

CS 331

Computer Vision

09 / 15 / 2026

Instructor: Michael Eckmann

Today's Topics

- Questions? / Comments?
- HSV vs. RGB
- Grayscale
- Cross-correlation / Convolution operation
 - Smoothing mask
 - First derivative mask
 - Second derivative mask

Grayscale

- Last time we talked about RGB and grayscale.
- If we have RGB values for a pixel and we want to convert it to grayscale, we would like to get the intensity of the pixel.
- Grayscale images have just one value per pixel representing its intensity.
- Any thoughts on how you might convert RGB (each with 0-255) to Grayscale (0-255)?

Grayscale

- Conversion of an RGB image to grayscale is usually done by the following formula:
- $\text{Gray value} = 0.299 * R + 0.587 * G + 0.114 * B$
- Result will be a 1 channel image (1 value per pixel)
- Recall V channel in HSV
- V = Value --- high values are brighter, but this value for different hues/saturations do not all have the same perceived brightness
- V was calculated to be the largest of the R, G or B

Grayscale



- An original color image

Michael Eckmann - Skidmore
College - CS 331 - Fall 2026

Grayscale



- Grayscale using the formula $0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B$

Michael Eckmann - Skidmore
College - CS 331 - Fall 2026

Grayscale



- An original color image

Michael Eckmann - Skidmore
College - CS 331 - Fall 2026

Grayscale



- Grayscale using the formula $0.333 \cdot R + 0.333 \cdot G + 0.333 \cdot B$

Grayscale

- The Cross-correlation and Convolution operations on the next slides all assume a grayscale (i.e. one channel) image.
- Or you can do the cross-correlation or convolution on one channel of a multichannel image. E.g. just the V channel of an HSV image.

Reading

In Szeliski book

In section 3.2 Linear Filtering

read pages 119 (bottom) through 121,

skip page 122

and do not worry too much about the formulas

read page 123 up to middle of page 124 (up to and including 1st
paragraph of section 3.2.1 Separable filtering)

examine figure 3.14 on 125

read section 3.2.2 Examples of linear filtering which starts on bottom
of page 125 and ends top of 127

Cross-correlation

- A mask (aka kernel, aka filter) is usually a small (e.g. 3x3 or 5x5) square of values (sometimes called weights)
- Cross-correlation of a mask and an image involves aligning the origin (usually the middle cell) of the mask with a pixel in question and multiplying the corresponding mask values (weights) with the pixels in the original image and summing them all up. This summed up value is the value of the new pixel.
- Typical masks will either sum up to 0 or 1. A smoothing mask will sum up to 1. A derivative mask will sum up to 0.
- Concern:
 - The result of the sum of the multiplications is not guaranteed to be within the range 0 .. 255, if the weights do not add up to 1

Convolution

118

Computer Vision: Algorithms and Applications (September 20, 2020 draft)

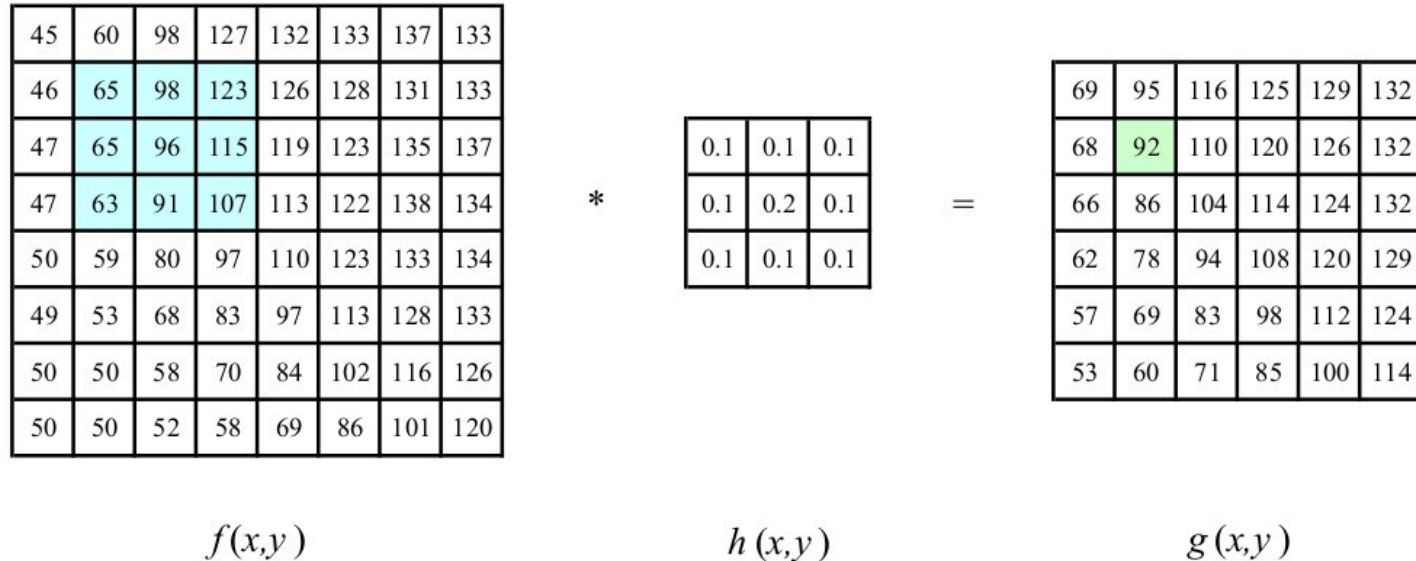


Figure 3.10 *Neighborhood filtering (convolution): The image on the left is convolved with the filter in the middle to yield the image on the right. The light blue pixels indicate the source neighborhood for the light green destination pixel.*

$$.1*65 + .1*98 + .1*123 + .1*65 + .2*96 + .1*115 + .1*63 + .1*91 + .1*107 = 91.9$$

Derivative and Smoothing masks

- Derivative masks
 - values have opposite signs in order to obtain a high response in signal regions of high contrast
 - sum of values is 0, so a 0 results from constant regions
 - first derivative masks produce high abs values at points of high contrast
 - second derivative masks produce zero-crossings at points of high contrast
- Smoothing masks
 - values are positive and sum to 1, so resulting value is same as input in constant regions, but has a blurring effect on non-constant regions

Smoothing examples



Michael Eckmann - Skidmore
College - CS 331 - Fall 2026

Smoothing examples



Michael Eckmann - Skidmore
College - CS 331 - Fall 2026

Smoothing examples



Michael Eckmann - Skidmore
College - CS 331 - Fall 2026

Let's look at an example

- Let's consider this mask

1 0 -1

- Applied to this 1 dimensional image:

50 50 50 50 50 125 125 125 125 125

Let's look at an example

- Let's consider this mask

1 0 -1

- Applied to this 1 dimensional image:

50 50 50 50 50 125 125 125 125 125

- Resulting image:

0 0 0 -75 -75 0 0 0

Let's look at an example

- Let's consider this mask

1 0 -1

- Applied to this 1 dimensional image:

50 50 50 50 50 125 125 125 125 125

- Resulting image:

0 0 0 -75 -75 0 0 0

- Notice: in areas of constant values, the result is 0, but in areas where there are differences/high contrast, the result is non-zero (high absolute value).

Second derivative masks

- Examples of two dimensional second derivative masks:

$$\begin{matrix} 0 & -1 & 0 \end{matrix}$$
$$\begin{matrix} -1 & 4 & -1 \end{matrix}$$
$$\begin{matrix} 0 & -1 & 0 \end{matrix}$$
$$\begin{matrix} -1 & -1 & -1 \end{matrix}$$
$$\begin{matrix} -1 & 8 & -1 \end{matrix}$$
$$\begin{matrix} -1 & -1 & -1 \end{matrix}$$

Second derivative masks

- Example of a one dimensional second derivative mask:

-1 2 -1

Second derivative masks

original image:

10 10 10 10 15 20 25 30 30 30 30

notice there is a gradual edge (I will draw on board) there and if you used a 1st derivative mask like: 1 0 -1 we get:

0 0 -5 -10 -10 -10 -5 0 0

The first derivative mask seems to detect stronger edge pixels in the 3 middle ones and less strong edge pixels next to those.

original image again:

10 10 10 10 15 20 25 30 30 30 30

if you used a 2nd derivative mask like: -1 2 -1 we get:

0 0 -5 0 0 0 5 0 0

Second derivative masks

The second derivative mask detects where the two "sides" of the edge are.

Here the second derivative mask is negative where the image function is **concave up** and is positive where the image function is **concave down**

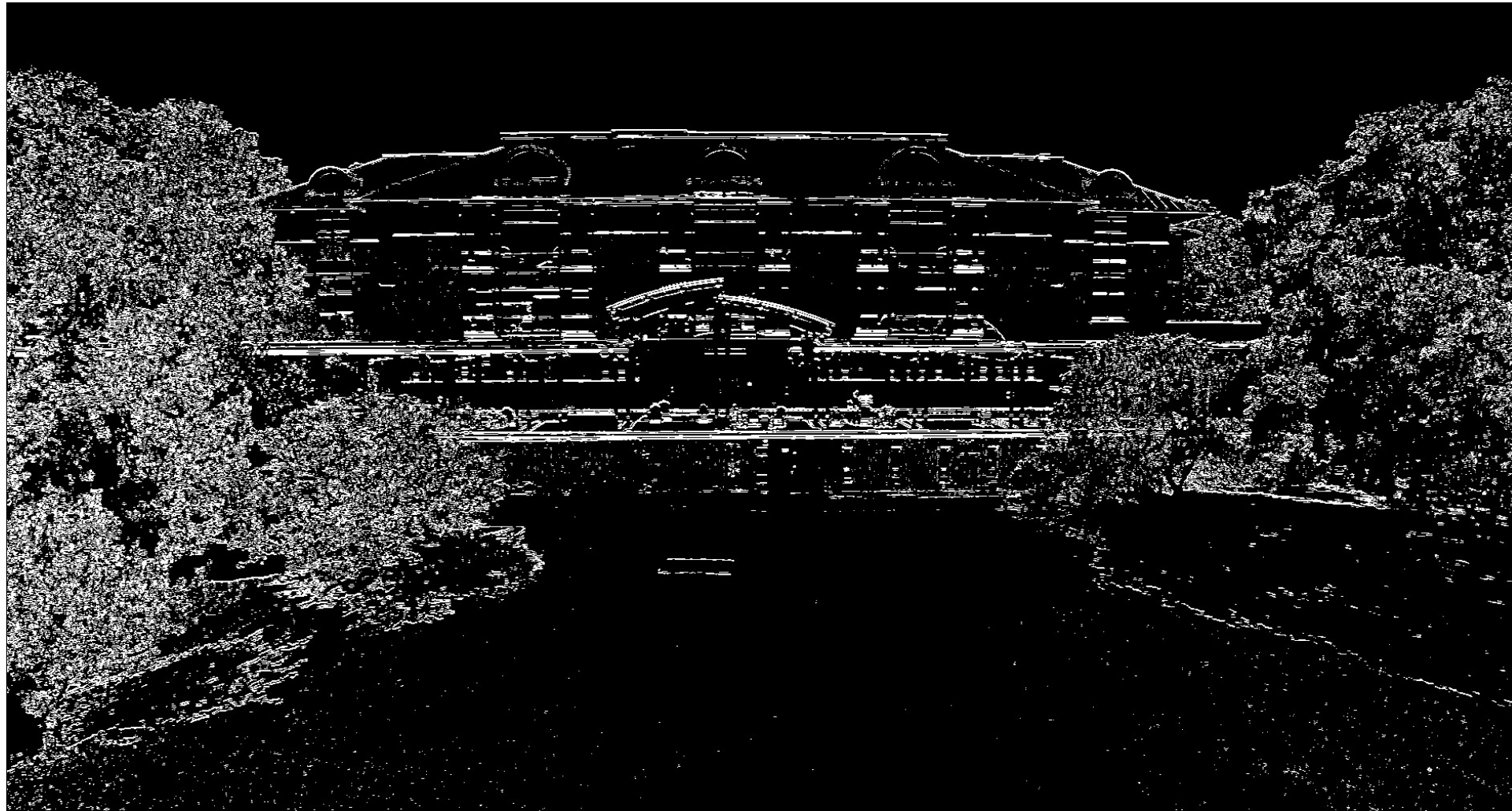
You may have learned in calculus that this is the opposite. So a true second derivative mask really should be something like: $\begin{bmatrix} 1 & -2 & 1 \end{bmatrix}$ rather than $\begin{bmatrix} -1 & 2 & -1 \end{bmatrix}$ and that will get you a positive value of the second derivative when the function is concave up and a negative for concave down

Show some edge images (based on thresholds)



Michael Eckmann - Skidmore
College - CS 331 - Fall 2026

Show some edge images (based on thresholds)



Michael Eckmann - Skidmore
College - CS 331 - Fall 2026

Show some edge images (based on thresholds)



Show some edge images (based on thresholds)



Show some edge images (based on thresholds)

