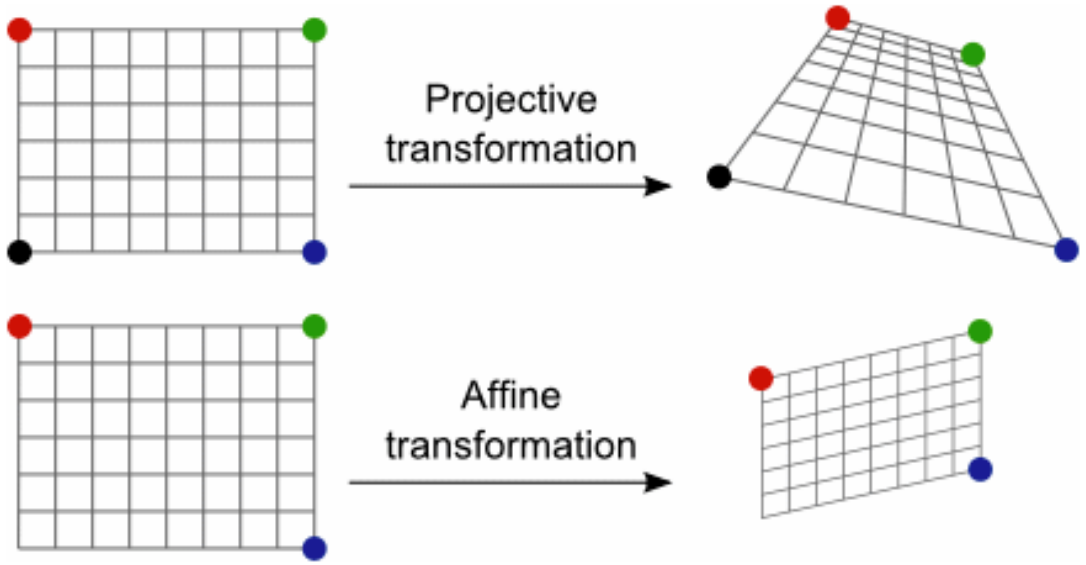
Core Operations

PABLO MALDONADO

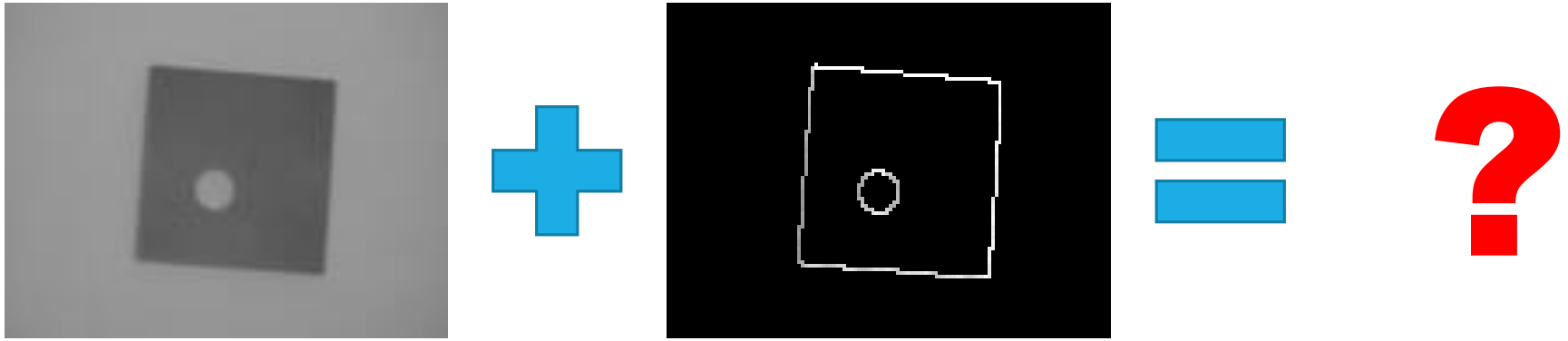
Geometric Transformations



https://www.graphicsmill.com/docs/gm/affine-and-projective-transformations.htm#DifferenceBetweenProjectiveAndAffine

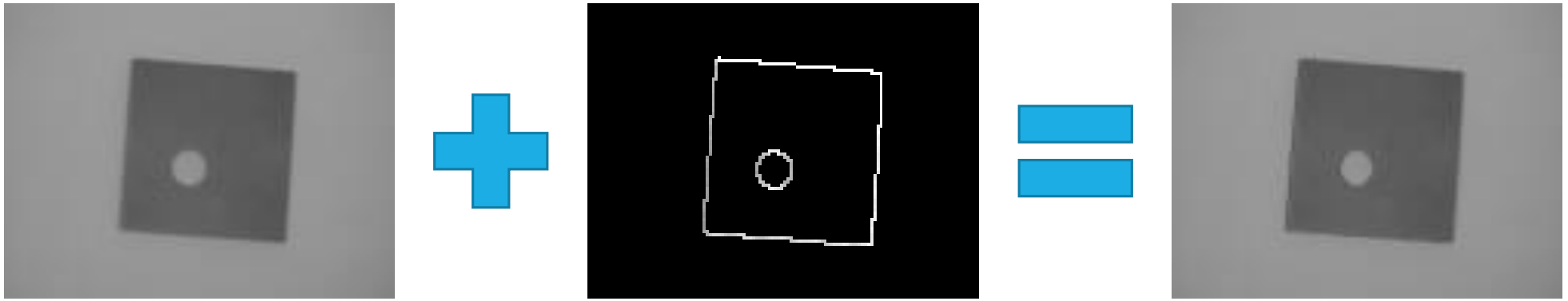
Image arithmetic

Image addition



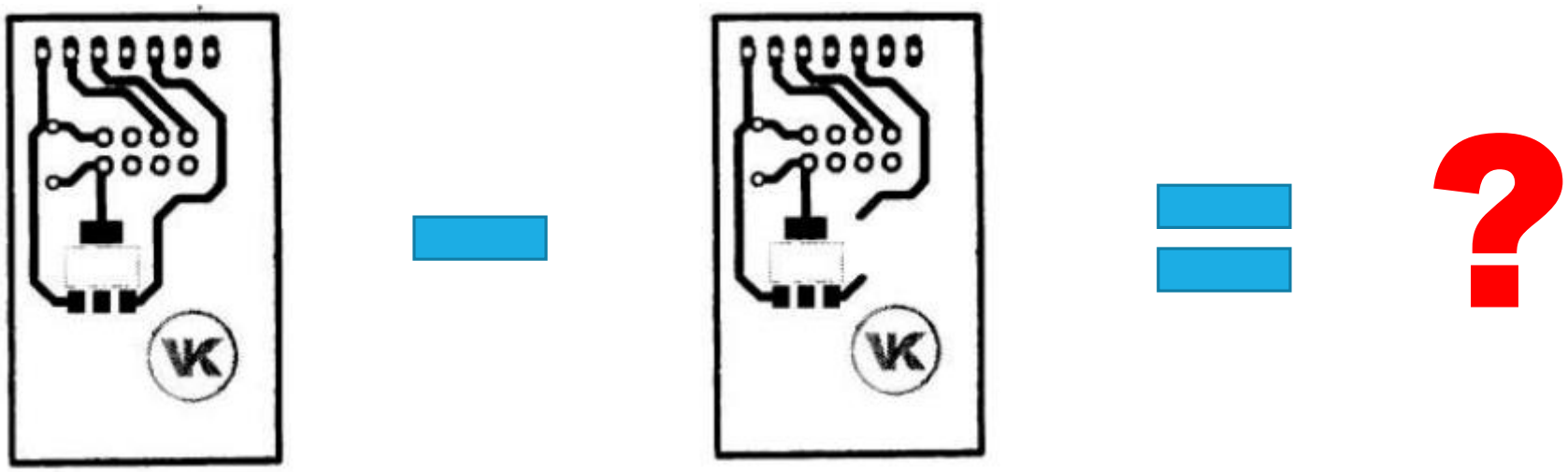
https://homepages.inf.ed.ac.uk/rbf/HIPR2/pixadd.htm

Image addition



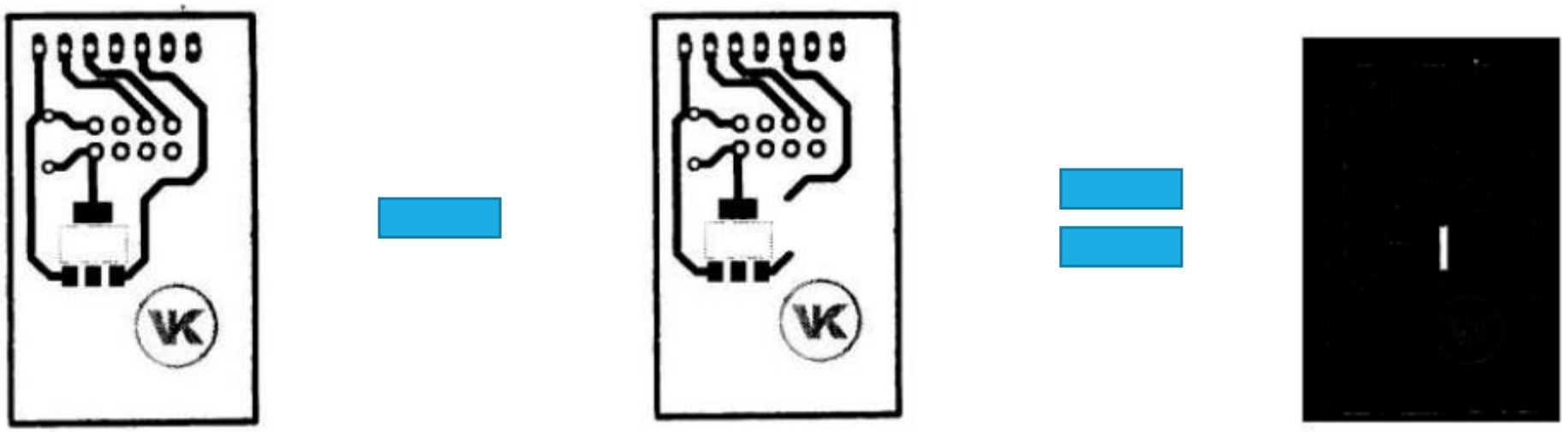
https://homepages.inf.ed.ac.uk/rbf/HIPR2/pixadd.htm

Image subtraction



https://www.electroschematics.com/10482/pcb-defects-detection-opencv/

Image subtraction



https://www.electroschematics.com/10482/pcb-defects-detection-opencv/

Thresholds

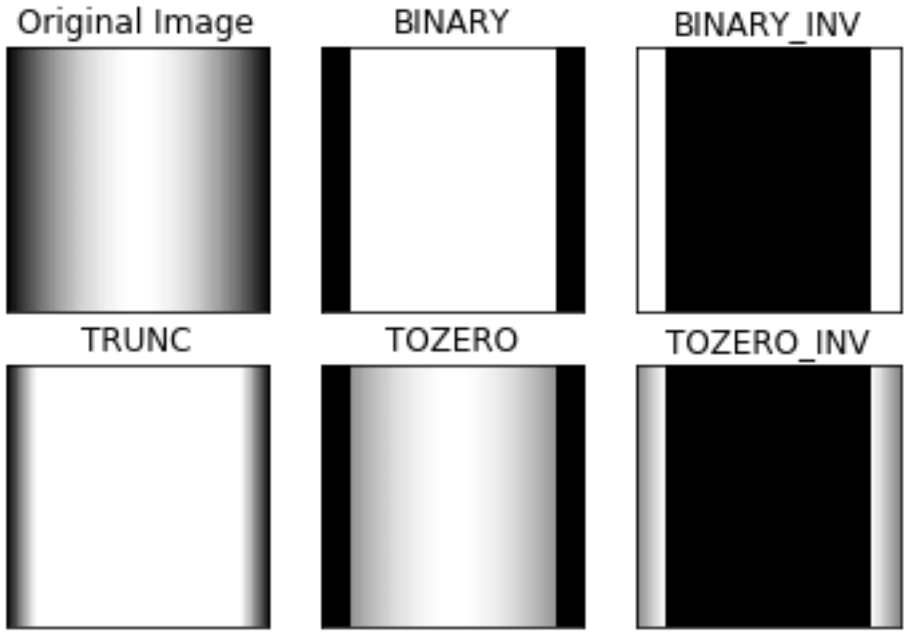
Different threshold operators

- **Threshold value = 127**

cv2.threshold(img,127,255,cv2.XXX)

- Issue: What happens with bimodal images?

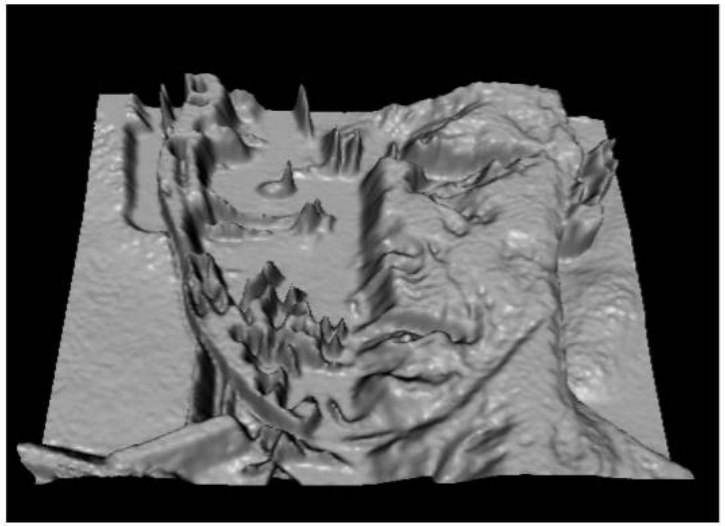
- Maybe we want a “smarter” solution

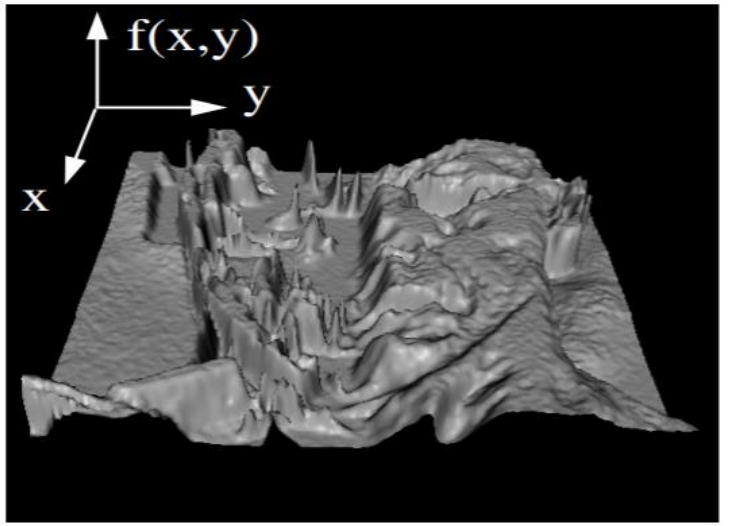


More transformations

Images as functions

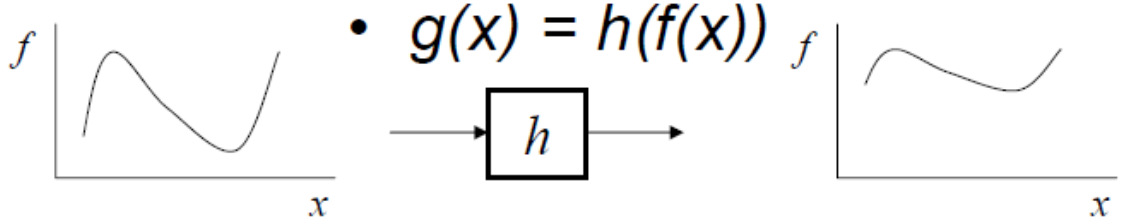
- We store an image as an array of vector values.
- However, it is morally a vector-valued **function** f(x,y) (real-valued function for gray images). Formally, it can be extended by interpolation to its full support (the *carrier*).
- Function composition ⇔ Image transformation.
- Function convolution ⇔ Apply a local filter.




f(x,y)
y
x

Function composition

- image filtering: change ***range*** of image
  - $g(x) = h(f(x))$



- image warping: change ***domain*** of image
  - $g(x) = f(h(x))$

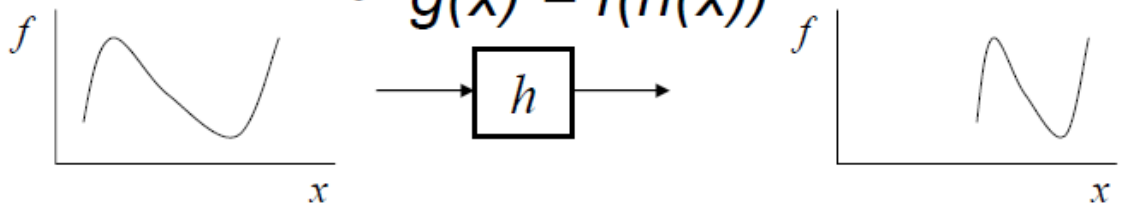
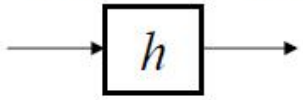
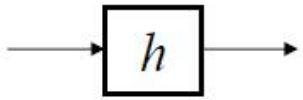


Image transformation

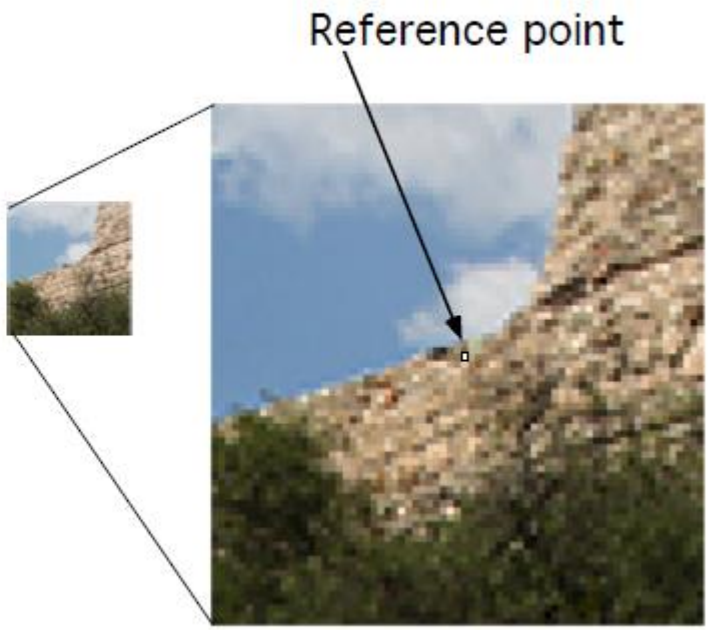
- image filtering: change ***range*** of image
  - $g(x) = h(f(x))$



- image warping: change ***domain*** of image
  - $g(x) = f(h(x))$



Window operations



R. Klette. Concise Computer Vision.

Noise filtering

Local filtering

- Replace each pixel by a linear combination of its neighbors.
- The prescription for the linear combination is called the “convolution kernel”.

10	5	3
4	5	1
1	1	7

Local image data

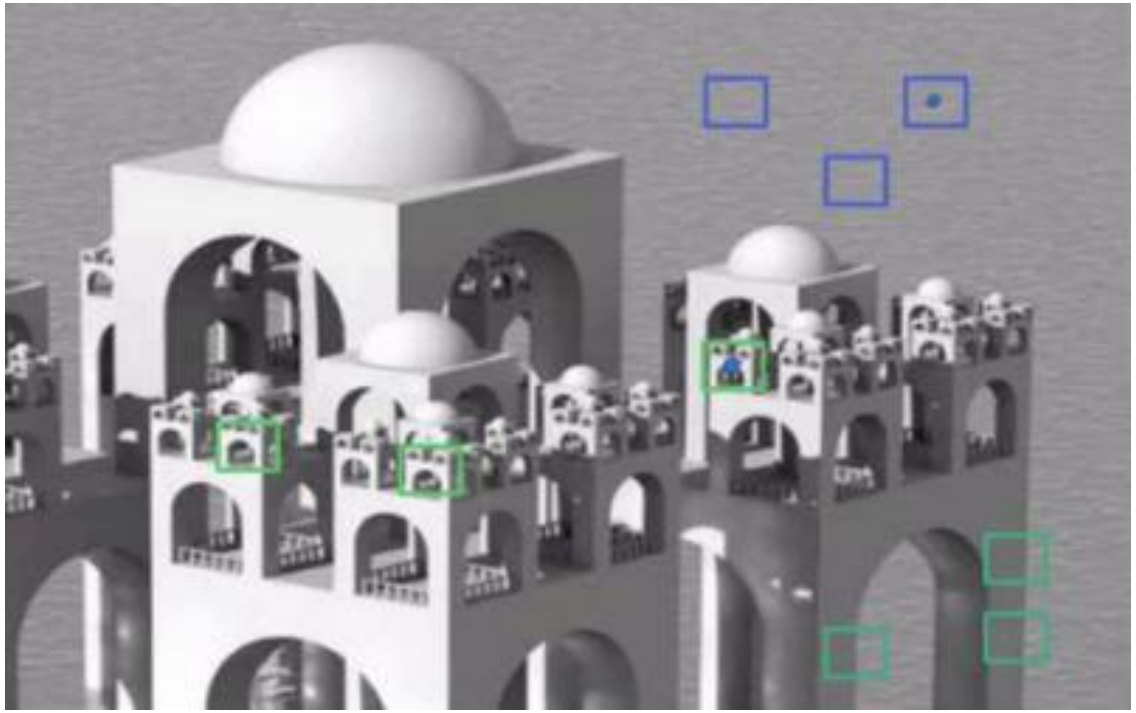
0	0	0
0	0.5	0
0	1	0.5

kernel

	7	

Modified image data
(shown at one pixel)

Beyond local noise



Exercise: PCB detection

- Download the PCBData from the course repository.

(original source: https://github.com/tangsanli5201/DeepPCB)

1. A simple way to detect PCB defects is by subtracting the template from the image. The difference would highlight the defective region.
2. How robust is this method if you add noise to the image?
3. Try this method with more realistic images:

https://www.electroschematics.com/10482/pcb-defects-detection-opencv/