

Lecture note 3 : Image processing

프로젝트 기반 딥러닝 이미지처리
한국인공지능아카데미 x Hub Academy

강사 : 김형욱 (hyounguk1112@gmail.com)



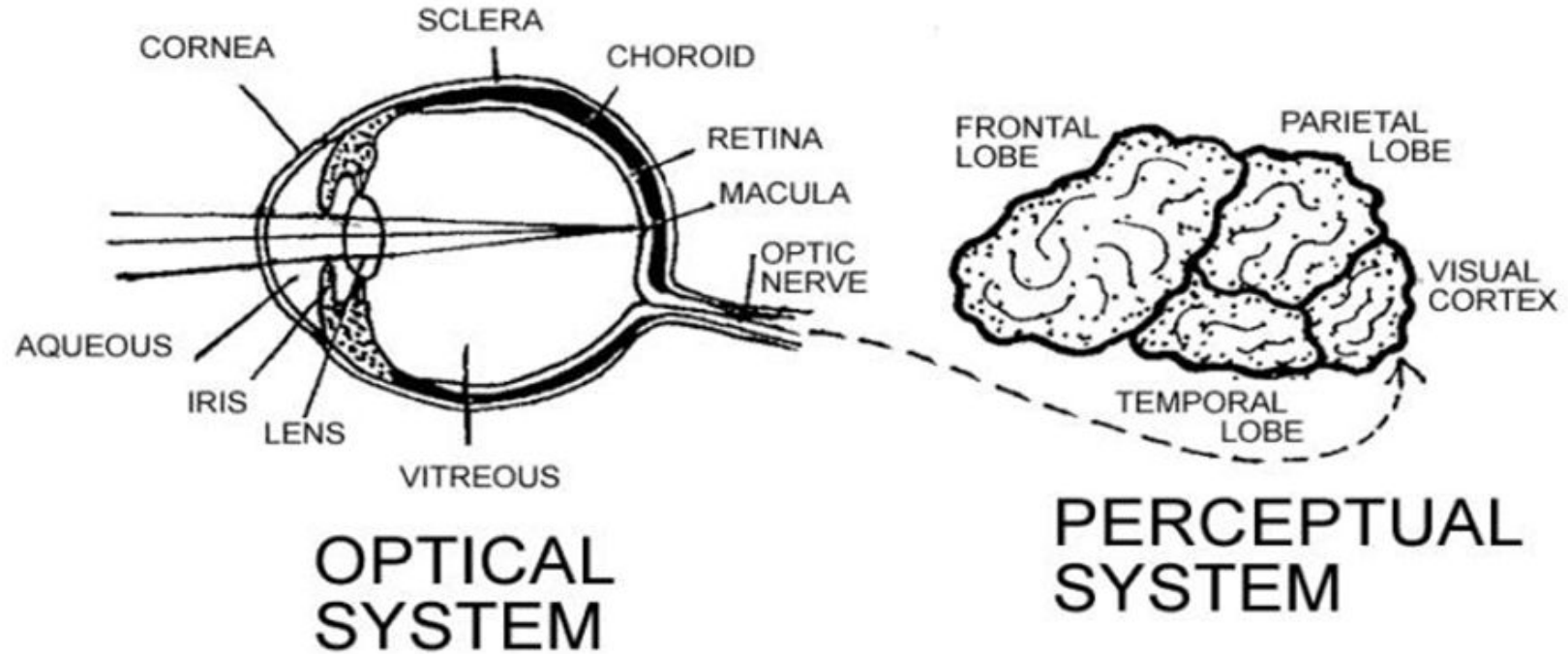
Agenda



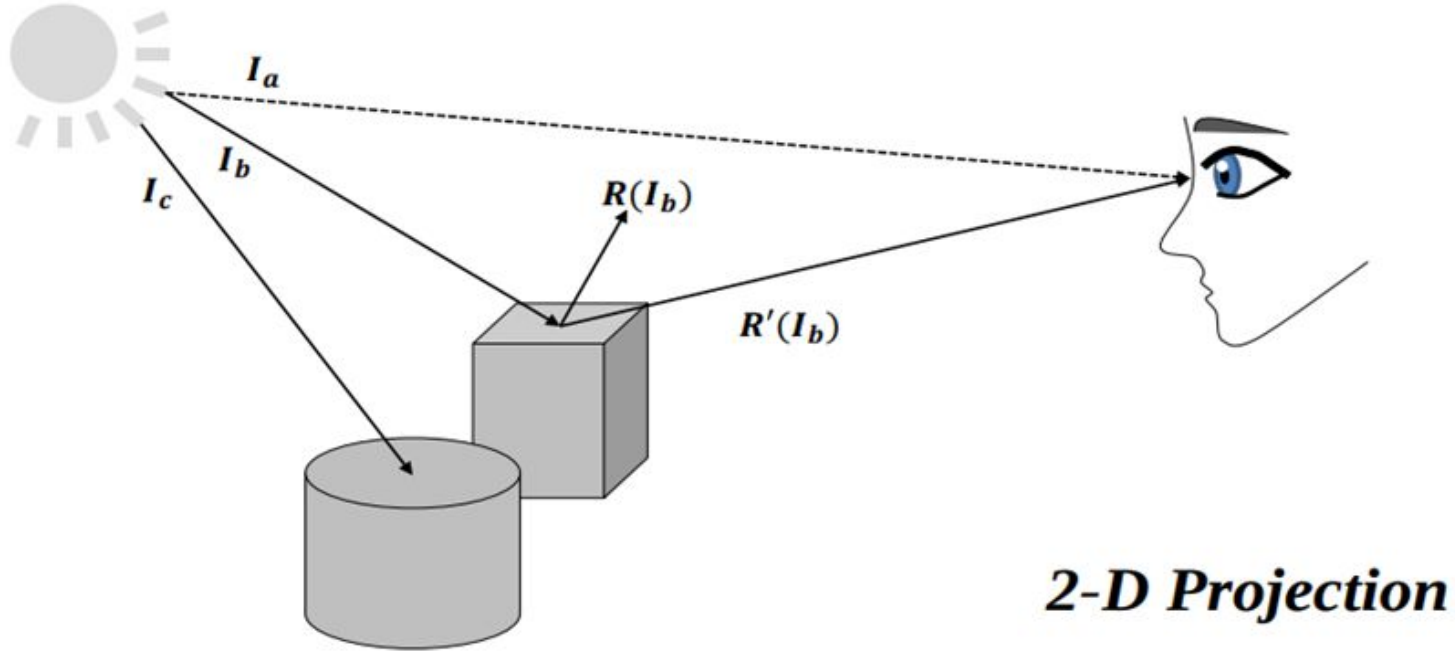
- 1 Computer Image
- 2 Image processing
- 3 Image feature
- 4 Practice : Color filtering

Computer Image

Human vision system



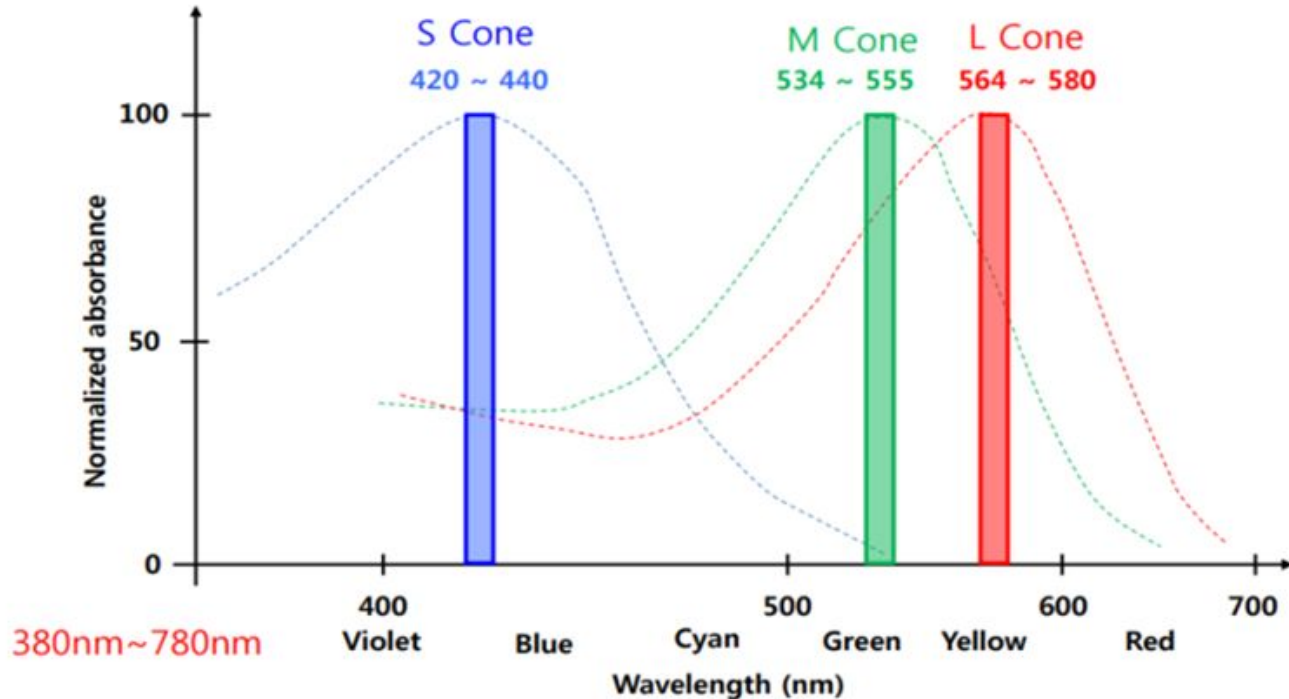
Human vision system



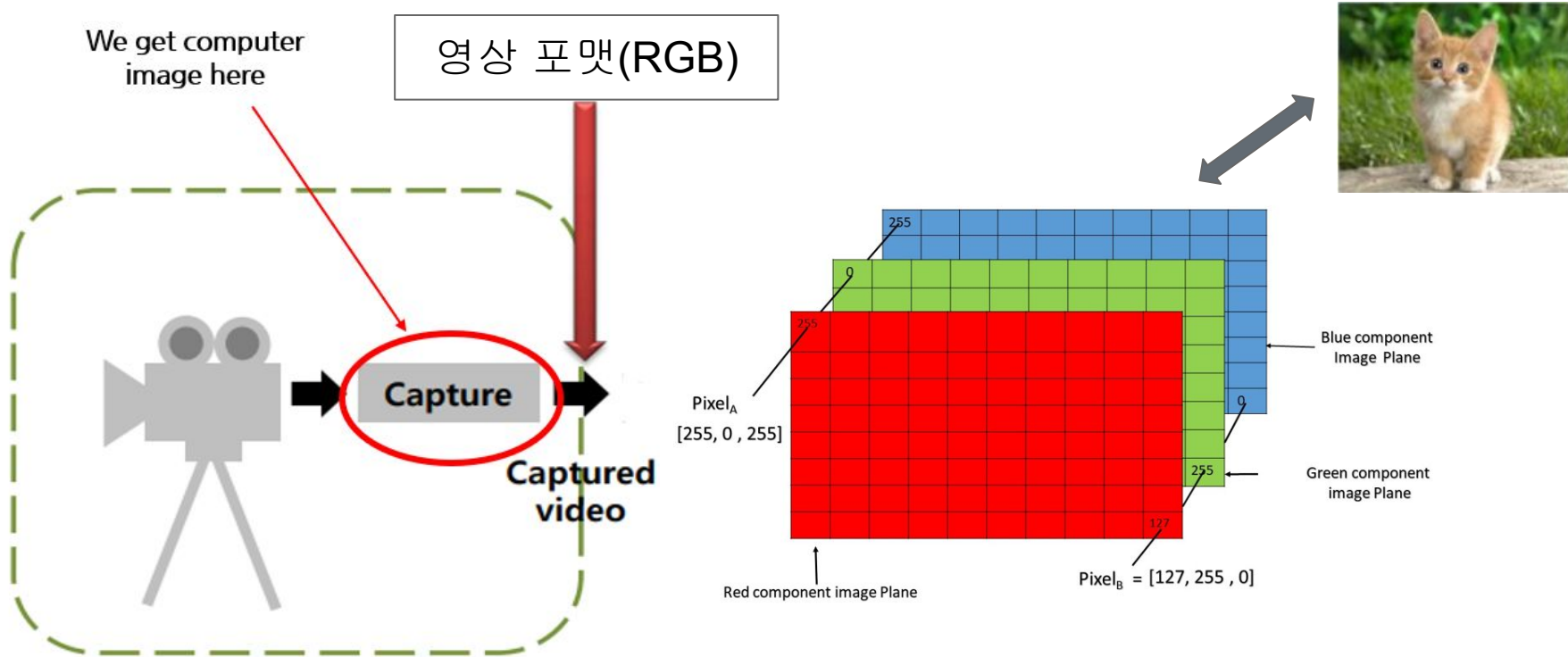
Human vision system

- **Rod and Cone**

- 간상세포(Rod): 약한 자극의 어두운 빛에도 반응
- 원추세포(Cone): 비교적 강한 자극의 밝은 빛에 반응 → 3 종류



Machine vision system



Bayer filter and Bayer pattern

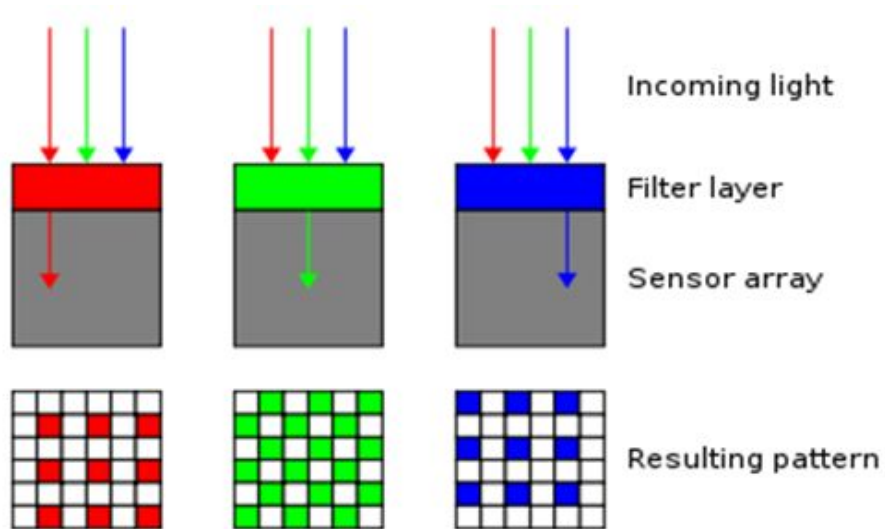


그림1 : 3가지 종류의 Bayer filter

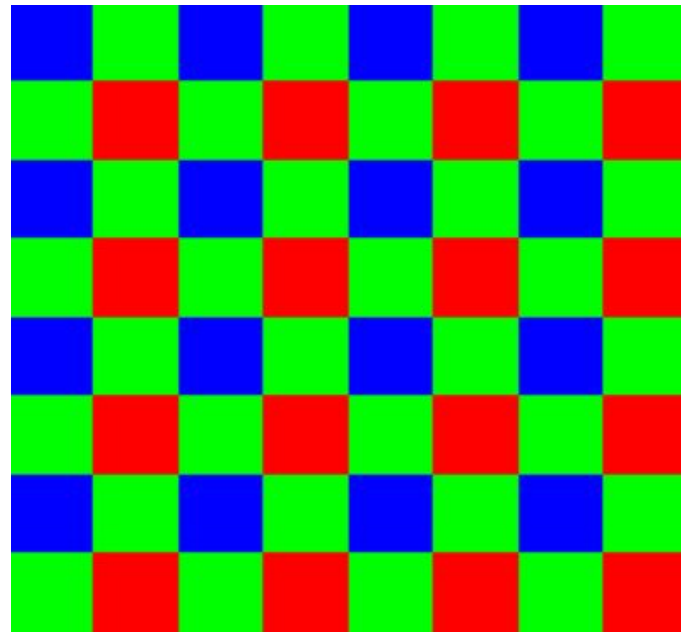
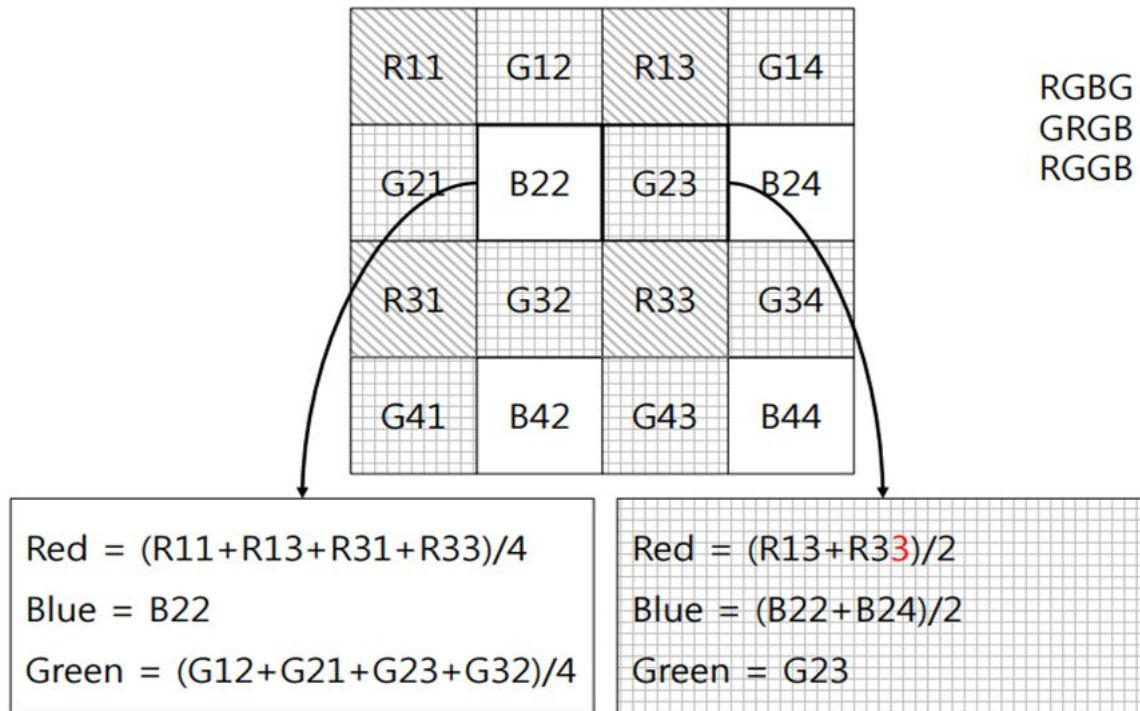


그림2 : Bayer pattern

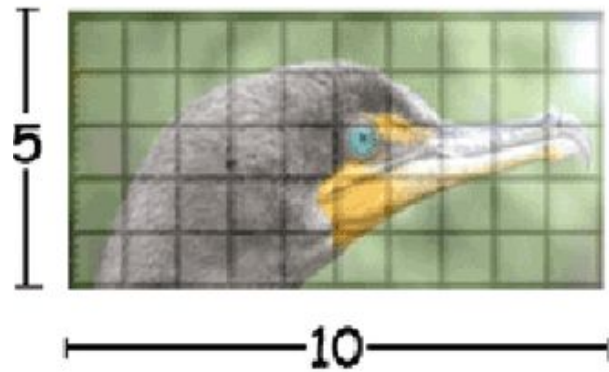
Bayer pattern : How to make RGB format?



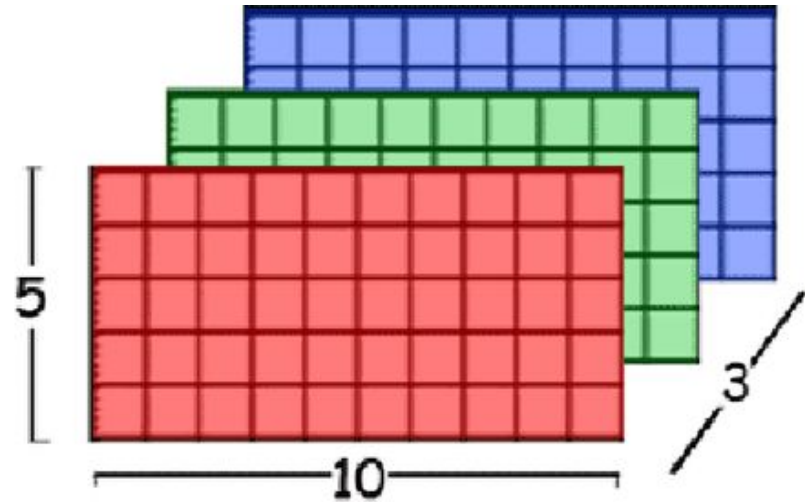
Bayer filter 의 보간법 (Interpolation)

Image representation in Computer!

1. RGB format



Original Color Image



Matlab RGB Matrix

Image representation in Computer!

1. RGB format

RGB of each pixel
= 3 values
= (R, G, B)

0.2235	0.1294	Blue	0.4196			
0.5804	0.2902	0.0627	0.2902	0.2902	0.4824	
0.5804	0.0627	0.0627	0.0627	0.2235	0.2588	
0.5176	0.1922	0.0627	Green	0.1922	0.2588	0.2588
0.5176	0.1294	0.1608	0.1294	0.1294	0.2588	0.2588
0.5176	0.1608	0.0627	0.1608	0.1922	0.2588	0.2588
0.5490	0.2235	0.5490	Red	0.7412	0.7765	0.7765
0.5490	0.3882	0.5176	0.5804	0.5804	0.7765	0.7765
0.5490	0.2588	0.2902	0.2588	0.2235	0.4824	0.2235
0.2235	0.1608	0.2588	0.2588	0.1608	0.2588	0.2588
0.2588	0.1608	0.2588	0.2588	0.2588	0.2588	0.2588



Image representation in Computer!

1. RGB format : color palette

- To define any color, specify (R, G, B) values.
- 0 is minimum and 1 (혹은 255) is maximum. / 16,777,216가지 색상 표현 가능
-



[0 0 0]



[1 1 1]



[0.29 0.29 0.29]



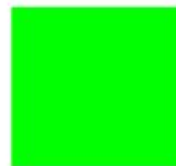
[0.78 0.78 0.78]



[1 0 0]



[0.39 0 0]



[0 1 0]



[0 0.39 0]



[0 0 1]



[0 0 0.39]



[1 1 0]



[0 1 1]



[1 0 1]

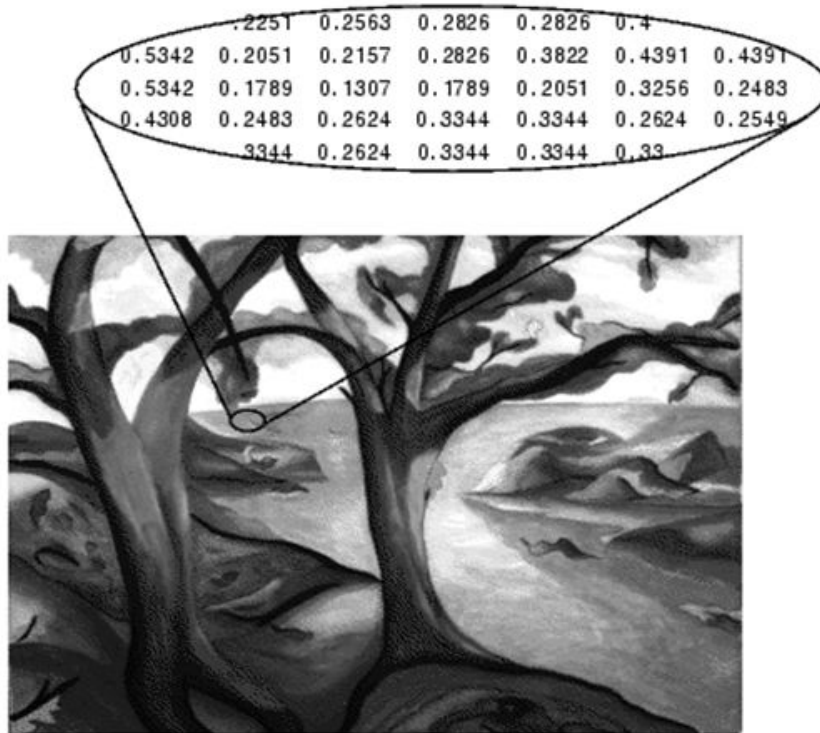


[0.93 0.49 0.19]

Image representation in Computer!

2. Gray format

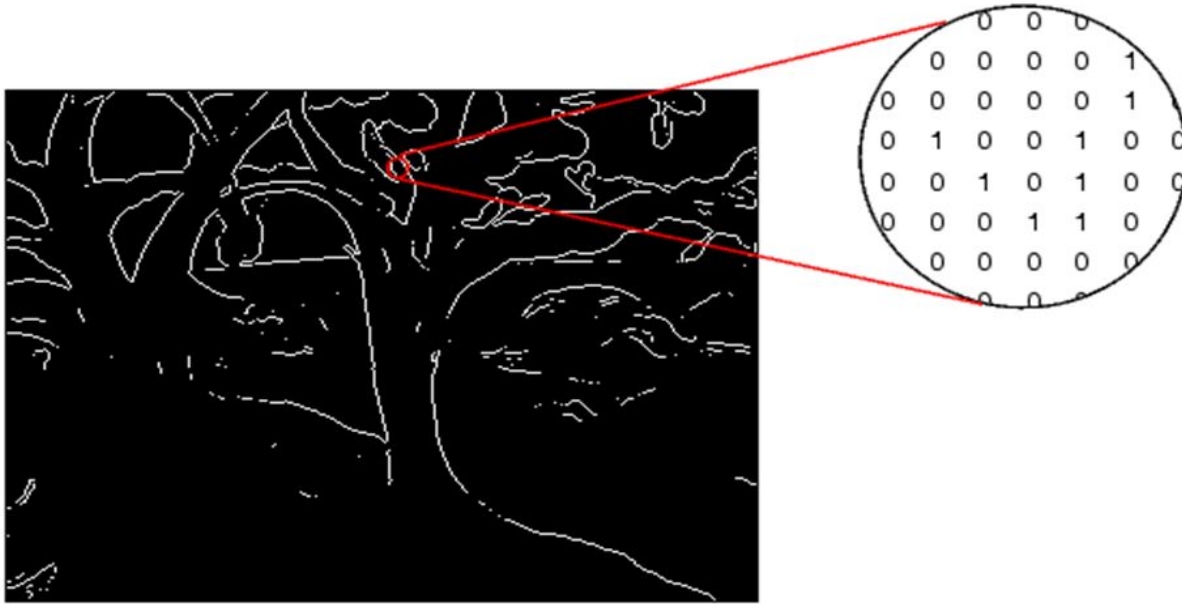
Gray of each pixel
= 1 values
= intensity of brightness



Pixel Values in an Intensity Image Define Gray Levels

Image representation in Computer!

2. Binary image (or mask image)



Pixels in a Binary Image Have Two Possible Values: 0 or 1

Image coordinate (좌표계)

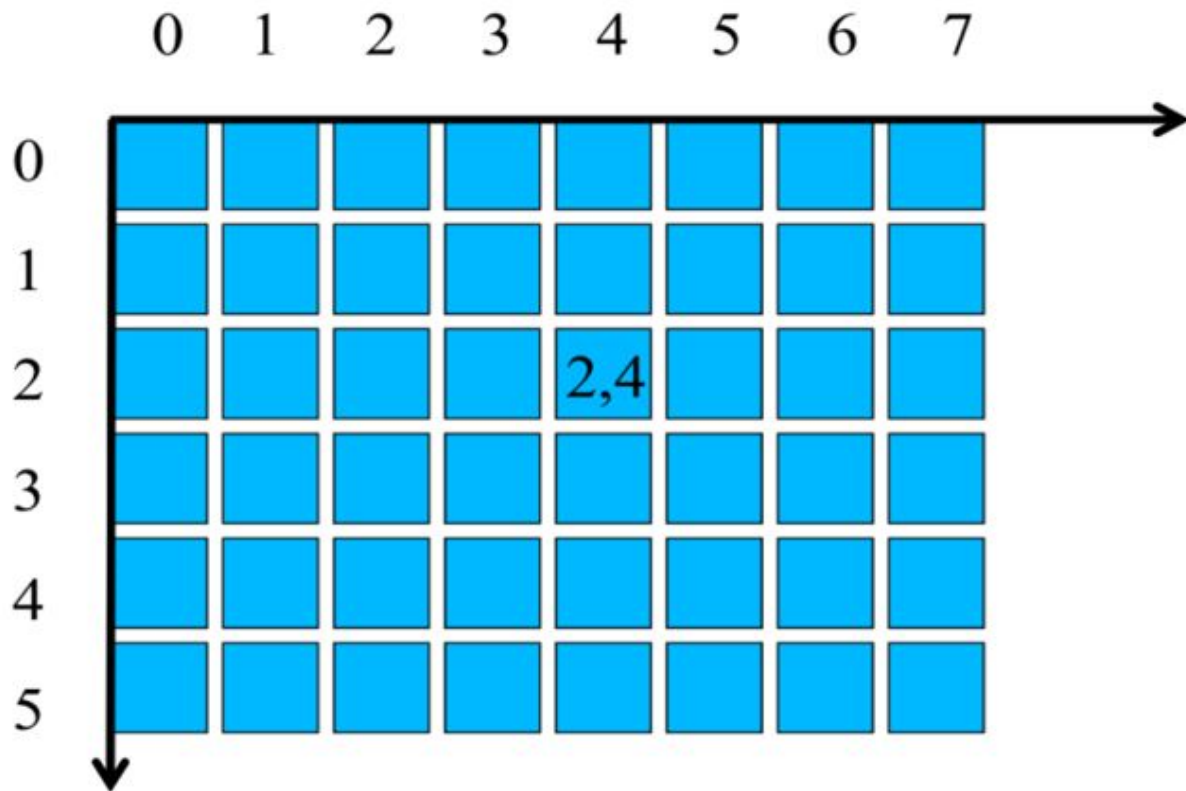
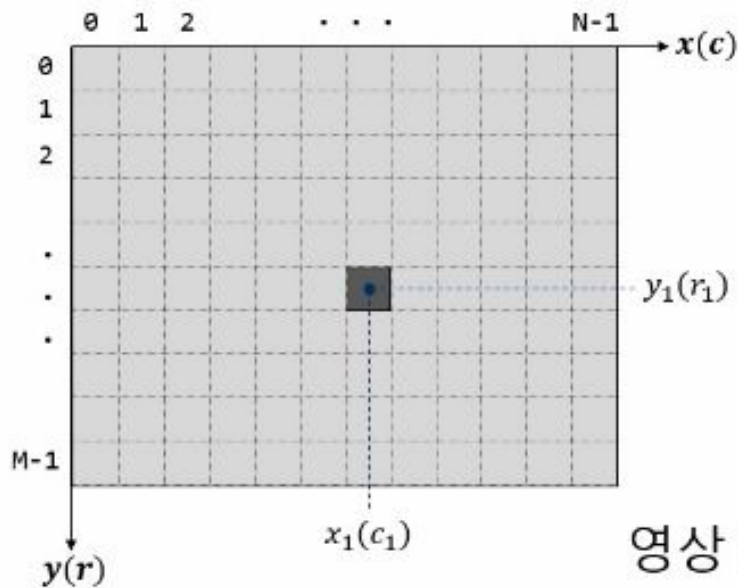


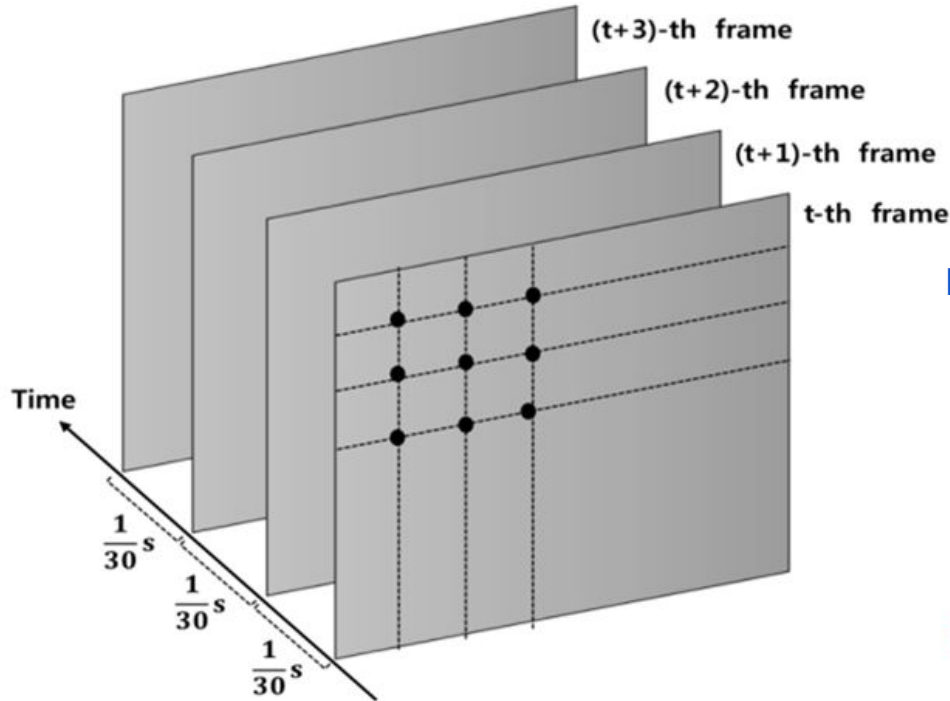
Image coordinate (좌표계)



영상 좌표 $\Rightarrow (x_l, y_l)$

행렬 위치 $\Rightarrow (r_l, c_l)$

Discrete Timeline



Frame per second(fps)

영화: $1/24$

BW NTSC TV: $1/30$

Color NTSC TV: $1/29.97$

PAL/SECAM TV: $1/25$

HDTV: $1/30$

UHDTV: $1/60, 1/120$

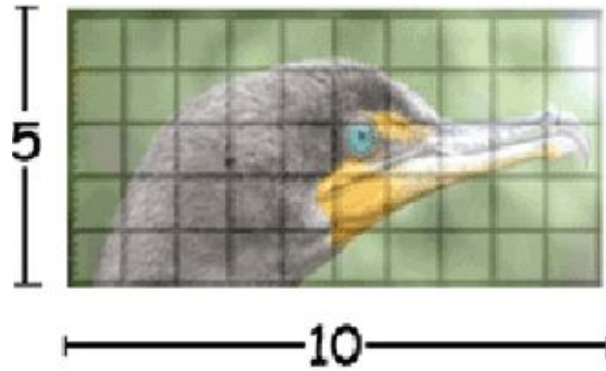
$$30 \times \frac{1000}{1001} \cong 29.97$$

동영상을 표현하는 시간 도메인 및 프레임 간격

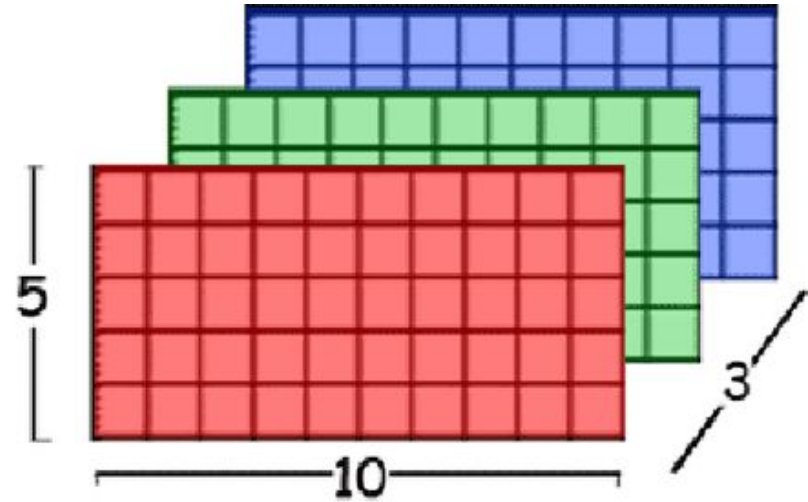
Discrete Timeline

이름	해상도	비율	설명
nHD	640×360	16:9	Full HD의 1/9 ^[21] . one ninth of a Full HD frame
qHD	960×540	16:9	Full HD의 1/4. one q uarter of Full- HD
HD	1280×720	16:9	Full HD의 4/9. 720p ^[22] , 720i
HD+	1600×900	16:9	HD Plus
FHD	1920×1080	16:9	Full HD ^[23] 1080p, 1080i
WQHD/QHD	2560×1440	16:9	HD의 4배. Quad HD/Quad HD, 2K ^[24] , 1440p, 1440i
QHD+	3200×1800	16:9	Quad HD Plus ^[25]
UHD	3840×2160	16:9	Full HD의 4배. Ultra HD, 4K ^[26] 4K UHD
UHD+	5120×2880	16:9	Ultra HD Plus, 5K ^[27]
FUHD	7680×4320	16:9	Ultra HD의 4배. Full Ultra HD, 8K ^[28] 8K UHD

Digital Image : RGB format



Original Color Image



Matlab RGB Matrix

Image processing

영상처리(Image processing)

- 영상이 가진 노이즈를 줄이거나, 분석을 위해 단순화(불필요한 정보 제거)하는 과정
- 절차적 / 병렬적으로 한 가지 프로그램에서 여러가지 영상처리방법이 사용될 수 있음.
- 딥러닝 방식의 AI 프로그램 이전에만 사용되는 것(preprocessing)은 아니다. 반대로 딥러닝을 사용해준 후 원하는 특정 정보를 얻기 위해 후처리(postprocessing)로 사용되는 경우도 많다.
- Opencv는 대표적으로 사용되는 영상처리/분석 라이브러리로, C++/Python 등의 프로그래밍 언어를 지원한다.

Geometric transform of Images

변환(Transform)이란 수학적으로 표현하면 아래와 같습니다.

“좌표 x 를 좌표 x' 로 변환하는 함수”

예로는 사이즈 변경(Scaling), 위치변경(Translation), 회전(Rotation) 등이 있습니다.

변환의 종류에는 몇가지 분류가 있습니다.

- 강체변환(Rigid-Body) : 크기 및 각도가 보존(ex; Translation, Rotation)
- 유사변환(Similarity) : 크기는 변하고 각도는 보존(ex; Scaling)
- 선형변환(Linear) : Vector 공간에서의 이동. 이동변환은 제외.
- Affine : 선형변환과 이동변환까지 포함. 선의 수평성은 유지.(ex; 사각형->평행사변형)
- Perspective : Affine변환에 수평성도 유지되지 않음. 원근변환

Geometric transform of Images

1. Scaling(or resizing, 크기 변화)
2. Translation(영상 이동)
3. Rotation(영상회전)
4. Geometric transform
 - Affine
 - Perspective

Cropping

- 영상처리 중 가장 빈도수가 높은 처리 방법 중 하나이다.
- 영상 내 노이즈를 배제하고, 분석 대상을 좁혀 연산량과 성능을 증가시킬 수 있다.
- 자동화할 시 **Padding**이 필요할 수 있다.



Color filtering

- Cropping과 함께 영상처리 중 가장 빈도수가 높은 처리 방법 중 하나이다.
- HSV 값들을 사용하여, **object**에 대한 색의 범위를 정해주고, **filtering**한다.

Tip : color filtering을 하여 **object**를 구하는 것이 아니라, **object**를 color filtering하기 좋은 것이 좋은 응용인 경우가 많다.



Image feature

Why we need image features

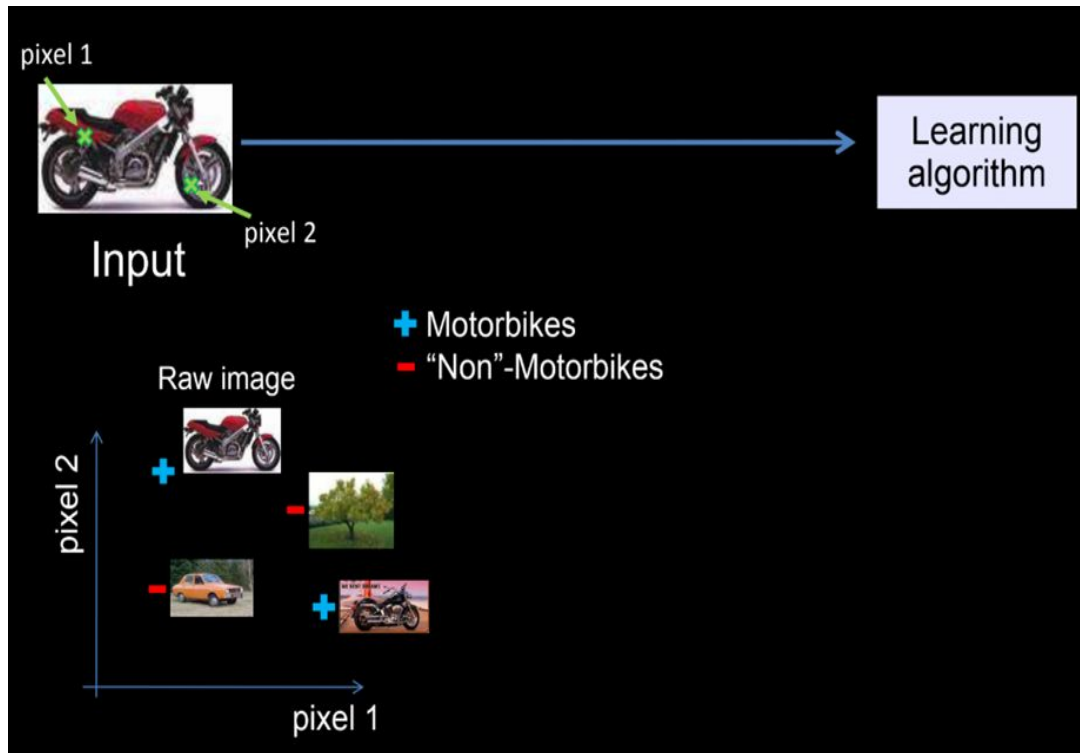
You see this:



But the camera sees this:

194	210	201	212	199	213	215	195	178	158	182	209
180	189	190	221	209	205	191	167	147	115	129	163
114	126	140	188	176	165	152	140	170	106	78	88
87	103	115	154	143	142	149	153	173	101	57	57
102	112	106	131	122	138	152	147	128	84	58	66
94	95	79	104	105	124	129	113	107	87	69	67
68	71	69	98	89	92	98	95	89	88	76	67
41	56	68	99	63	45	60	82	58	76	75	65
20	43	69	75	56	41	51	73	55	70	63	44
50	50	57	69	75	75	73	74	53	68	59	37
72	59	53	66	84	92	84	74	57	72	63	42
67	61	58	65	75	78	76	73	59	75	69	50

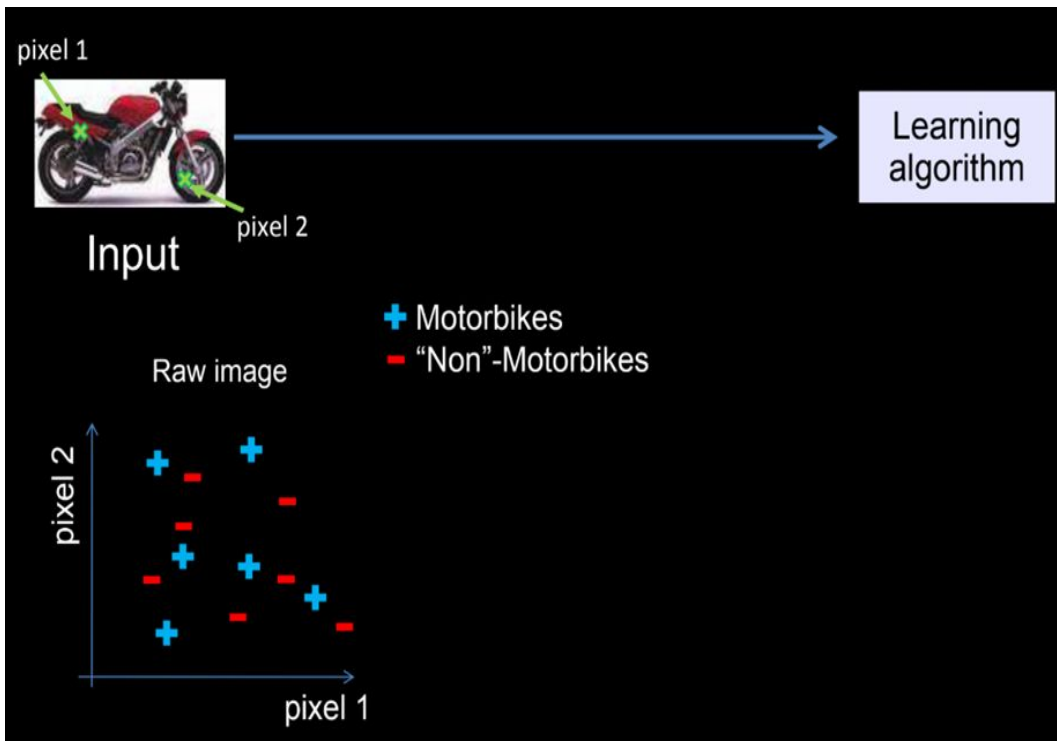
Why we need image features



기본적으로 분류는 데이터가 가지고 있는 값들을 사용한다.

영상의 경우도 그럴까?

Why we need image features

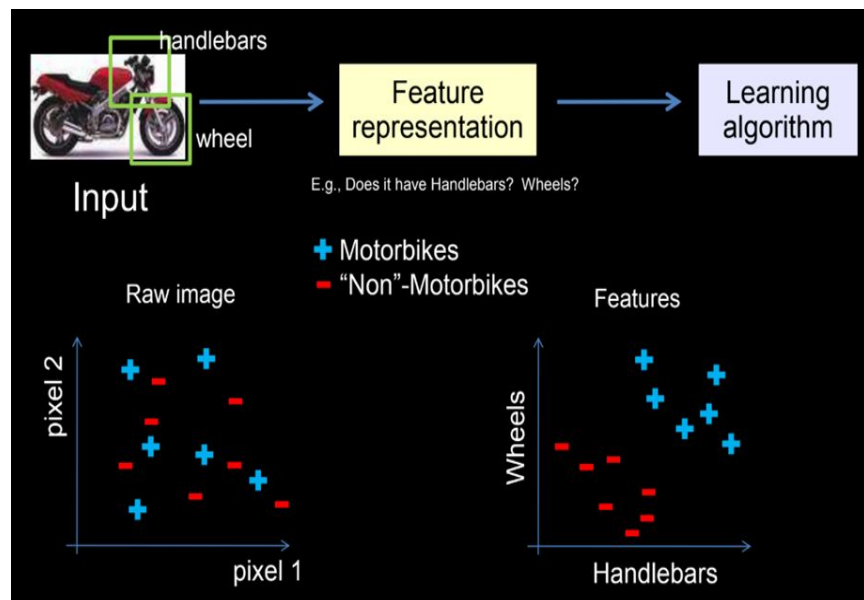
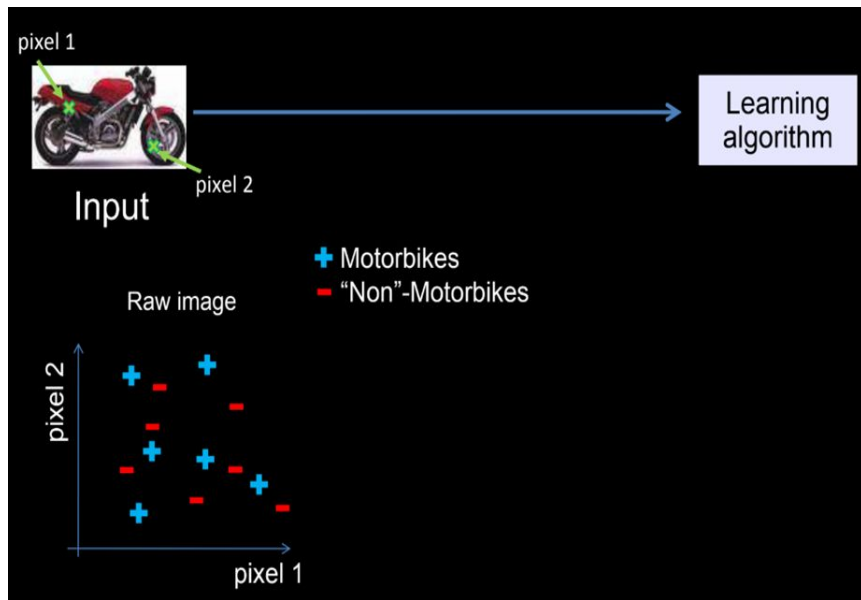


기본적으로 분류는 데이터가 가지고 있는 값들을 사용한다.

영상의 경우도 그럴까?

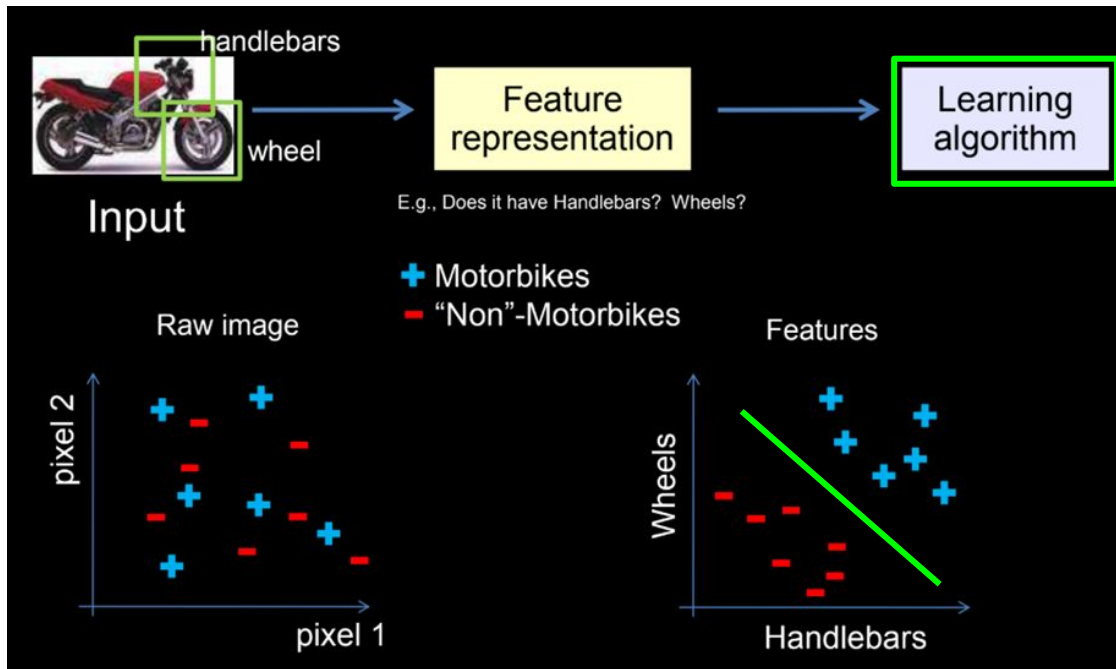
영상의 개별 픽셀값들을 통한 분류는 매우 어렵다!

Why we need image features



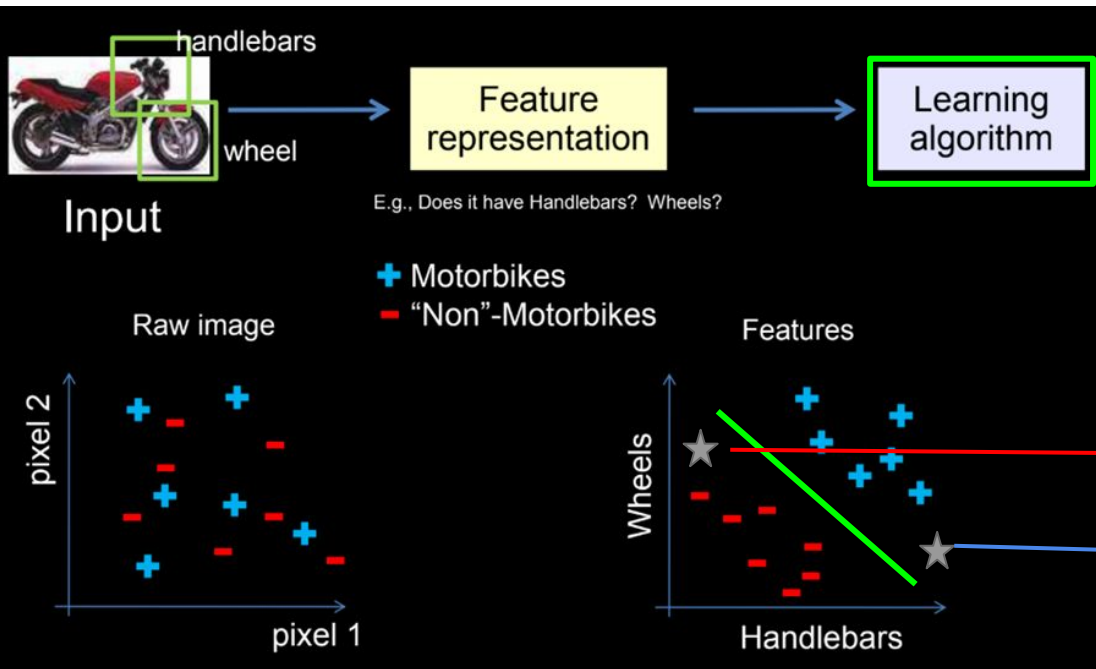
각 클래스의 객체는 공통적으로 가지고 있는 하위 개념들을 통해 구성됨(추상화 개념)
만일 분류 문제를 풀기 위한 하위 개념들을 찾아낼 수 있다면 픽셀 단위로는 해결하기
어려웠던 문제를 풀 수 있다.

Why we need image features



각 영상(data)를 하위 개념들의 조합(new representation)으로 나타낼 수 있다면,
머신러닝 알고리즘을 통해 더 나은 성능을 보이는 분류 모델을 학습할 수 있다.

Why we need image features



이러한 하위 개념을 **Feature**라고 부르며, 새로운 **representation**을 바탕으로 머신러닝 모델을 학습.

이 경우 모터바이크 여부를 판단할 수 있는 분류 모델

predict : It's not a motorbike

predict : It's a motorbike

Convolutional filter(1)

1. **영상의 특징**을 분석하기 위해서는 픽셀 단위가 아닌 픽셀들이 이루는 상관성(패턴)을 분석할 필요가 있다.
2. 영상과 같은 행렬의 지역 상관성을 분석하는데 적합한 **컨볼루션 필터(Convolutional Filter)**가 유용한 분석 도구가 될 수 있음.
3. 컨볼루션 필터는 **가중치 공간 필터(Weighted spatial filter)**로 가중치 값에 따라 다양한 영상처리가 가능함.

Convolutional filter(2)

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9



Averaging filter
Blurring

0	-1	0
-1	5	-1
0	-1	0



Opposite filter
sharpening

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

Horizontal

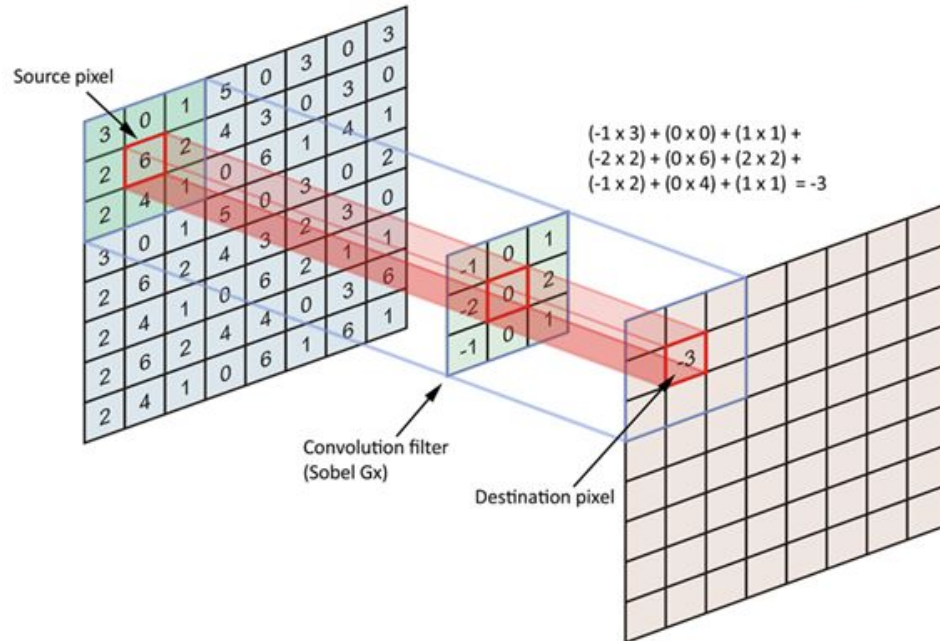
-1	0	1
-2	0	2
-1	0	1

Vertical



Sobel filter
Edge detecting

Calculation of Convolutions(1)



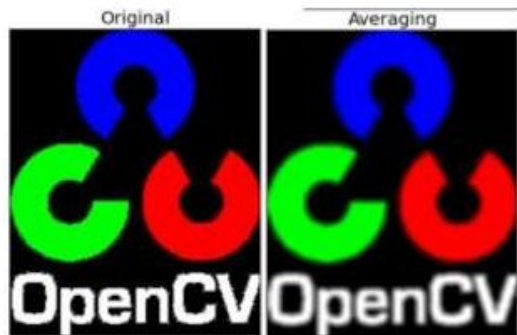
Region moves on a original image and make a processed new image. The movement of Region is called window sliding.

Calculation of Convolutions(2)

3_0	3_1	2_2	1	0
0_2	0_2	1_0	3	1
3_0	1_1	2_2	2	3
2	0	0	2	2
2	0	0	0	1

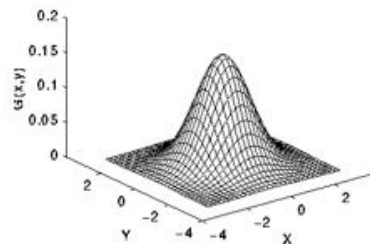
12.0	12.0	17.0
10.0	17.0	19.0
9.0	6.0	14.0

Example : Averaging filter



$$K = \frac{1}{25} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

<5x5 average filter>



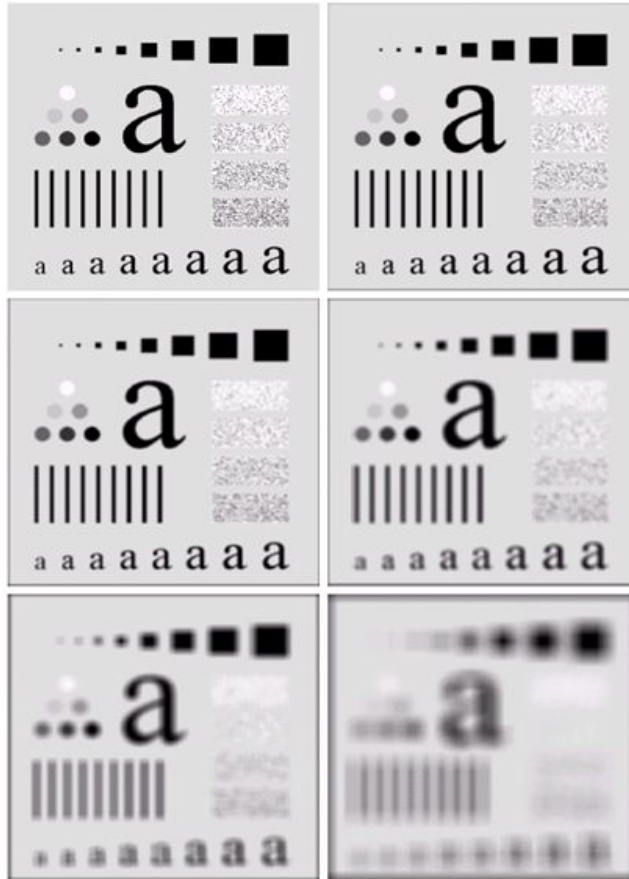
<Gaussian filter>

Example : Averaging filter

$$\frac{1}{9} \times \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

$$\frac{1}{16} \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

(1) Averaging filter



Averaging window Sizes :

원본	3
5	9
15	35

(2) Sobel filter(edge feature)

Original image



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

G_x



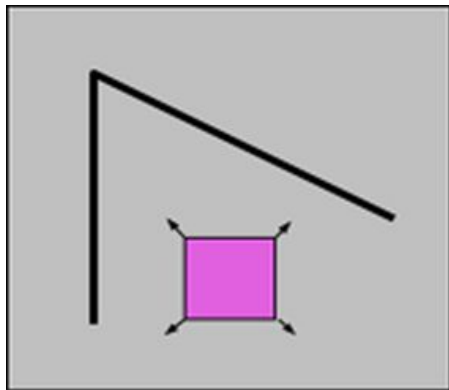
1	2	1
0	0	0
-1	-2	-1

G_y

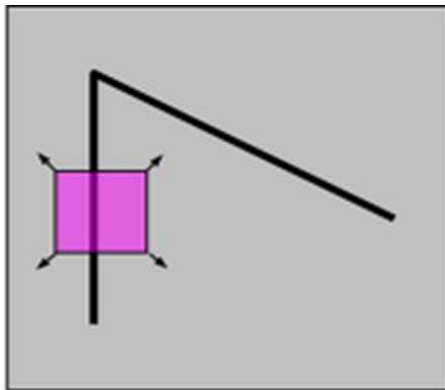


Result of Sobel operator

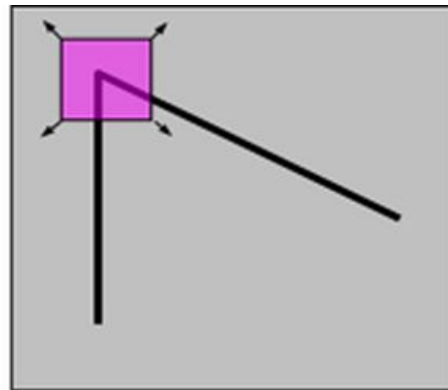
(3) harris corner detector(corner feature)



“flat” region:
no change in all
directions

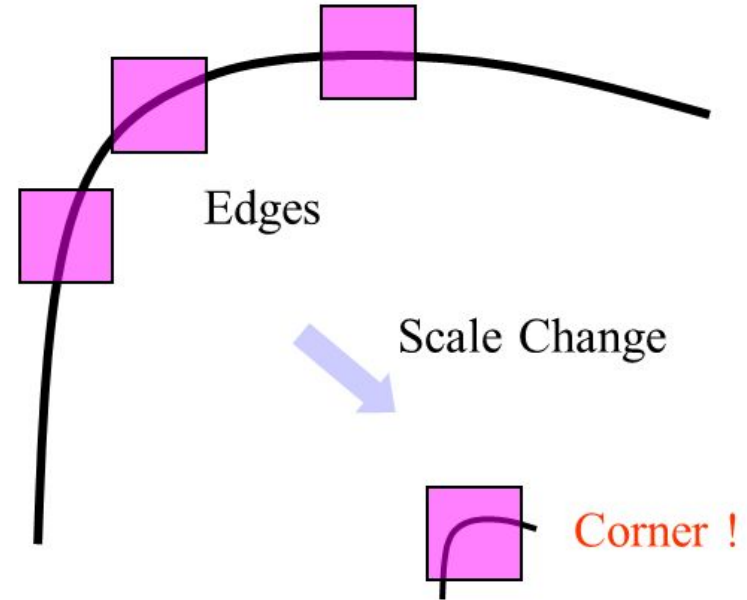


“edge”:
no change along the
edge direction



“corner”:
significant change in
all directions

(3) harris corner detector(corner feature)

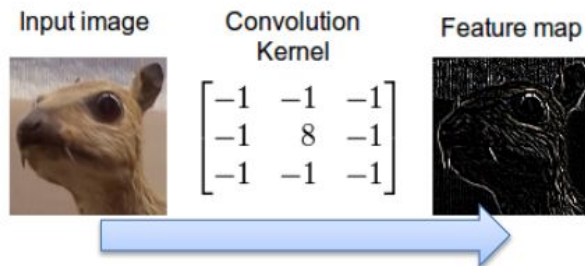


Convolutional filter(3)

컨볼루션의 다양한 이름

- 영상으로부터 불필요한 정보를 제거할 때 : **Filter**
- 주어진 문제에 적합한 정보를 얻기 위할 때 : **Feature**
- 동일한 사이즈의 영상 패치(patch)를 하나의 값으로 치환할 때 : **Kernel**

올바른 컨볼루션 필터의 가중치를 선택하는 것은 성공적인 분석에 가장 중요함 => 어떻게 최적의 가중치를 찾을까?

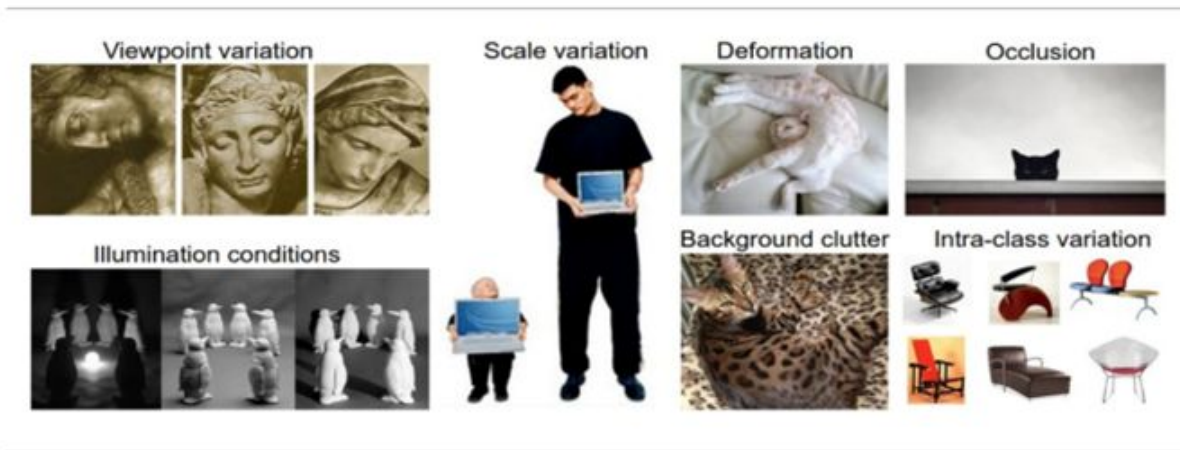


What is good feature?

A prominent part or characteristic of an image

Why?

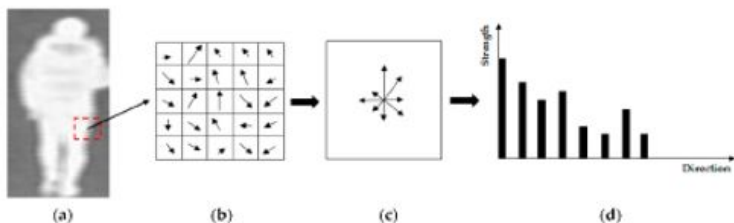
images of only one object could be different with scale, rotation, lighting, perspective imaging, partial occlusion



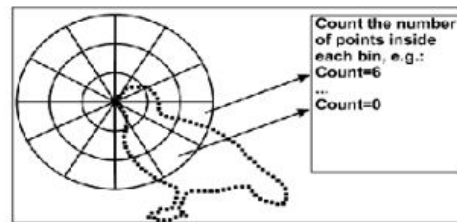
What is good feature?

Well-designed features

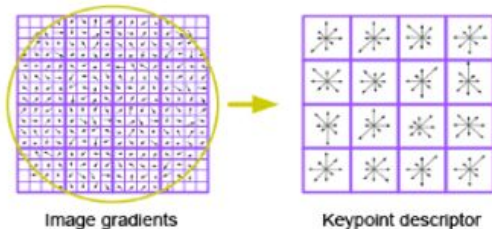
HoG (Histogram of Gradients)



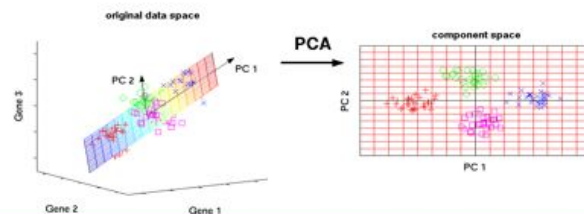
Shape Context



SIFT (Scale-Invariant Feature Transform)



PCA (Principal Components Analysis)



Thank you!