nodeflux

Extending Vision Beyond Imagination

Week 1 - Computer Vision Basic

www.nodeflux.io

Outline

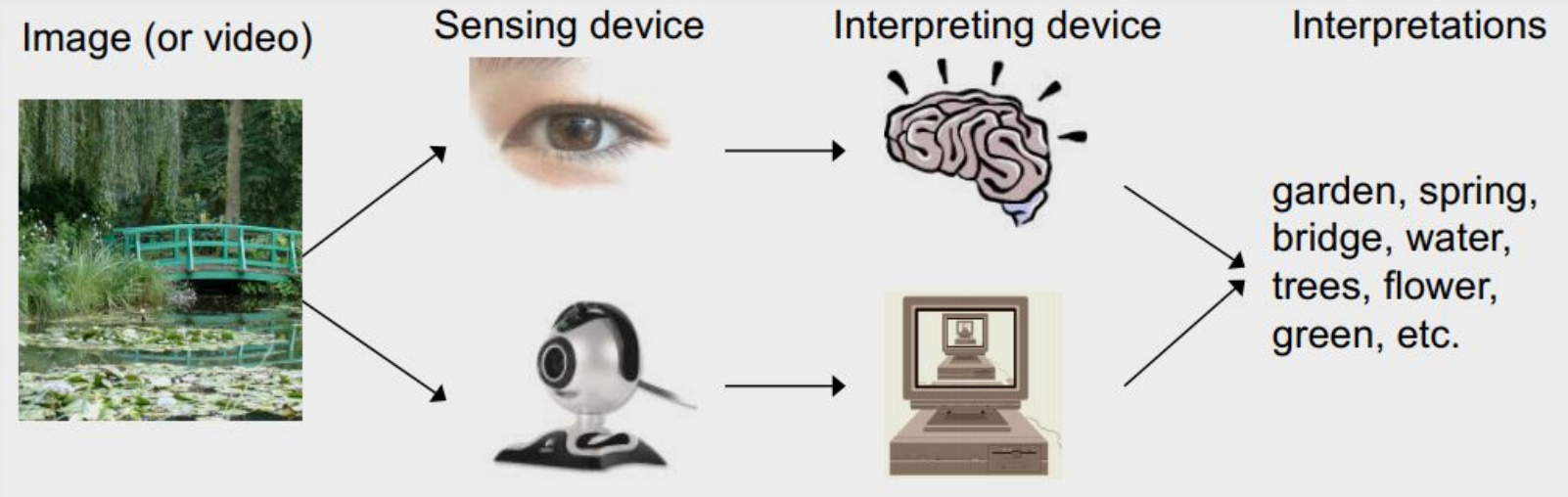
- Section 1 Introduction
- Section 2 Image Processing
- Section 3 Augmentation

nodeflux

Extending Vision Beyond Imagination

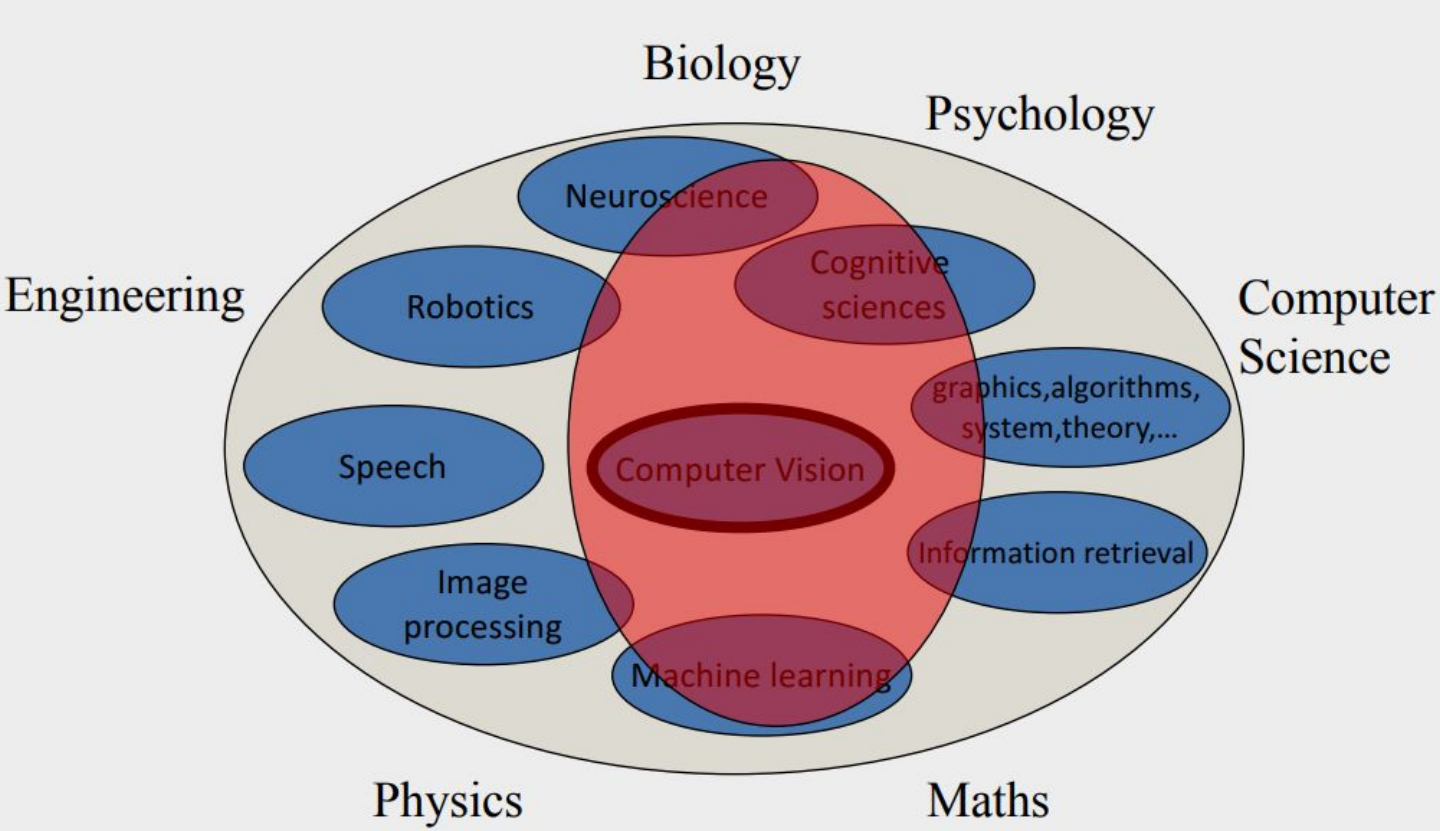
Section 1 - Introduction

What is (Computer) Vision?



Ref : CS131

Related Field



Ref :
CS131

The goal of computer vision

- To bridge the gap between pixels and “meaning”

What we see

0	3	2	5	4	7	6	9	8
3	0	1	2	3	4	5	6	7
2	1	0	3	2	5	4	7	6
5	2	3	0	1	2	3	4	5
4	3	2	1	0	3	2	5	4
7	4	5	2	3	0	1	2	3
6	5	4	3	2	1	0	3	2
9	6	7	4	5	2	3	0	1
8	7	6	5	4	3	2	1	0

What a computer sees

Ref :
CS131

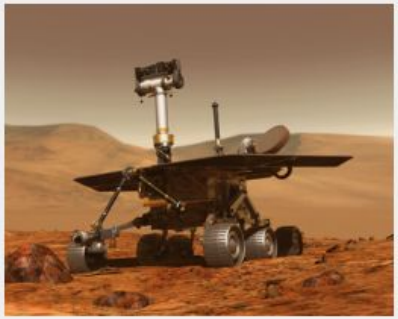
What information can we extract from images?

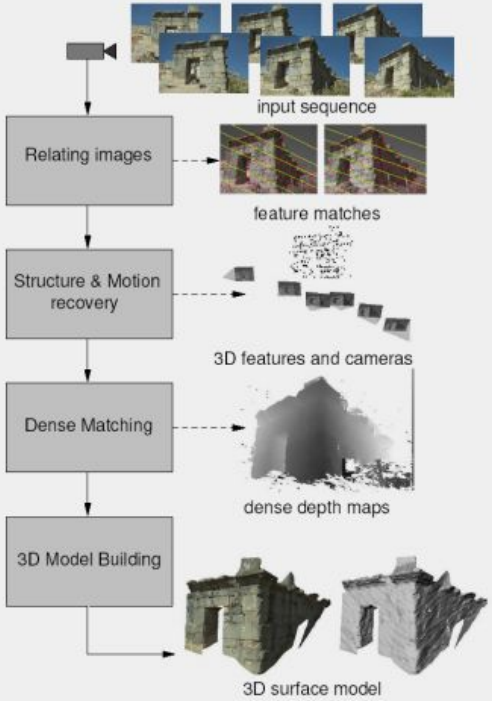
- Metric 3D Information
- Semantic Information

Ref :
CS131

Metric 3D Information

Vision as measurement device





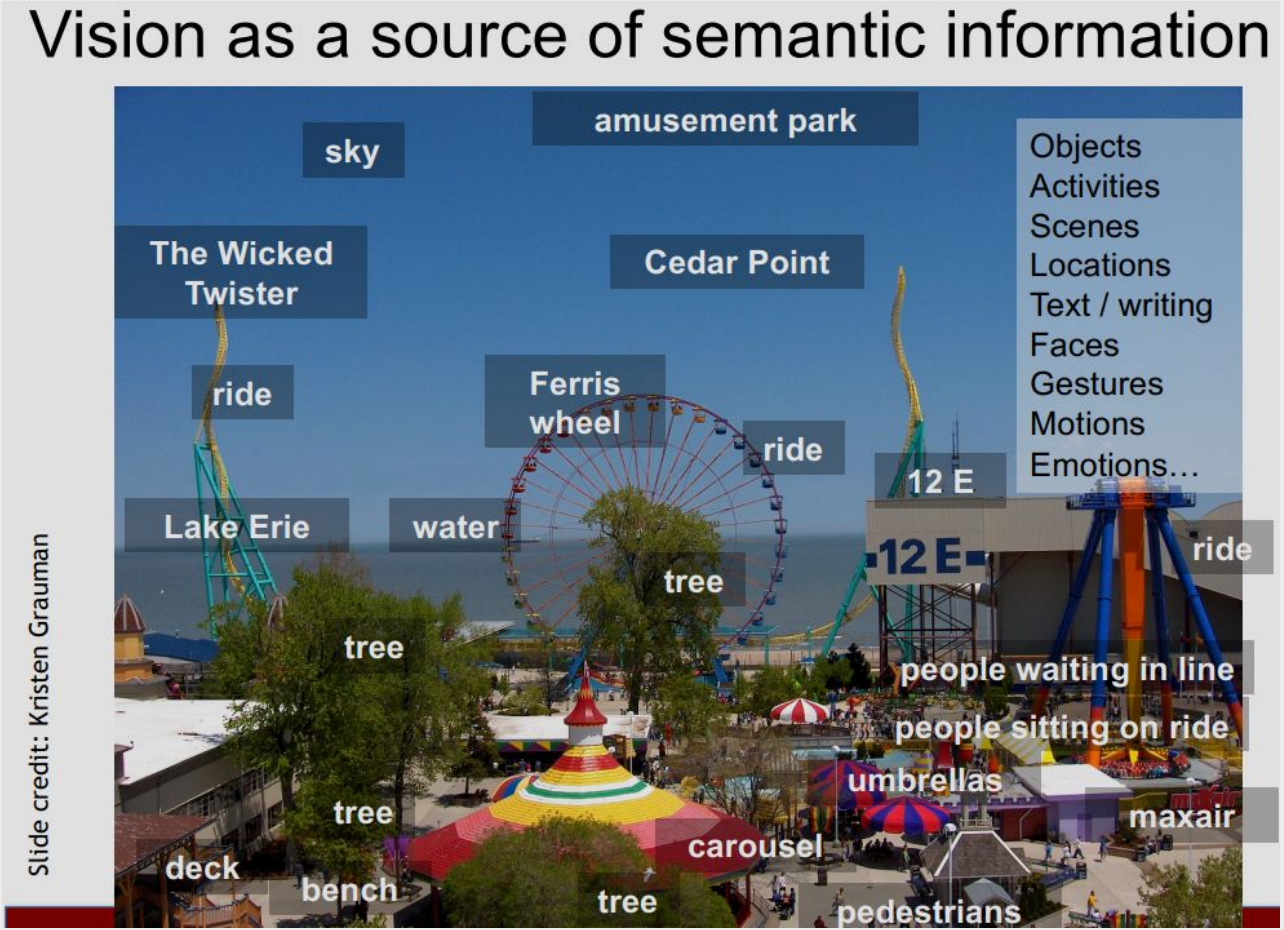
Pollefeys et al.

Goesele et al.

Ref :
CS131

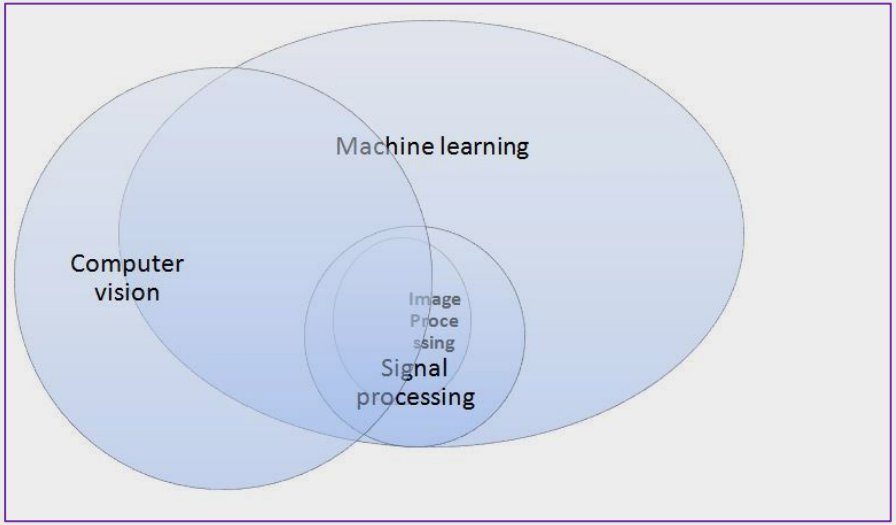
Semantic Information

Vision as a source of semantic information



Ref :
CS131

Image Processing, Computer Vision, & Machine Learning



The Main Differences Is Their Output

Domain	Input	Output
Image processing	Image	Image
Signal processing	Signal	Signal, quantitative information, e.g. Peak location.
Computer vision	Image/video	Image, quantitative/qualitative information, e.g. size, color, shape, classification, etc…
Machine learning	Any feature signal, from e.g. image, video, sound, etc..	Signal, quantitative/qualitative information, image,…

What Computer Vision Can Actually Do

Classification

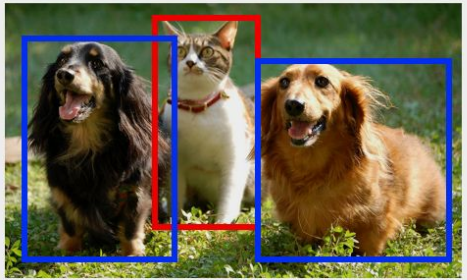
CAT

Classification + Localization

CAT

Single object

Object Detection



CAT, DOG

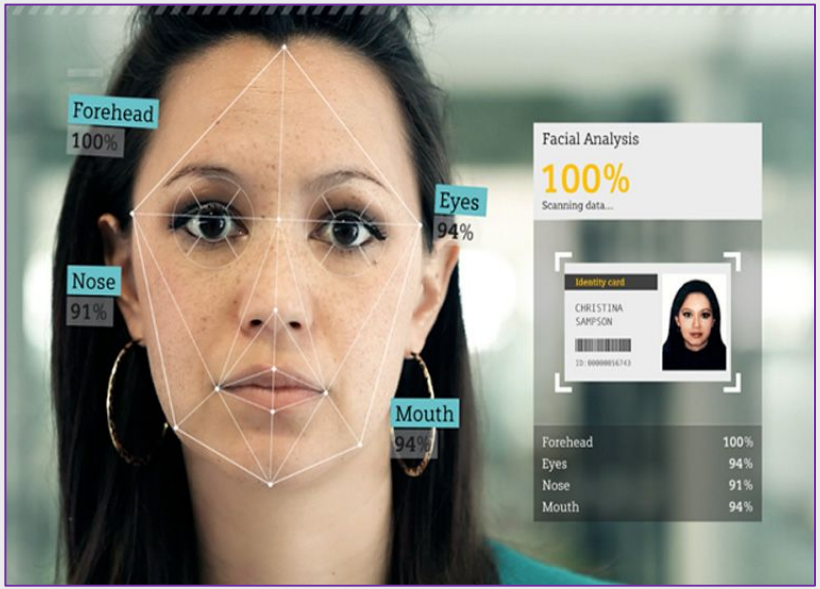
Instance Segmentation

CAT, DOG

Multiple objects

The Application

Face Recognition



The Application

Self Driving Car

The Application

Generate 3D Environment

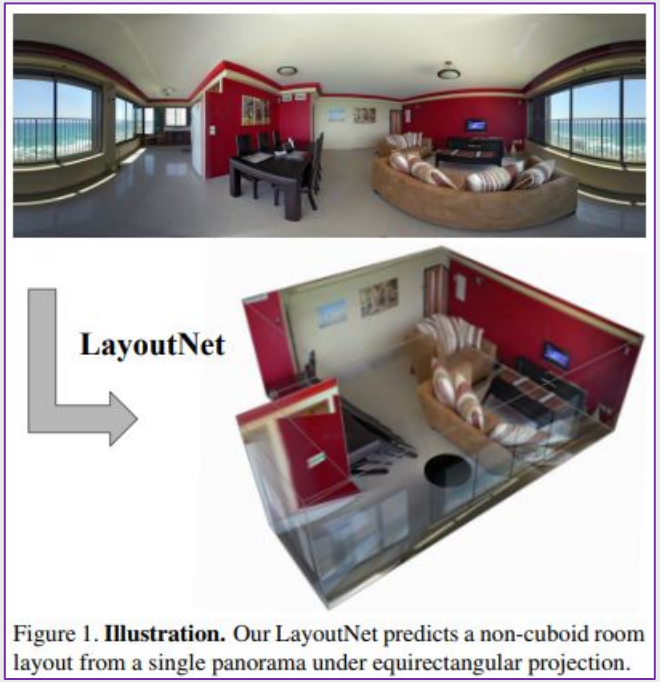


Figure 1. **Illustration.** Our LayoutNet predicts a non-cuboid room layout from a single panorama under equirectangular projection.

Image Types

- Binary Images : contain pixel that are either black or white
- Grayscale Images: have a wider range of intensity than black and white. with range between 0 and 255
- Color Image: image with multiple color channel, can be represented with RGB, LAB, HSV (usually RGB)

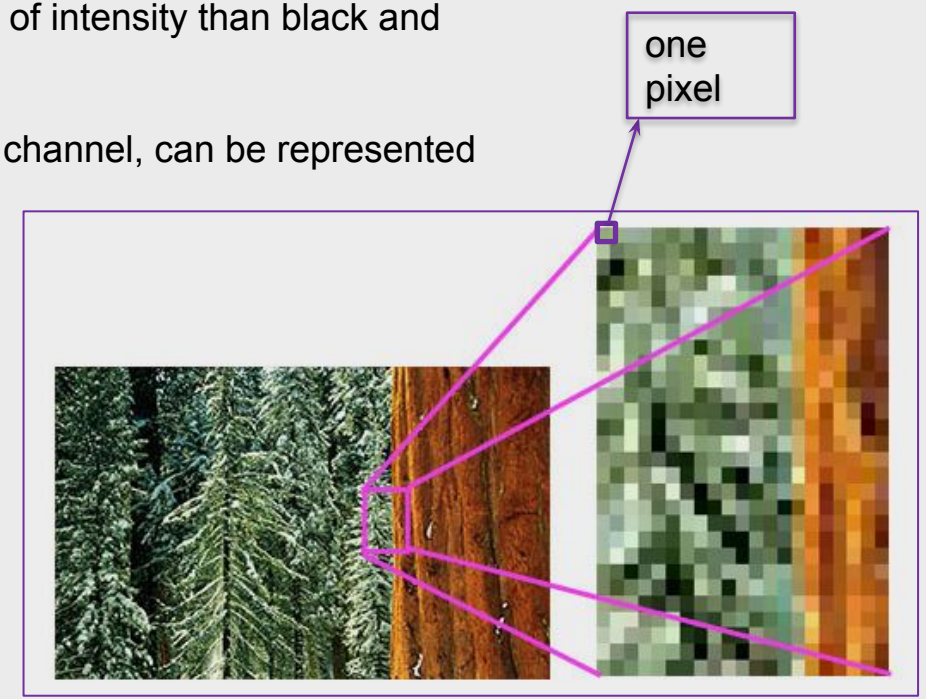


Image Histograms

Histograms measure the frequency of brightness within the image,
how many pixel value appear in an image

Ref: CS131

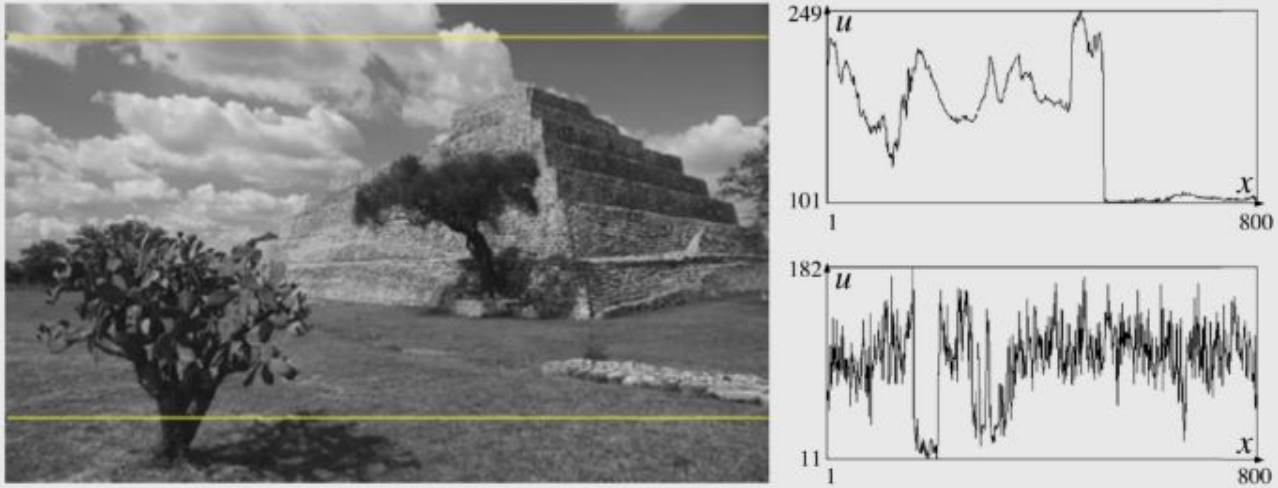
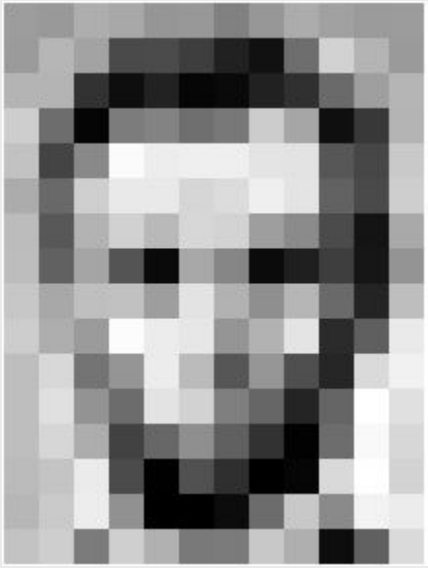


Figure 2: The image is sampled at two vertical positions, sampling a patch of sky and sampling a patch of grass. The corresponding histograms are shown to the right. Adapted from the accompanying lecture slide (Slide 23, slide credit Dr. Mubarak Shah

How Computer See Image



157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

f(1,17) = 195

Pixel location

195

Range value of pixel (8 bit) :
0 (black) - 255 (white)

Reading Pixel Coordinate

x coordinate

y coordinate

$(0,0)$	$(0,1)$	$(0,2)$	$(0,3)$
$(1,0)$	$(1,1)$	$(1,2)$	$(1,3)$
$(2,0)$	$(2,1)$	$(2,2)$	$(2,3)$
$(3,0)$	$(3,1)$	$(3,2)$	$(3,3)$

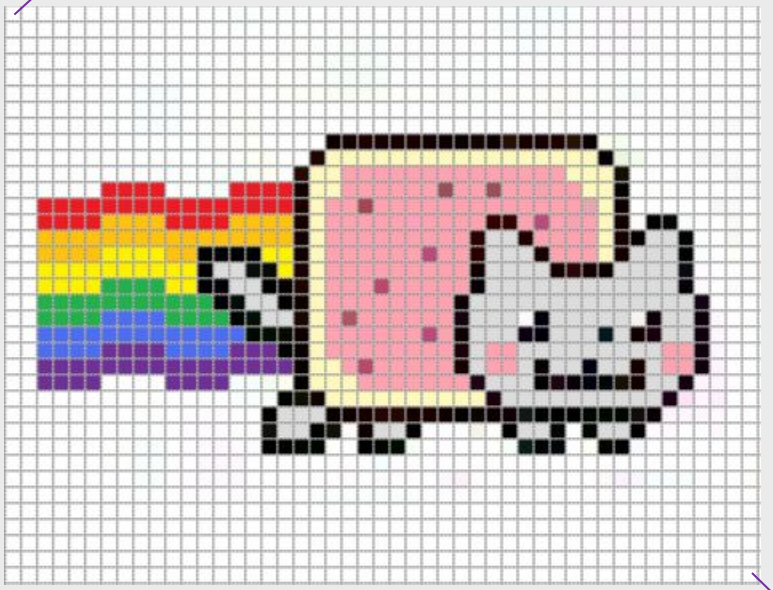


Image as Function

- **An Image** as a function f from R^2 to R^M:
  - $f(x, y)$ gives the **intensity** at position (x, y)
  - Defined over a rectangle, with a finite range:

$$f: \underbrace{[a,b] \times [c,d]}_{\text{Domain support}} \rightarrow \underbrace{[0,255]}_{\text{range}}$$

- A color image: $f(x,y) = \begin{bmatrix} r(x,y) \\ g(x,y) \\ b(x,y) \end{bmatrix}$

Image as Discrete

- Images are usually **digital** (**discrete**):
  - **Sample** the 2D space on a regular grid
- Represented as a matrix of integer values

pixel

j

i

62	79	23	119	120	105	4	0
10	10	9	62	12	78	34	0
10	58	197	46	46	0	0	48
176	135	5	188	191	68	0	49
2	1	1	29	26	37	0	77
0	89	144	147	187	102	62	208
255	252	0	166	123	62	0	31
166	63	127	17	1	0	99	30

Ref:
https://mpatacchiola.github.io/blog/2016/11/12/the-simplest-classifier-histogram-intersection.html

Loading the Image

Using OpenCV, Skimage, pyplot

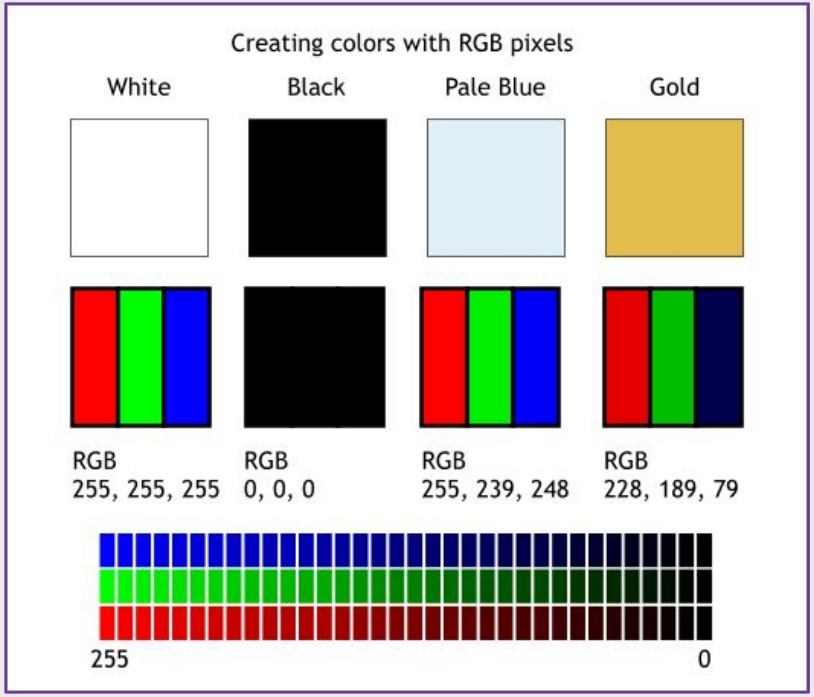
OpenCV	Skimage	pyplot

#load image
img = cv2.imread('images.jpg',-1)
 	
#show Image
cv2.imshow('image', img)
	#load image
	img = skimage.io.imread('images.jpg', 0)
	
	#show Image
	skimage.io.imshow(img)
		#load image
		iimg = plt.imread('images.jpg')
		
		#show Image
		plt.imshow(img)

IMPORTANT

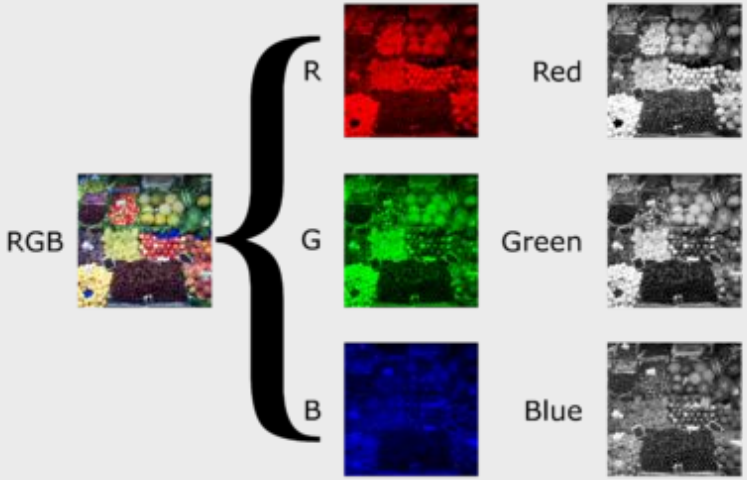
OpenCV Reads the Image using **BGR** format, while Skimage and pyplot use **RGB**

Color In Image



Colored Images Consist of **3 Channel : RGB**

Same Range value of pixel (8 bit) :
0 - 255



RGB to Gray

Grayscale =
((0.3 * R) + (0.59 * G) + (0.11 * B))

OpenCV :
cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

nodeflux

Extending Vision Beyond Imagination

Section 2 - Image Processing

www.nodeflux.io

MFM Workshop File

https://bit.ly/2Gpwf9k

WIFI

NODEFLUX - Guests
wearenodeflux

Why do we need Image Processing?

in order for the machine to understand the input image, the information must be extracted before being given to the classifier

General Pipeline of computer vision problem

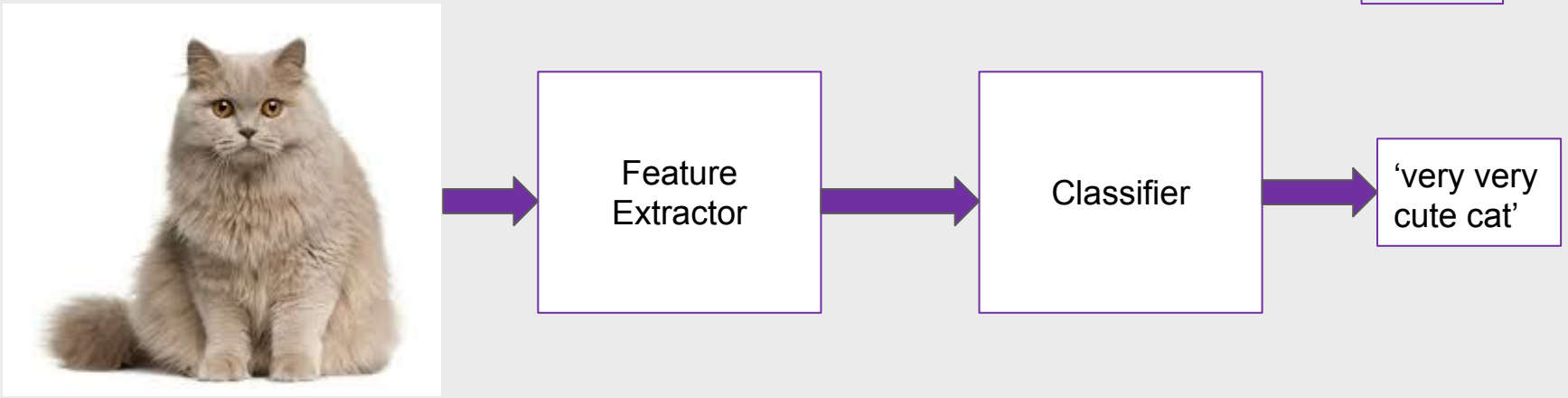


Image Segmentation: Thresholding

Image segmentation based on a simple threshold:

$$g[n,m] = \begin{cases} 255, & f[n,m] > 100 \\ 0, & \text{otherwise.} \end{cases}$$

Ref: CS131

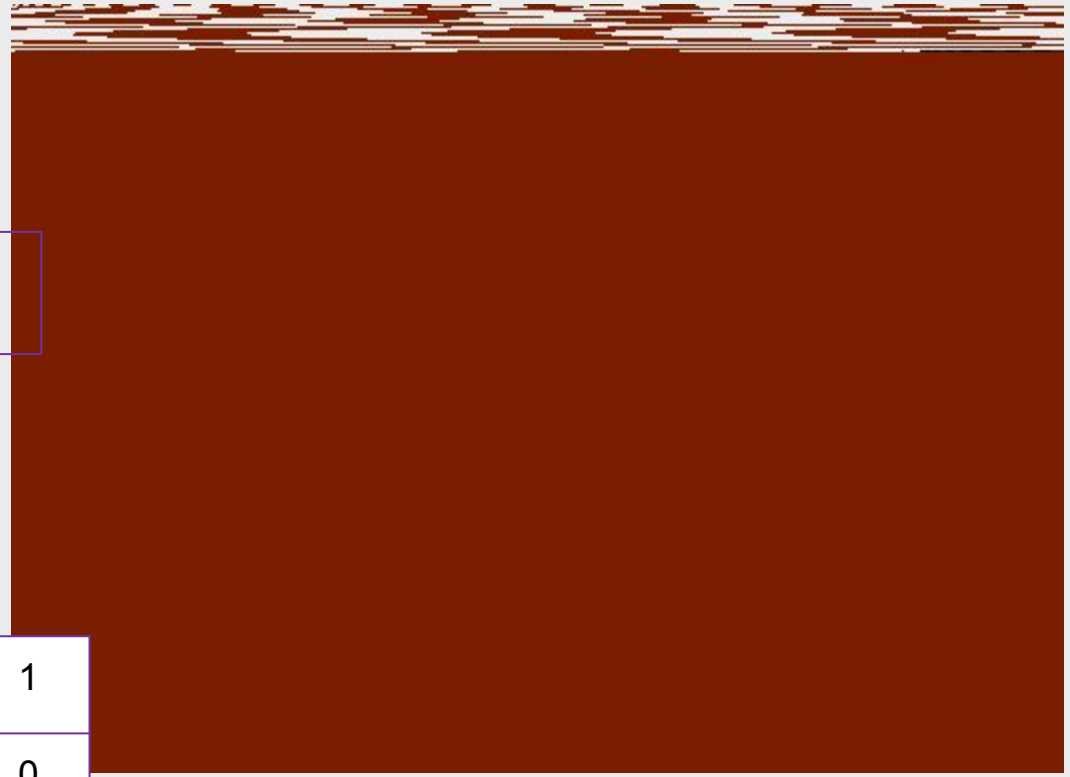
Basic Thresholding

an operation that converts an image into a binary image

Convolution Filters

System that uses information from neighboring pixels to filter the target pixel

Filter that convolve the kernel (convolutional matrix) to the image



Kernel		
1	0	1
0	1	0
1	0	1

Different Kernel Value, Different Result

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

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Kernel

Try it yourself!

Identity

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

bot sobel

$$\begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix}$$

emboss

$$\begin{pmatrix} -2 & -1 & 0 \\ -1 & 1 & 1 \\ 0 & 1 & 2 \end{pmatrix}$$

blur

$$\begin{pmatrix} 0.0625 & 0.125 & 0.0625 \\ 0.125 & 0.25 & 0.125 \\ 0.0625 & 0.125 & 0.0625 \end{pmatrix}$$

left sobel

$$\begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix}$$

outline

$$\begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix}$$

right sobel

$$\begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}$$

Sharpen

$$\begin{pmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

top sobel

$$\begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}$$

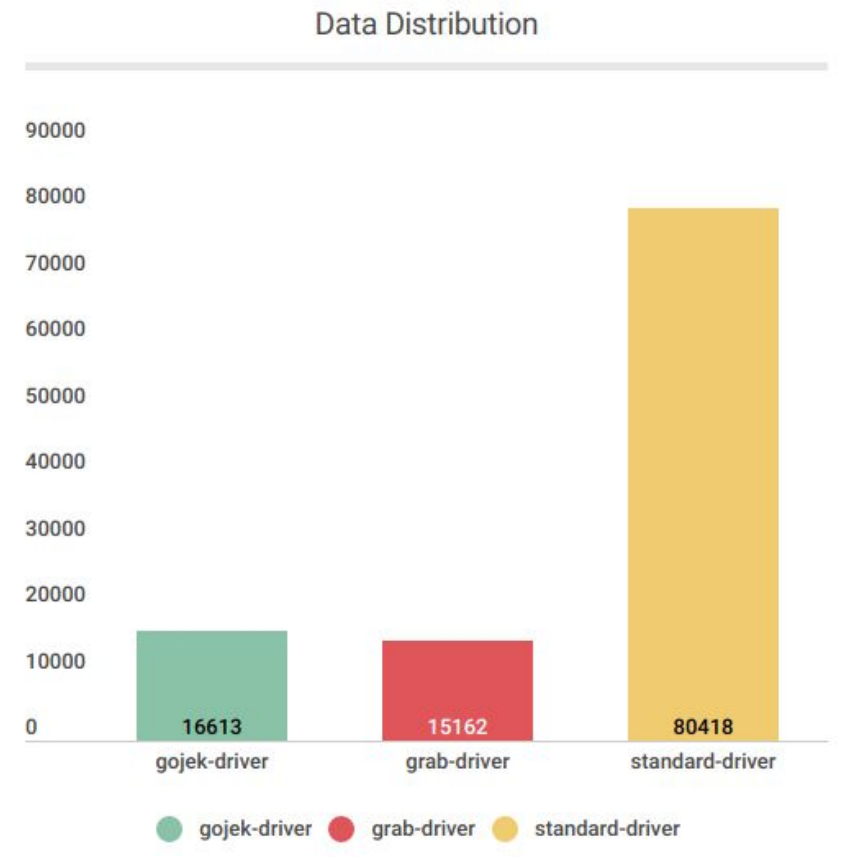
Short Summary

nodeflux

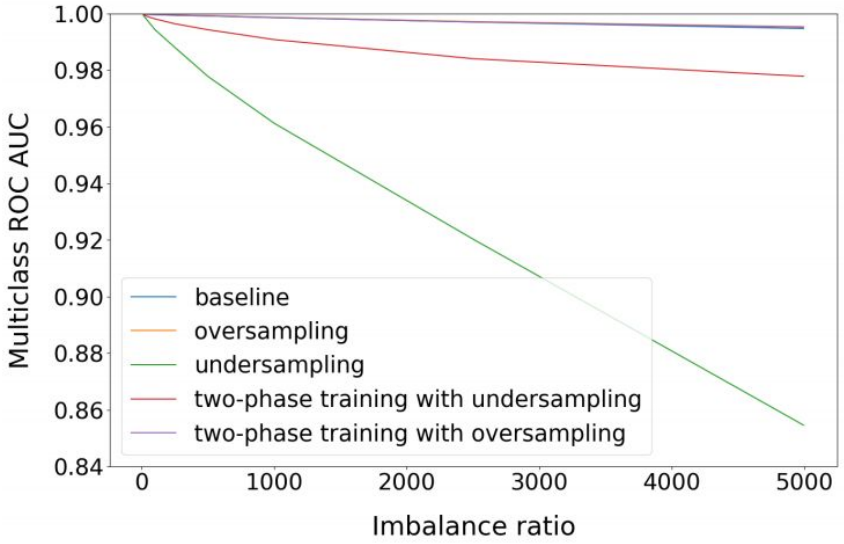
Extending Vision Beyond Imagination

Section 3 - Image Augmentation

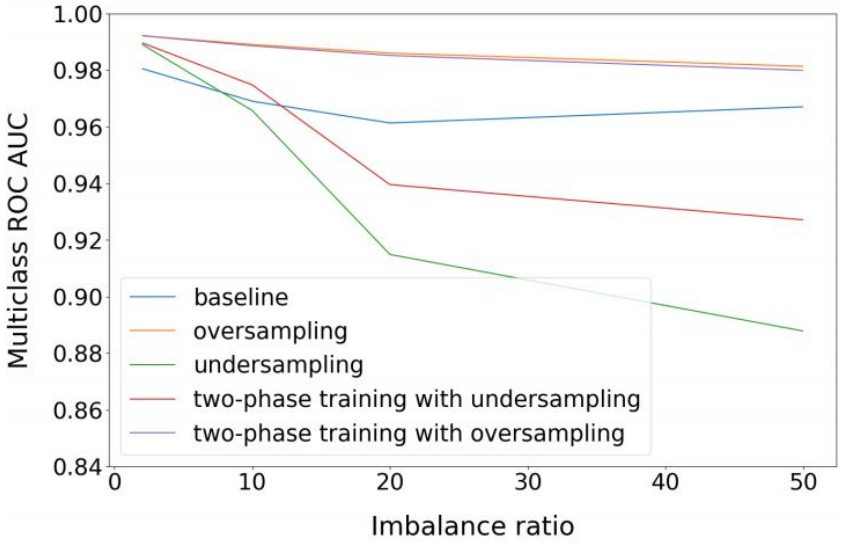
Imbalanced Dataset



Effect on ML/DL model training

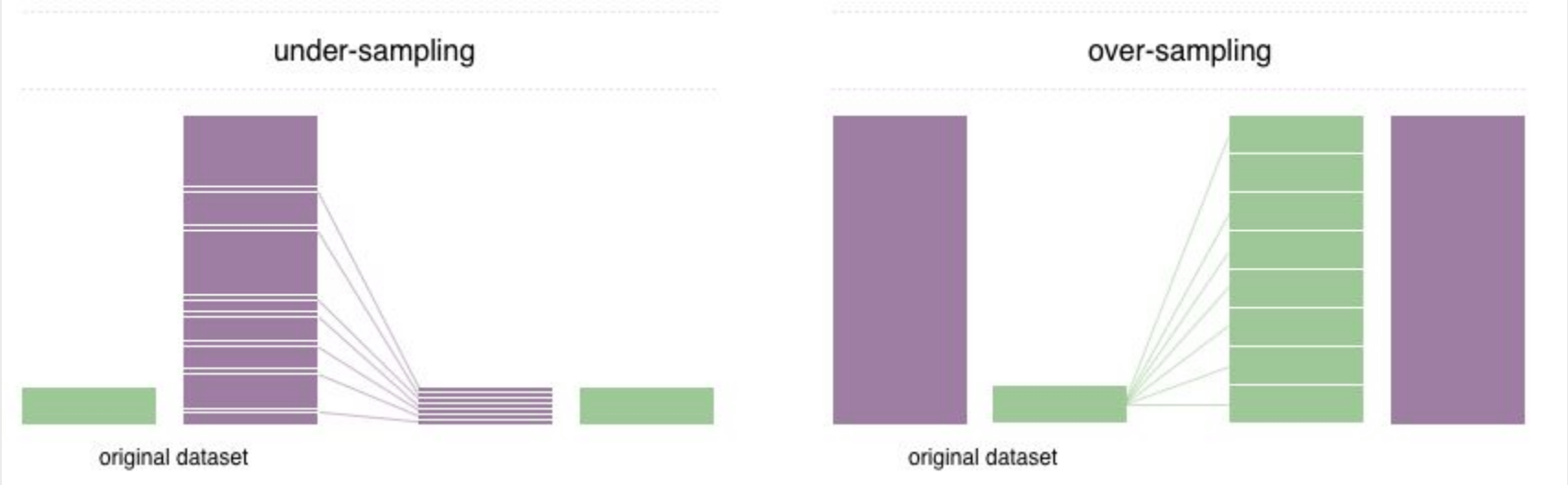


(a) MNIST

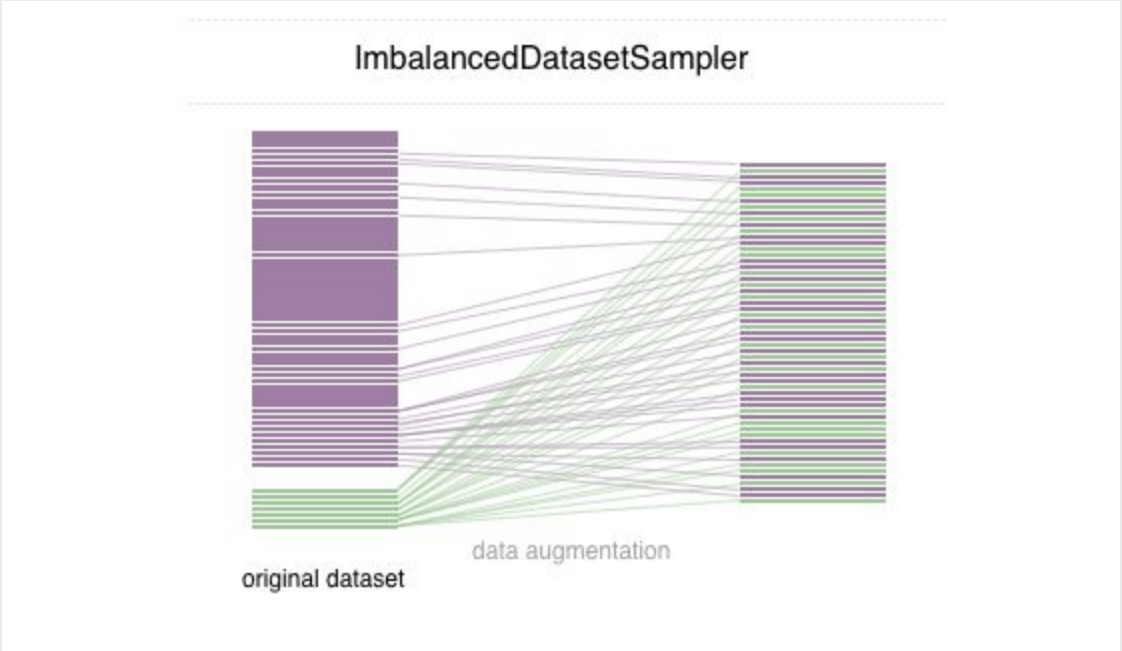


(b) CIFAR-10

How to handle?



How to handle? (on Image Dataset)



What is Image Augmentation?

Common Image Augmentation Method

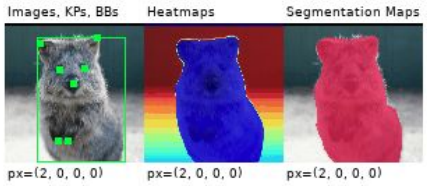
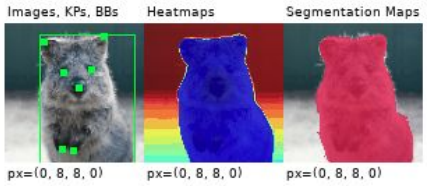
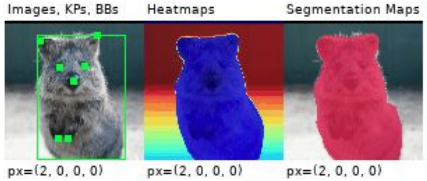
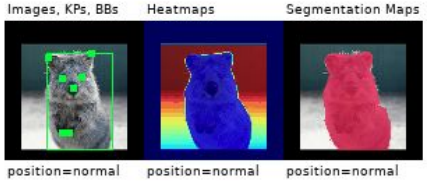
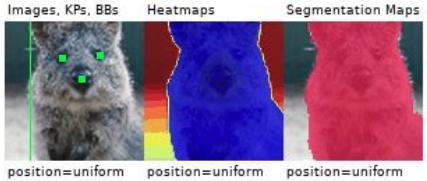
blur				
GaussianBlur	AverageBlur	MedianBlur	BilateralBlur (sigma_color=250, sigma_space=250)	MotionBlur (angle=0)
 sigma=0.25	 k=1	 k=1	 d=1	 k=3
MotionBlur (k=5)				
 angle=0				

Common Image Augmentation Method

color	
AddToHueAndSaturation	Grayscale
 value=-25	 alpha=0.2

Common Image Augmentation Method

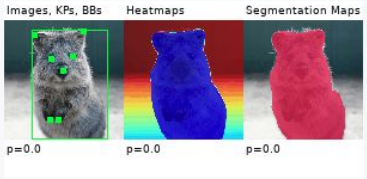
size

CropAndPad	Crop	
		
Pad	PadToFixedSize (height'=height+32, width'=width+32)	
		
CropToFixedSize (height'=height-32, width'=width-32)		
		

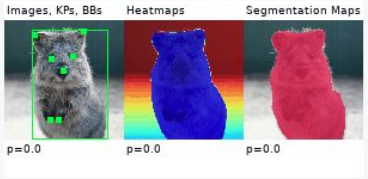
Common Image Augmentation Method

flip

Fliplr

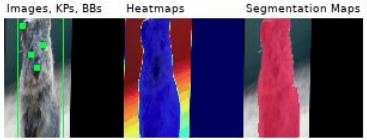


Flipud

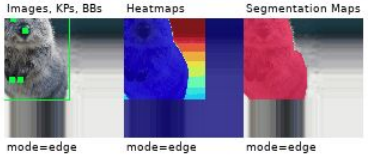


geometric

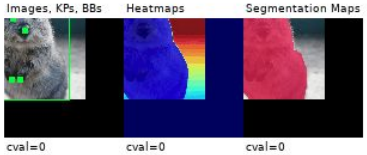
Affine



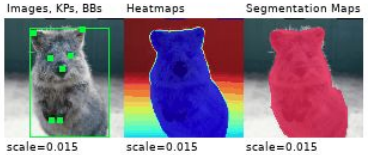
Affine: Modes



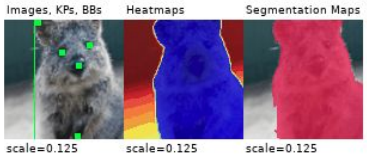
Affine: cval



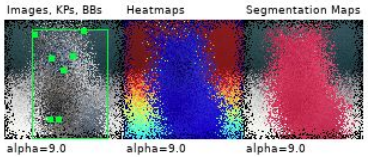
PiecewiseAffine



PerspectiveTransform



ElasticTransformation
(sigma=0.2)



Thank you

www.nodeflux.io