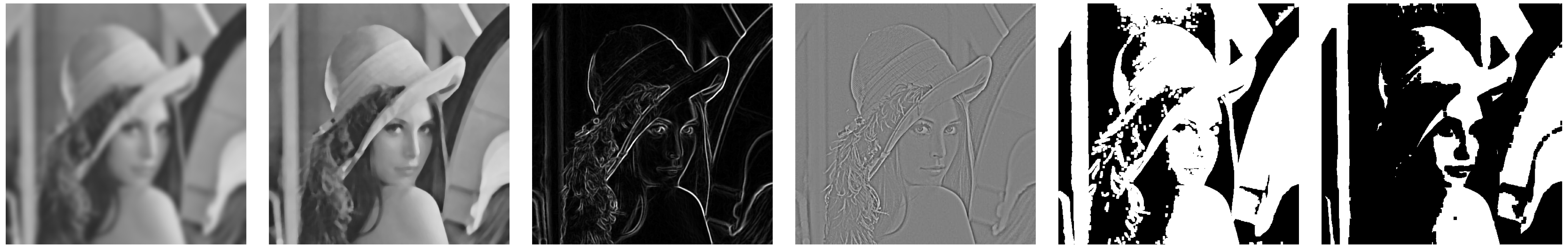
THE UNIVERSITY OF BRITISH COLUMBIA

CPSC 425: Computer Vision



Lecture 3: Image Filtering

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

Lecture 3: Goal

Start to develop tools for (simple) processing of images

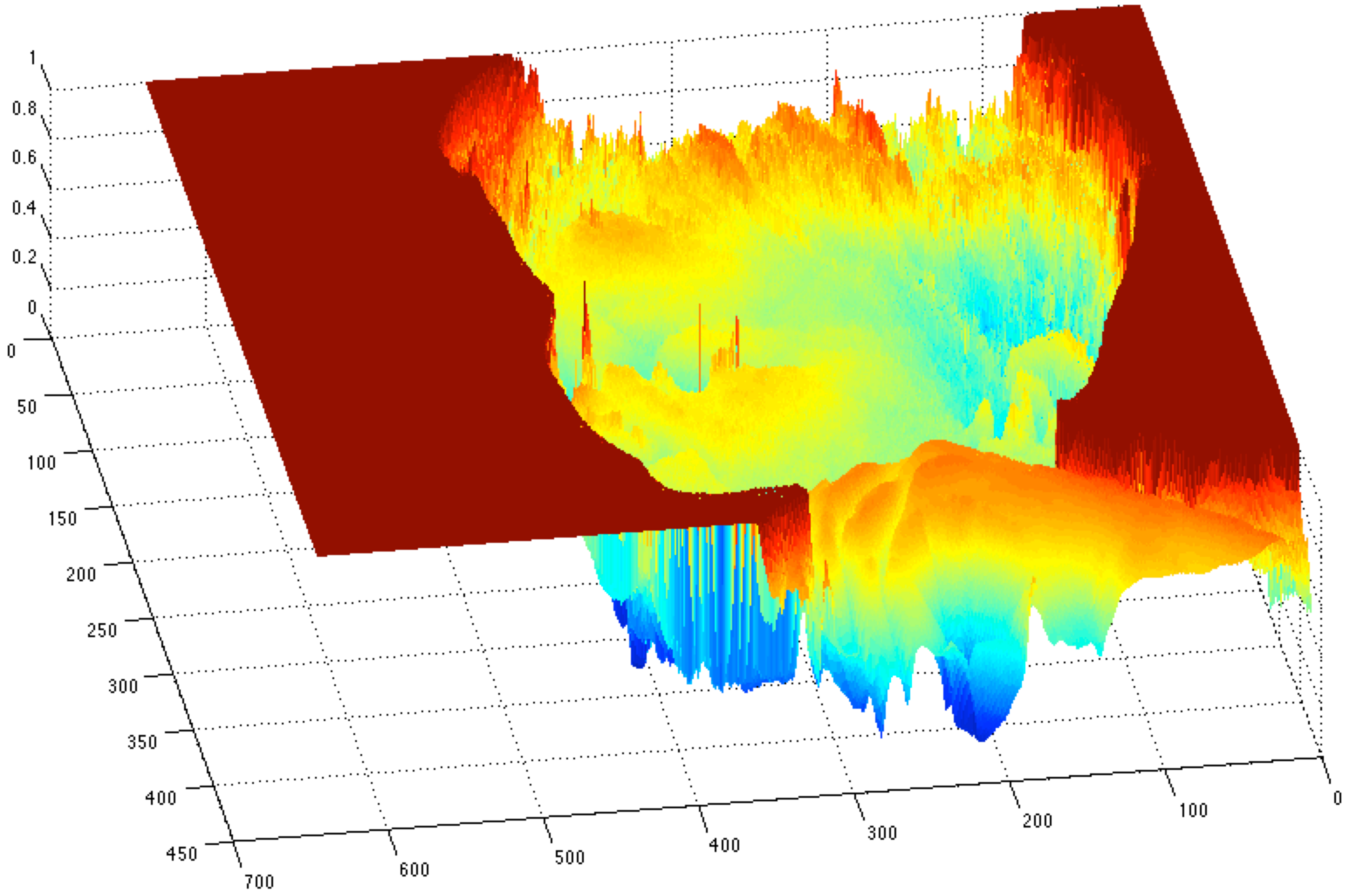
(the “tools” we going to learn over the next few lectures will be broadly useful, including in CNNs)

Image as a **2D Function**

A (grayscale) image is a 2D function

grayscale image

$I(X,Y)$



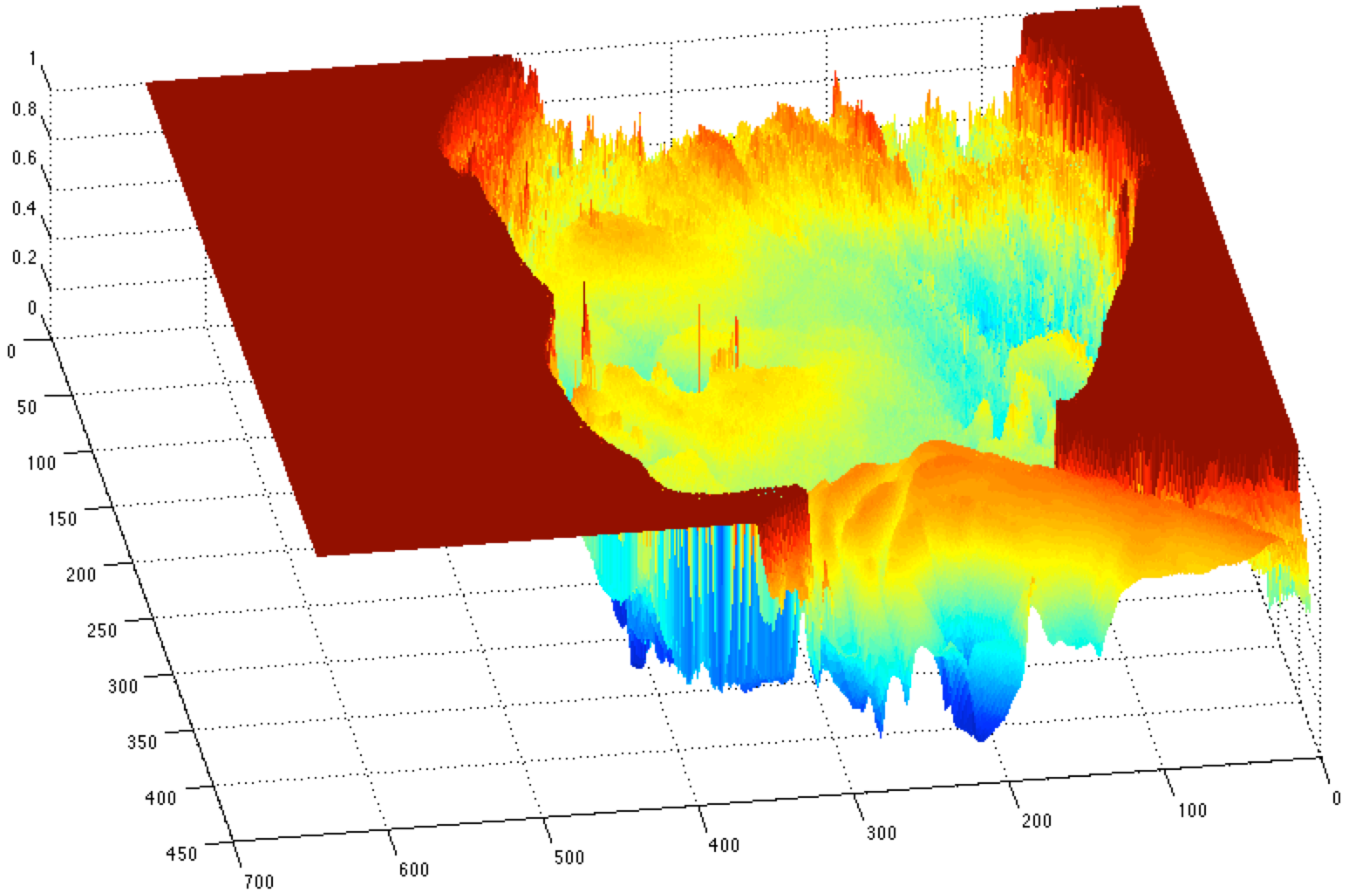
Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

Image as a **2D Function**

A (grayscale) image is a 2D function

grayscale image

$I(X, Y)$



domain: $(X, Y) \in ([1, width], [1, hight])$

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

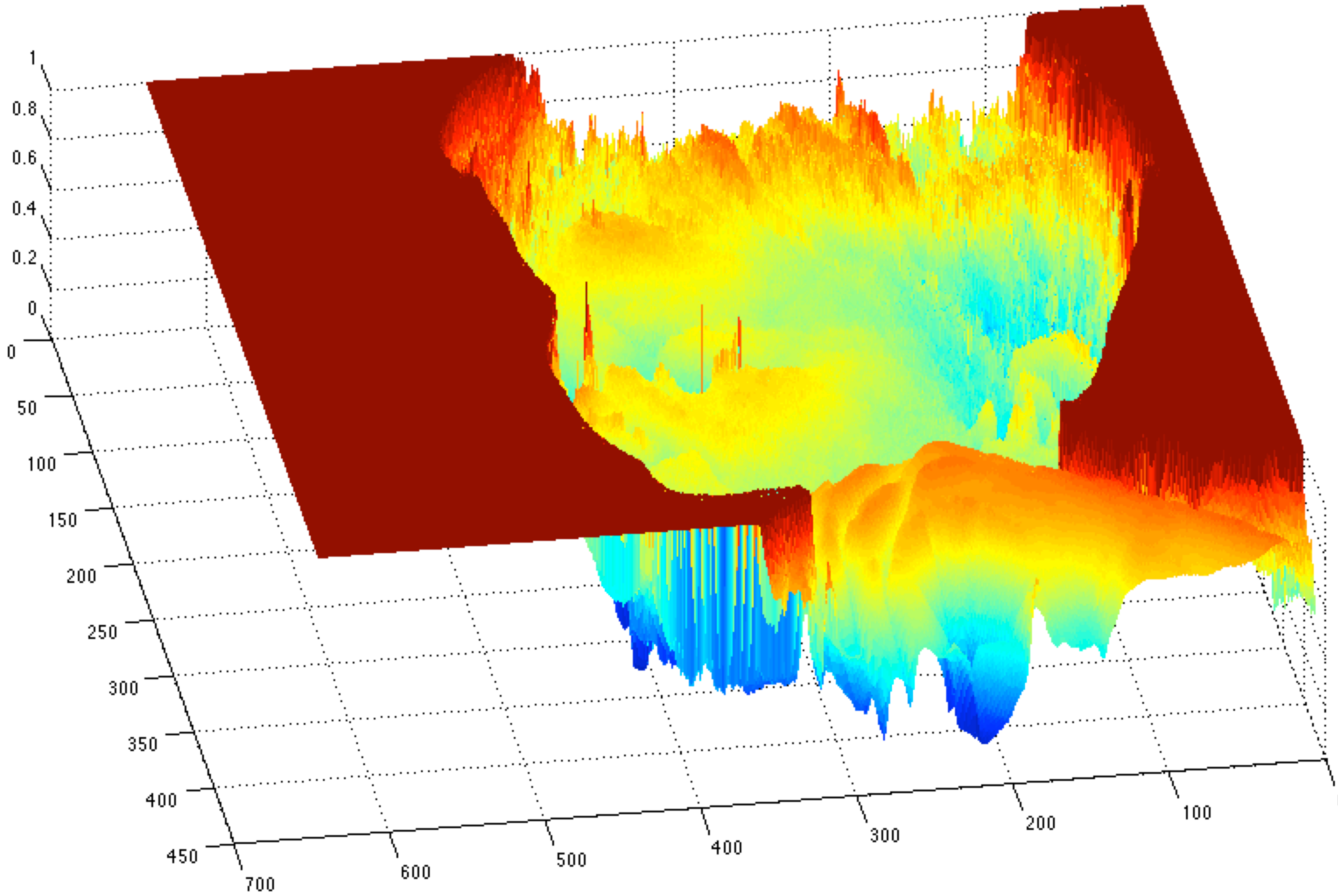
Image as a **2D Function**

A (grayscale) image is a 2D function

grayscale image

What is the **range** of the image function?

$I(X, Y)$



domain: $(X, Y) \in ([1, width], [1, hight])$

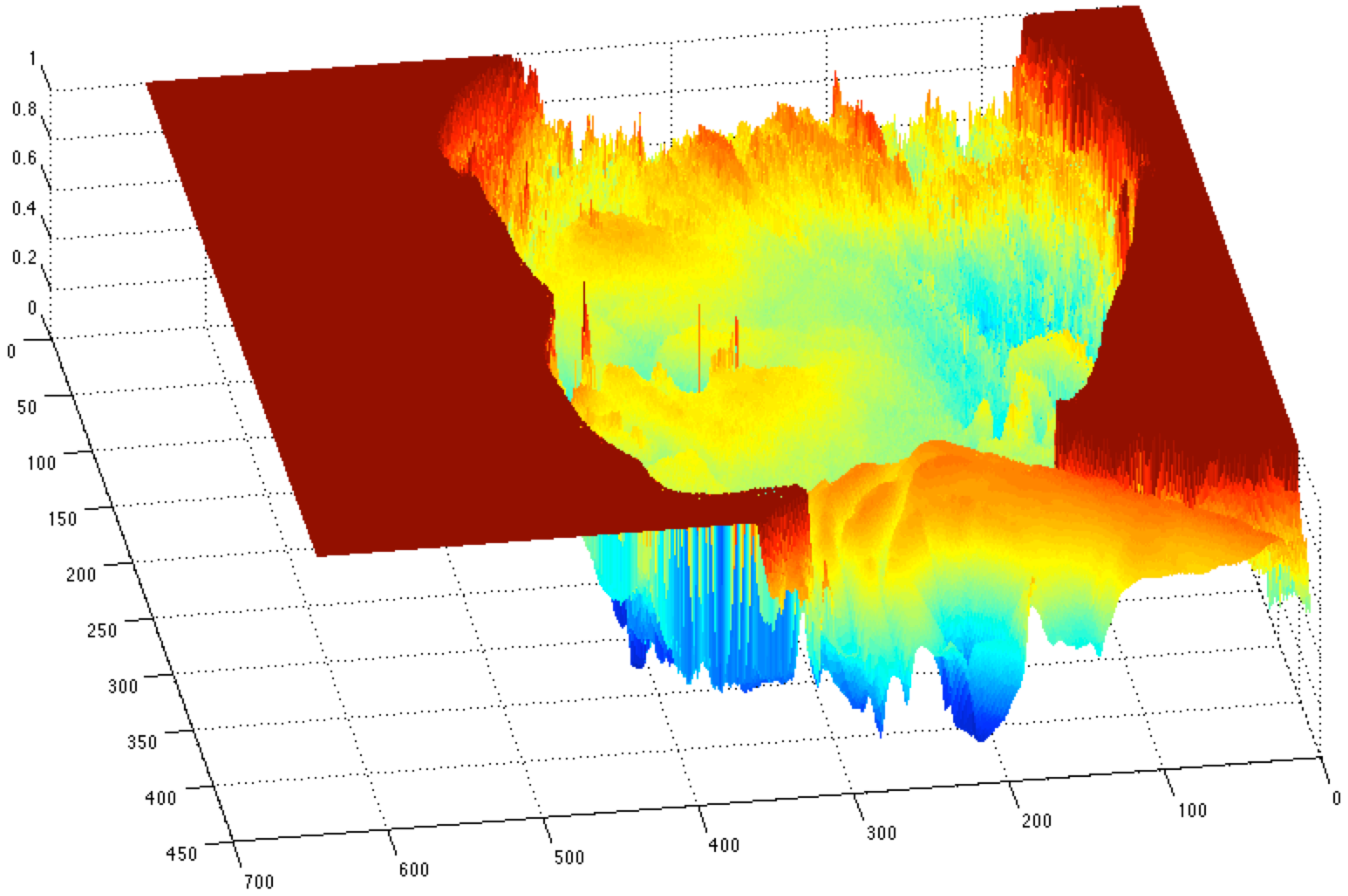
Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

Image as a **2D Function**

A (grayscale) image is a 2D function

grayscale image

$I(X,Y)$



What is the **range** of the image function?

$$I(X,Y) \in [0,255] \in \mathbb{Z}$$

domain: $(X,Y) \in ([1, width], [1, hight])$

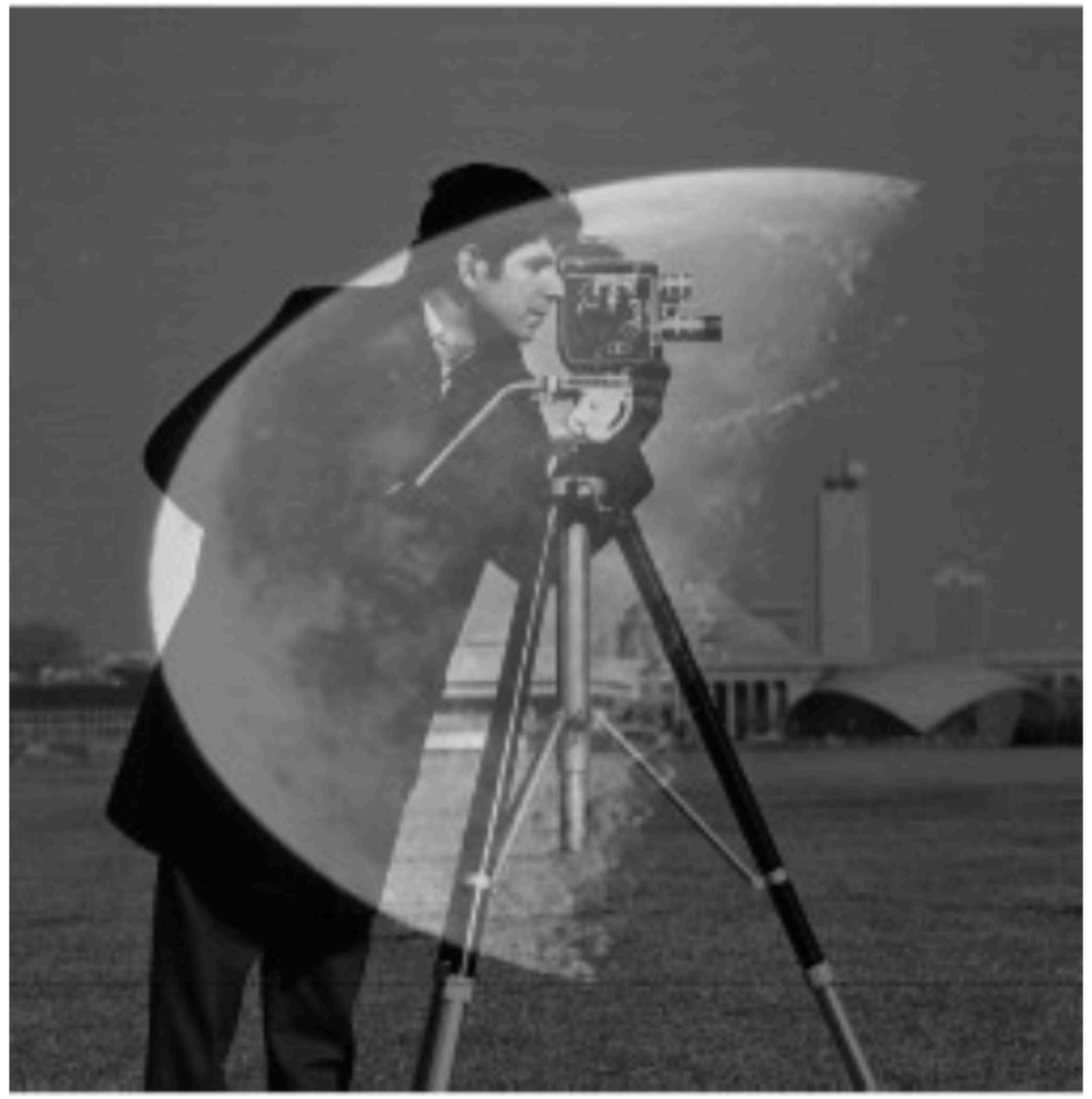
Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

Adding two Images

Since images are functions, we can perform operations on them, e.g., **average**

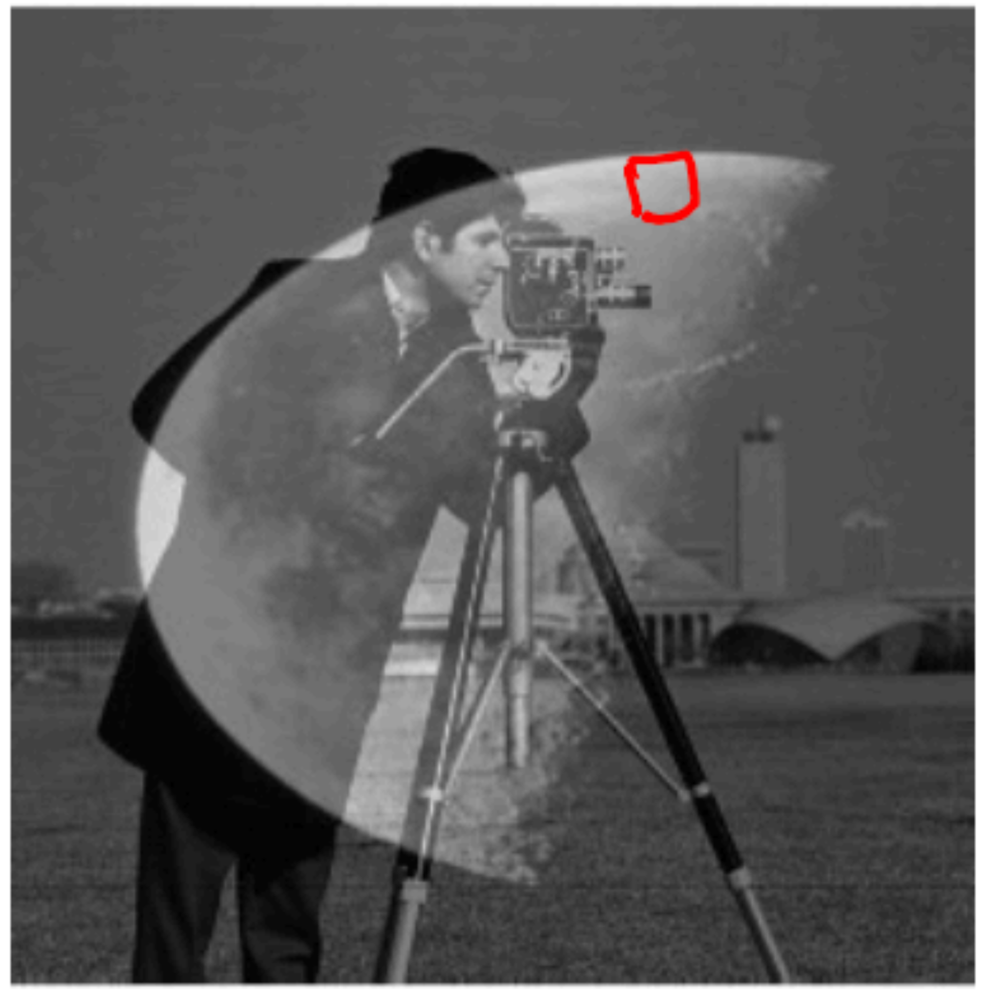
$I(X, Y)$

$G(X, Y)$

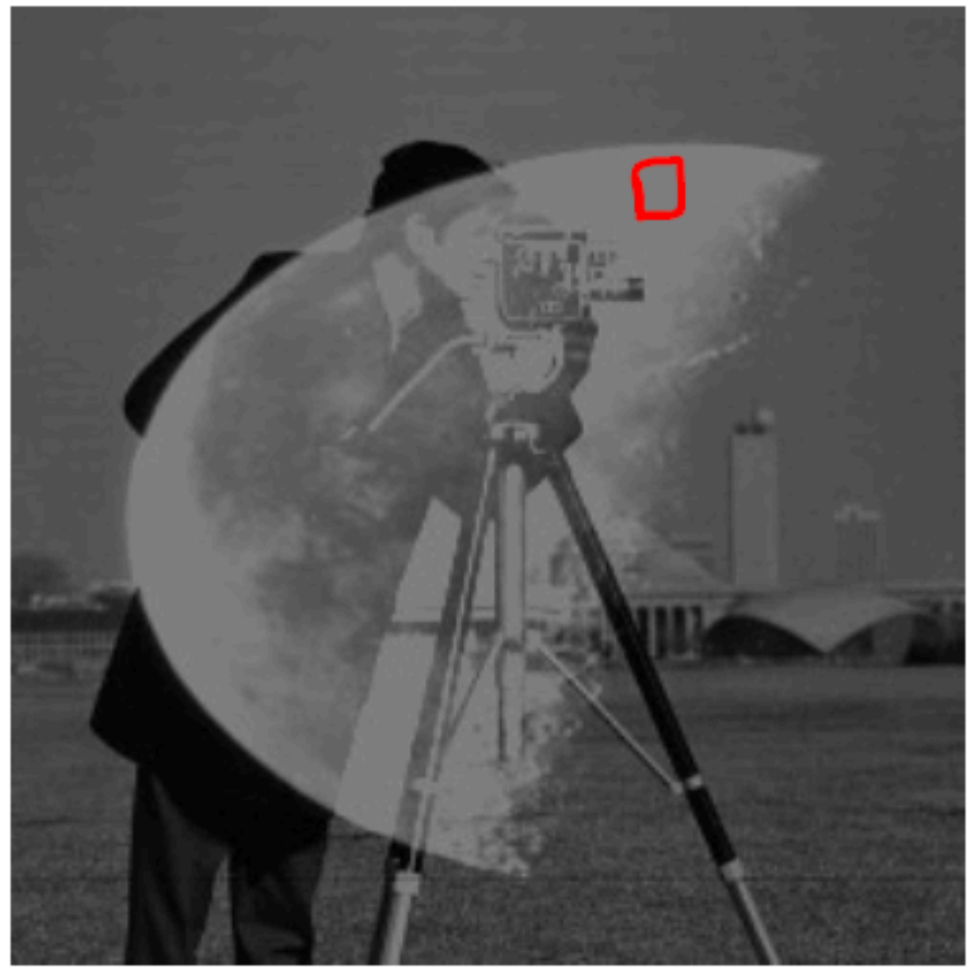


$$\frac{I(X, Y)}{2} + \frac{G(X, Y)}{2}$$

Adding two Images

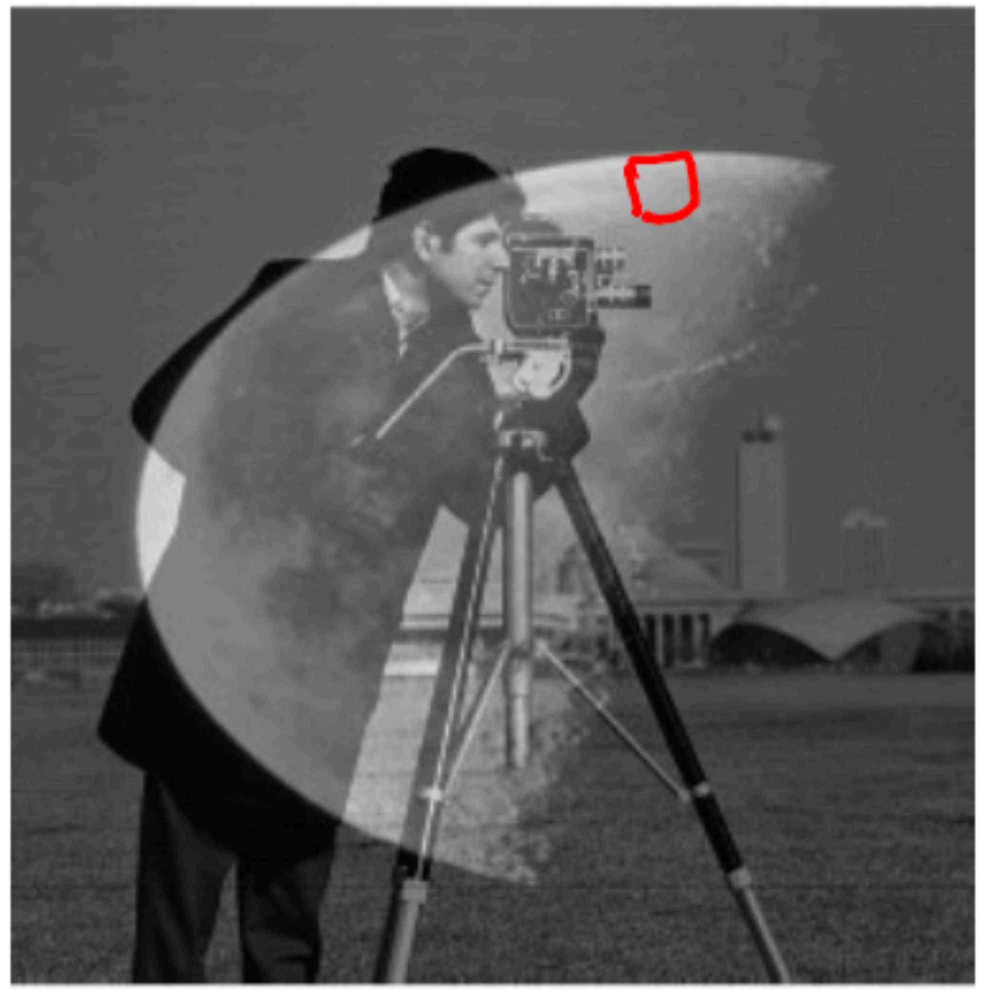


$$a = \frac{I(X,Y)}{2} + \frac{G(X,Y)}{2}$$

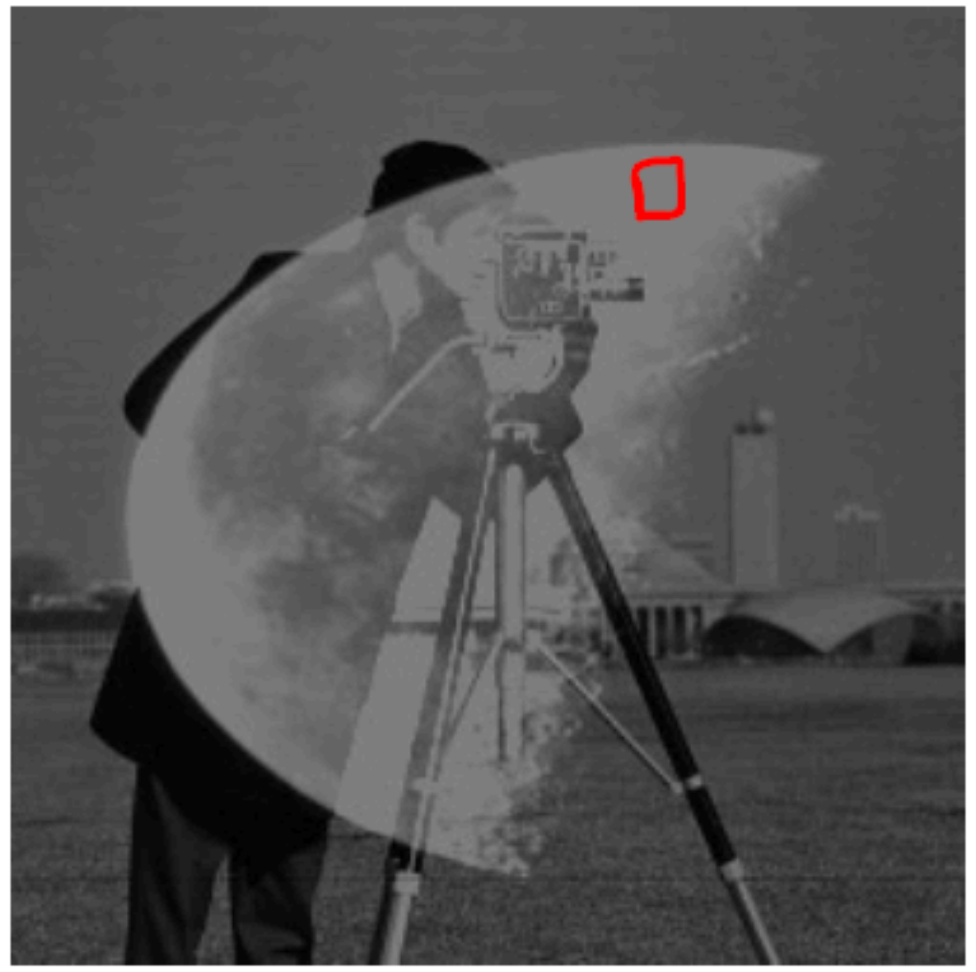


$$b = \frac{I(X,Y) + G(X,Y)}{2}$$

Adding two Images



$$a = \frac{I(X,Y)}{2} + \frac{G(X,Y)}{2}$$



$$b = \frac{I(X,Y) + G(X,Y)}{2}$$

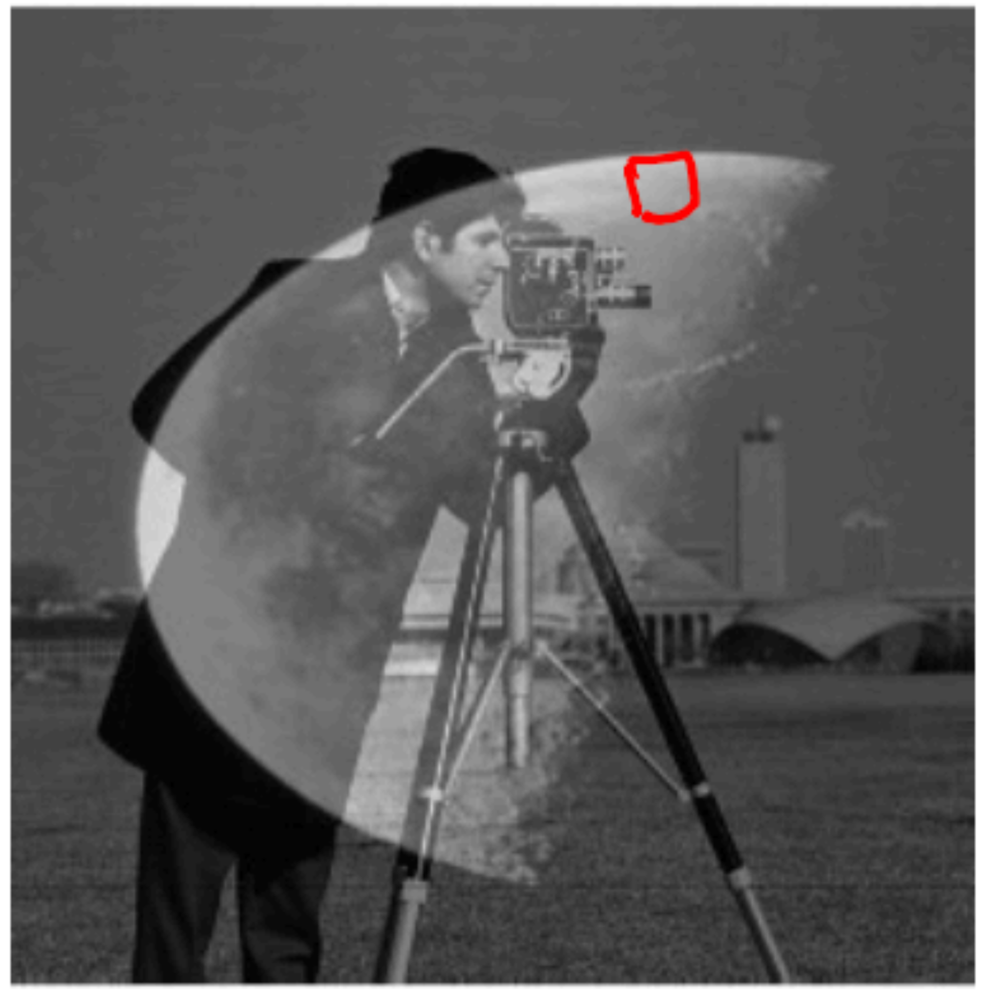
Question:

$$a = b$$

$$a > b$$

$$a < b$$

Adding two Images



Red pixel in camera man image = 98

Red pixel in moon image = 200

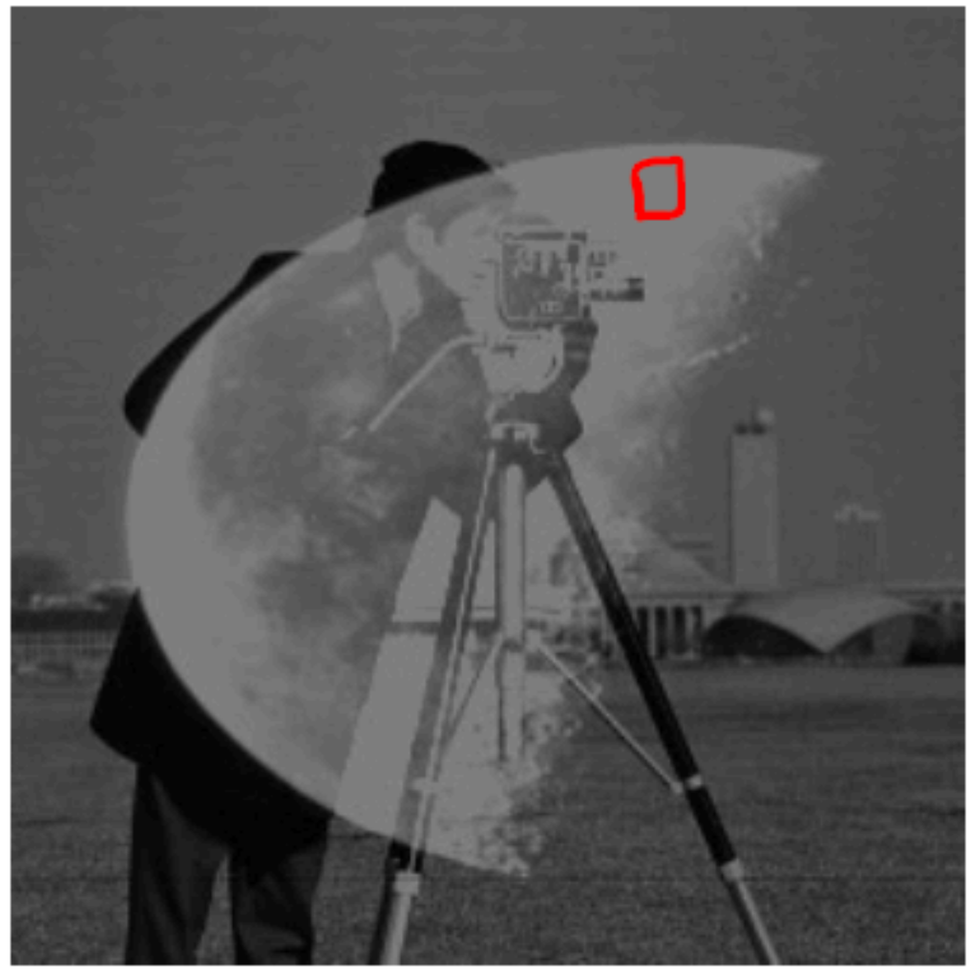
$$\frac{98}{2}+\frac{200}{2}=49+100=149$$

Question:

$$a = b$$

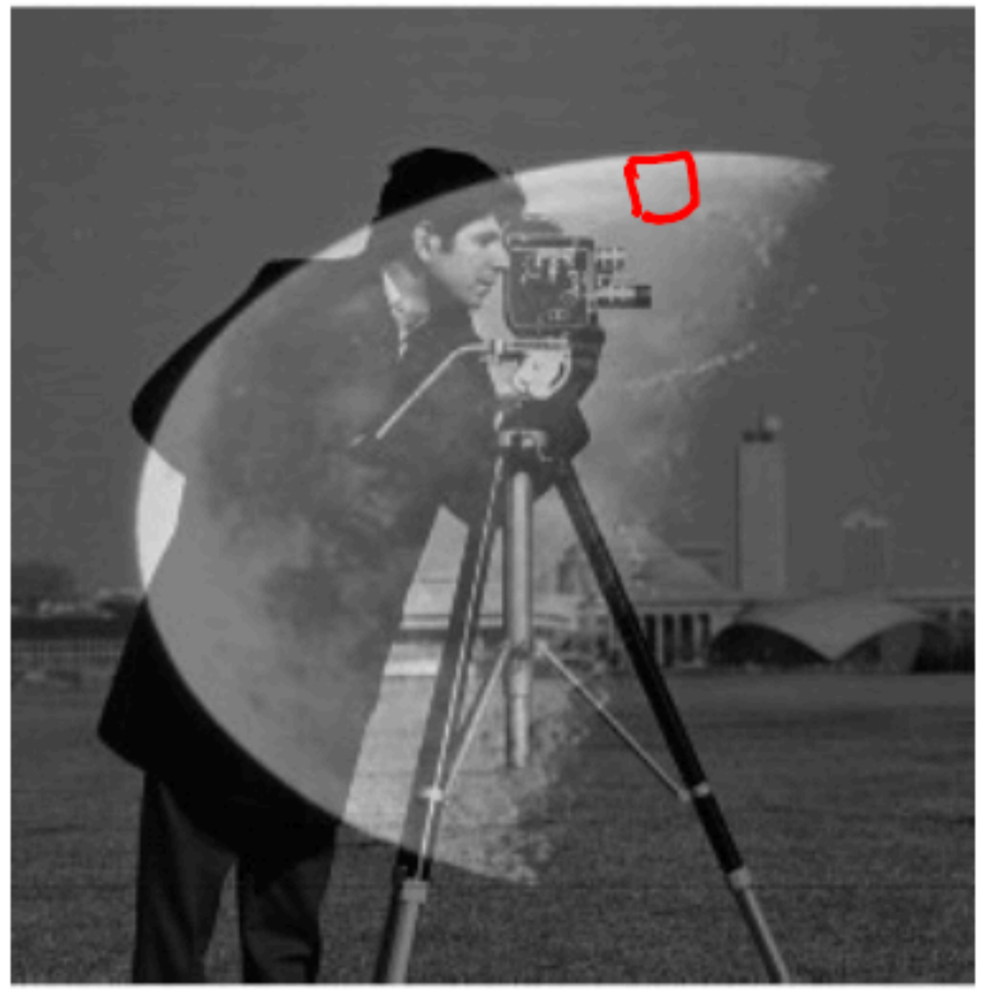
$$\boxed{a > b}$$

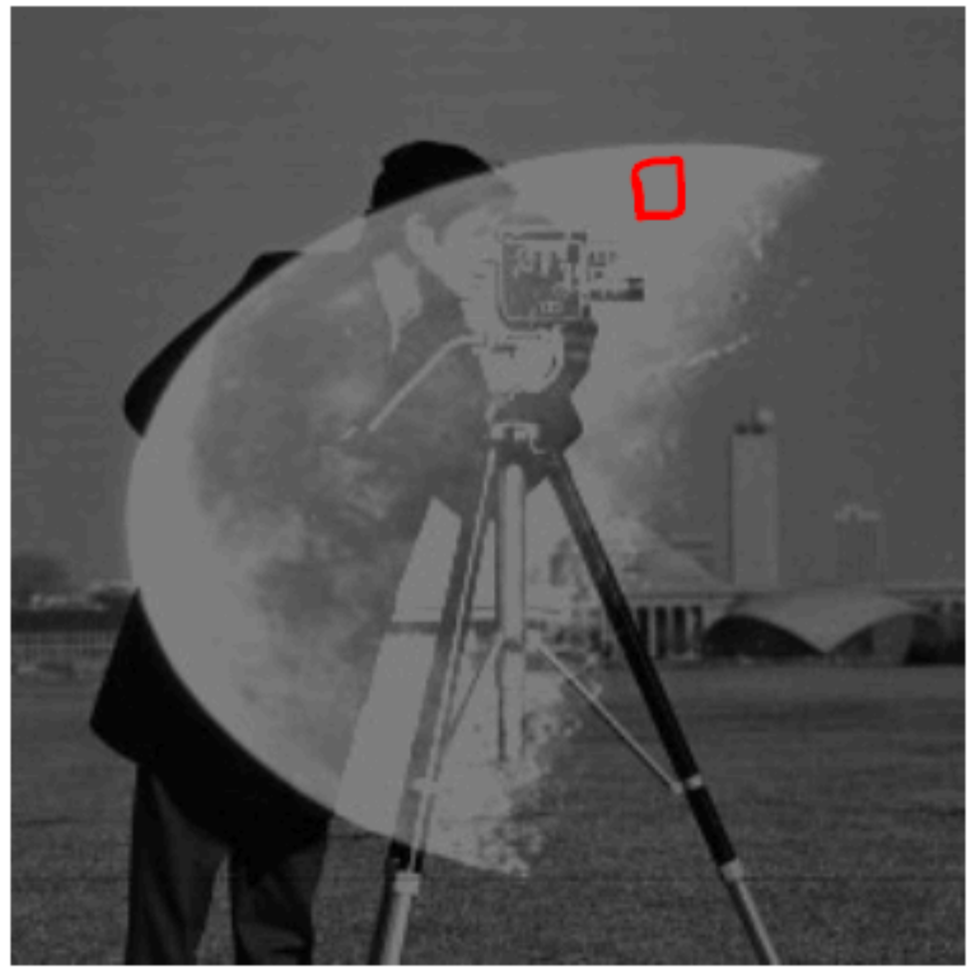
$$a < b$$



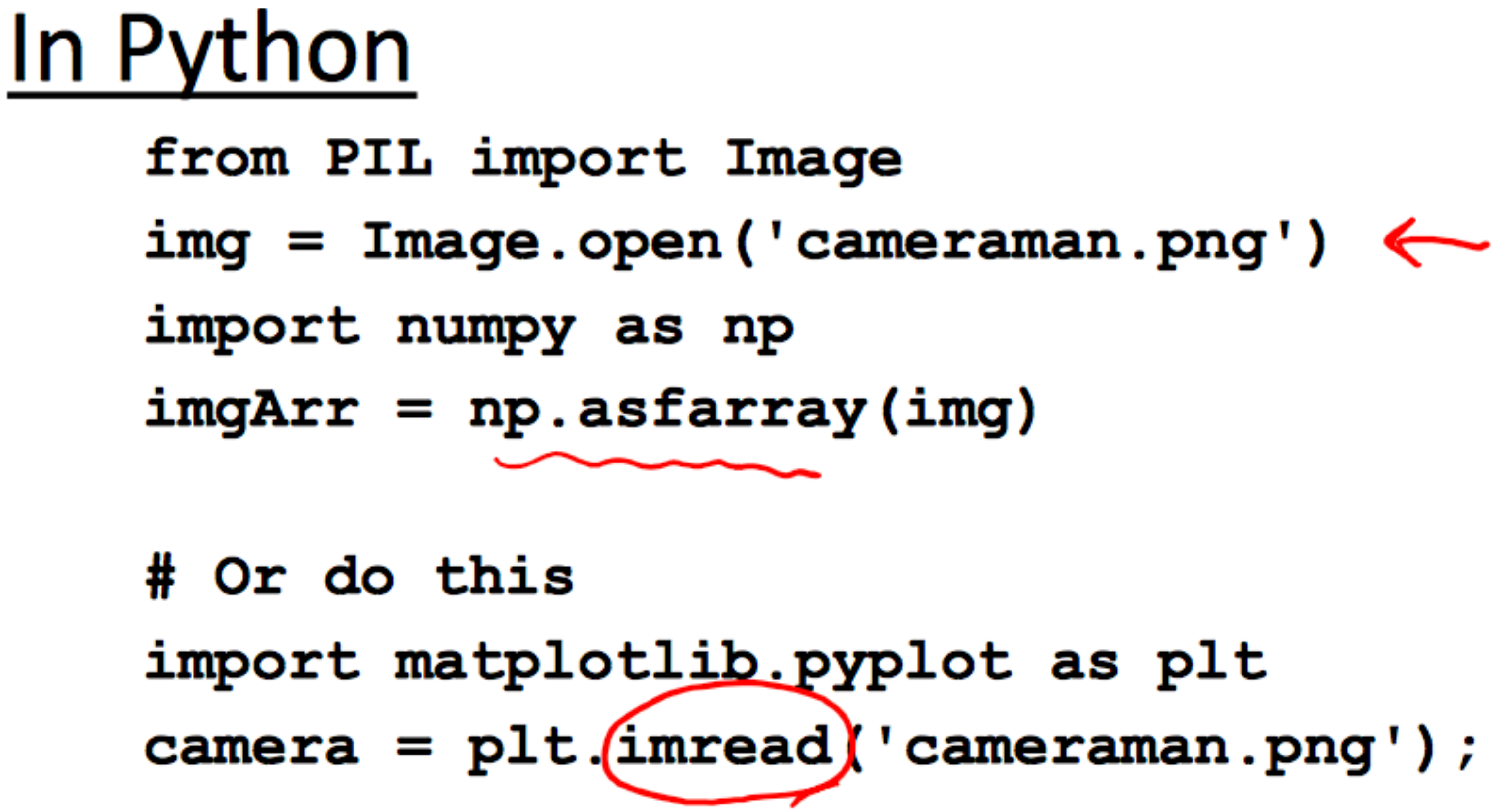
$$\frac{98+200}{2}=\frac{\lfloor 298 \rfloor}{2}=\frac{255}{2}=127$$

Adding two Images





It is often convenient to convert images to **doubles** when doing processing

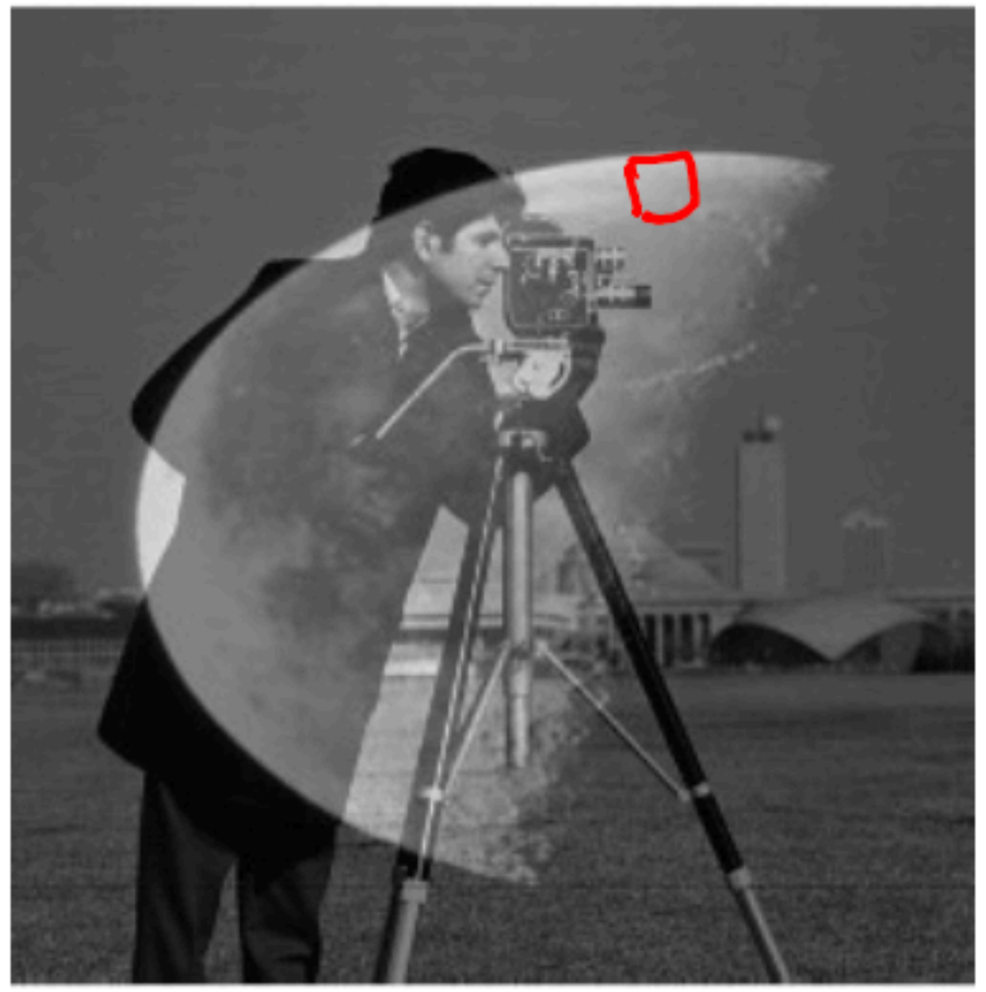


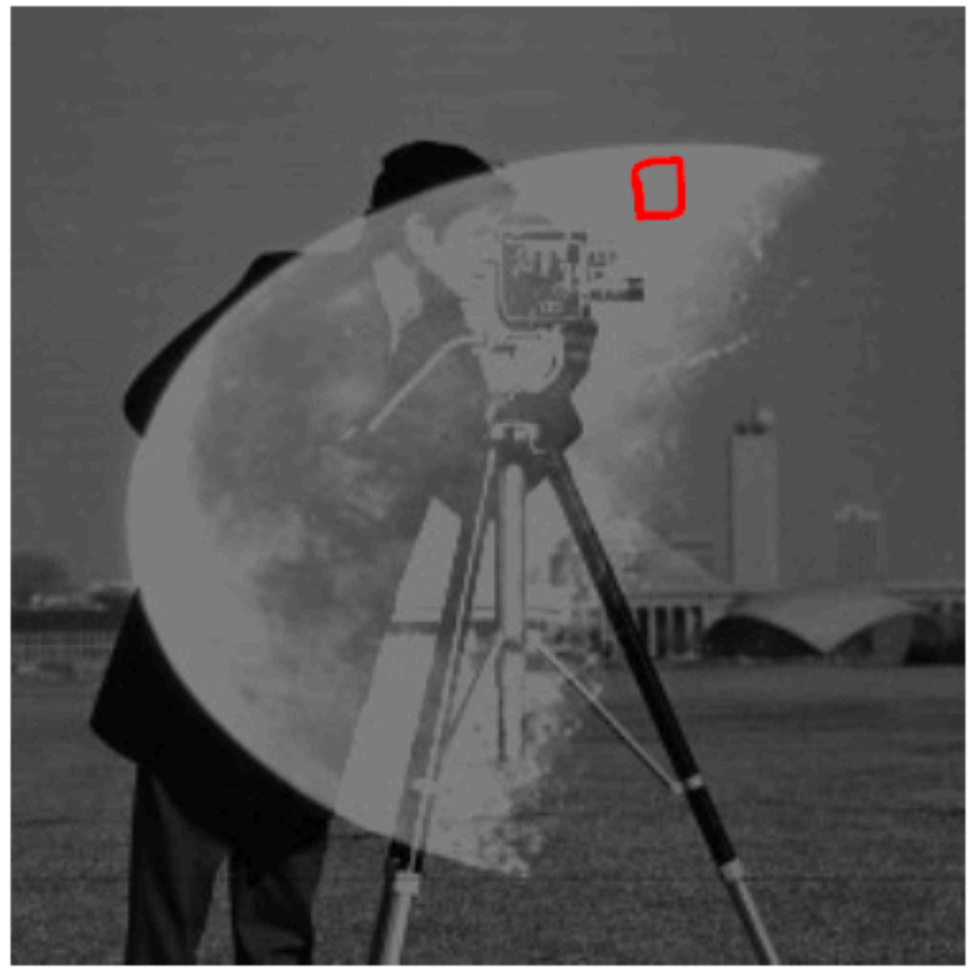
In Python

```
from PIL import Image
img = Image.open('cameraman.png')
import numpy as np
imgArr = np.asfarray(img)

# Or do this
import matplotlib.pyplot as plt
camera = plt.imread('cameraman.png');
```

Adding two Images

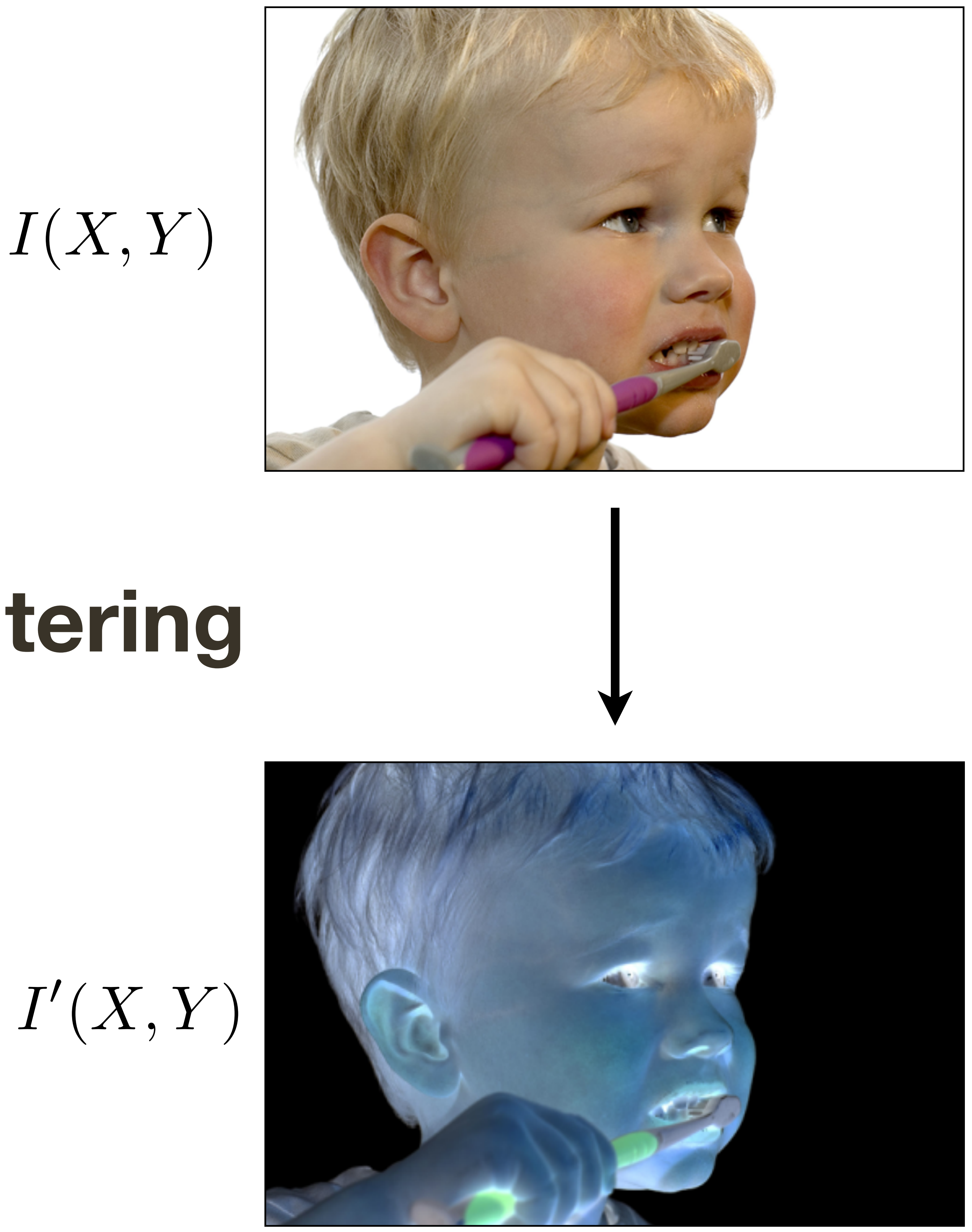




This will save you a **LOT** of headache in homeworks:

1. Convert to **doubles**
2. (optionally) Normalize image to [0,1] range (by dividing by 255)
3. Perform any **computations** needed
4. (optionally) Undo normalization (by multiplying by 255)
5. **Clamp** values between [0, 255]
6. Convert to **uint8**

What types of **transformations** can we do?

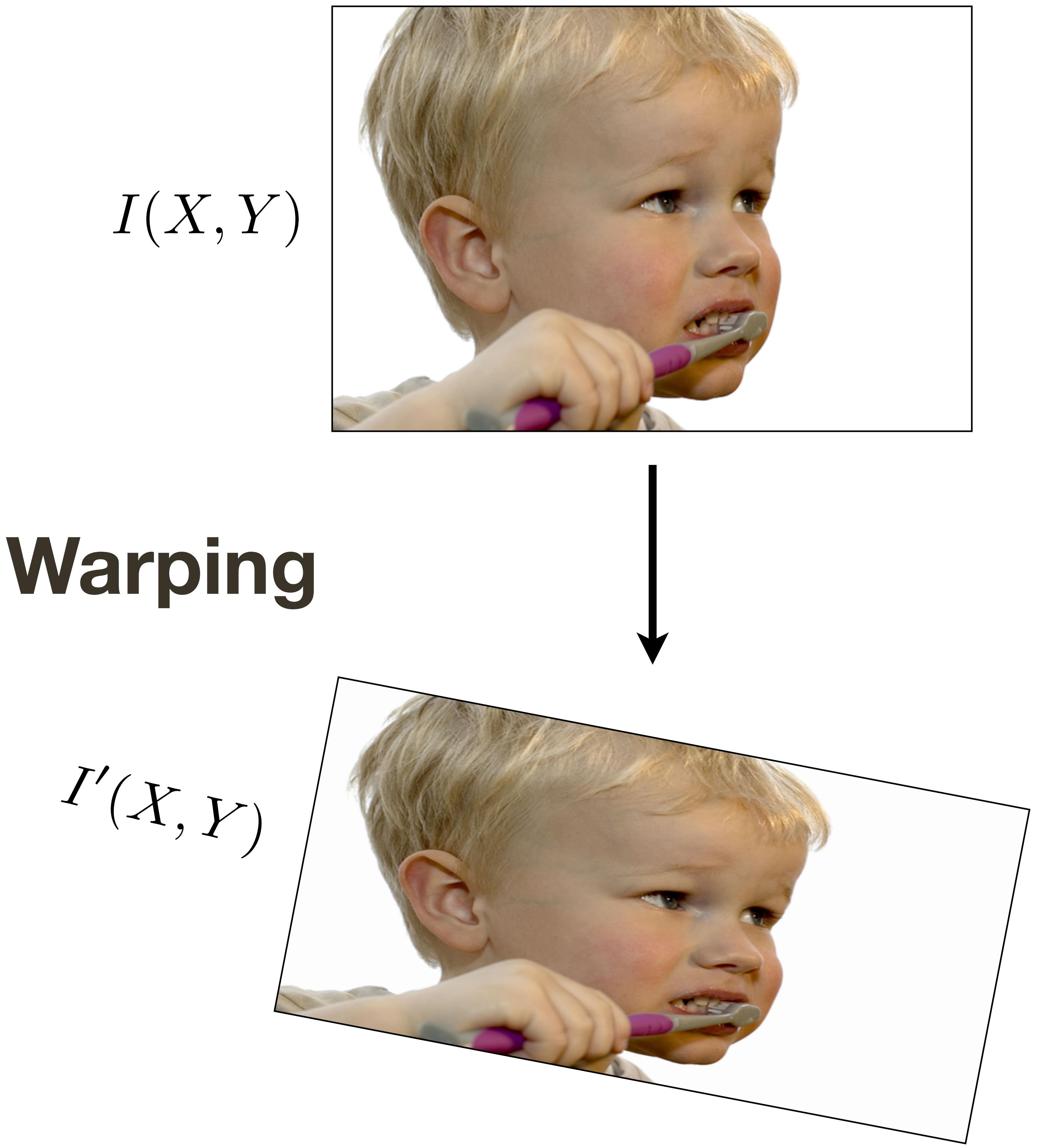


$I(X,Y)$

Filtering

$I'(X,Y)$

changes range of image function



$I(X,Y)$

Warping

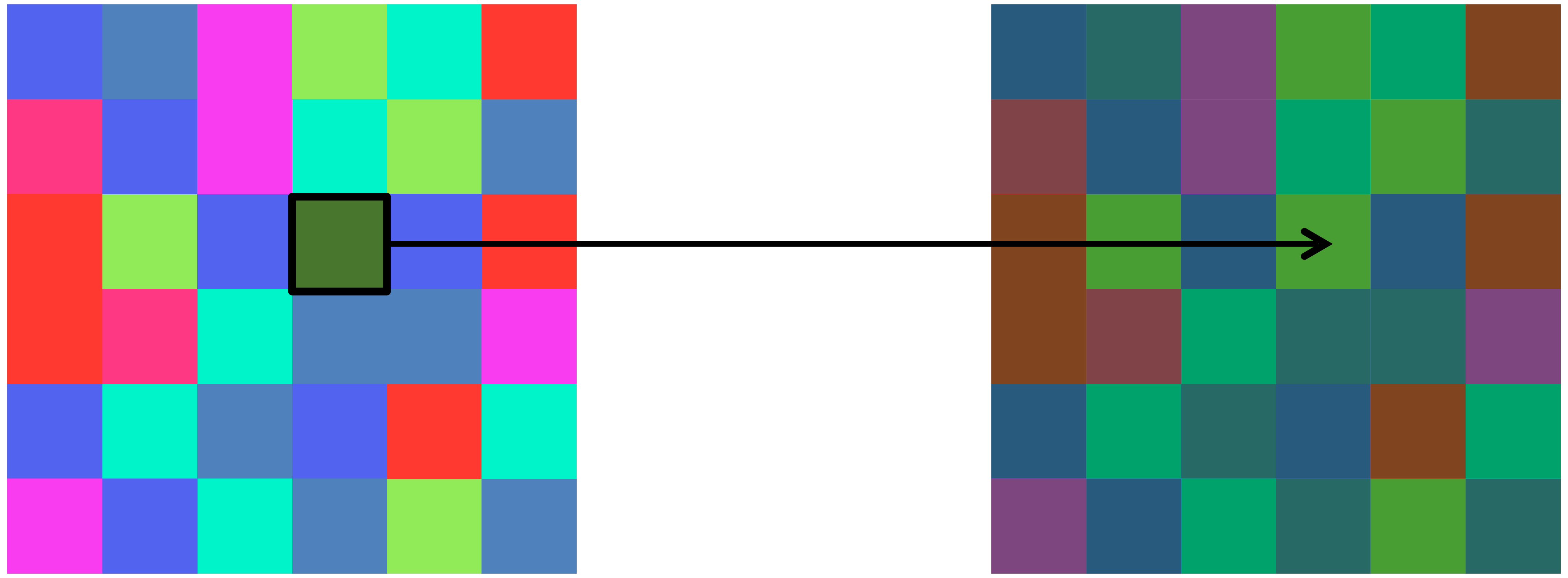
$I'(X,Y)$

changes domain of image function

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

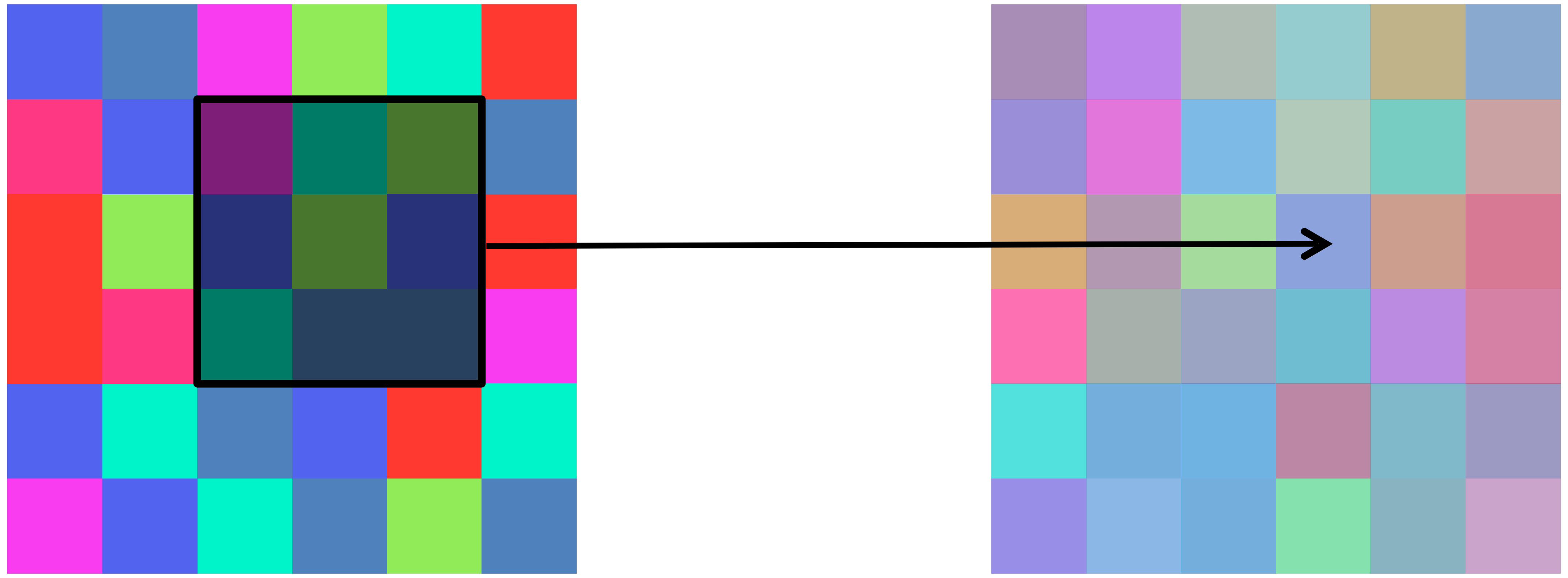
What types of **filtering** can we do?

Point Operation



point processing

Neighborhood Operation

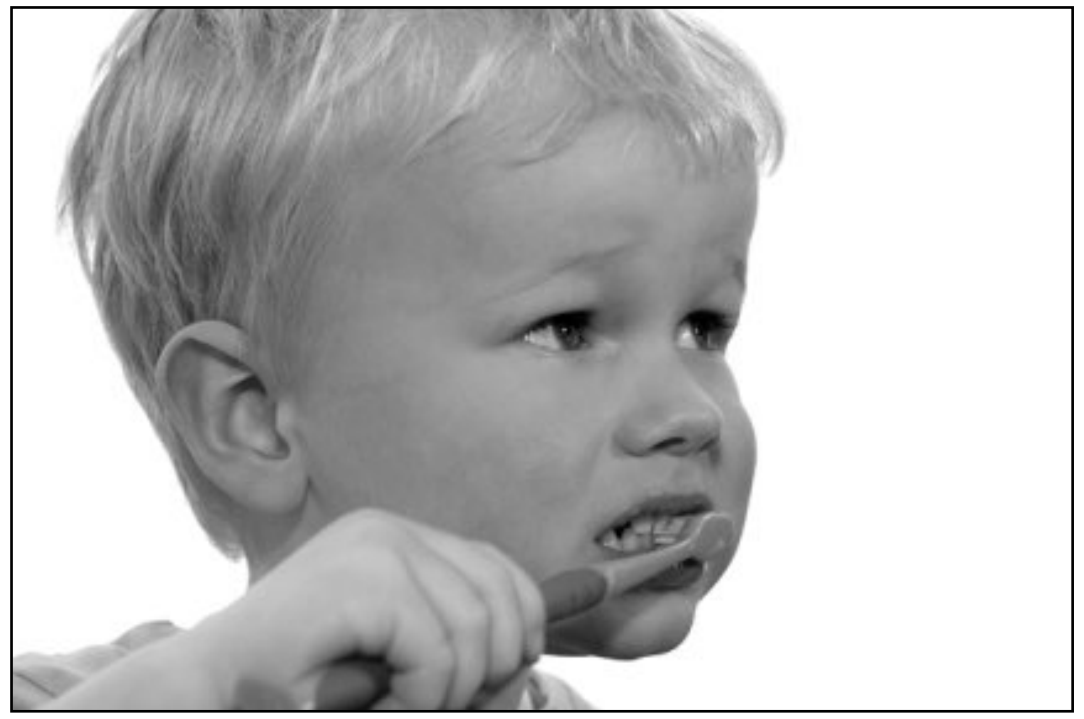


“filtering”

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

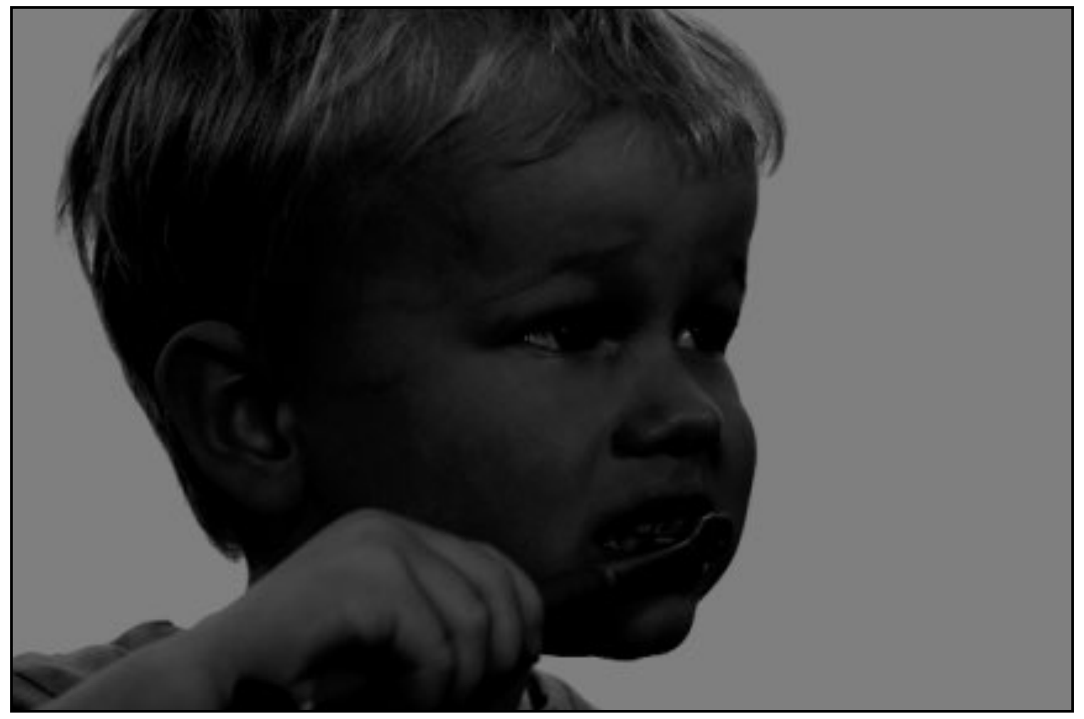
Examples of **Point Processing**

original

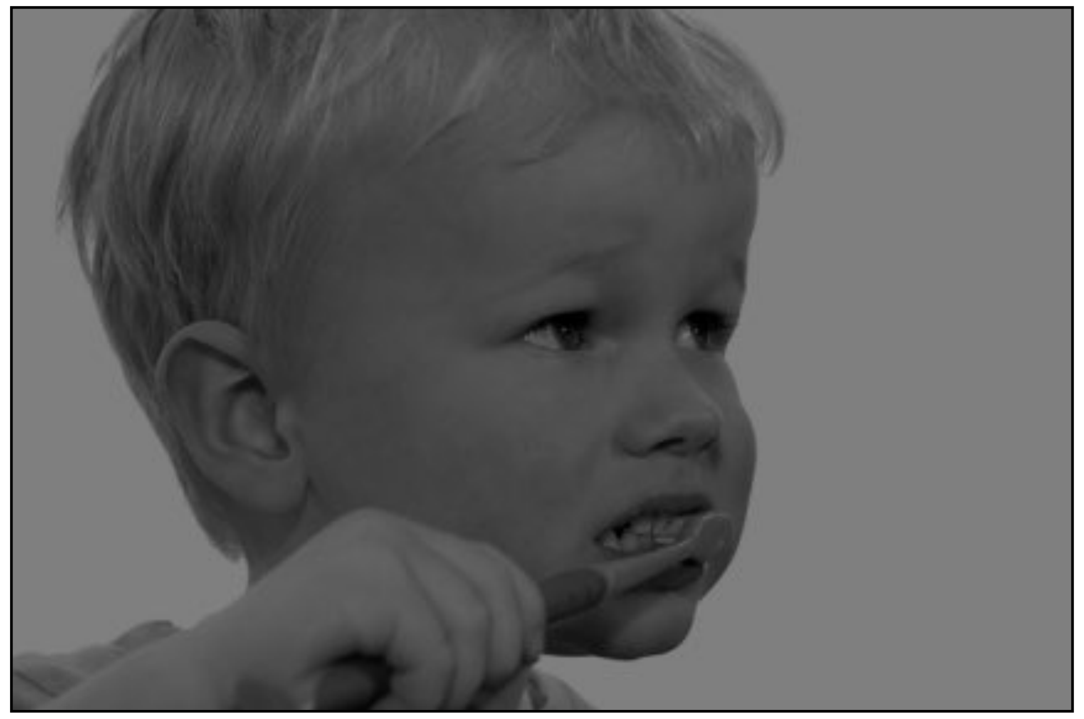


$I(X,Y)$

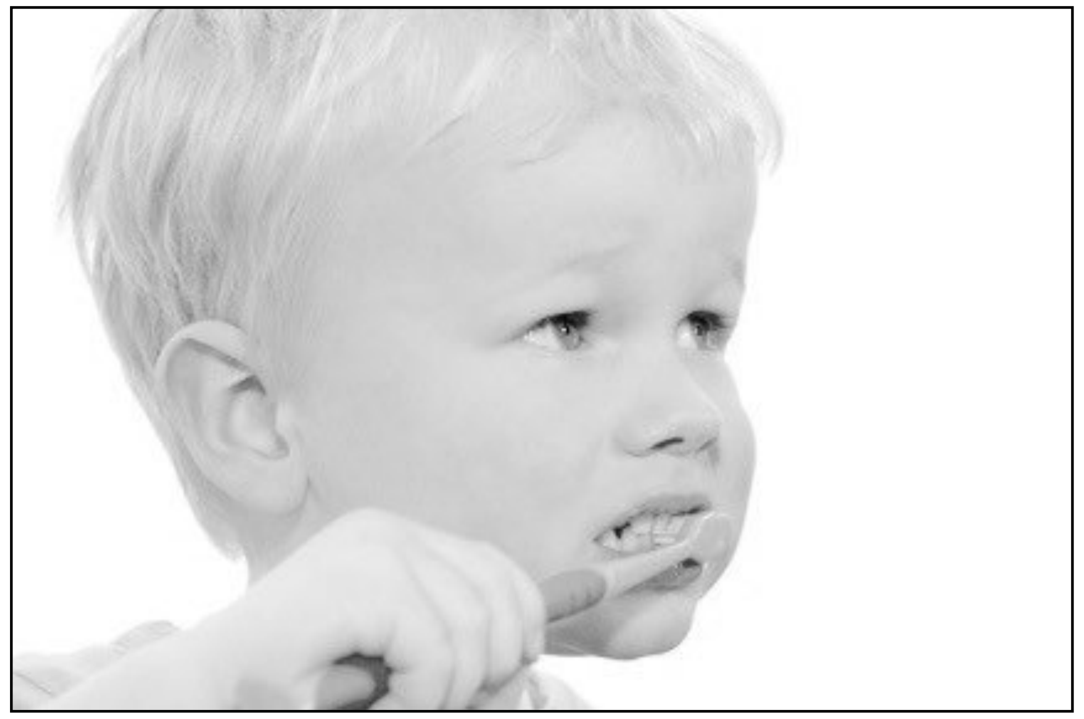
darken



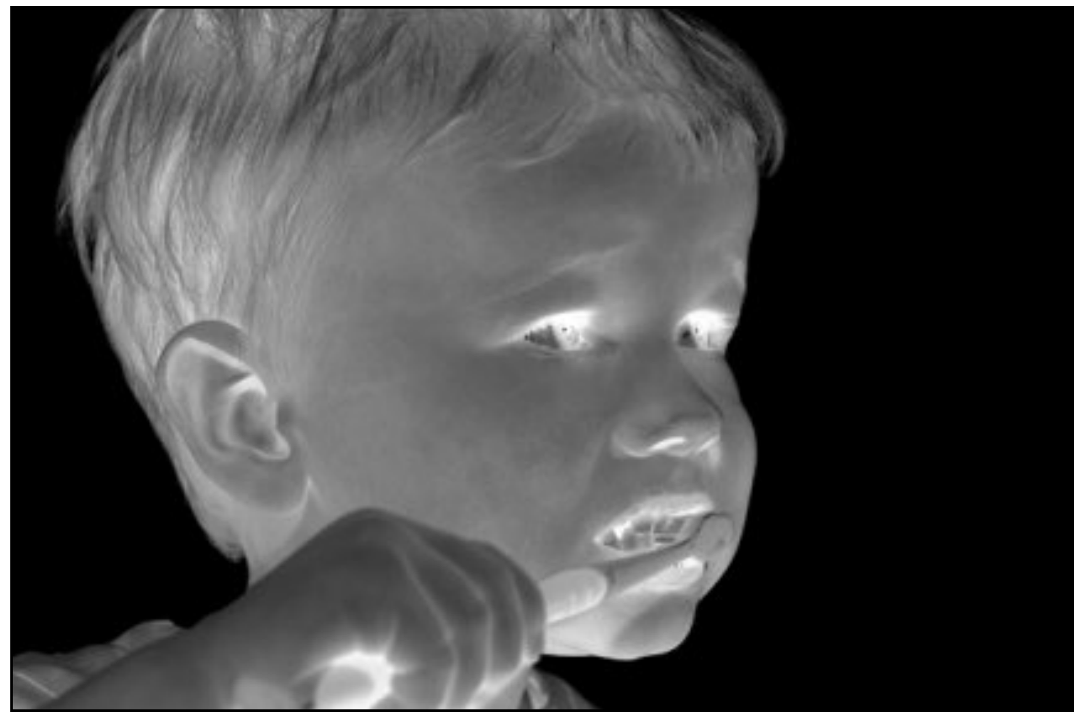
lower contrast



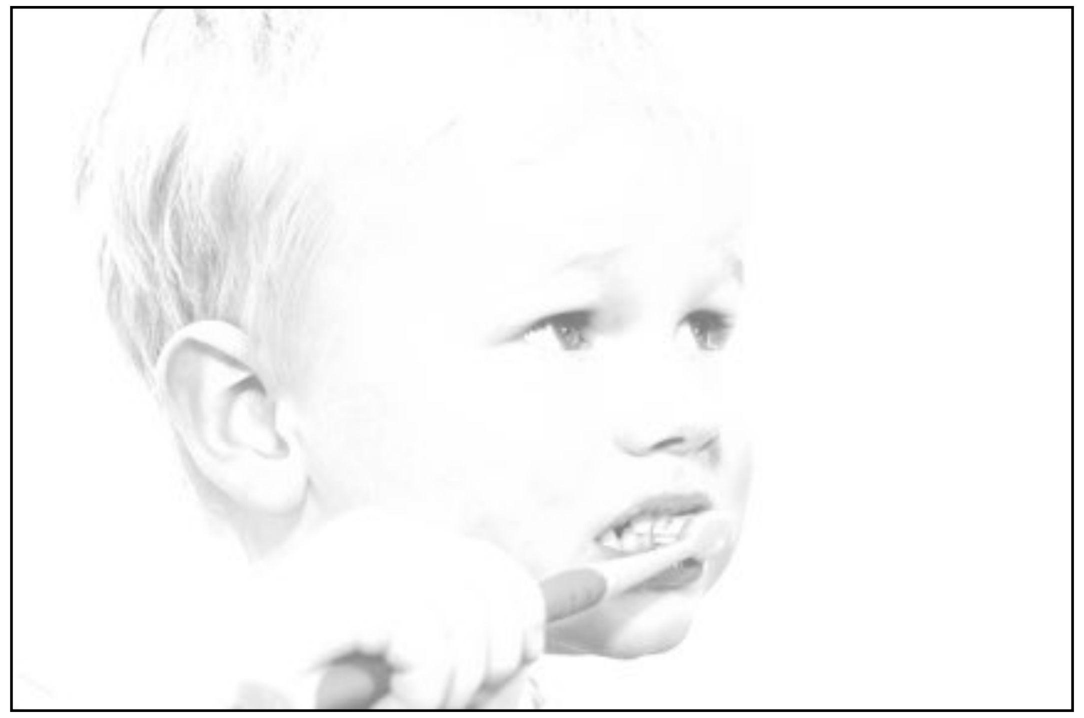
non-linear lower contrast



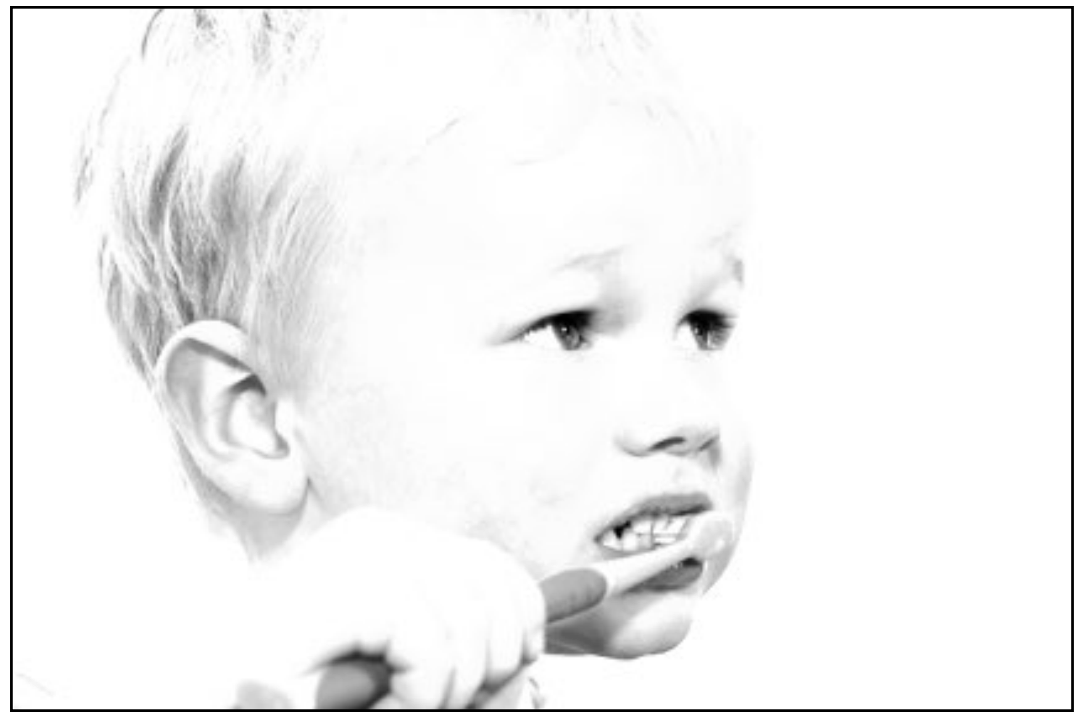
invert



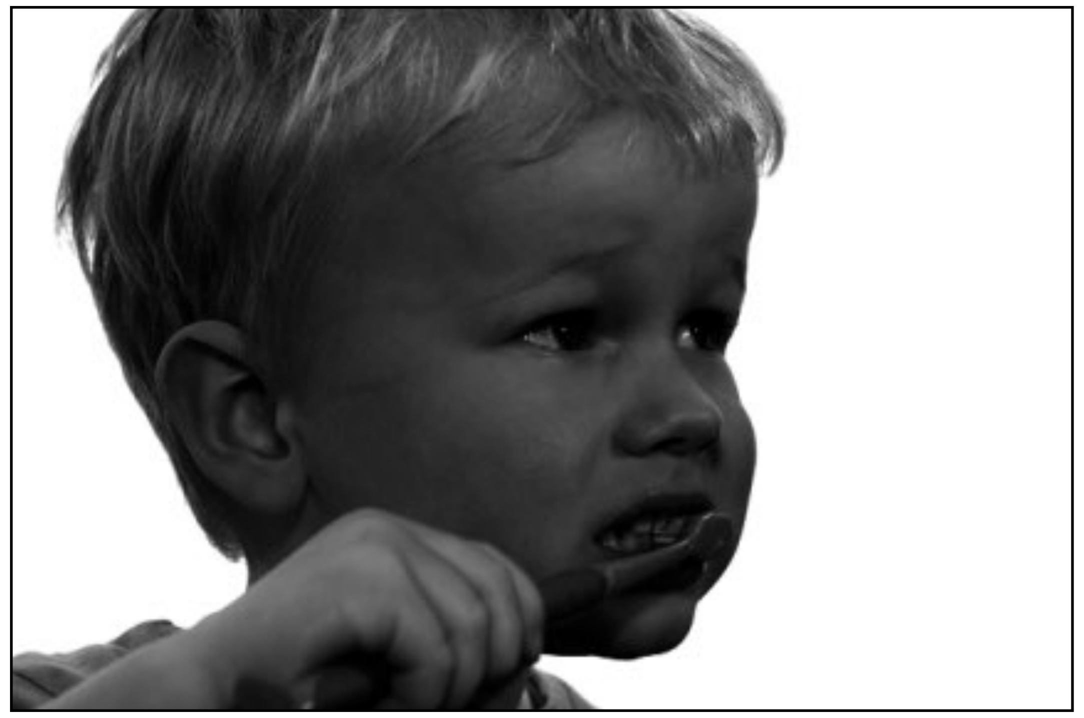
lighten



raise contrast



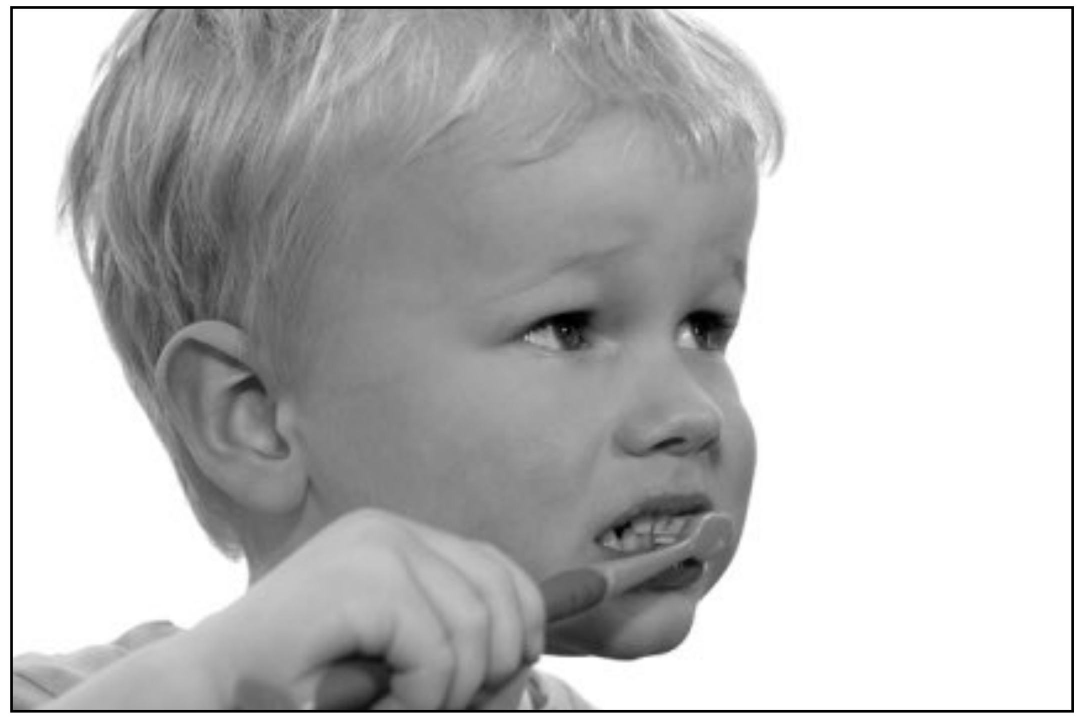
non-linear raise contrast



Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

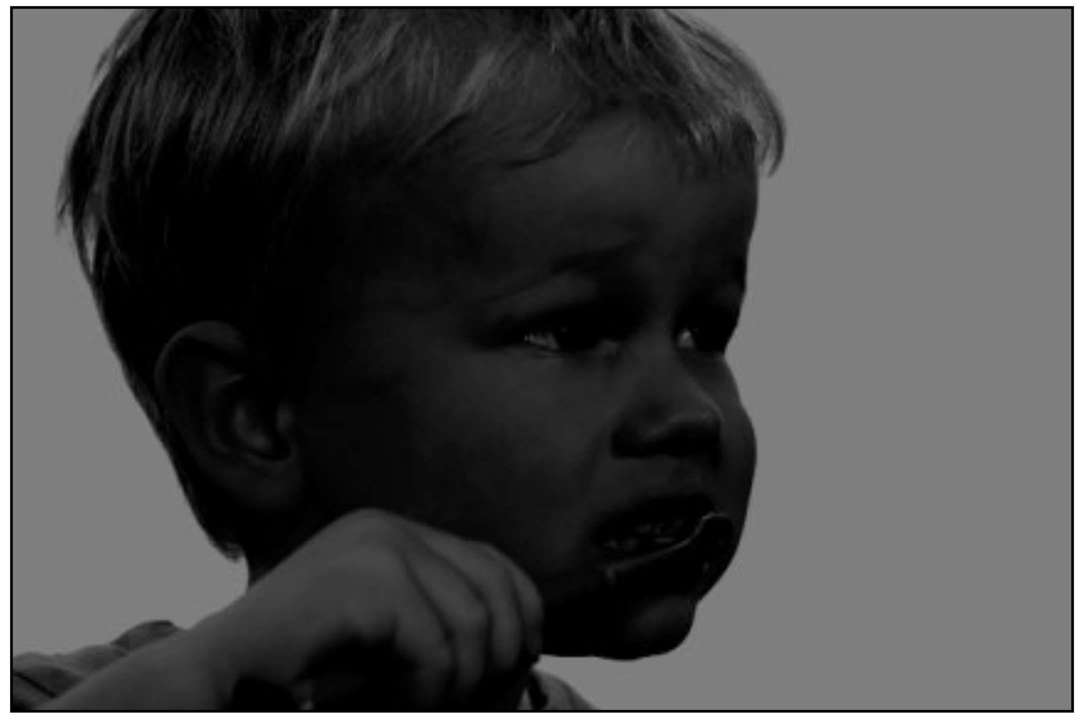
Examples of **Point Processing**

original



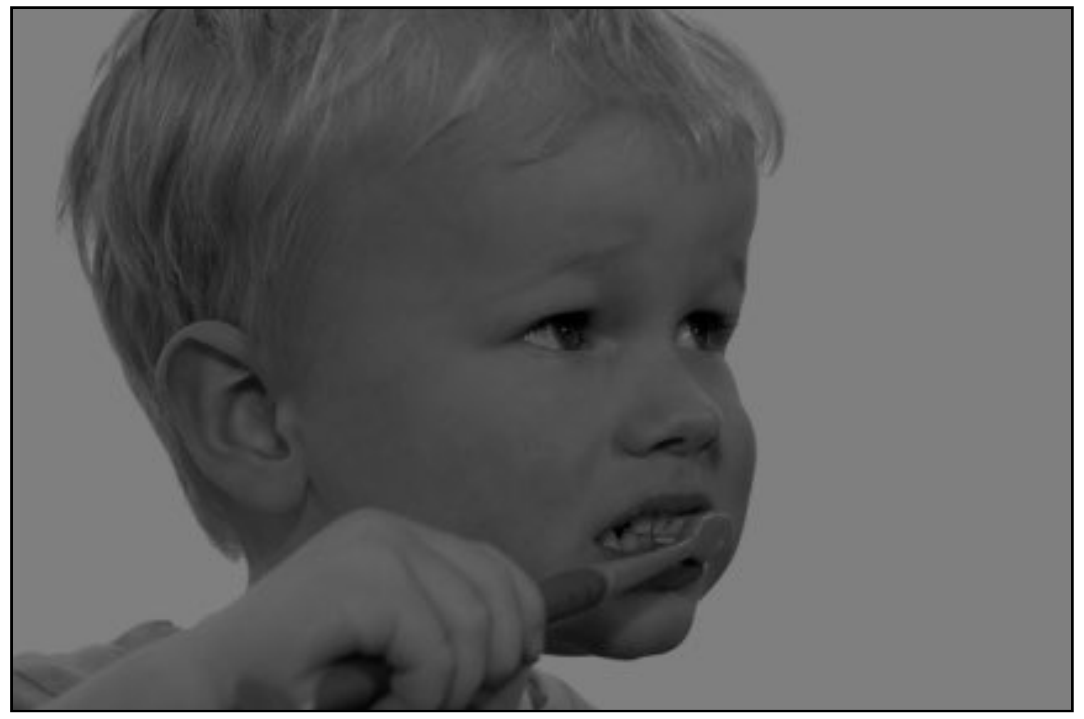
$I(X, Y)$

darken

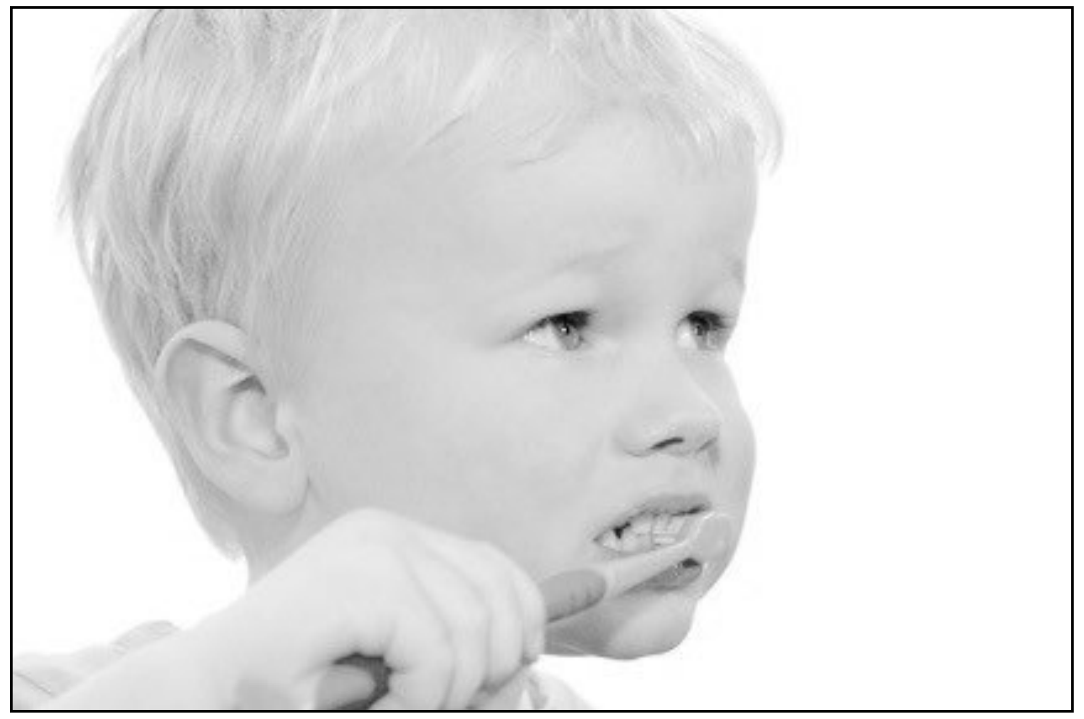


$I(X, Y) - 128$

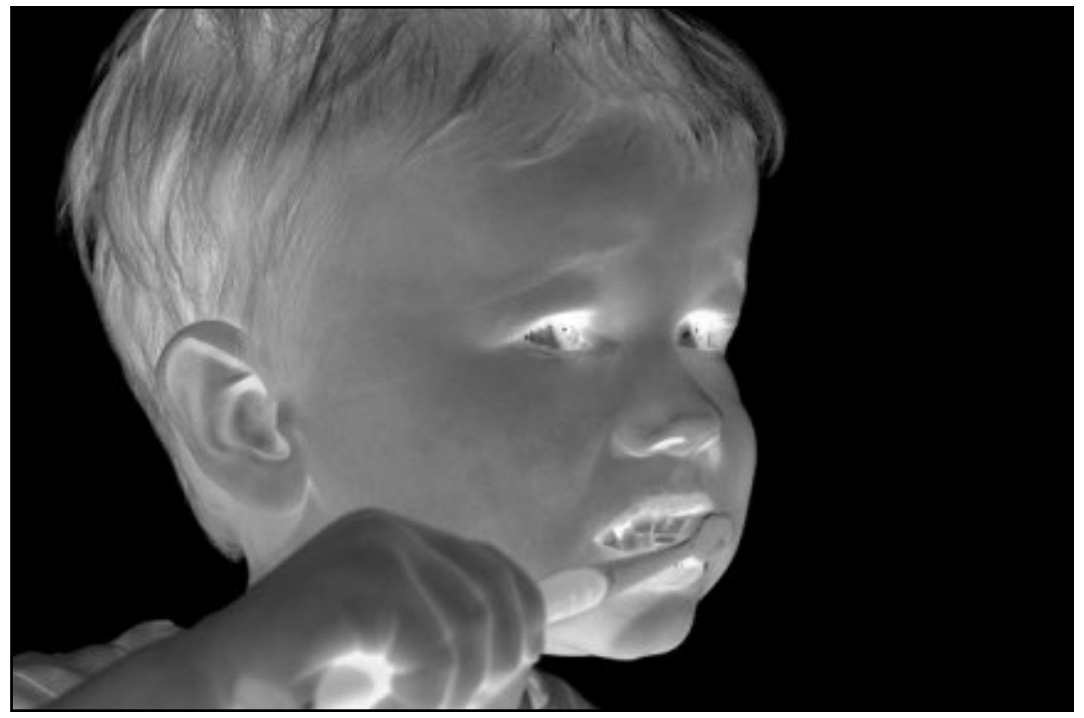
lower contrast



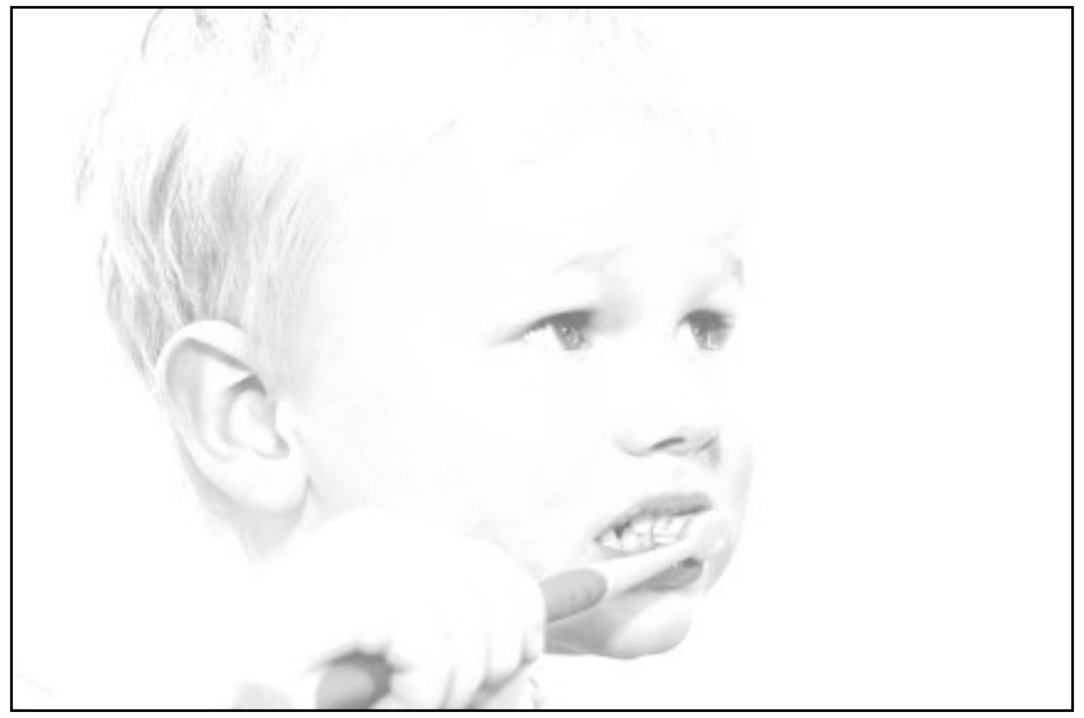
non-linear lower contrast



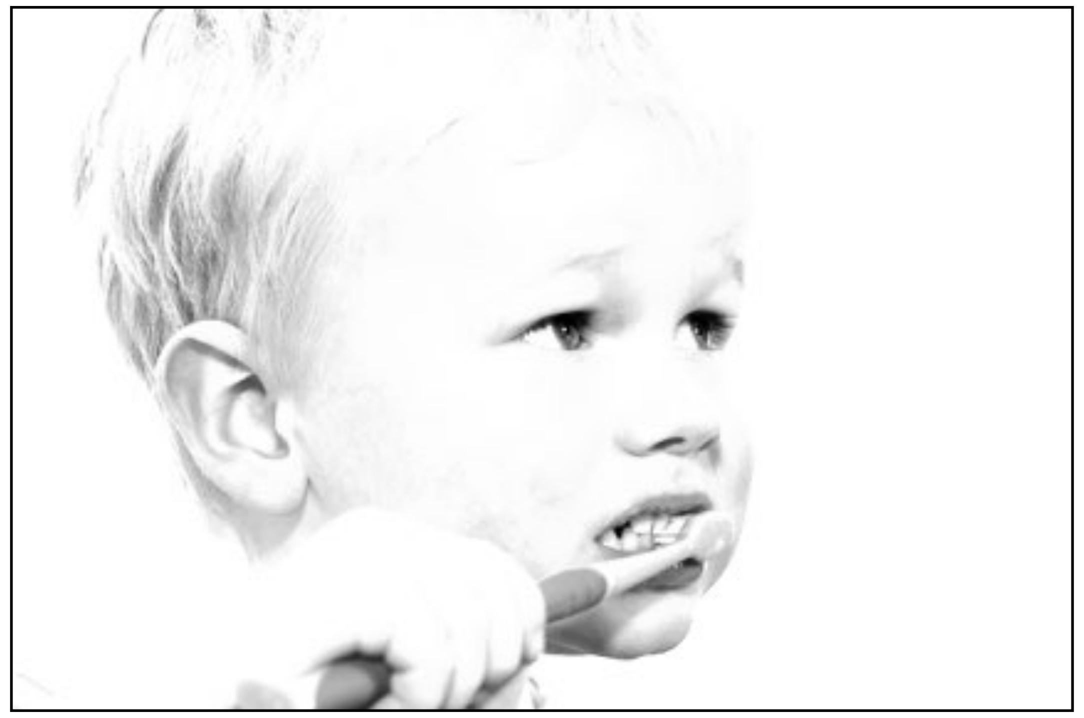
invert



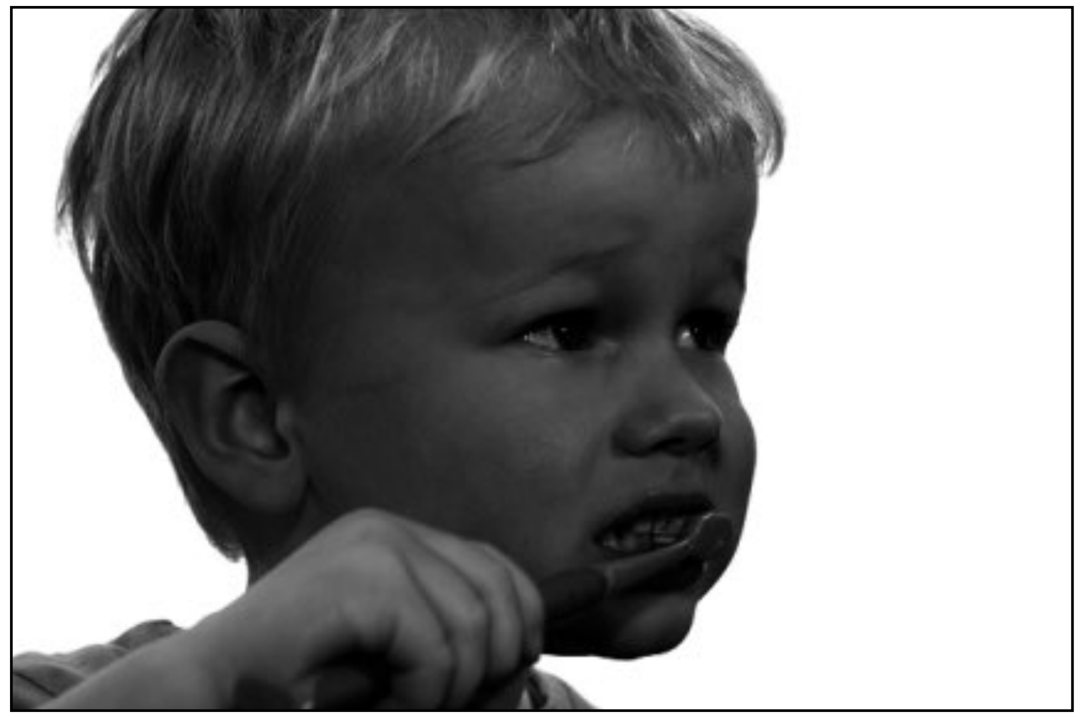
lighten



raise contrast



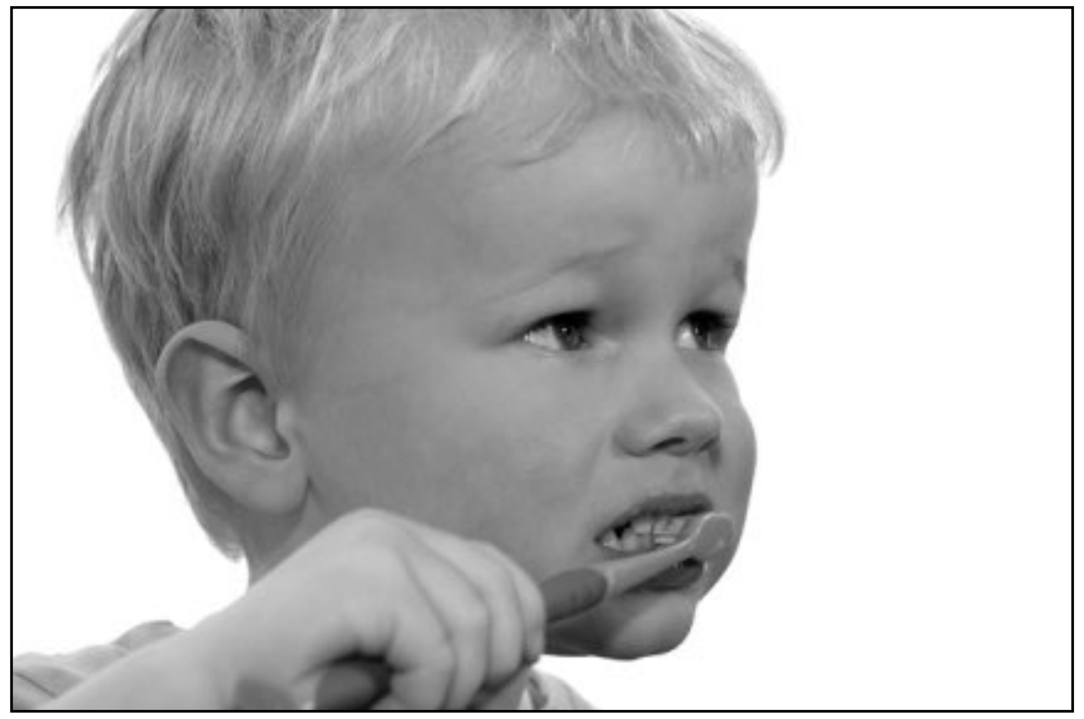
non-linear raise contrast



Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

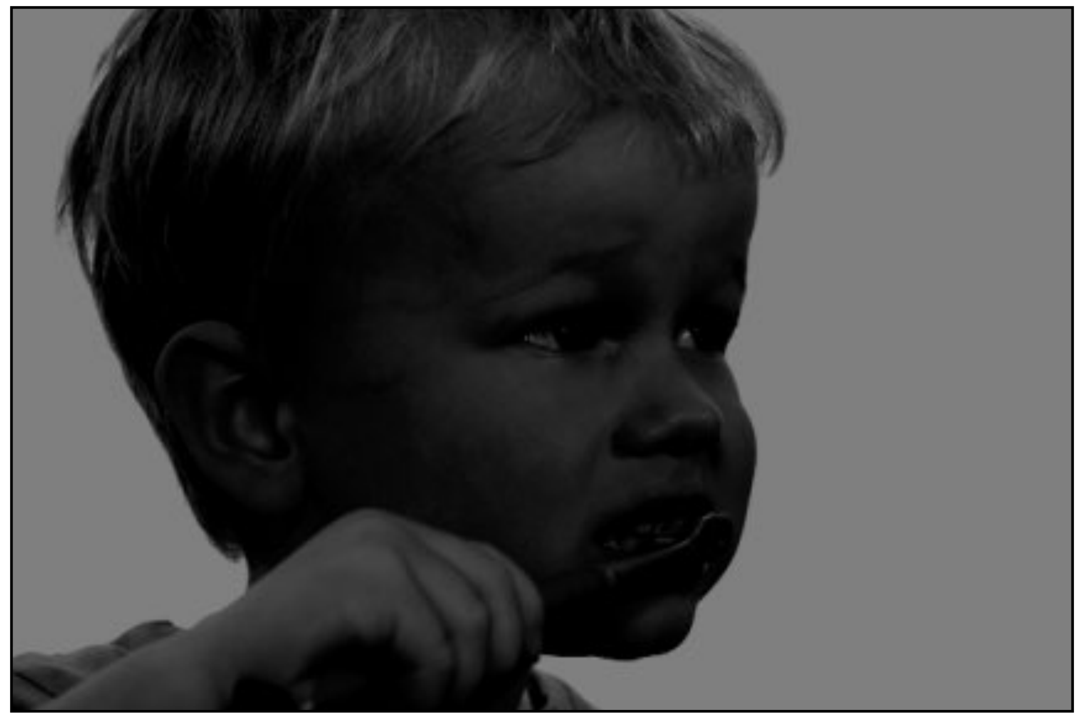
Examples of **Point Processing**

original



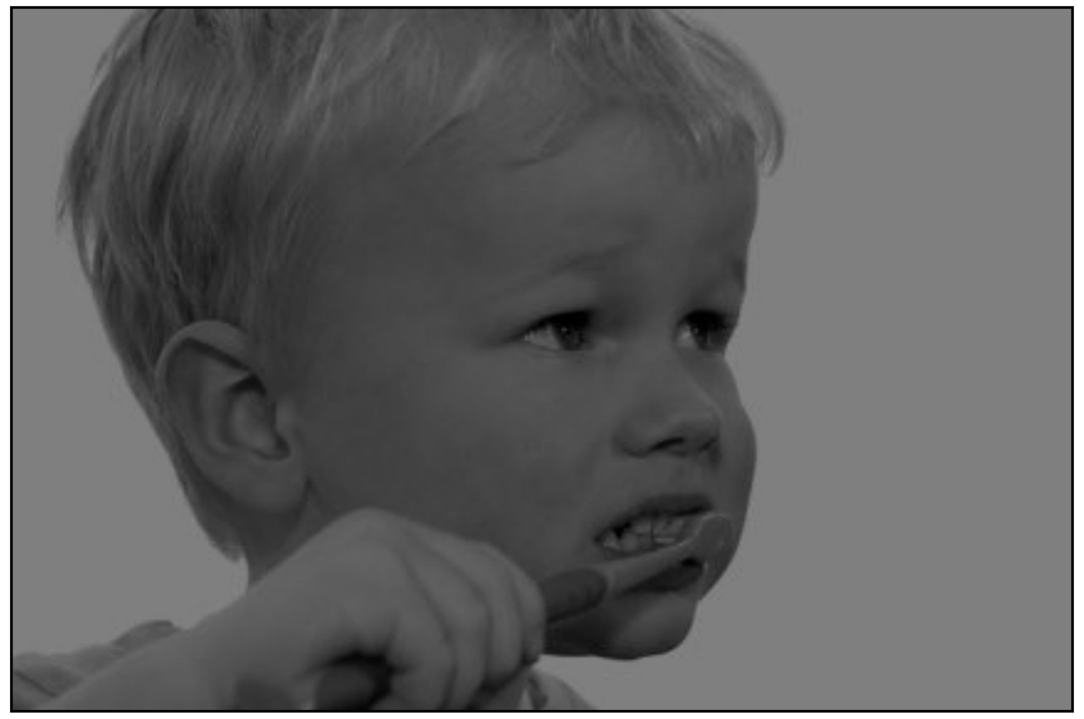
$I(X,Y)$

darken



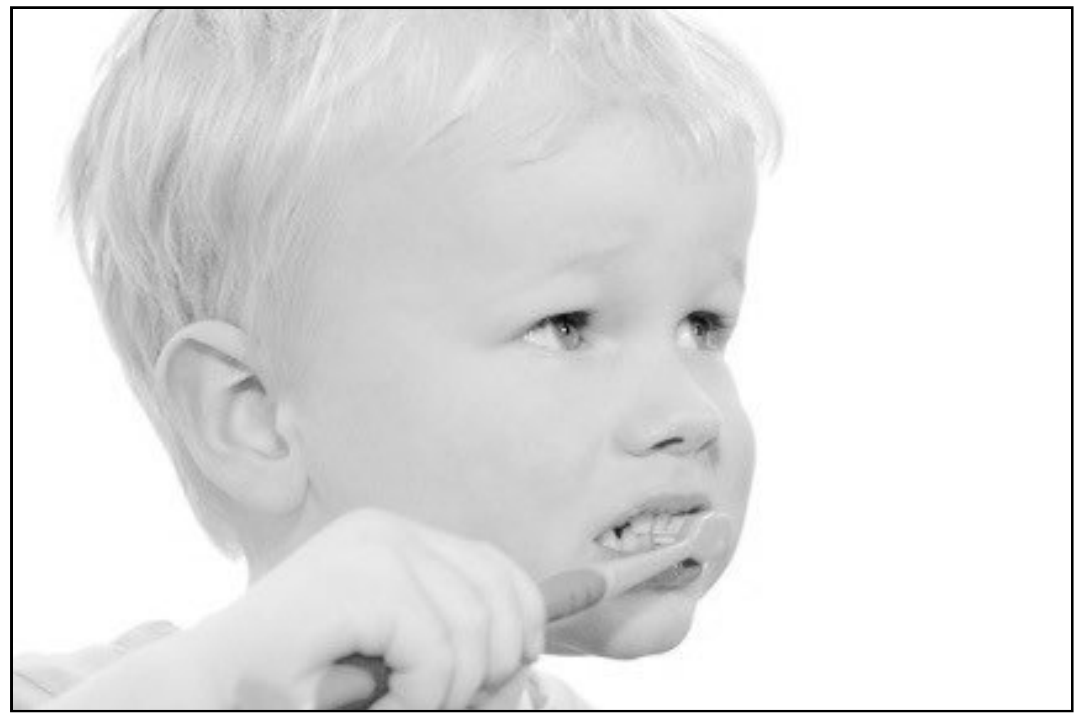
$I(X,Y) - 128$

lower contrast

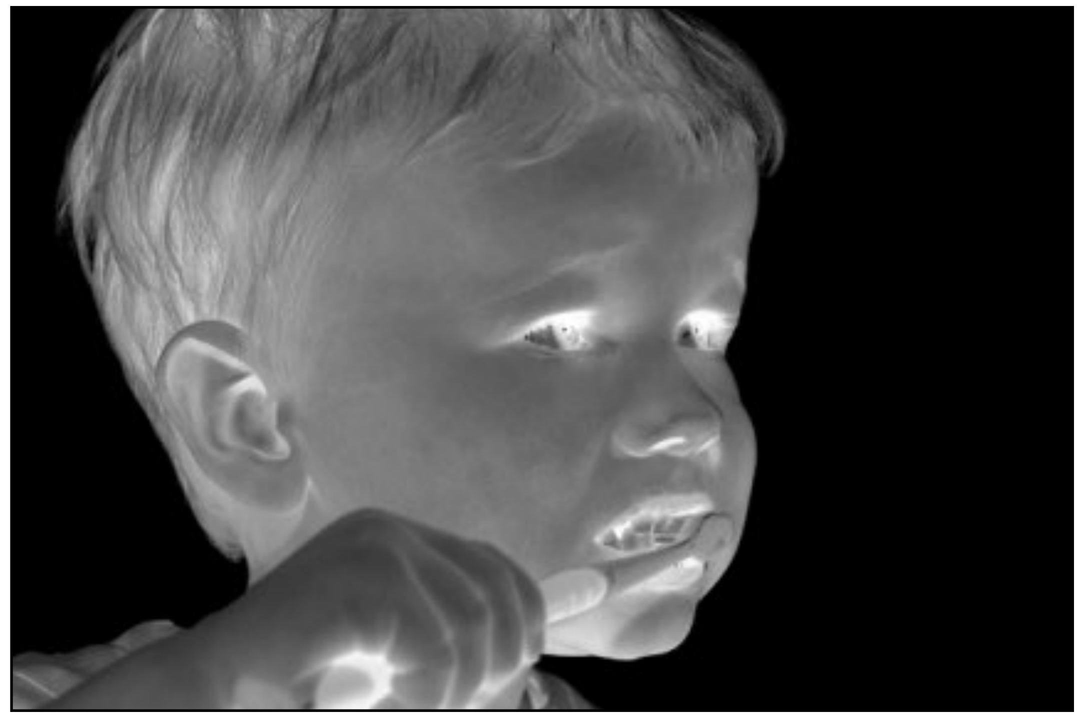


$$\frac{I(X,Y)}{2}$$

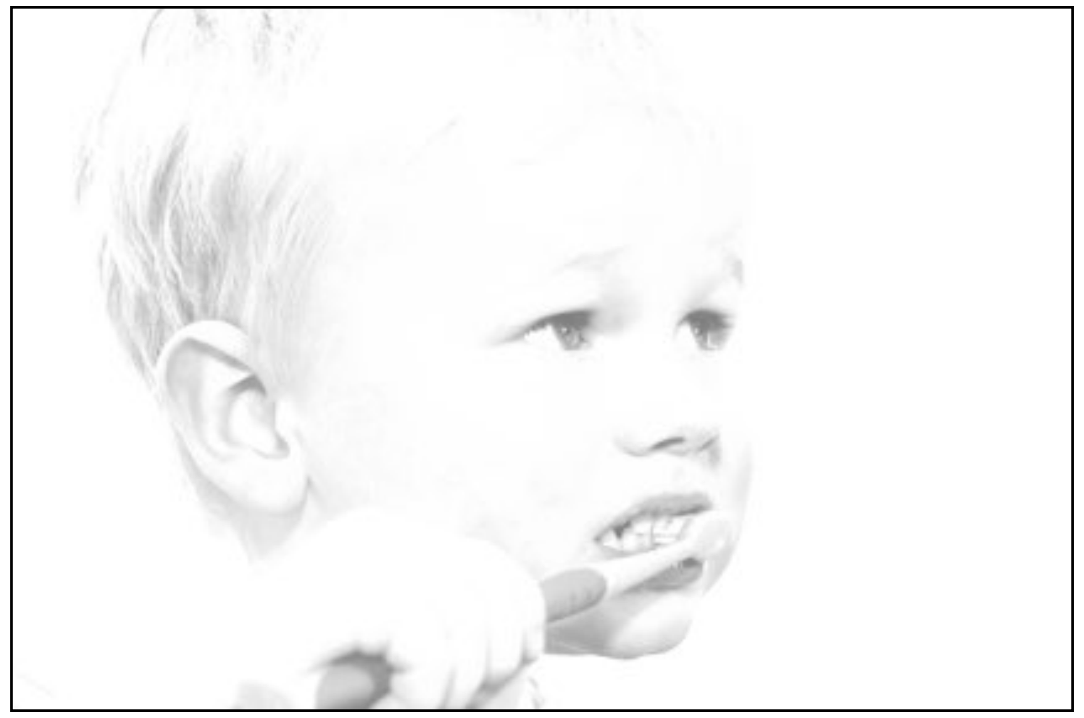
non-linear lower contrast



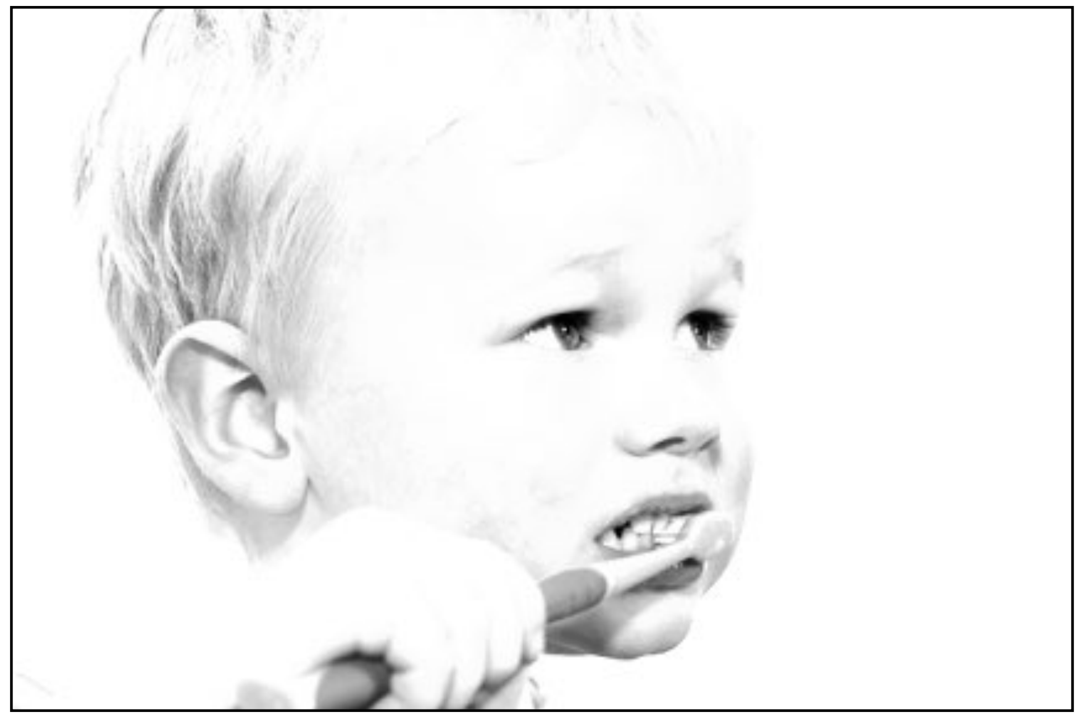
invert



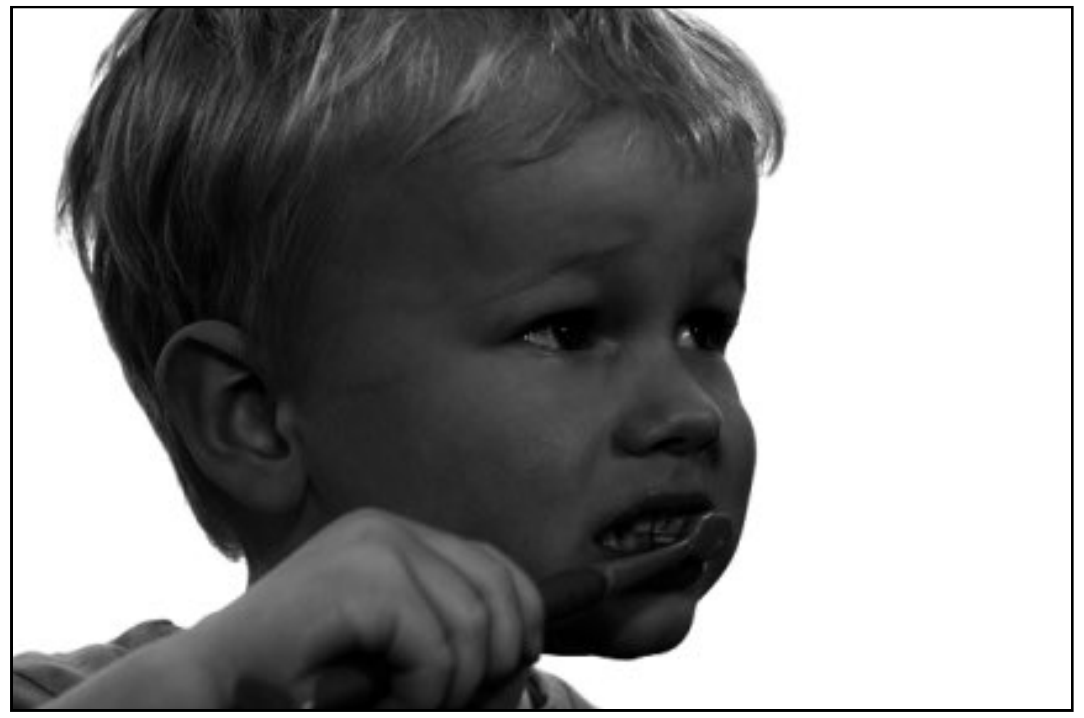
lighten



raise contrast



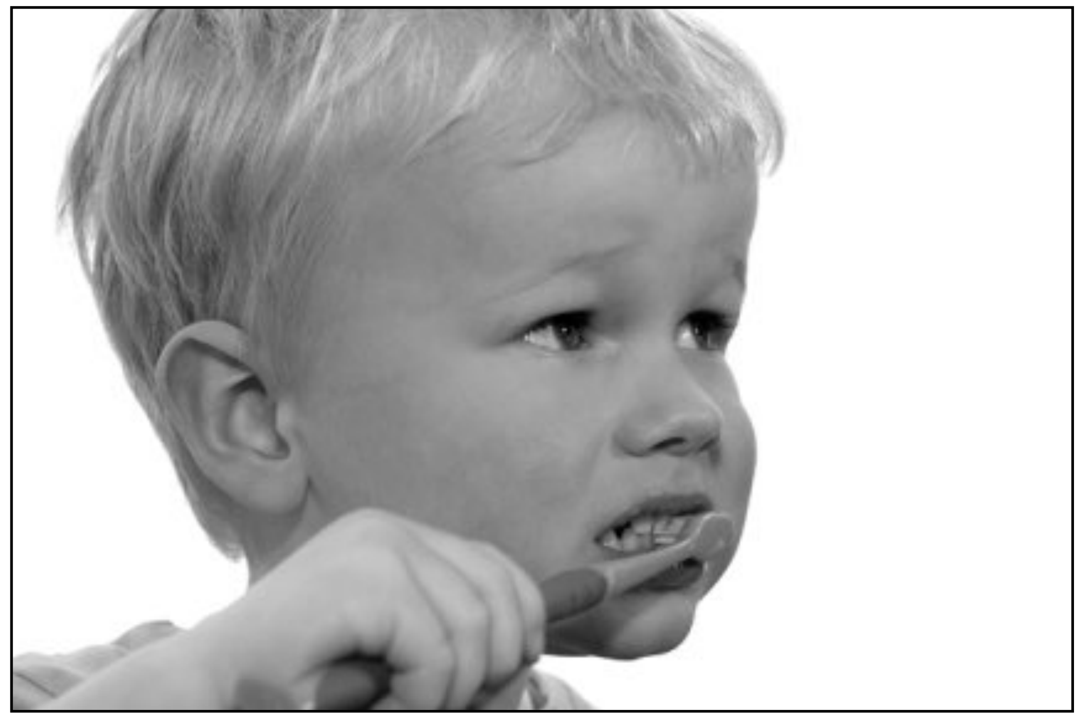
non-linear raise contrast



Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

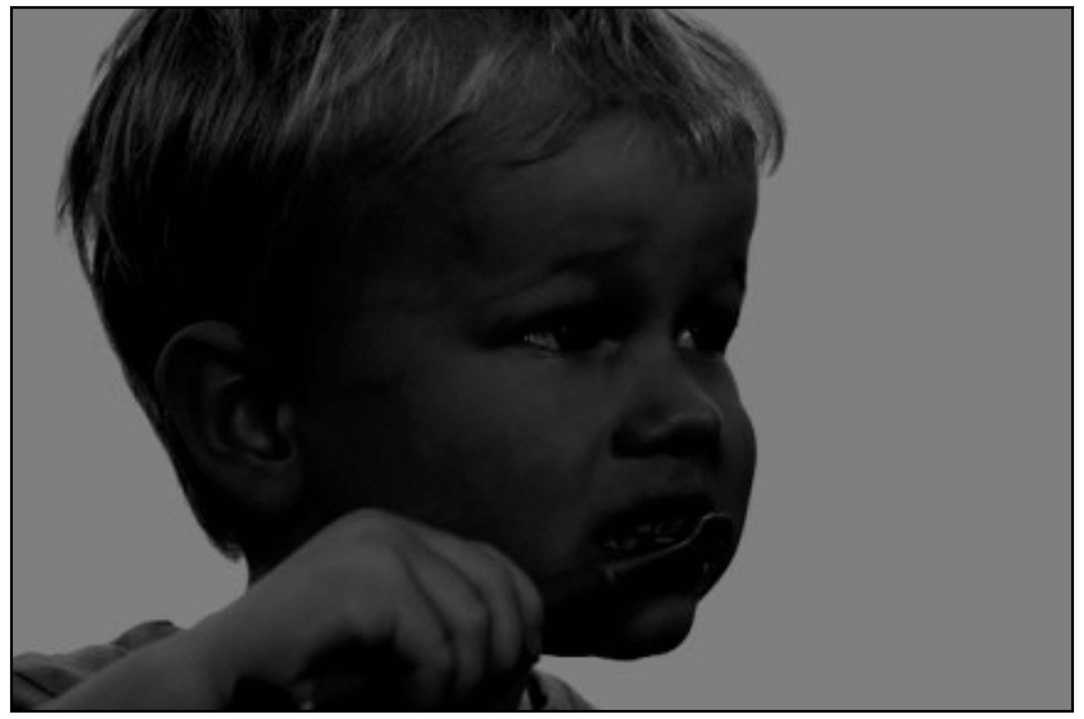
Examples of **Point Processing**

original



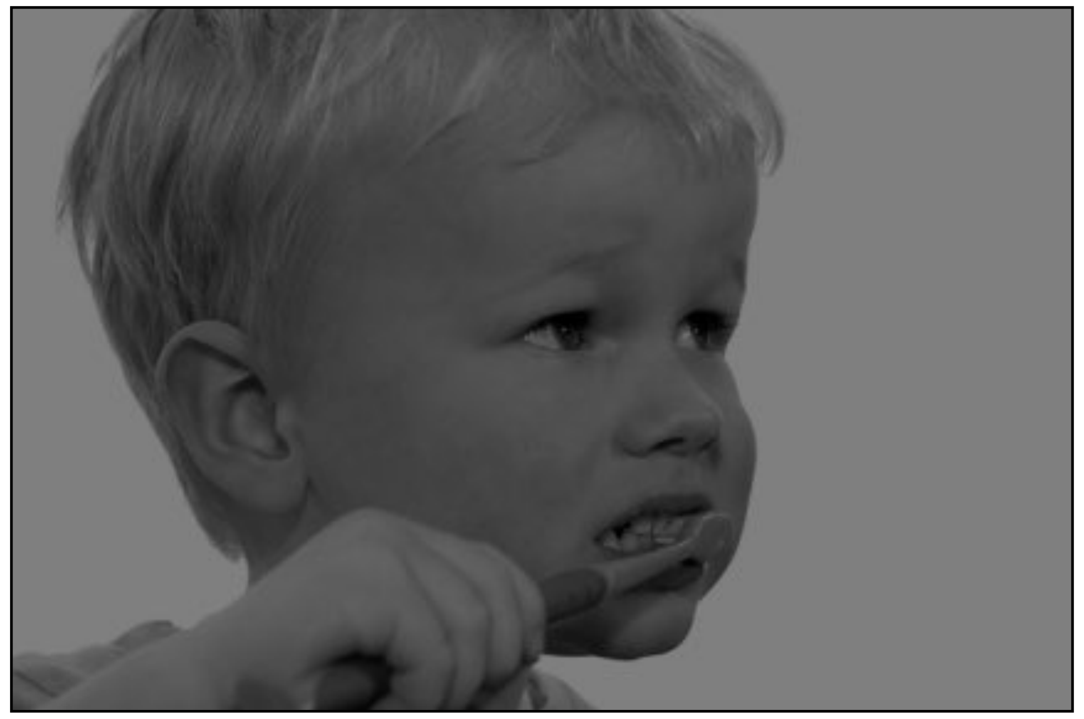
$$I(X,Y)$$

darken



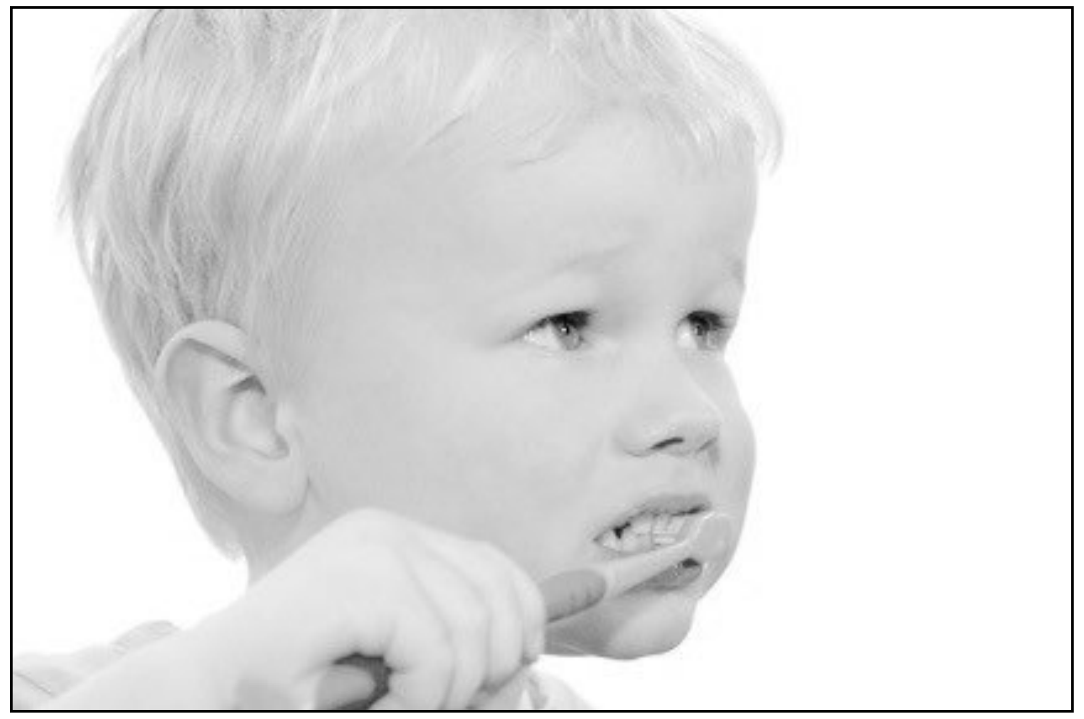
$$I(X,Y) - 128$$

lower contrast



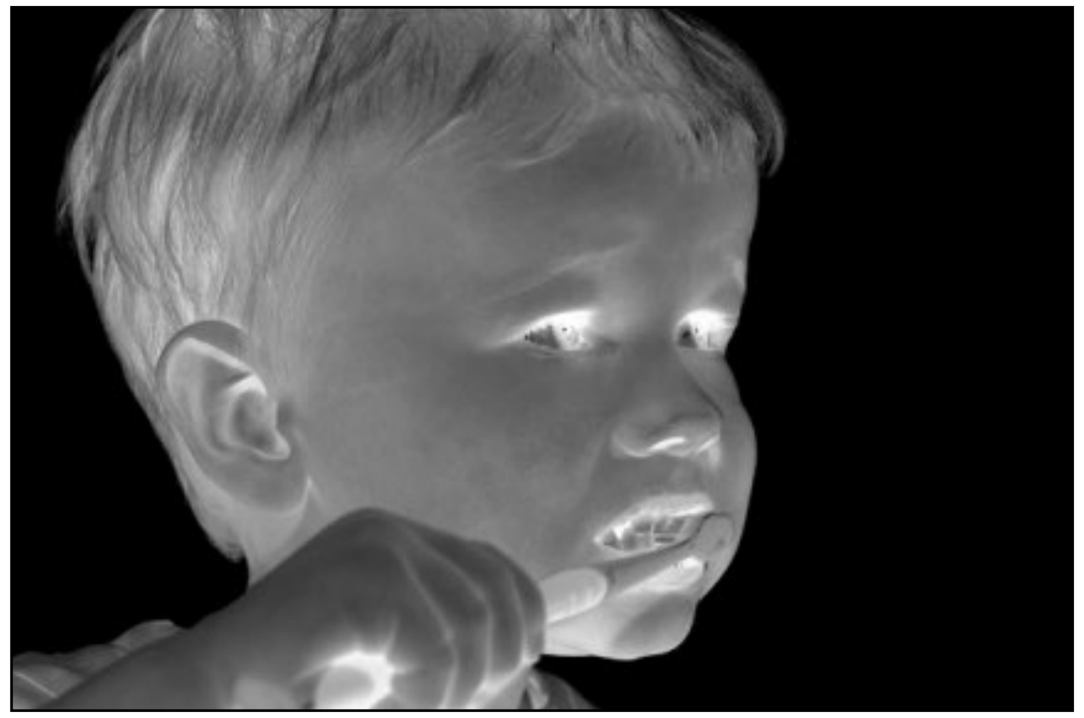
$$\frac{I(X,Y)}{2}$$

non-linear lower contrast

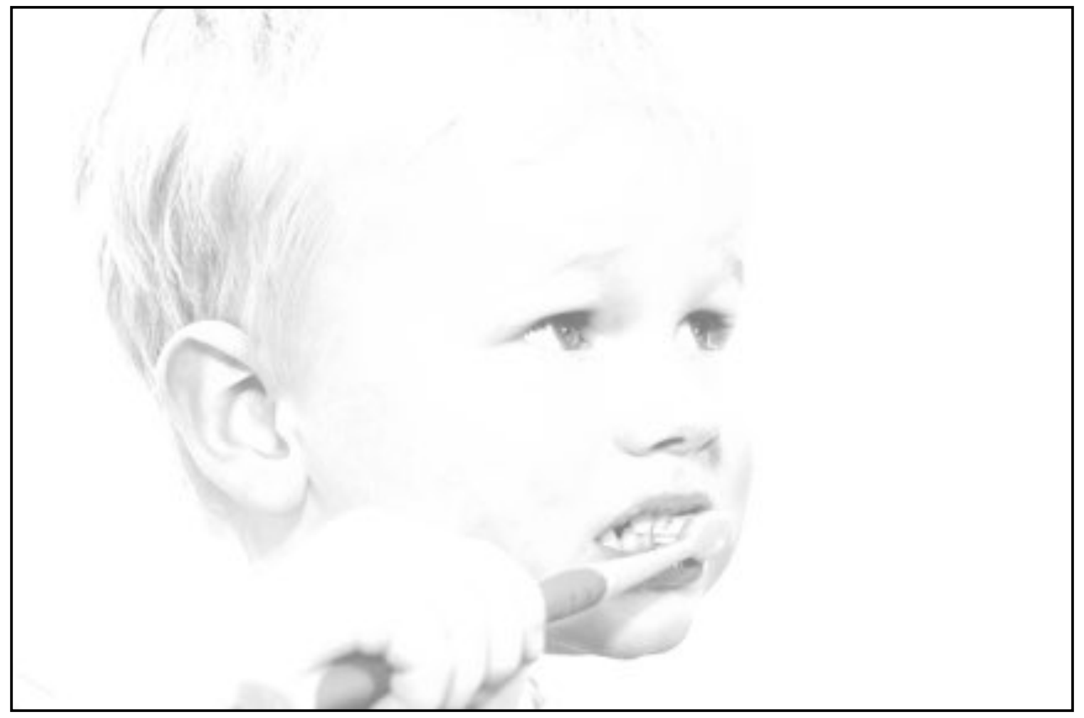


$$\left(\frac{I(X,Y)}{255}\right)^{1/3} \times 255$$

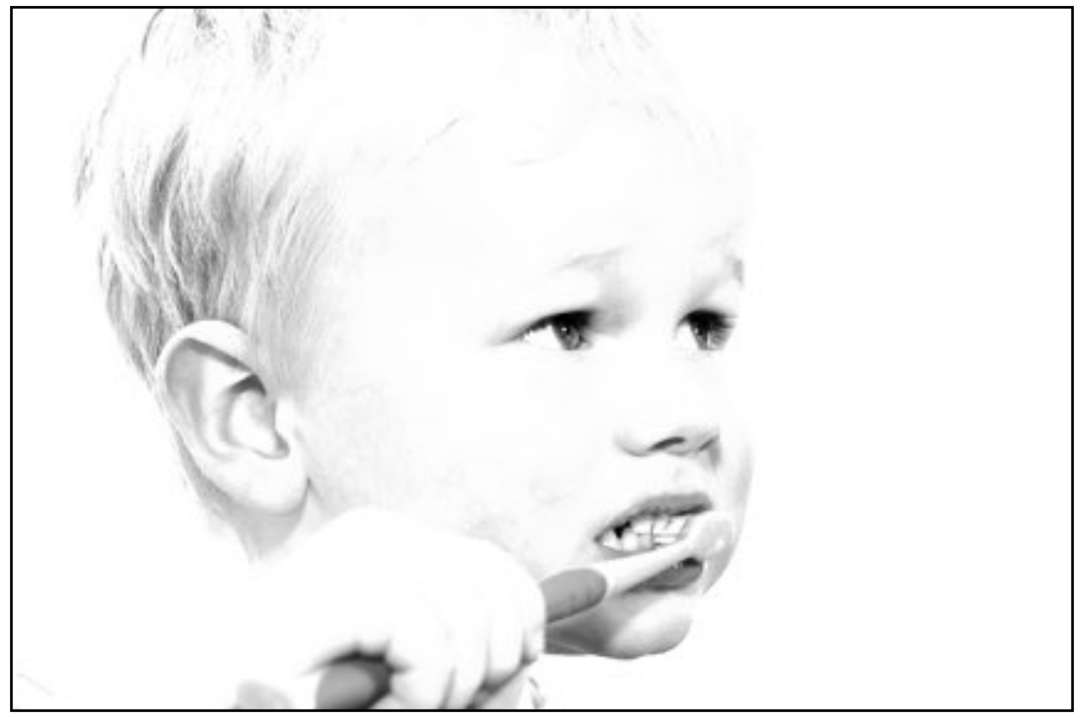
invert



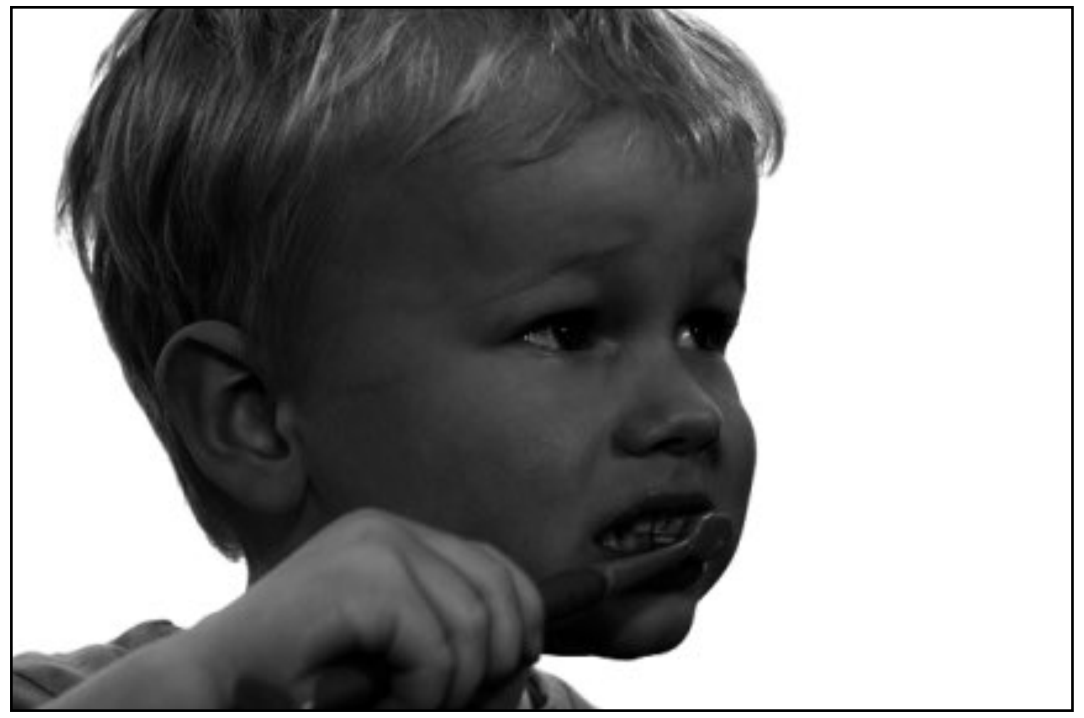
lighten



raise contrast



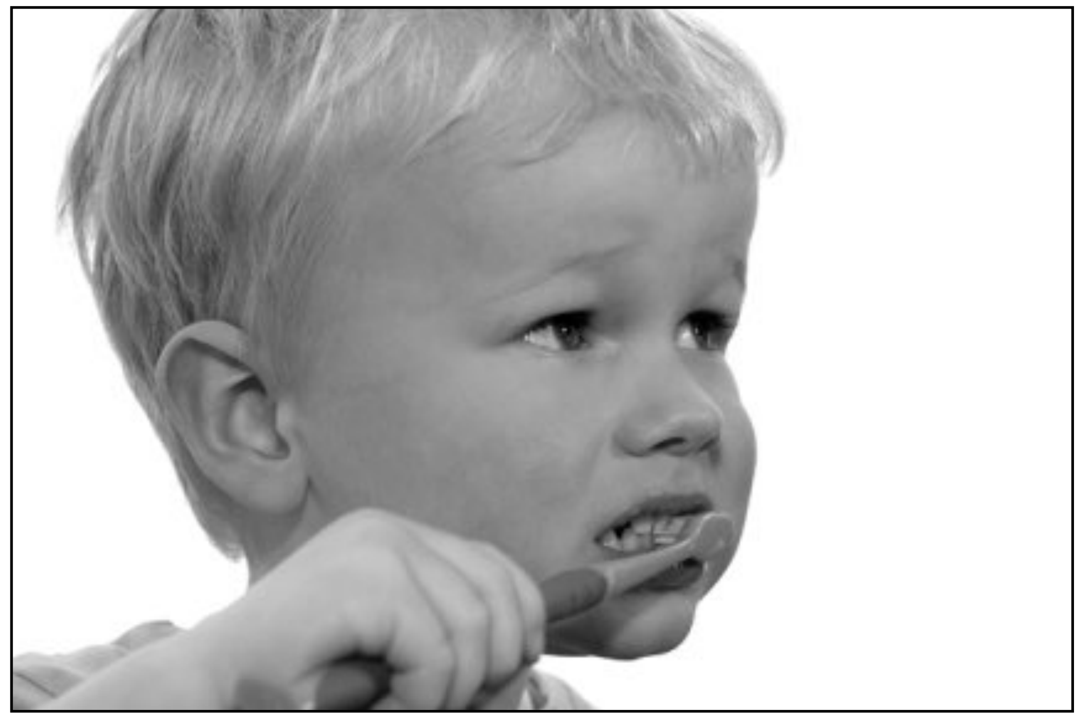
non-linear raise contrast



Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

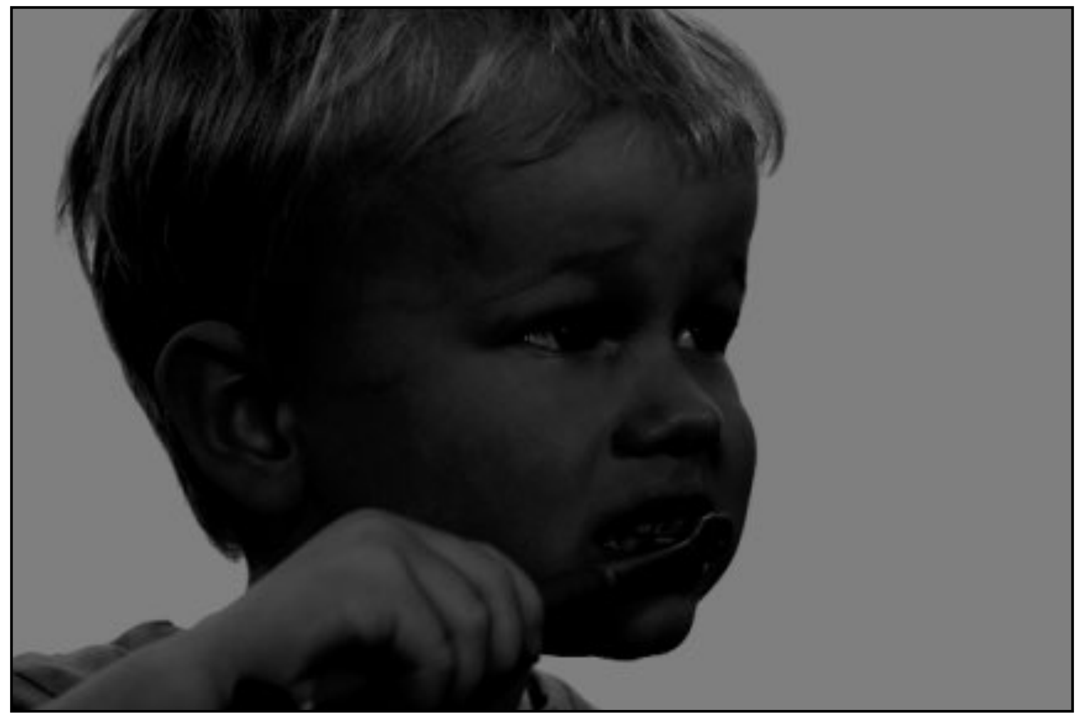
Examples of **Point Processing**

original



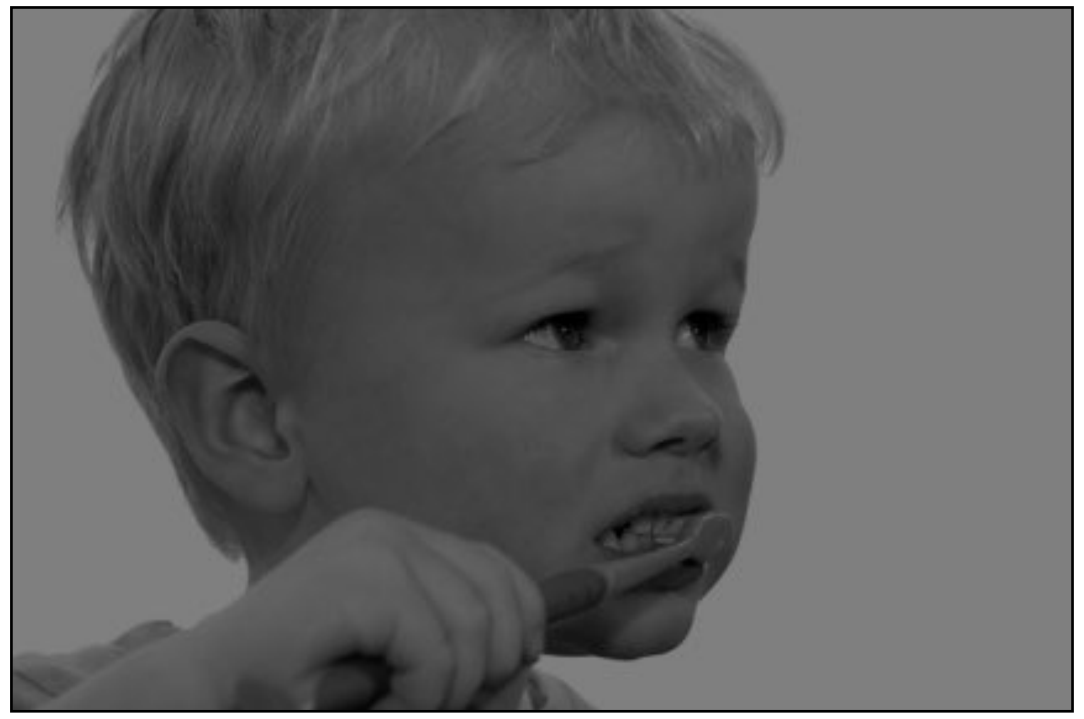
$$I(X,Y)$$

darken



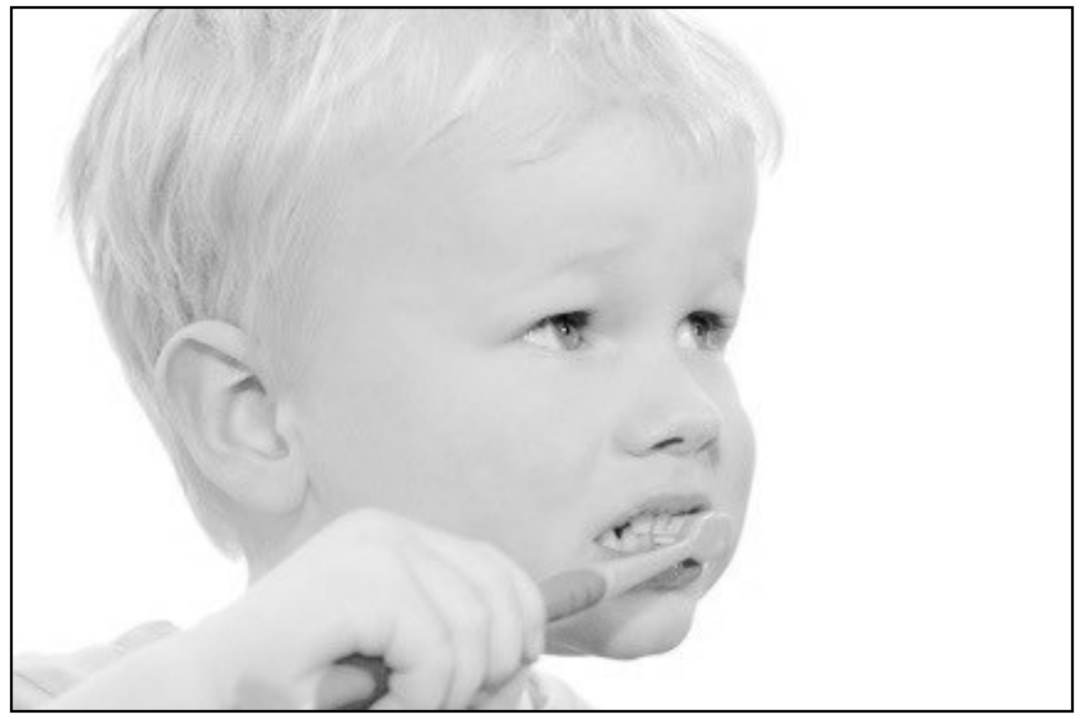
$$I(X,Y) - 128$$

lower contrast



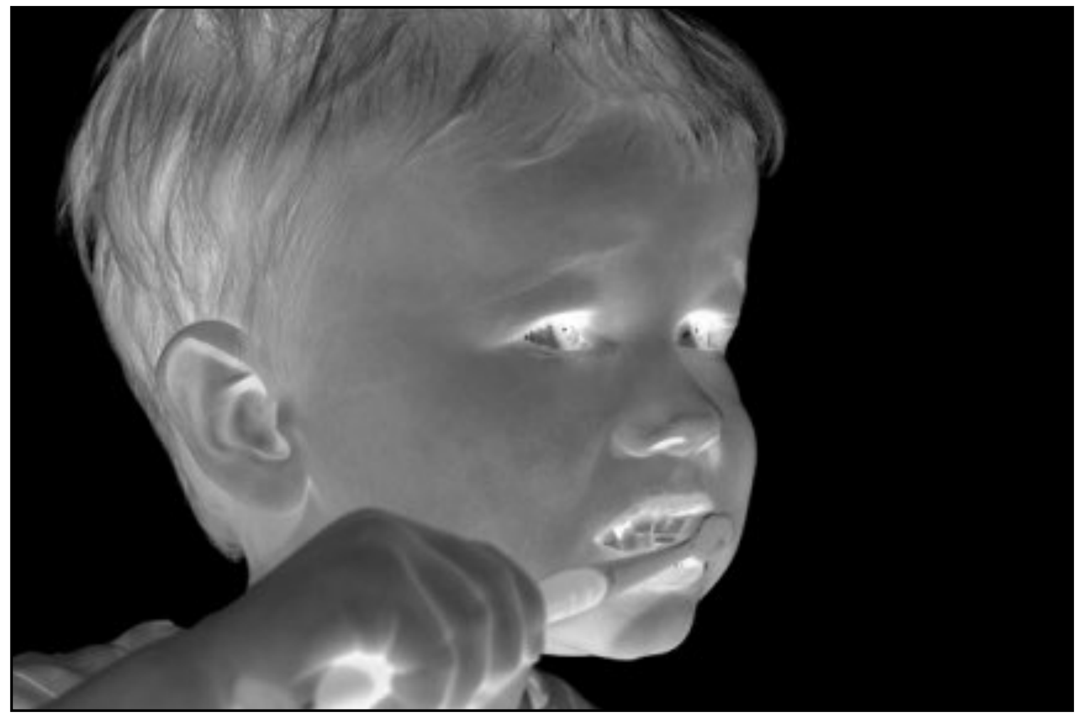
$$\frac{I(X,Y)}{2}$$

non-linear lower contrast



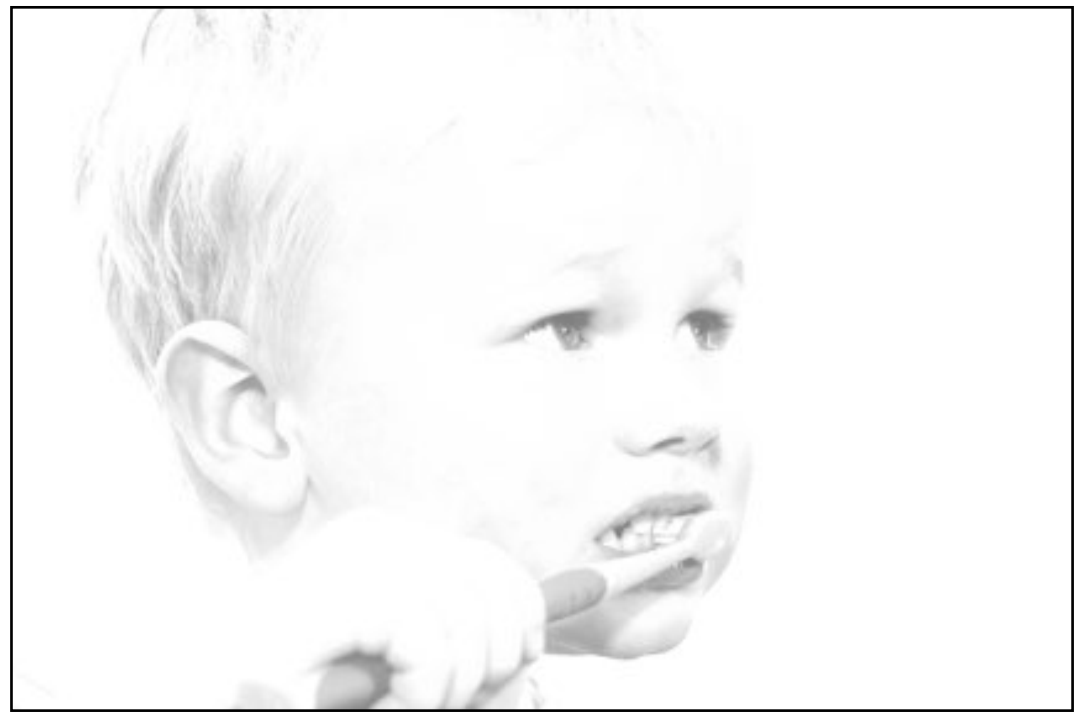
$$\left(\frac{I(X,Y)}{255}\right)^{1/3} \times 255$$

invert

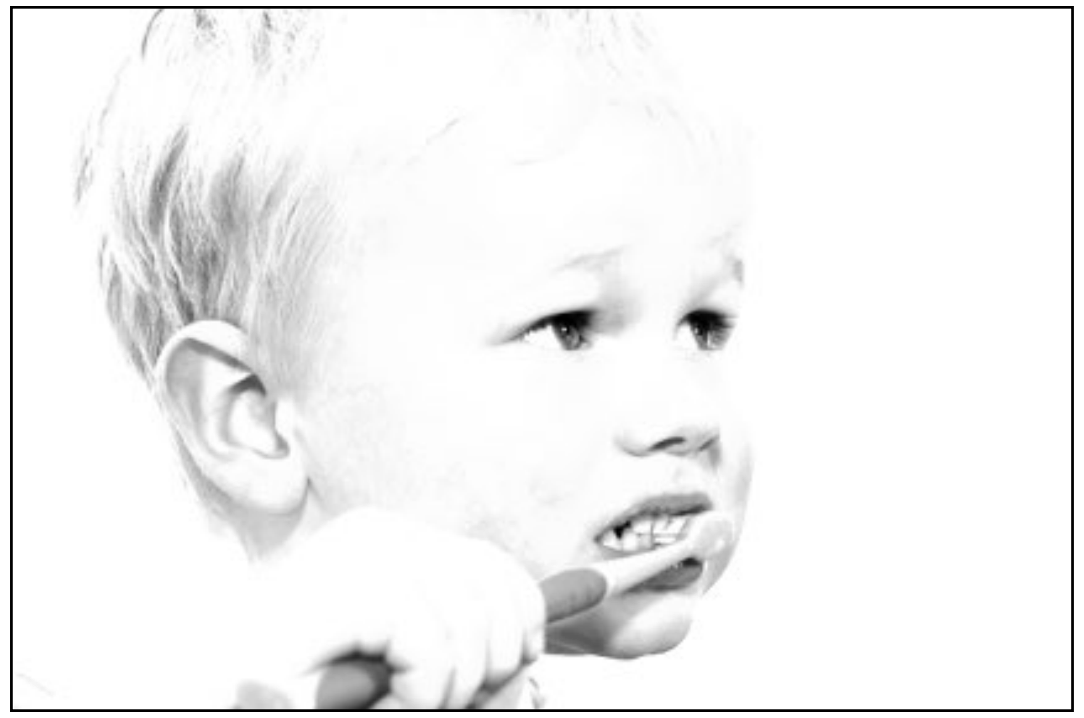


$$255 - I(X,Y)$$

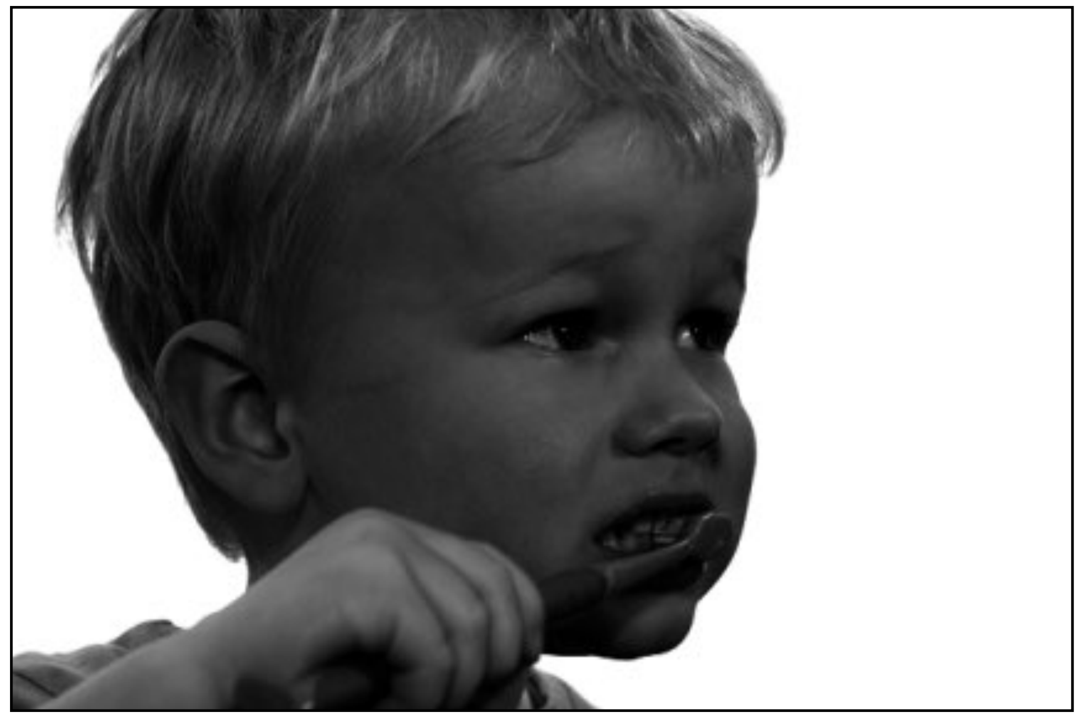
lighten



raise contrast



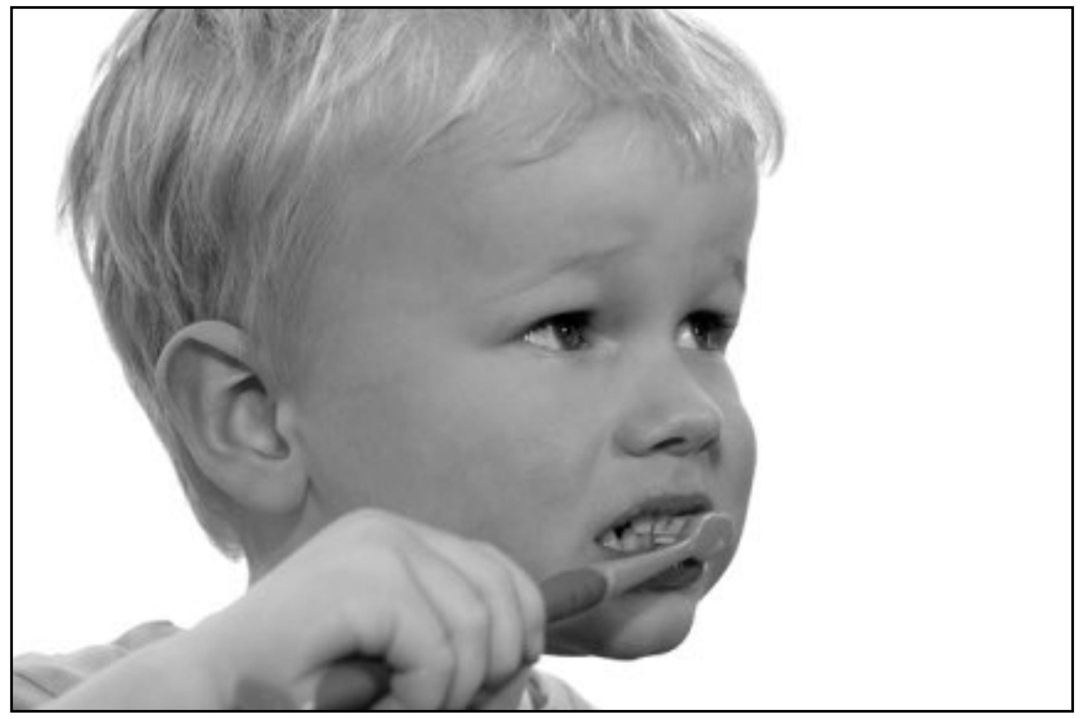
non-linear raise contrast



Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

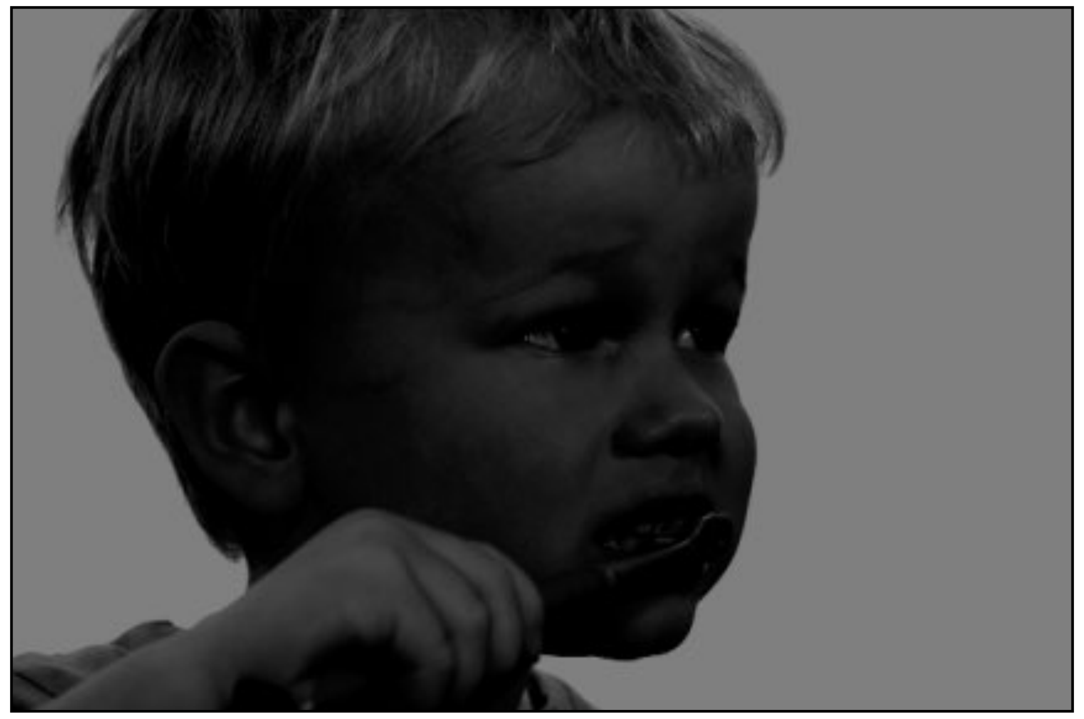
Examples of **Point Processing**

original



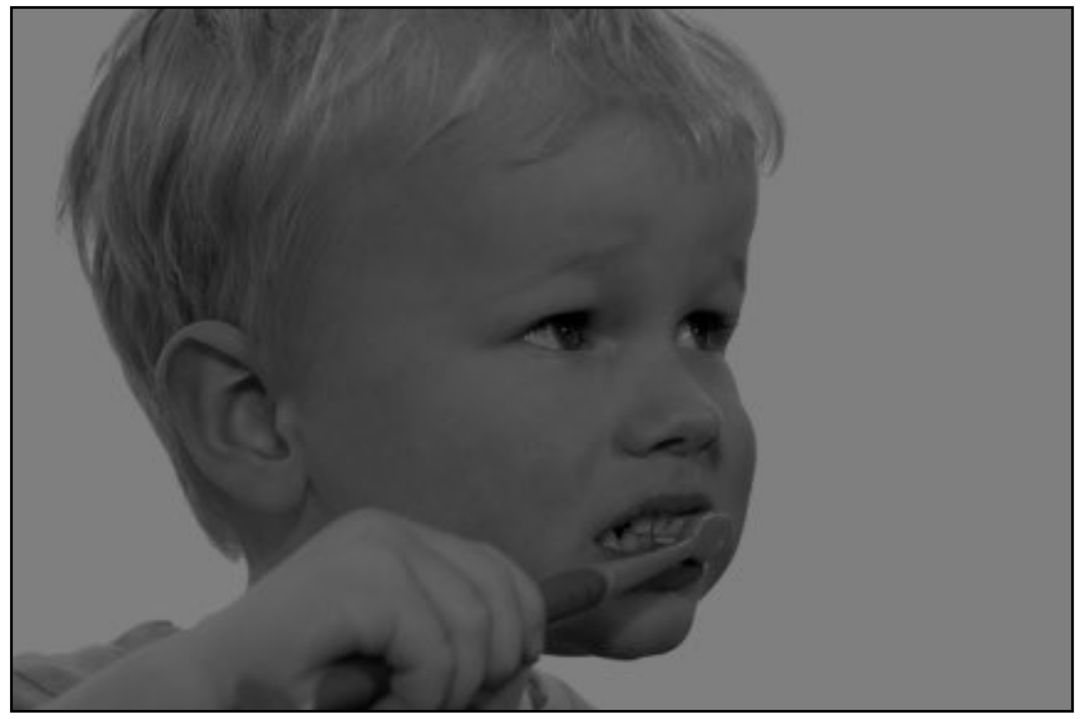
$$I(X,Y)$$

darken



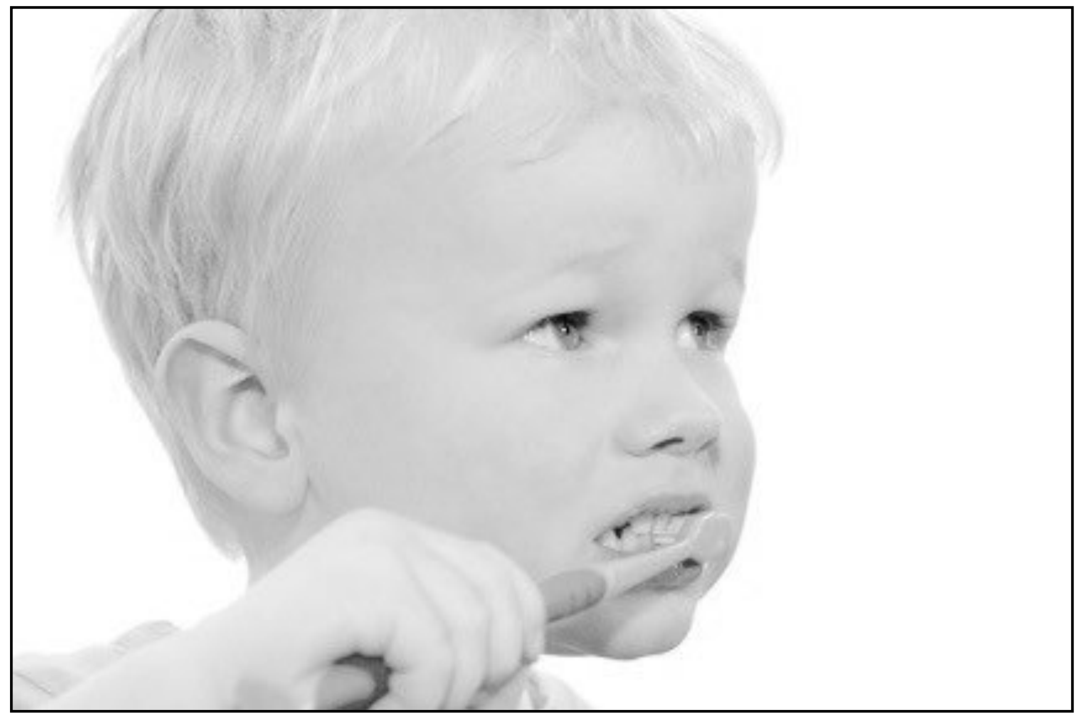
$$I(X,Y) - 128$$

lower contrast



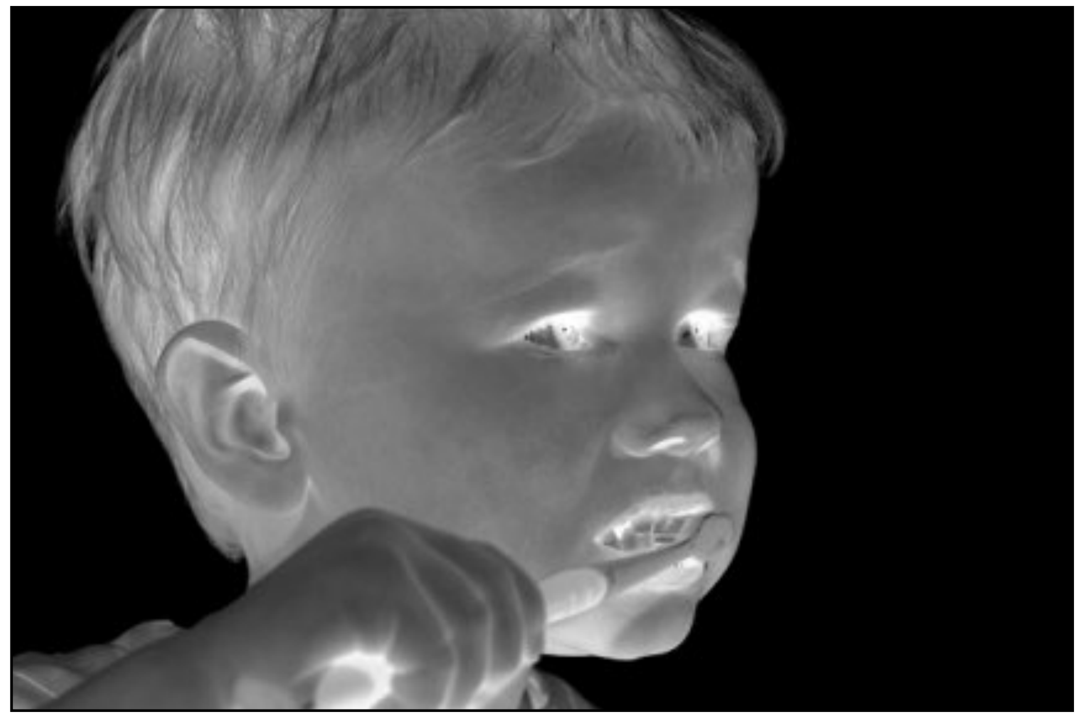
$$\frac{I(X,Y)}{2}$$

non-linear lower contrast



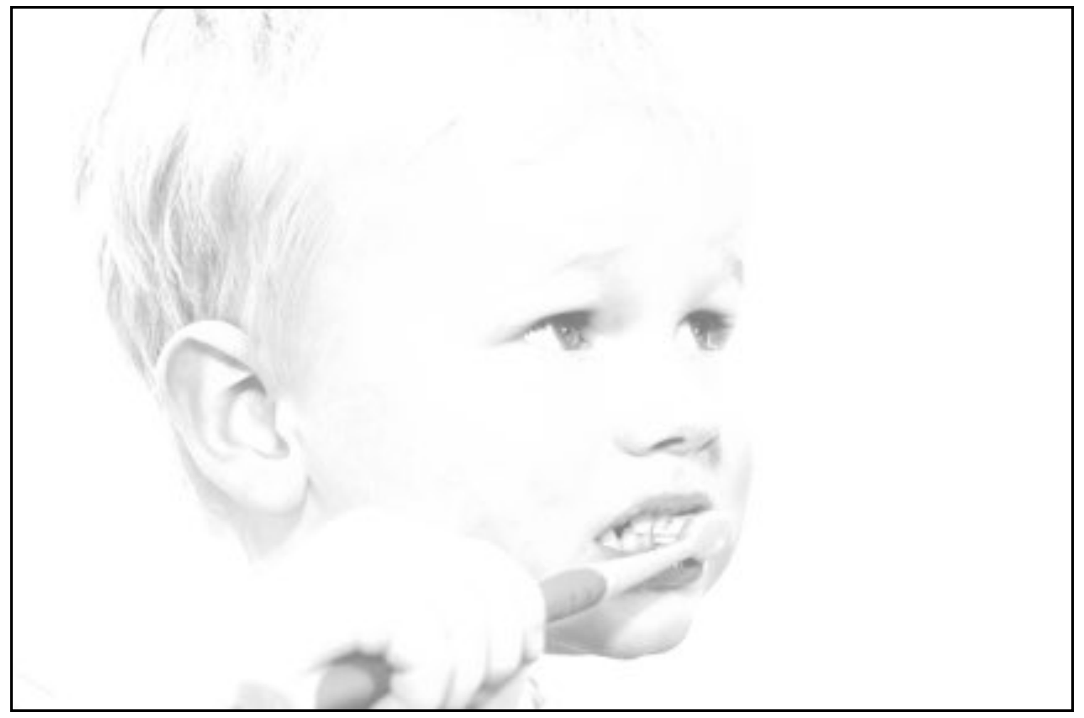
$$\left(\frac{I(X,Y)}{255}\right)^{1/3} \times 255$$

invert



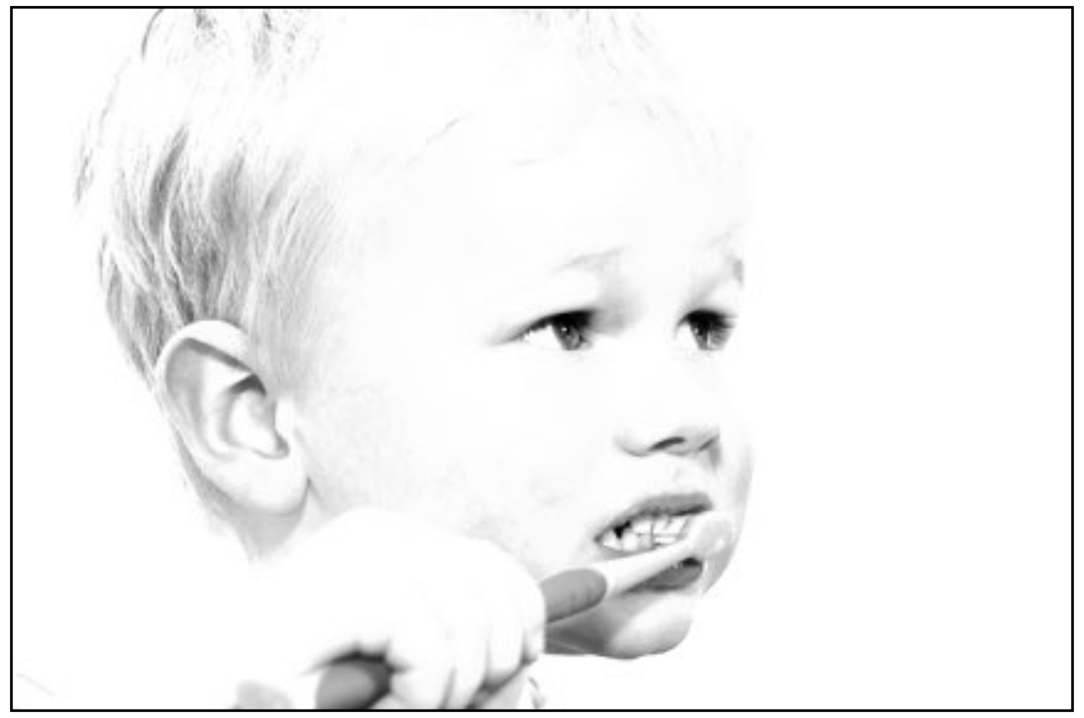
$$255 - I(X,Y)$$

lighten

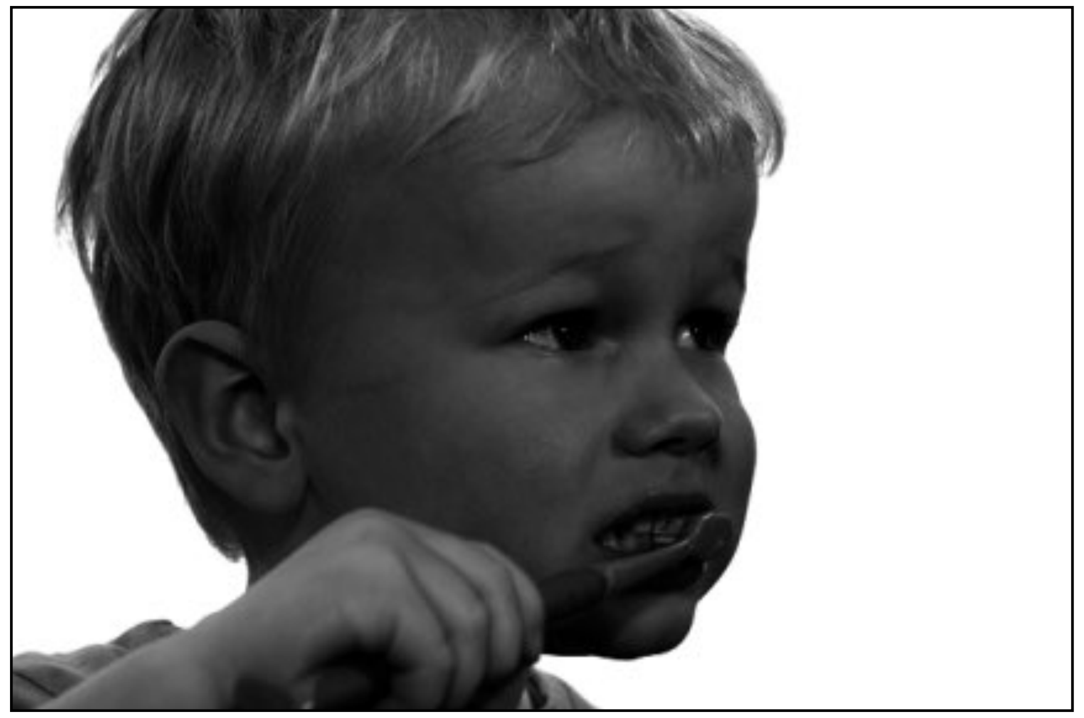


$$I(X,Y) + 128$$

raise contrast



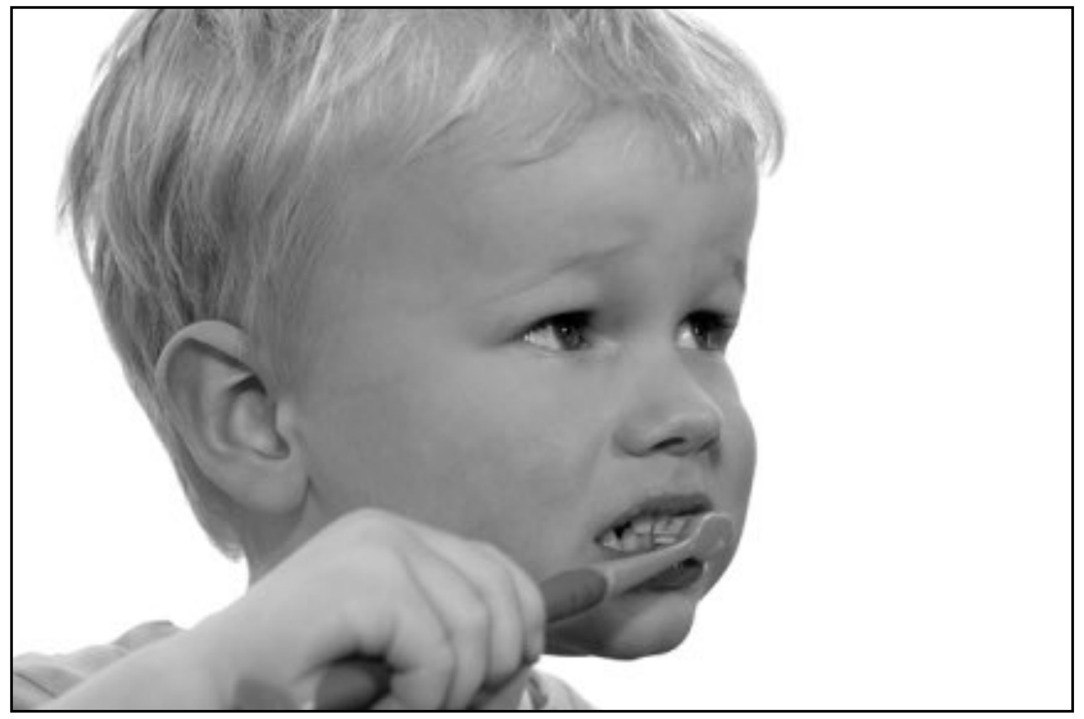
non-linear raise contrast



Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

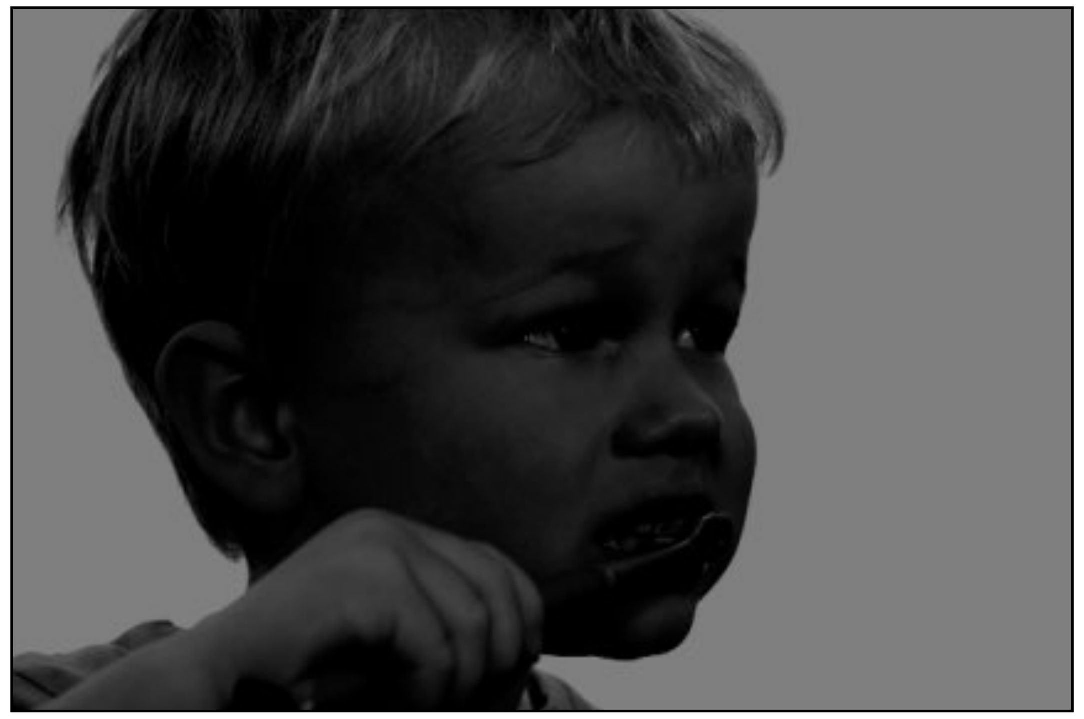
Examples of **Point Processing**

original



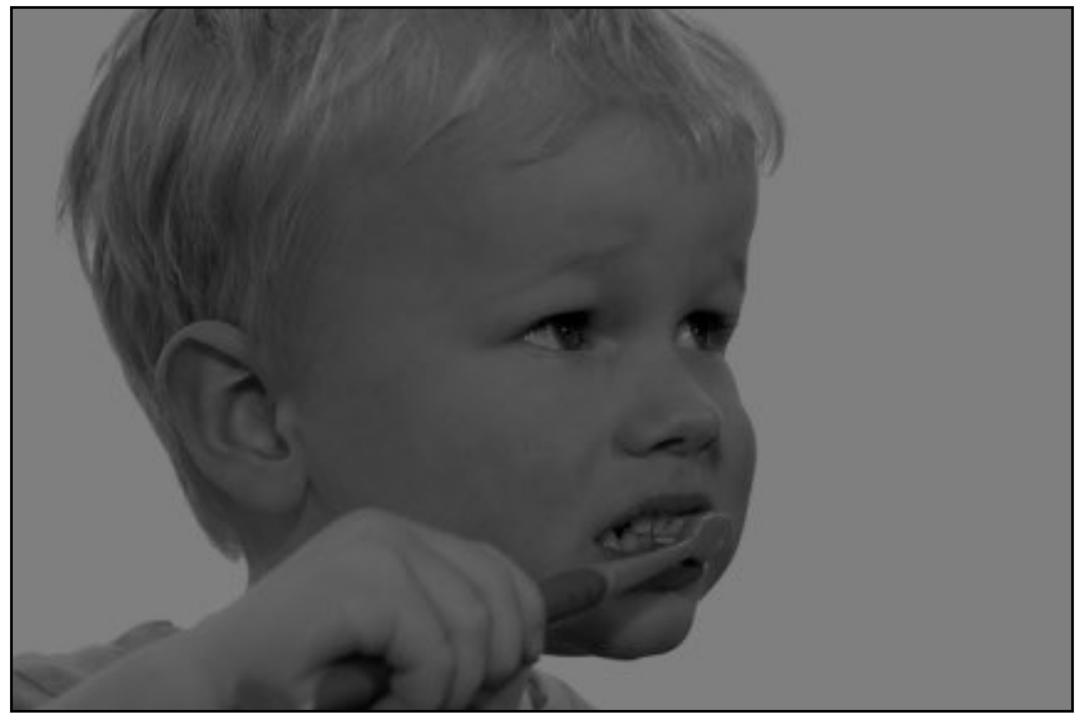
$I(X,Y)$

darken



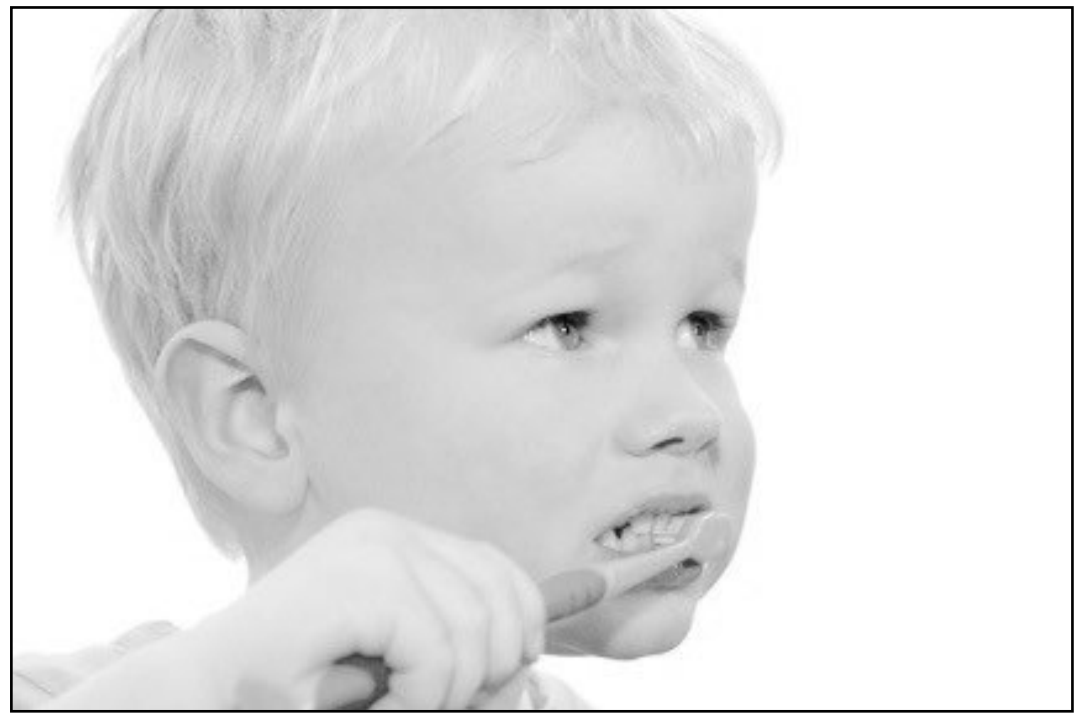
$I(X,Y) - 128$

lower contrast



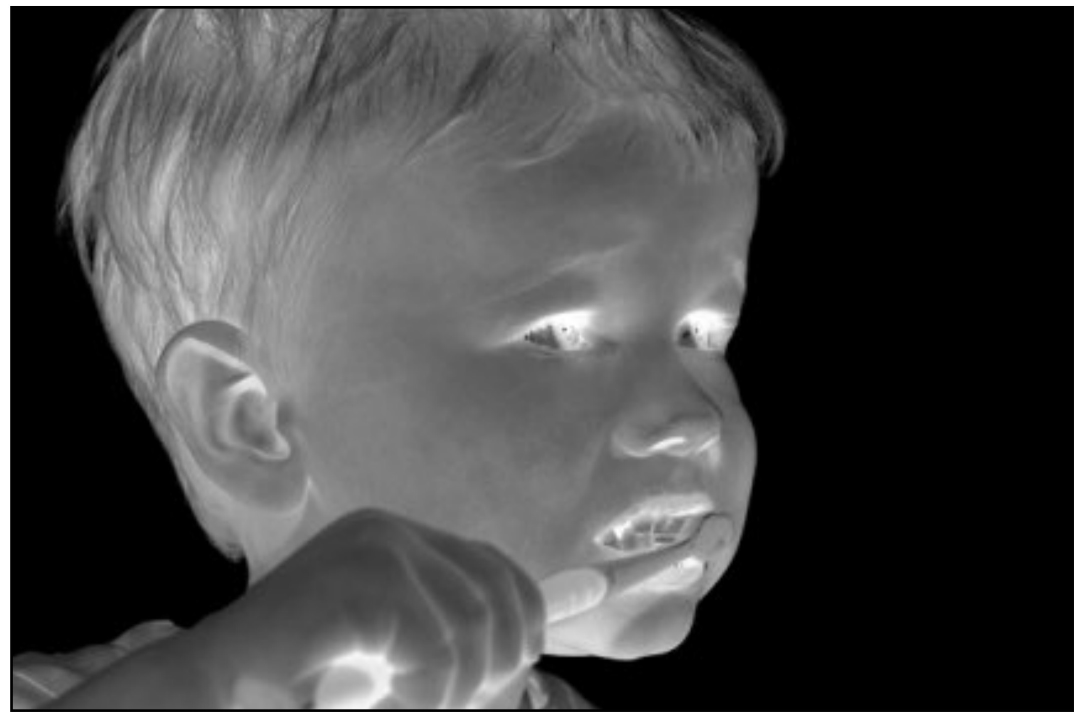
$\frac{I(X,Y)}{2}$

non-linear lower contrast



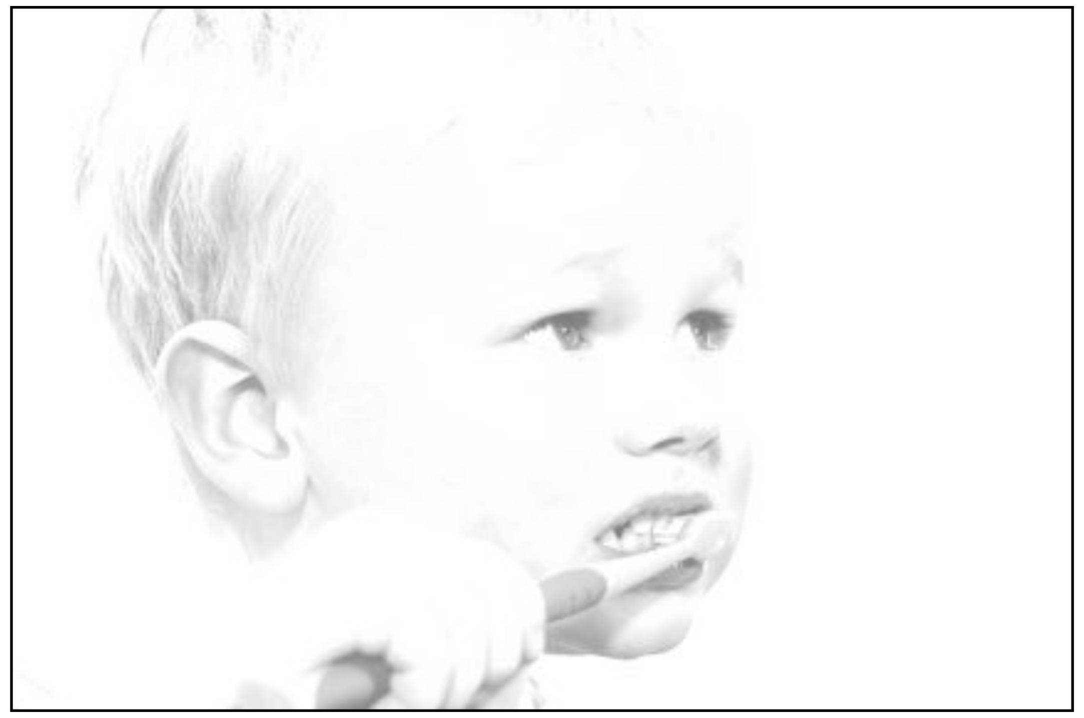
$\left(\frac{I(X,Y)}{255}\right)^{1/3} \times 255$

invert



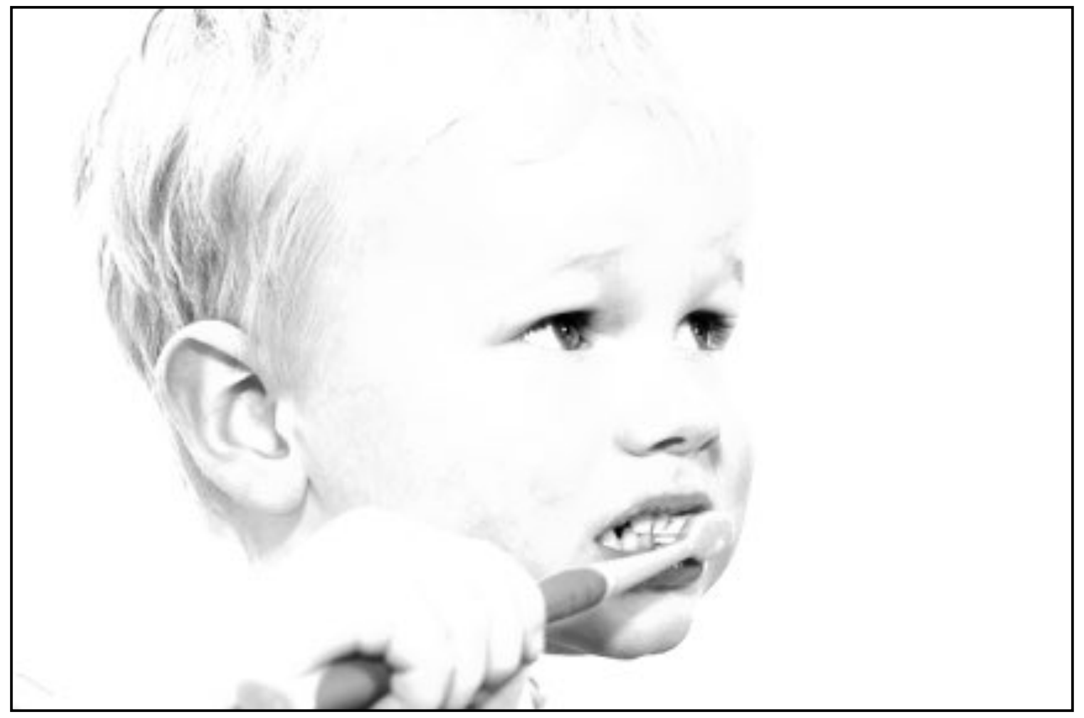
$255 - I(X,Y)$

lighten



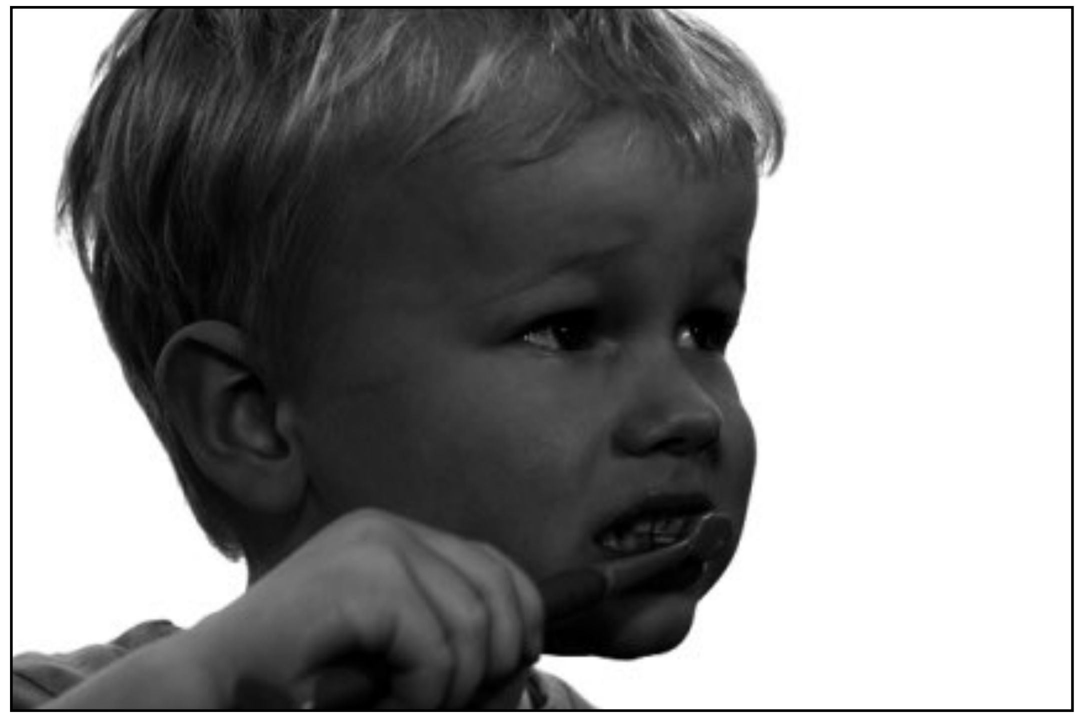
$I(X,Y) + 128$

raise contrast



$I(X,Y) \times 2$

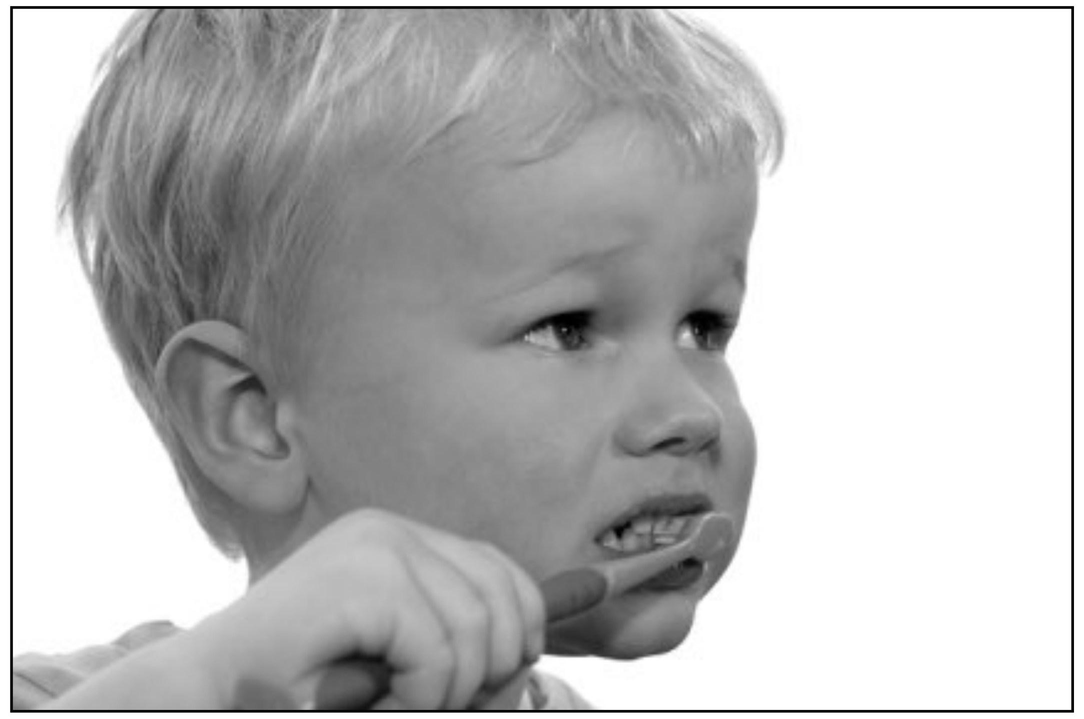
non-linear raise contrast



Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

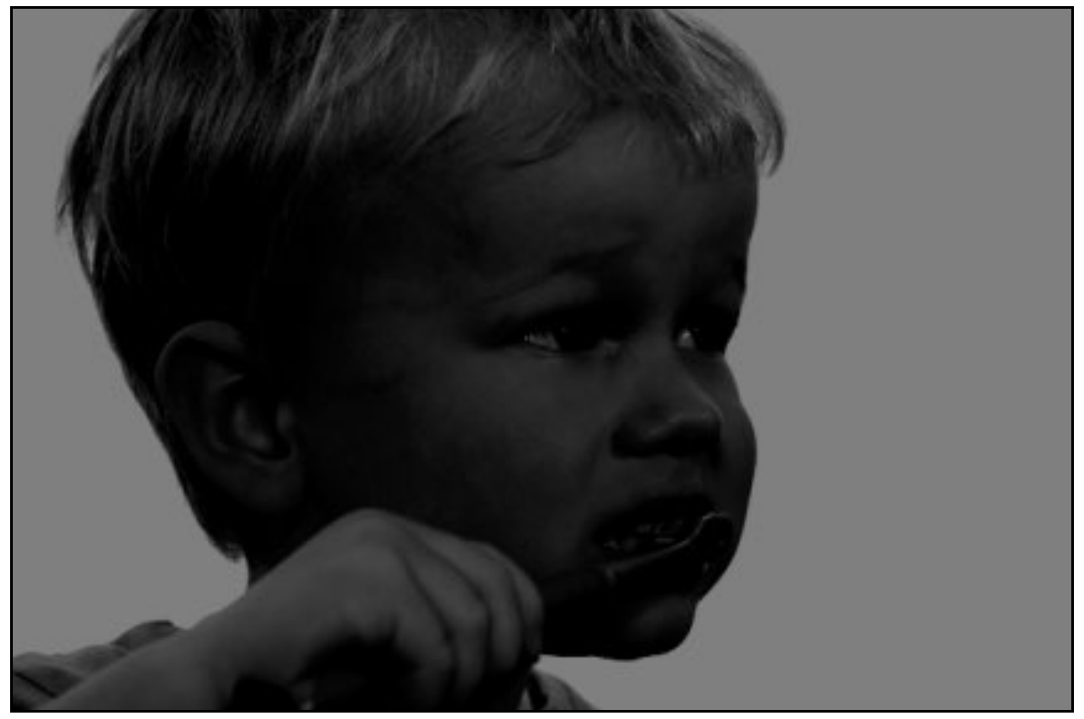
Examples of **Point Processing**

original



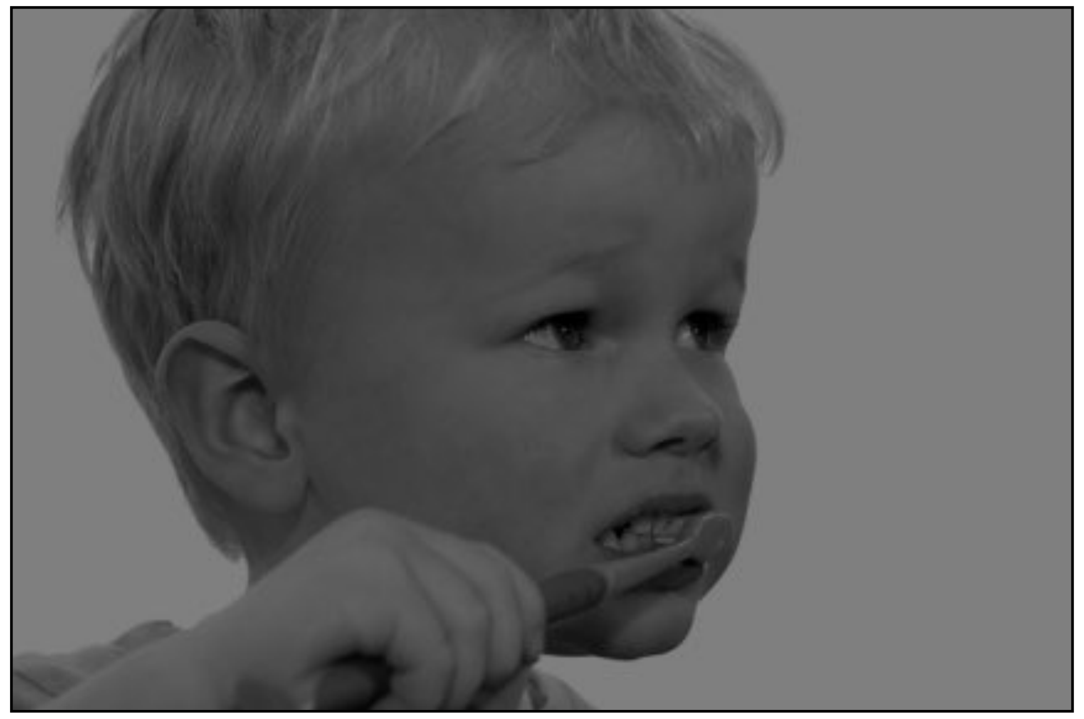
$I(X,Y)$

darken



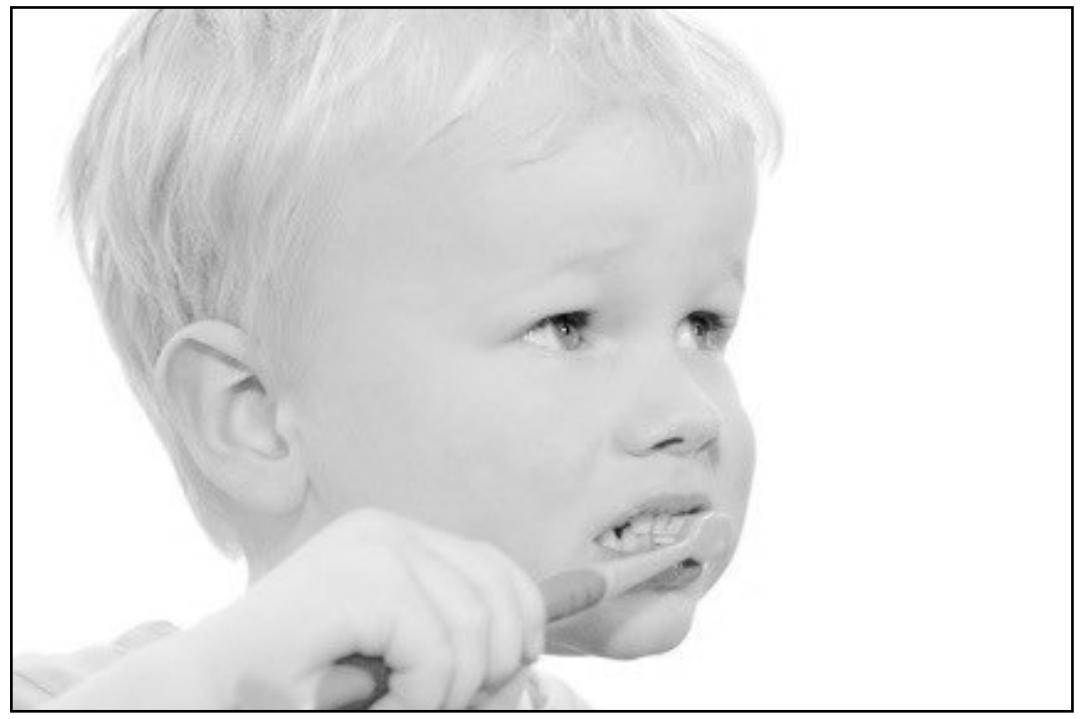
$I(X,Y) - 128$

lower contrast



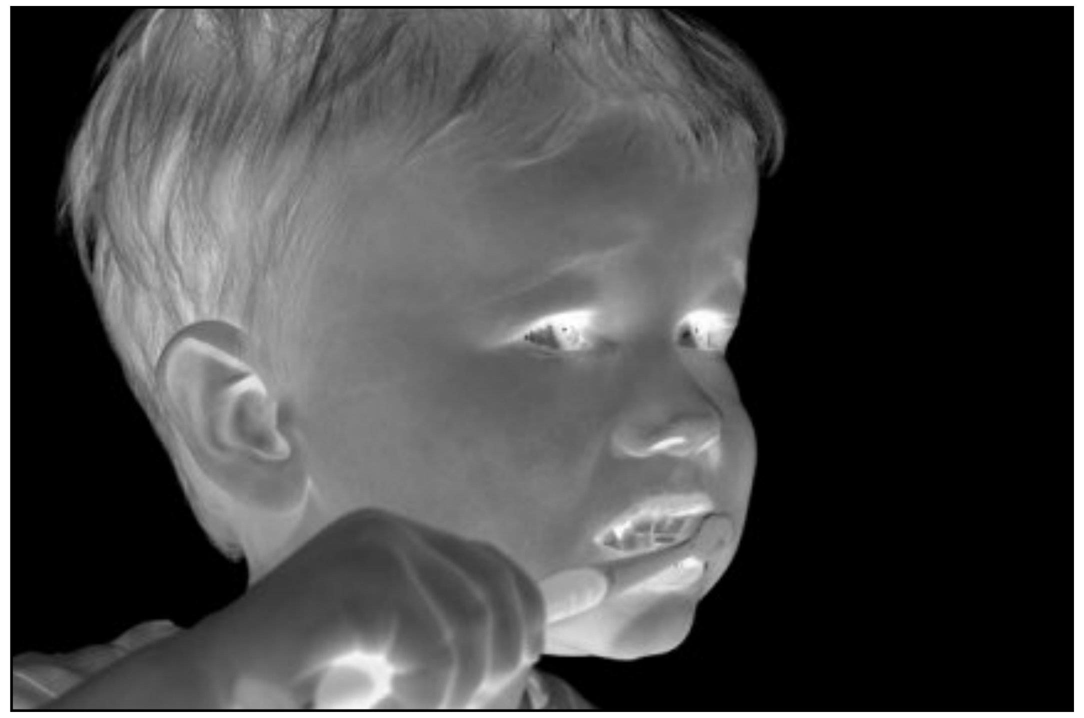
$\frac{I(X,Y)}{2}$

non-linear lower contrast



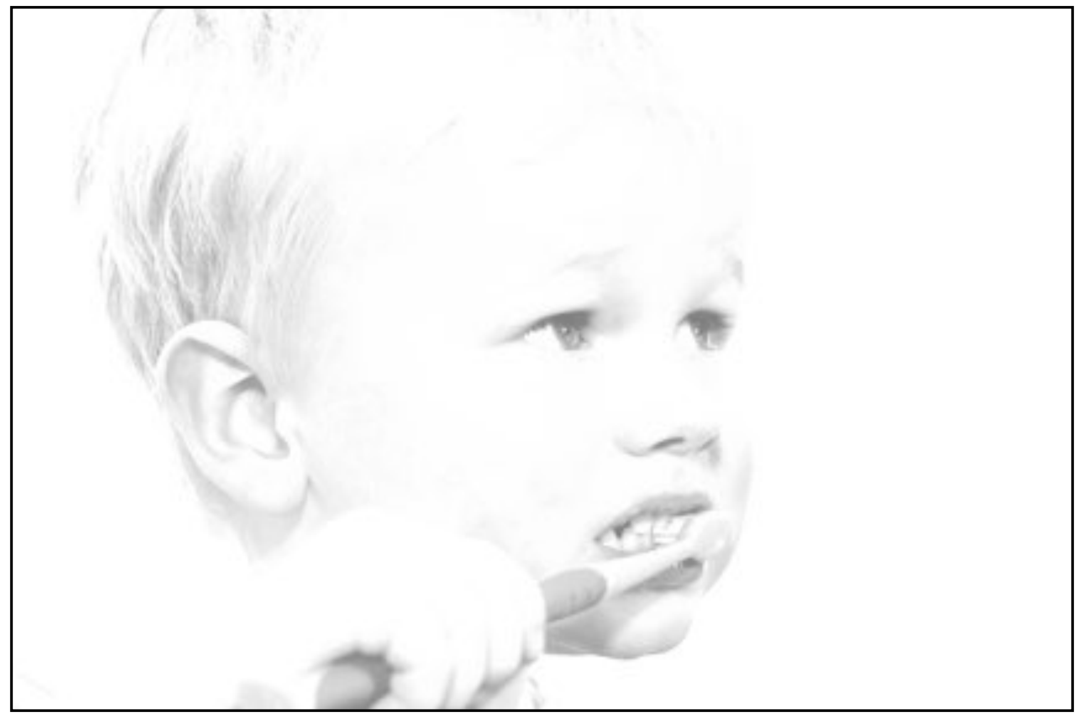
$\left(\frac{I(X,Y)}{255}\right)^{1/3} \times 255$

invert



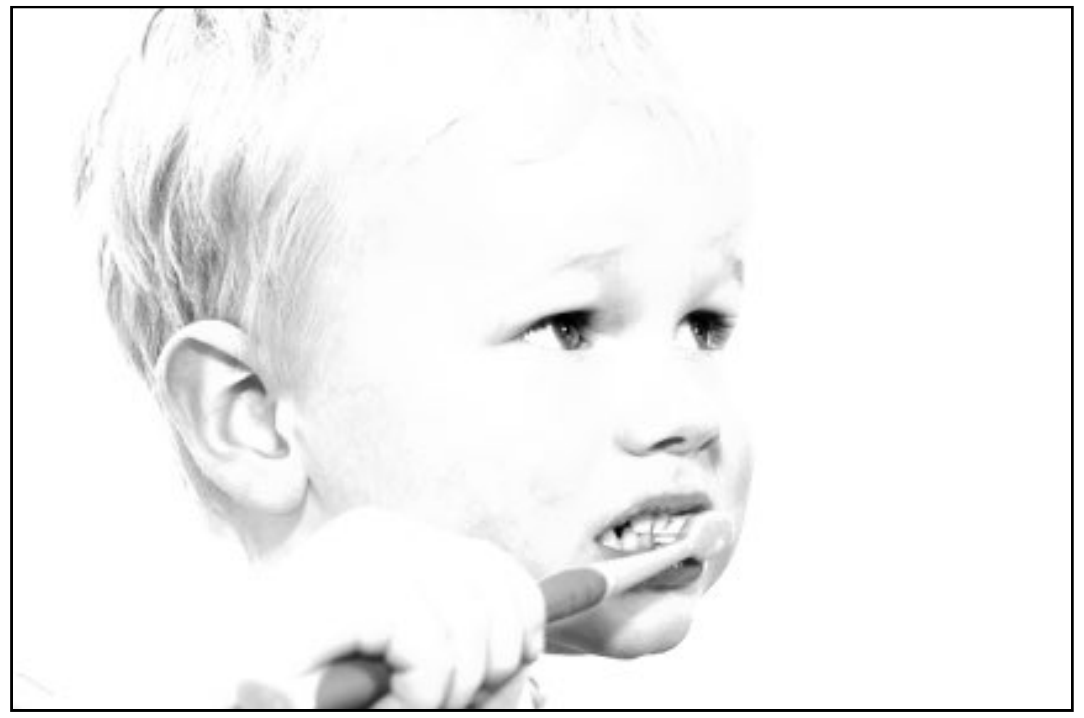
$255 - I(X,Y)$

lighten



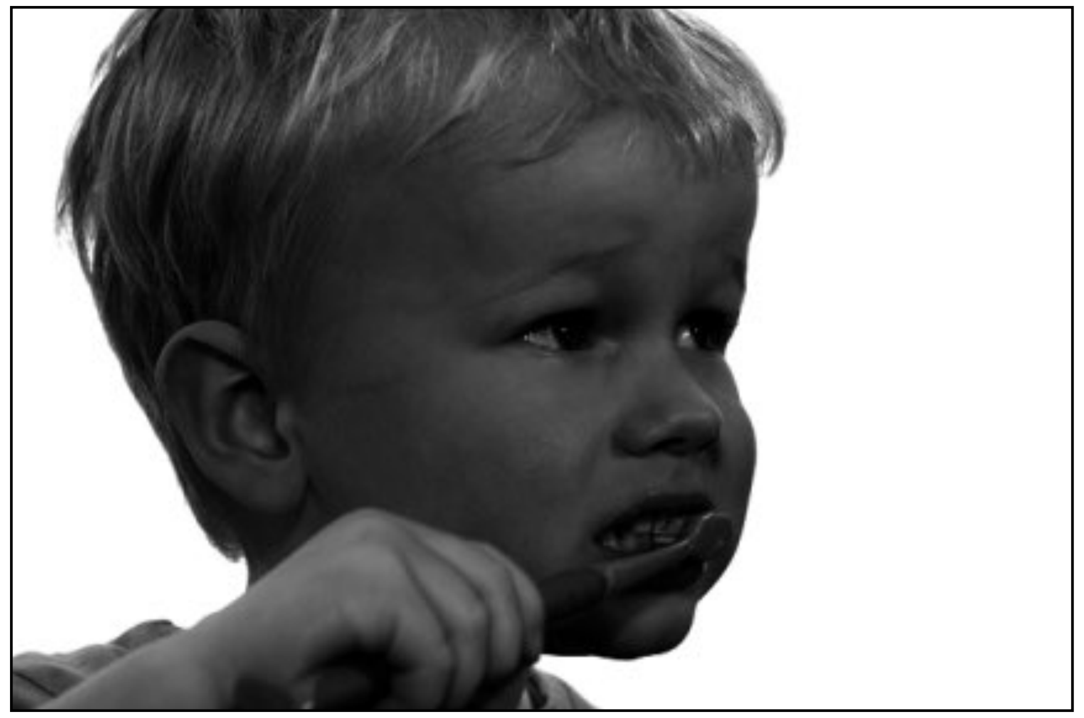
$I(X,Y) + 128$

raise contrast



$I(X,Y) \times 2$

non-linear raise contrast



$\left(\frac{I(X,Y)}{255}\right)^{2} \times 255$

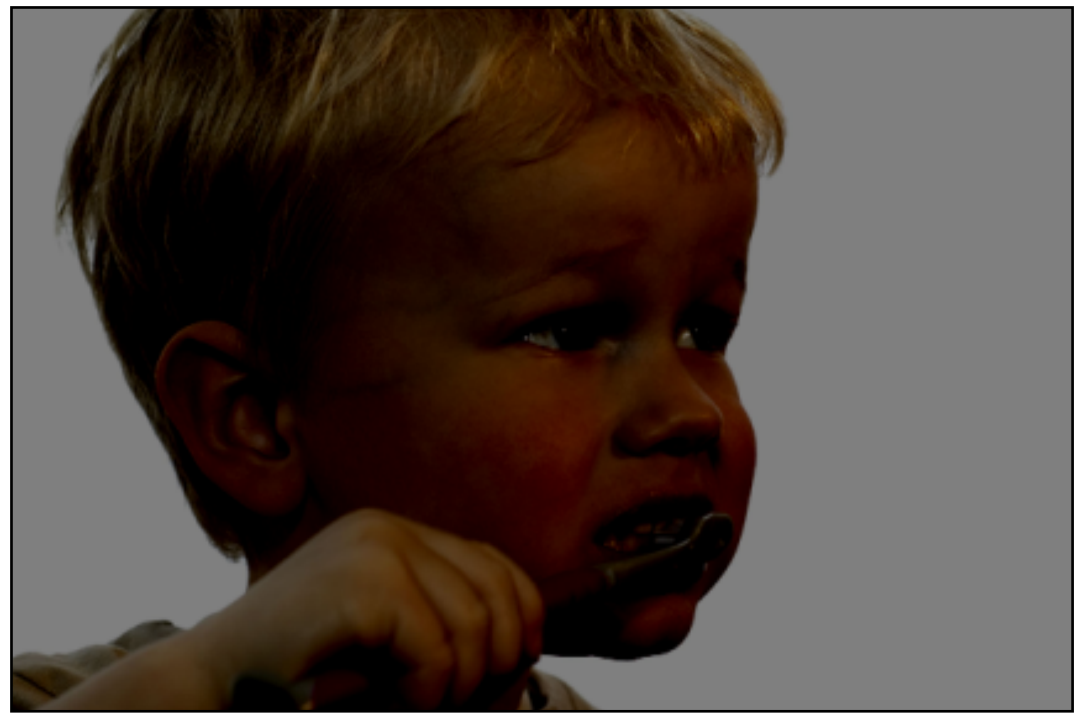
Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

Examples of **Point Processing**

original

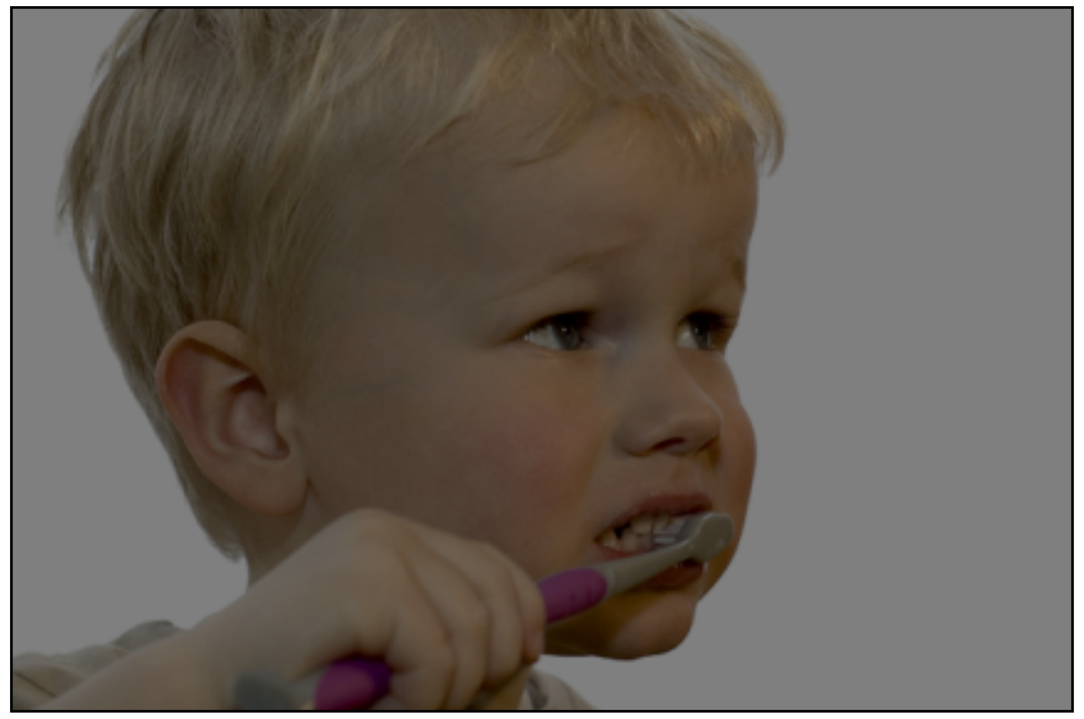
$$I(X,Y)$$

darken



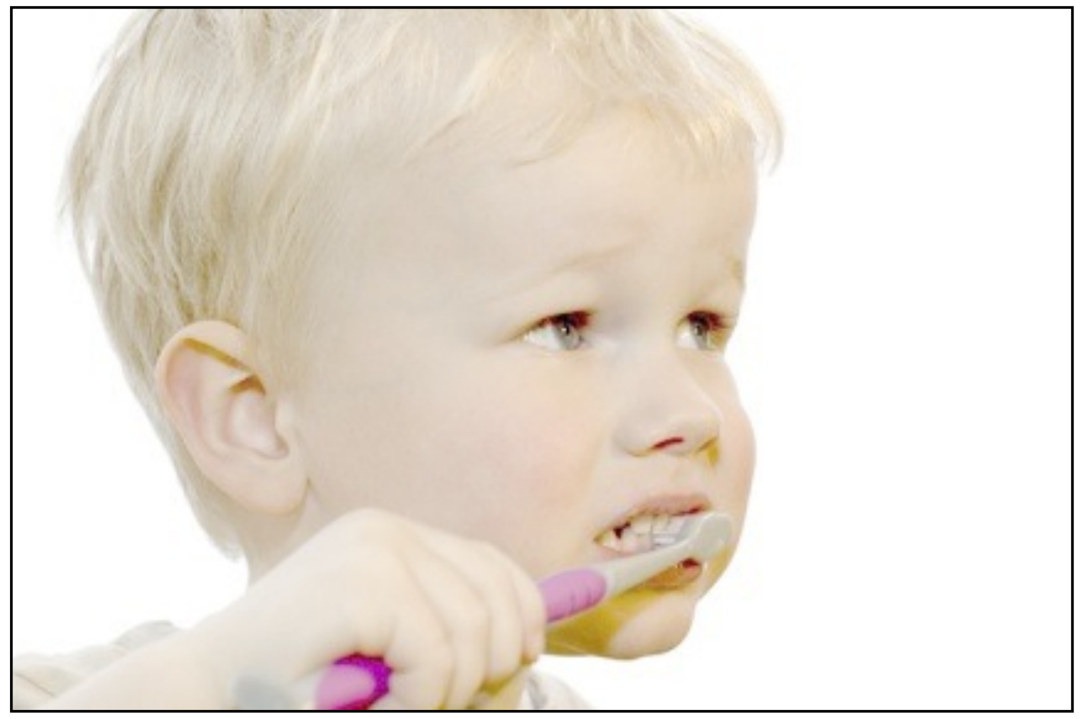
$$I(X,Y) - 128$$

lower contrast



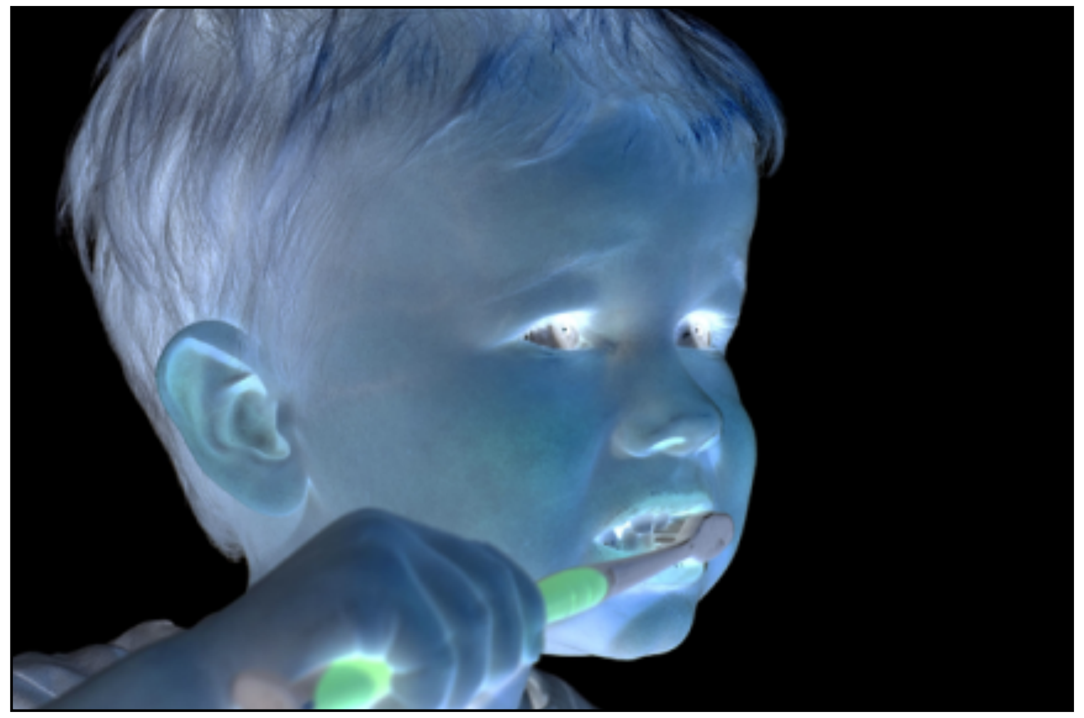
$$\frac{I(X,Y)}{2}$$

non-linear lower contrast



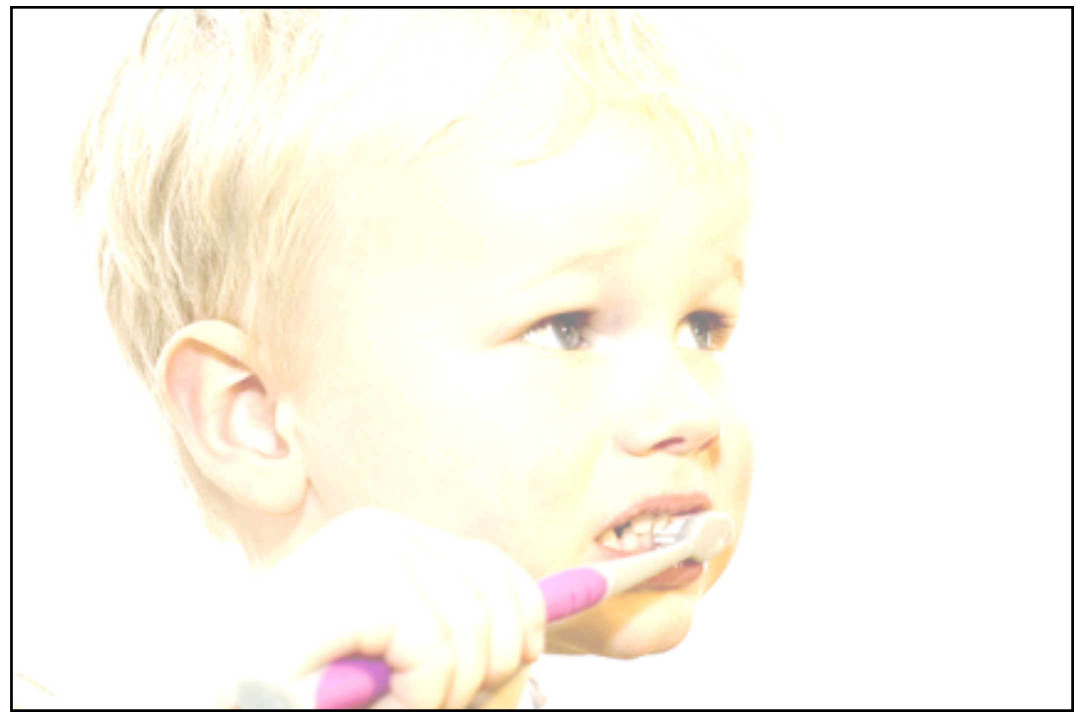
$$\left(\frac{I(X,Y)}{255}\right)^{1/3} \times 255$$

invert



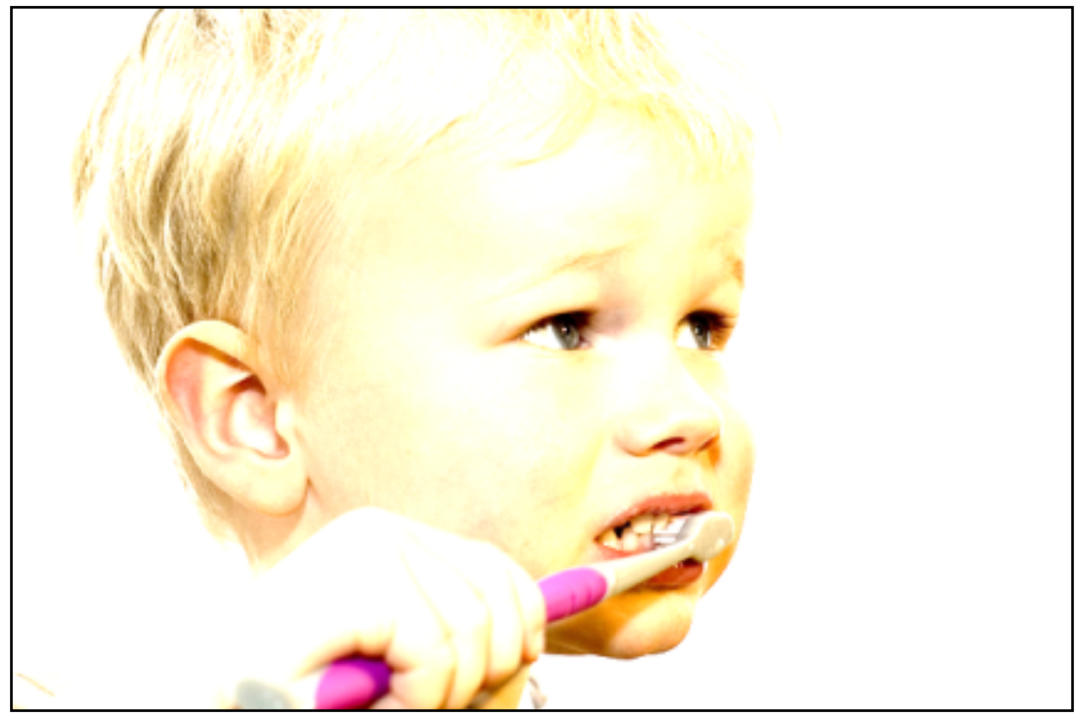
$$255 - I(X,Y)$$

lighten



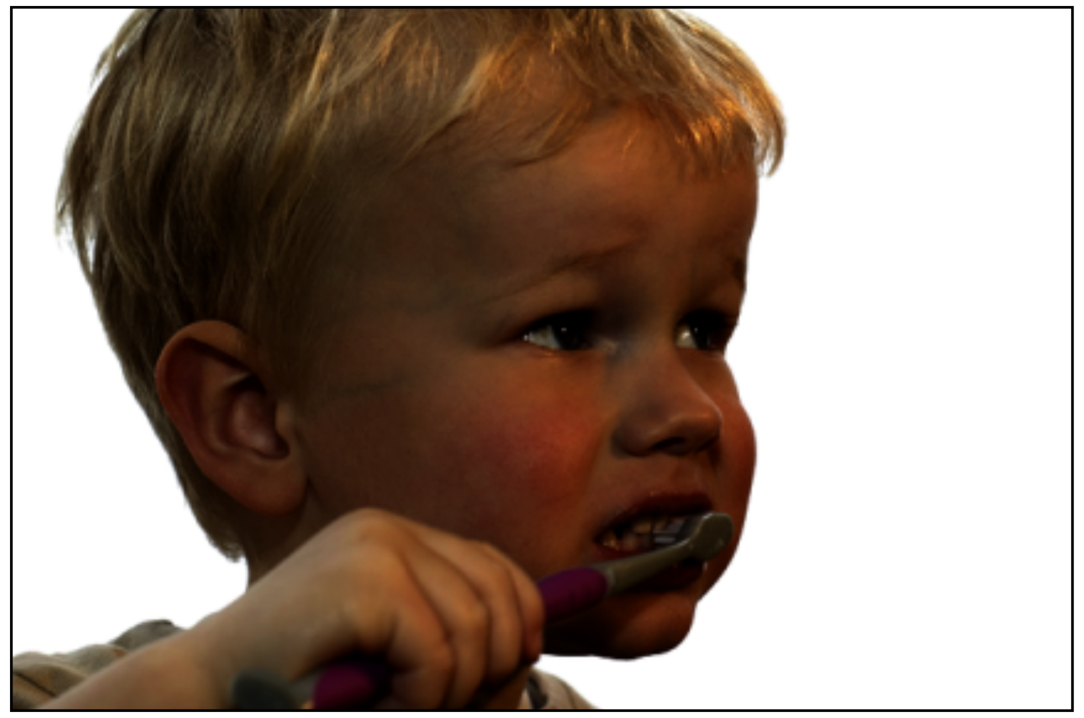
$$I(X,Y) + 128$$

raise contrast



$$I(X,Y) \times 2$$

non-linear raise contrast



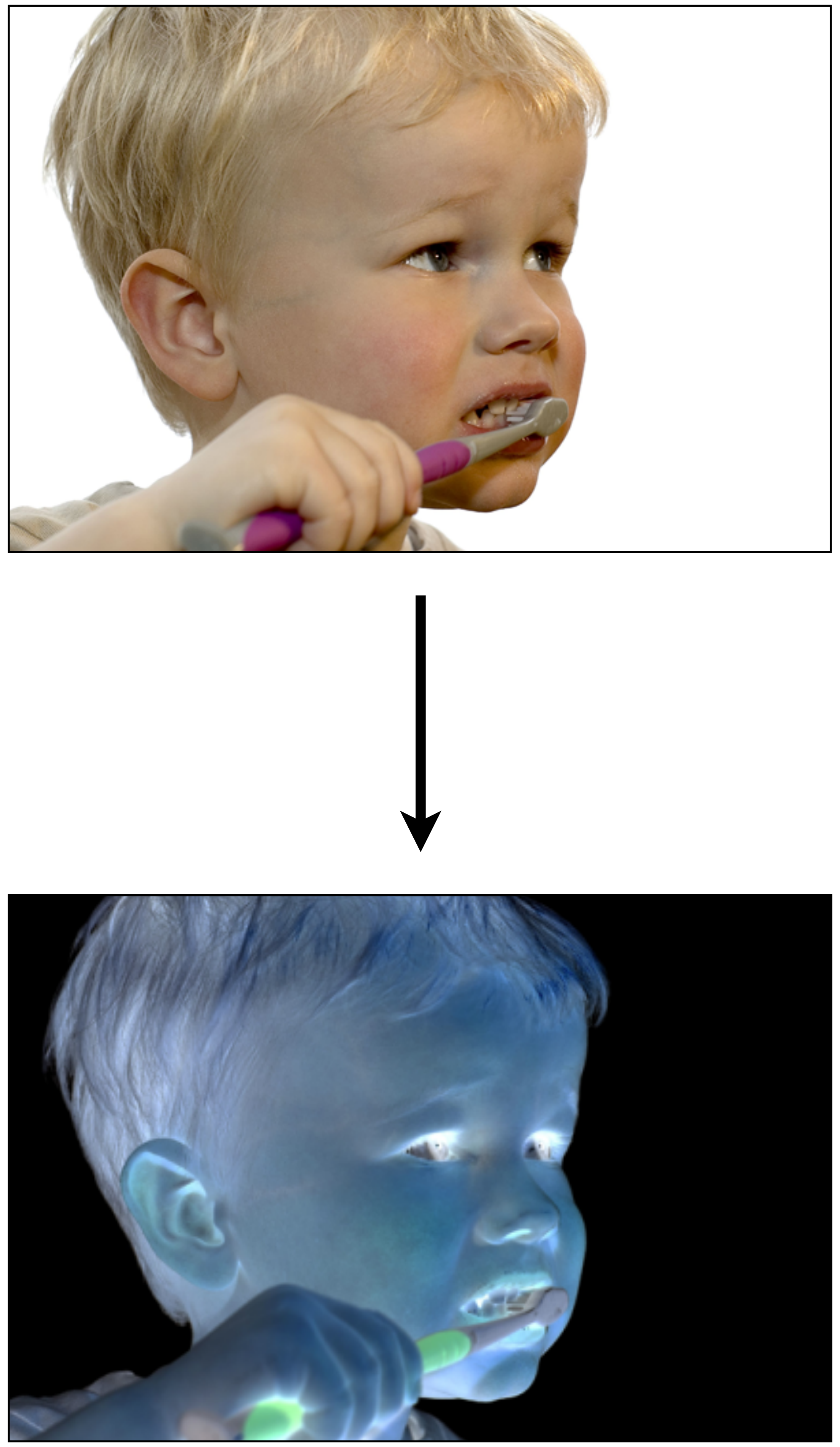
$$\left(\frac{I(X,Y)}{255}\right)^{2} \times 255$$

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

What types of **transformations** can we do?

$I(X,Y)$

Filtering

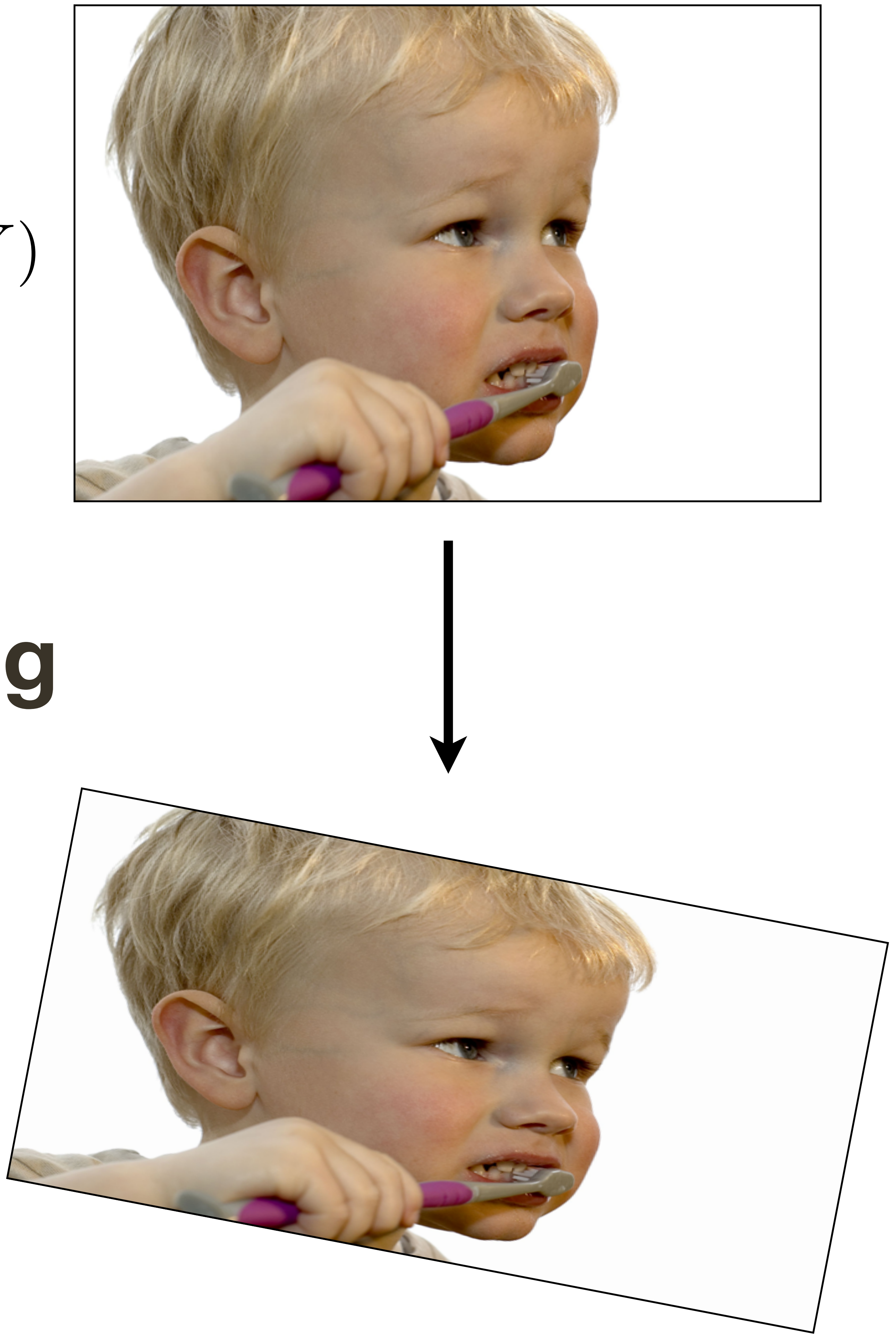


$I'(X,Y)$

changes range of image function

$I(X,Y)$

Warping



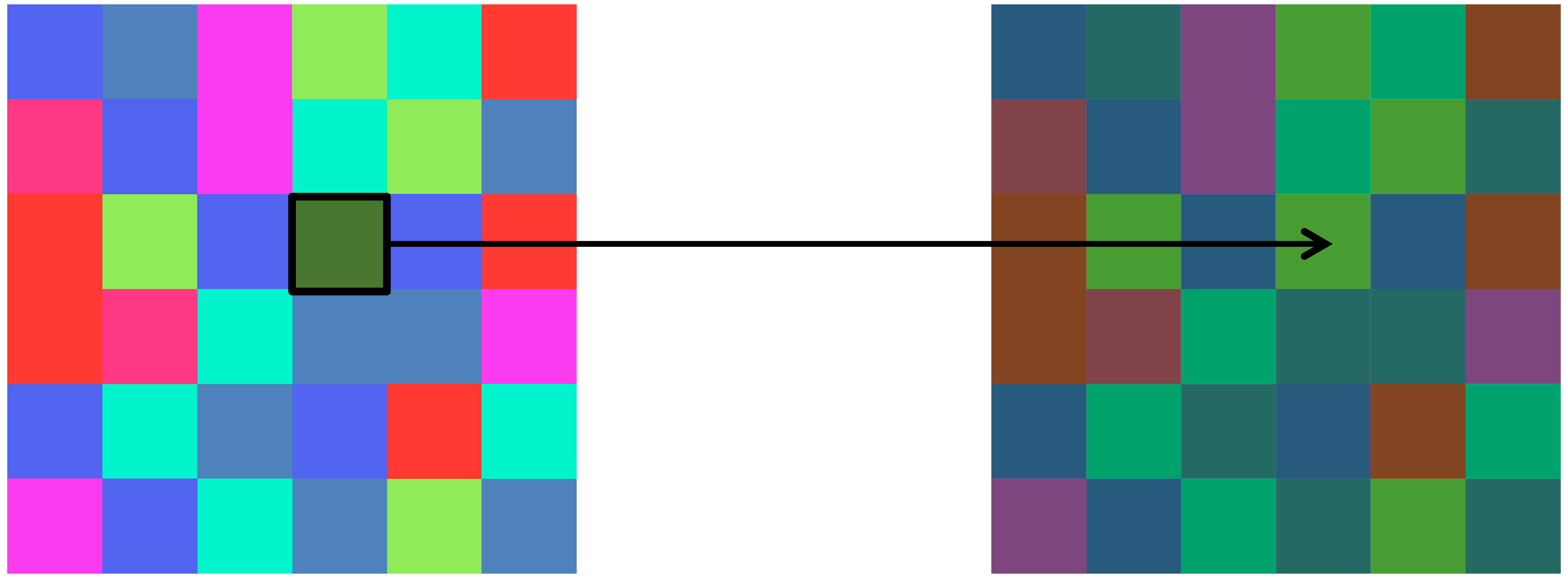
$I'(X,Y)$

changes domain of image function

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

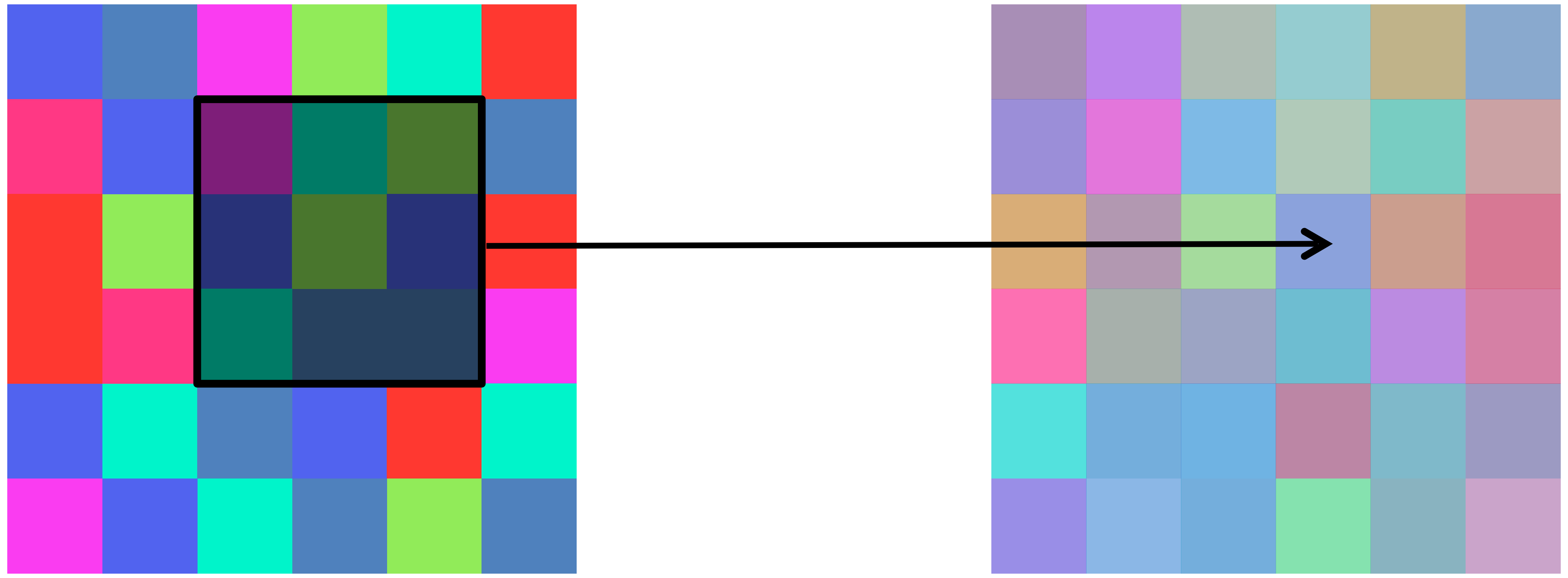
What types of **filtering** can we do?

Point Operation



point processing

Neighborhood Operation



“filtering”

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)