AI + Image

NJU AIA

Wan Shenghua 2021.04.03

Content

- **What is image**
  - Image fundamentals
  - Image processing
  - Useful libraries
- **Some applications**
  - GeoAI
  - Medical image processing
- **AI + Image**
  - Convolutional Neural Networks
  - Some breakthourgh

Content

- **What is image**
  - Image fundamentals
  - Image processing
  - Useful libraries
- **Some applications**
  - GeoAI
  - Medical image processing
- **AI + Image**
  - Convolutional Neural Networks
  - Some breakthourgh

What is image?

- An image is a spatial representation of a two-dimensional or three-dimensional scene.
- An image is an array, or a matrix pixels (picture elements) arranged in columns and rows.

Image acquisition

Image acquisition is the first process.

Generally, the image acquisition stage involves preprocessing, such as scaling.

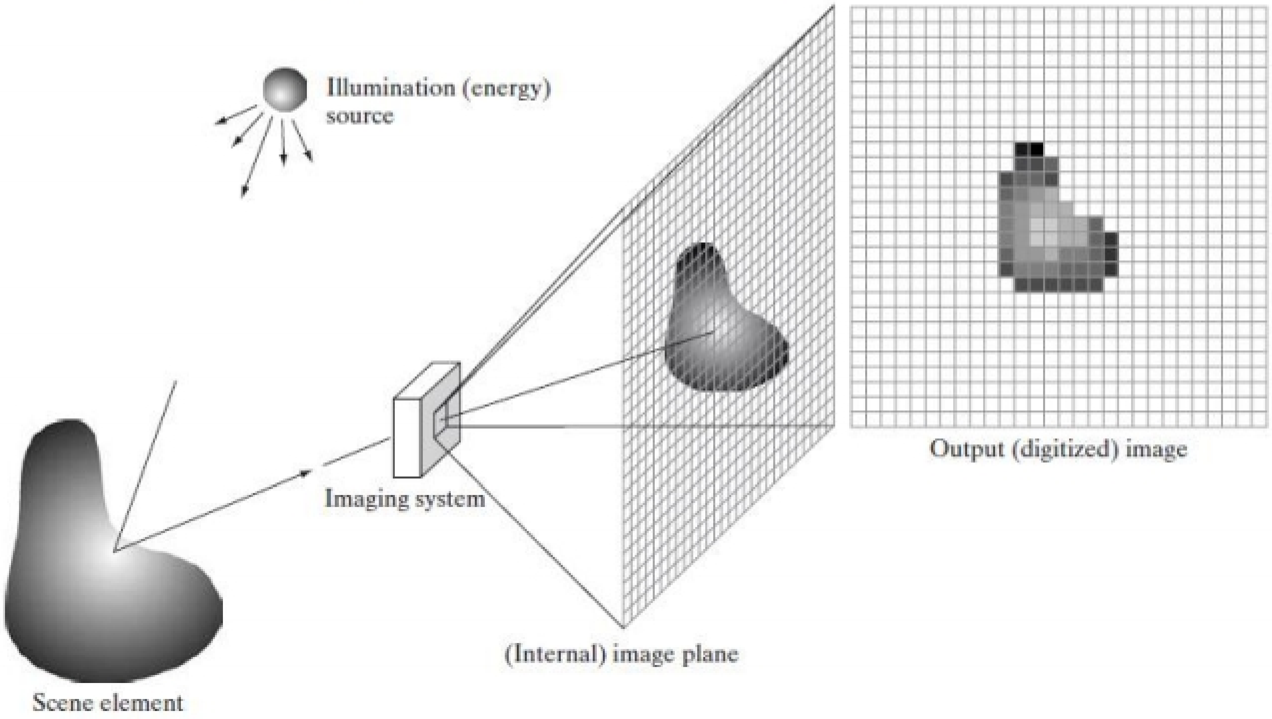
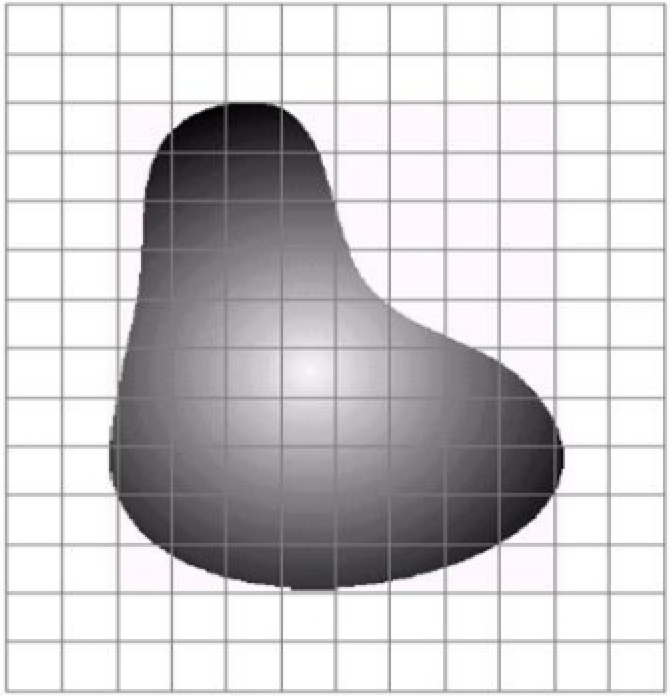


Image fundamentals

- Remember that a digital image is always only an **approximation** of a real world scene.



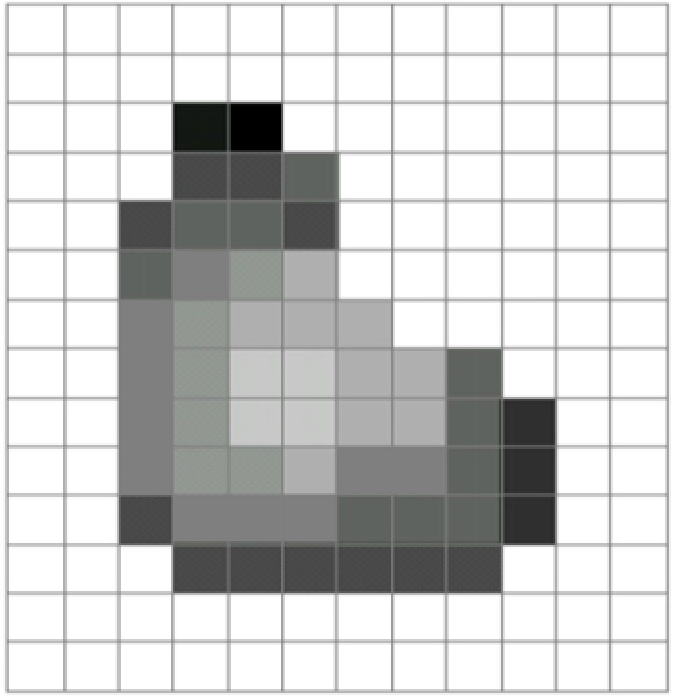


Image fundamentals

- An image may be defined as a two-dimensional function, **f(x, y)**, where **x** and **y** are spatial (plane) coordinates, and the amplitude of **f** at any pair of coordinates (x,y) is called the **intensity** or **gray level** of the image at that point.

- **Digital Image:**
  When x, y and the intensity values of f are all finite, discrete quantities, we call the image a digital image.

- **Color Image:**

$$f(x,y) = \begin{bmatrix} r(x,y) \\ g(x,y) \\ b(x,y) \end{bmatrix}$$

Image fundamentals

An Image:

$g(x, y)$

Discretization

$g(i, j)$

Quantization

$f(i, j)$ Digital Image

$f(i_0, j_0)$: Picture Element, Image Element, Pel, Pixel

Content

- **What is image**
  - Image fundamentals
  - **Image processing**
  - Useful libraries
- **Some applications**
  - GeoAI
  - Medical image processing
- **AI + Image**
  - Convolutional Neural Networks
  - Some breakthourgh

What is image processing

What is digital image processing

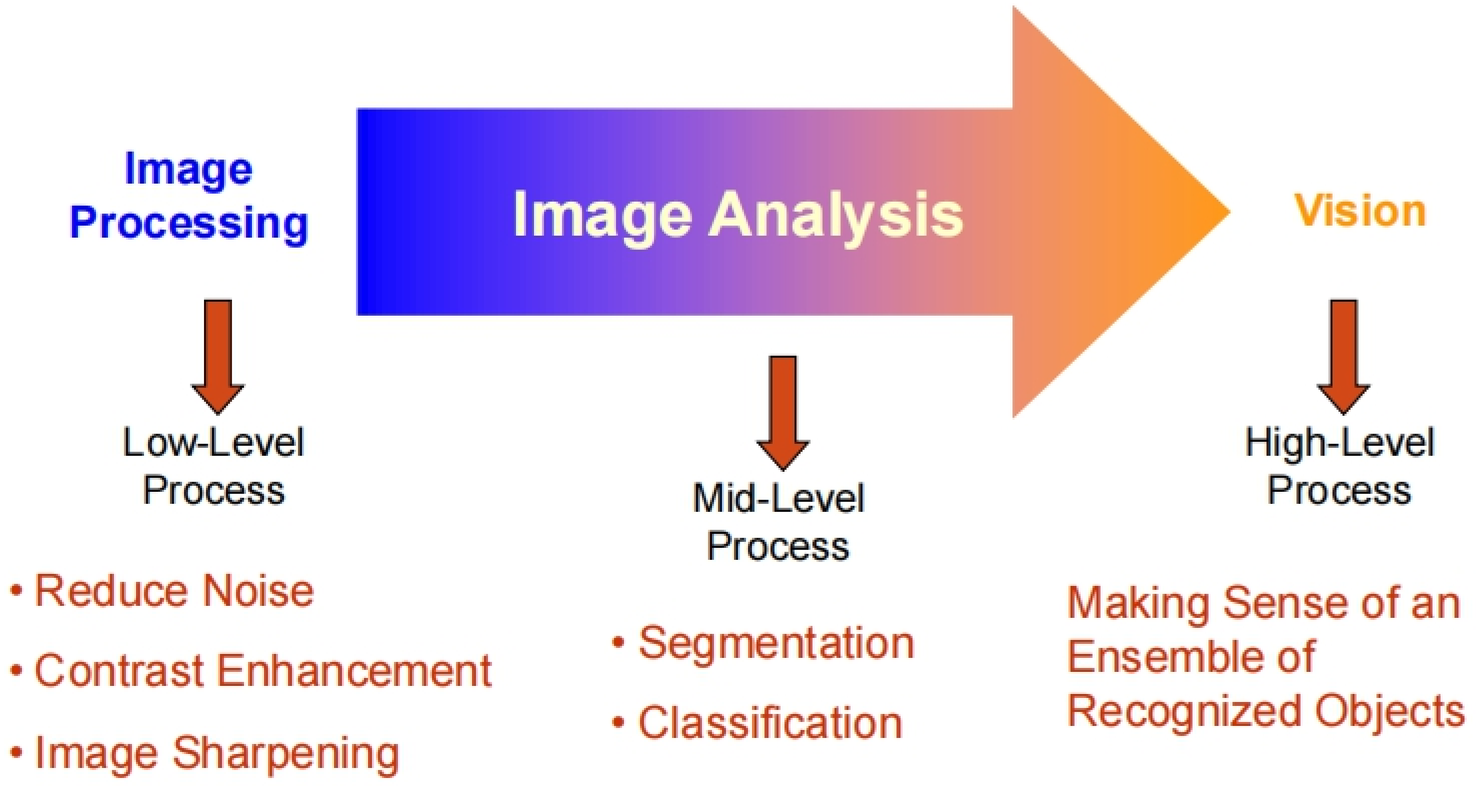


Image processing

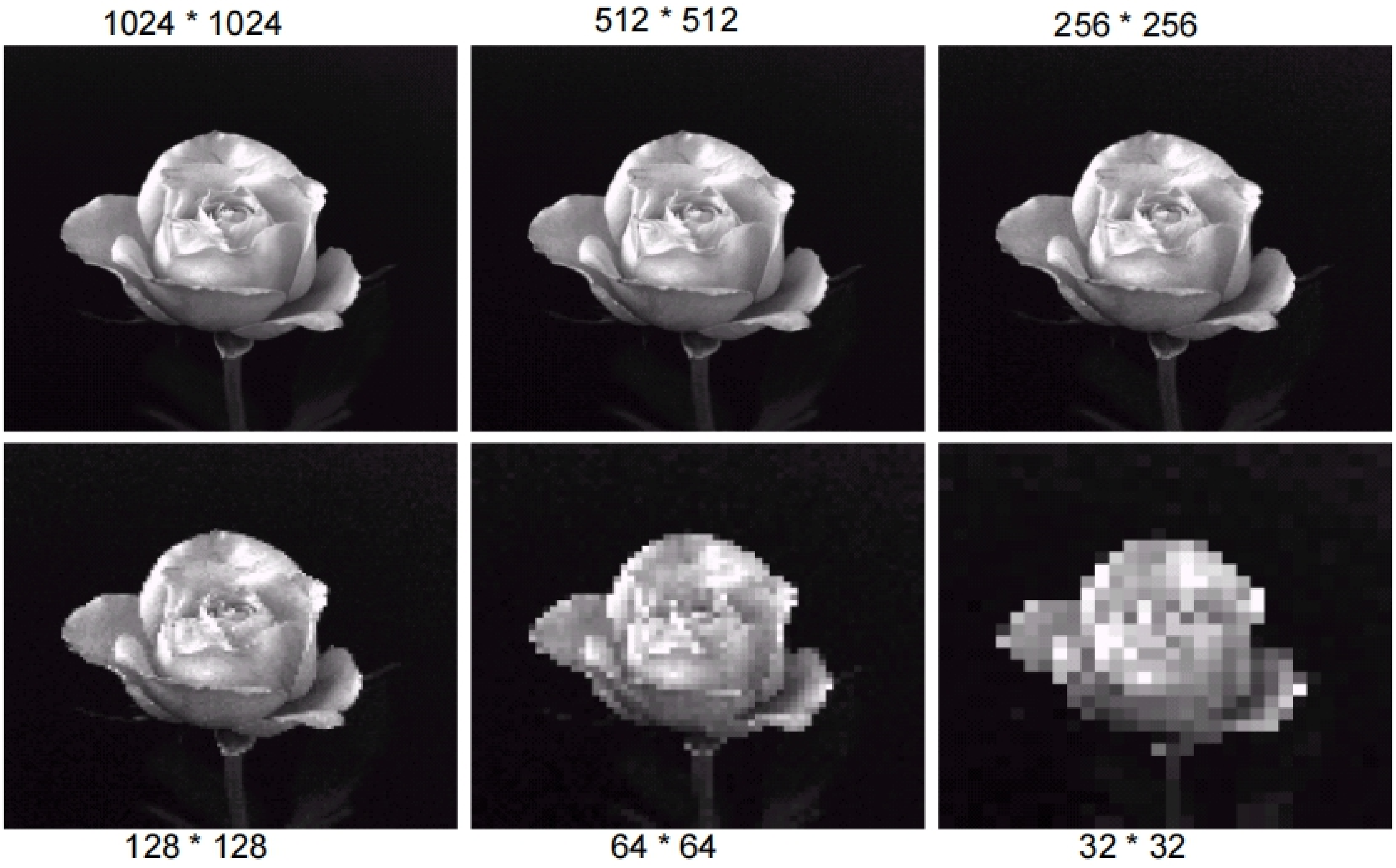
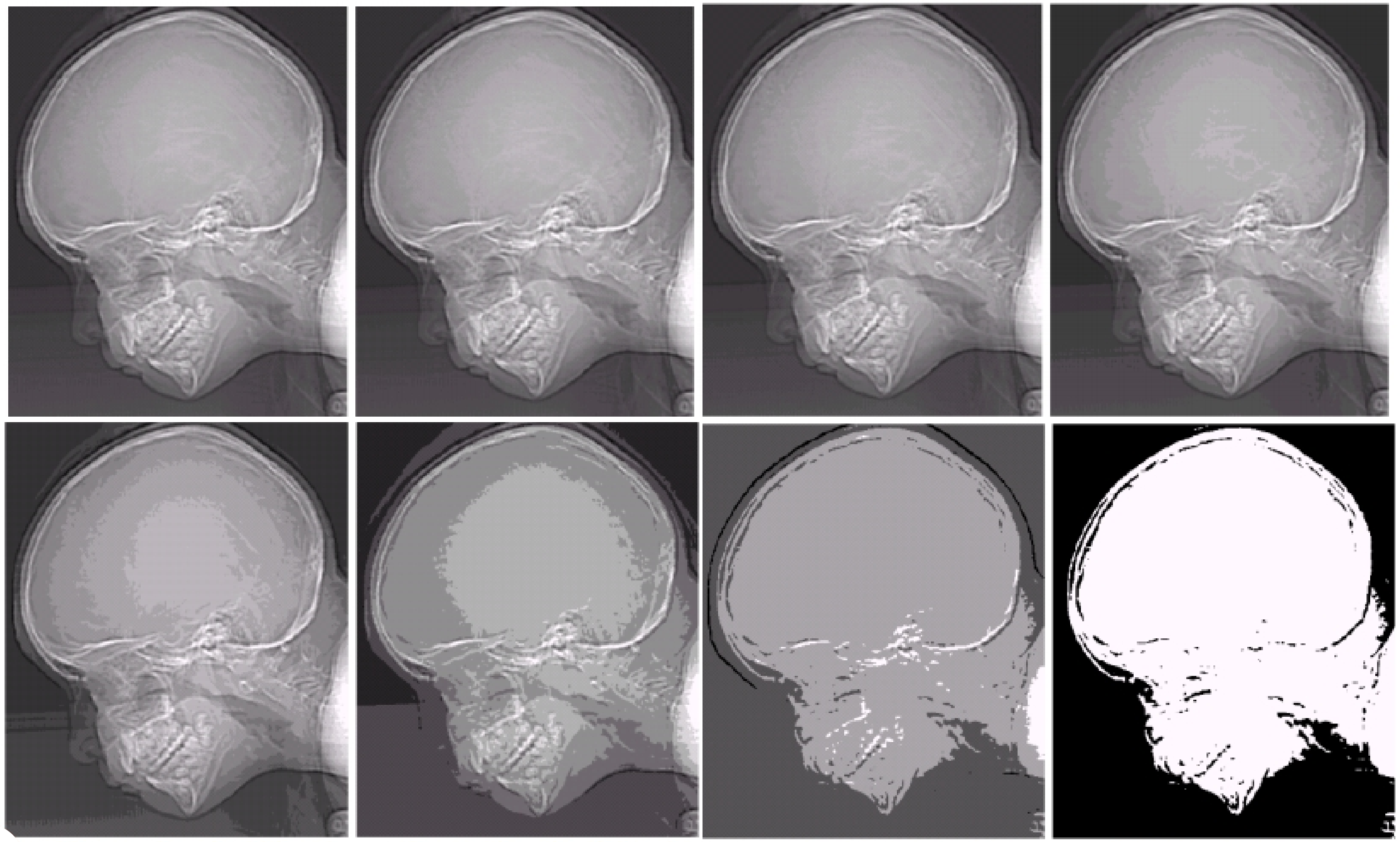


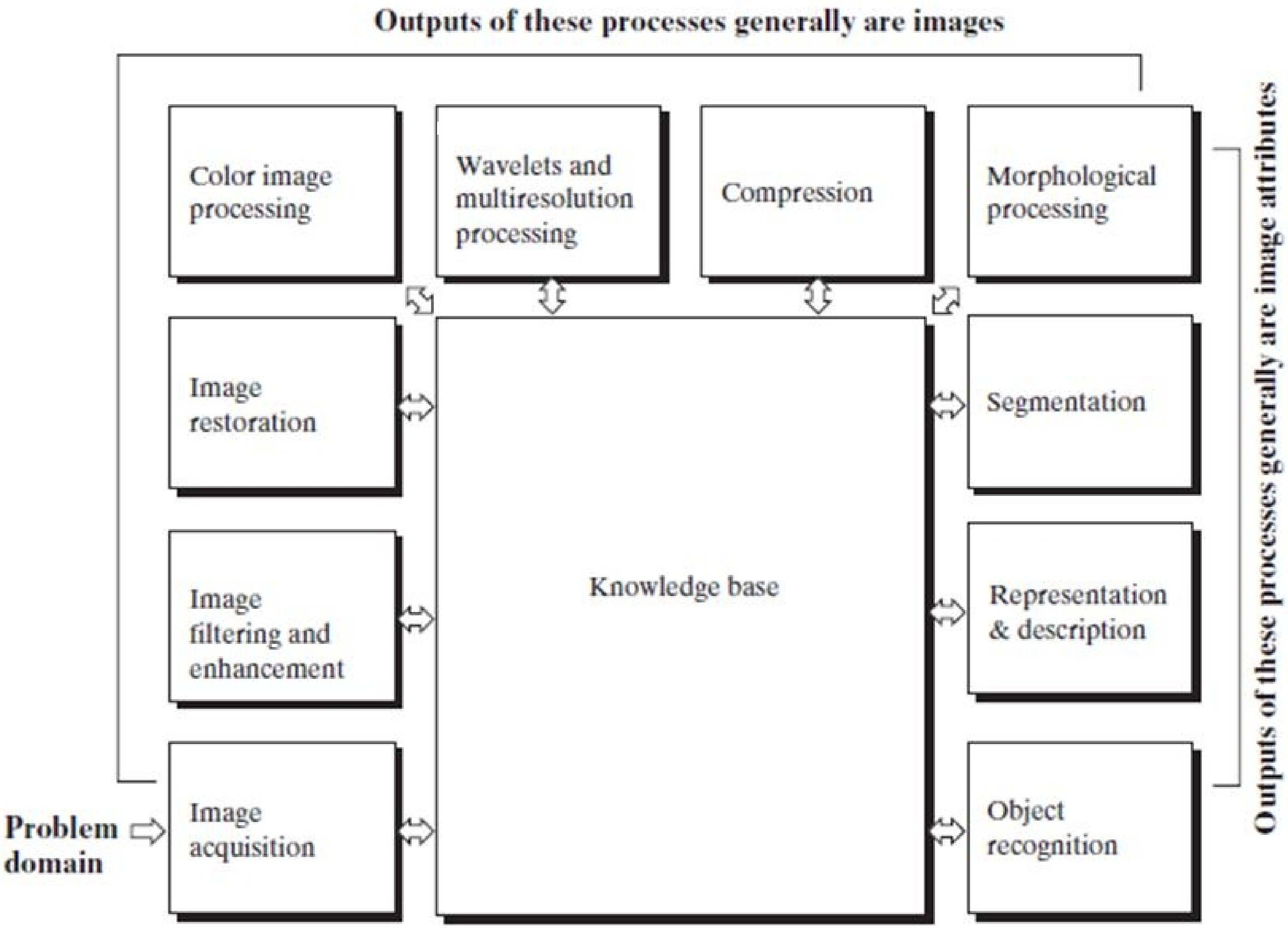
Image processing

256 grey levels (8 bits per pixel) 128 grey levels (7 bpp) 64 grey levels (6 bpp) 32 grey levels (5 bpp)



16 grey levels (4 bpp) 8 grey levels (3 bpp) 4 grey levels (2 bpp) 2 grey levels (1 bpp)

Image processing



Content

- What is image
  - Image fundamentals
  - Image processing
  - Useful libraries
- Some applications
  - GeoAI
  - Medical image processing
- AI + Image
  - Convolutional Neural Networks
  - Some breakthourgh

Useful libraries

pip install opencv-python

https://opencv.org/

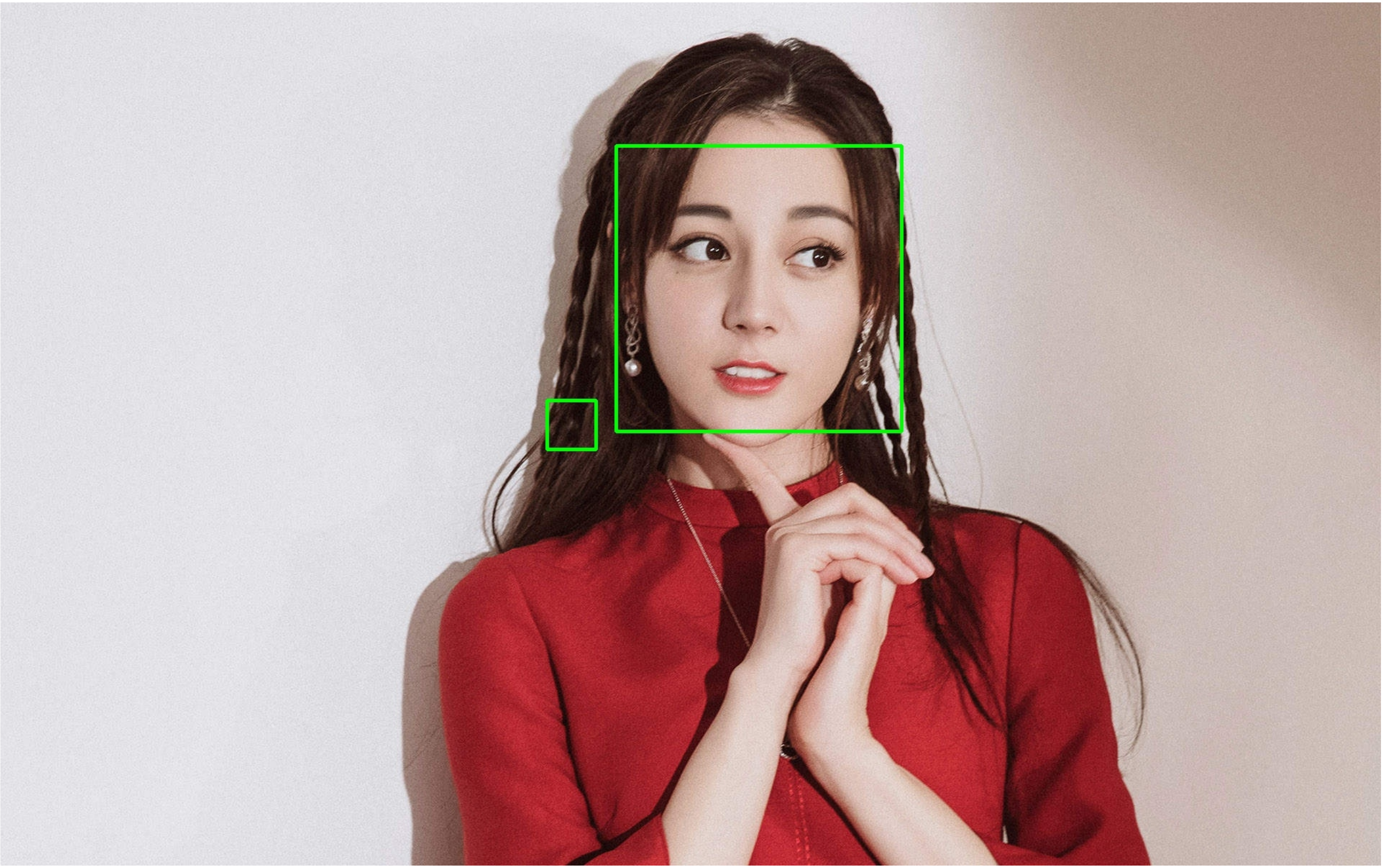
1. loading images

```
import cv2

 # colored Image
Img_rgb = cv2.imread ("Penguins.jpg",1)

 # Black and White (gray scale)
Img_gray = cv2.imread ("Penguins.jpg",0)

 # show image
cv2.imshow("Penguins", img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

2. Face detection

```
import cv2

# Create a CascadeClassifier Object
face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade_frontalface_default.xml')
# Reading the image as it is
img = cv2.imread("mmexport1550811773146.jpg")
# Reading the image as gray scale image
gray_img = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
# Search the co-ordintes of the image
faces = face_cascade.detectMultiScale(gray_img, scaleFactor = 1.05,  minNeighbors=5)
for x,y,w,h in faces:
    img = cv2.rectangle(img, (x,y), (x+w,y+h),(0,255,0),3)
resized = cv2.resize(img, (int(img.shape[1]/7),int(img.shape[0]/7)))
cv2.imshow("Gray", resized)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Useful libraries



Useful libraries

pip install Pillow

https://python-pillow.org/

1. loading and rotating images

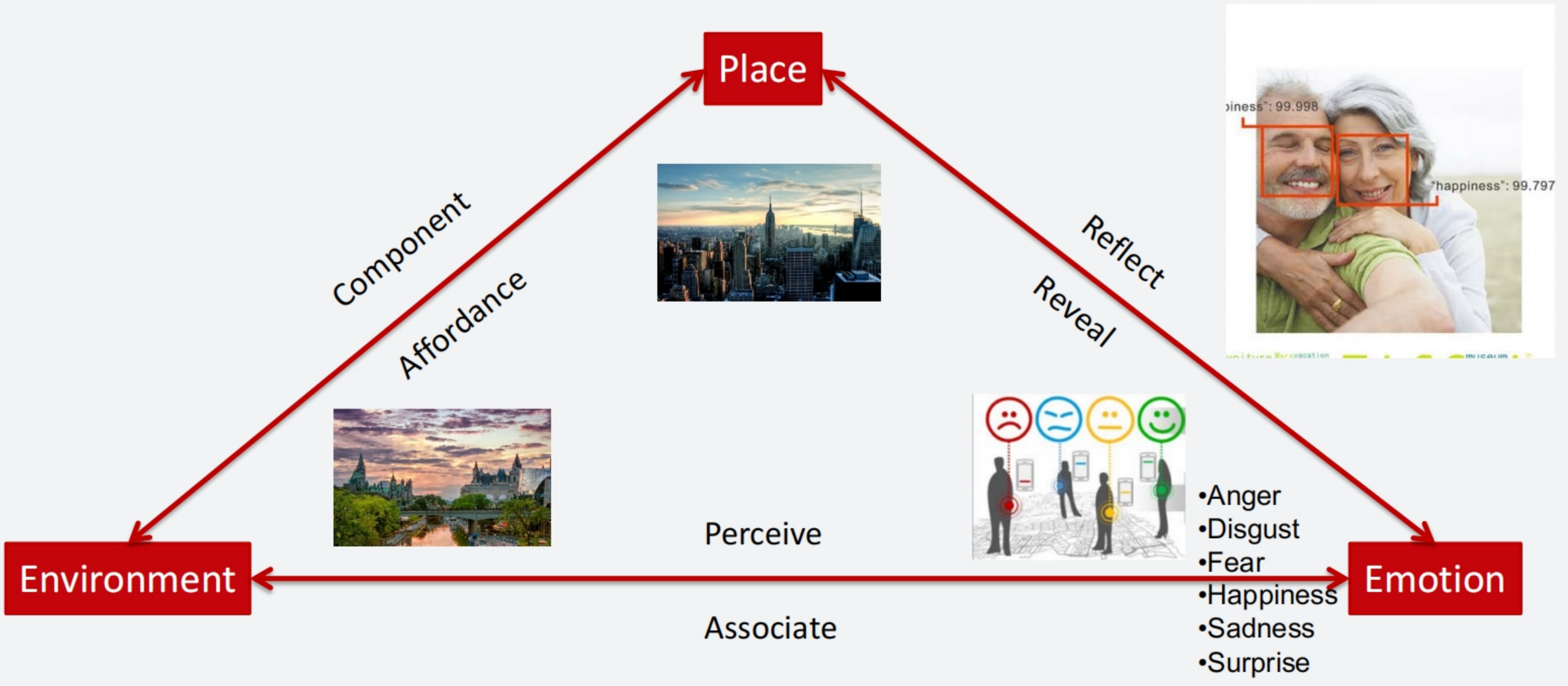
```python
from PIL import Image

#Open image using Image module
im = Image.open("reba.jpg")
#Show actual Image
im.show()
#Show rotated Image
im = im.rotate(45)
im.save("reba_rotate.jpg")
im.show()
```

Content

- What is image
  - Image fundamentals
  - Image processing
  - Useful libraries
- Some applications
  - GeoAI
  - Medical image processing
- AI + Image
  - Convolutional Neural Networks
  - Some breakthourgh

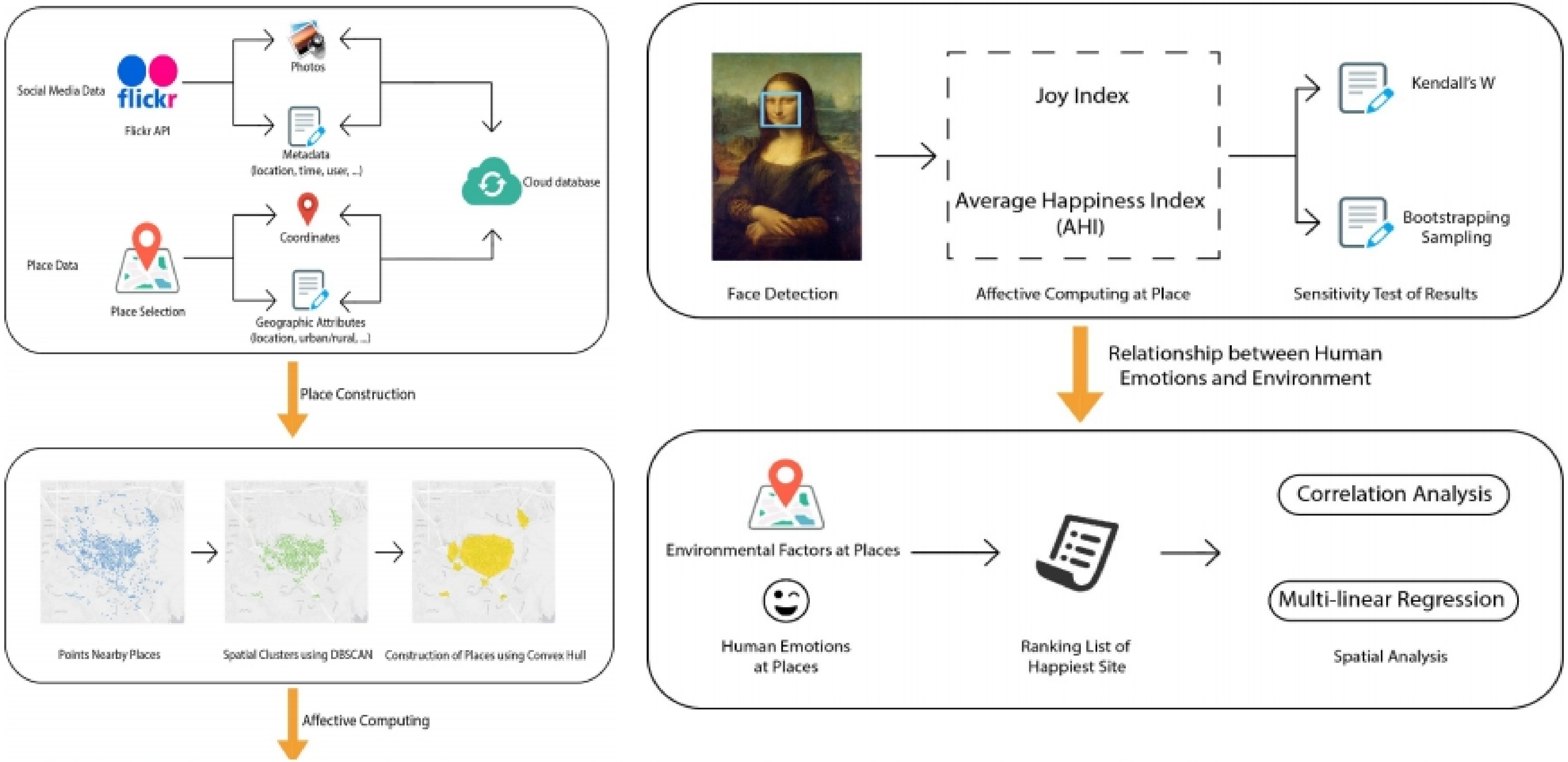
GeoAI

Human emotions at different places (Affective Computing)



GeoAI

Place Emotion Framework



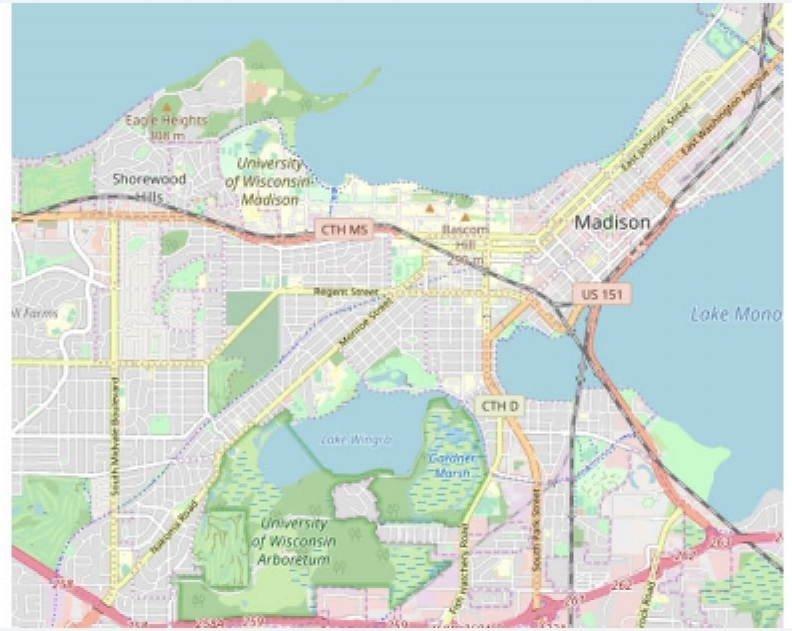
Kang, Y., Jia, Q., Gao, S., Zeng, X., Wang, Y., Angsuesser, S., Liu, Y., Ye. X., & Fei, T. (2019).. (2019). Extracting human emotions at different places based on facial expressions and spatial clustering analysis. *Transactions in GIS, 23(3).*

A Big Picture

Transferring Map Styles using Generative Adversarial Nets (GANs)

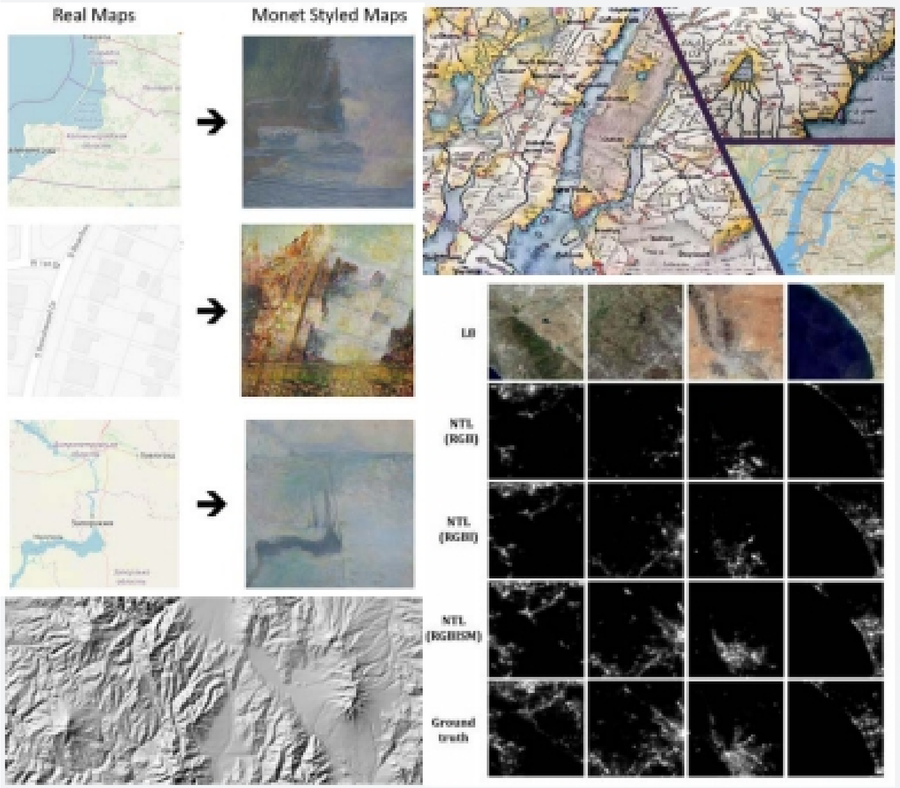
Woodcut Style

Old Fashion Style



- A ***map style*** is an aesthetically cohesive and distinct set of cartographic design characteristics.
- Two maps can have a very different look and feel based on their map style, even if depicting the same information or region.

GeoAI



AI + Maps

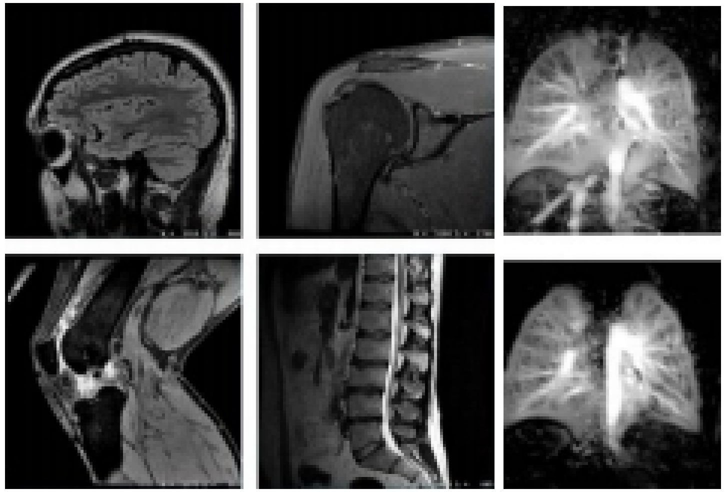
- Transfer art design styles to maps
- Satellite images to maps
- Translate multispectral to nighttime imagery
- Generation of shaded relief basemaps
- Extraction of features and symbols

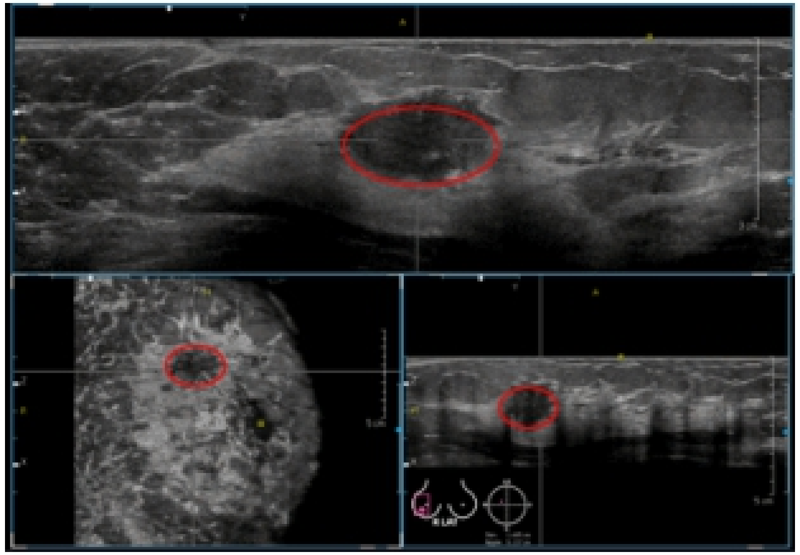
Kang, Y., Gao, S., & Roth, R. E. (2019). Transferring multiscale map styles using generative adversarial networks. *International Journal of Cartography*, 1-27.
Xu, C., & Zhao, B. (2018). Satellite Image Spoofing: Creating Remote Sensing Dataset with Generative Adversarial Networks (Short Paper). In 10th International conference on geographic information science (GIScience 2018). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.
Huang, X., Xu, D., Li, Z., & Wang, C. (2019). Translating multispectral imagery to nighttime imagery via conditional generative adversarial networks. *arXiv preprint arXiv:2001.05848*.
Duan, W., Chiang, Y., Leyk, S., Uhl, J. and Knoblock, C. (2020). Automatic alignment of contemporary vector data and georeferenced historical maps using reinforcement learning. IJGIS, 34(4).
Li, W., & Hsu, C. Y. (2020). Automated terrain feature identification from remote sensing imagery: a deep learning approach. IJGIS, 34(4), 637-660.
https://medium.com/geoai/integrating-deep-learning-with-gis-70e7c5aa9dfe

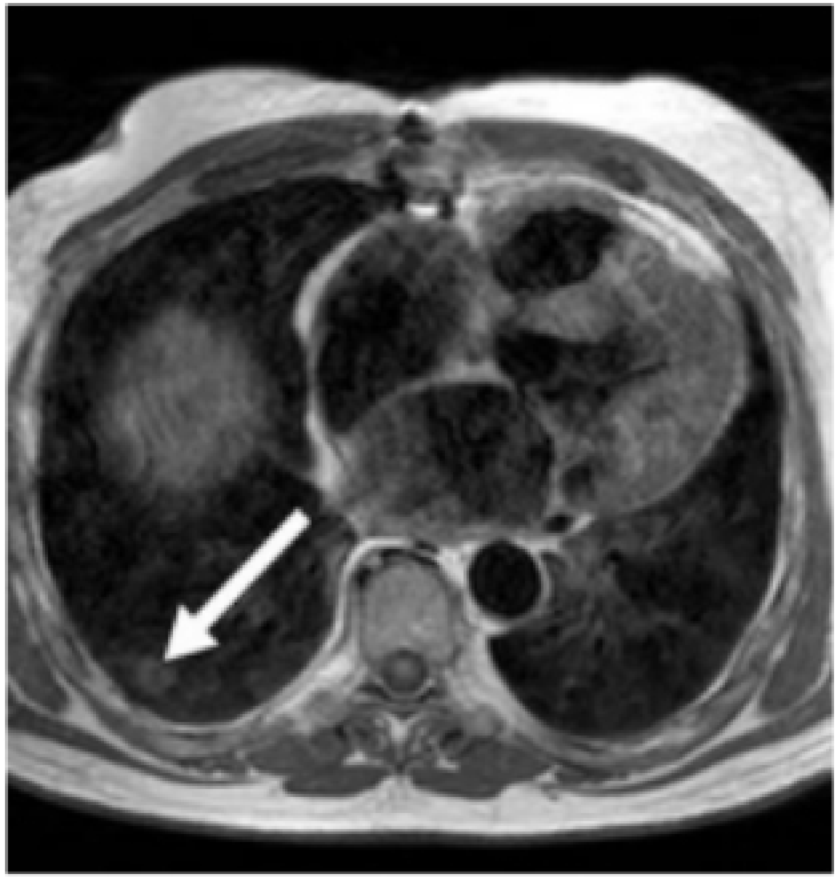
Content

- What is image
  - Image fundamentals
  - Image processing
  - Useful libraries
- Some applications
  - GeoAI
  - Medical image processing
- AI + Image
  - Convolutional Neural Networks
  - Some breakthourgh

Medical image processing



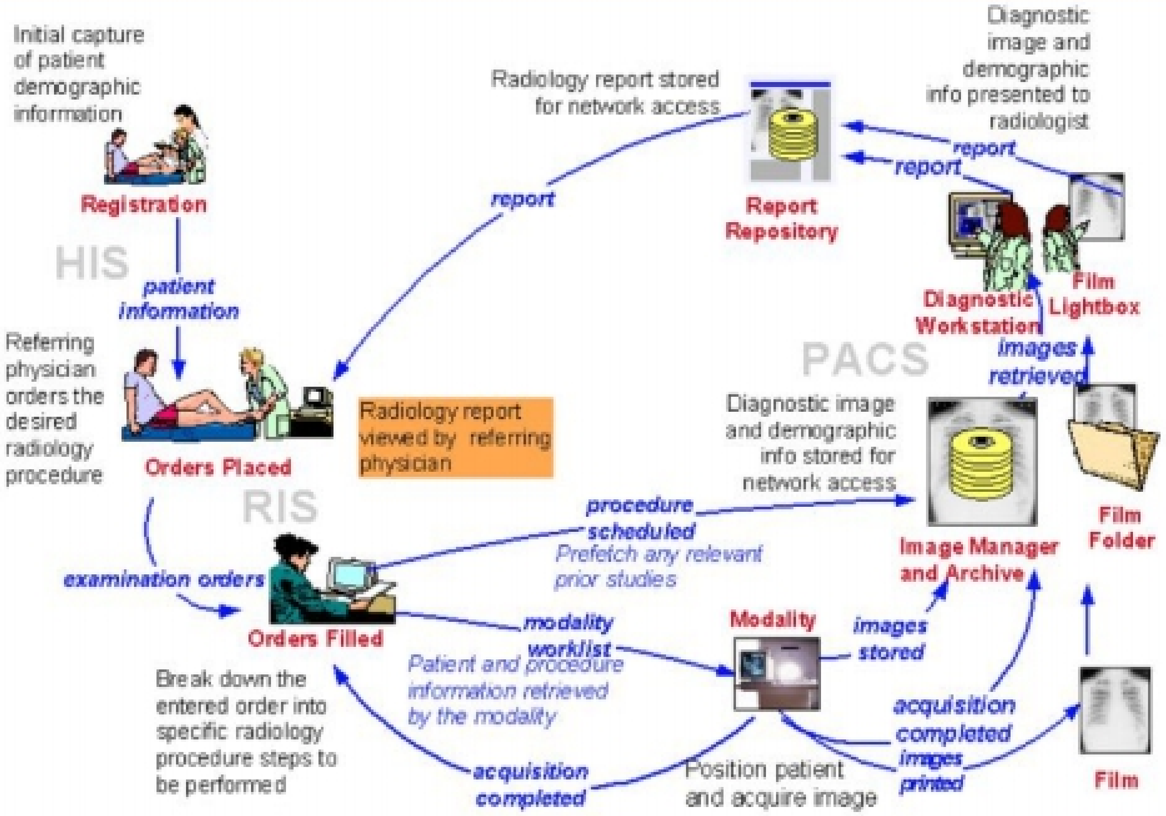




https://www.cancer.org

Medical image processing

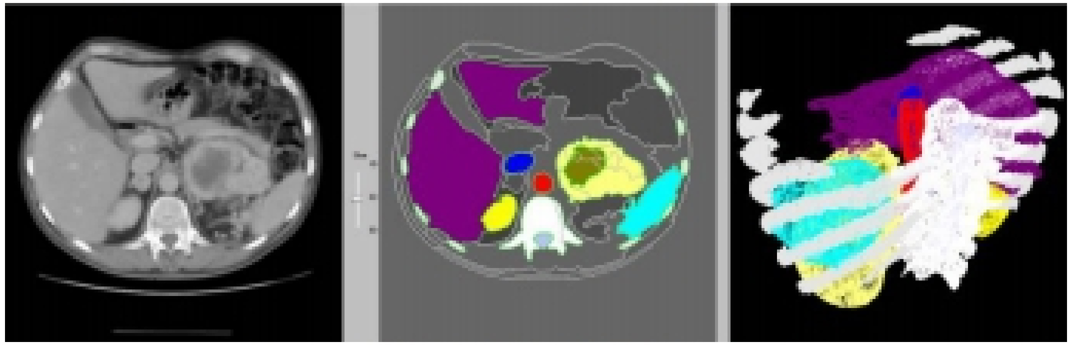
Current Radiologist's Workflow



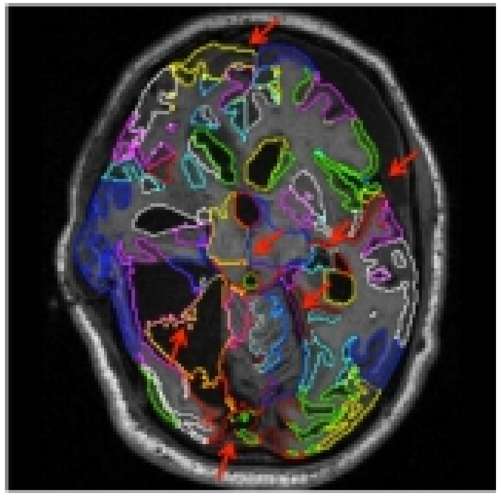
Medical image segmentation

Segmentation is the process of partitioning an image into different meaningful segments.

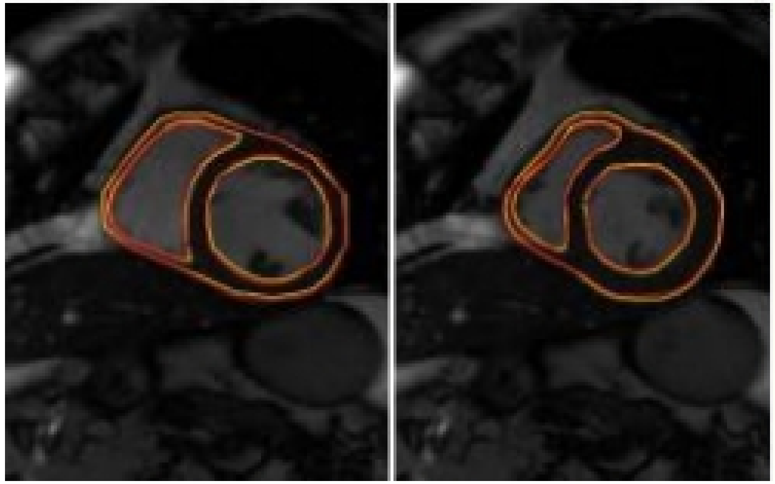
In medical imaging, these segments often correspond to different tissue classes, organs, pathologies, or other biologically relevant structures.



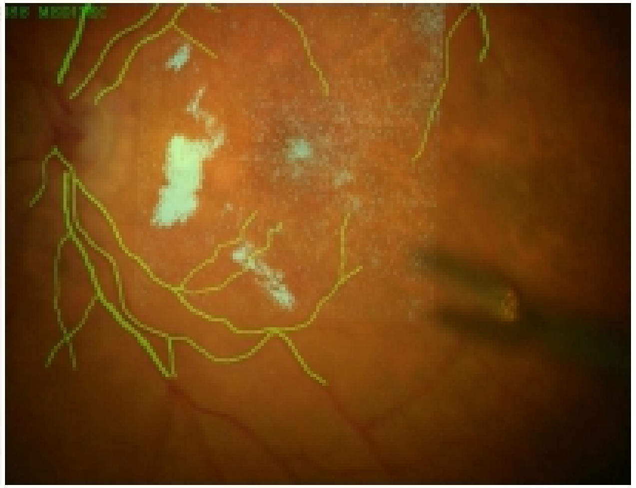
abdominal CT scan.
the rib cage, liver, kidneys, spleen, blood vessels and a renal tumor



whole-brain segmentation
C. Ledig 2015

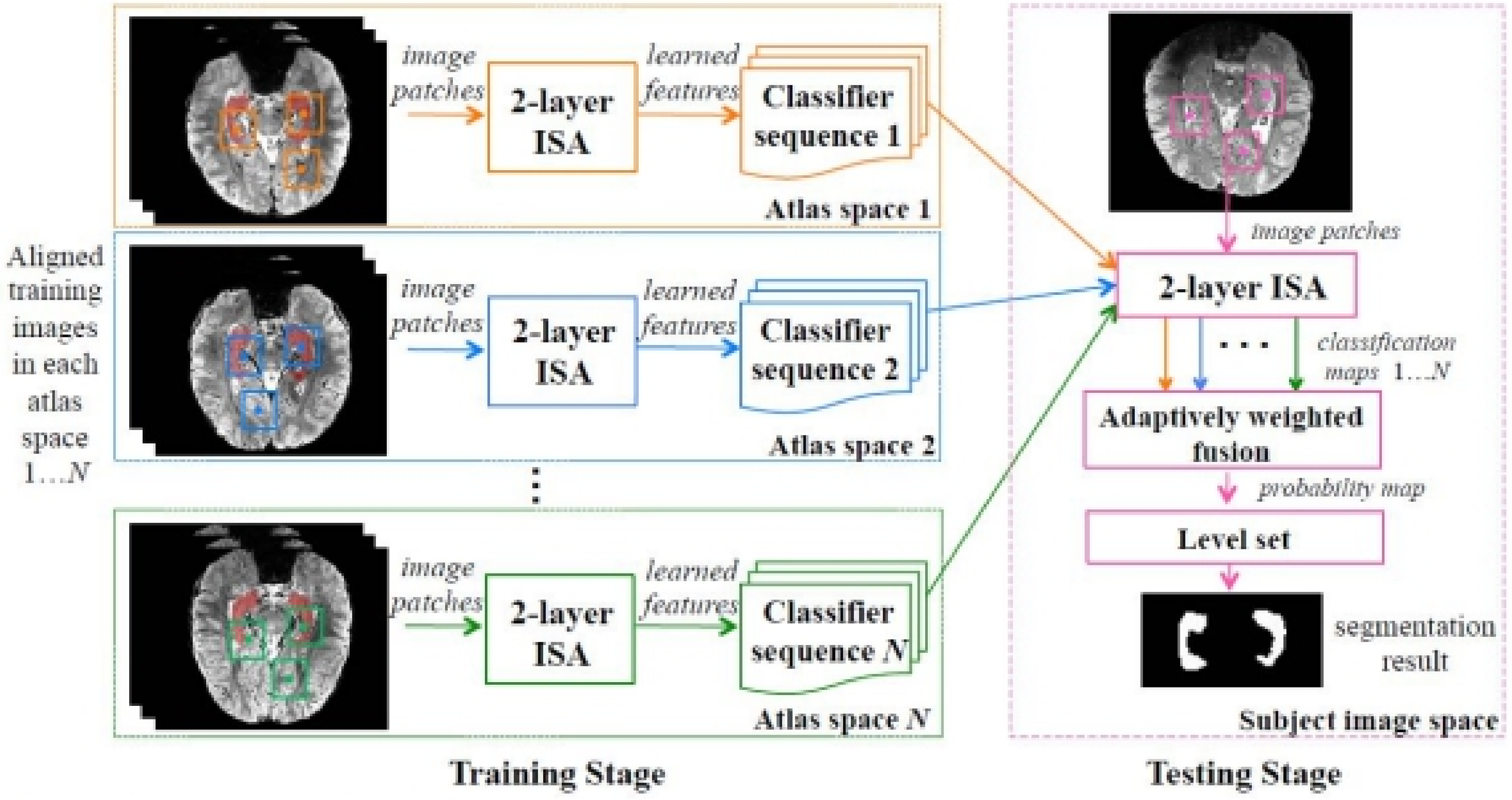


endocardium and epicardium at end of diastole and systole. M.Rezaei 2013



Retina segmentation
M.Rezaei 2014

Medical image segmentation



ISA: Independent Subspace Analysis

UNC, Dinggang Shen-2017

Content

- What is image
- Image fundamentals
- Image processing
- Useful libraries
- Some applications
- GeoAI
- Medical image processing
- AI + Image
- **Convolutional Neural Networks**
- Some breakthourgh

Applications in CV

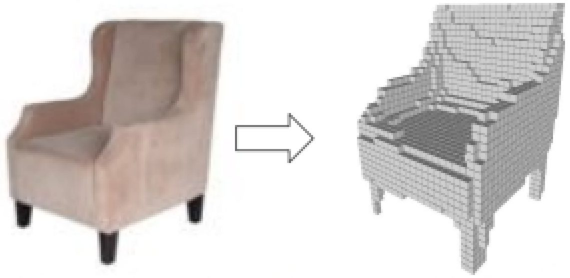
Object detection
car

Action recognition
bicycling

Visual relationship detection
<person - holding - hammer>

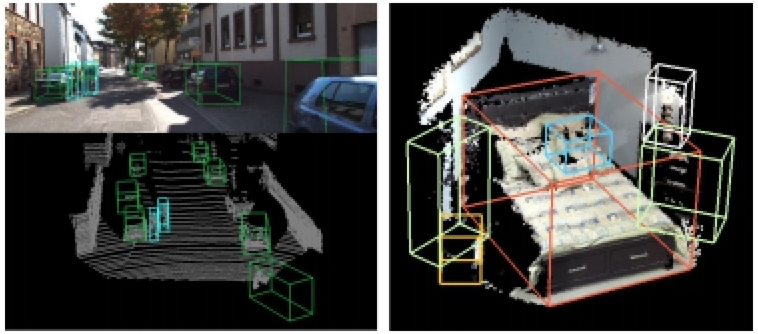
Applications in CV

3D Vision & Robotic Vision

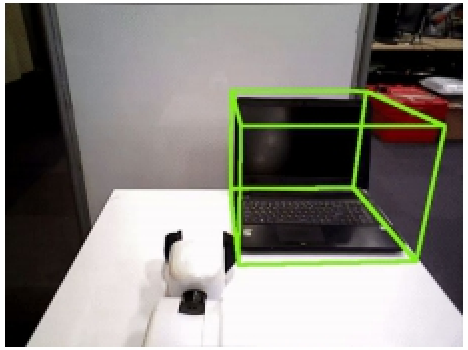


Choy et al., 3D-R2N2: Recurrent Reconstruction Neural Network (2016)

Mandlekar and Xu et al., Learning to Generalize Across Long-Horizon Tasks from Human Demonstrations (2020)



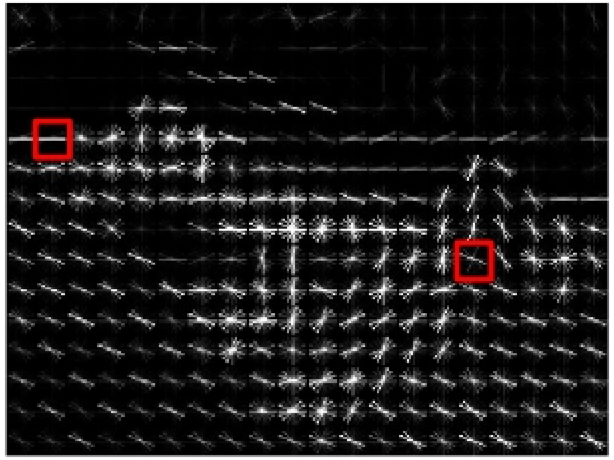
Xu et al., PointFusion: Deep Sensor Fusion for 3D Bounding Box Estimation (2018)

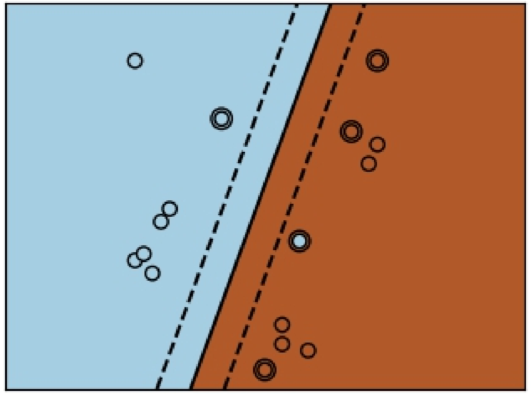


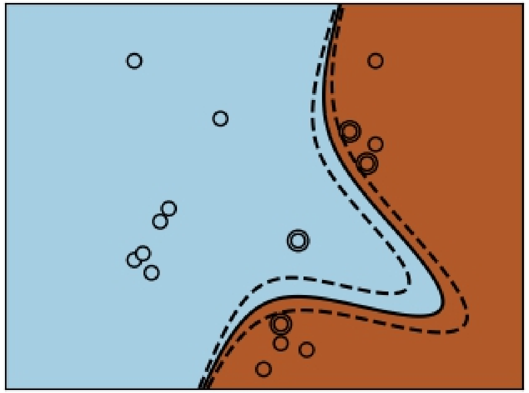
Wang et al., 6-PACK: Category-level 6D Pose Tracker with Anchor-Based Keypoints (2020)

Traditional methods

Example: Histogram of Oriented Gradients (HoG)







Divide image into 8x8 pixel regions
Within each region quantize edge direction into 9 bins

Example: 320x240 image gets divided into 40x30 bins; in each bin there are 9 numbers so feature vector has 30*40*9 = 10,800 numbers

Lowe, "Object recognition from local scale-invariant features", ICCV 1999
Dalal and Triggs, "Histograms of oriented gradients for human detection," CVPR 2005

Current methods

Neural networks: without the brain stuff

(**Before**) Linear score function: $f = Wx$

(**Now**) 2-layer Neural Network $f = W_2 \max(0, W_1 x)$

or 3-layer Neural Network

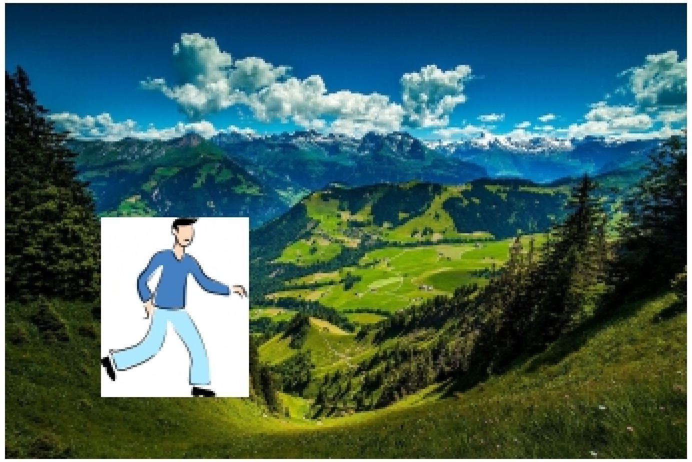
$$f = W_3 \max(0, W_2 \max(0, W_1 x))$$

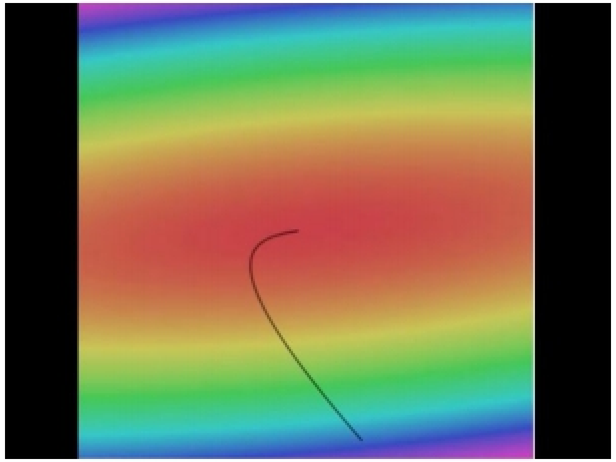
$$x \in \mathbb{R}^D, W_1 \in \mathbb{R}^{H_1 \times D}, W_2 \in \mathbb{R}^{H_2 \times H_1}, W_3 \in \mathbb{R}^{C \times H_2}$$

(In practice we will usually add a learnable bias at each layer as well)

Current methods

Finding the best W: Optimize with Gradient Descent





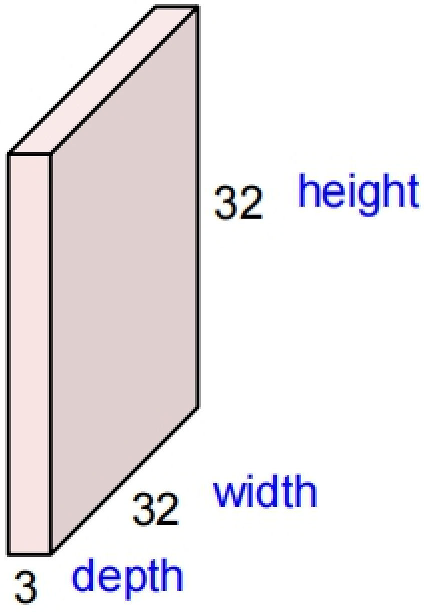
Gradient descent

$$\frac{df(x)}{dx} = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

```
# Vanilla Gradient Descent

while True:
  weights_grad = evaluate_gradient(loss_fun, data, weights)
  weights += - step_size * weights_grad # perform parameter update
```

Convolution Layer

32x32x3 image -> preserve spatial structure

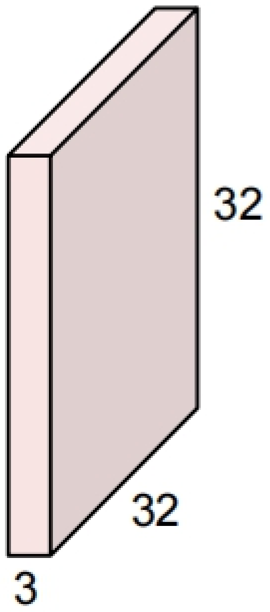


Convolutional Neural Networks

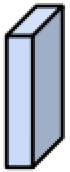
Convolution Layer

Filters always extend the full depth of the input volume

32x32x3 image



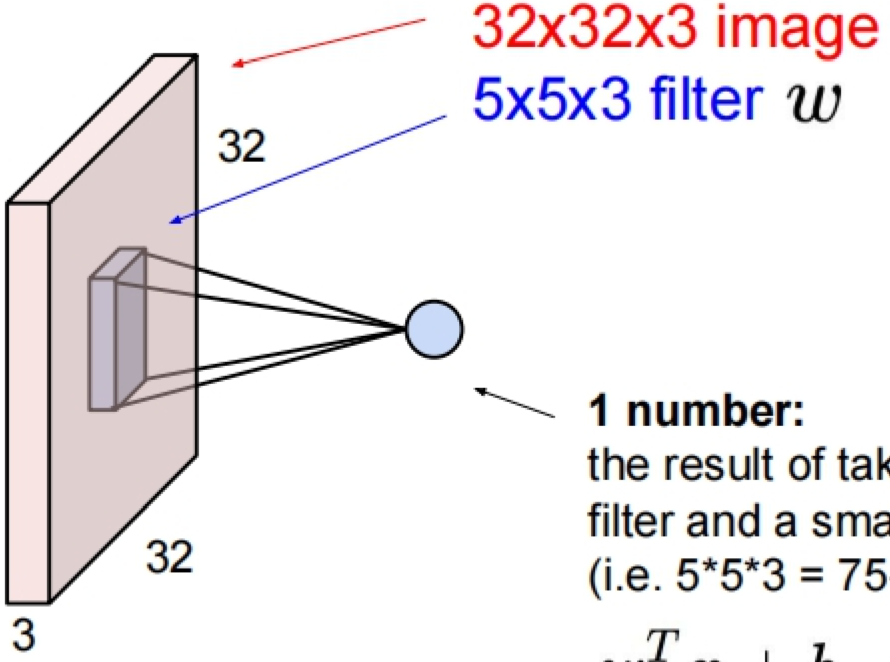
5x5x3 filter



Convolve the filter with the image i.e. “slide over the image spatially, computing dot products”

Convolutional Neural Networks

Convolution Layer

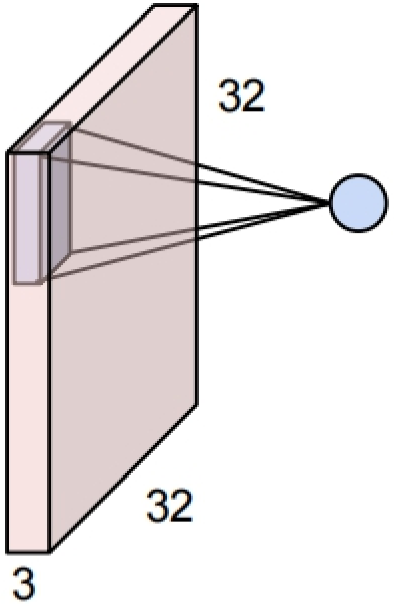


1 number:
the result of taking a dot product between the filter and a small 5x5x3 chunk of the image
(i.e. 5*5*3 = 75-dimensional dot product + bias)

$$w^T x + b$$

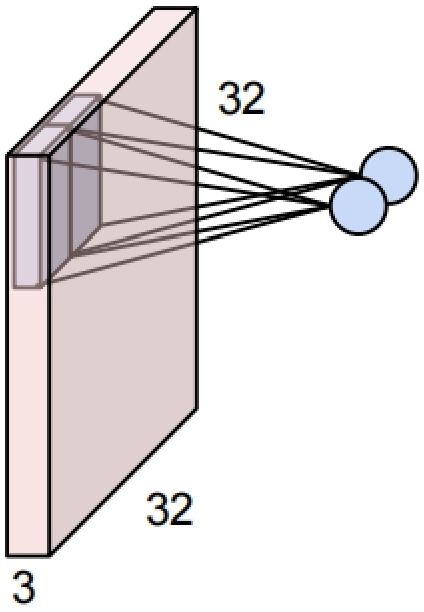
Convolutional Neural Networks

Convolution Layer



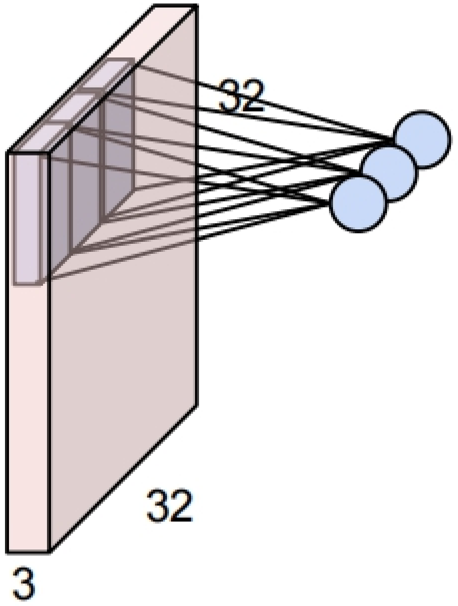
Convolutional Neural Networks

Convolution Layer



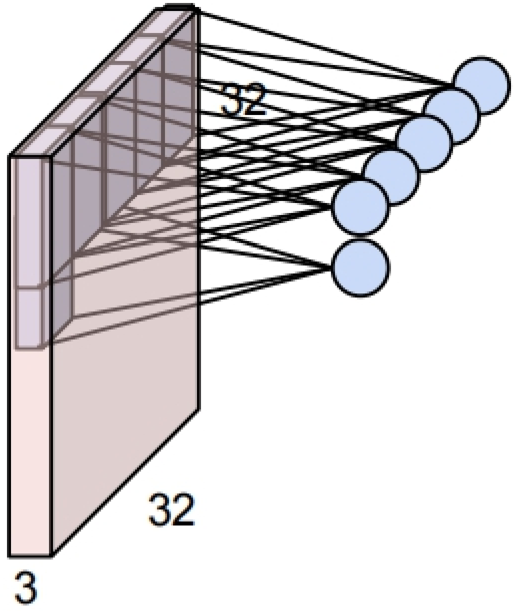
Convolutional Neural Networks

Convolution Layer



Convolutional Neural Networks

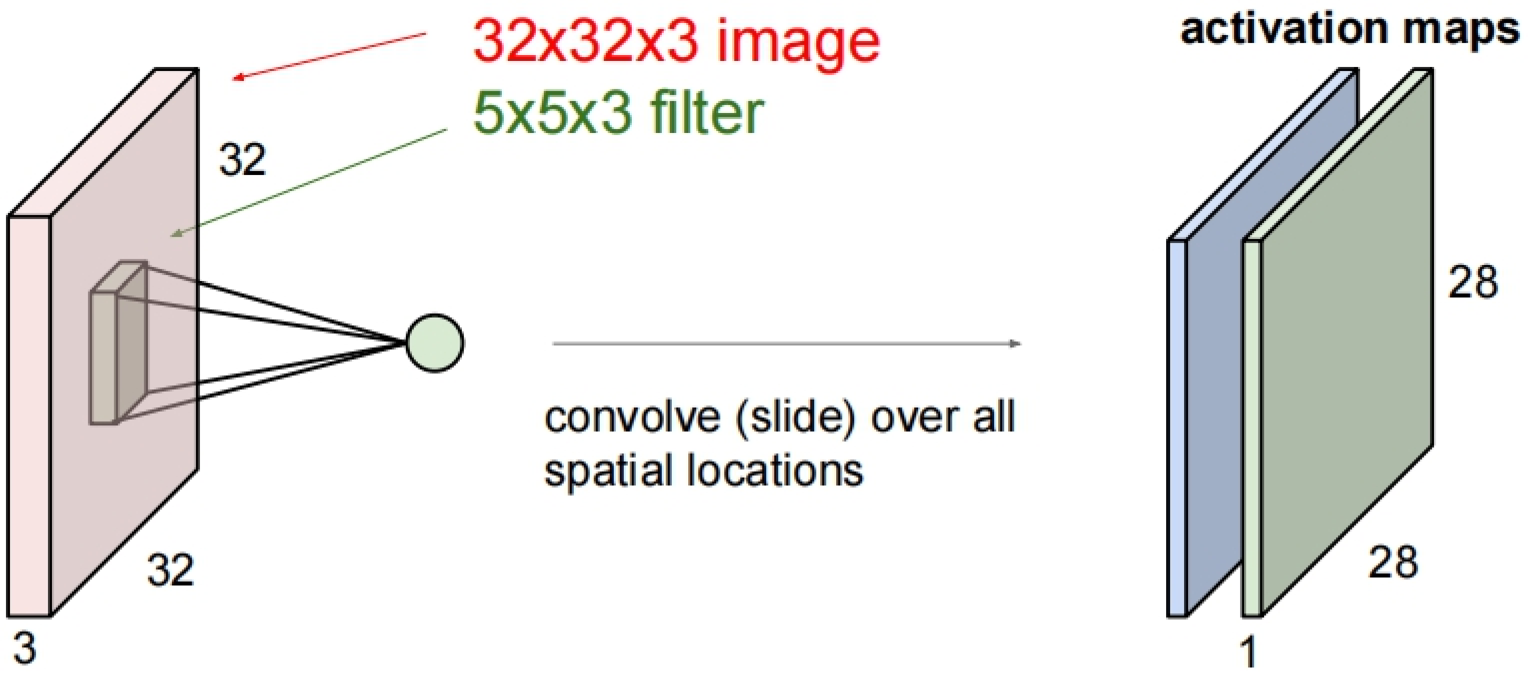
Convolution Layer



Convolutional Neural Networks

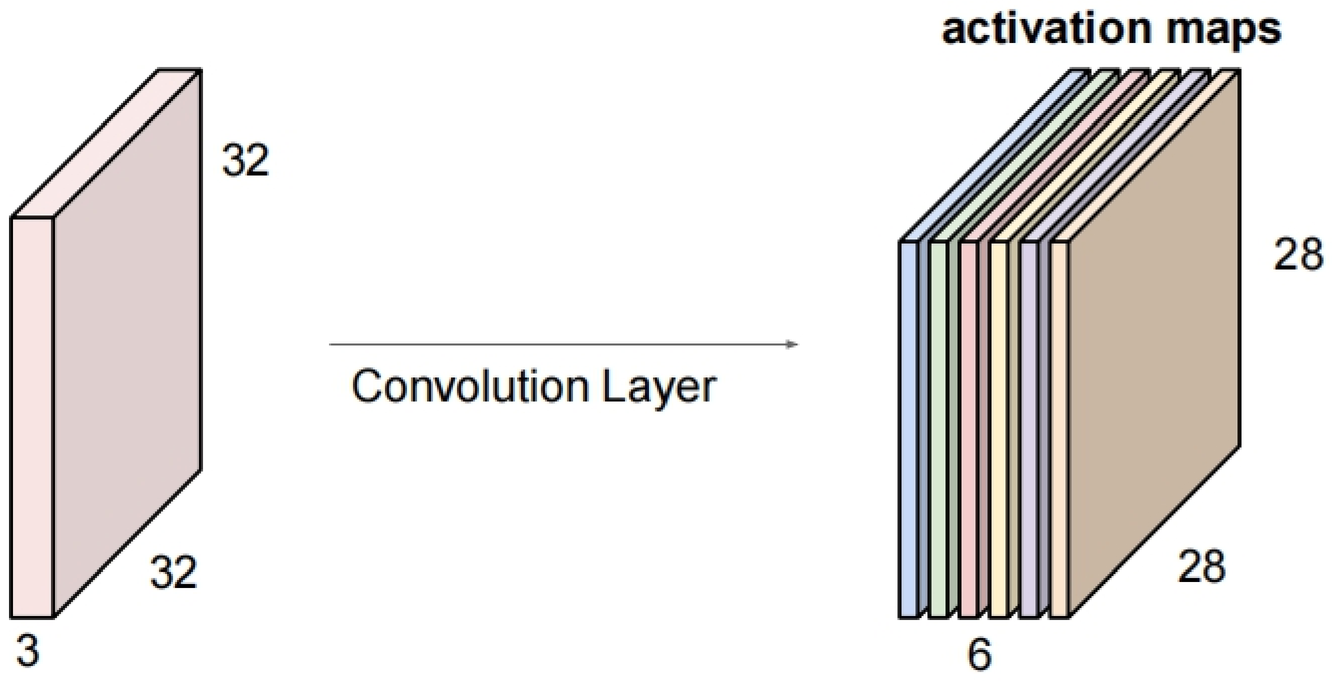
Convolution Layer

consider a second, green filter



Convolutional Neural Networks

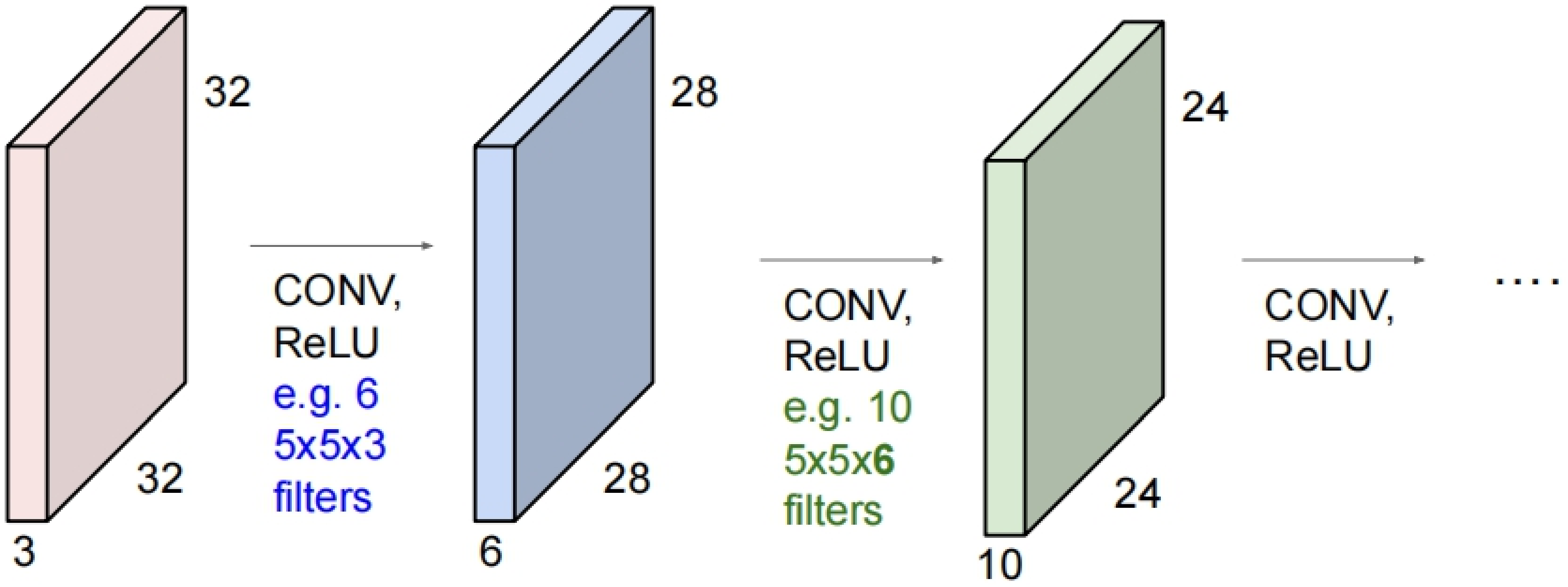
For example, if we had 6 5x5 filters, we'll get 6 separate activation maps:



We stack these up to get a "new image" of size 28x28x6!

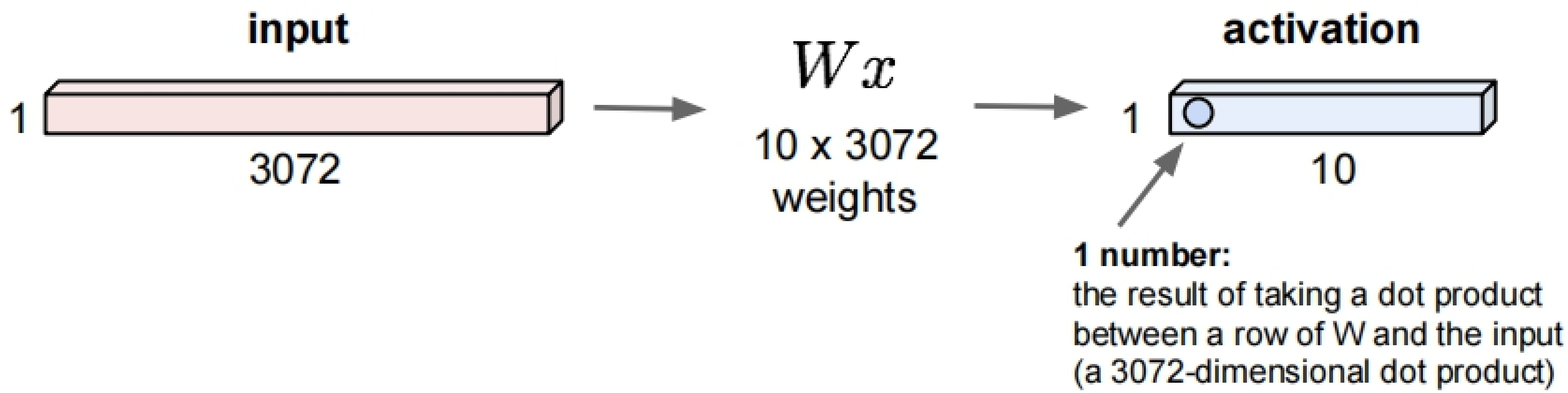
Convolutional Neural Networks

Preview: ConvNet is a sequence of Convolution Layers, interspersed with activation functions



Fully Connected Layer

32x32x3 image -> stretch to 3072 x 1

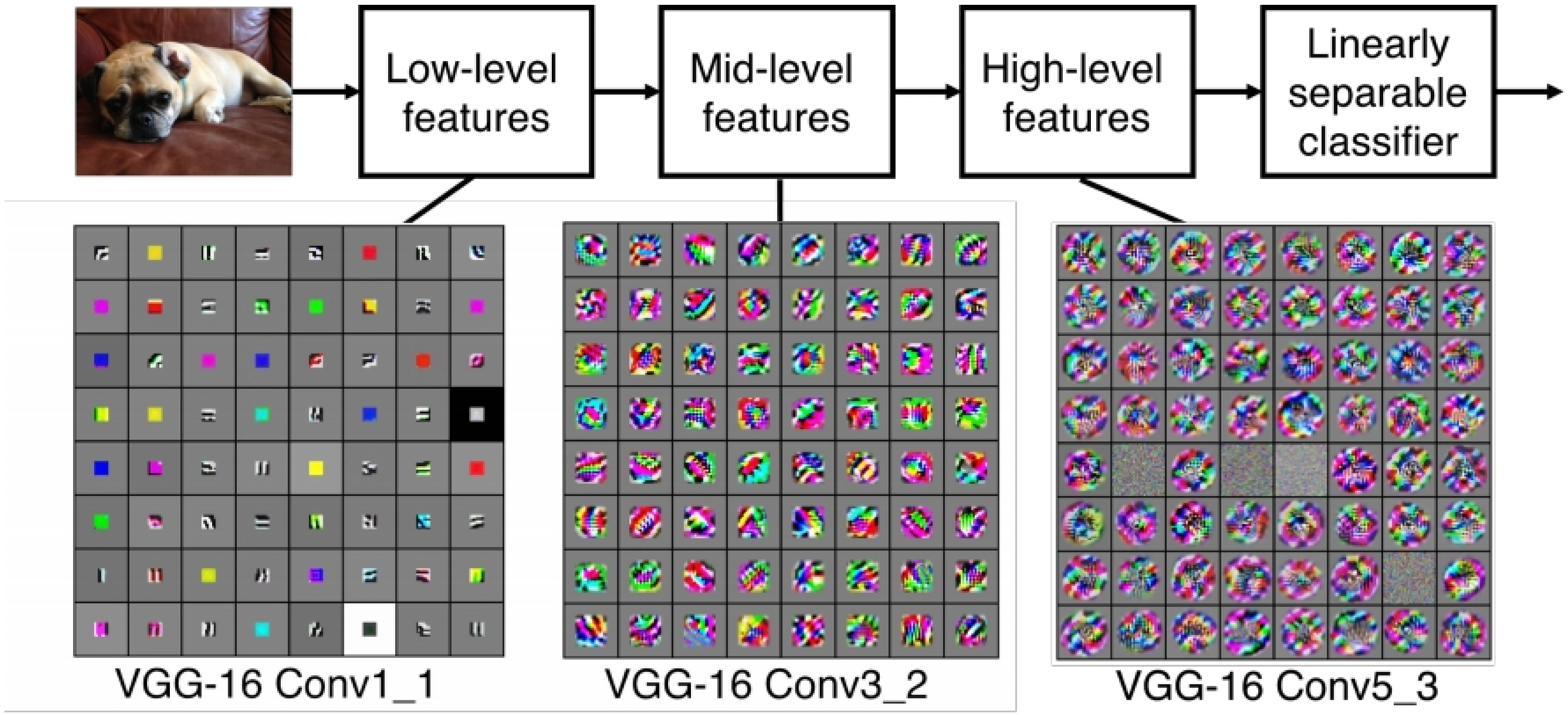


Convolutional Neural Networks

Preview

[Zeiler and Fergus 2013]

Visualization of VGG-16 by Lane McIntosh. VGG-16 architecture from [Simonyan and Zisserman 2014].



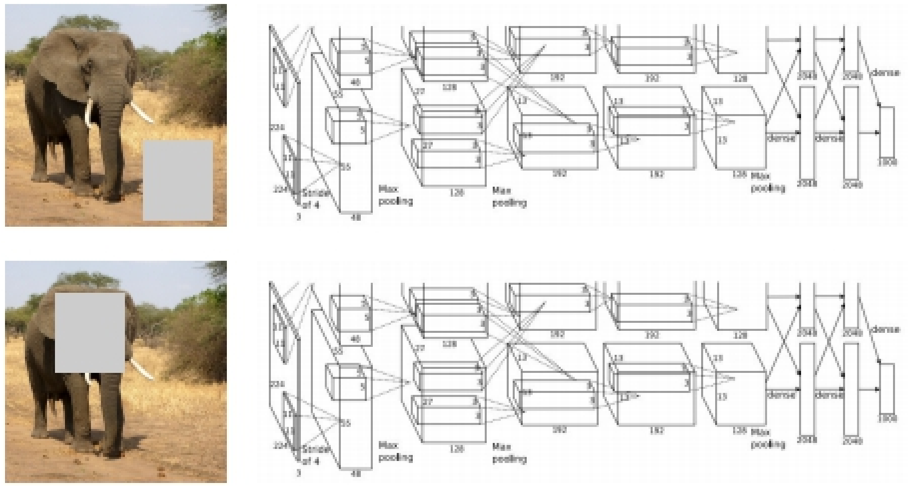
Content

- **What is image**
  - Image fundamentals
  - Image processing
  - Useful libraries
- **Some applications**
  - GeoAI
  - Medical image processing
- **AI + Image**
  - Convolutional Neural Networks
  - Some breakthourgh

How to interpret it?

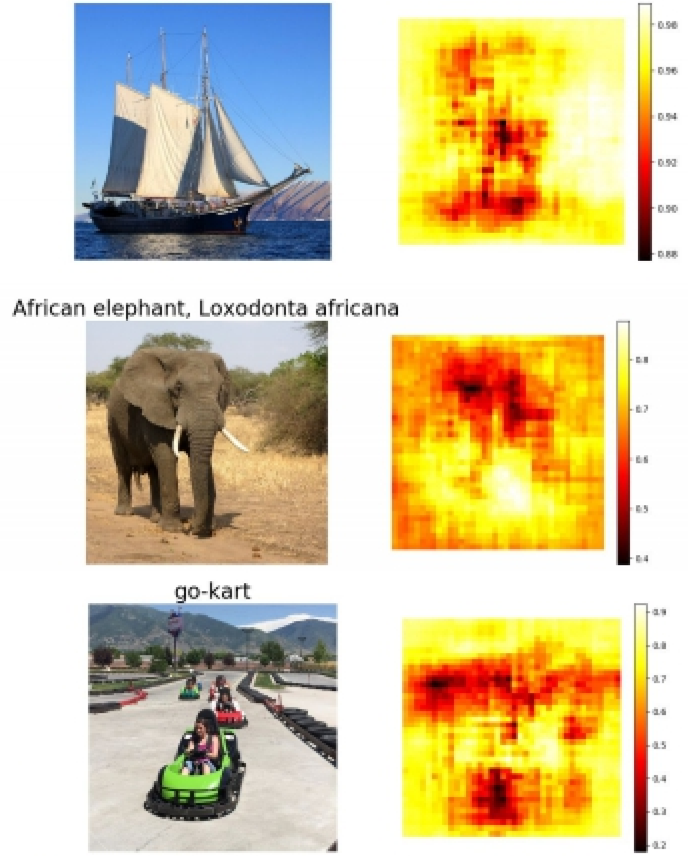
Which pixels matter: Saliency via Occlusion

Mask part of the image before feeding to CNN, check how much predicted probabilities change



P(elephant) = 0.95

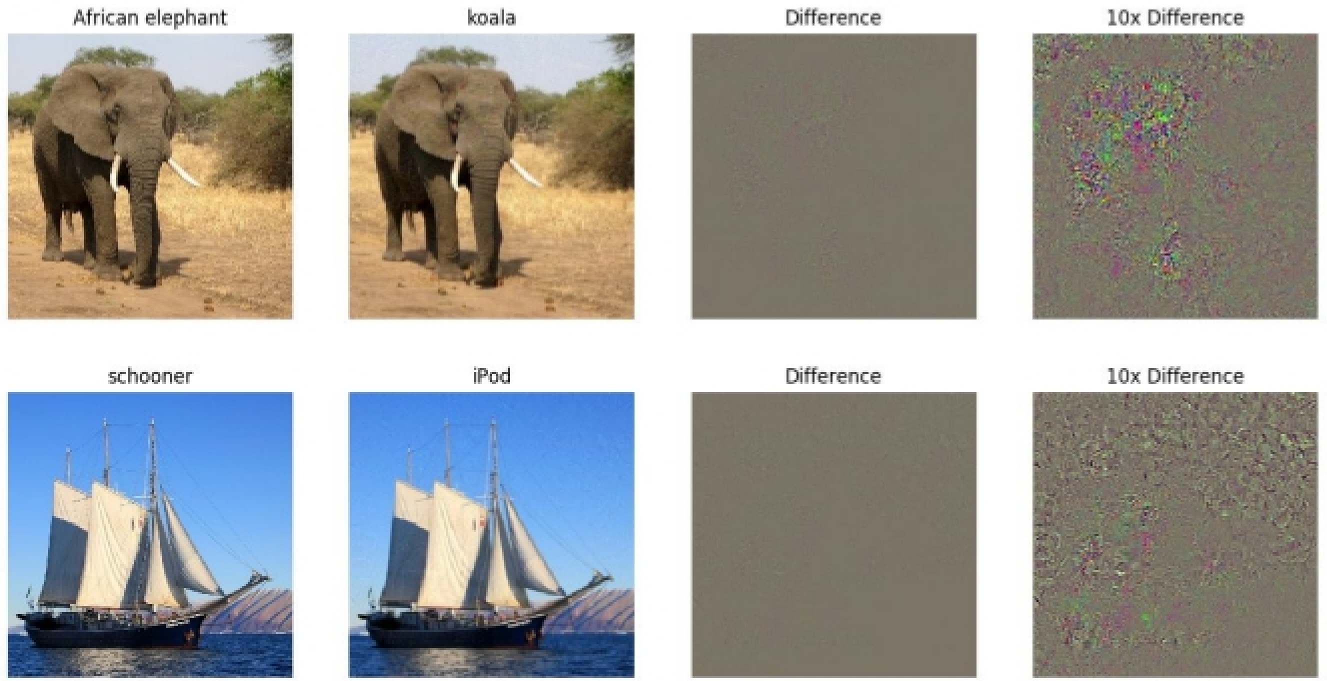
P(elephant) = 0.75



Zeiler and Fergus, "Visualizing and Understanding Convolutional Networks", ECCV 2014

Adversarial attack

Fooling Images / Adversarial Examples

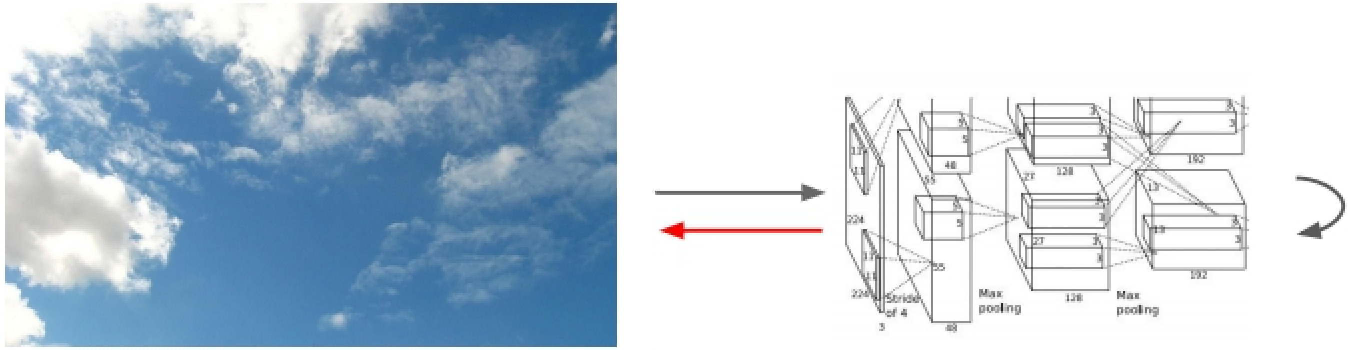


Boat image is CC0 public domain
Elephant image is CC0 public domain

DeepDream

DeepDream: Amplify existing features

Rather than synthesizing an image to maximize a specific neuron, instead try to **amplify** the neuron activations at some layer in the network



Choose an image and a layer in a CNN; repeat:

1. Forward: compute activations at chosen layer
2. Set gradient of chosen layer *equal to its activation*
3. Backward: Compute gradient on image
4. Update image

Equivalent to:

$$I^* = \arg\max_I \sum_i f_i(I)^2$$

Mordvintsev, Olah, and Tyka, "Inceptionism: Going Deeper into Neural Networks", Google Research Blog. Images are licensed under CC-BY 4.0

DeepDream