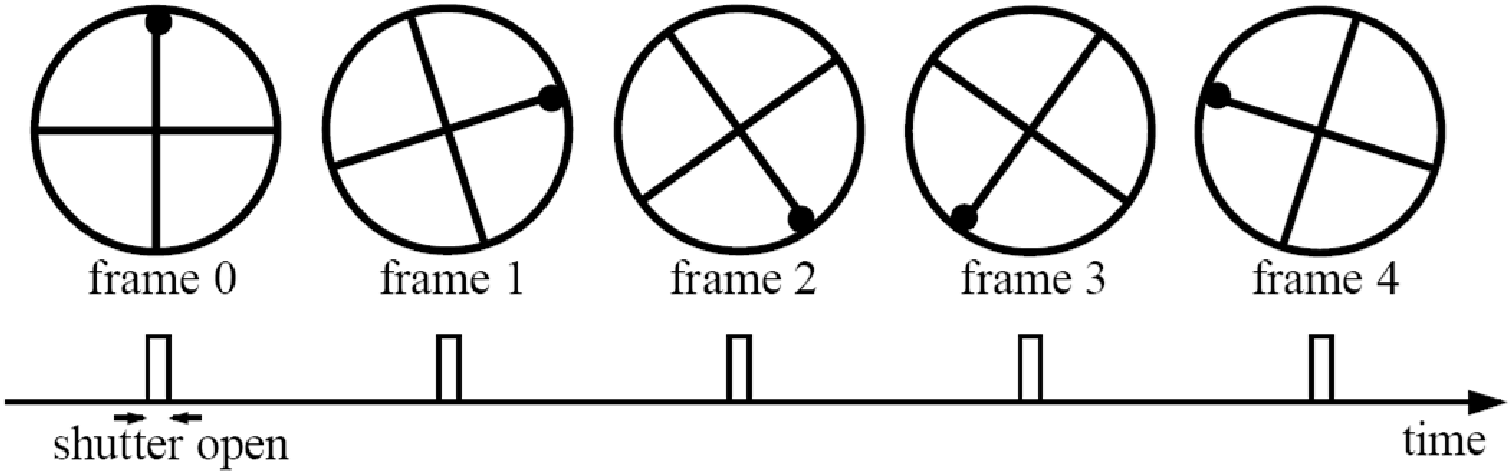
Temporal Aliasing

- Imagine a spoked wheel moving to the right (clockwise rotation)

— Mark wheel with dot so we can see what’s happening

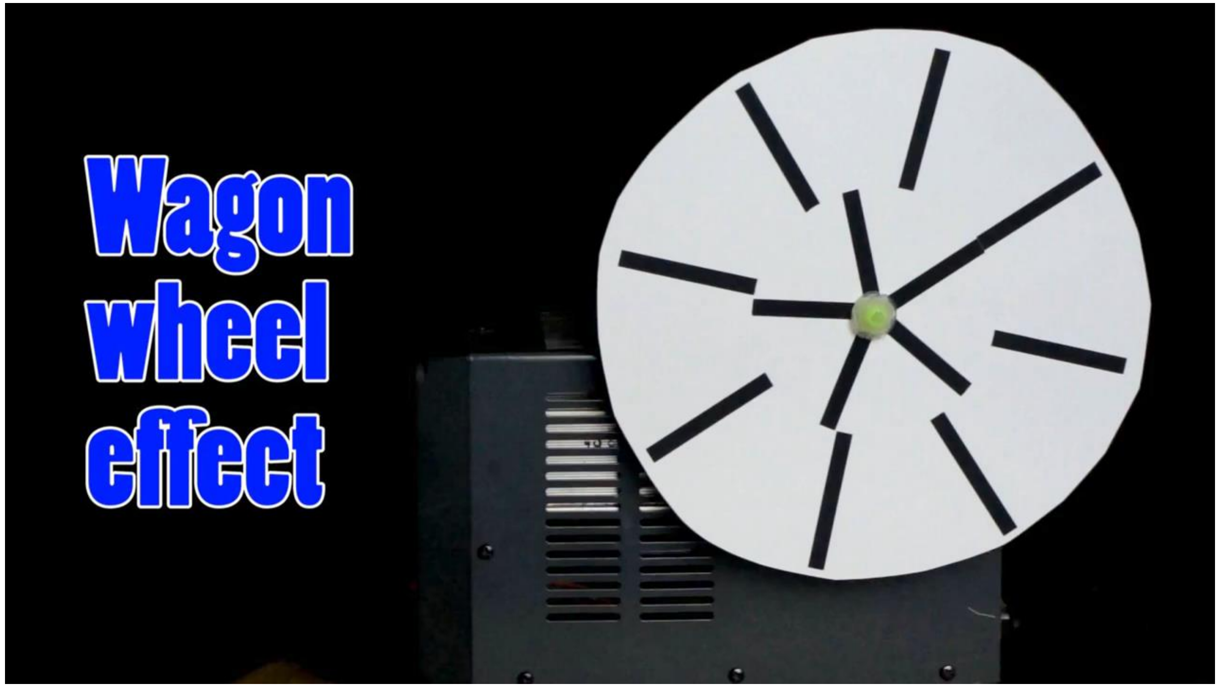
— If camera shutter is only open for a fraction of a frame time

  - frame time = 1/30 sec for video, 1/24 sec for film



- Without dot, wheel appears to be rotating slowly backwards

Temporal Aliasing



https://www.youtube.com/watch?v=HqKfCUW17QM

https://www.youtube.com/watch?v=yr3ngmRuGUc

Anti-Aliasing

- **Oversampling**

— Sample more than we think we need and average (i.e., area sampling)

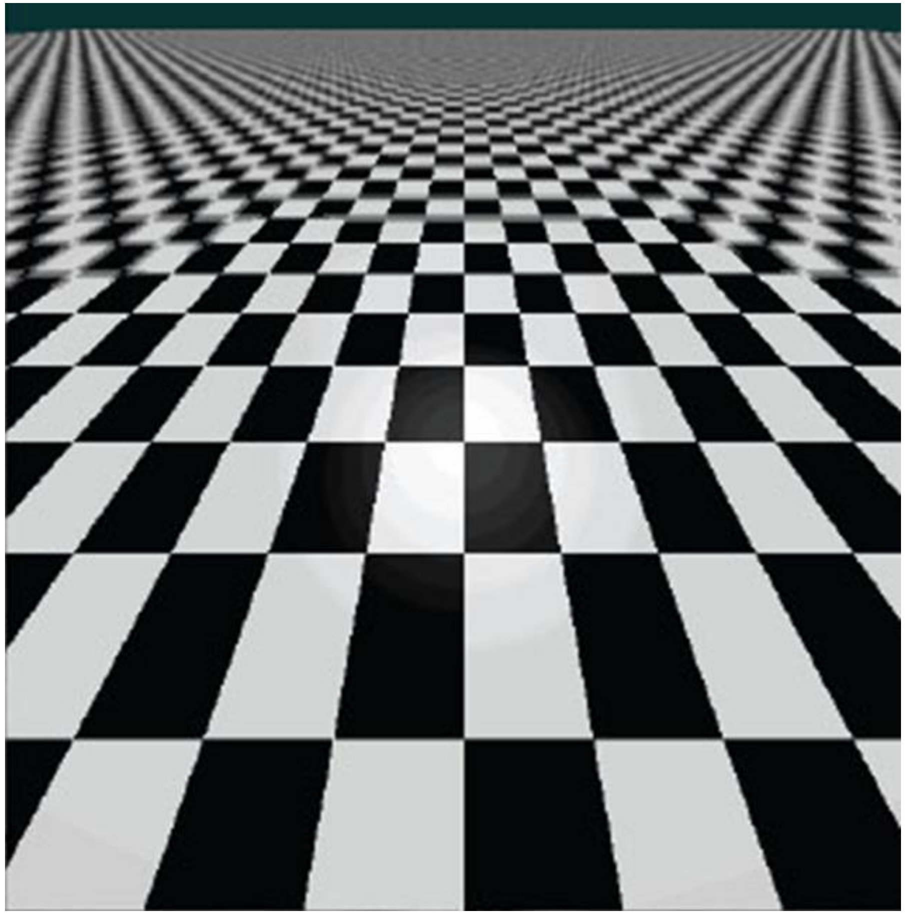
- **Smoothing**

— Remove some of the details that cause aliasing

— Lose information, but better than aliasing artifacts

Anti-Aliasing: **Oversampling**

Aliasing artifacts



Anti-aliasing by oversampling

Anti-Aliasing: **Smoothing**

Sampling
without smoothing

Sampling
with smoothing

1/2 Sampling

1/4 Sampling (2x zoom)

1/8 Sampling (4x zoom)

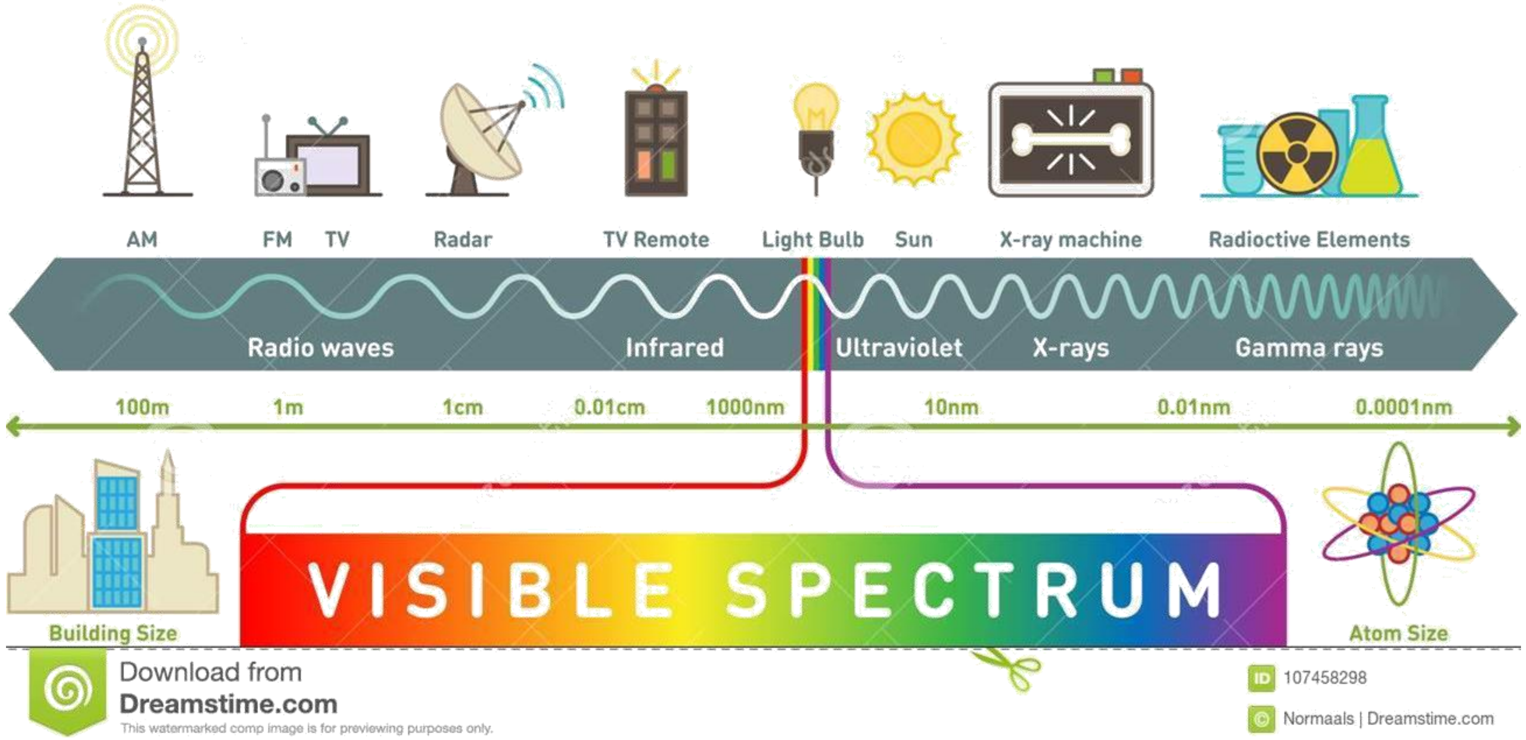
Topics

- Template Matching
- Sampling Theory
- Color Filter Arrays

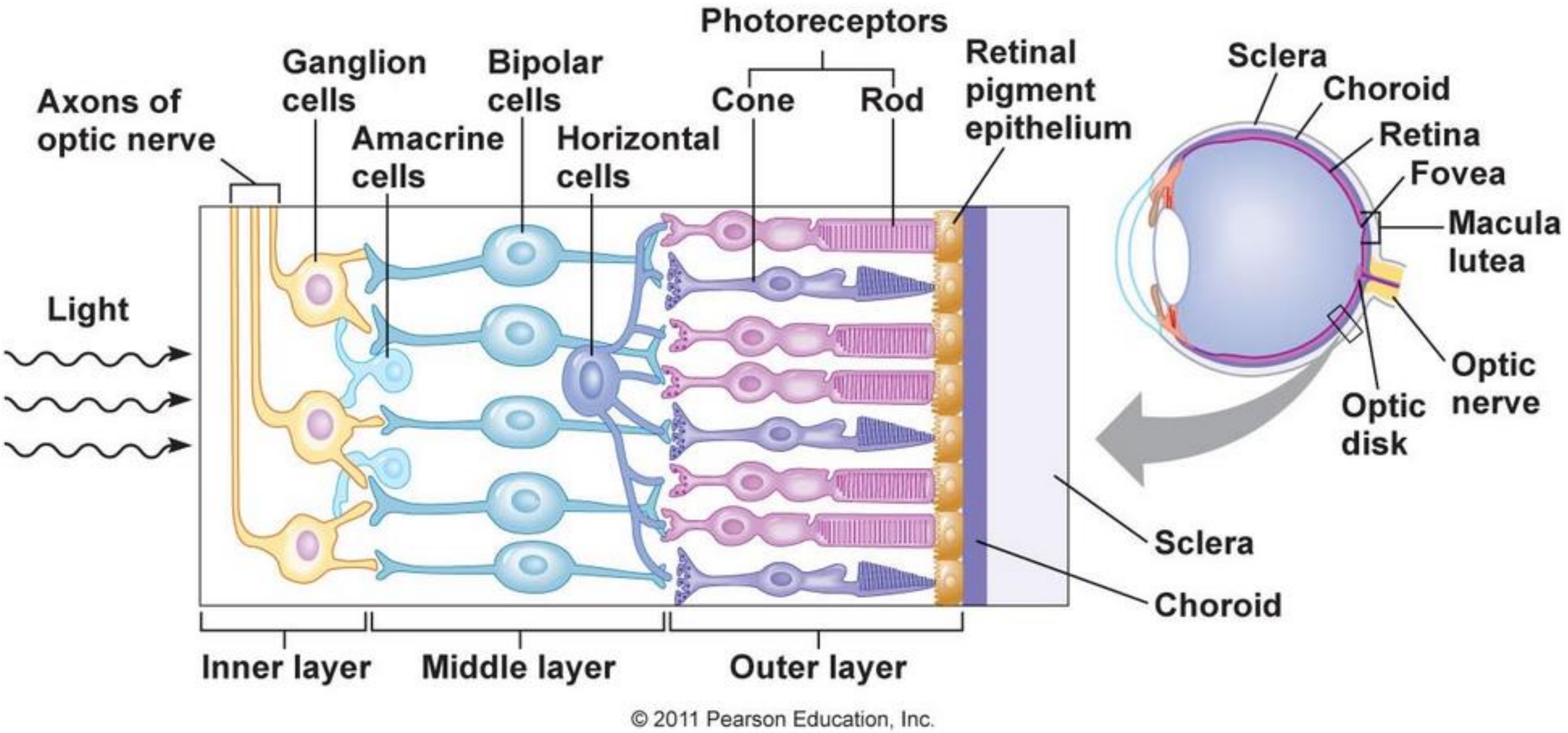
***Note:** Many of these slides in this course were adapted from Computer Vision at CMU (16-385) and UBC (CPSC425)

Color: 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



Retina in Our Eyes



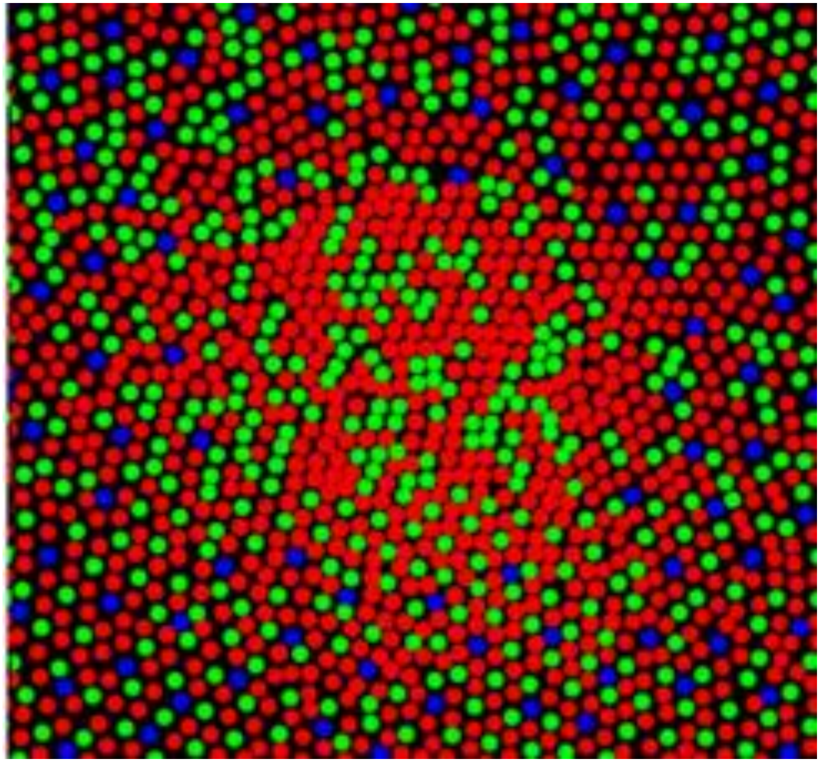
Rods & Cones

- **Rods**
  - For low luminance vision (< 1 cd/m^2): **Scotopic/night** vision
  - ~120 million per retina
  - Primarily located **outside the fovea**
  - One type of photo-pigment (rhodopsin) with peak spectral responsivity at ≈ 510 nm

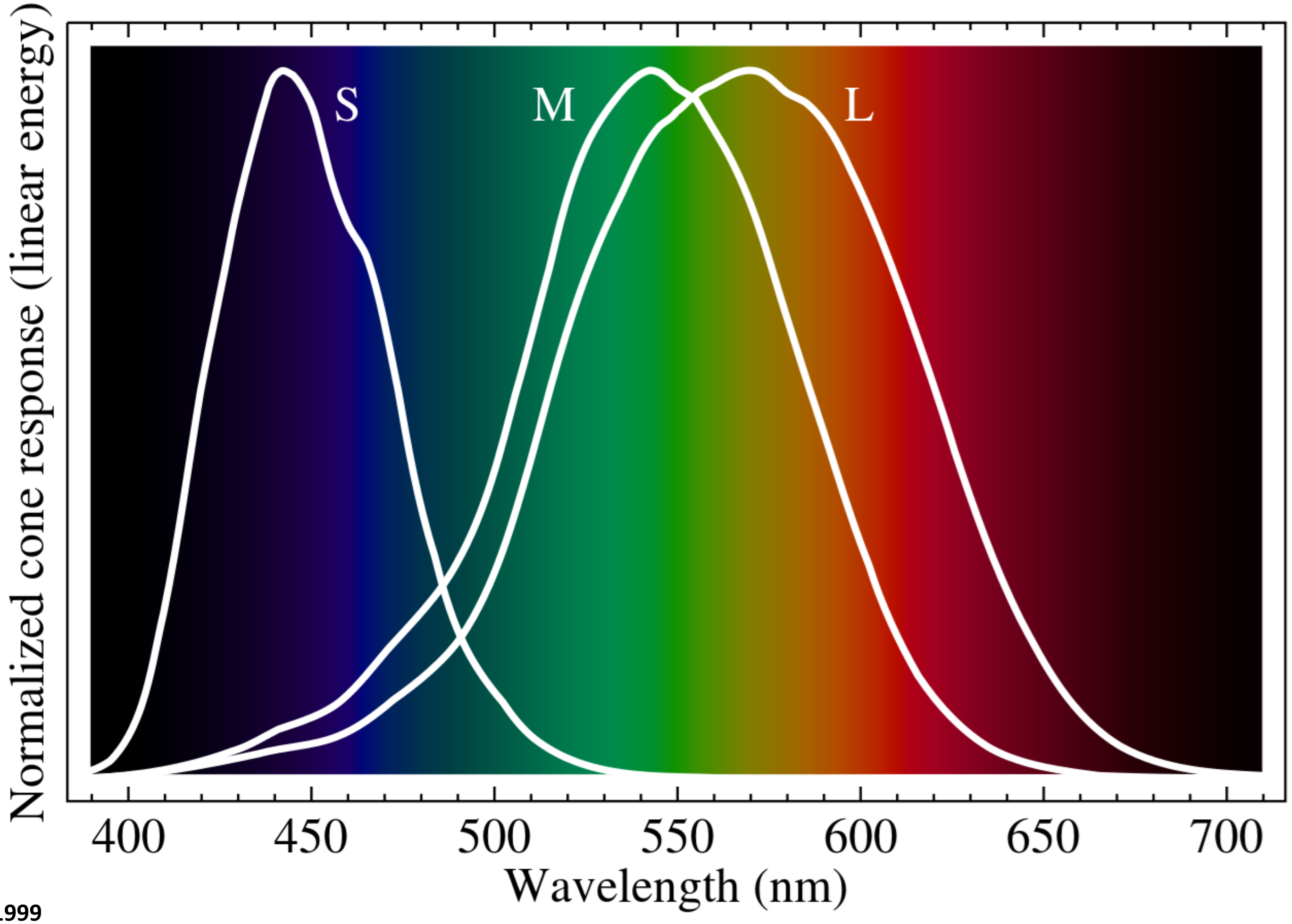
- **Cones**
  - For High luminance level (> 100 cd/m^2): **Photopic** vision
  - ~7 million per retina
  - Primarily located **in the fovea**
  - Three types of cones (S, M, L) with peak spectral sensitivities at different wavelength

Cones in Retina

- Ratio of L:M:S ≅ 40:20:1



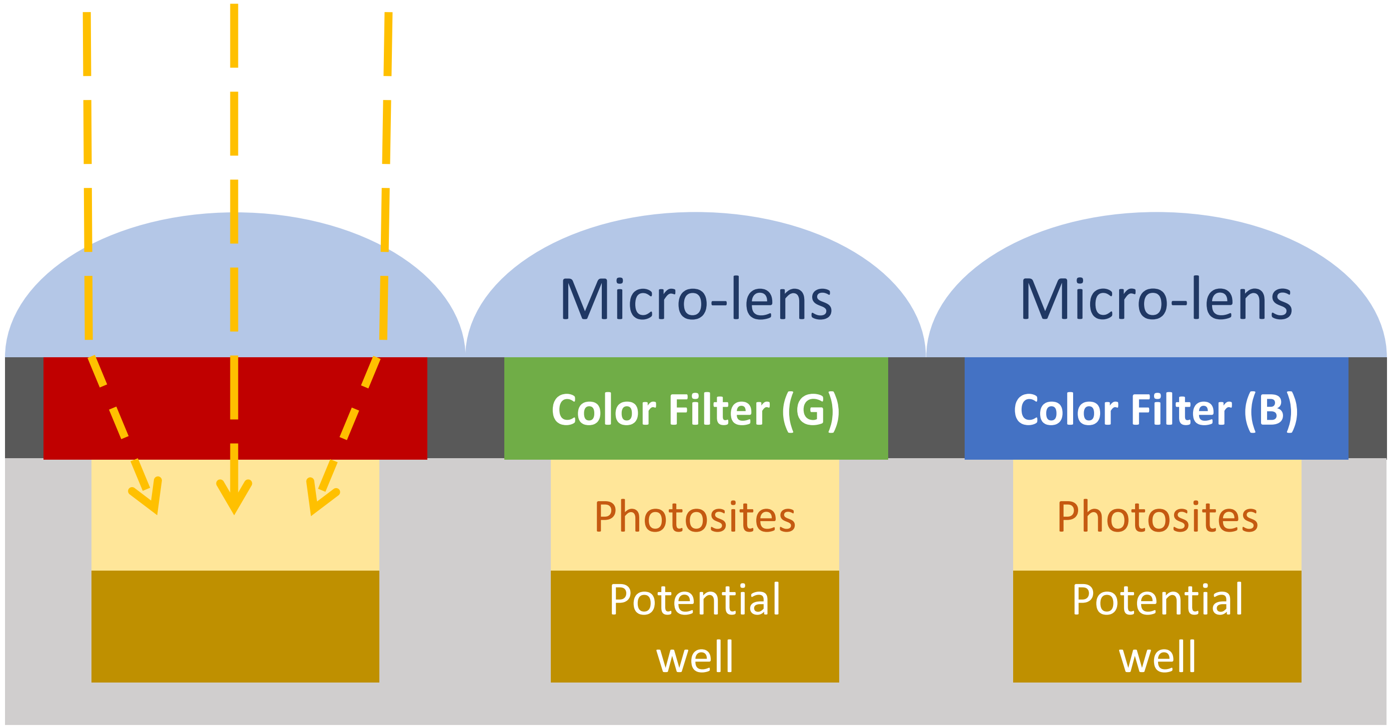
●: L ●: M ●: S

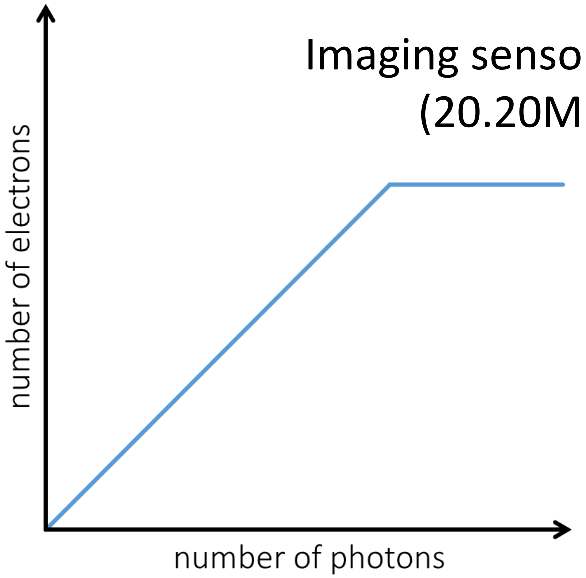


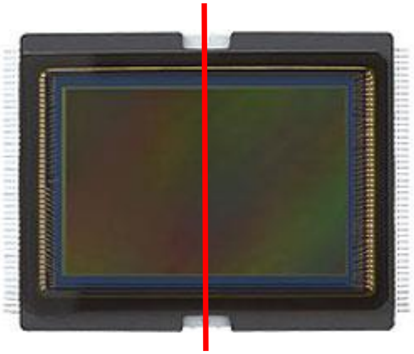
A. Roorda and D. R. Williams, The arrangement of the three cone classes in the living human eye, **Nature, 1999**
https://en.wikipedia.org/wiki/Cone_cell

Color Filter Arrays

- Color filter array (CFA) is **a mosaic of tiny color filters placed over the pixel sensors of an image sensor** to capture color information







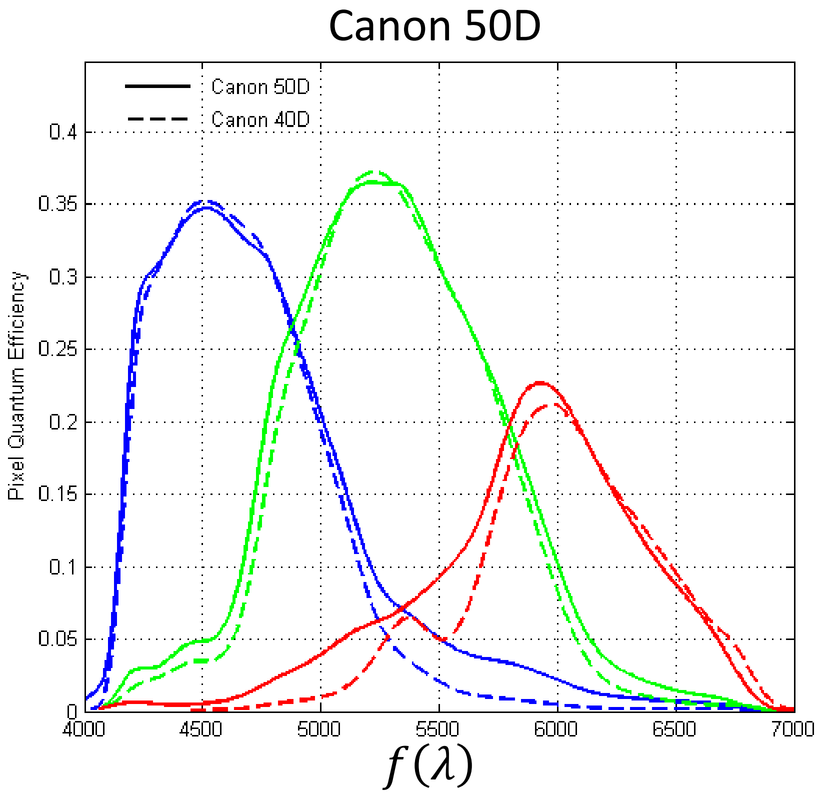
Imaging sensor: Canon 6D sensor (20.20MP, Full-frame)

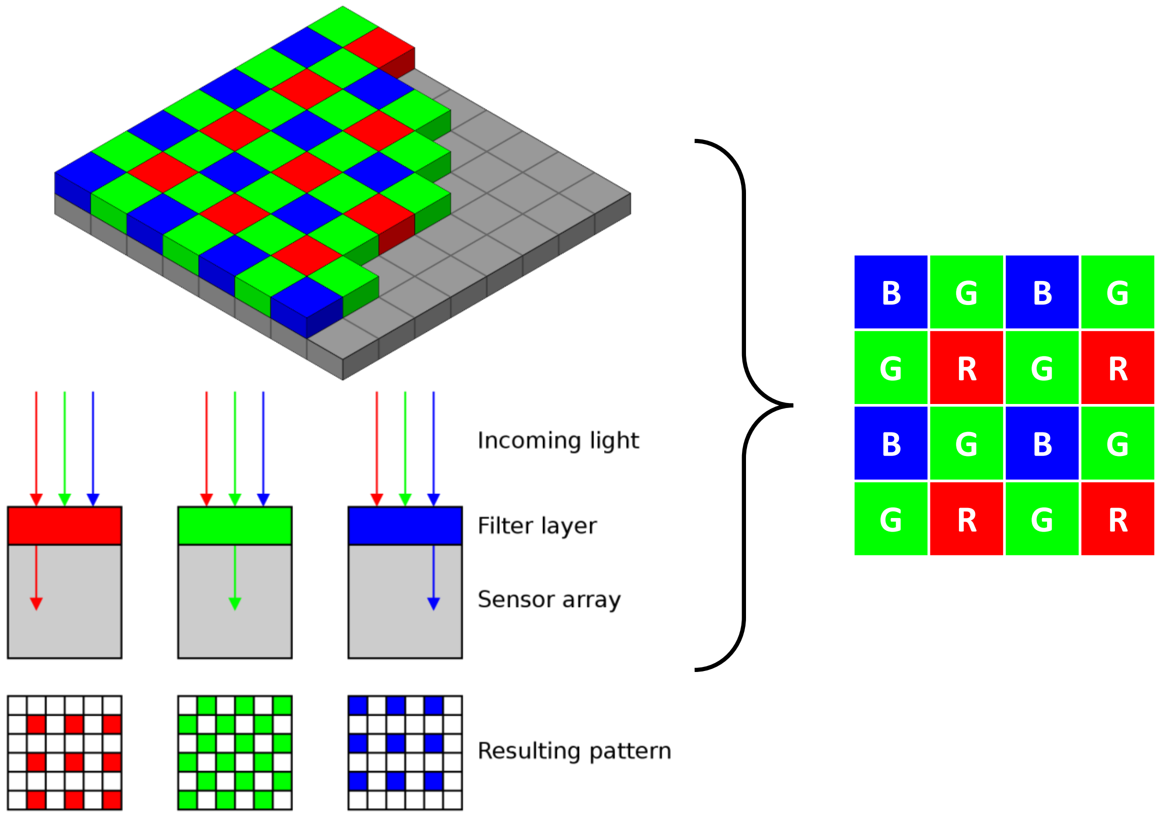
Color Filters

- **Two design choices:**

— What spectral sensitivity functions $f(\lambda)$ to use for each color filter?

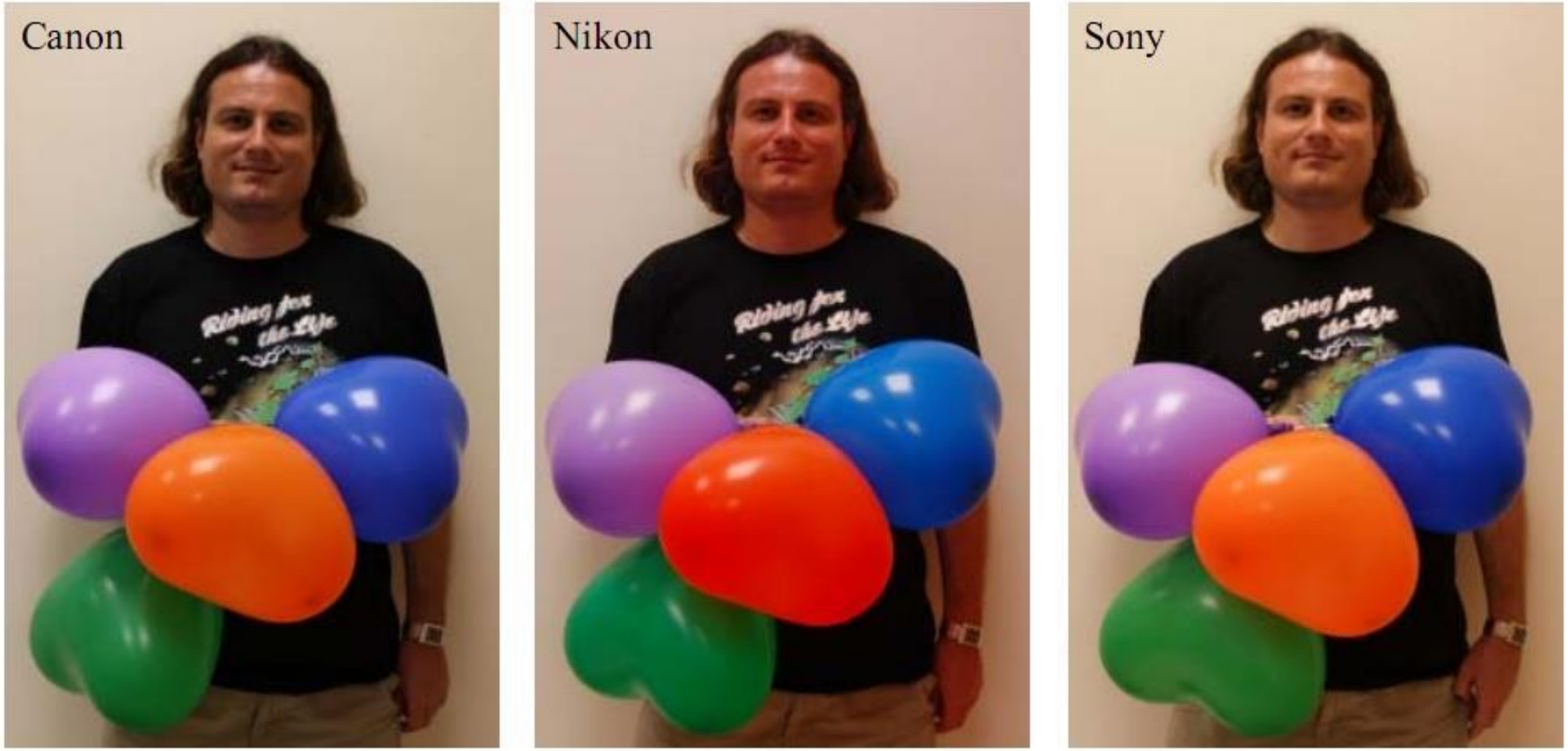
— How to spatially arrange ("**mosaic**") different color filters?





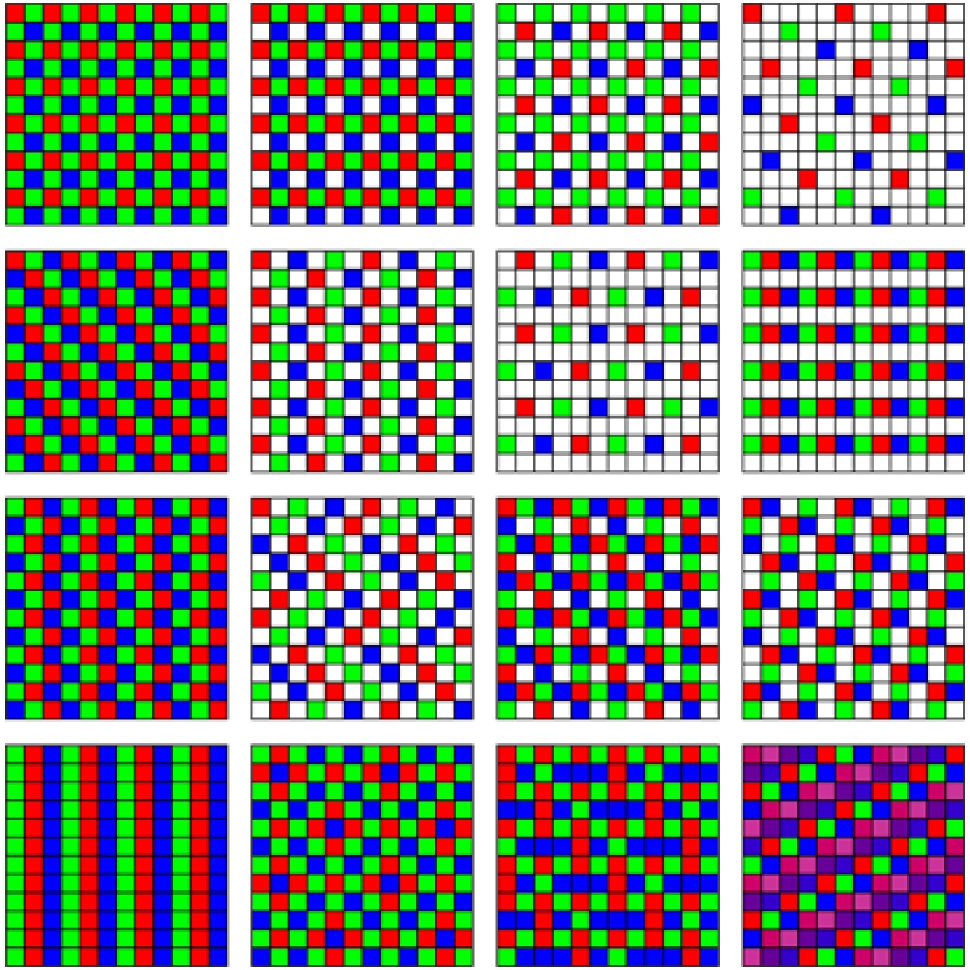
Color Filters: **Different Spectral Sensitivity Functions**

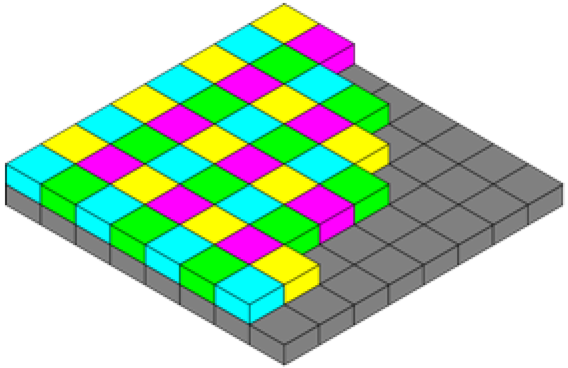
- Each camera has its more or less unique, and most of the time secret
- Same scene captured using 3 different cameras with identical settings



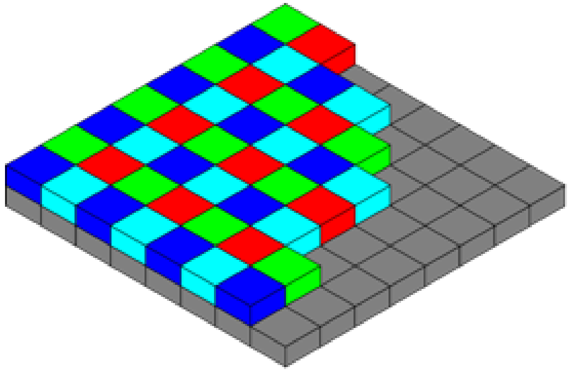
Color Filters: **Different Color Filter Arrays (CFAs)**

- Finding the **best** CFA mosaic is an active research area





CYGM
Canon IXUS, Power-shot

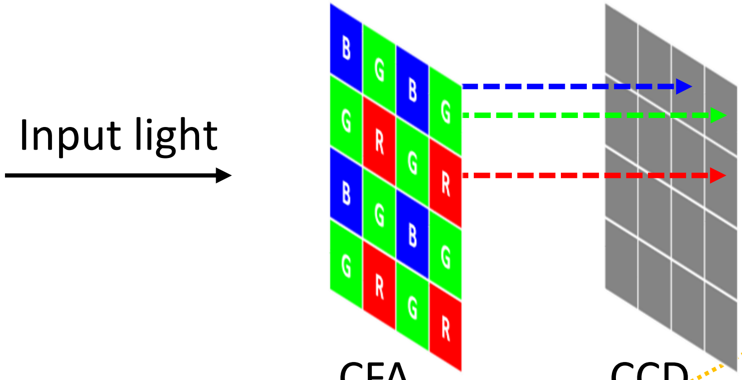


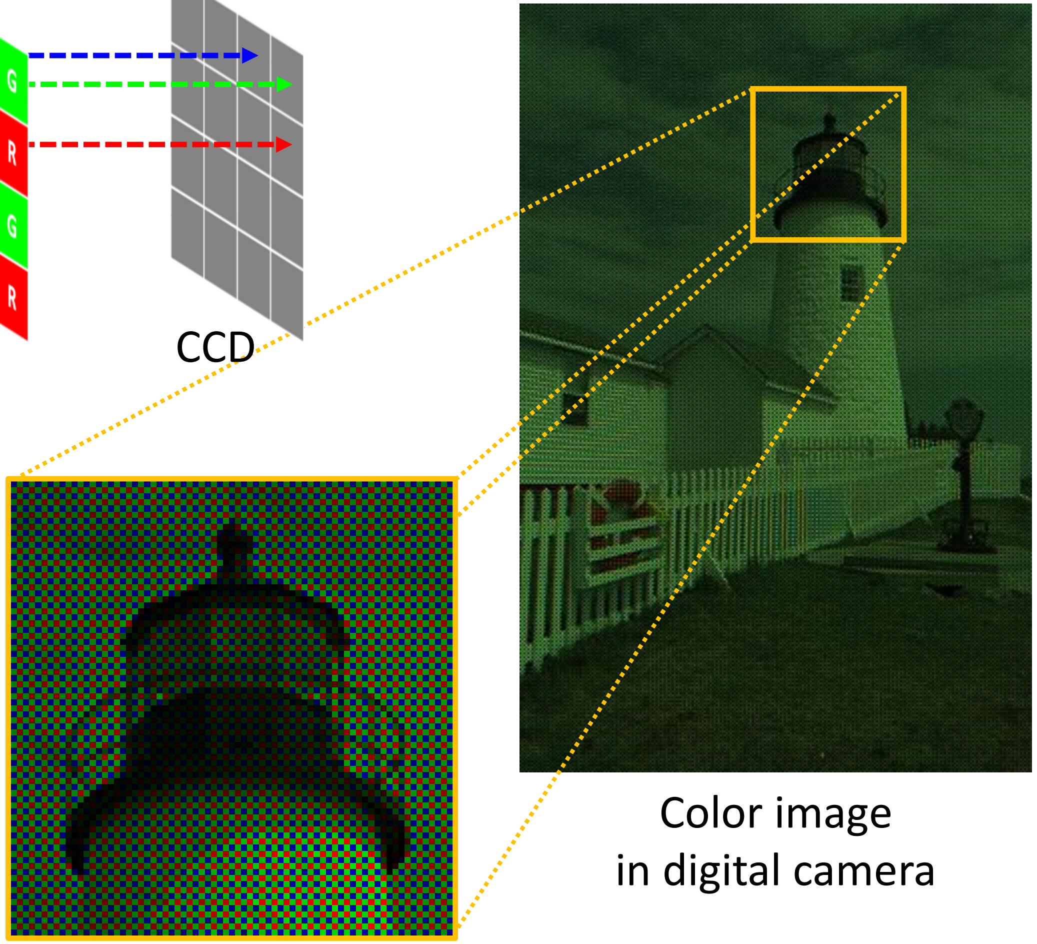
RGBE
Sony Cyber-shot

CFA Demosicing: Color Interpolation

- **Color interpolation** from neighbors to produce full RGB images from mosaicked sensor output

Scene
in real world



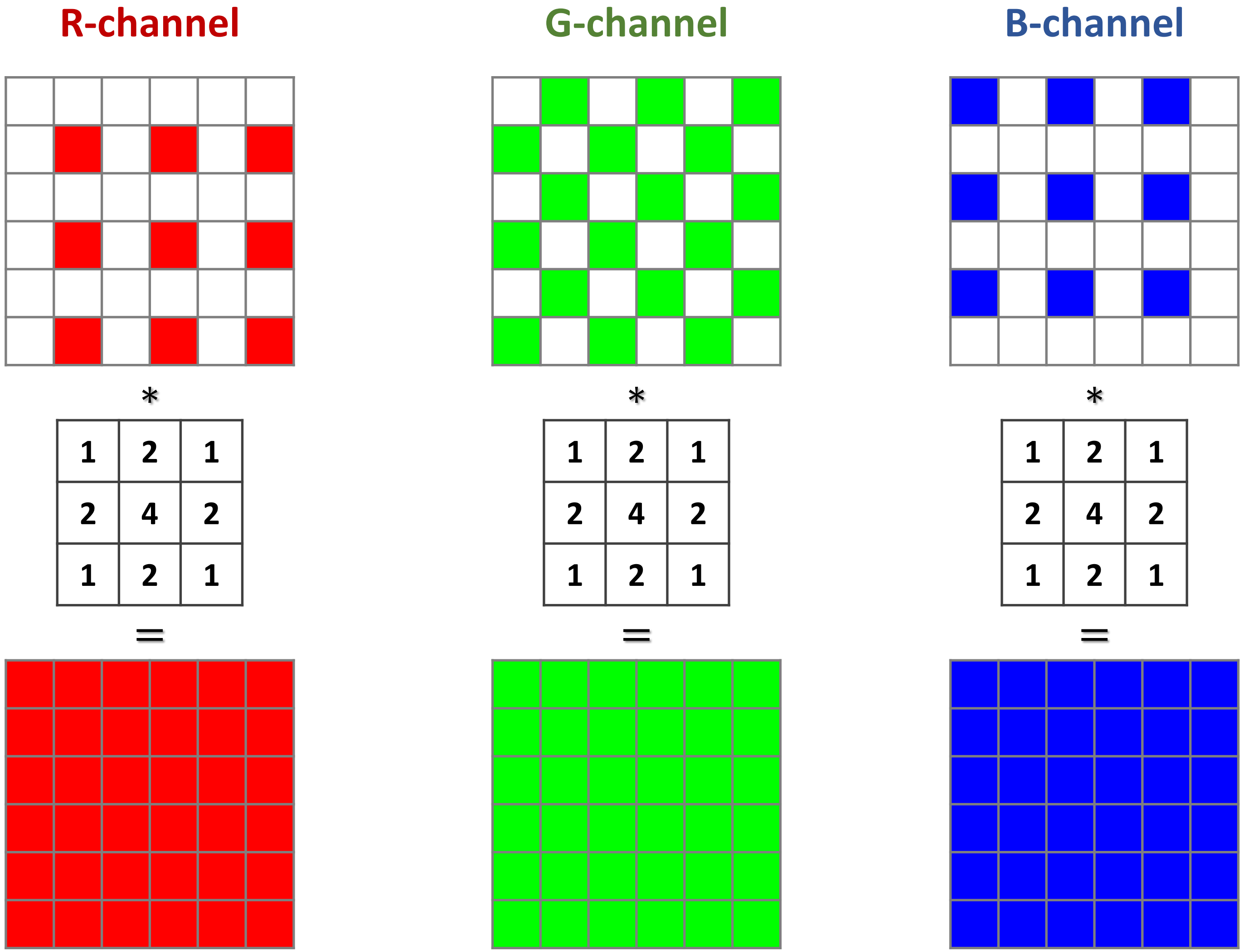


Color image
in digital camera

CFA Demosicing: Color Interpolation

- **Bilinear** interpolation

Original scene



Demosaicing result

CFA Demosicing: Visual Results

Original scene

Bilinear interpolation

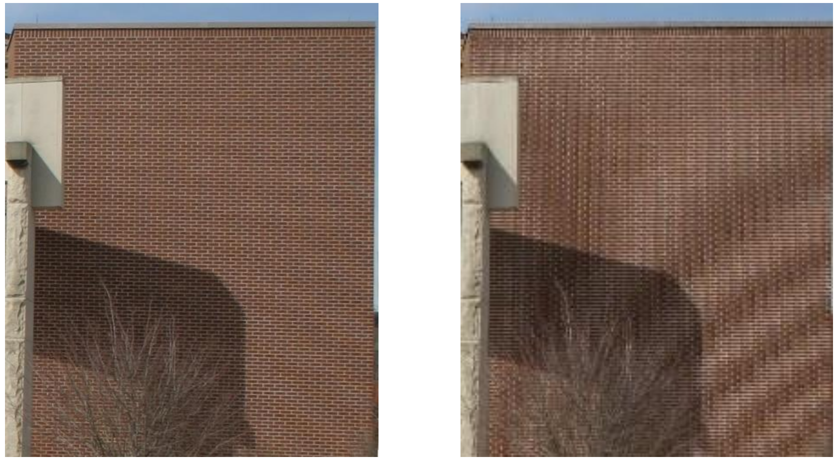
Edge-based interpolation

CFA Demosicing: **Artifacts Examples**

- Artifacts by color subsampling and filtering

Zippering artifact

False color



Aliasing

Blurring artifact

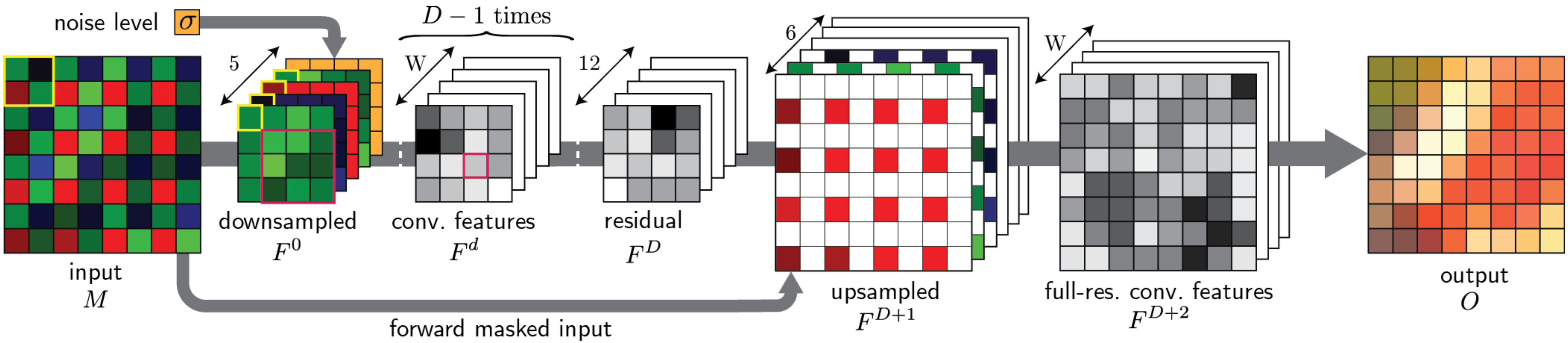
CFA Demosicing **with Deep Learning**

- Deep Learning-based Demosaicking and Denoising

M. Gharbiet al., Deep Joint Demosaicking and Denoising, **ACM TOG, 2016**

CFA Demosicing **with Deep Learning**

- Deep Learning-based Demosaicking and Denoising



M. Gharbiet al., Deep Joint Demosaicking and Denoising, **ACM TOG, 2016**

Summary: **Sampling & Color**

- In the **continuous** case, images are functions of two spatial variables
- The **discrete** case is obtained from the continuous case via **sampling** (i.e. spatial tessellation, grayscale 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

- **Color Filter Arrays (CFAs)** allow capturing of mosaiced color information; the layout of the mosaic is called **Bayer pattern**
- **Demosaicing** is the process of taking the RAW image and **interpolating missing color pixels** per channel