Image Pyramids

Computational Photography (CSCI 3240U)

Faisal Z. Qureshi
http://vclab.science.ontariotechu.ca
OntarioTech UNIVERSITY

Quiz – Nov 6

7x7 image. $f(x,y) = 3x + y^2$ $x, y \in [0, 6]$.

Compute gradient at $(2,3)$

Solution

$\frac{\partial f}{\partial x} = 3$ $\frac{\partial f}{\partial y} = 2y$

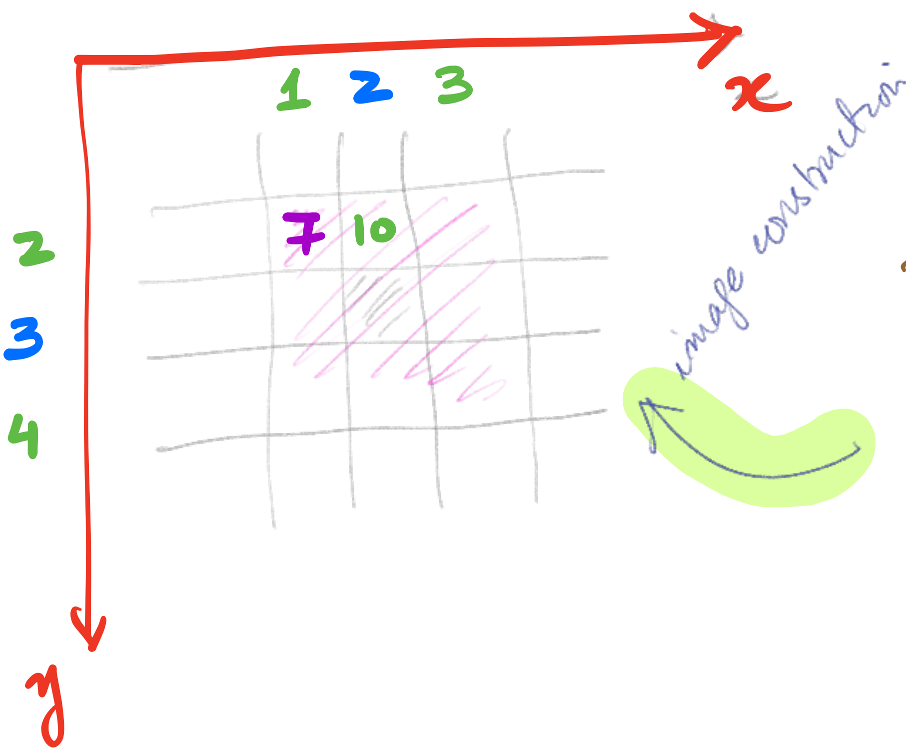
$$\nabla f = \begin{bmatrix} 3 \\ 2y \end{bmatrix}$$

$$\nabla f \Big|_{(2,3)} = \begin{bmatrix} 3 \\ 6 \end{bmatrix}$$

Example

$f(x,y,z) = x^3 y + z y^2$

$$\nabla f = \begin{bmatrix} \partial f/\partial x \\ \partial f/\partial y \\ \partial f/\partial z \end{bmatrix} = \begin{bmatrix} 3x^2 y \\ x^3 + 2yz \\ y^2 \end{bmatrix}$$



$f(x,y) = 3x + y^2$

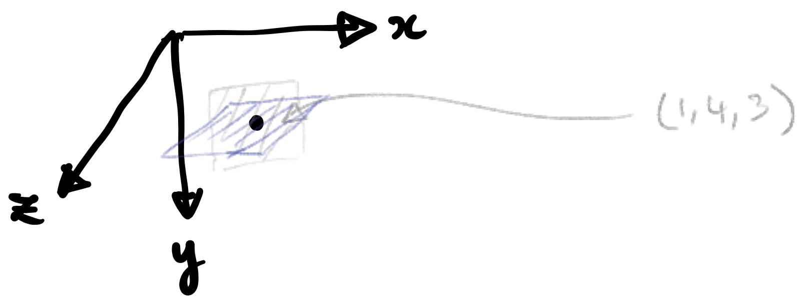
$f(1,2) = 3 + 4 = 7$

$f(2,2) = 6 + 4 = 10$

A 3D Example:

$$f(x,y) = x^3 y + y^2 z$$

Evaluate gradient at location $(1,4,3)$.



For x and y.

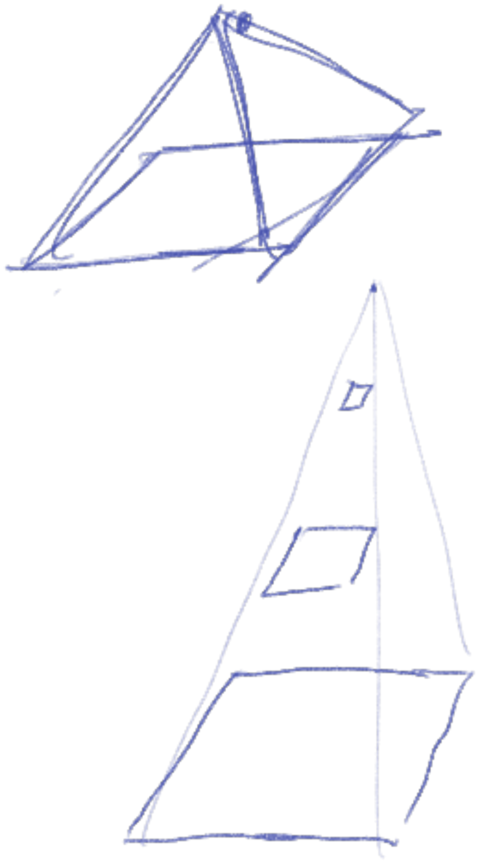
(0,4,3) (1,4,3) (2,4,3)
(0,3,3) (1,3,3) (2,3,3)
(0,5,3) (1,5,3) (2,5,3)

For z

(0,4,2) (1,4,2) (2,4,2)
(0,4,3) (1,4,3) (2,4,3)
(0,4,4) (1,4,4) (2,4,4)

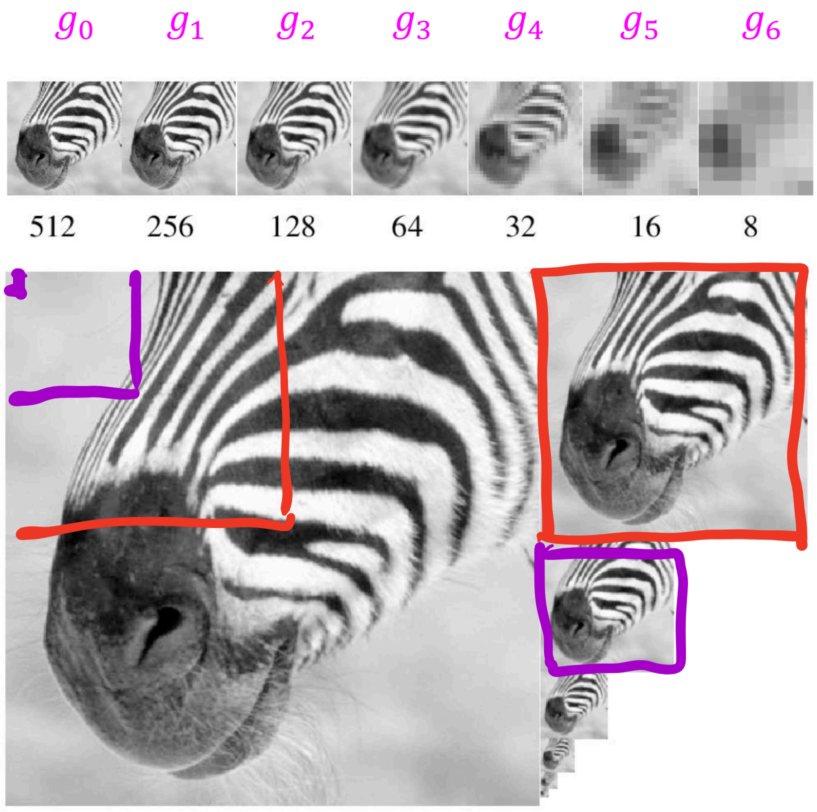
Today’s lecture

- Gaussian image pyramids
- Laplacian image pyramids
- Laplacian blending



Gaussian image pyramids

- Blur the image with a Gaussian kernel
- Reduce image dimensions by half
  - Discard every other row and column
- Repeat
  - Till the desired numbers of levels have been reached or till the image because 1x1



Courtesy: Forsyth

How to construct a Gaussian Pyramid.

Subsample operation.

$$\begin{bmatrix} 3 & 4 & 6 & 1 \\ 2 & 1 & 0 & 0 \\ 9 & 10 & 3 & 7 \\ 8 & 1 & 3 & 4 \end{bmatrix} \longrightarrow \begin{bmatrix} 3 & 6 \\ 9 & 3 \end{bmatrix}$$

If we just do this, we will observe aliasing artifacts.

↑ avoid

low-pass filter

discard every other row and every other column.

Scheme:

$g_0 \leftarrow I$ (input image)

$g_0' = G_\sigma * g_0 \longleftarrow g_0' = w * g_0$

↑ low-pass filter

$g_1 = \text{subsample}(g_0')$

$g_1' = G_\sigma * g_1$

$g_2 = \text{subsample}(g_1')$

$\ddots$

$g_{L-1}' = G_\sigma * g_{L-1}$

$g_L = \text{subsample}(g_{L-1}')$

$$I \longrightarrow g_0, g_1, g_2, \dots, g_L$$

$\in \mathbb{R}^{W \times H}$ $\qquad$ $\in \mathbb{R}^{\frac{W}{2^L} \times \frac{H}{2^L}}$

Lets say $I \in \mathbb{R}^{6000 \times 4000}$

↳ Construct a Gaussian pyramid and specify how much memory we need to store this pyramid.

Answer: ① Find out possible levels.

$$\frac{4000}{2^L} \geq 1$$

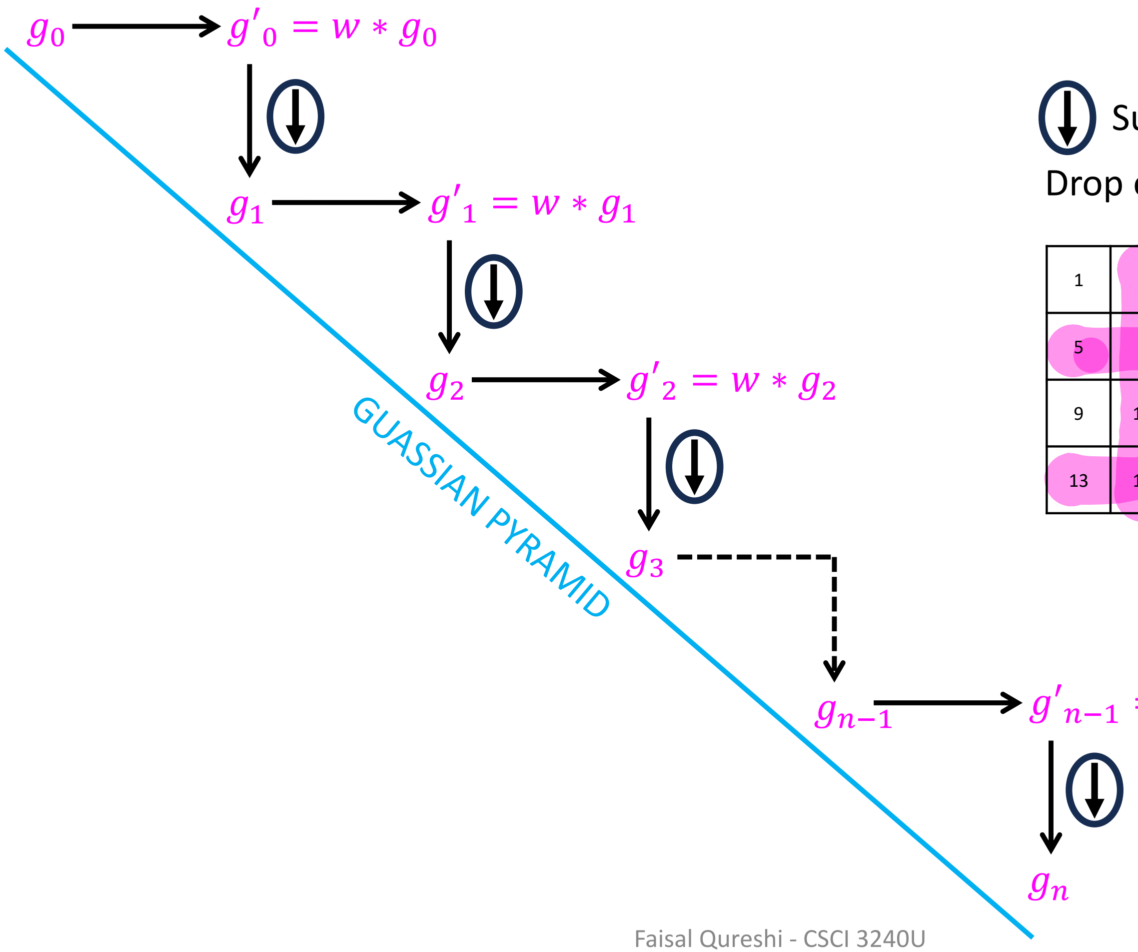
$$\Rightarrow 4000 \geq 2^L$$

$$\Rightarrow \log 4000 \geq L \log 2$$

$$\Rightarrow L \leq \log_2 4000$$

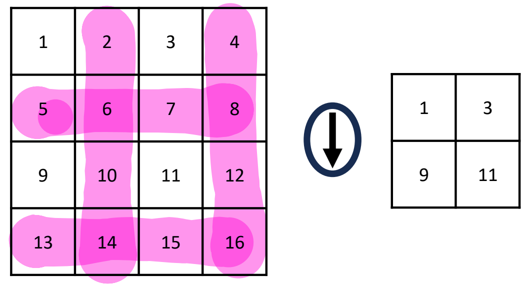
$$\Rightarrow L \leq \log_2 4000 \approx 11.96..$$

Gaussian image pyramids



Subsample

Drop every other row and column



Laplace operator

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

Compute the Laplace operator output for the shaded pixel

$I =$

1	1	9	8	1
8	8	8	8	8
1	3	5	8	1
5	3	2	8	6

Example:

$f(x,y) = x^3 y + y^2$

$\frac{\partial f}{\partial x}$ $\quad f_x = 3x^2 y \qquad f_y = x^3 + 2y$

$\frac{\partial^2 f}{\partial x^2}$ $\quad f_{xx} = 6xy \qquad f_{yy} = 2$

$\nabla^2 f = 6xy + 2$

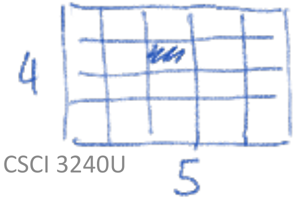
① $I_x = H_x * I$

② $I_{xx} = H_x * I_x$

③ $I_y = H_y * I$

④ $I_{yy} = H_y * I_y$

⑤ $\nabla^2 I = I_{xx} + I_{yy}$



Sobel

$H_x =$

1	0	-1
2	0	-2
1	0	-1

$H_y =$

1	2	1
0	0	0
-1	-2	-1

3x3 Gaussian kernel (approx.)

$G =$ 1/16

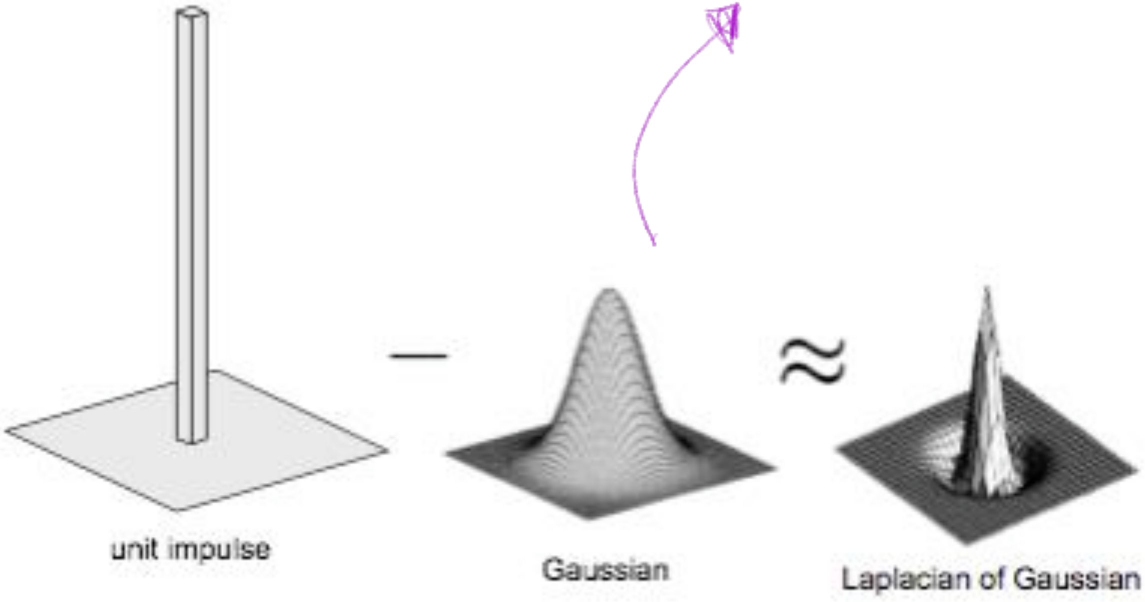
1	2	1
2	4	2
1	2	1

Laplacian of a Gaussian

$$G(\vec{x};\vec{\mu},\Sigma) = \frac{1}{2\pi|\Sigma|^{-1}} e^{-\frac{1}{2}(\vec{x}-\mu)^T\Sigma^{-1}(\vec{x}-\vec{\mu})}$$

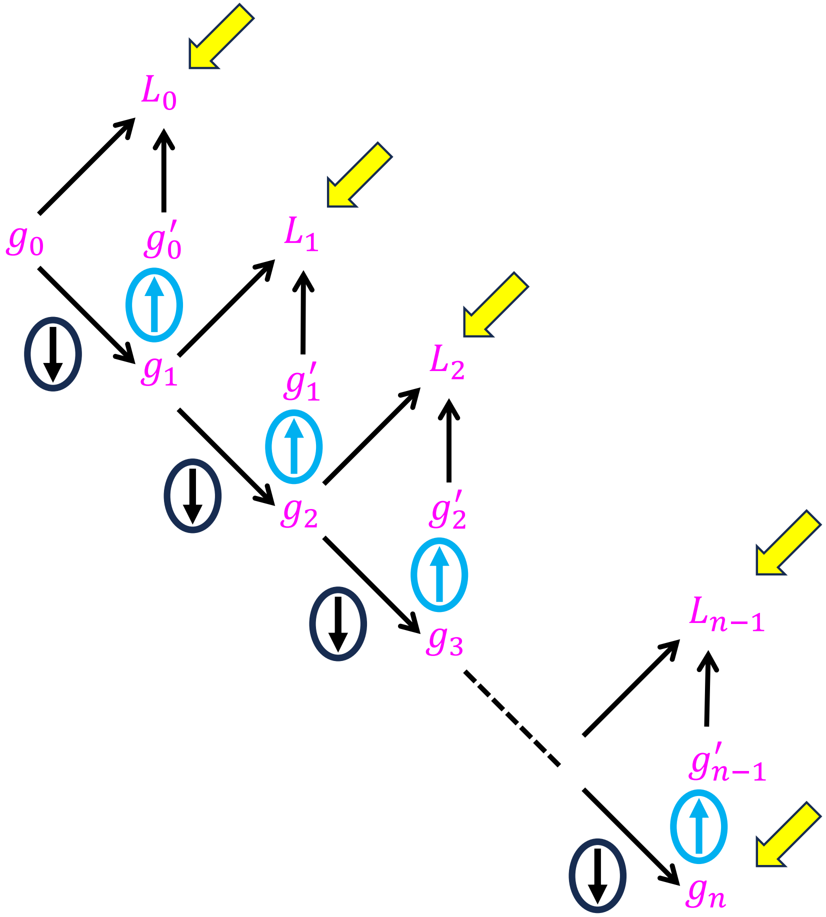
$\vec{x} \in \mathbb{R}^2$

We can approximate the Laplacian of a Gaussian as follows



Source: Lazebnik

Laplacian image pyramids



Pyramid Down

1. $\tilde{g}_i = w * g_i$

2. Downsample $\tilde{g}_i$ by dropping every other row or column

Pyramid Up

1. Upsample g_i by adding 0 between samples to create $\tilde{g}_i$

2. $g'_{i-1} = w * \tilde{g}_i$. This will fill in the 0s added in the previous step

1	0	3	0
0	0	0	0
5	0	7	0
0	0	0	0

1	3
9	11

How to construct a Laplacian Pyramid.

$g_0 = I \in \mathbb{R}^{S \times S}$

$\check{g}_0 = G_\sigma * g_0 \in \mathbb{R}^{S \times S}$

$g_1 = \text{downsample}(\check{g}_0) \in \mathbb{R}^{\frac{S}{2} \times \frac{S}{2}}$

$\hat{g}_0 = \text{upsample}(g_1) \in \mathbb{R}^{S \times S}$

$g_0' = w * \hat{g}_0 \in \mathbb{R}^{S \times S}$

$L_0 = g_0 - g_0' \in \mathbb{R}^{S \times S}$

$$\begin{bmatrix} 2 & 7 & 3 & 3 \\ 6 & 9 & 1 & 4 \\ 4 & 0 & 5 & 5 \\ 1 & 6 & 7 & 6 \end{bmatrix} \quad \begin{bmatrix} 2 & 3 \\ 4 & 5 \end{bmatrix} \quad \begin{bmatrix} 2 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 \\ 4 & 0 & 5 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Gaussian Pyramid.

First level of Laplacian Pyramid

L_0, g_1

$\check{g}_1 = G_\sigma * g_1$

$g_2 = \text{downsample}(\check{g}_1)$

$\hat{g}_1 = \text{upsample}(g_2)$

$g_1' = w * \hat{g}_1$

$L_1 = g_1 - g_1'$

$I \to L_0, L_1, \ldots, L_{N-1}, g_N$

Laplacian image pyramid

Given image $I = g_0$

- Convolve g_i with low-pass filter w and down-sample by half to construct g_{i+1}
  - Downsample by discarding every other row and column
- Upsample g_{i+1} by double by inserting 0s and and interpolating the missing values by convolving it with w and create g'_i
- Compute $L_i = g_i - g'_i$
- Repeat