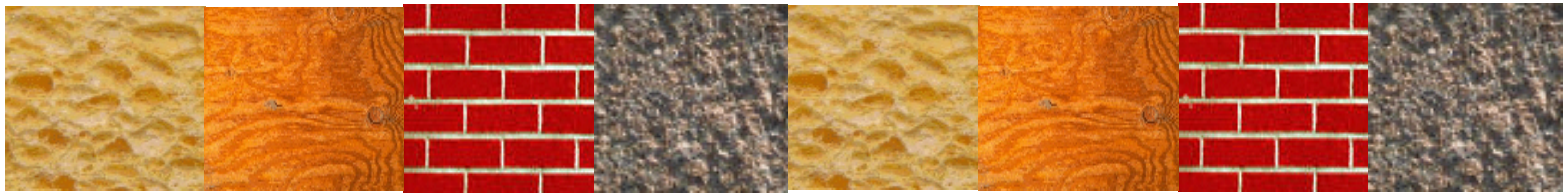
THE UNIVERSITY OF BRITISH COLUMBIA

CPSC 425: Computer Vision



Lecture 17: Texture (cont)

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

Menu for Today (**October 19, 2020**)

Topics:

- Texture Analysis
- Colour
- Colour Matching Experiments

Redings:

- **Today's** Lecture: Forsyth & Ponce (2nd ed.) 3.1-3.3
- **Next** Lecture: **Midterm**

Menu for Today (**October 19, 2020**)

Reminders:

— **Midterm** (during **class** next time on **Wednesday, October 21st**)

Also **11pm-Midnight** on **Wednesday, October 21st**

(only for remote students in timezones where exam falls at night)

Lock down browser, randomized exams, etc.

pre-Midterm Quiz (0 points, due **Tuesday by 4pm**)

— Midterm **Prep**

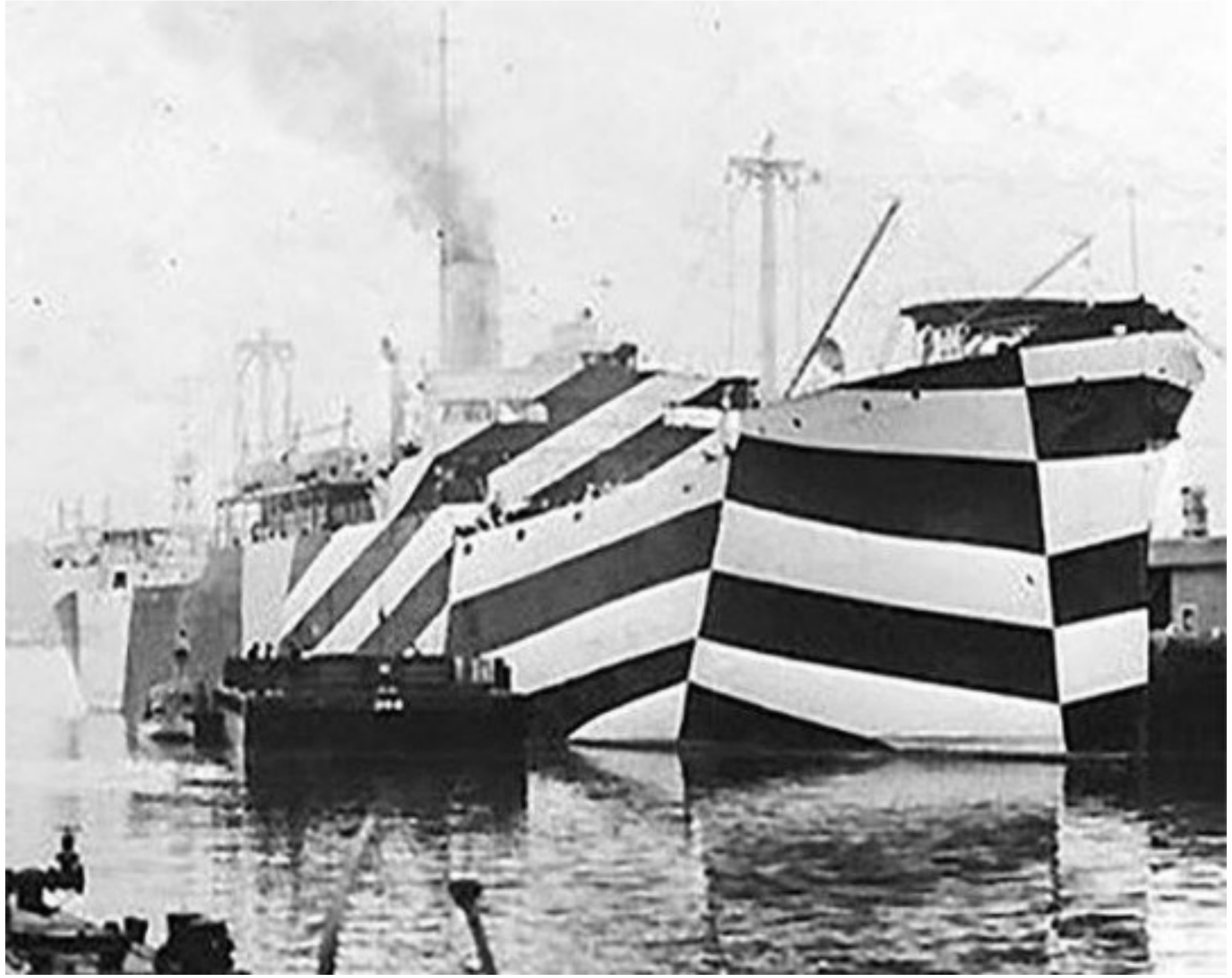
Sample questions with answers and extra slides are on **Canvas**.

Extra office hours (**today** and **Tuesday**)

— **Assignment 3**: Texture Synthesis due **Monday, October 26th**

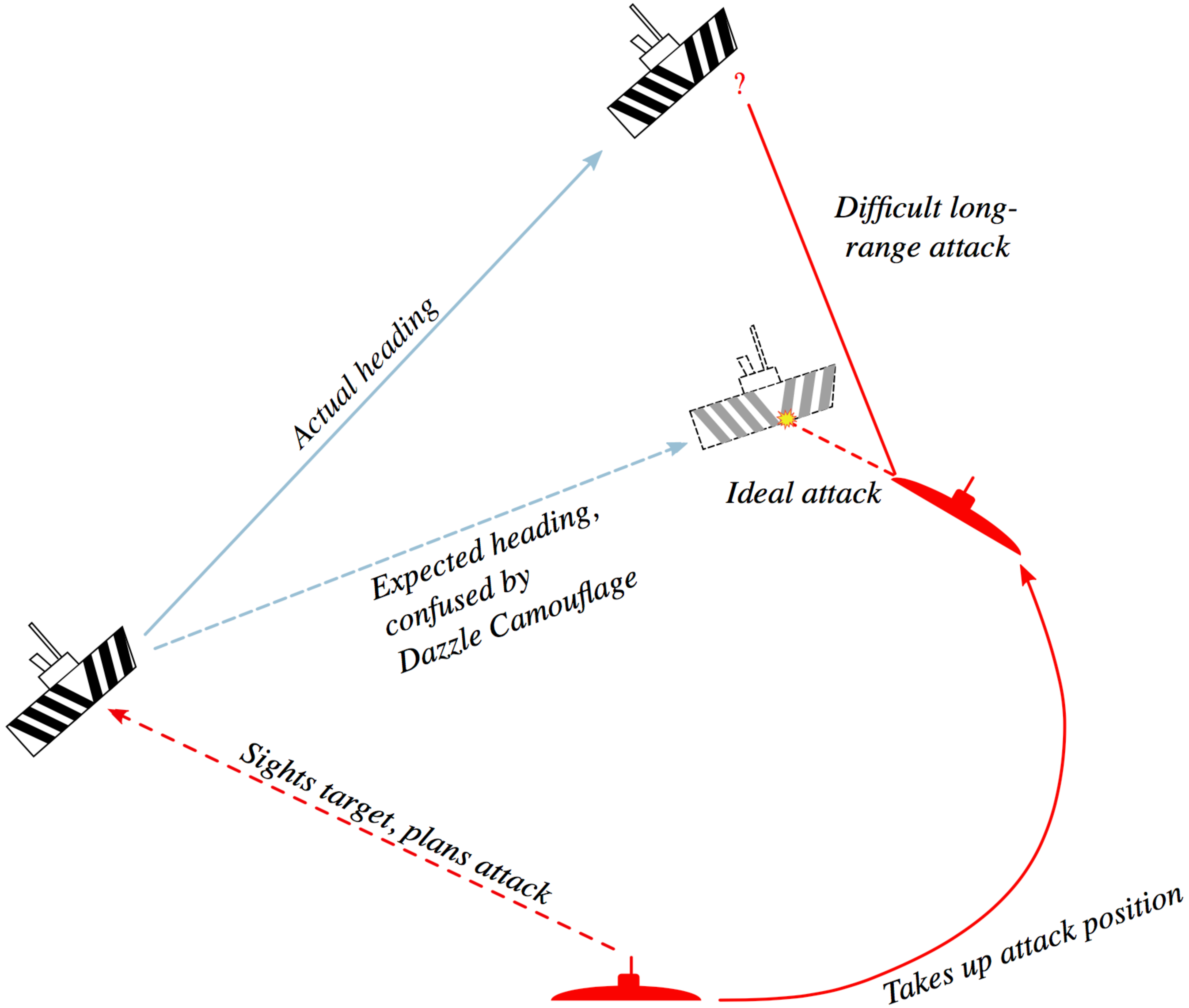
Today's "**fun**" Example: Dazzle Camouflage

A type of ship camouflage that uses strongly contrasted colours and shapes to make it difficult to estimate the ship's speed and heading



Today's "**fun**" Example: Dazzle Camouflage

A type of ship camouflage that uses strongly contrasted colours and shapes to make it difficult to estimate the ship's speed and heading



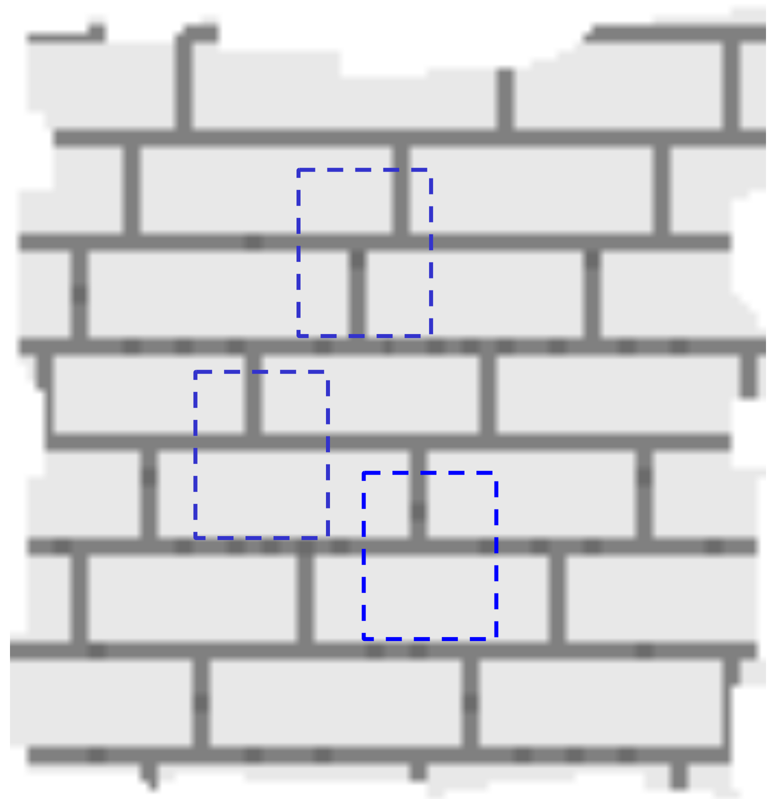
Lecture 16: Re-cap of Texture

We will look at two main questions:

1. How do we represent texture?
   → Texture **analysis**

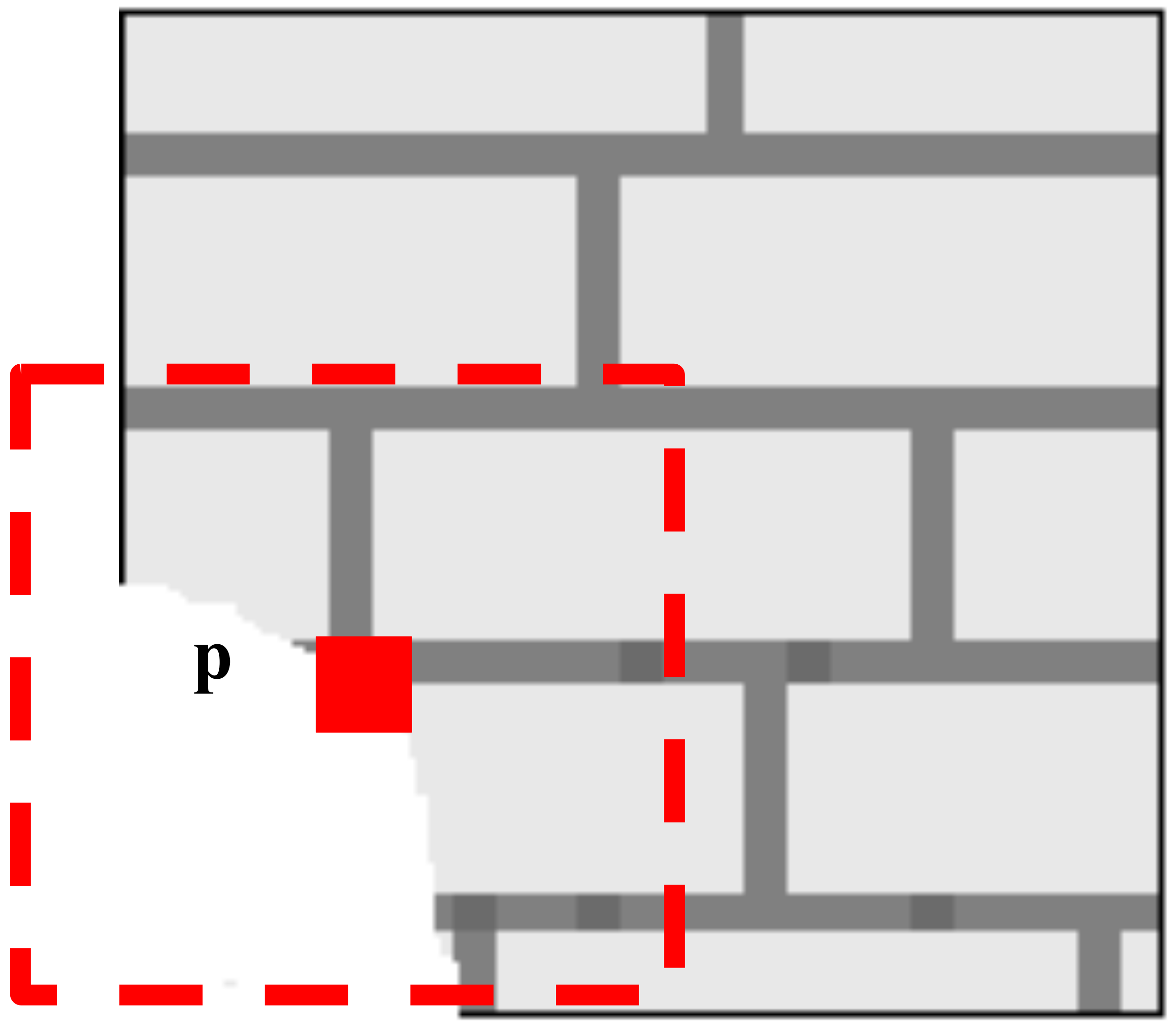
2. How do we generate new examples of a texture?
   → Texture **synthesis**

Lecture 16: Re-cap of Texture Synthesis



Infinite sample image

SAMPLE →

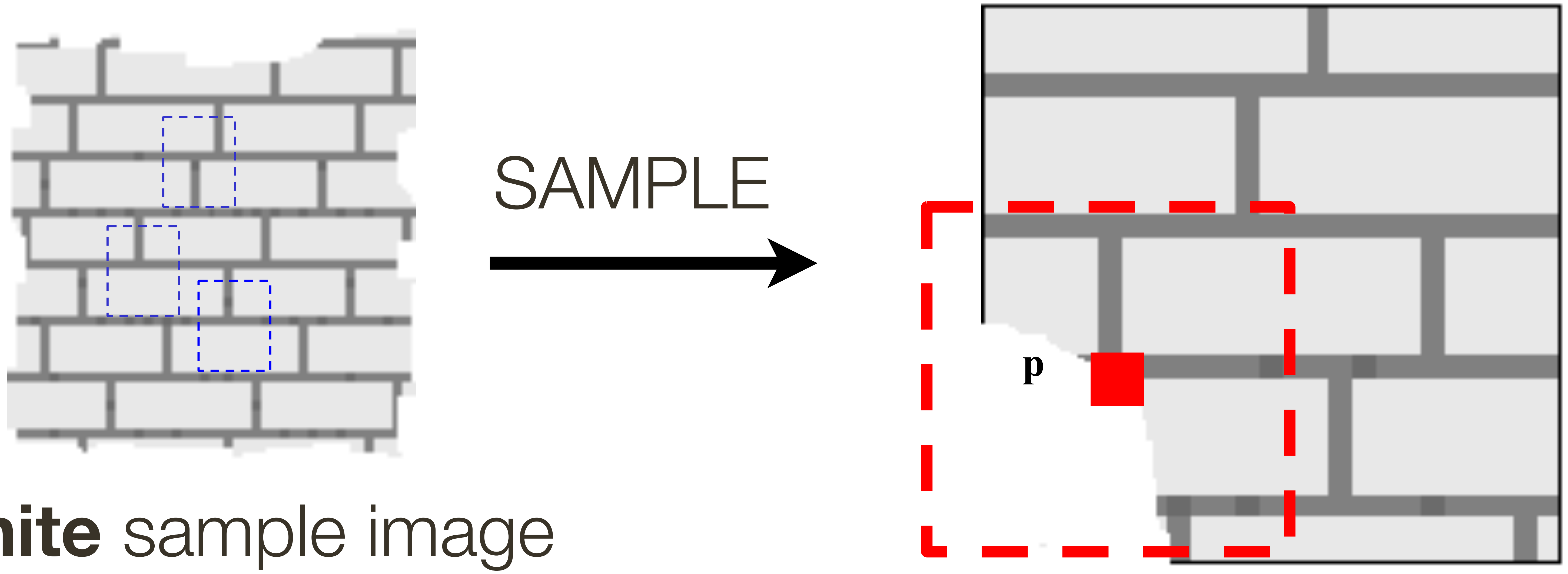


— What is **conditional** probability distribution of *p*, given the neighbourhood window?

— Directly search the input image for all such neighbourhoods to produce a **histogram** for *p*

— To **synthesize** *p*, pick one match at random

Lecture 16: Re-cap of Texture Synthesis

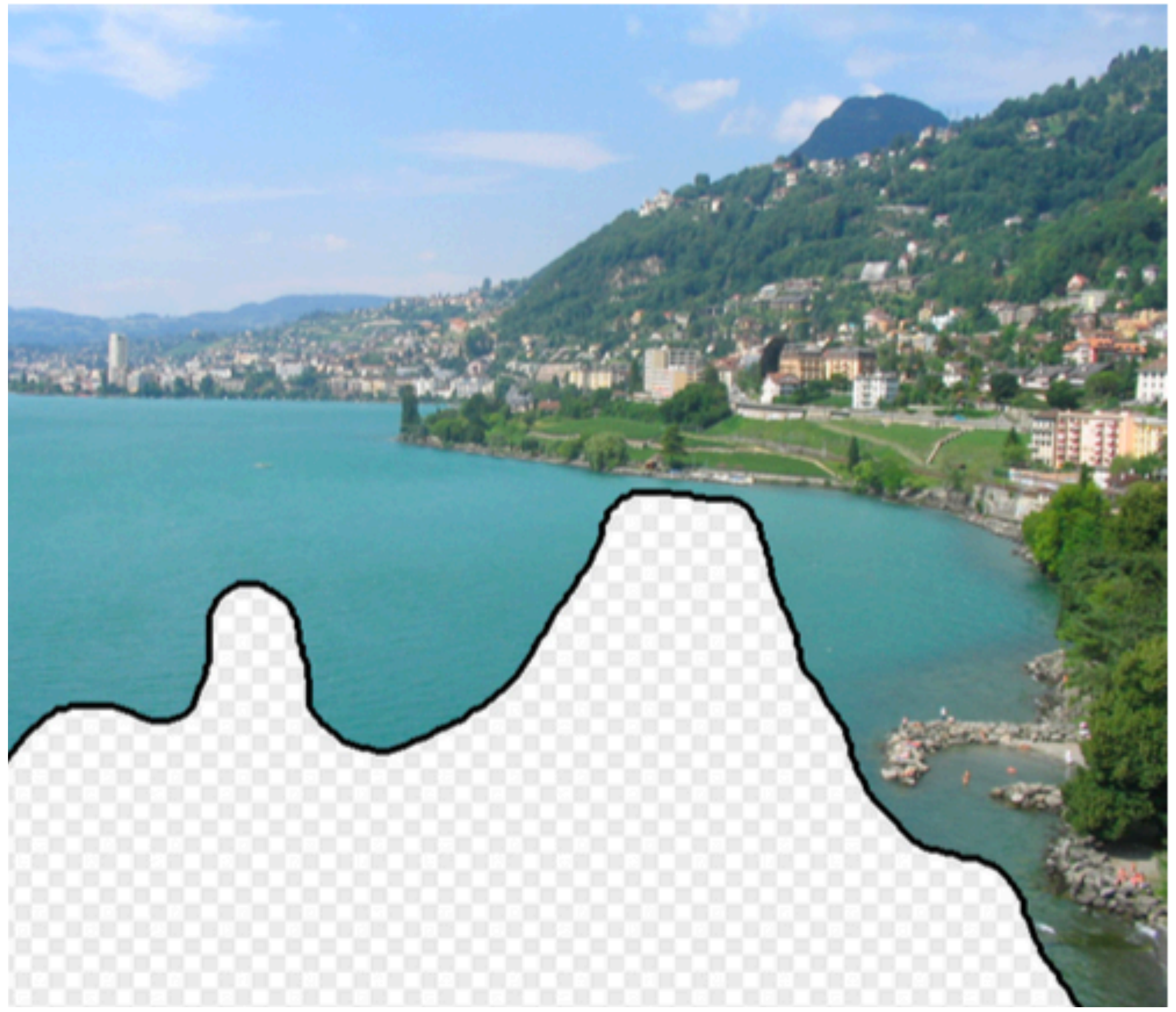


Infinite sample image

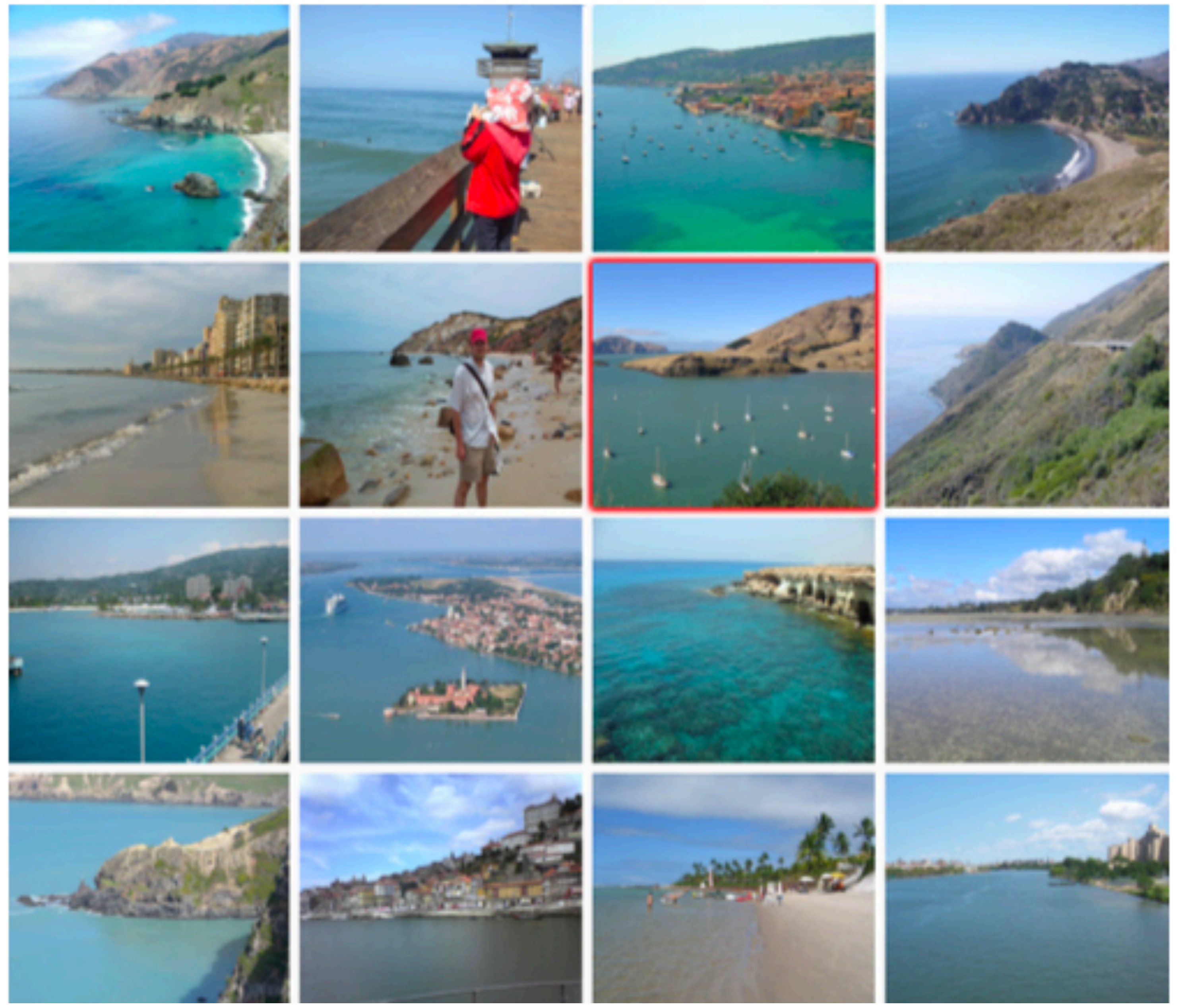
— Since the sample image is finite, an exact neighbourhood match might not be present

— Find the **best match** using SSD error, weighted by Gaussian to emphasize local structure, and take all samples within some distance from that match

Lecture 16: Re-cap of Texture Synthesis



Input



Scene Matches

Output

Figure Credit: Hays and Efros 2007

Lecture 16: Texture Representation

Observation: Textures are made up of generic sub-elements, repeated over a region with similar statistical properties

Idea: Find the sub-elements with filters, then represent each point in the image with a summary of the pattern of sub-elements in the local region

Question: What filters should we use?

Answer: Human vision suggests spots and oriented edge filters at a variety of different orientations and scales

Lecture 16: Texture Representation

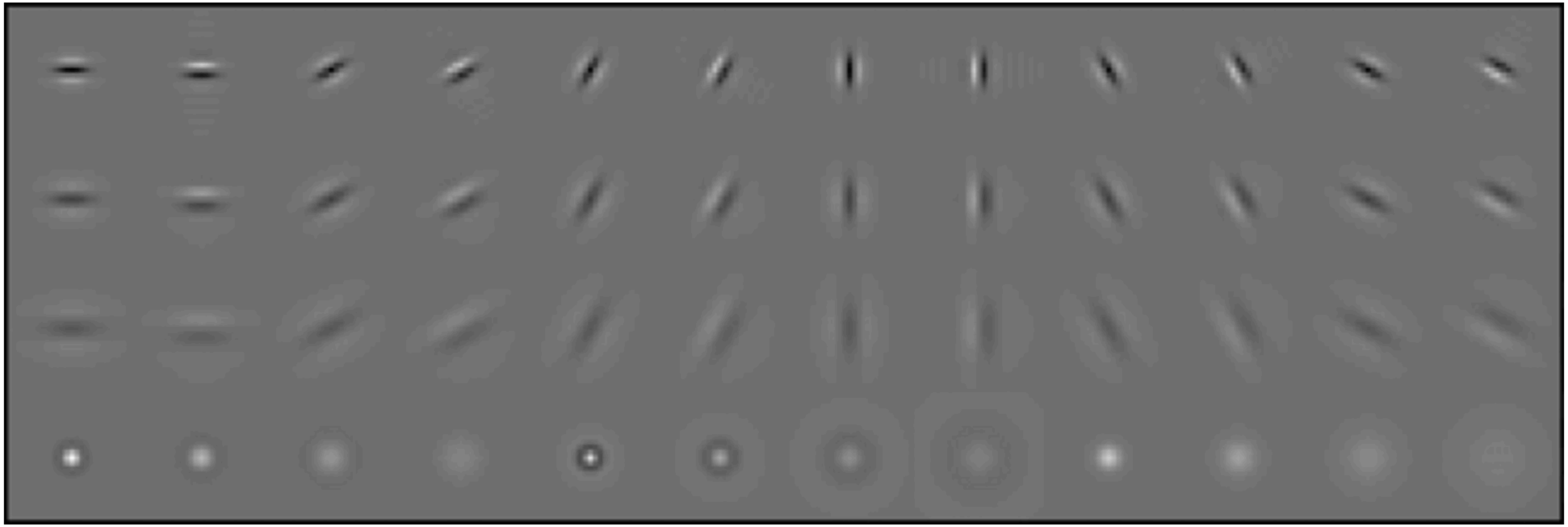


Figure Credit: Leung and Malik, 2001

Texture **Representation**

Observation: Textures are made up of generic sub-elements, repeated over a region with similar statistical properties

Idea: Find the sub-elements with filters, then represent each point in the image with a summary of the pattern of sub-elements in the local region

Question: What filters should we use?

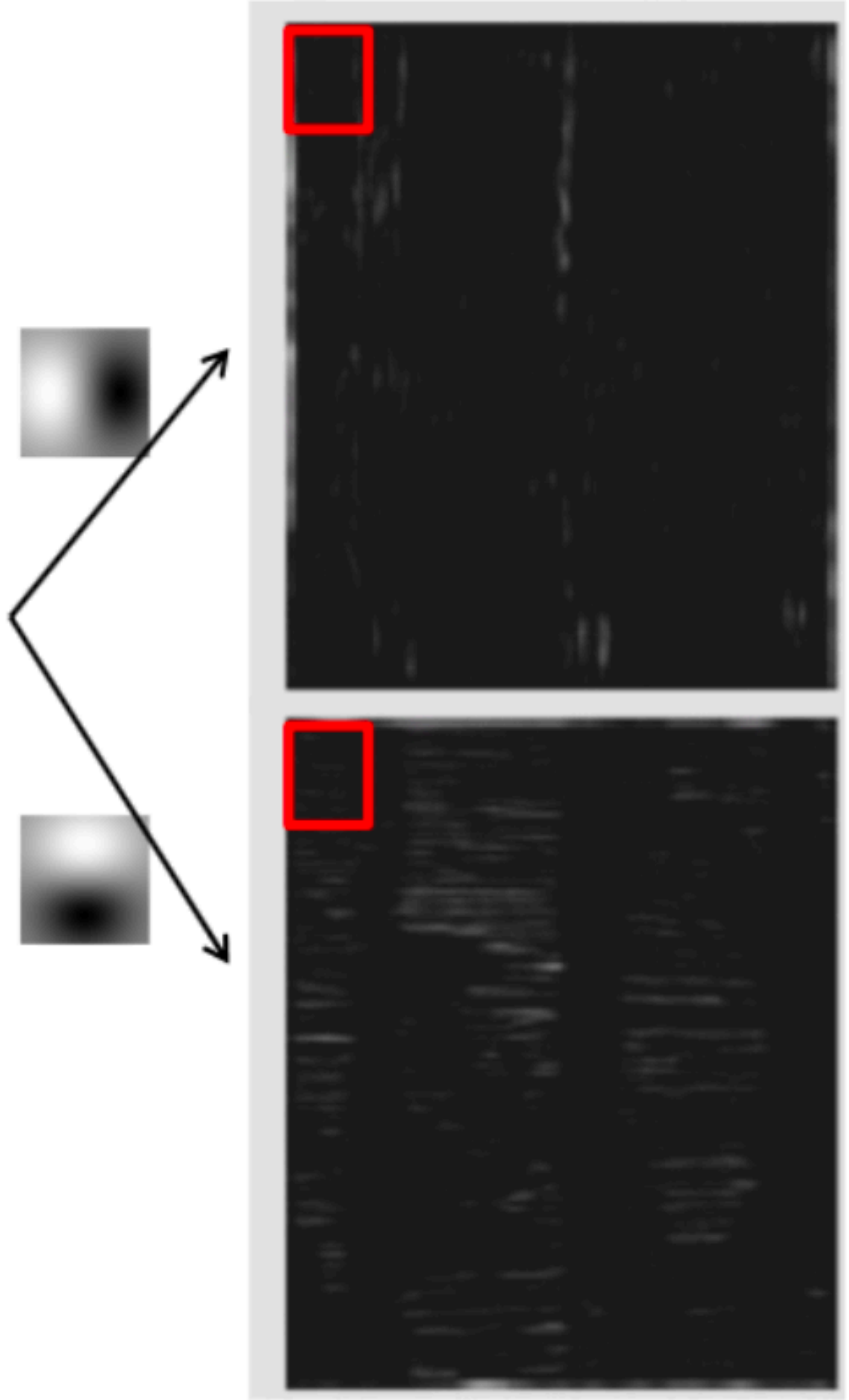
Answer: Human vision suggests spots and oriented edge filters at a variety of different orientations and scales

Question: How do we "summarize"?

Answer: Compute the mean or maximum of each filter response over the region
— Other statistics can also be useful

Texture **Representation**

original image



derivative filter responses, squared

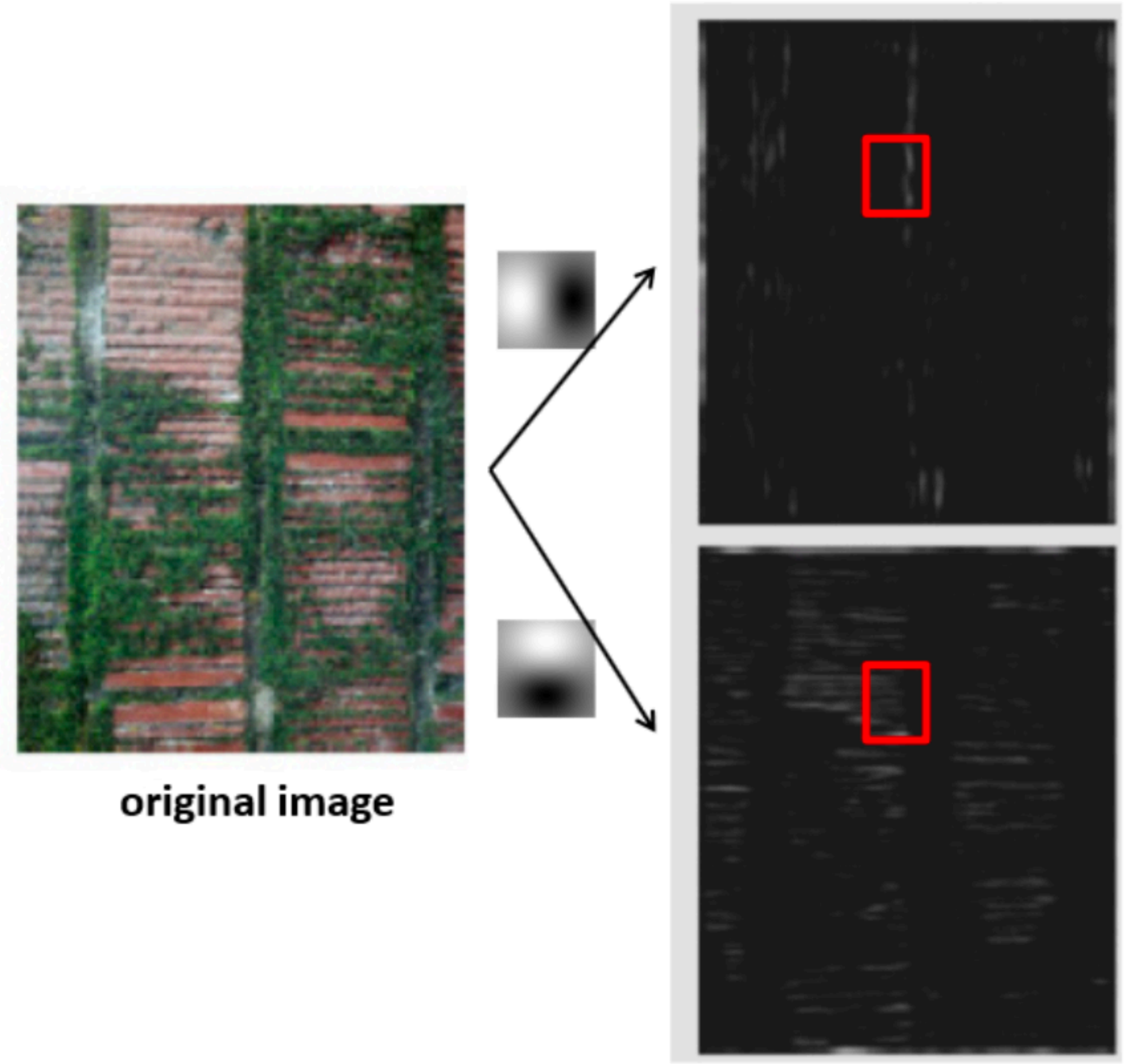
	mean d/dx value	mean d/dy value
Win. #1	4	10

⋮

statistics to summarize patterns in small windows

Slide Credit: Trevor Darrell

Texture **Representation**



original image

derivative filter responses, squared

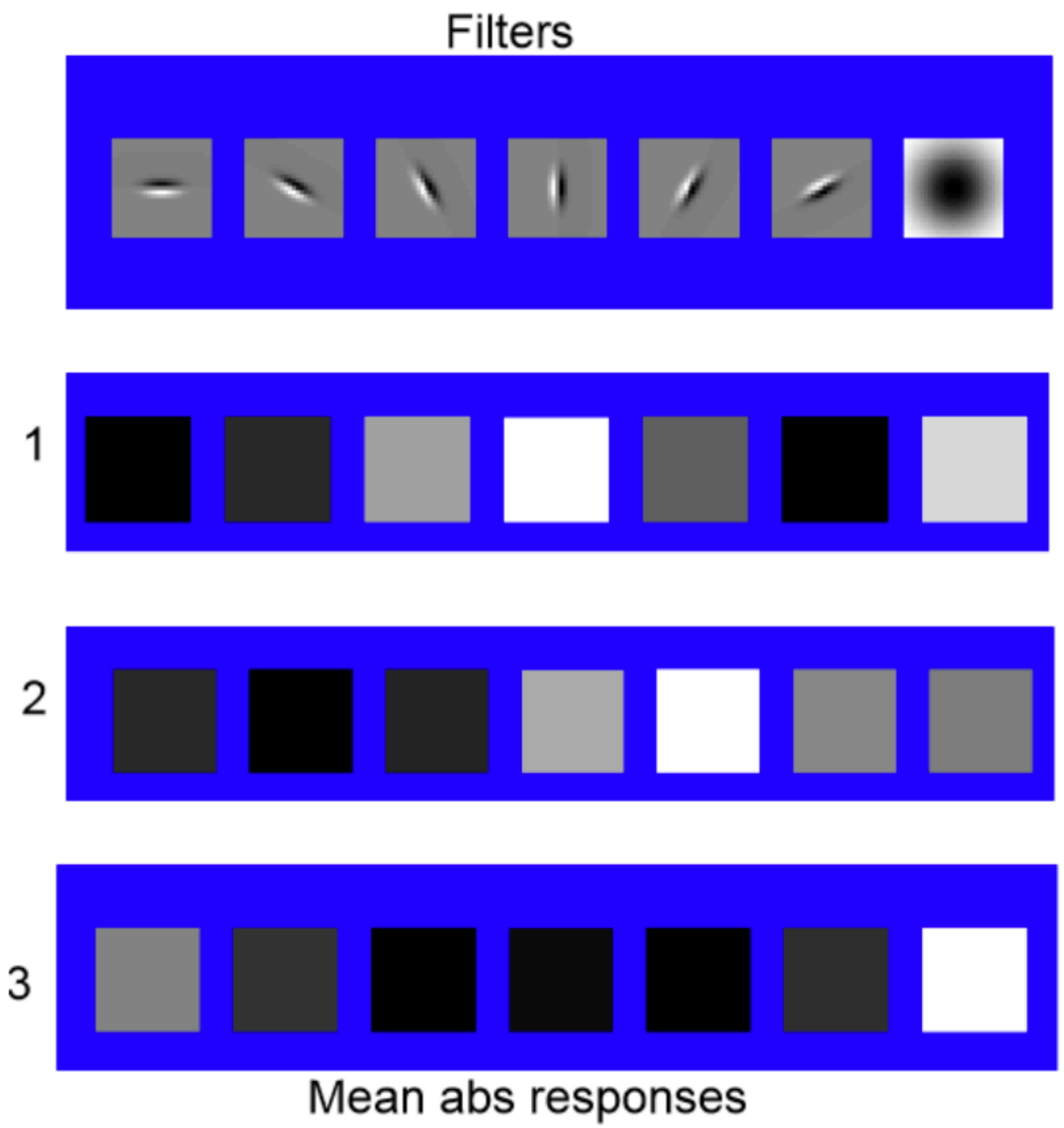
	mean d/dx value	mean d/dy value
Win. #1	4	10
Win.#2 ⋮	18	7
Win.#9	20	20

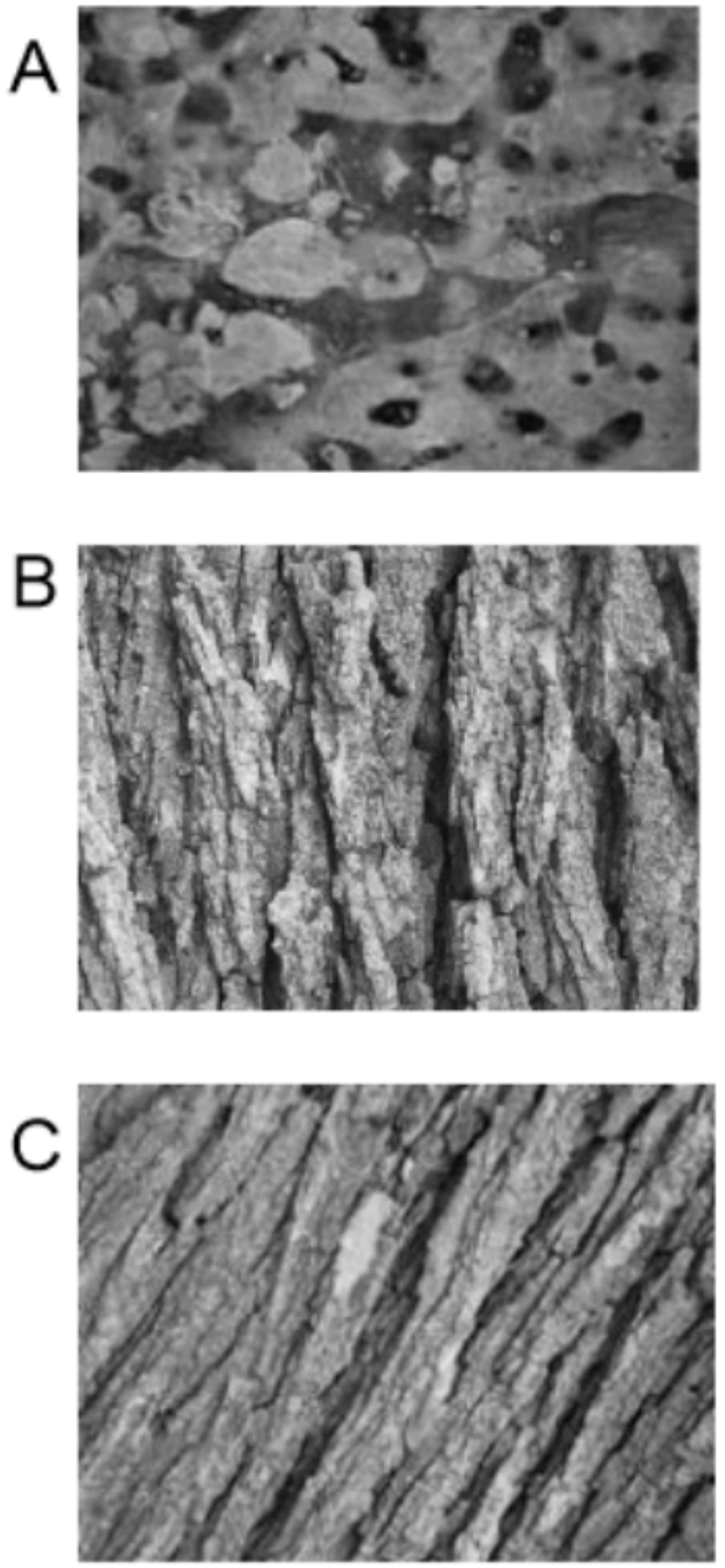
⋮

statistics to summarize patterns in small windows

Slide Credit: Trevor Darrell

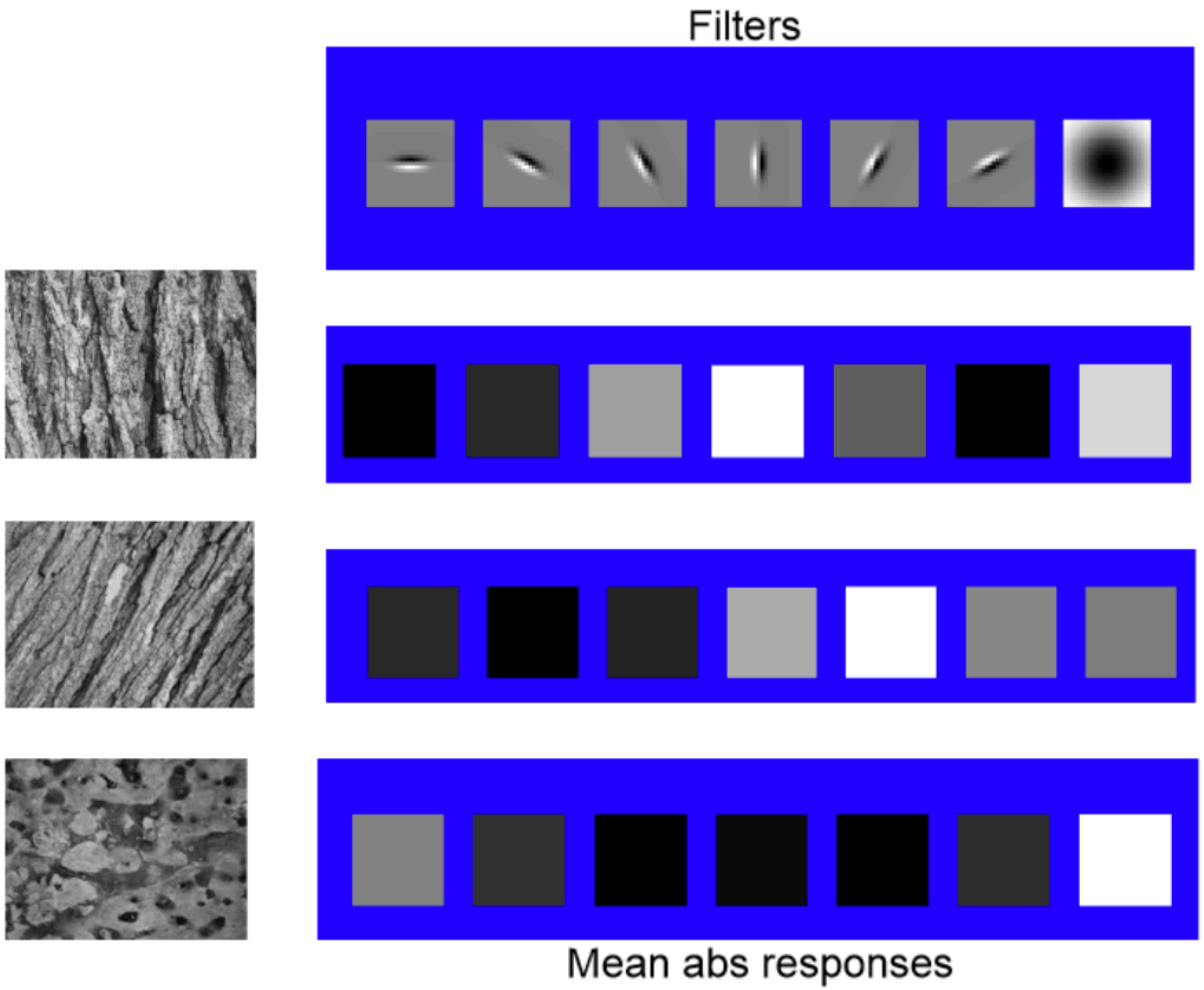
A Short **Exercise**: Match the texture to the response





Slide Credit: James Hays

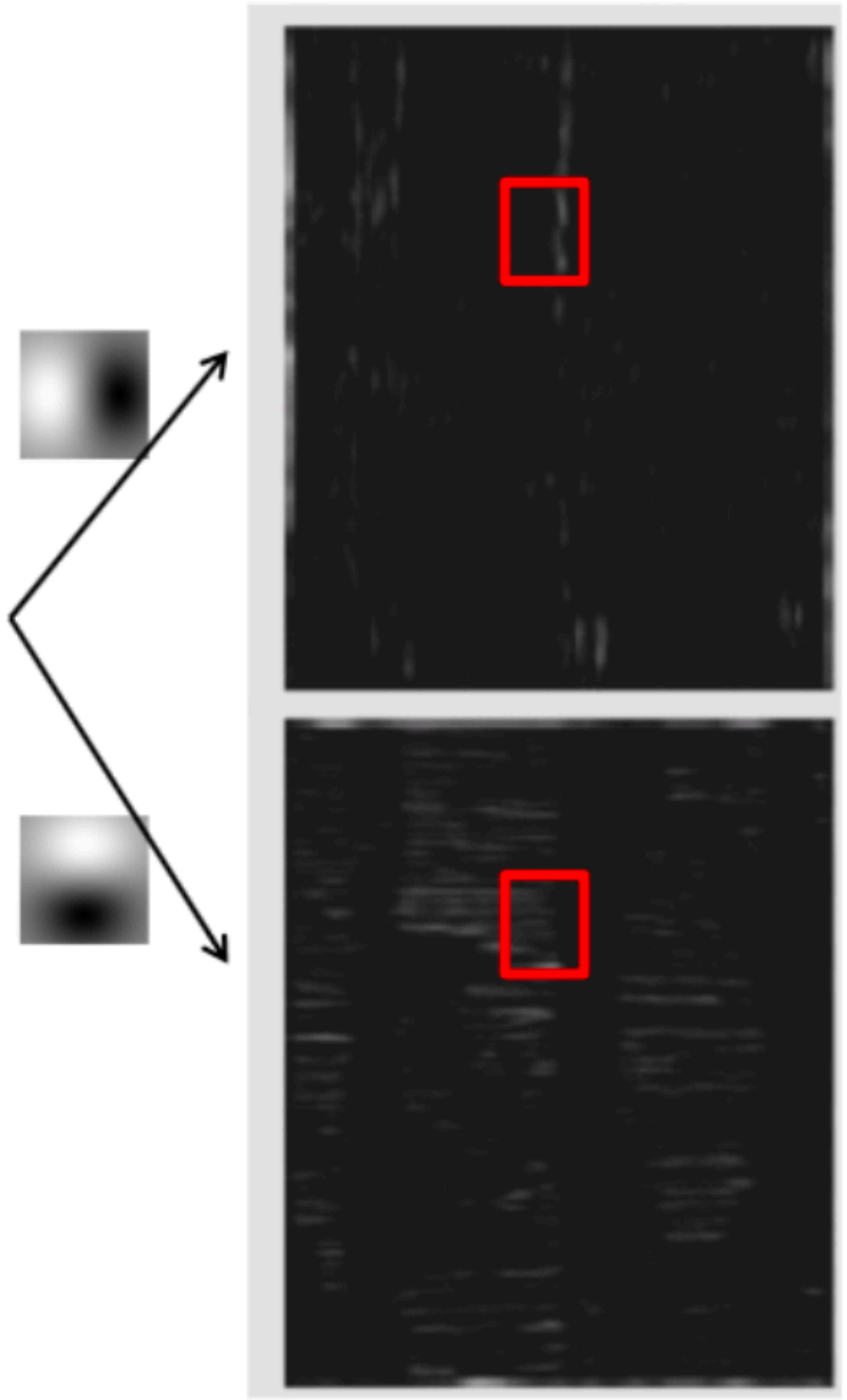
A Short **Exercise**: Match the texture to the response



Slide Credit: James Hays

Texture **Representation**

original image



derivative filter responses, squared

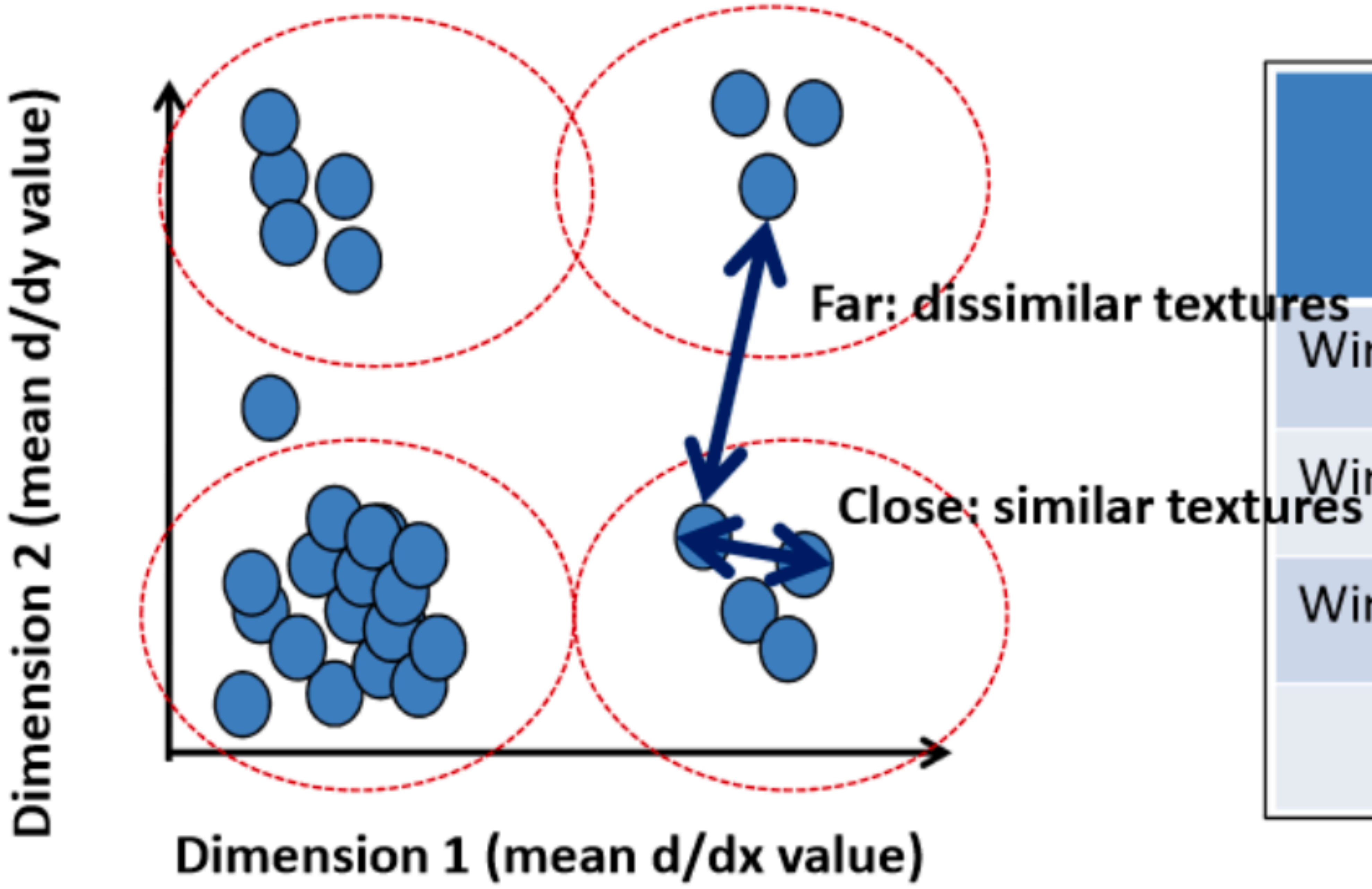
	<u>mean d/dx value</u>	<u>mean d/dy value</u>
Win. #1	4	10
Win.#2	18	7
⋮		
Win.#9	20	20

⋮

statistics to summarize patterns in small windows

Slide Credit: Trevor Darrell

Texture **Representation**



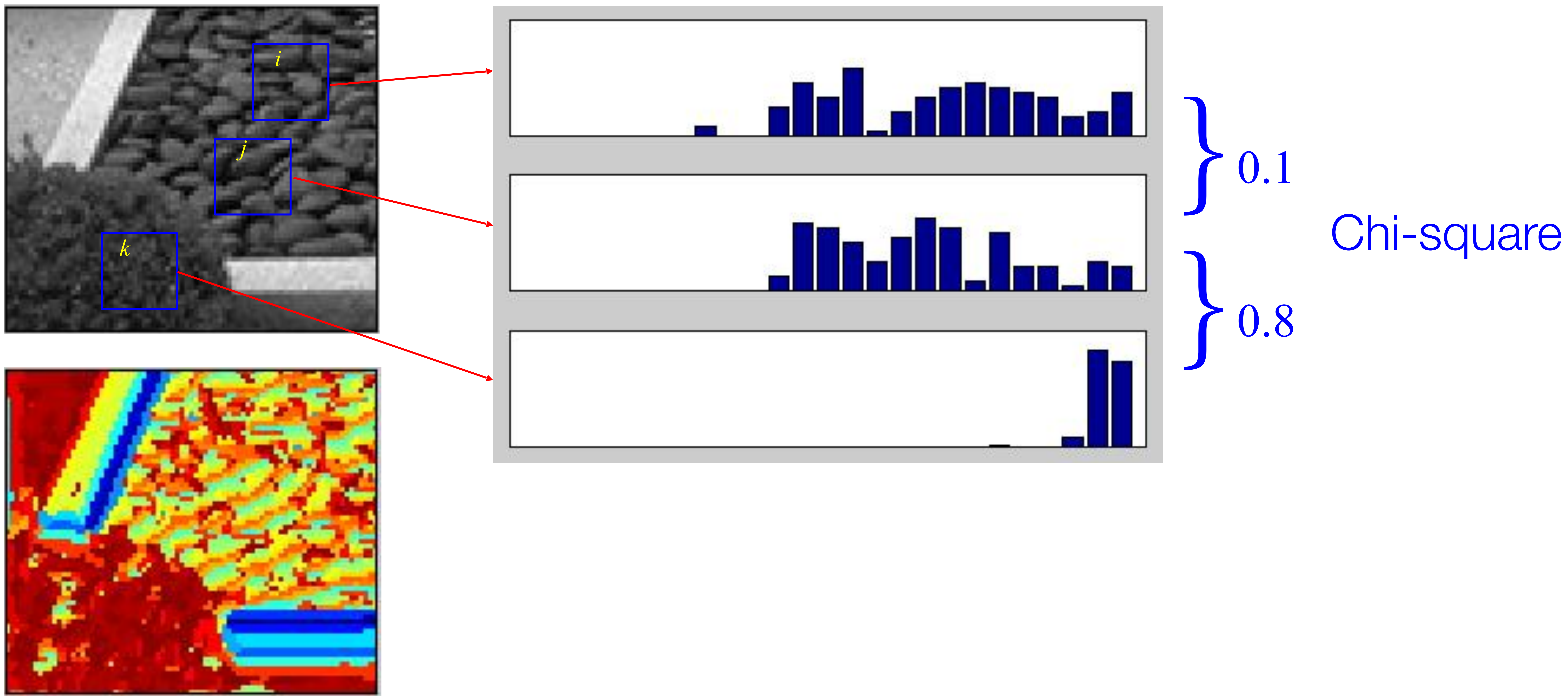
	mean d/dx value	mean d/dy value
Win. #1	4	10
Win.#2	18	7
⋮		
Win.#9	20	20

⋮

statistics to summarize patterns in small windows

Slide Credit: Trevor Darrell

Texture **Representation**



Bag-of-Words Representation

Take a large **corpus of text**:

Bag-of-Words Representation

Take a large **corpus of text**:

— Represent every **letter** by a 26 dimensional (unit) vector

$$a = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ 0 \end{bmatrix} \qquad b = \begin{bmatrix} 0 \\ 1 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ 0 \end{bmatrix}$$

Bag-of-Words Representation

Take a large **corpus of text**:

— Represent every **letter** by a 26 dimensional (unit) vector

— Represent each **word** by an average of letter representations in it

$$ab = \begin{bmatrix} \frac{1}{2} \\ \frac{1}{2} \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ 0 \end{bmatrix}$$

Bag-of-Words Representation

Take a large **corpus of text**:

— Represent every **letter** by a 26 dimensional (unit) vector

— Represent each **word** by an average of letter representations in it

— Cluster the words, to get a “**dictionary**”. Words that have very similar representations would get clustered together (e.g., smile and smiled)

Bag-of-Words Representation

Take a large **corpus of text**:

— Represent every **letter** by a 26 dimensional (unit) vector

— Represent each **word** by an average of letter representations in it

— Cluster the words, to get a “**dictionary**”. Words that have very similar representations would get clustered together (e.g., smile and smiled)

— Now represent every document by **histogram** of “dictionary” atoms by associating every word to an atom that is closest in terms of distance in 26D

Bag-of-Words Representation

Take a large **corpus of text**:

— Represent every **letter** by a 26 dimensional (unit) vector

— Represent each **word** by an average of letter representations in it

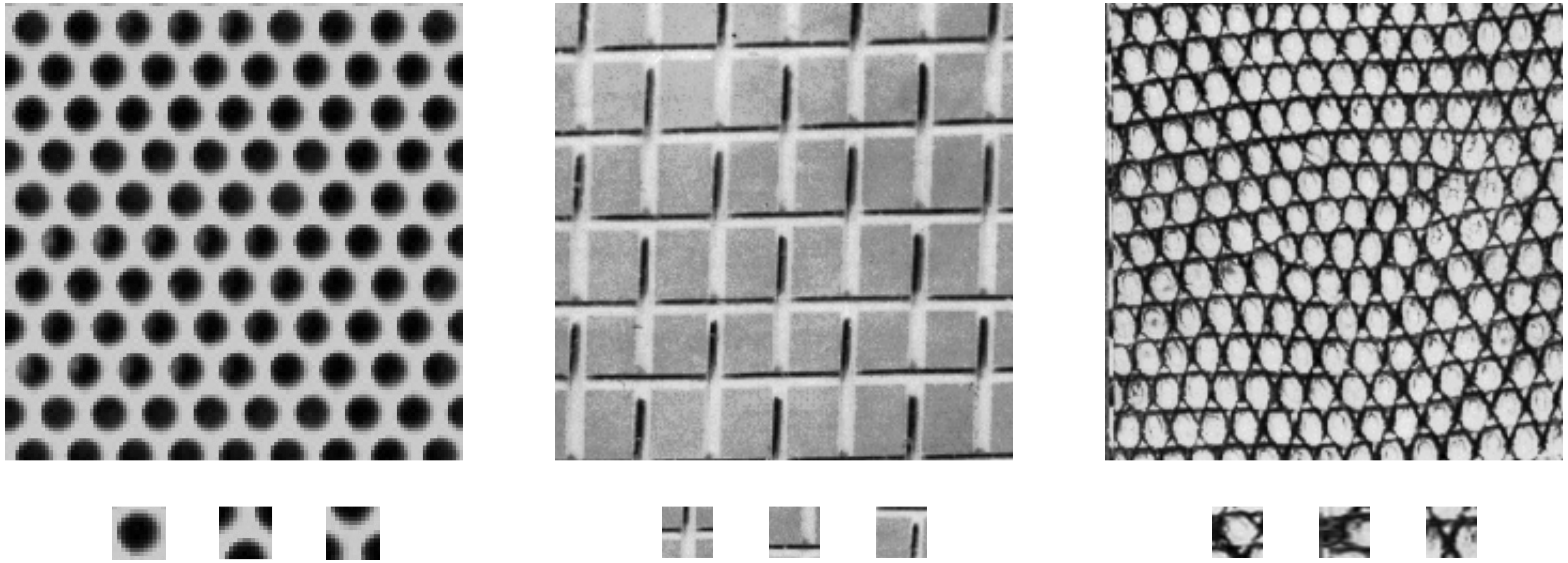
— Cluster the words, to get a "**dictionary**". Words that have very similar representations would get clustered together (e.g., smile and smiled)

— Now represent every document by **histogram** of "dictionary" atoms by associating every word to an atom that is closest in terms of distance in 26D

corpus of text = collection of images
letter = pixel location
word = patch with pixel in the center
dictionary = textons

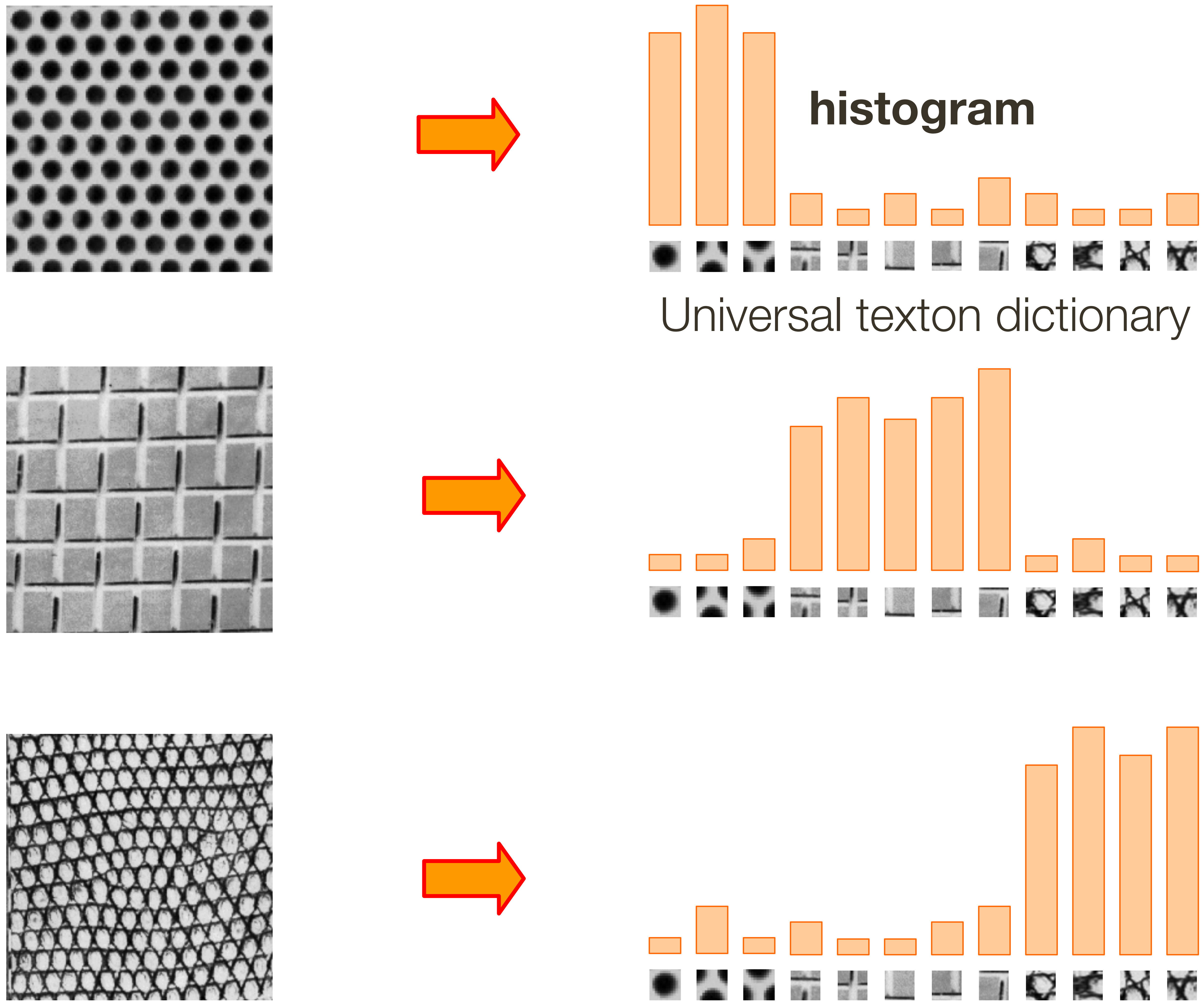
Texture representation and recognition

- Texture is characterized by the repetition of basic elements or **textons**
- For stochastic textures, it is the **identity of the textons**, not their spatial arrangement, that matters

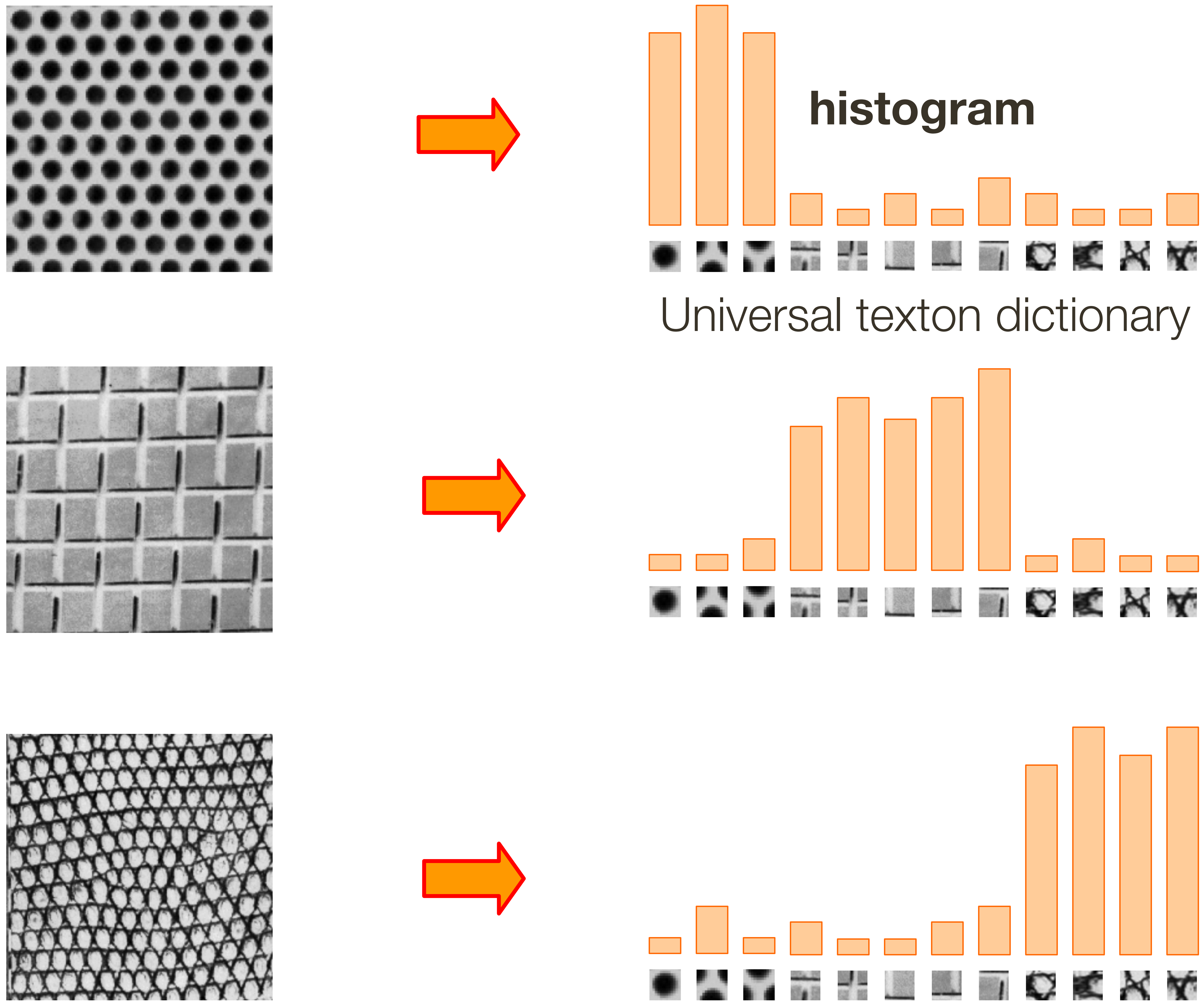


Julesz, 1981; Cula & Dana, 2001; Leung & Malik 2001; Mori, Belongie & Malik, 2001; Schmid 2001; Varma & Zisserman, 2002, 2003; Lazebnik, Schmid & Ponce, 2003

Texture representation and recognition



Texture representation and recognition



Julesz, 1981; Cula & Dana, 2001; Leung & Malik 2001; Mori, Belongie & Malik, 2001; Schmid 2001; Varma & Zisserman, 2002, 2003; Lazebnik, Schmid & Ponce, 2003

Summary

Texture representation is hard

— difficult to define, to analyze

— texture synthesis appears more tractable

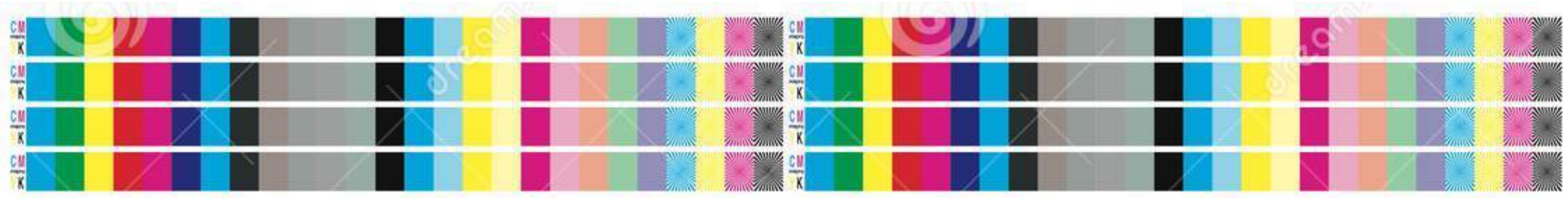
Objective of texture **synthesis** is to generate new examples of a texture

— Efros and Leung: Draw samples directly from the texture to generate one pixel at a time. A “data-driven" approach.

Approaches to texture embed assumptions related to human perception

THE UNIVERSITY OF BRITISH COLUMBIA

CPSC 425: Computer Vision



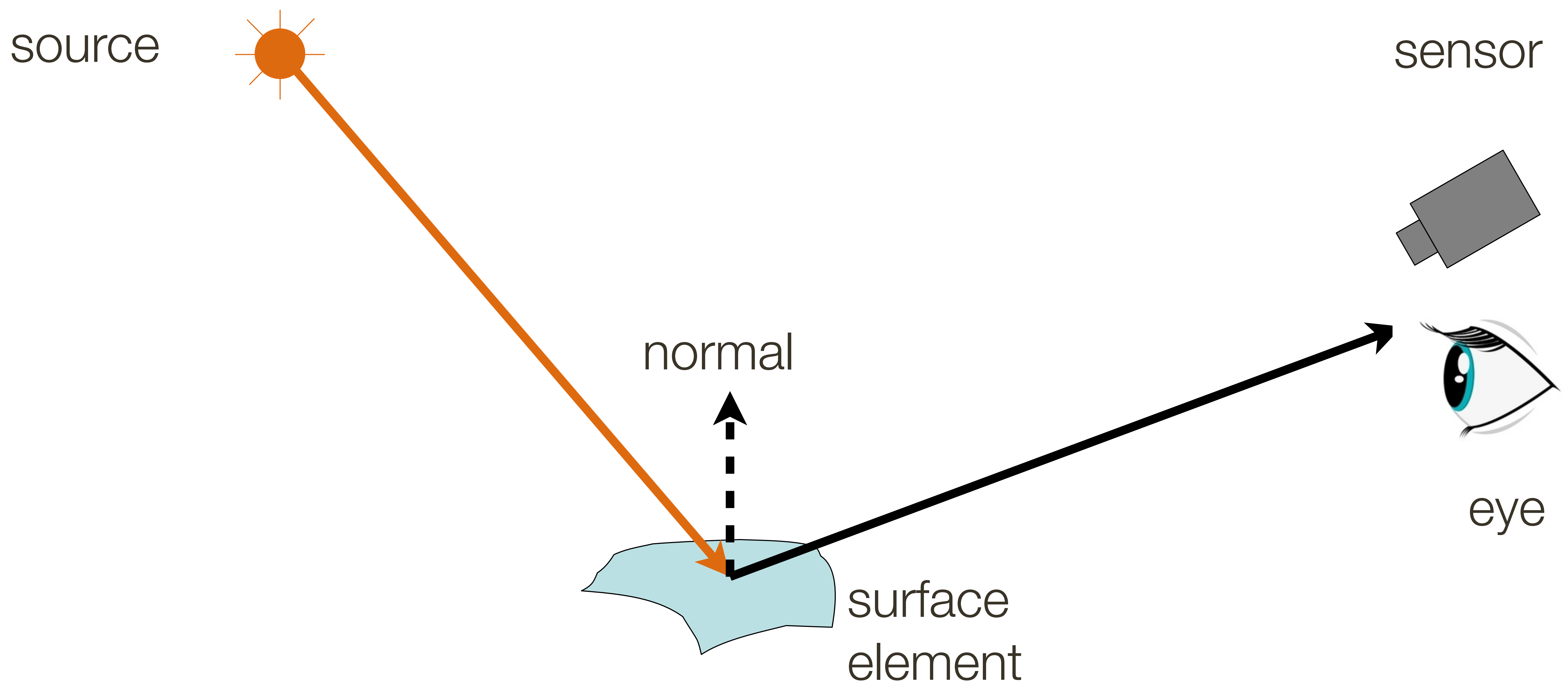
Lecture 11: Color

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

Overview: Image Formation, Cameras and Lenses

The **image formation process** that produces a particular image depends on

- — **Lightening** condition
- — Scene **geometry**
- — **Surface** properties
- — Camera **optics**



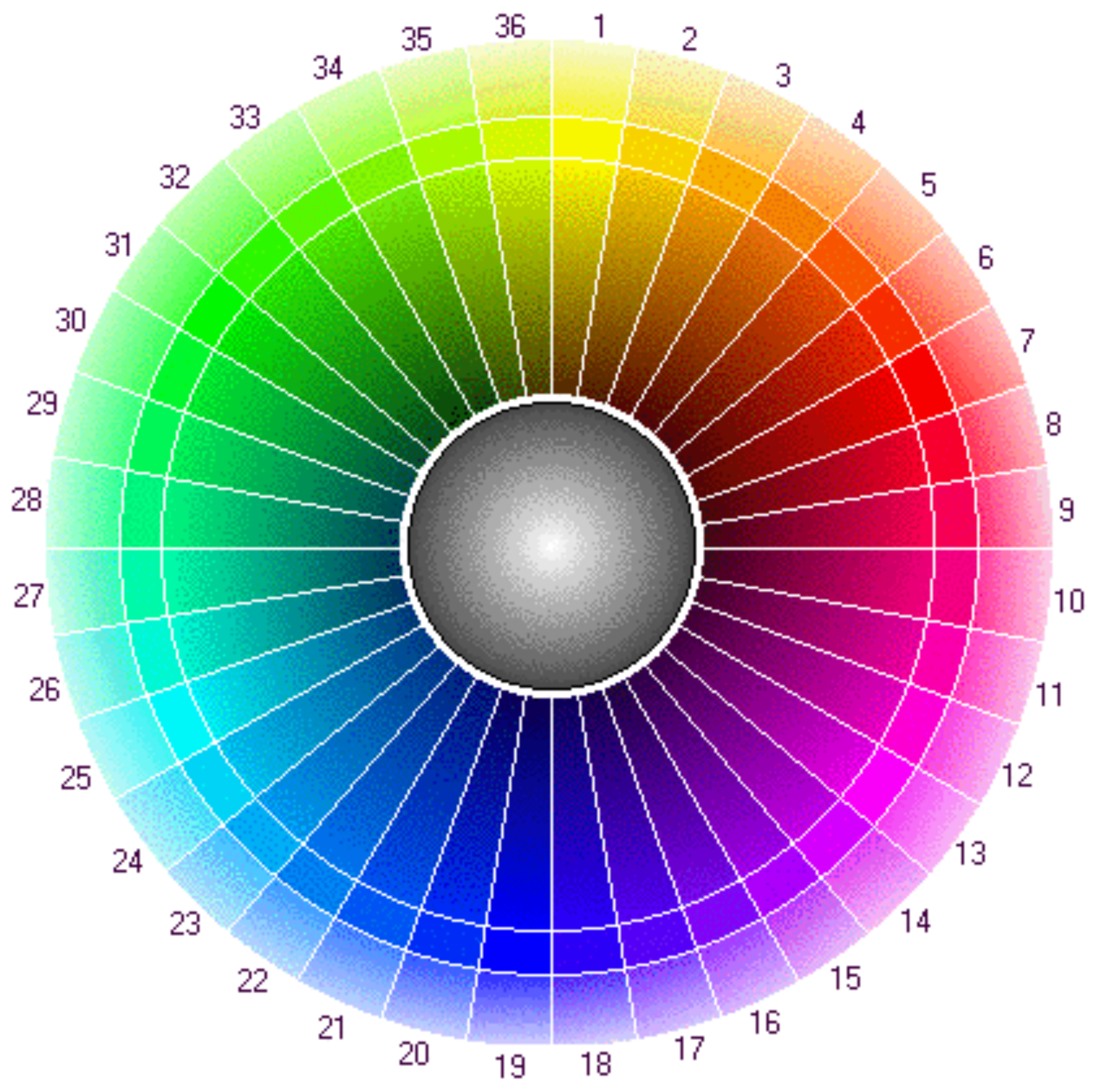
Sensor (or eye) **captures amount of light** reflected from the object

Colour

— Light is produced in different amounts at different wavelengths by each light source

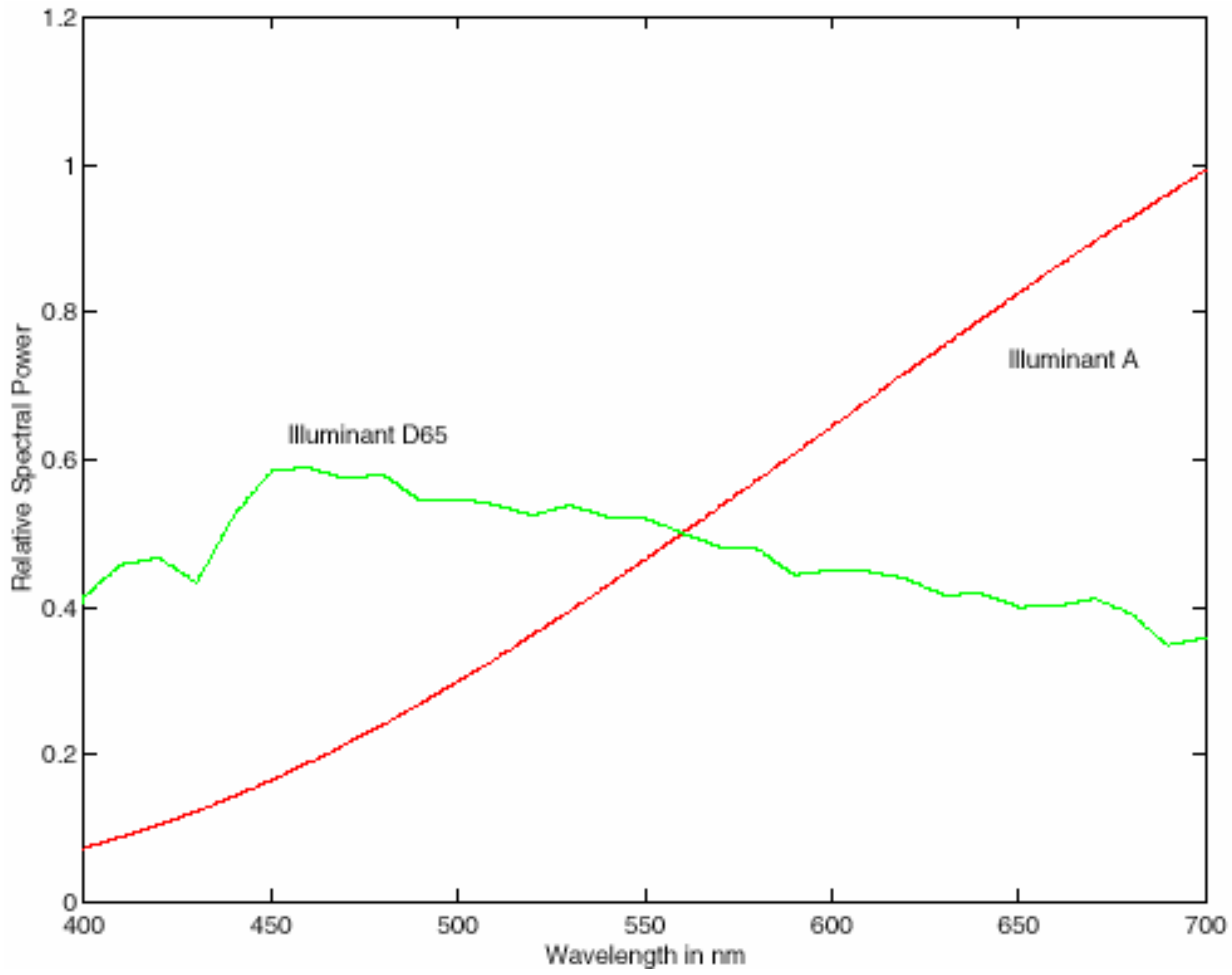
— Light is differentially reflected at each wavelength, which gives objects their natural colour (**surface albedo**)

— The sensation of colour is determined by the human visual system, based on the product of light and reflectance



Relative **Spectral Power** of Two Illuminants

Relative spectral power plotted against wavelength in nm



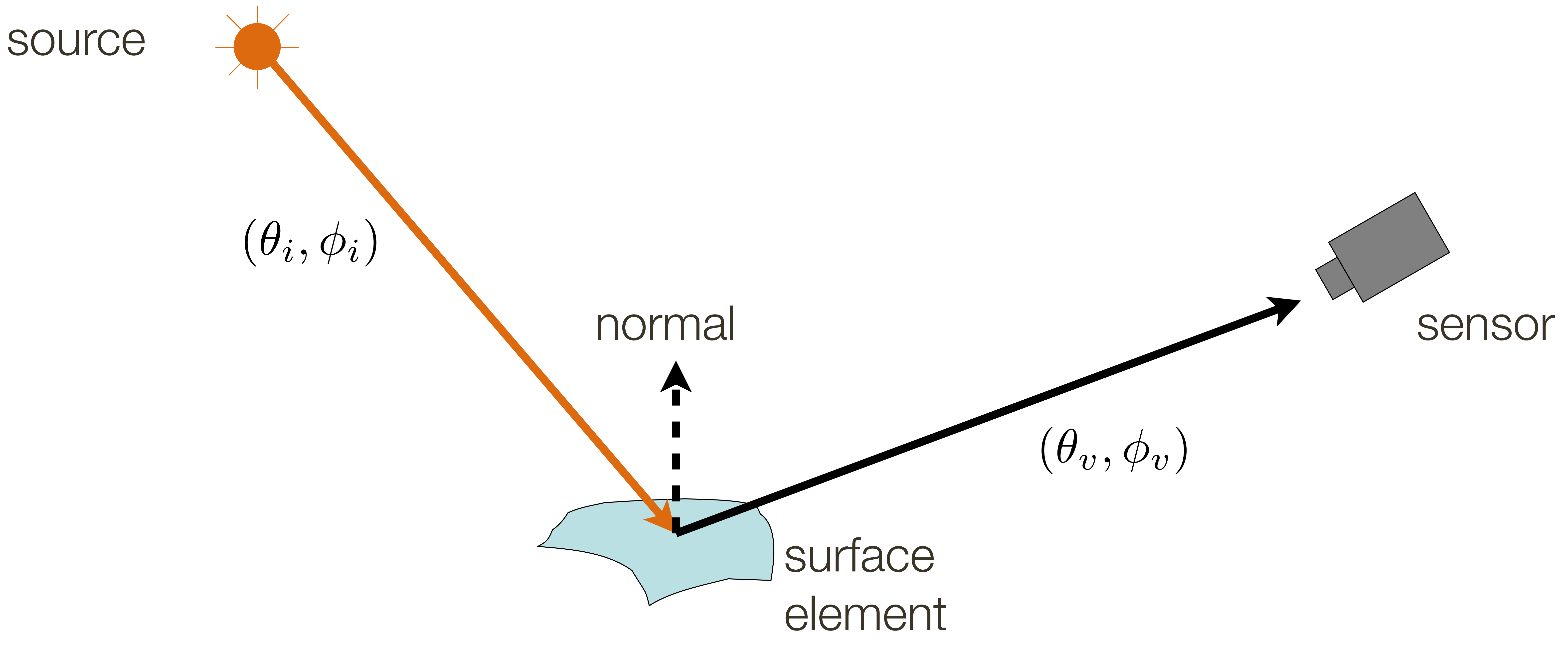
Incandescent Light Bulb

Sunlight

Forsyth & Ponce (2nd ed.) Figure 3.4

(small) **Graphics** Review

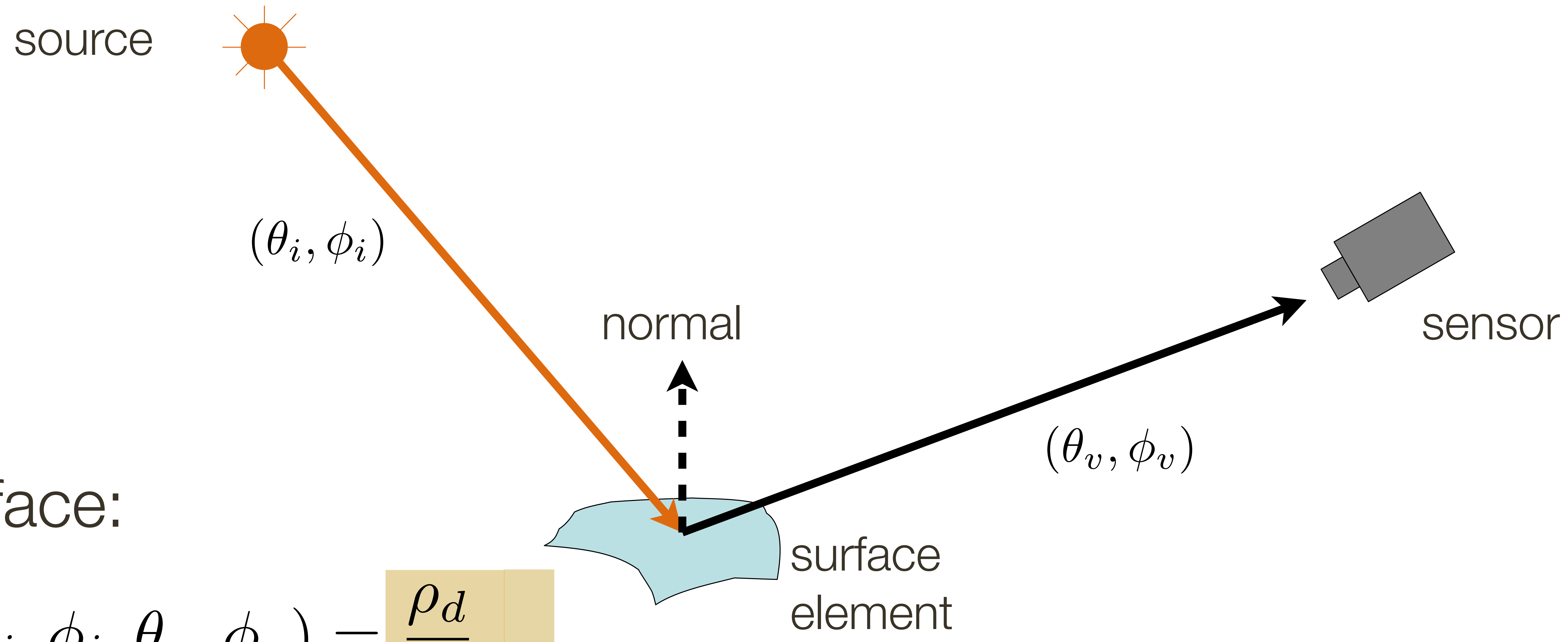
Surface reflection depends on both the **viewing** (θ_v, ϕ_v) and **illumination** (θ_i, ϕ_i) direction, with Bidirectional Reflection Distribution Function: $\mathbf{BRDF}(\theta_i, \phi_i, \theta_v, \phi_v)$



Slide adopted from: Ioannis (Yannis) Gkioulekas (CMU)

(small) **Graphics** Review

Surface reflection depends on both the **viewing** (θ_v, ϕ_v) and **illumination** (θ_i, ϕ_i) direction, with Bidirectional Reflection Distribution Function: $\mathbf{BRDF}(\theta_i, \phi_i, \theta_v, \phi_v)$



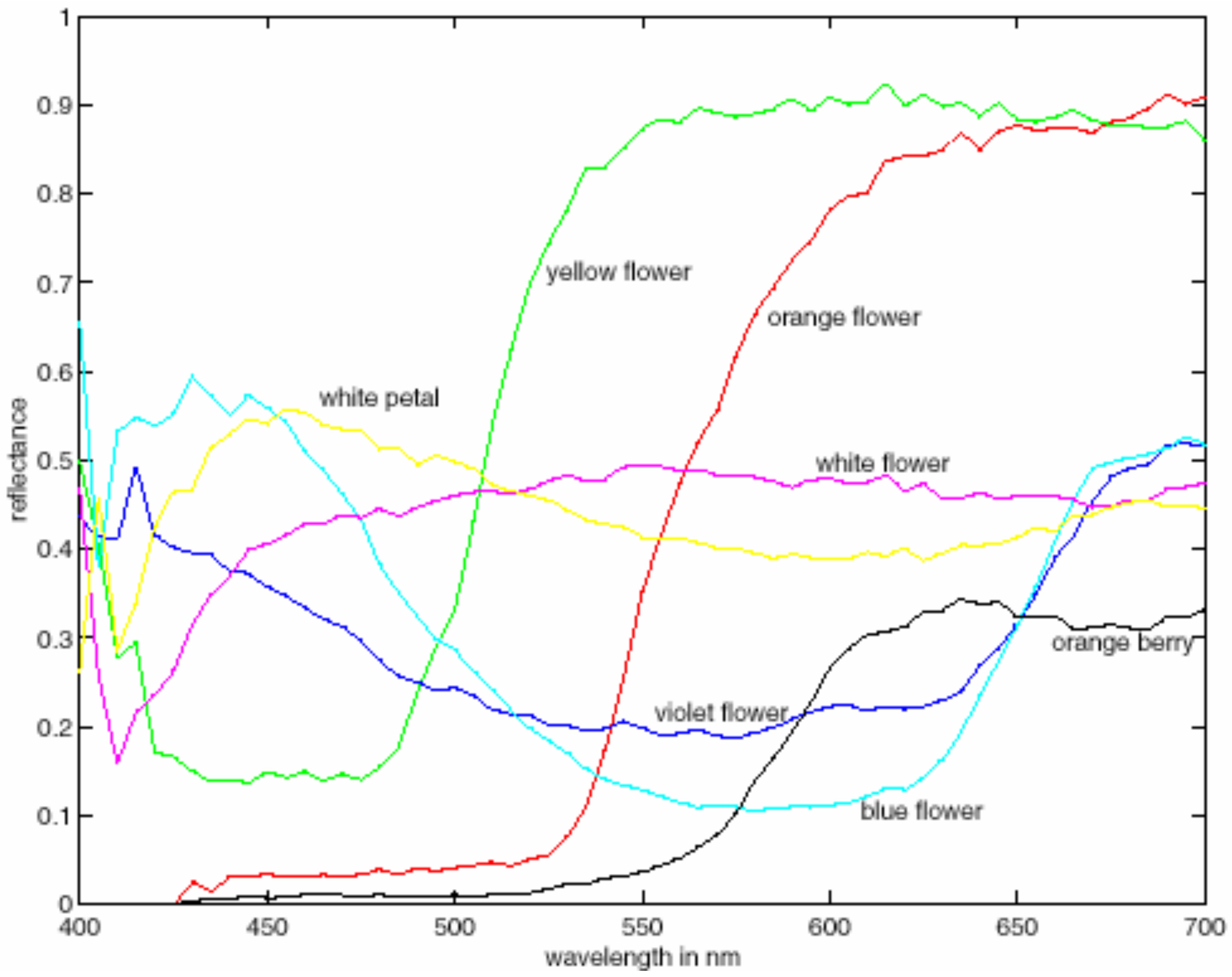
Lambertian surface:

$$\mathbf{BRDF}(\theta_i, \phi_i, \theta_v, \phi_v) = \frac{\rho_d}{\pi}$$

constant, called **albedo**

Slide adopted from: Ioannis (Yannis) Gkioulekas (CMU)

Spectral **Albedo** of Natural Surfaces



Forsyth & Ponce (2nd ed.) Figure 3.6

Colour Appearance

Reflected light **at each wavelength** is the product of illumination and surface reflectance at that wavelength

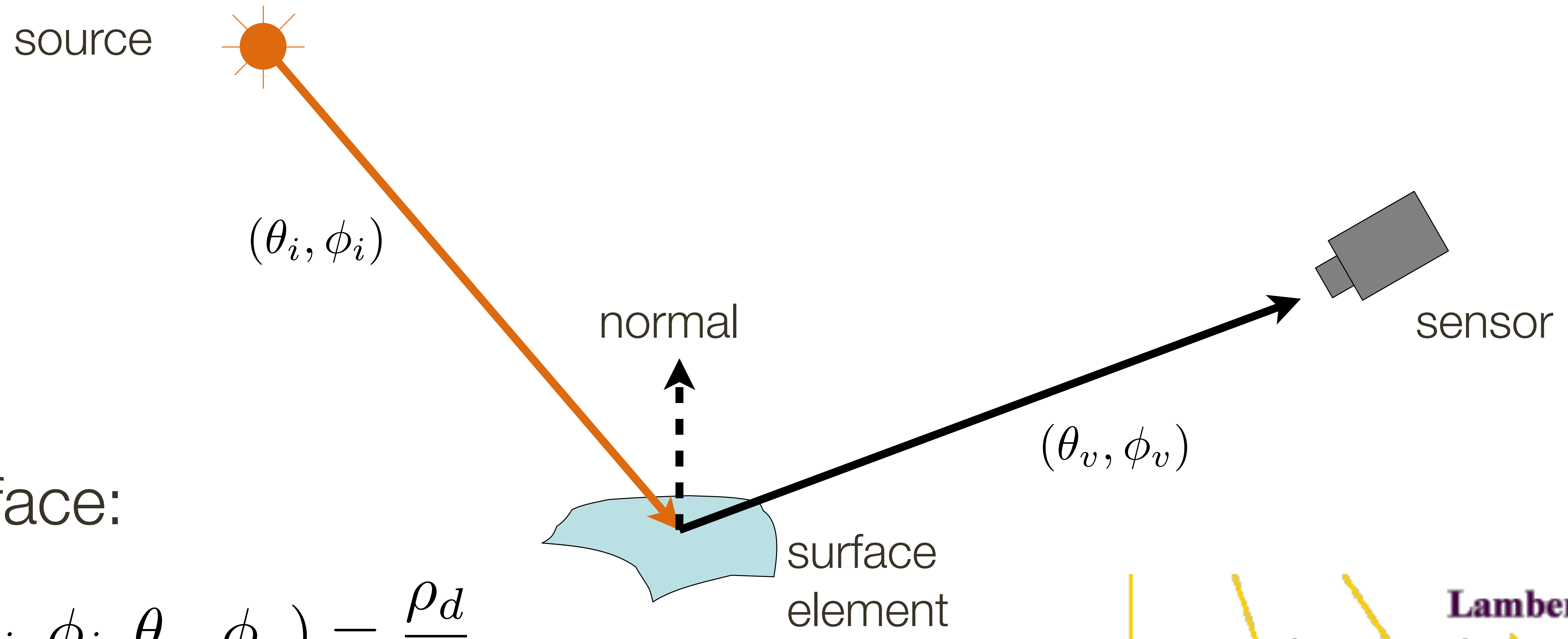
Surface reflectance often is modeled as having two components:

— **Lambertian** reflectance: equal in all directions (diffuse)

— **Specular** reflectance: mirror reflectance (shiny spots)

(small) **Graphics** Review

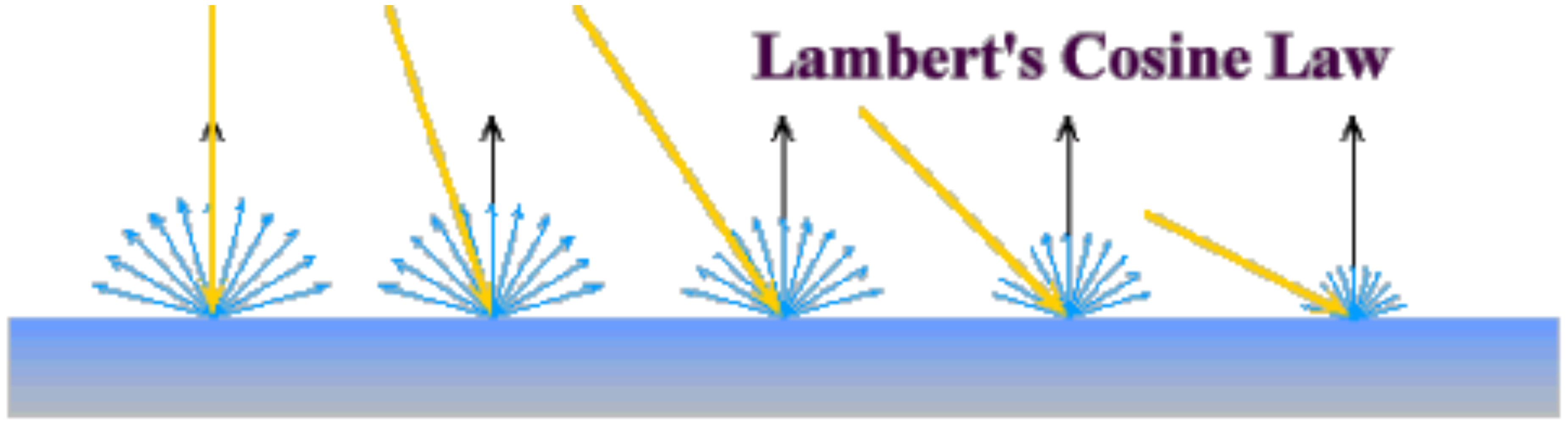
Surface reflection depends on both the **viewing** (θ_v, ϕ_v) and **illumination** (θ_i, ϕ_i) direction, with Bidirectional Reflection Distribution Function: $\mathbf{BRDF}(\theta_i, \phi_i, \theta_v, \phi_v)$



Lambertian surface:

$$\mathbf{BRDF}(\theta_i, \phi_i, \theta_v, \phi_v) = \frac{\rho_d}{\pi}$$

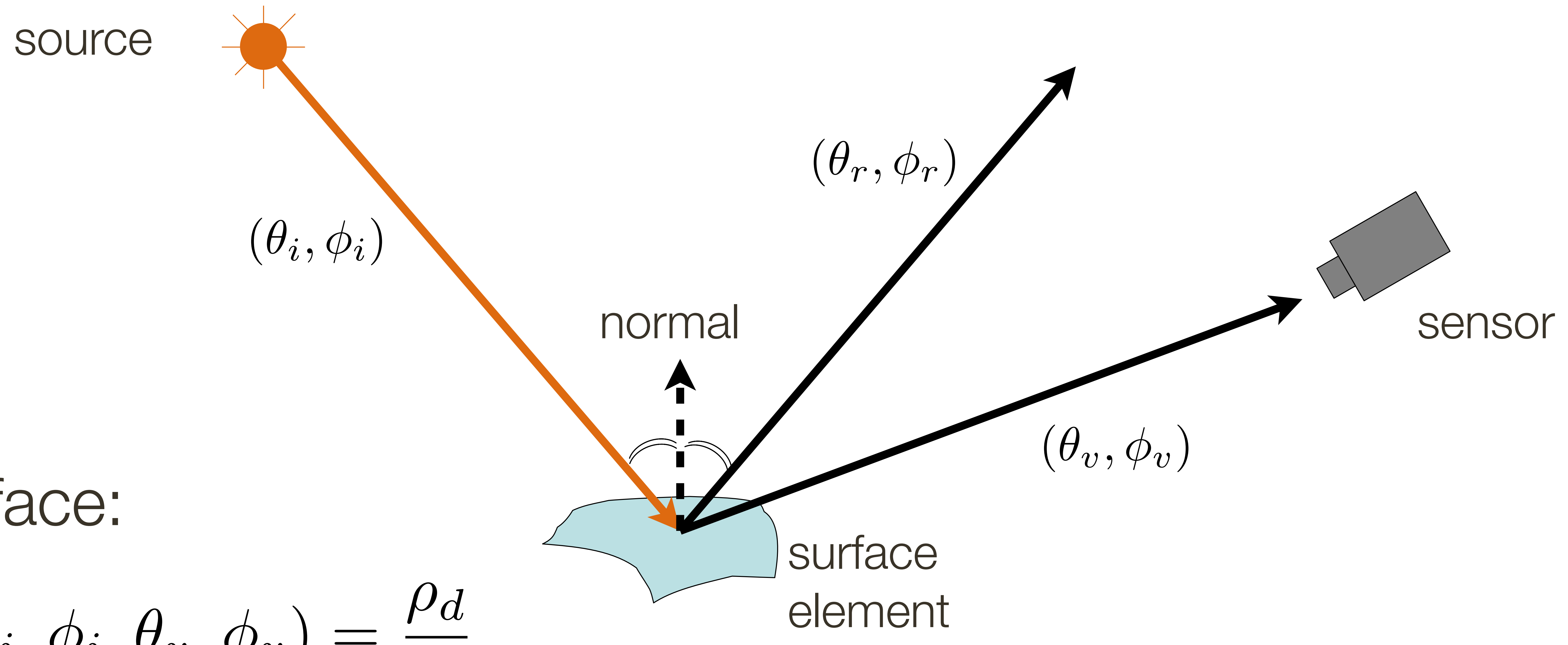
$$L = \frac{\rho_d}{\pi} I(\vec{i} \cdot \vec{n})$$



Slide adopted from: Ioannis (Yannis) Gkioulekas (CMU)

(small) **Graphics** Review

Surface reflection depends on both the **viewing** (θ_v, ϕ_v) and **illumination** (θ_i, ϕ_i) direction, with Bidirectional Reflection Distribution Function: $\mathbf{BRDF}(\theta_i, \phi_i, \theta_v, \phi_v)$



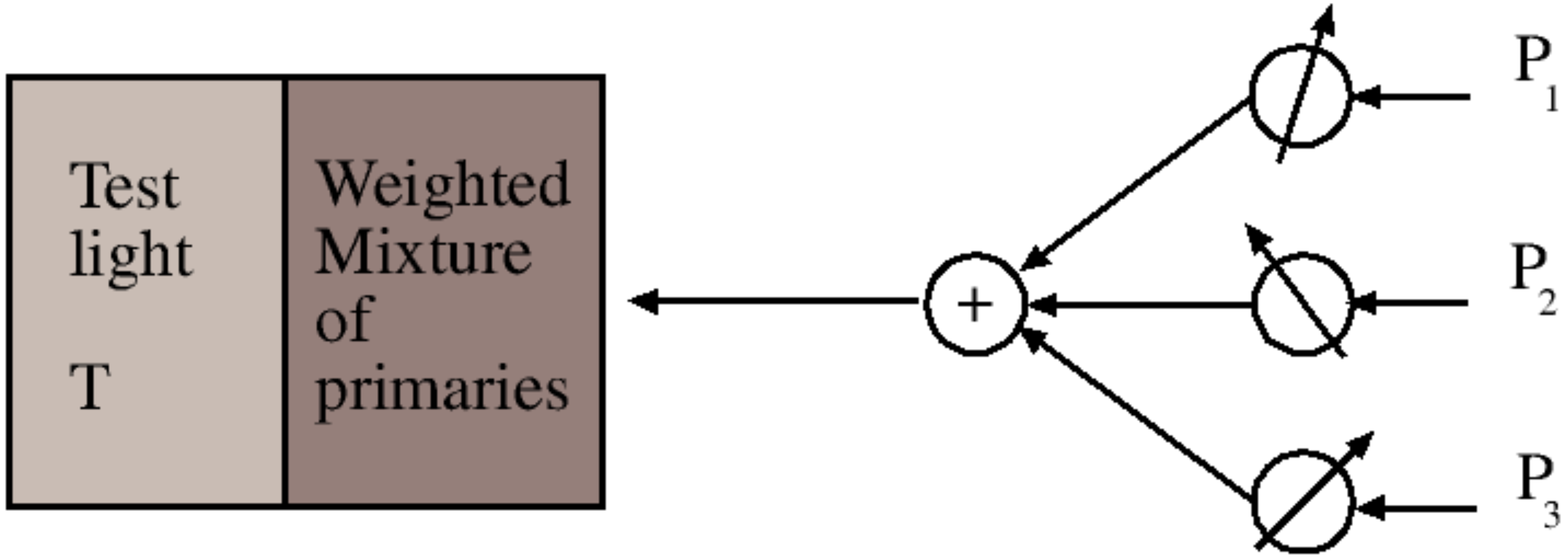
Lambertian surface:

$$\mathbf{BRDF}(\theta_i, \phi_i, \theta_v, \phi_v) = \frac{\rho_d}{\pi}$$

Mirror surface: all incident light reflected in one directions $(\theta_v, \phi_v) = (\theta_r, \phi_r)$

Slide adopted from: Ioannis (Yannis) Gkioulekas (CMU)

Color Matching Experiments



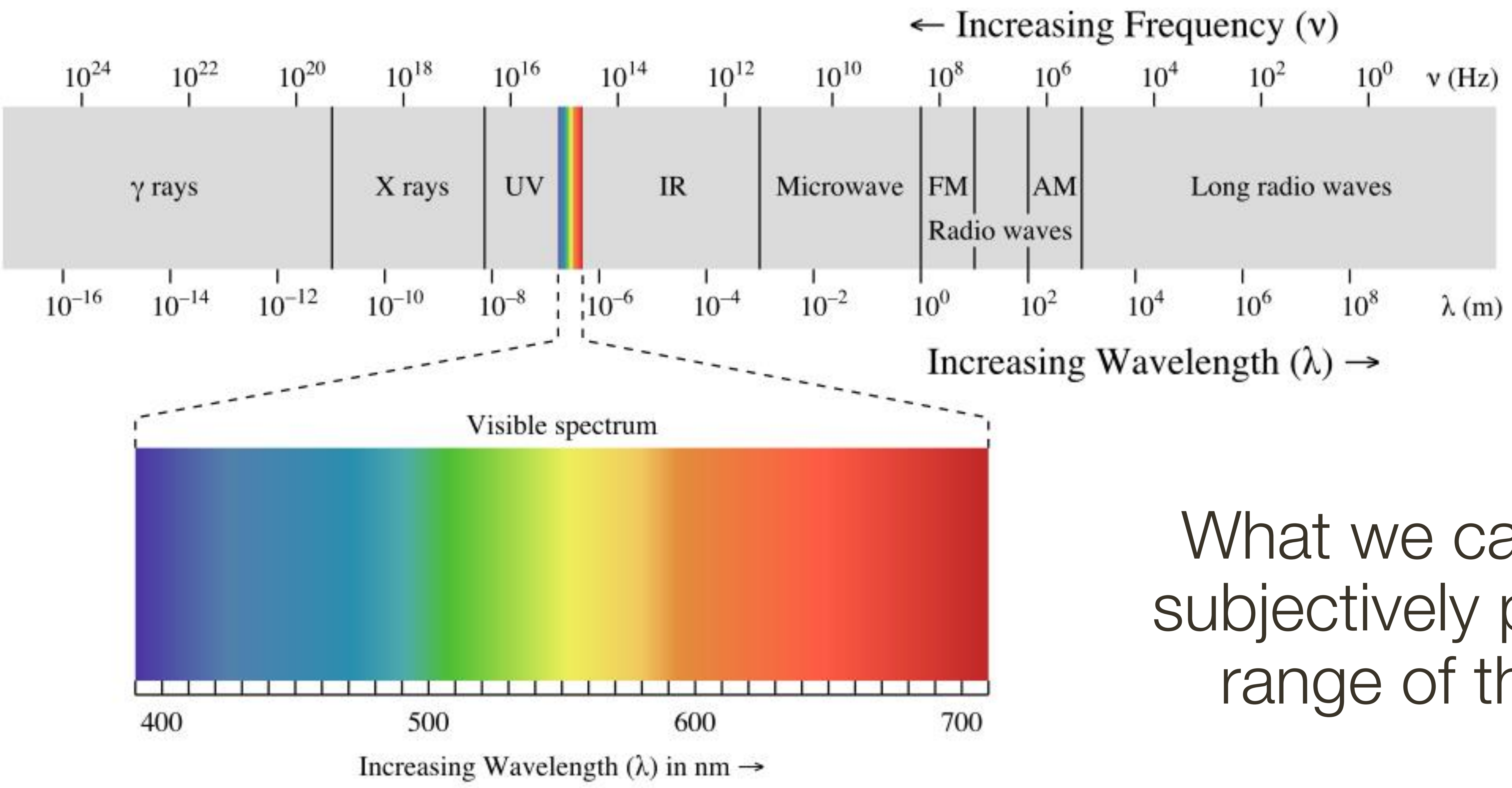
Forsyth & Ponce (2nd ed.) Figure 3.2

Show a split field to subjects. One side shows the light whose colour one wants to match. The other a weighted mixture of three primaries (fixed lights)

$$T = w_1 P_1 + w_2 P_2 + w_3 P_3$$

Recall: 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.

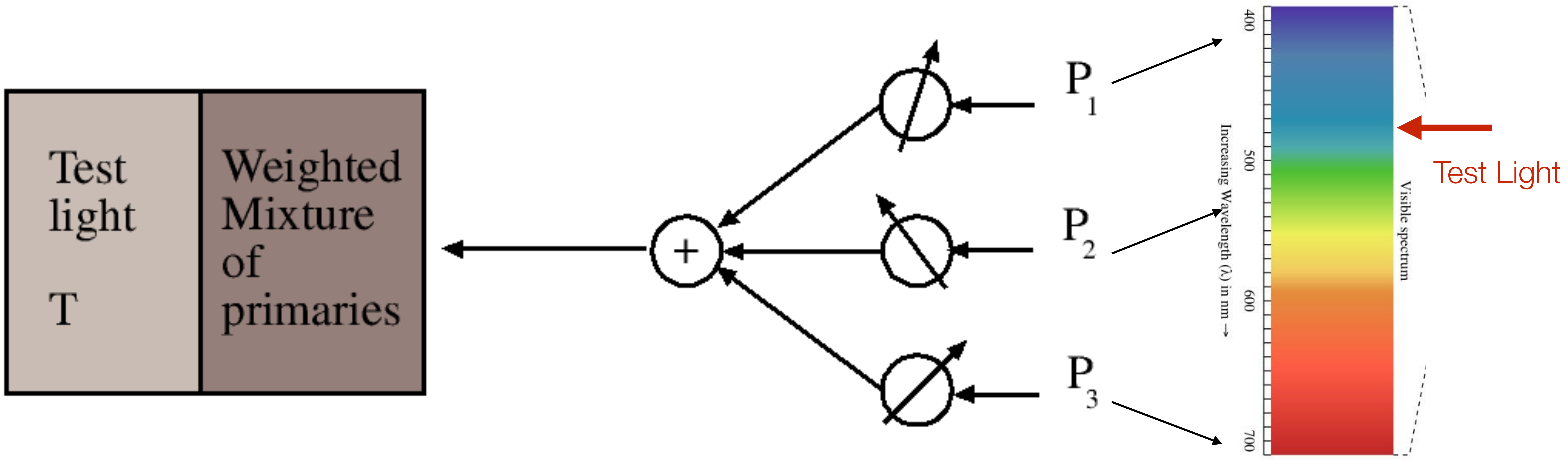


electromagnetic spectrum

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

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

Color Matching Experiments



Forsyth & Ponce (2nd ed.) Figure 3.2

Show a split field to subjects. One side shows the light whose colour one wants to match. The other a weighted mixture of three primaries (fixed lights)

$$T = w_1 P_1 + w_2 P_2 + w_3 P_3$$

Color Matching Experiments

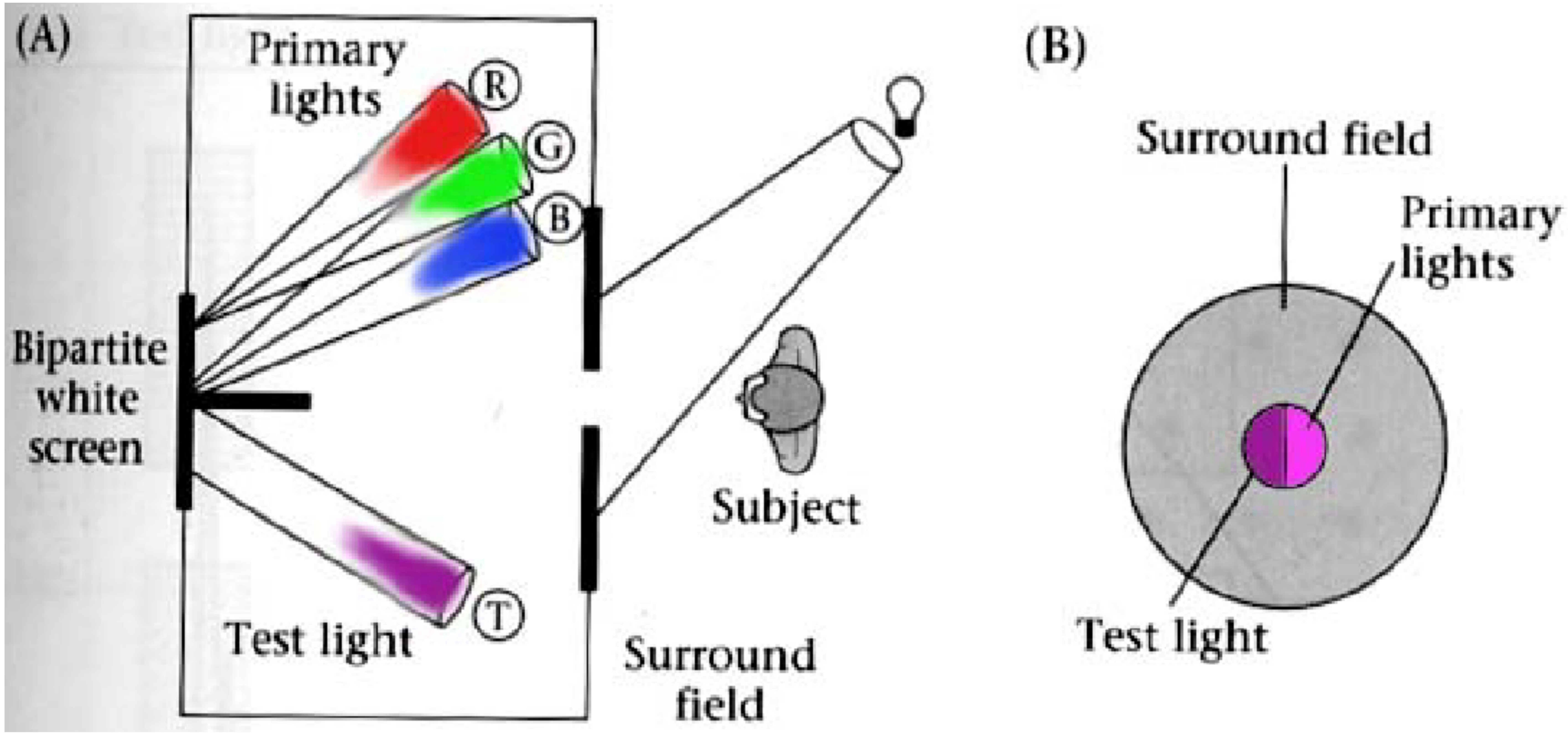
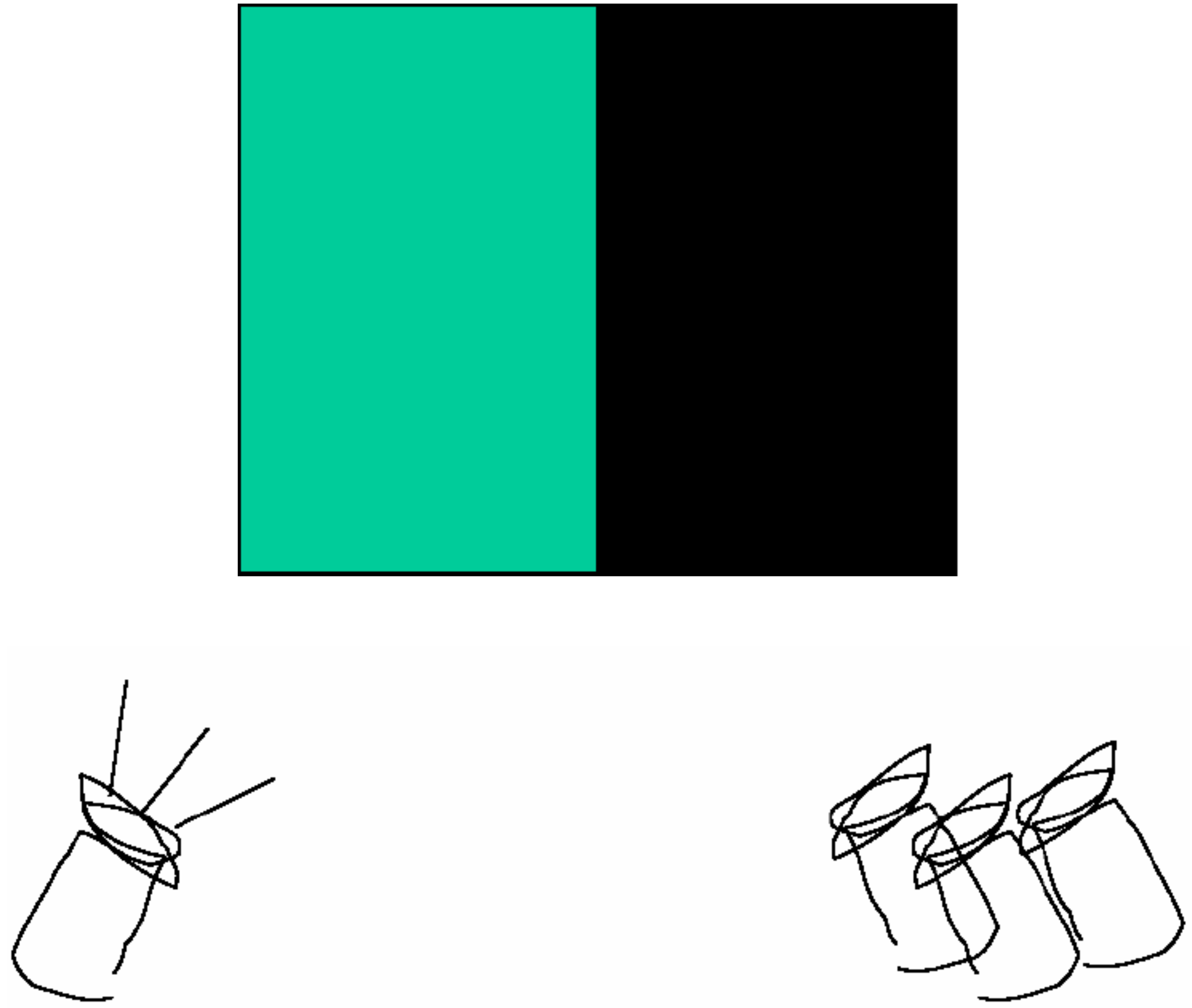


Figure Credit: Brian Wandell, Foundations of Vision, Sinauer Associates, 1995

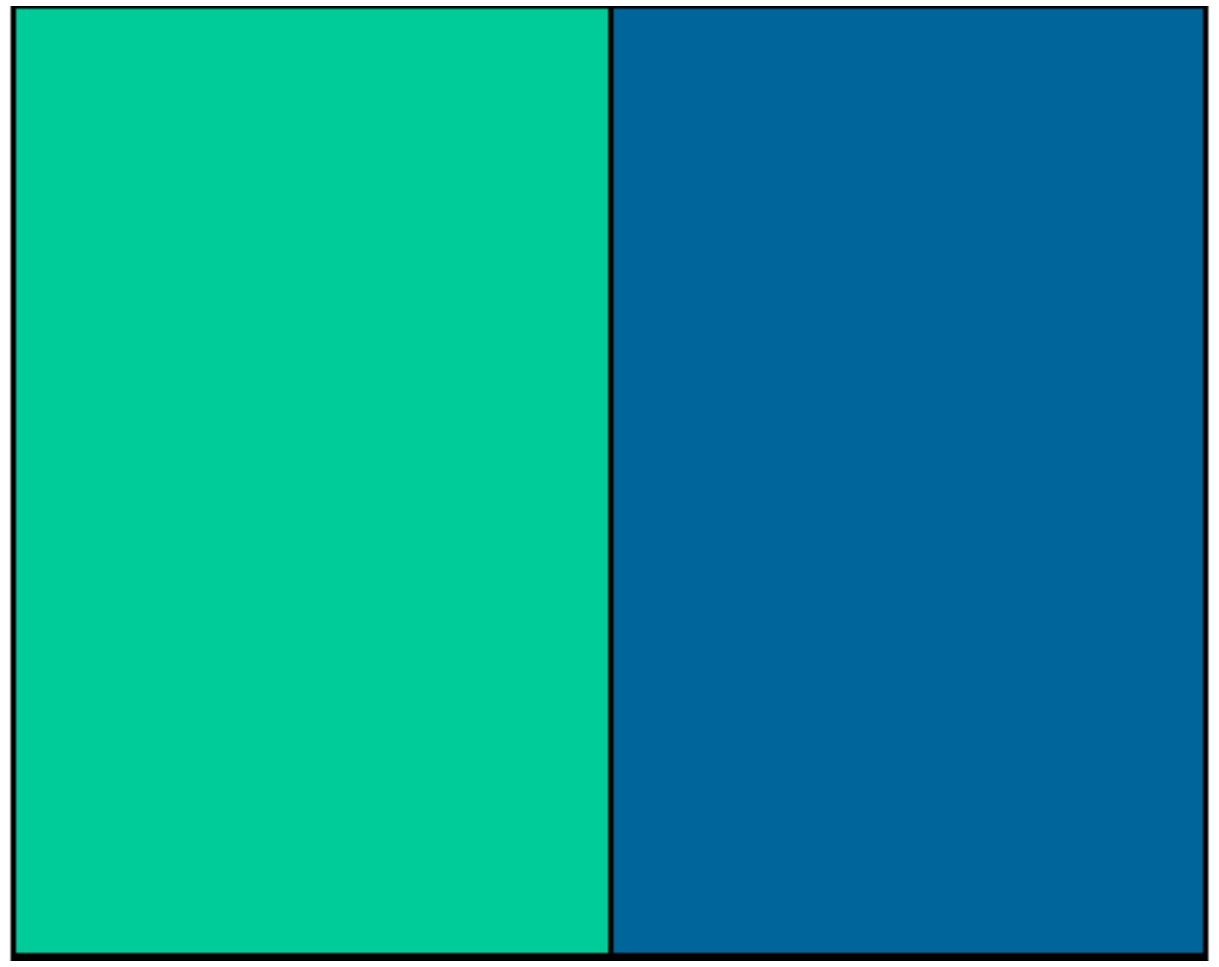
Example 1: Color Matching Experiment

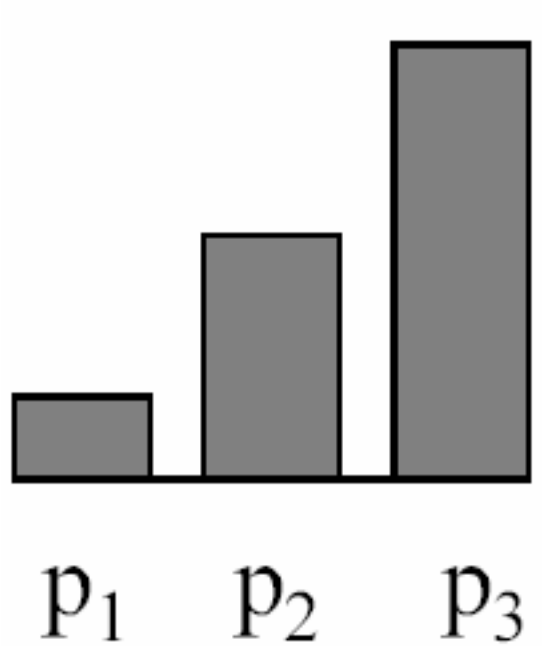


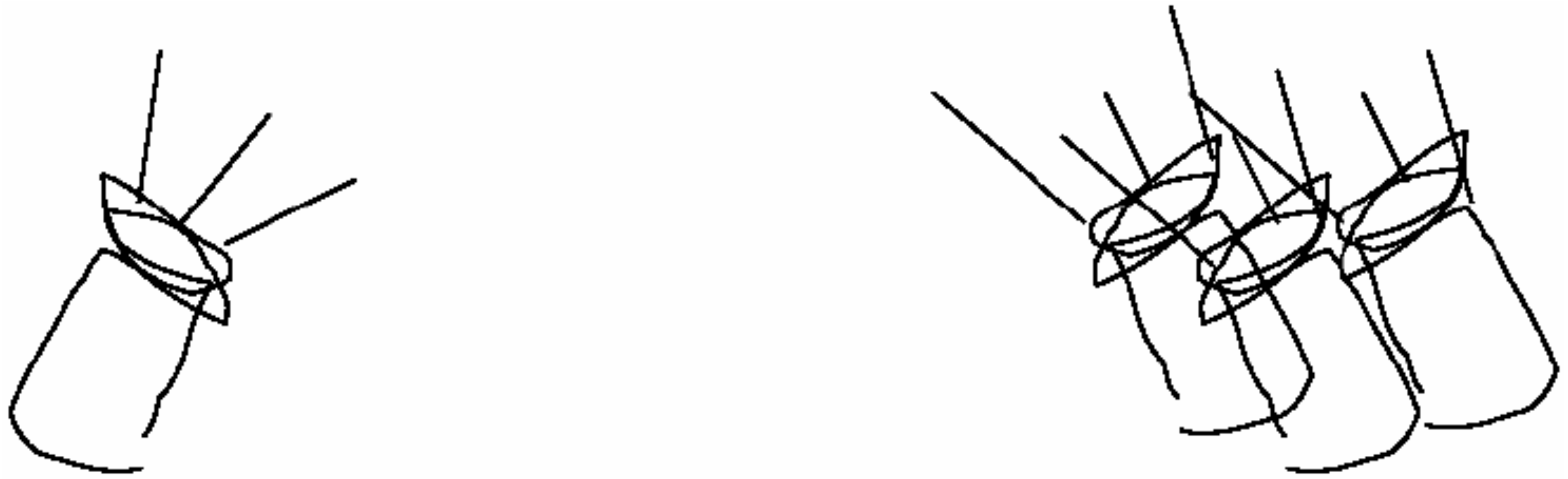
knobs here

Example Credit: Bill Freeman

Example 1: Color Matching Experiment



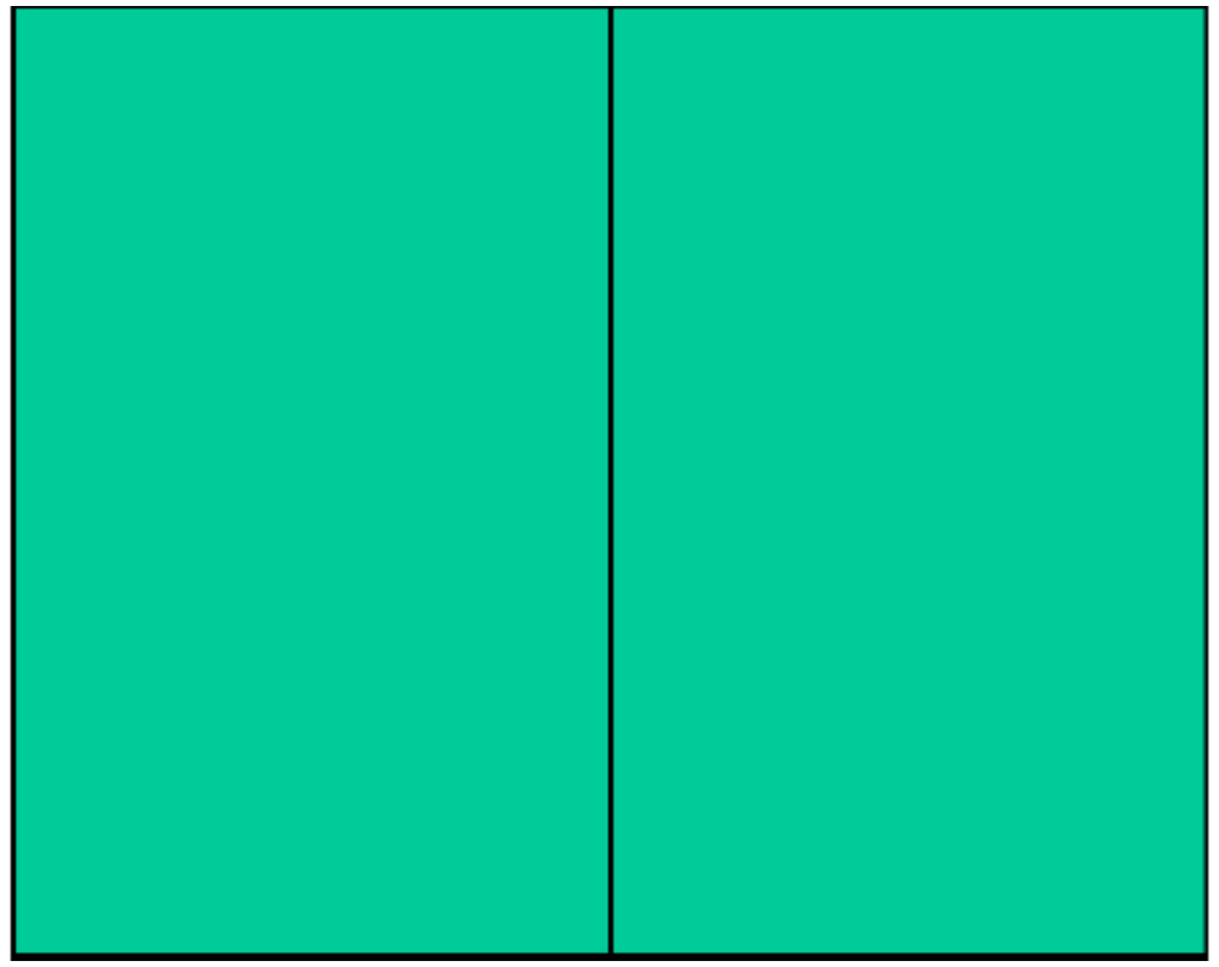


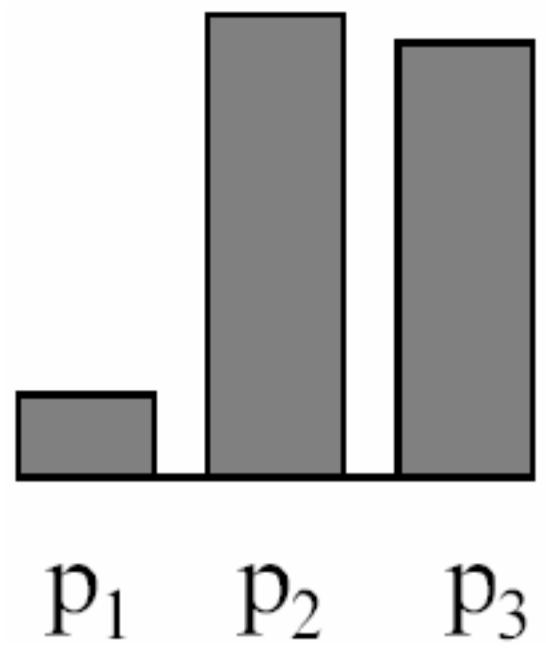


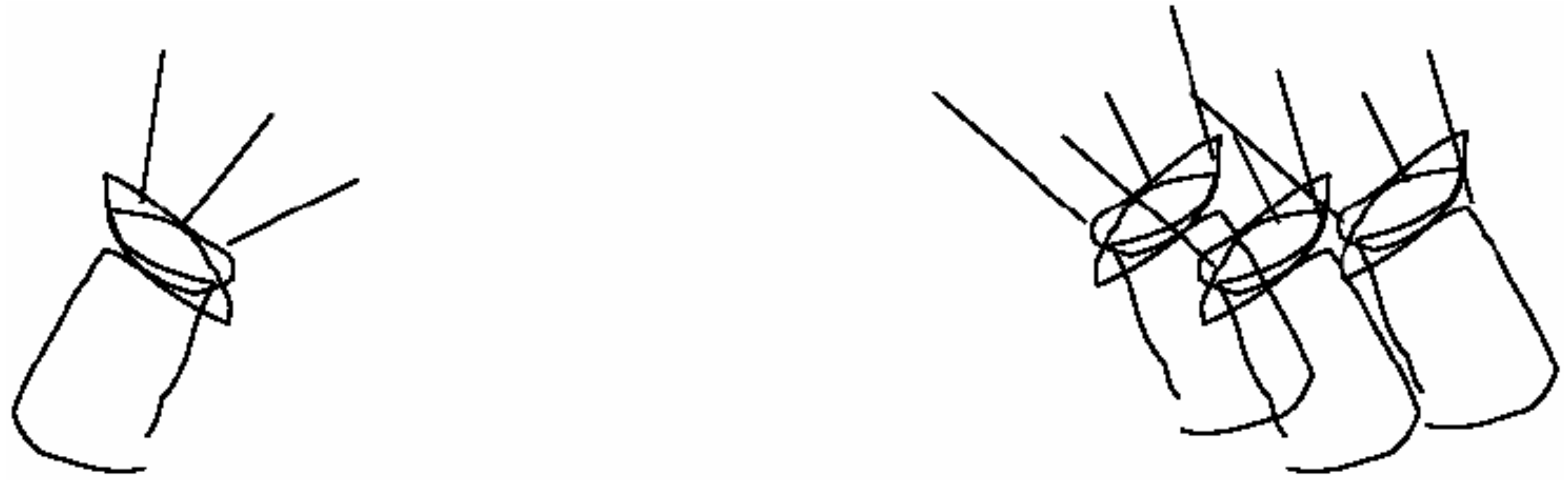
knobs here

Example Credit: Bill Freeman

Example 1: Color Matching Experiment





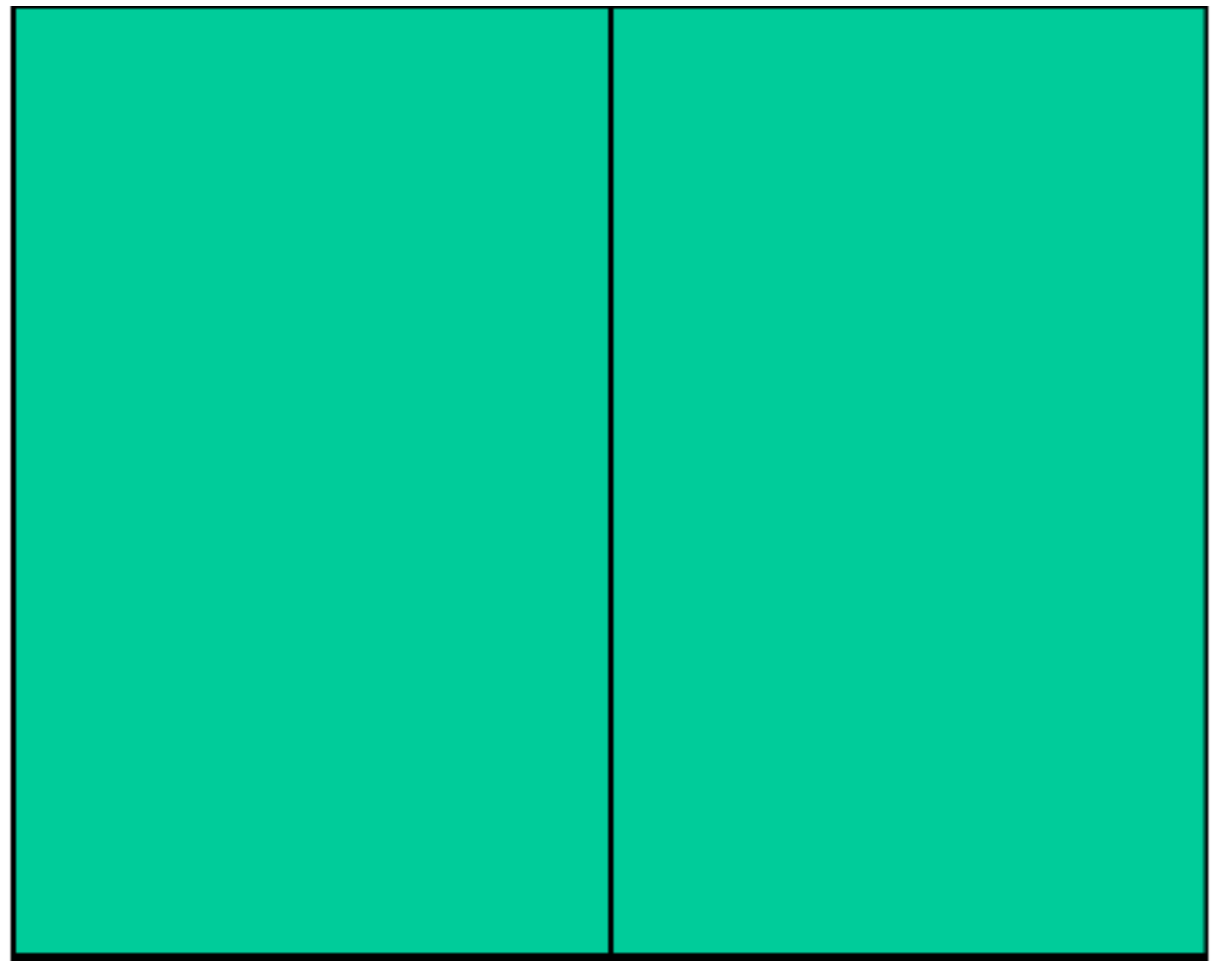


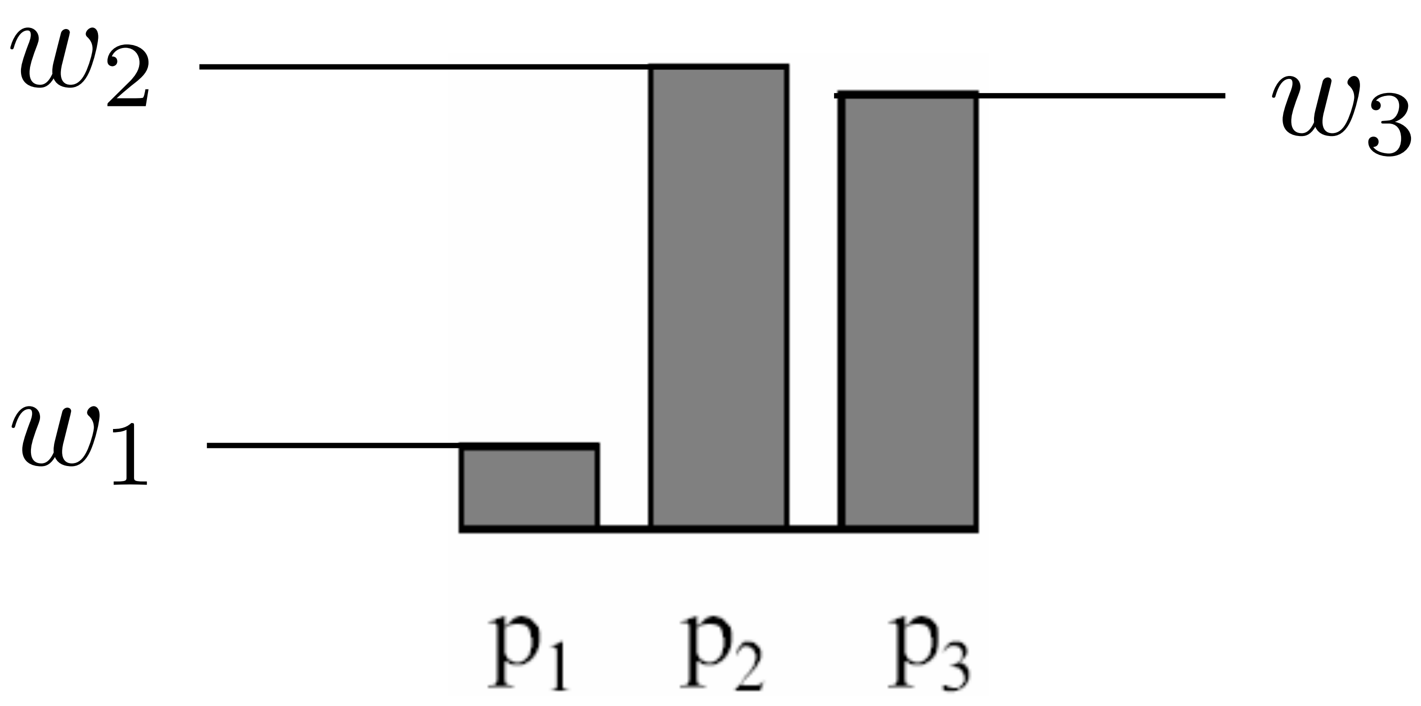
knobs here

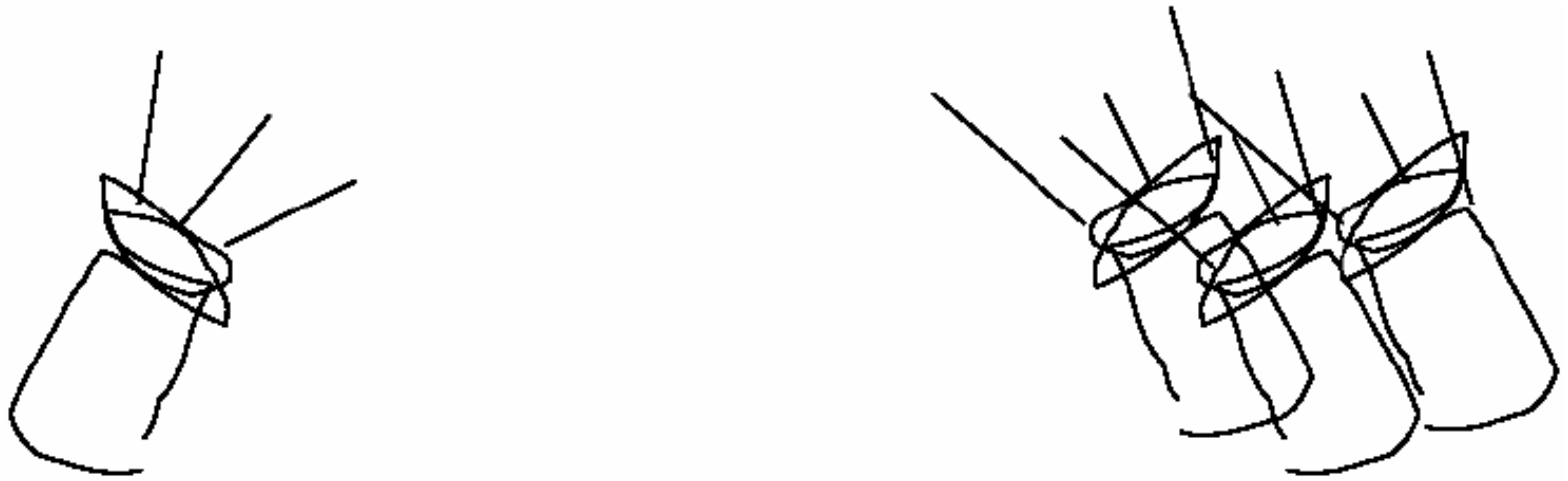
Example Credit: Bill Freeman

Example 1: Color Matching Experiment

$$T = w_1 P_1 + w_2 P_2 + w_3 P_3$$



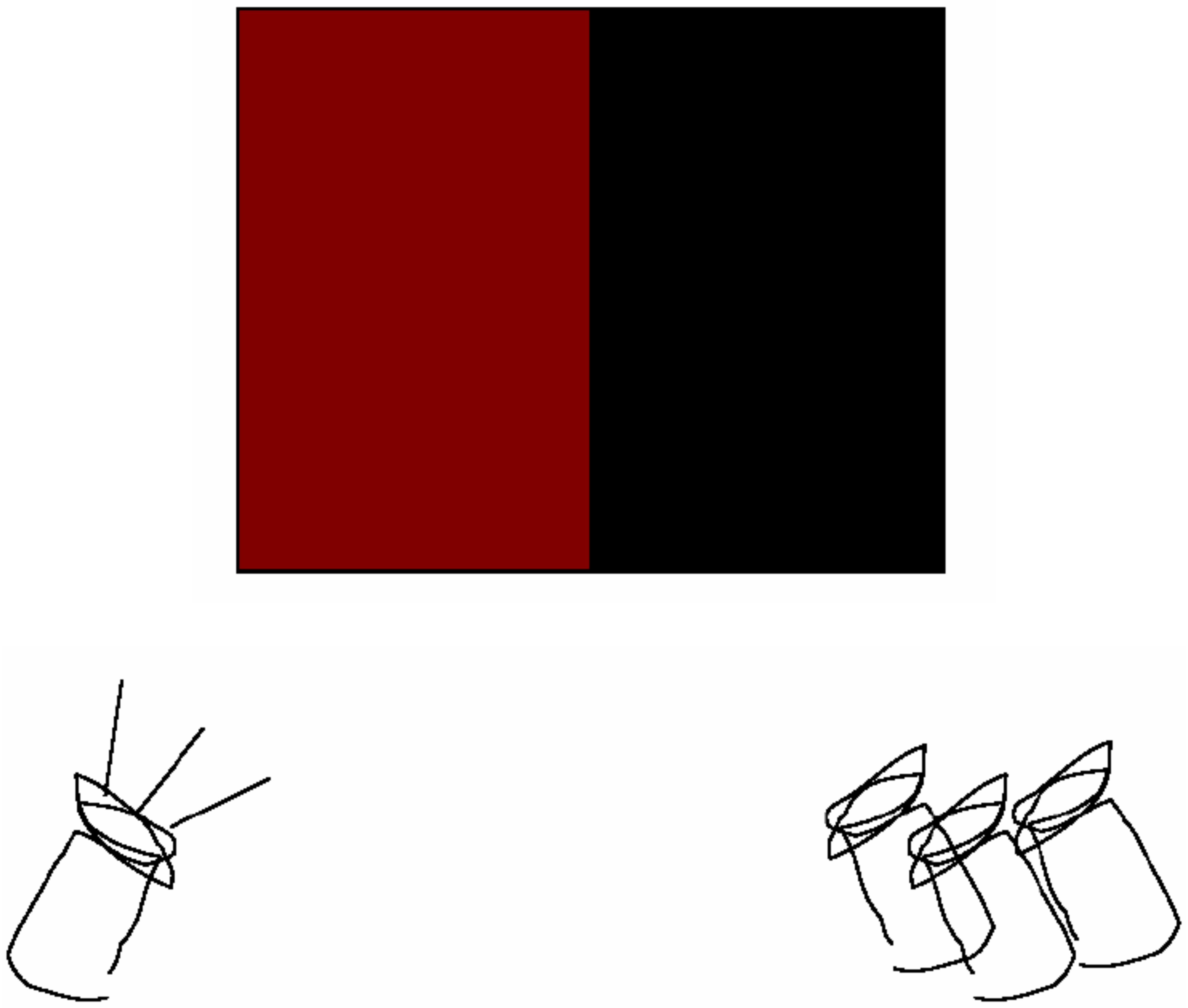




knobs here

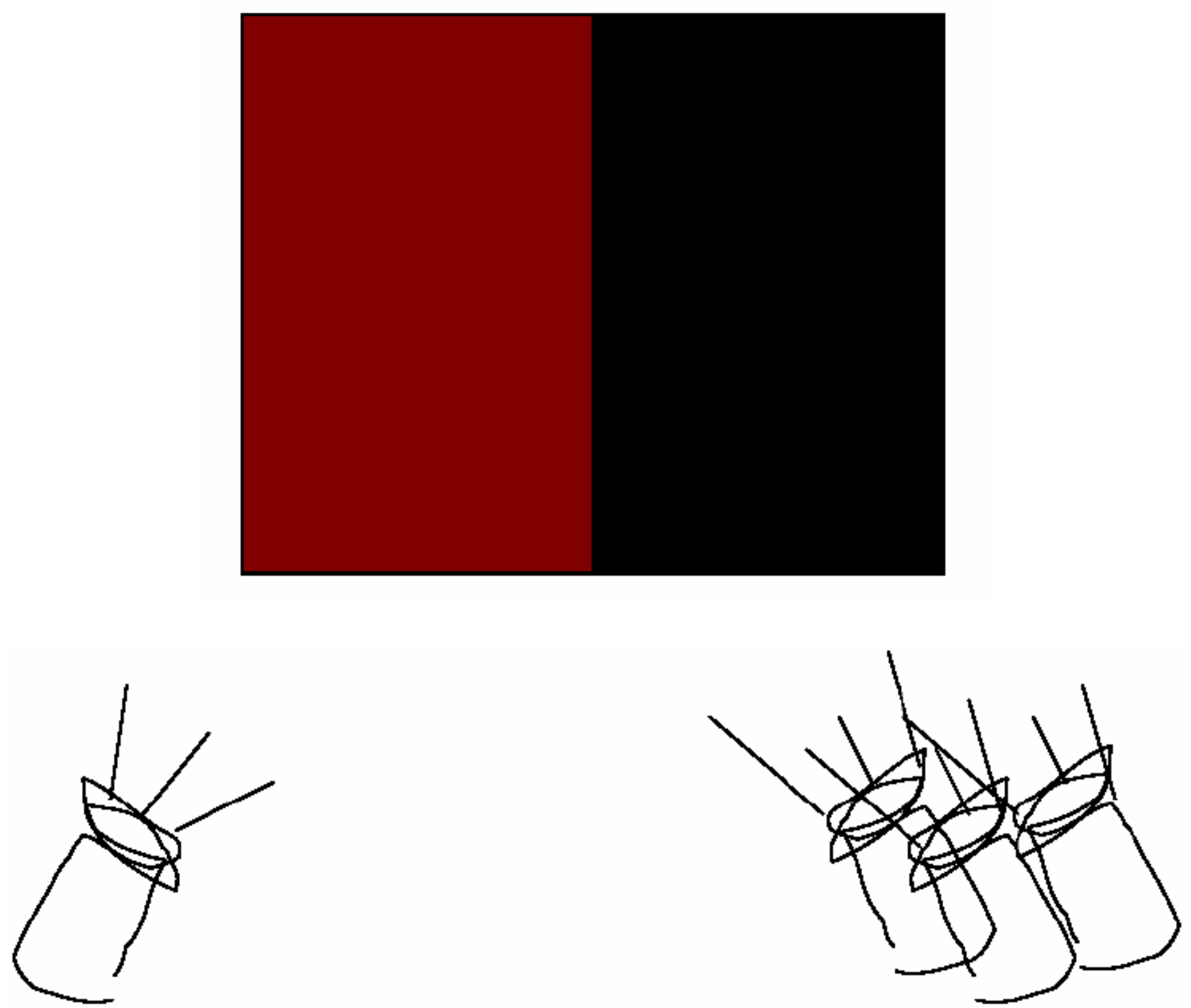
Example Credit: Bill Freeman

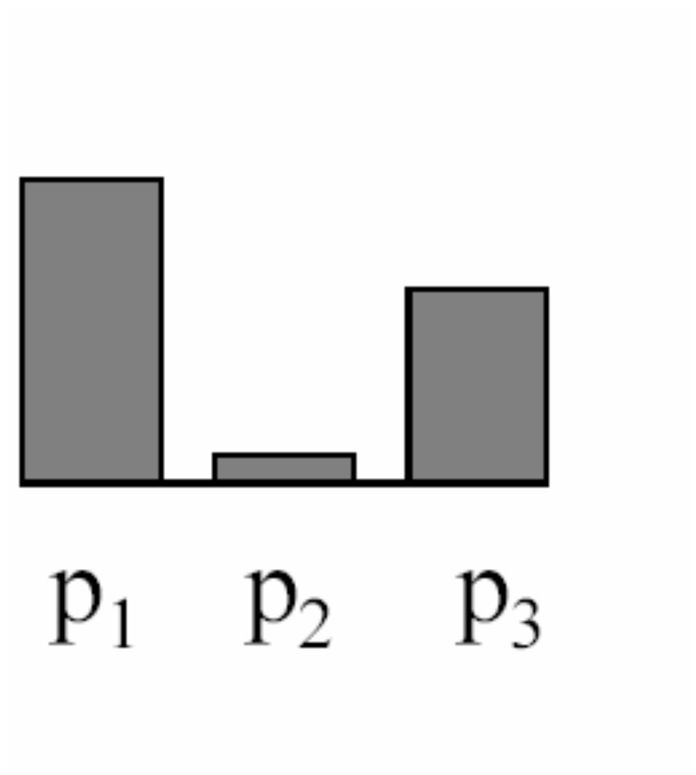
Example 2: Color Matching Experiment



Example Credit: Bill Freeman

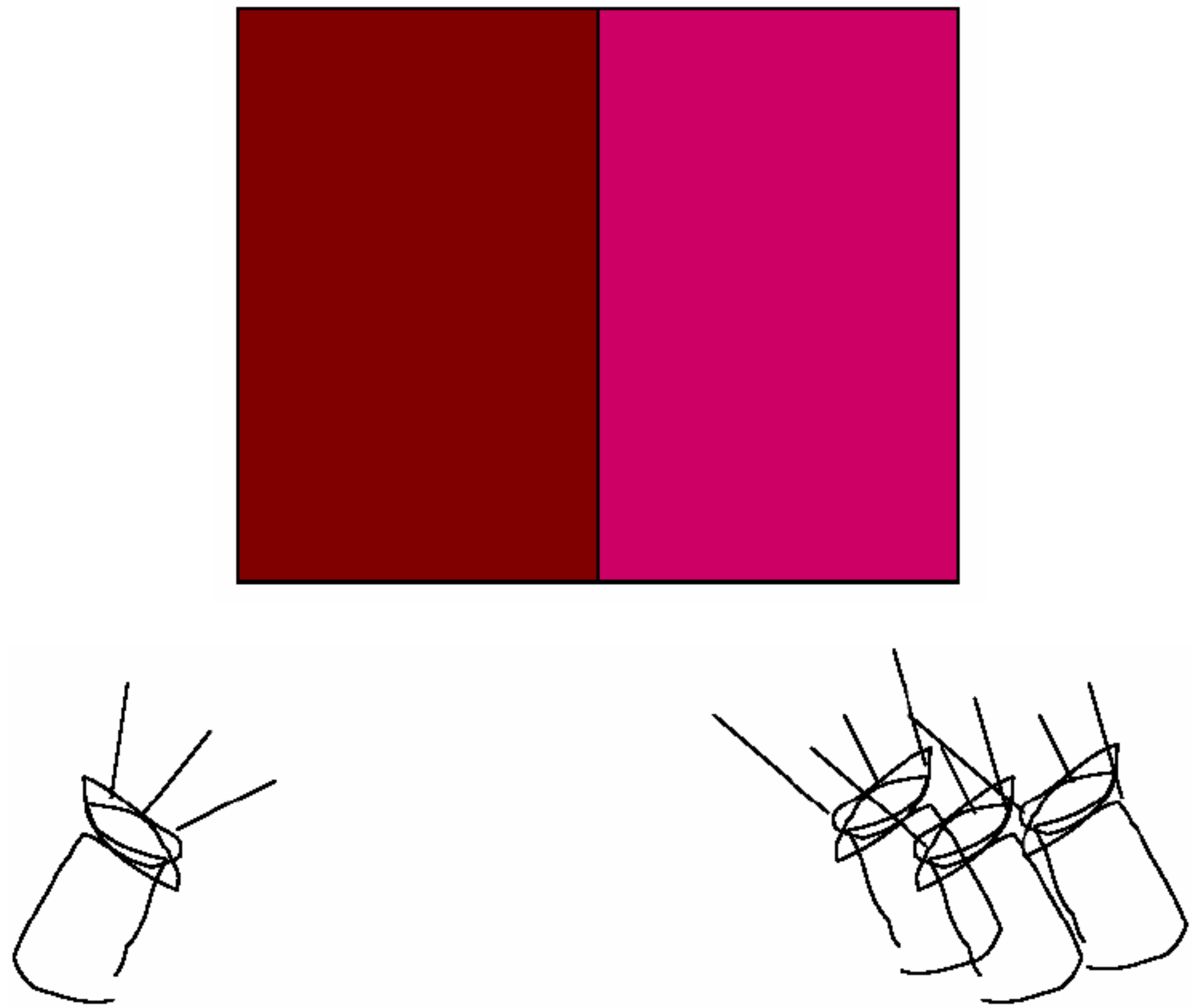
Example 2: Color Matching Experiment

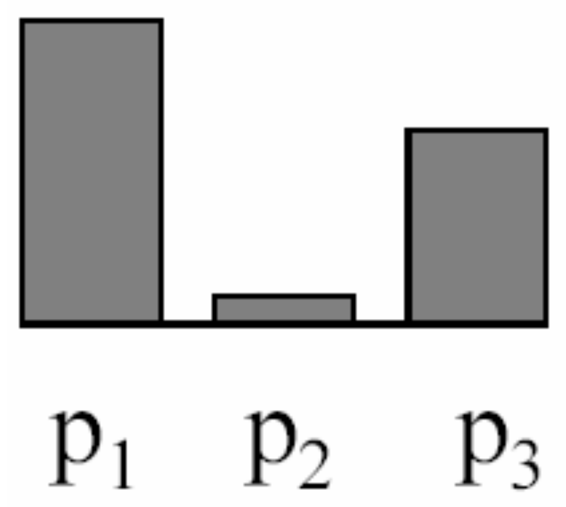




Example Credit: Bill Freeman

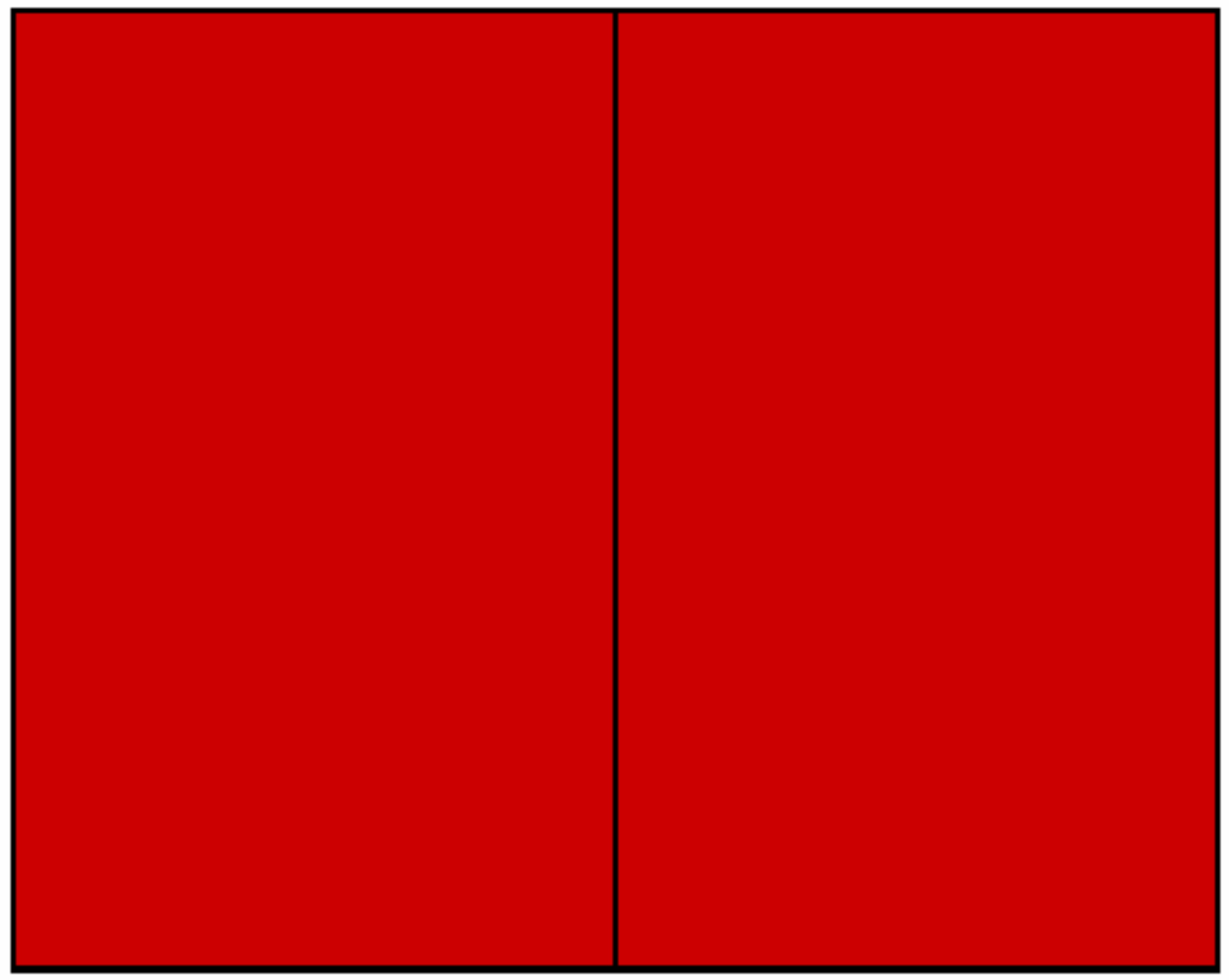
Example 2: Color Matching Experiment

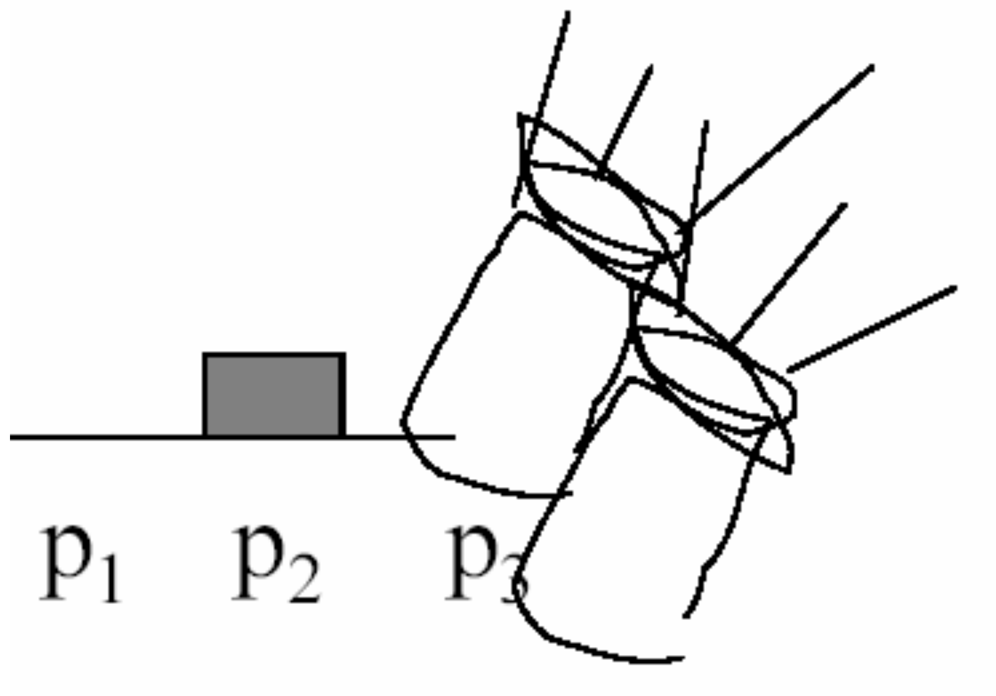


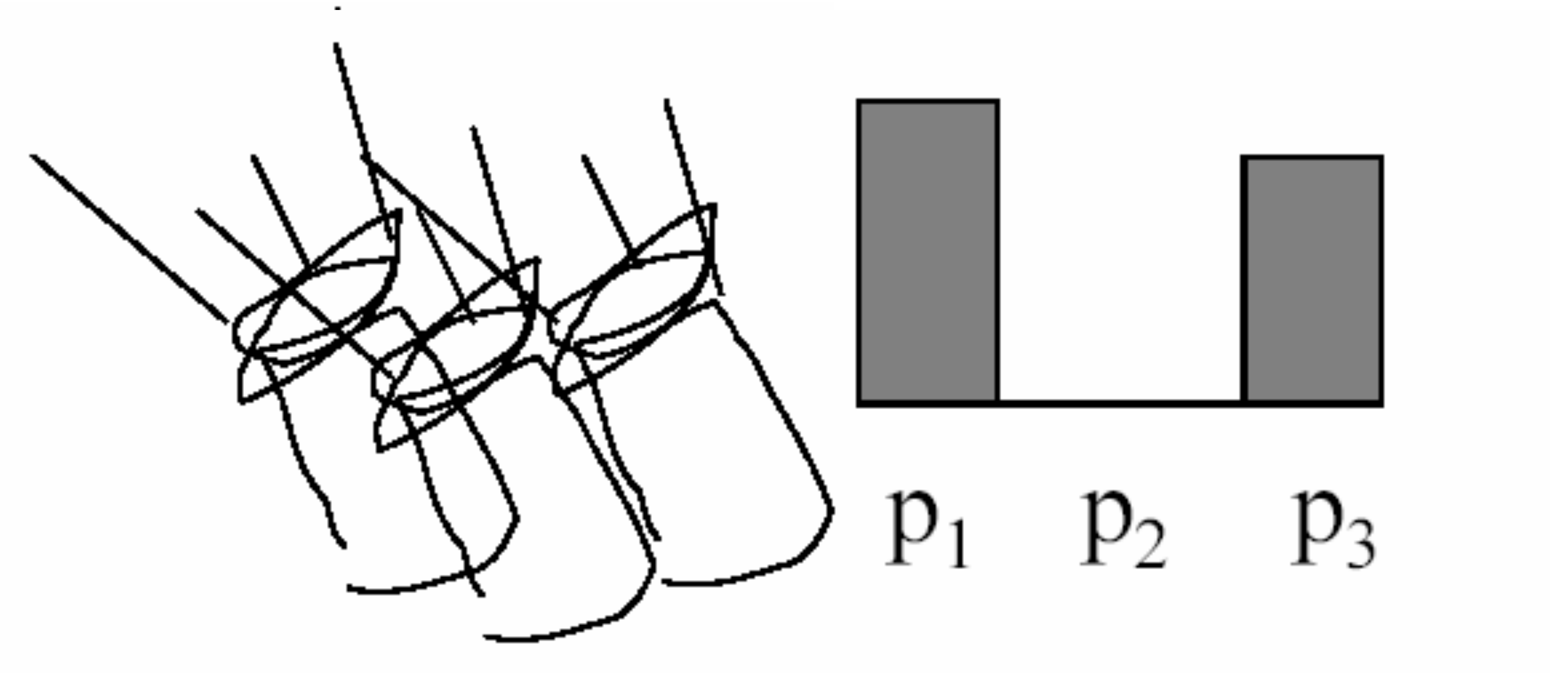


Example Credit: Bill Freeman

Example 2: Color Matching Experiment



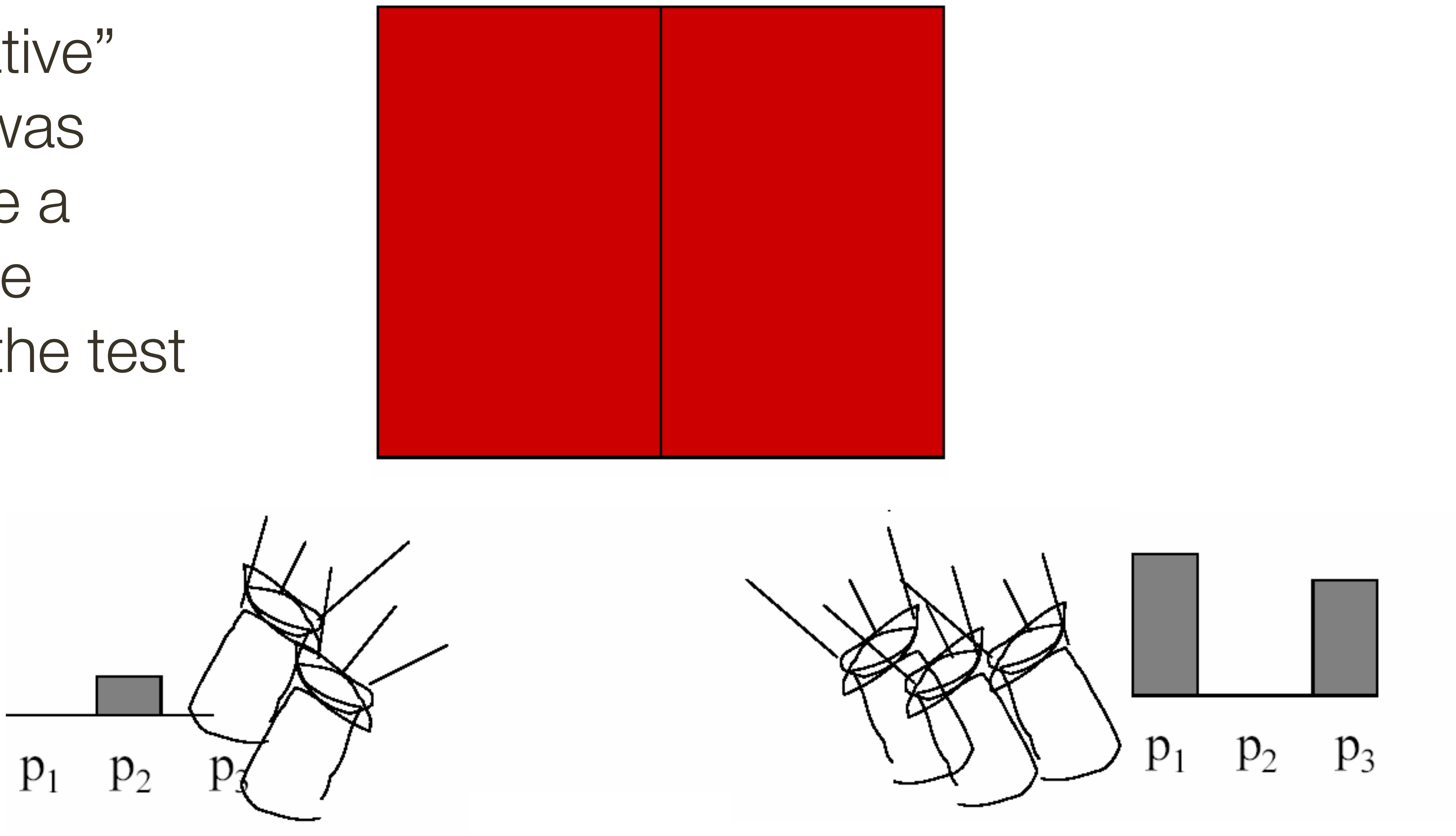




Example Credit: Bill Freeman

Example 2: Color Matching Experiment

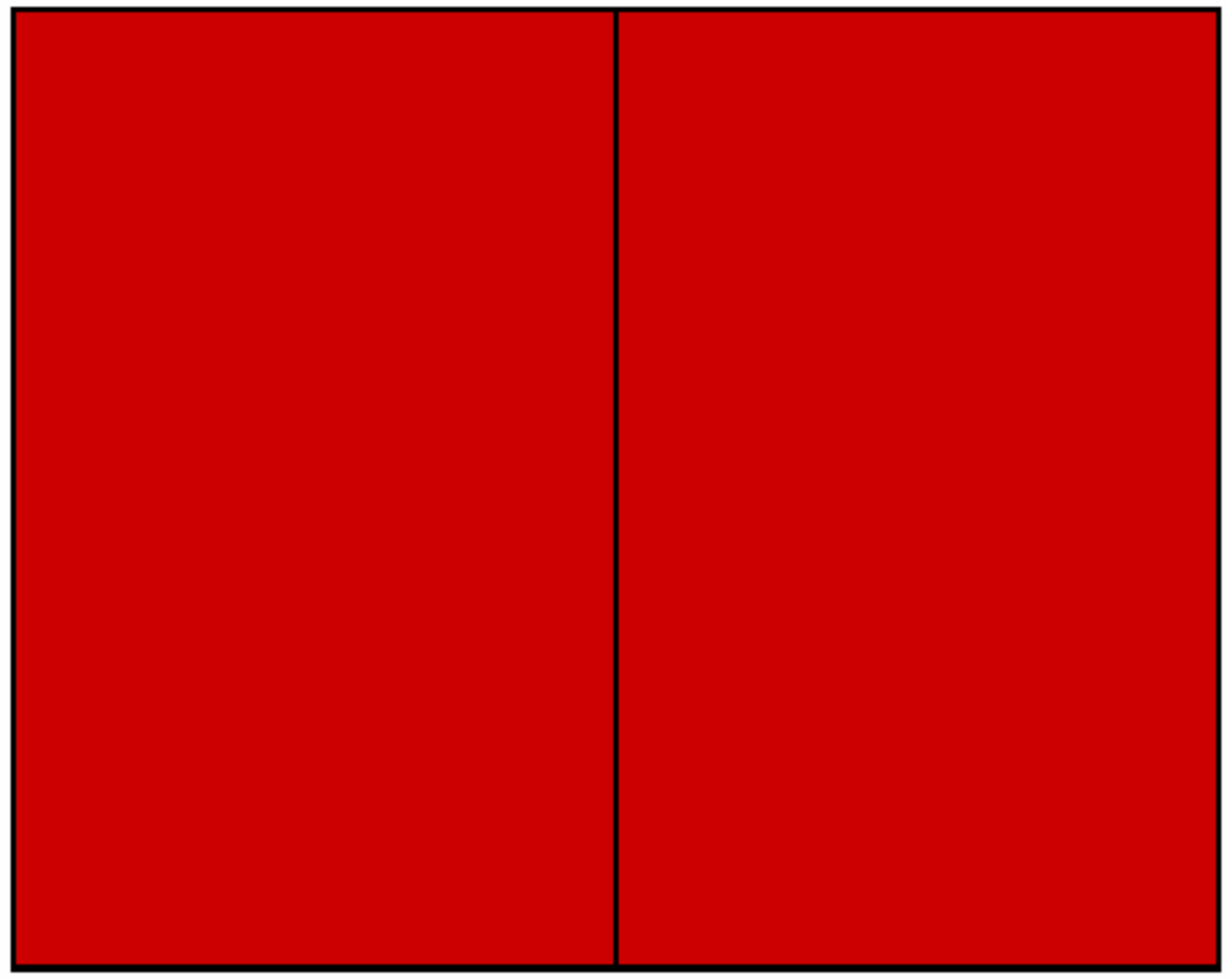
We say a “negative” amount of P_2 was needed to make a match , because we added it to the test color side



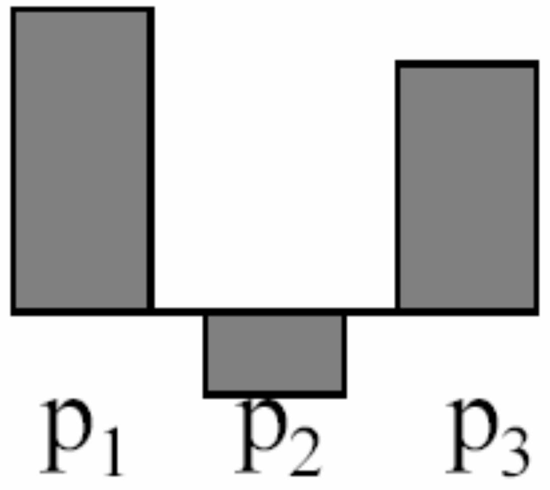
Example Credit: Bill Freeman

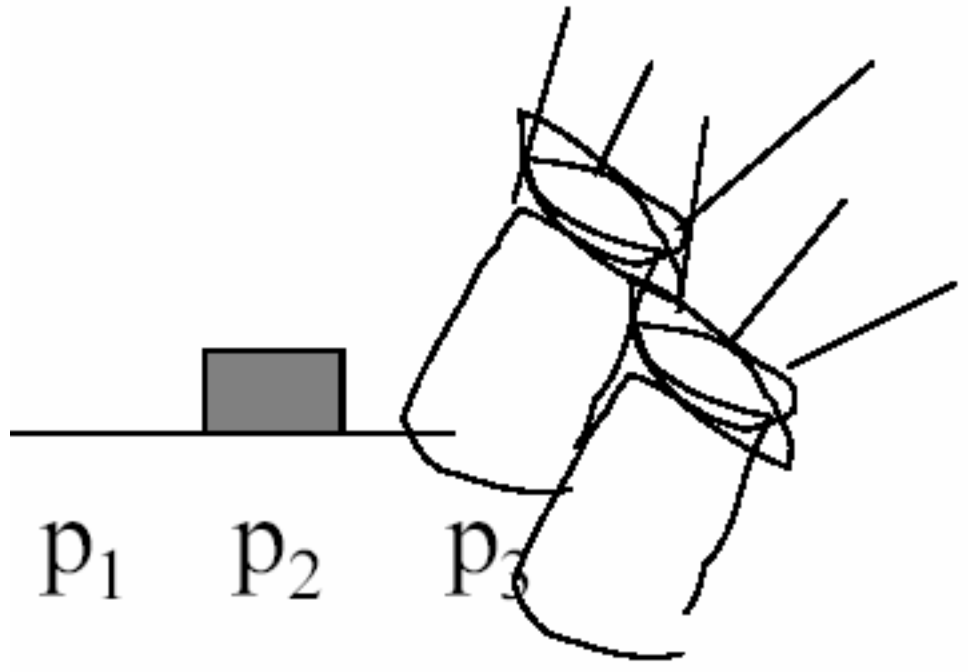
Example 2: Color Matching Experiment

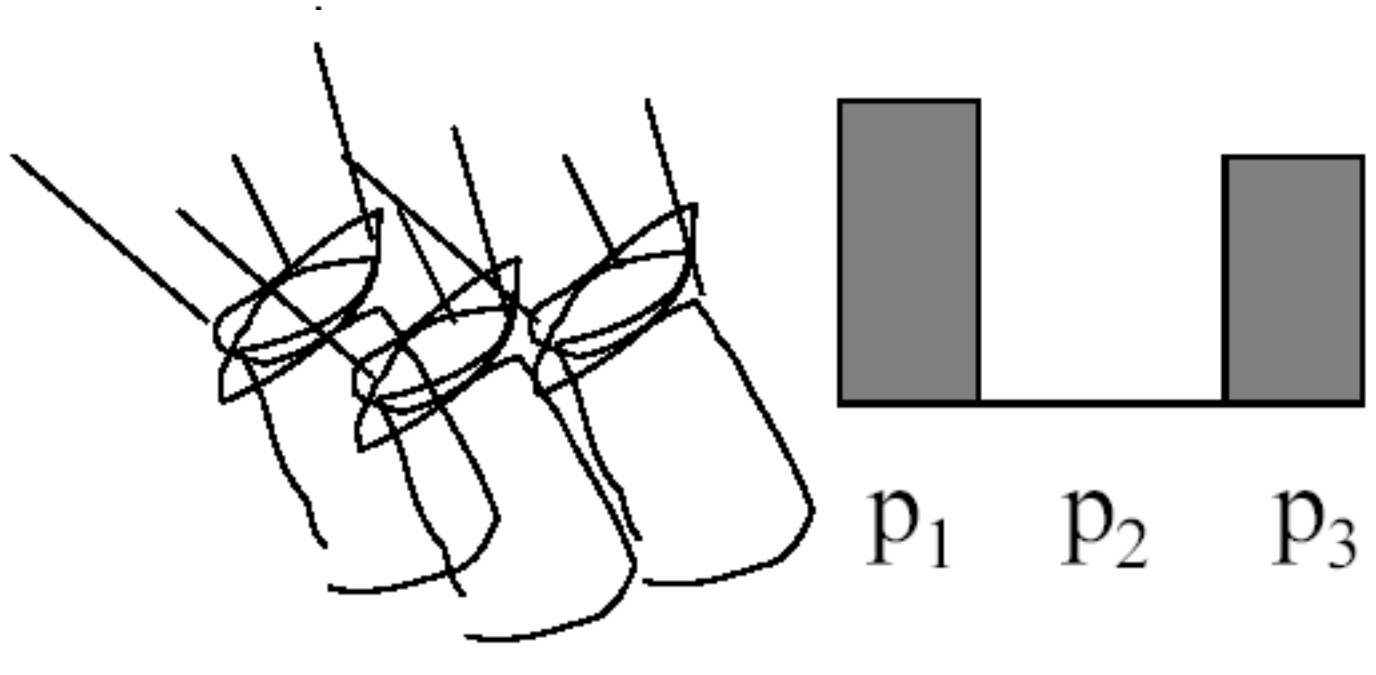
We say a "negative" amount of P_2 was needed to make a match , because we added it to the test color side



The primary color amount needed to match:







Example Credit: Bill Freeman

Color Matching Experiments

— Many colours can be represented as a positive weighted sum of A, B, C

— Write

$$M = aA + bB + cC$$

where the = sign should be read as “matches”

— This is **additive** matching

— Defines a colour description system
    — two people who agree on A, B, C need only supply (a, b, c)

Color Matching Experiments

— Some colours can't be matched this way

— Instead, we must write

$$M + aA = bB + cC$$

where, again, the = sign should be read as "matches"

— This is **subtractive** matching

— Interpret this as (–a, b, c)

Color Matching Experiments

— Some colours can’t be matched this way

— Instead, we must write

$$M + aA = bB + cC$$

where, again, the = sign should be read as “matches”

— This is **subtractive** matching

— Interpret this as (–a, b, c)

Problem for **designing displays**: Choose phosphors R, G, B so that **positive linear combinations** match a large set of colours

Principles of **Trichromacy**

Experimental facts:

Three primaries work for most people, provided we allow subtractive matching

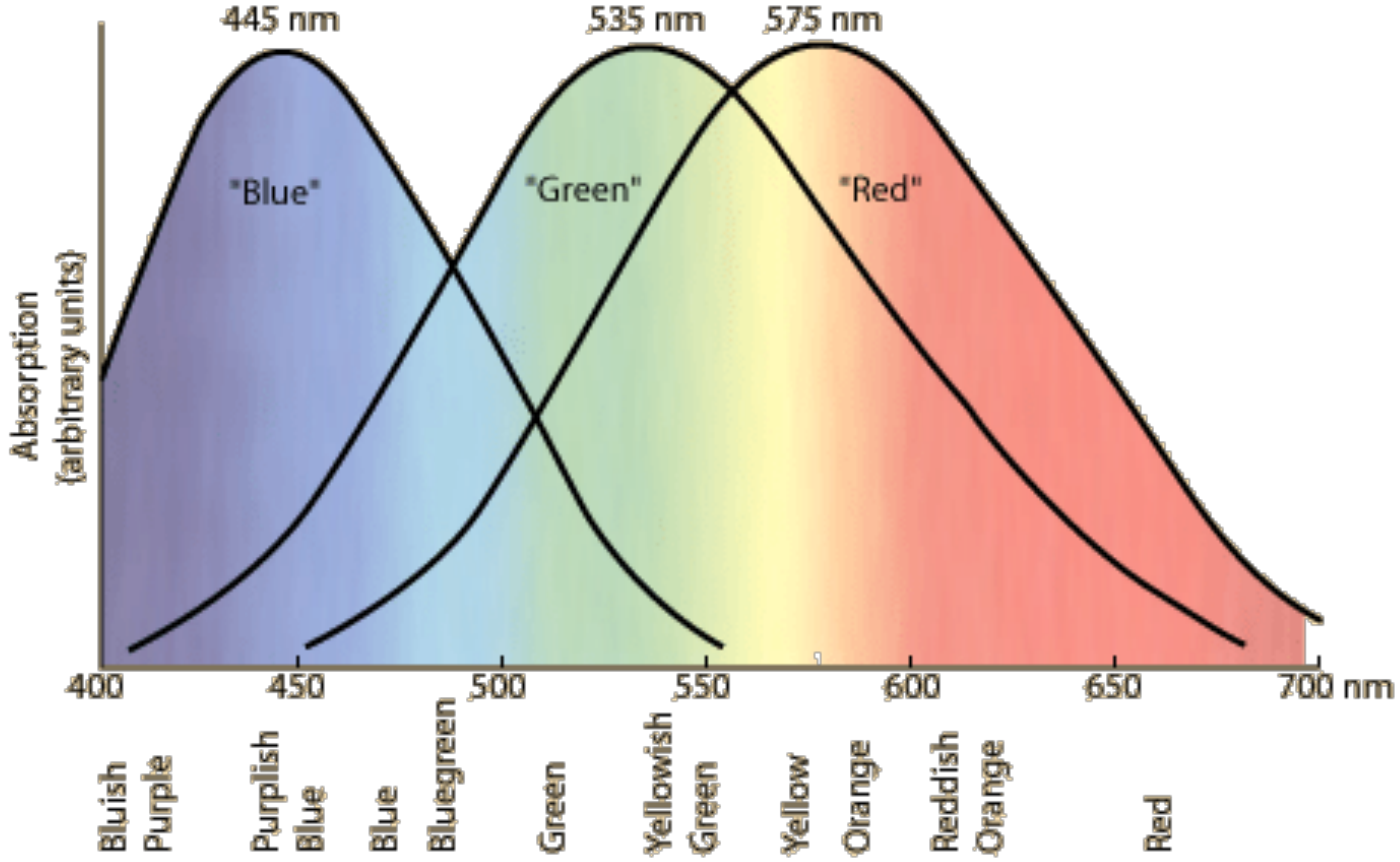
— Exceptional people can match with two or only one primary

— This likely is caused by biological deficiencies

Most people make the same matches

— There are some anomalous trichromats, who use three primaries but match with different combinations

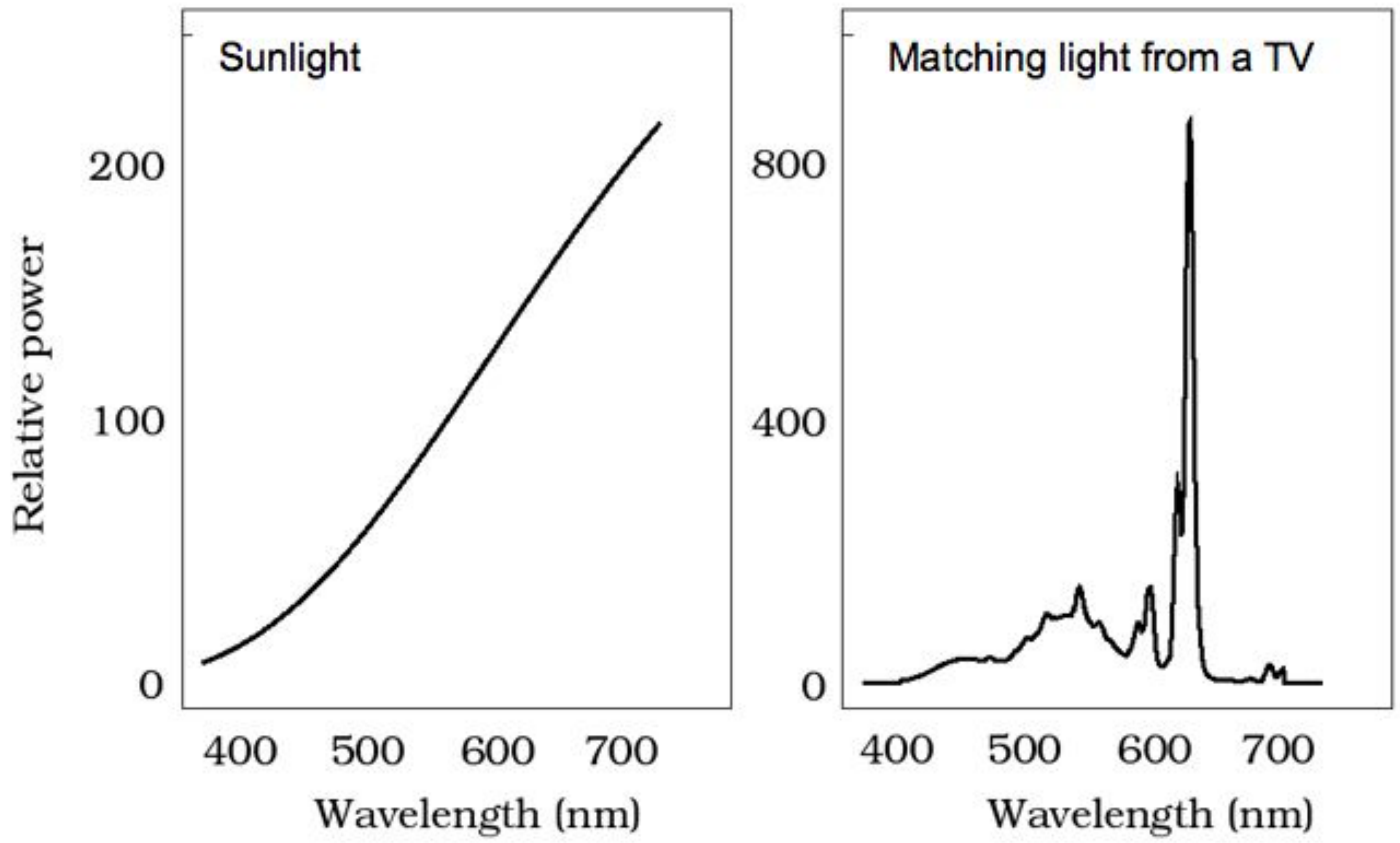
Human **Cone** Sensitivity



http://hyperphysics.phy-astr.gsu.edu/hbase/vision/colcon.html

Metameric Lights

Two lights whose spectral power distributions appear identical to most observers are called **metamers**.



(A) A tungsten bulb (B) TV monitor set to match (A)

Figure credit: Brian Wandell, Foundations of Vision, Sinauer Associates, 1995

Grassman's Laws

For colour matches:

— **symmetry**: $U = V \Leftrightarrow V = U$

— **transitivity**: $U = V \quad \text{and} \quad V = W \Rightarrow U = W$

— **proportionality**: $U = V \Leftrightarrow tU = tV$

— **additivity**: if any two of the statements are true, then so is the third

$$U = V,$$
$$W = X,$$
$$(U + W) = (V + X)$$

These statements mean that colour matching is, to an accurate approximation, linear.

Representing Colour

— Describing colours accurately is of practical importance (e.g. Manufacturers are willing to go to a great deal of trouble to ensure that different batches of their product have the same colour)

— This requires a standard system for representing colour.

Linear Color Spaces

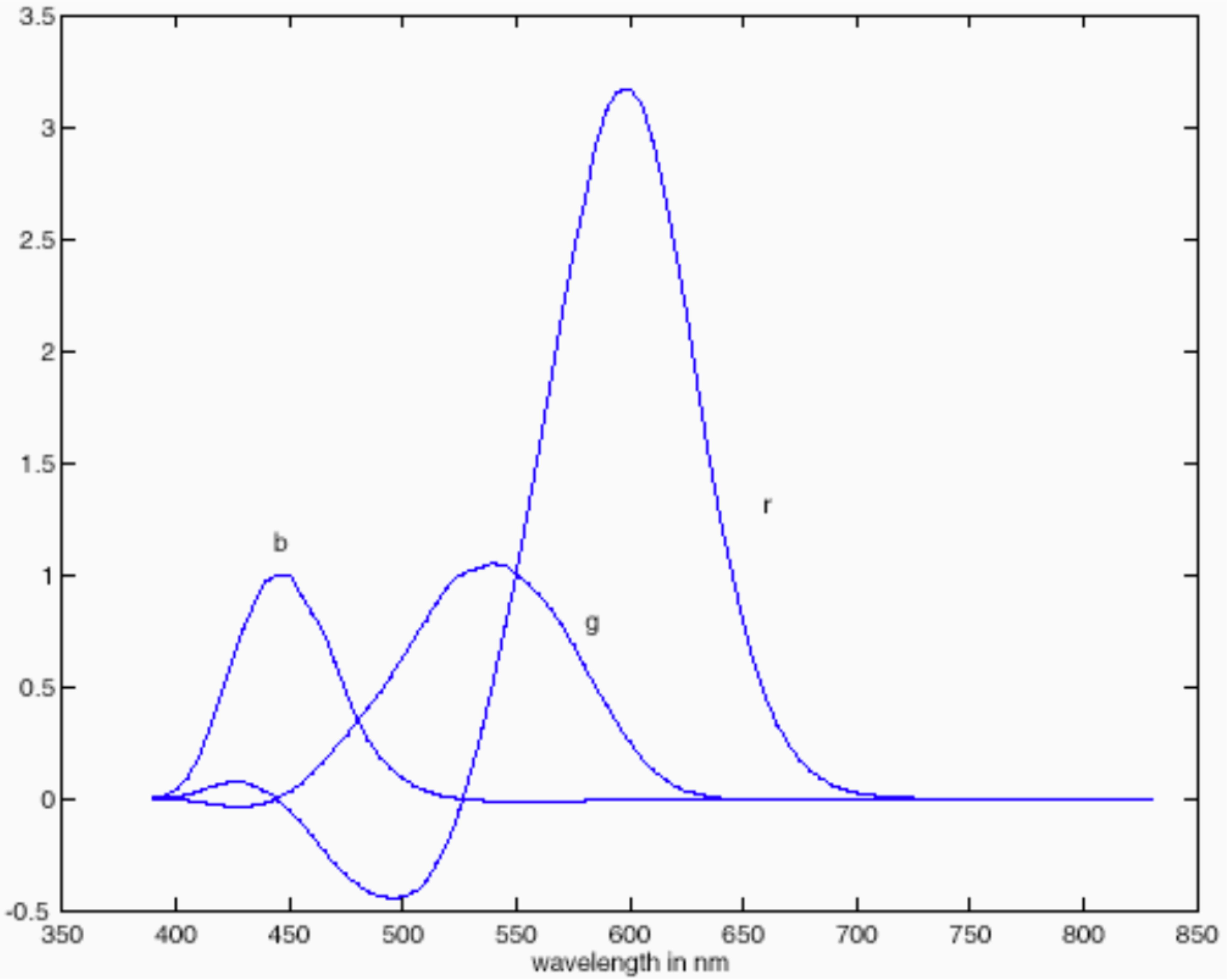
A choice of primaries yields a linear colour space
— the coordinates of a colour are given by the weights of the primaries used to match it

Choice of primaries is equivalent to choice of colour space

— **RGB**: Primaries are monochromatic energies, say 645.2 nm, 526.3 nm, 444.4 nm

— **CIE XYZ**: Primaries are imaginary, but have other convenient properties. Colour coordinates are (X , Y , Z), where X is the amount of the X primary, etc.

RGB Colour Matching Functions



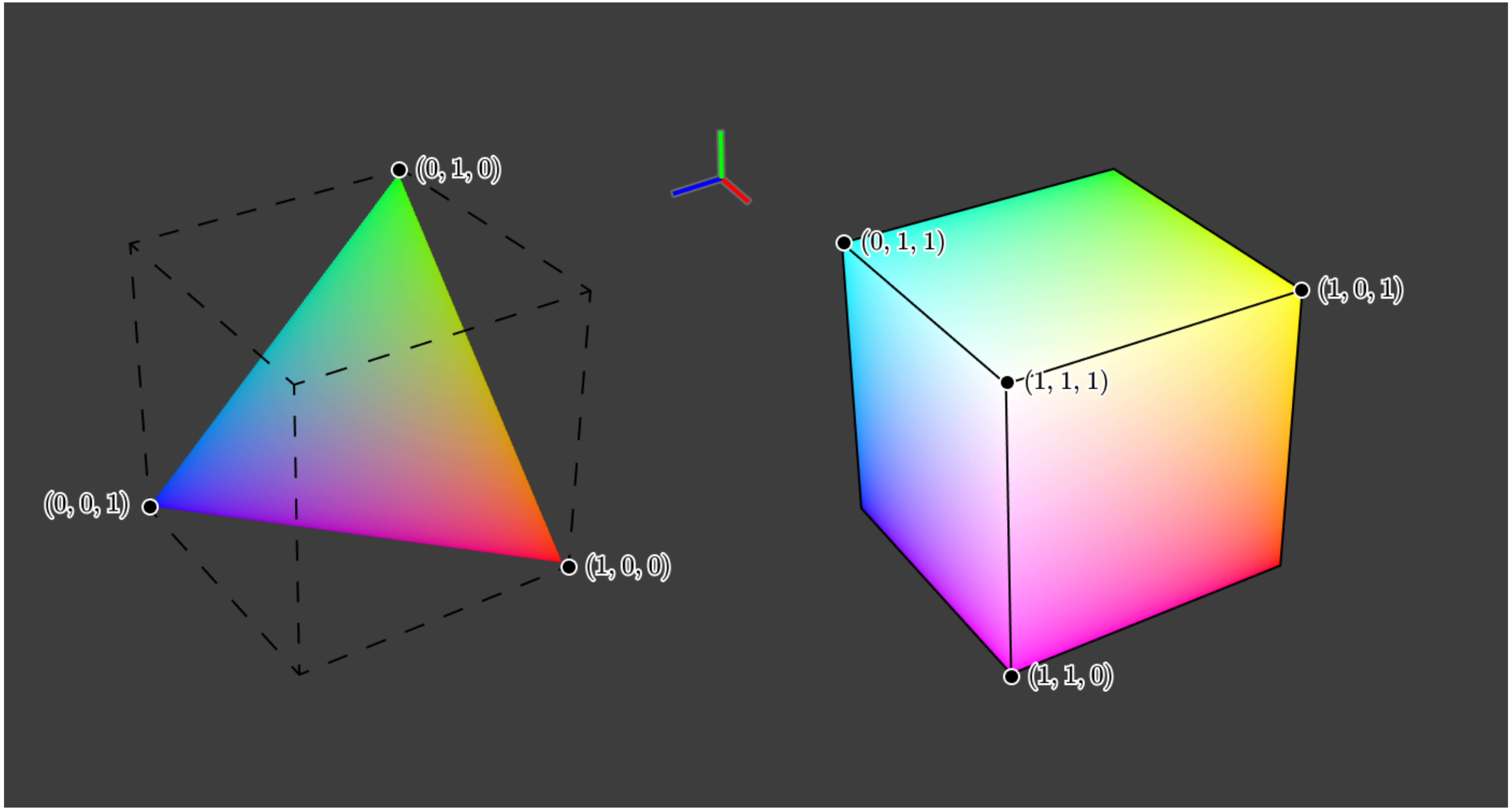
Forsyth & Ponce (2nd ed.) Figure 3.9

— Primaries monochromatic

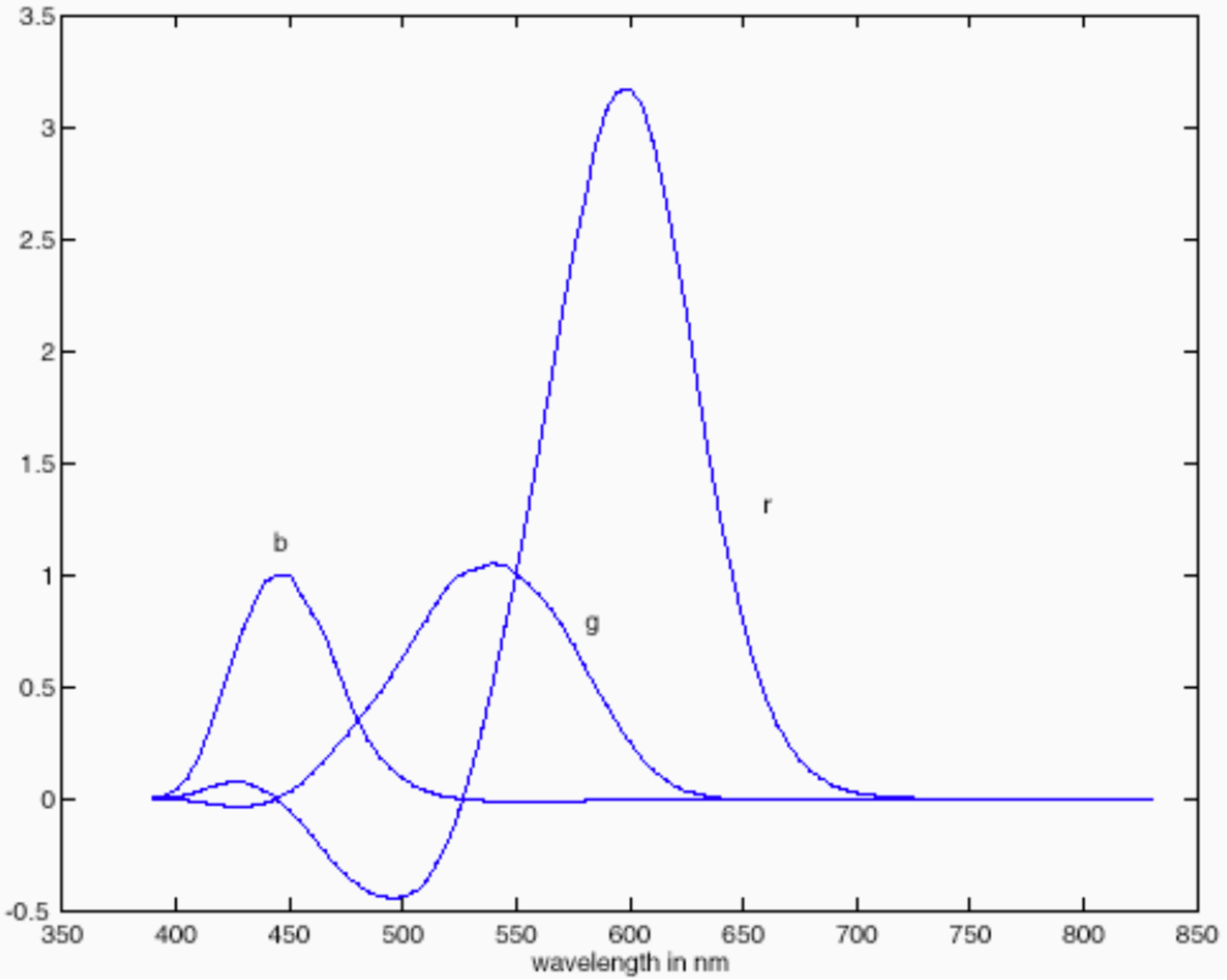
— Wavelengths 645.2, 526.3 and 444.4 nm

— Negative parts means some colours can be matched only subtractively

RGB Color Space



RGB Colour Matching Functions



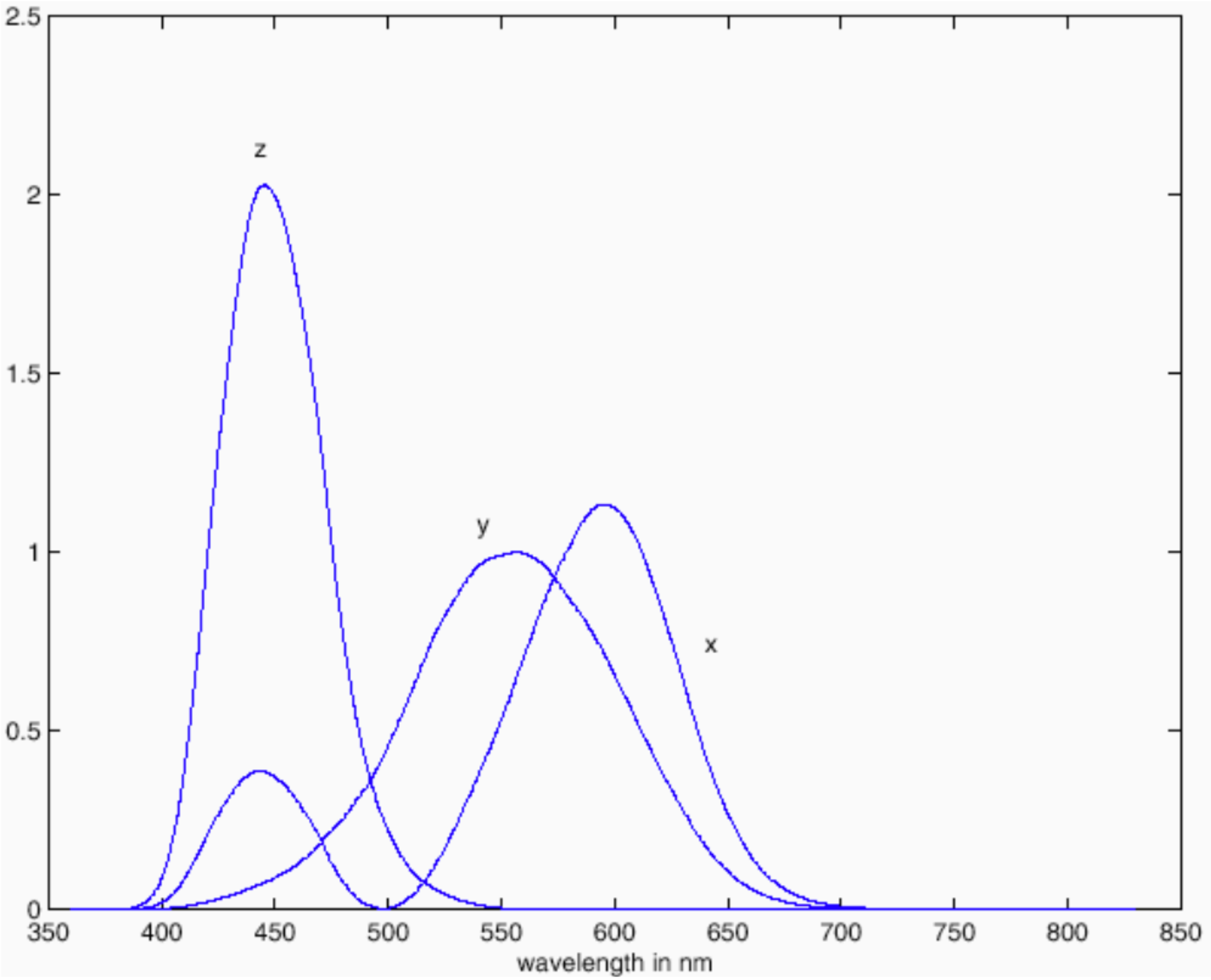
Forsyth & Ponce (2nd ed.) Figure 3.9

— Primaries monochromatic

— Wavelengths 645.2, 526.3 and 444.4 nm

— Negative parts means some colours can be matched only subtractively

RGB Colour Matching Functions



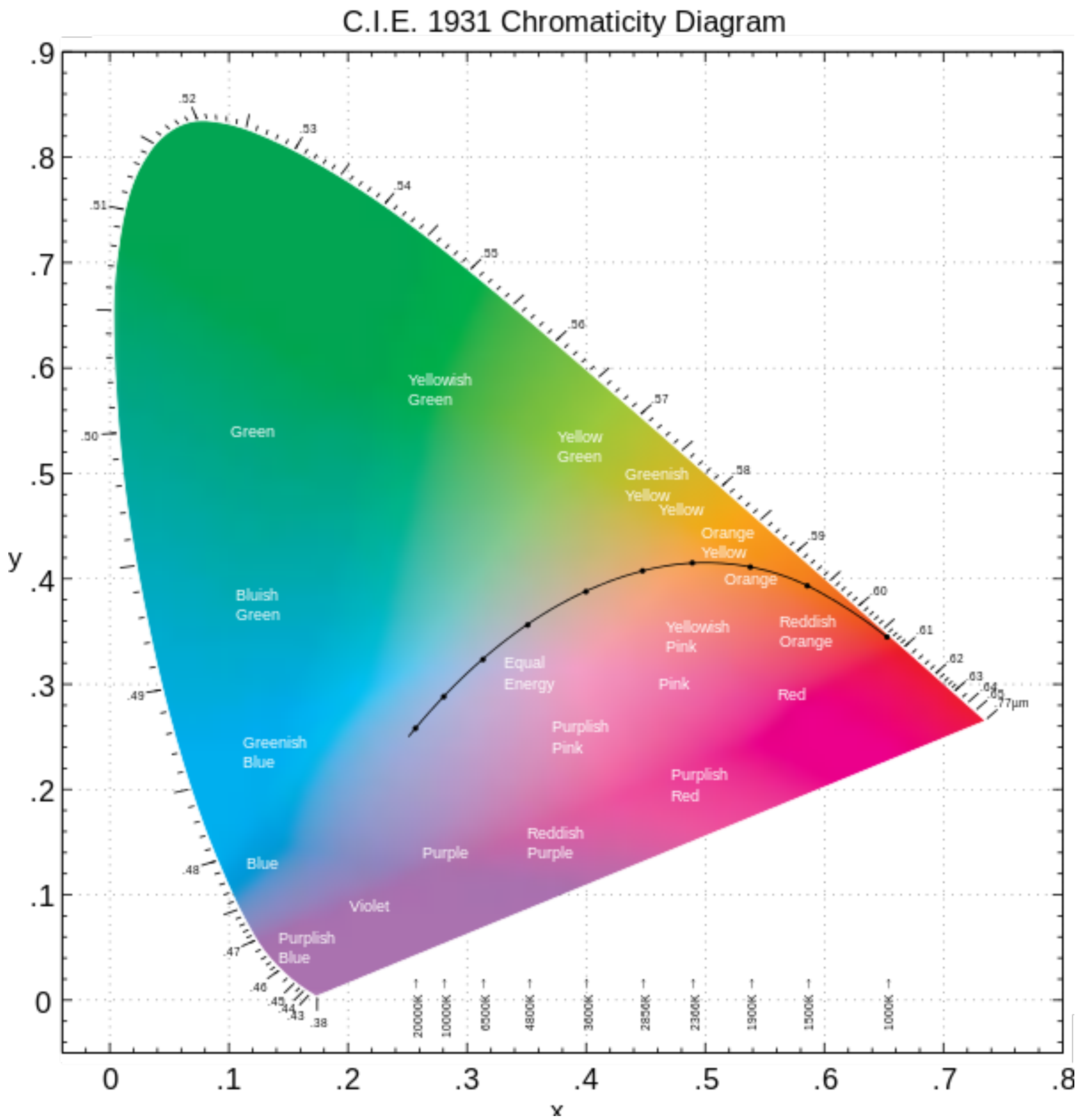
Forsyth & Ponce (2nd ed.) Figure 3.8

CIE XYZ: Colour matching functions are positive everywhere, but primaries are imaginary. Usually draw x, y, where

$$x = X/(X+Y+Z)$$
$$y = Y/(X+Y+Z)$$

Overall brightness is ignored

Geometry of Colour (CIE)



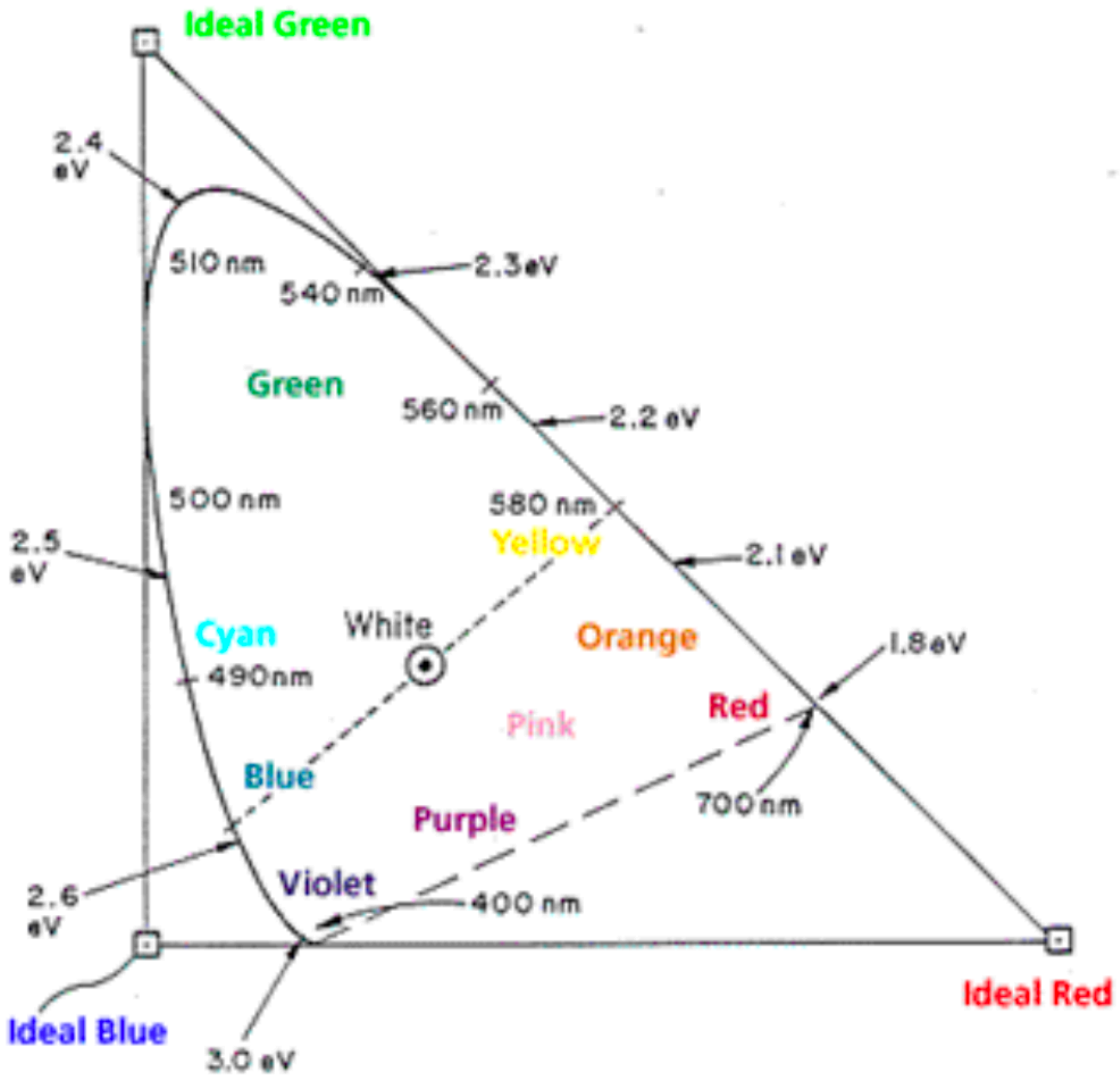
— White is in the center, with saturation increasing towards the boundary

— Mixing two coloured lights creates colours on a straight line

— Mixing 3 colours creates colours within a triangle

— Curved edge means there are no 3 actual lights that can create all colours that humans perceive!

Geometry of Colour (CIE)



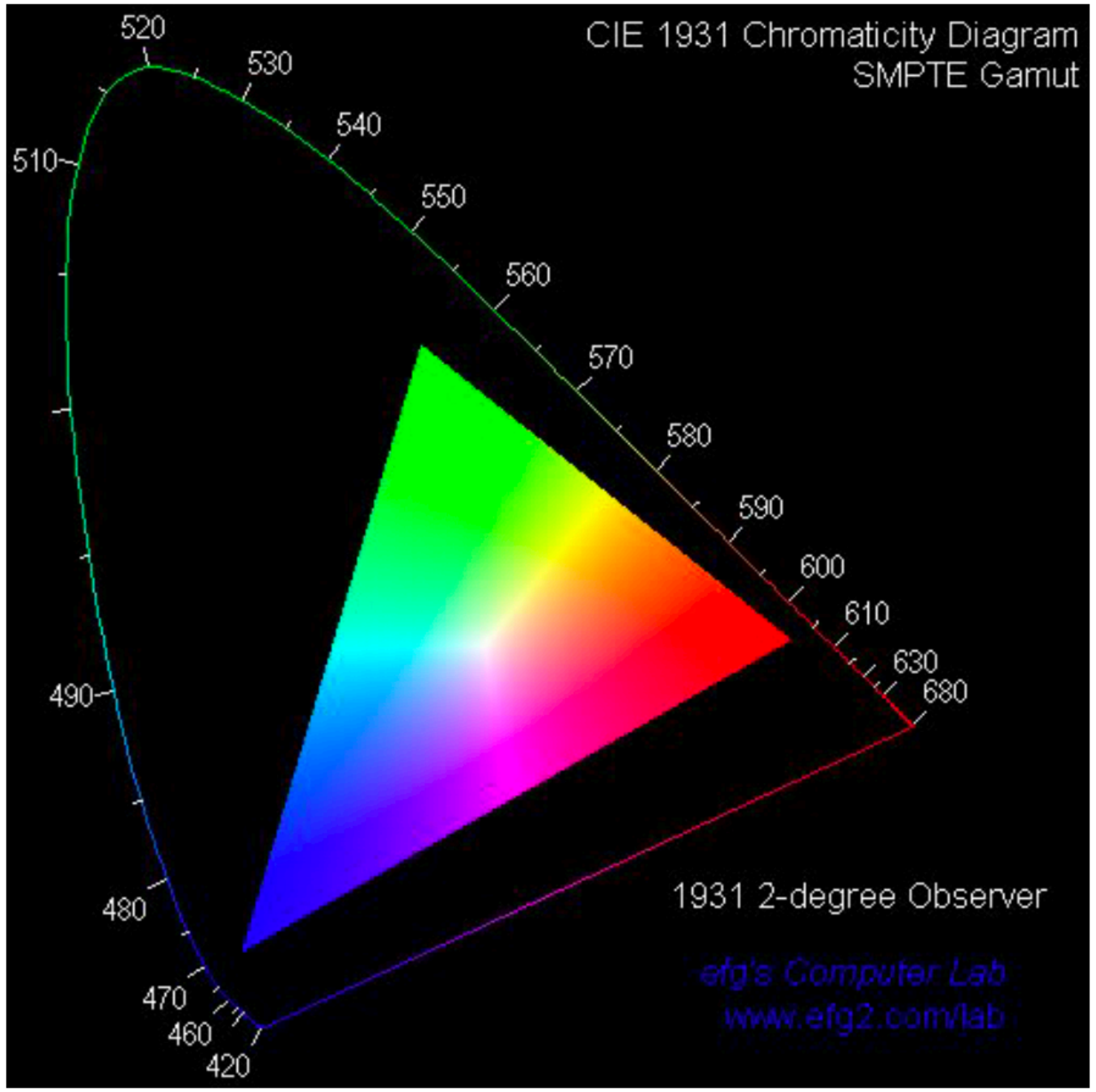
— White is in the center, with saturation increasing towards the boundary

— Mixing two coloured lights creates colours on a straight line

— Mixing 3 colours creates colours within a triangle

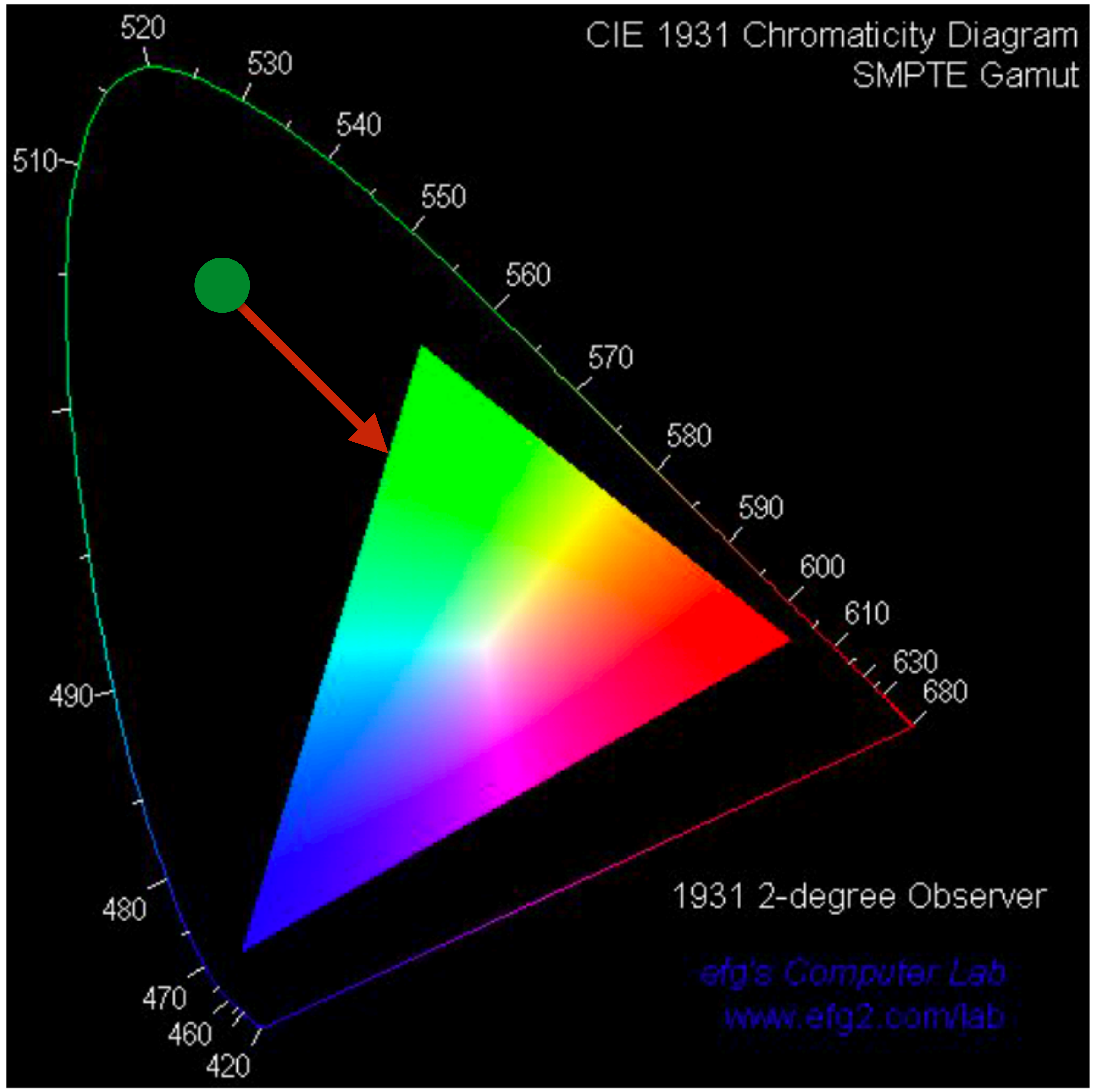
— Curved edge means there are no 3 actual lights that can create all colours that humans perceive!

RGB Colour Space



The sub-space of CIE colours that can be displayed on a typical computer monitor (phosphor limitations keep the space quite small)

RGB Colour Space



Adding **red** to the green color outside of the region brings it back to where it can be matched by **green** and **blue** RGB primaries

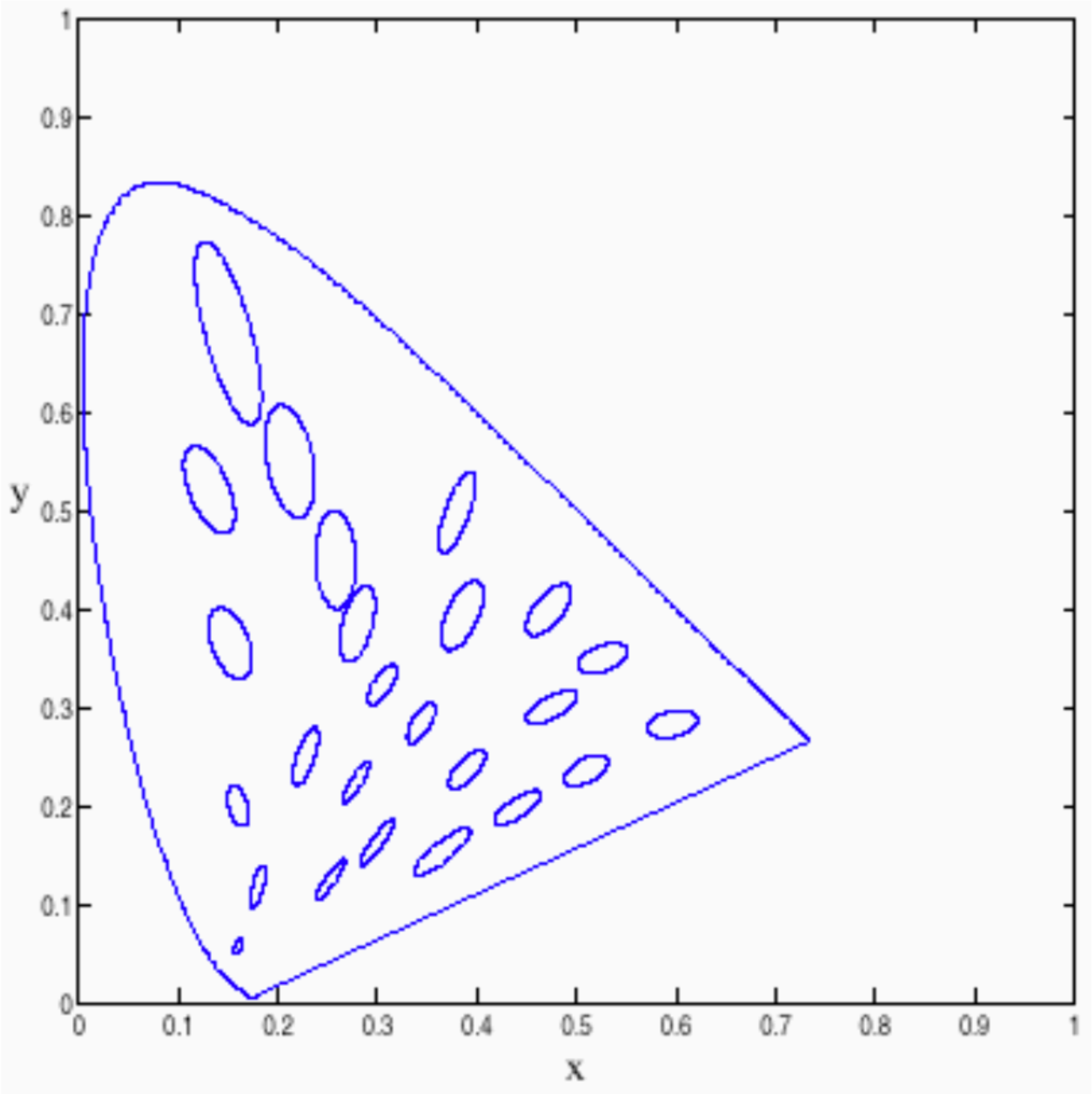
Uniform Colour Spaces

Usually one cannot reproduce colours exactly

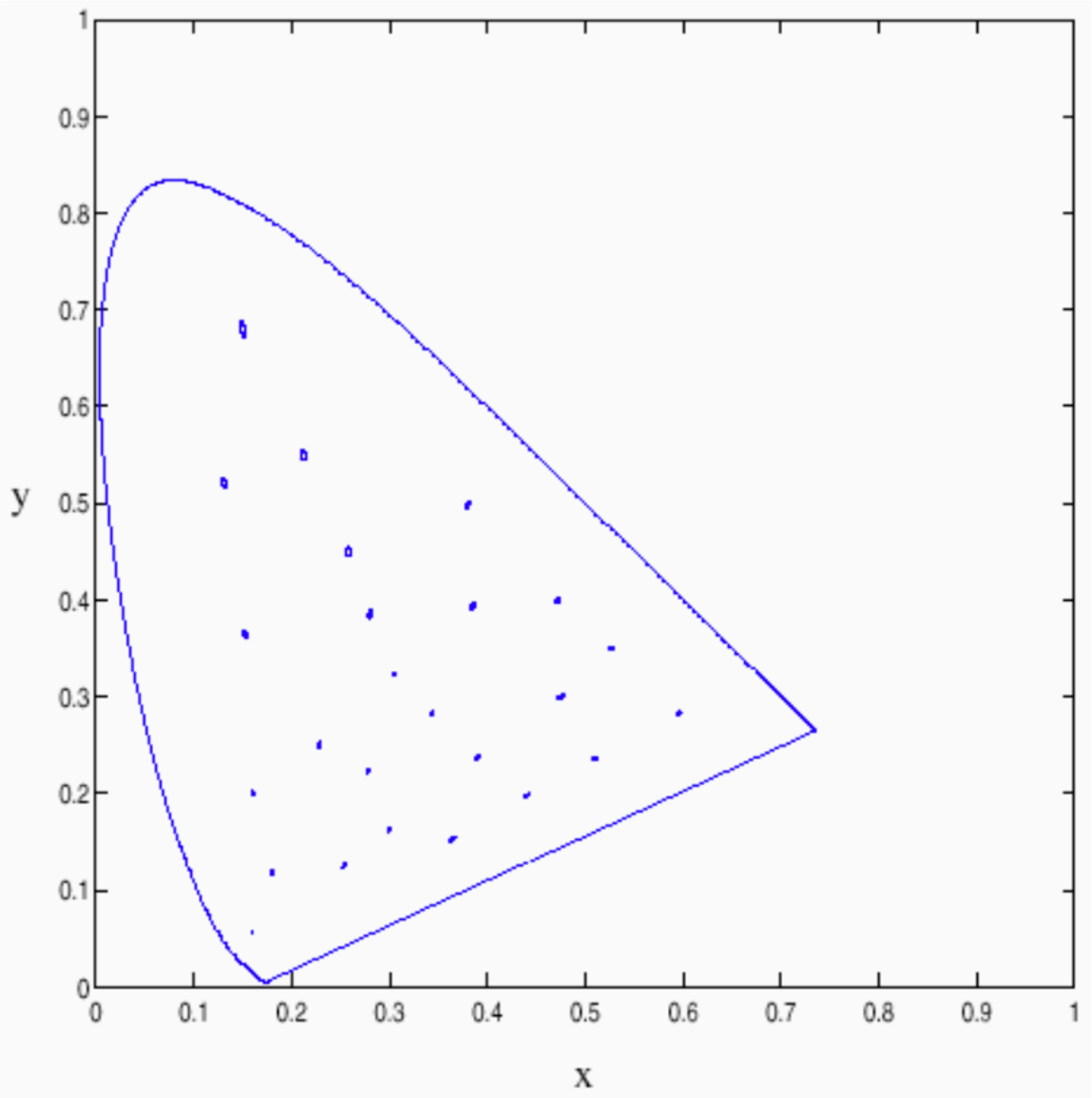
This means it is important to know whether a colour difference would be noticeable to a human viewer

Uniform Colour Spaces

McAdam Ellipses: Each ellipse shows colours perceived to be the same



10 times actual size



Actual Size

Forsyth & Ponce (2nd ed.) Figure 3.14

Uniform Colour Spaces

McAdam ellipses demonstrate that differences in x , y are a poor guide to differences in perceived colour

A **uniform colour space** is one in which differences in coordinates are a good guide to differences in perceived colour

— example: CIE LAB

HSV Colour Space

The coordinates of a colour in a linear space like RGB or CIE XYZ may not necessarily...

— encode properties that are common in language or important in applications

— capture human intuitions about the topology of colours, e.g. hue relations are naturally epxressed in a circle

HSV Colour Space

More natural description of colour for human interpretation

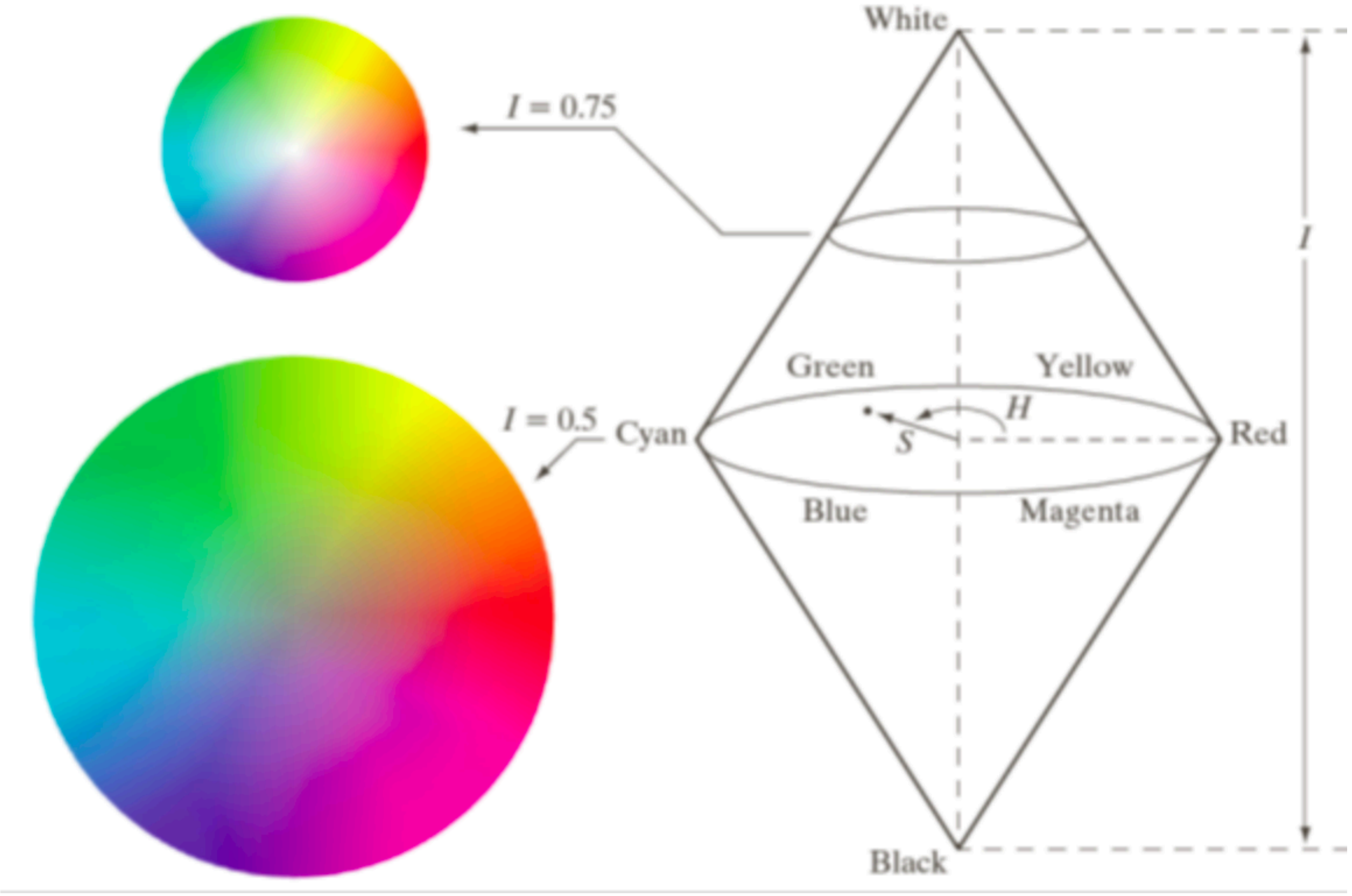
Hue: attribute that describes a pure colour
— e.g. ’red’, ’blue’

Saturation: measure of the degree to which a pure colour is diluted by white light
— pure spectrum colours are fully saturated

Value: intensity or brightness

Hue + saturation also referred to as **chromaticity**.

HSV Colour Space



Gonzalez and Woods, 2008

Colour **Constancy**

Image colour depends on both light colour and surface colour

Colour constancy: determine hue and saturation under different colours of lighting

It is surprisingly difficult to predict what colours a human will perceive in a complex scene
— depends on context, other scene information

Humans can usually perceive
— the colour a surface would have under white light

Colour **Constancy**

A classic experiment by Land and McCann

Environmental Effects

Chromatic adaptation: If the human visual system is exposed to a certain colour light for a while, colour perception starts to skew

Contrast effects: Nearby colours affect what is perceived

Summary

— Approaches to texture exploit pyramid (i.e. scaled) and oriented representations

— Human colour perception

  — colour matching experiments

  — additive and subtractive matching

  — principle of trichromacy

— RGB and CIE XYZ are linear colour spaces

— Uniform colour space: differences in coordinates are a good guide to differences in perceived colour

— HSV colour space: more intuitive description of colour for human interpretation

— (Human) colour constancy: perception of intrinsic surface colour under different colours of lighting