1. Introduction

Course review

<table>
<tr><th colspan="2">일차</th><th>구분</th><th>세부 내용</th><th>실습</th></tr>
<tr><td rowspan="6">1일차</td><td rowspan="3">오전</td><td>1. Introduction</td><td>Course Intro</td><td></td></tr>
<tr><td rowspan="2">2. Basic architecture</td><td>DNN</td><td></td></tr>
<tr><td>CNN and variants</td><td>ResNet</td></tr>
<tr><td rowspan="3">오후</td><td rowspan="2">3. Attention</td><td>Transformer</td><td></td></tr>
<tr><td>Vision Transformer and variants</td><td>ViT</td></tr>
<tr><td rowspan="3">4. Applications</td><td>Detection</td><td>DETR</td></tr>
<tr><td rowspan="6">2일차</td><td rowspan="3">오전</td><td>Tracking</td><td></td></tr>
<tr><td>Segmentation</td><td>U-net</td></tr>
<tr><td rowspan="3">5. Generative models</td><td>GAN, Latent representations</td><td></td></tr>
<tr><td rowspan="3">오후</td><td>Diffusion</td><td>DDPM</td></tr>
<tr><td>Text-to-Image, Latent diffusion</td><td>Stable diffusion</td></tr>
<tr><td>6. Closing</td><td>Course review</td><td></td></tr>
</table>

Course review

- **Lecture materials**
  - Contents
    - Technology trend & concepts
    - In-depth study on key papers
  - English
    - Clarity of meaning
  - Math & Equations
    - Understanding

- **Practice materials**
  - Implementations of 6 key papers
  - Pytorch
  - Jupyter notebook
    - Explanation in markdown and comments
    - Utilization of public LLMs

- **Test**
  - Code in the practice material

The Evolution of Computer Vision

https://en.wikipedia.org/wiki/Sobel_operator
https://en.wikipedia.org/wiki/Canny_edge_detector
https://gaussian37.github.io/vision-concept-optical_flow/
https://ics.uci.edu/~majumder/VC/211HW3/vlfeat/doc/overview/sift.html
https://learnopencv.com/support-vector-machines-svm/
https://medium.datadriveninvestor.com/haar-cascade-classifiers-237c9193746b

The early era of computer vision (1960 ~ 2010)

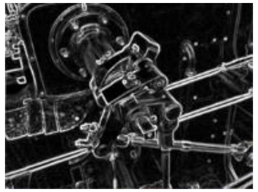
- **Feature engineering era**
    - Manually designed features

Early attempts

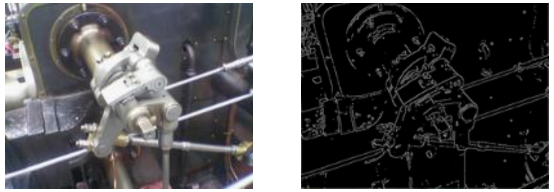
- Sobel filter

$$\mathbf{G}_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * \mathbf{A}$$

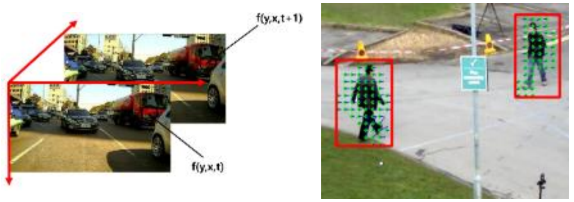
$$\mathbf{G}_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} * \mathbf{A}$$



- Canny edge detector

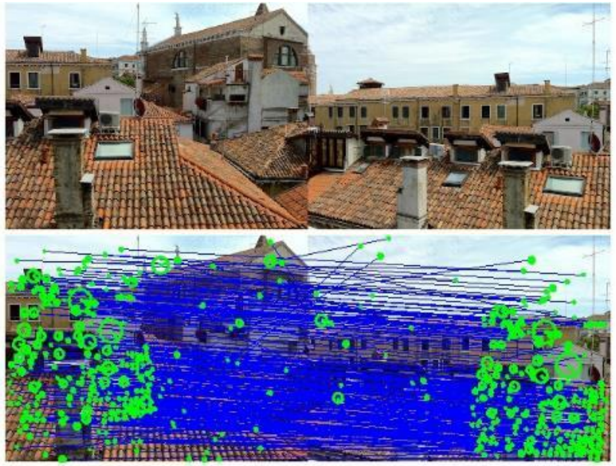


- Optical flow



Pattern recognition and feature engineering

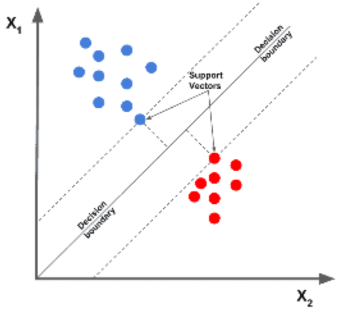
- SIFT (Scale-Invariant Feature Transform)



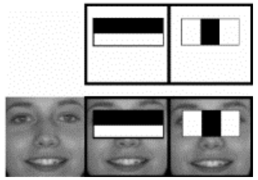
- HOG (Histograms of Oriented Gradient)

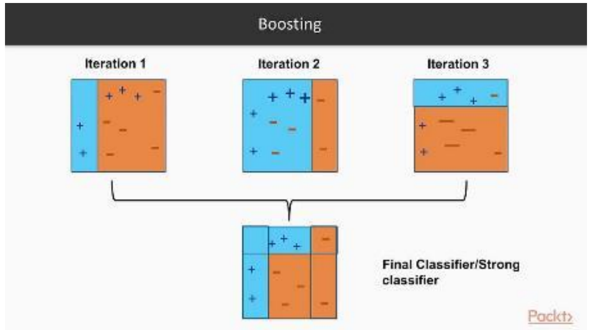
Machine learning-based Vision

- SVM (Support Vector Machine)



- Adaboost





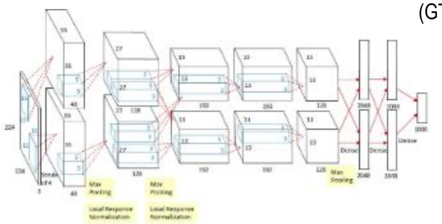
Deep learning revolution (2011 ~ 2020)

https://modulabs.co.kr/blog/alexnet_structure
https://daechu.tistory.com/10
https://www.geeksforgeeks.org/deep-learning/residual-networks-resnet-deep-learning/
https://dotiromoook.tistory.com/24
https://herbwood.tistory.com/15
https://modulabs.co.kr/blog/introducing-fully-convolutional-networks
https://kyujinpy.tistory.com/9
https://www.ultralytics.com/ko/blog/what-is-mask-r-cnn-and-how-does-it-work
https://lilianweng.github.io/posts/2018-08-12-vae/
https://www.linkedin.com/pulse/what-generative-adversarial-networks-gans-sushant-babbar-qpc9c

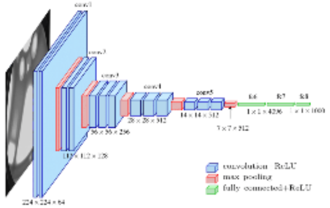
- **Paradigm shift to DNN and CNN**
  - From handcrafted features to end-to-end feature learning through deep neural networks

ImageNet breakthrough

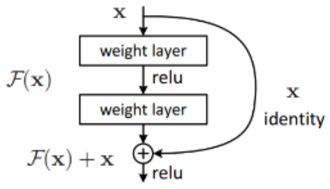
- AlexNet (Rebirth of CNN)
  - ReLu, Dropout, Max. pooling, Multi-GPUs (3GB)
    (GTX-580)



- VGGNet (3X3 filters)

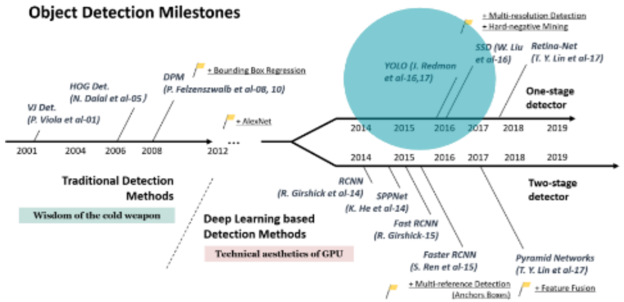


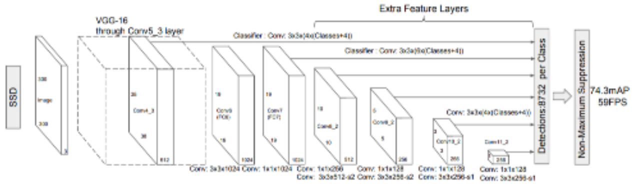
- ResNet (skip connection)



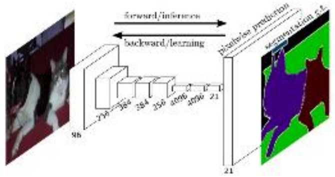
Vision applications

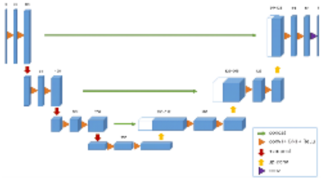
- Object detection & recognition (R-CNN, Fast R-CNN, Faster R-CNN, YOLO, SSD)

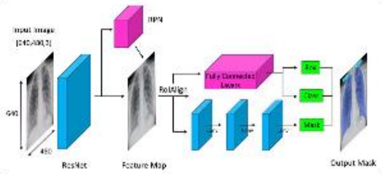




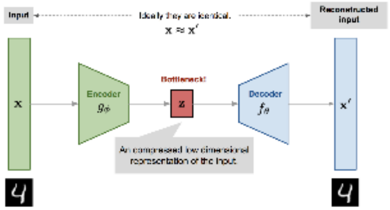
- Image segmentation & scene understanding (FCN, U-Net, Mask R-CNN, DeepLab)

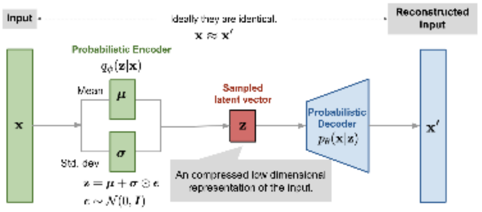


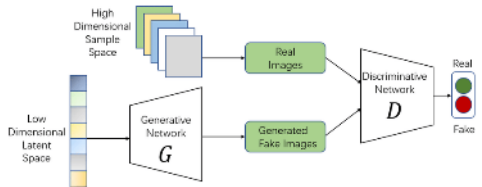




- Generative vision & representation learning (Auto Encoder, VAE, GAN)







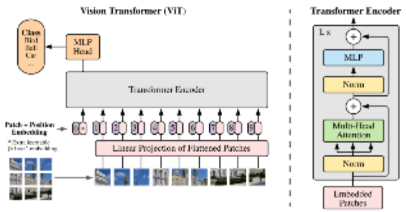
Integration with modern AI (2021 ~)

https://www.ultralytics.com/ko/blog/exploring-sam-3-meta-ais-new-segment-anything-model
https://velog.io/@rcchun/CLIP-%EB%AA%A8%EB%8D%B8-%EB%B6%84%EC%84%9D
https://sh-tsang.medium.com/review-align-scaling-up-visual-and-vision-language-representation-learning-with-noisy-text-2970ce0c4065
https://han0ahblog.tistory.com/3
https://sh-tsang.medium.com/review-dall-e-zero-shot-text-to-image-generation-f9de7a383374
https://textcortex.com/ko/post/what-is-gpt-4v-ision

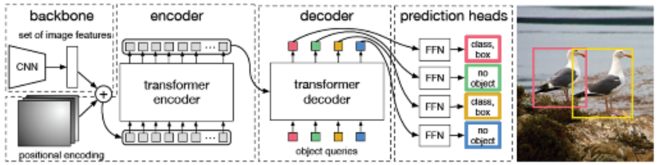
- **Multi-modal vision**
  - Understanding and generation

Transformer-based vision models

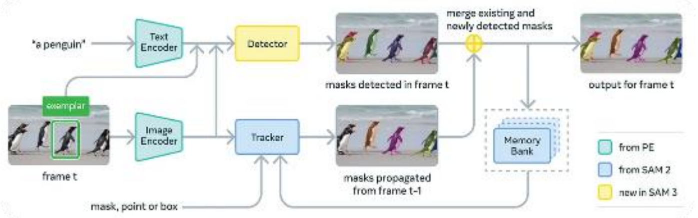
- Vision transformer (Google)



- DETR (Meta)

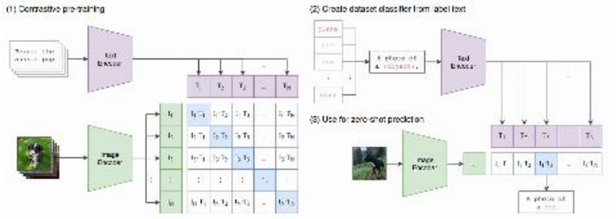


- SAM (Meta) ☞ Demo

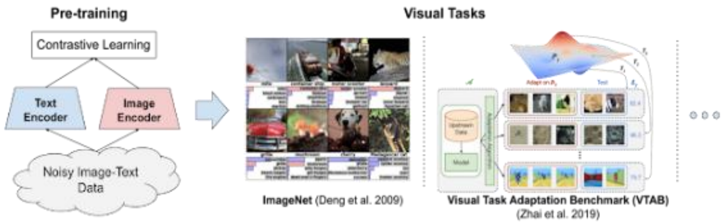


Vision-language integration

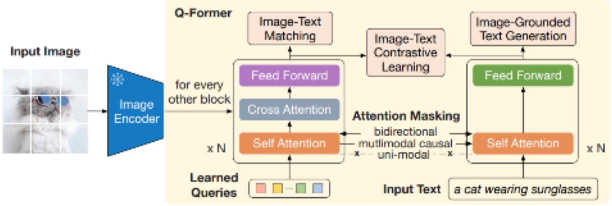
- CLIP (OpenAI)



- ALIGN (Google)



- BLIP (Salesforce) ☞ Demo

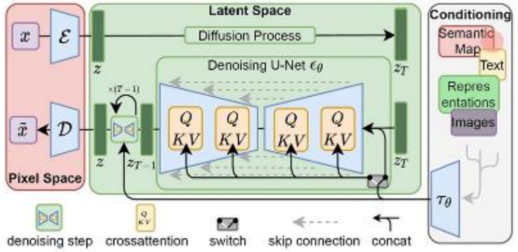


- Flamingo (DeepMind)

Generative & Interactive Vision

- DALL-E (OpenAI)

- Stable Diffusion (stability.ai) ☞ Demo



- GPT (OpenAI)

- Gemini (Google)

How vision models learn ? (1/2)

https://www.analyticsvidhya.com/blog/2021/10/human-pose-estimation-using-machine-learning-in-python/
A Simple Framework for Contrastive Learning of Visual Representations SimCLR, by Google Research, Brain Team 2020 ICML
https://medium.com/@gayatri_sharma/a-gentle-introduction-to-semi-supervised-learning-7afa5539beea
https://sanghyu.tistory.com/177

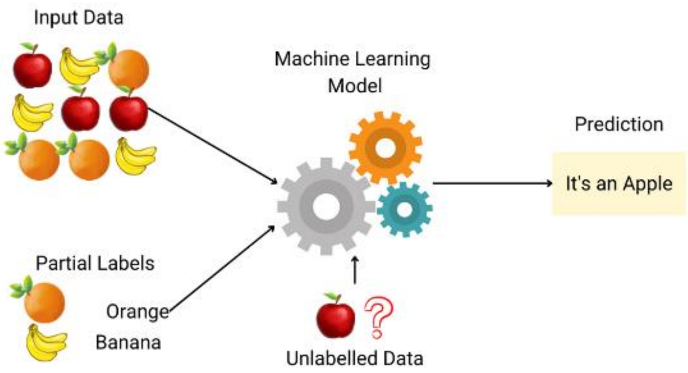
Supervised learning

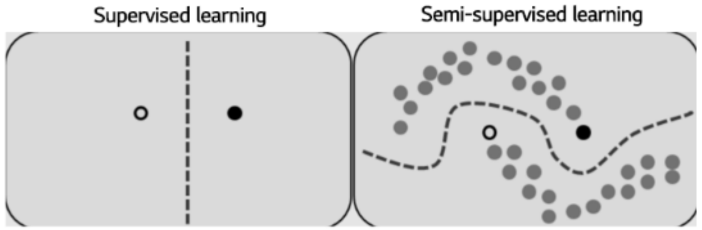
- **Classification**
  - Cat vs. Dog
  - Segmentation
    (i.e., pixel-wise classification)
- **Regression (Localization)**
  - Bounding box
    (i.e., x, y, w, h)
  - Pose estimation

Semi-supervised learning

- **Few labeled** and
  Many unlabeled dataset
- Soft **pseudo-labeling**
  - Noisy label filtering or
    replacement

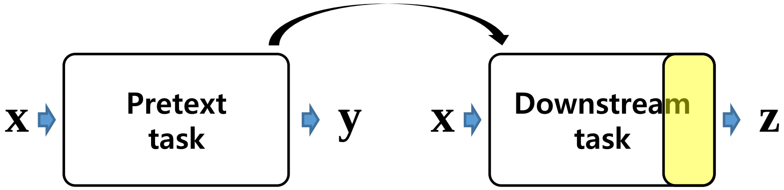
"Unsupervised Label Noise Modeling and Loss Correction", Arazo et al. (2020).

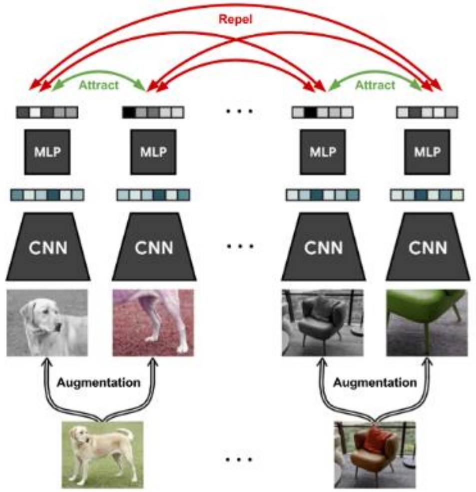




Self-supervised learning

- **Pretext task**
  - Pre-training
- **Downstream task**
  - Transfer learning
- BERT / GPT

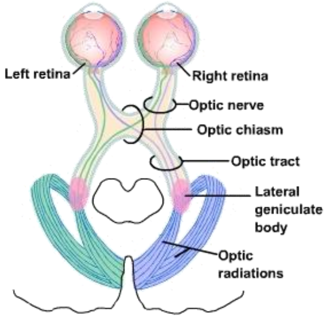
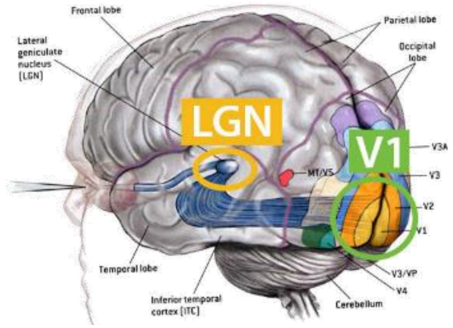
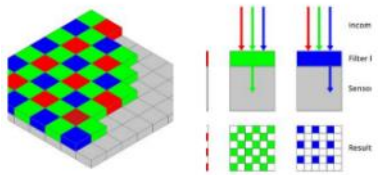
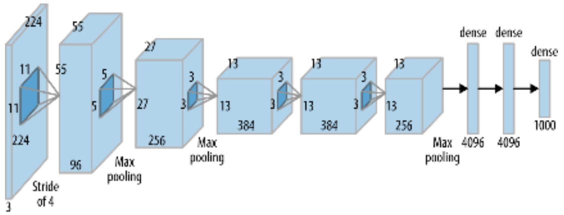




How vision models learn ? (2/2)

Category	Supervised Learning	Semi-Supervised Learning	Self-Supervised Learning
Definition	Learning with labeled data	Learning with few labeled along with large unlabeled data	Learning with no labels by generating labels from data itself
Learning Goal	Predict ground truth	Improve performance with limited labels	Learn useful data representations
Data Requirements	Large labeled dataset	Few labeled + many unlabeled dataset	Unlabeled dataset
Labeling Cost	Very high	Moderate	None
Vision Tasks	Classification Detection Segmentation	Low-label scenarios - Consistency regularization - Pseudo-labeling	Pretraining & feature learning - Pretrain on massive unlabeled image datasets - Fine-tune on specific tasks.

Human vision vs. Computer vision

Functional Stage	Human Visual System	Computer Vision System
Data Acquisition	Light captured by the retina through rods and cones.	Image captured as pixel arrays (RGB values) by camera sensors
Preprocessing & Signal Routing	LGN filters and routes visual information to cortex, organizing by color and motion	Image preprocessing (e.g., normalization, noise reduction, data augmentation)
Low-Level Feature Detection	Primary Visual Cortex (V1) detects edges, orientation, motion	Convolutional Layers in CNNs detect simple patterns (e.g., edges, textures)
Mid-Level Integration	Higher Visual Areas (V4, IT) integrate shape, color, and object identity	Deeper CNN / Transformer layers combine local features into global representations
High-Level Understanding	Prefrontal Cortex interprets visual information, linking it to memory and emotion	Fully connected layers / Vision Transformers assign semantic meaning (e.g., cat, car)
Decision & Action	Visual data informs motor cortex and decision-making (e.g., avoidance, recognition)	Visual outputs drive autonomous systems (e.g., robotics, navigation, or vision-language reasoning)
Learning & Adaptation	Learns from experience, feedback, and meaning association	Learns from large labeled datasets or reinforcement signals
Figures	 	 

https://nba.uth.tmc.edu/neuroscience/m/s2/chapter15.html
https://www.siam.org/publications/siam-news/articles/the-brain-is-a-dynamical-system/

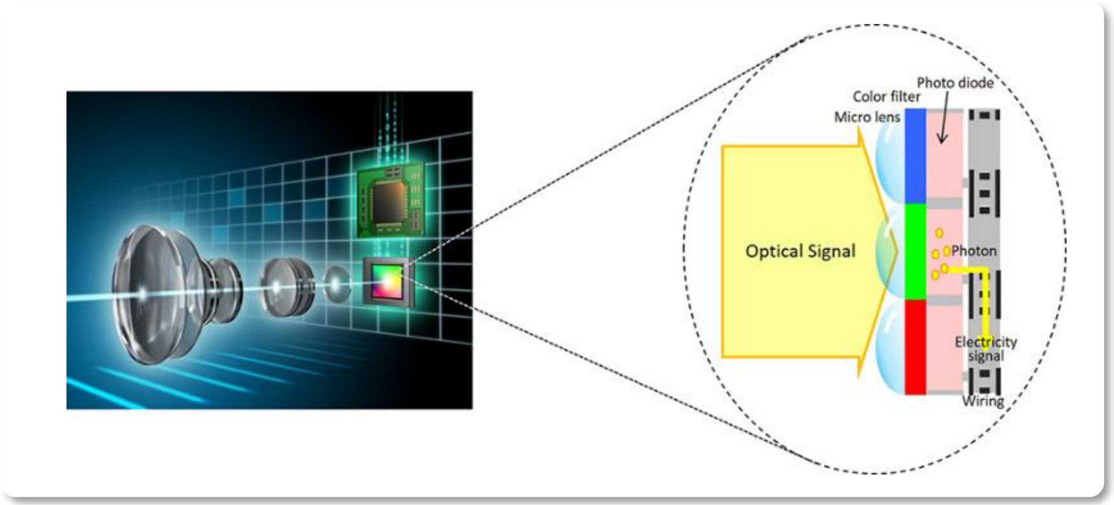
https://www.opto-e.com/en/basics/camera-basics
https://wikidocs.net/204498

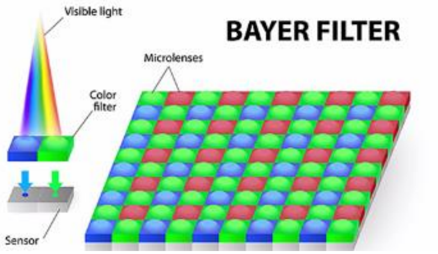
Background Knowledge

Camera image sensor

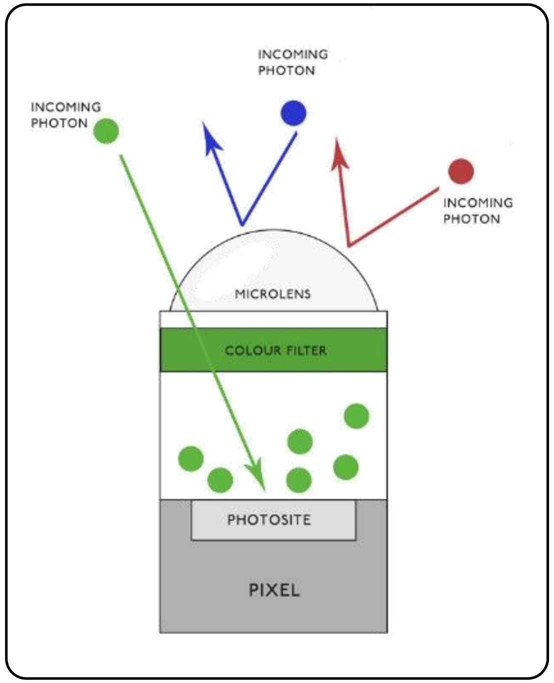
https://www.artiencegroup.com/kr/products/colorfilter/imgsensor/
https://blog.nordh.me/camera-sensor-formats/
https://www.vzense.com/blog/tech-briefs/learning-about-field-of-view-fov.html

3. **Bayer filter**
allows only certain wavelength of light





$$Y = 0.299R + 0.587G + 0.114B$$



1. **Lenses**
collect light

2. **Microlens**
increase the photon collection

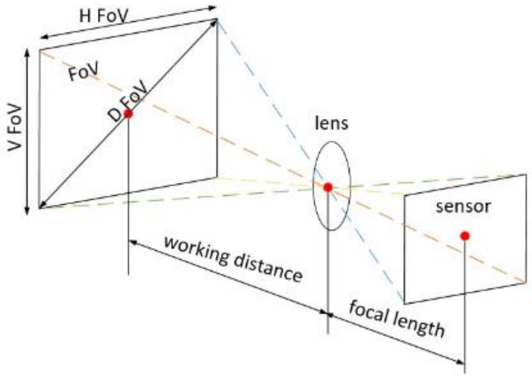
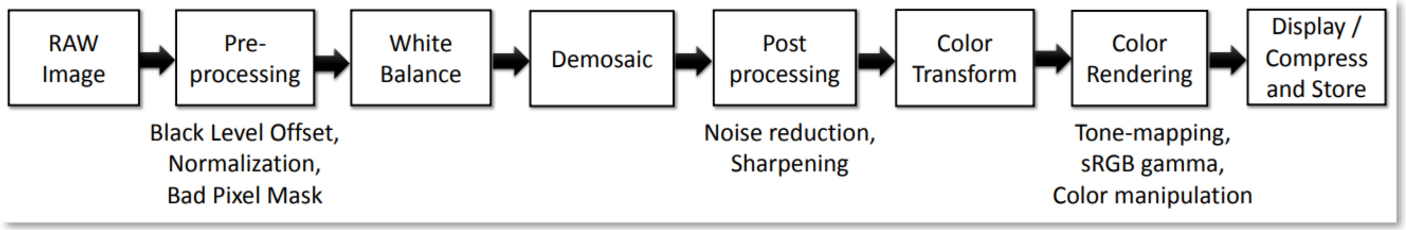
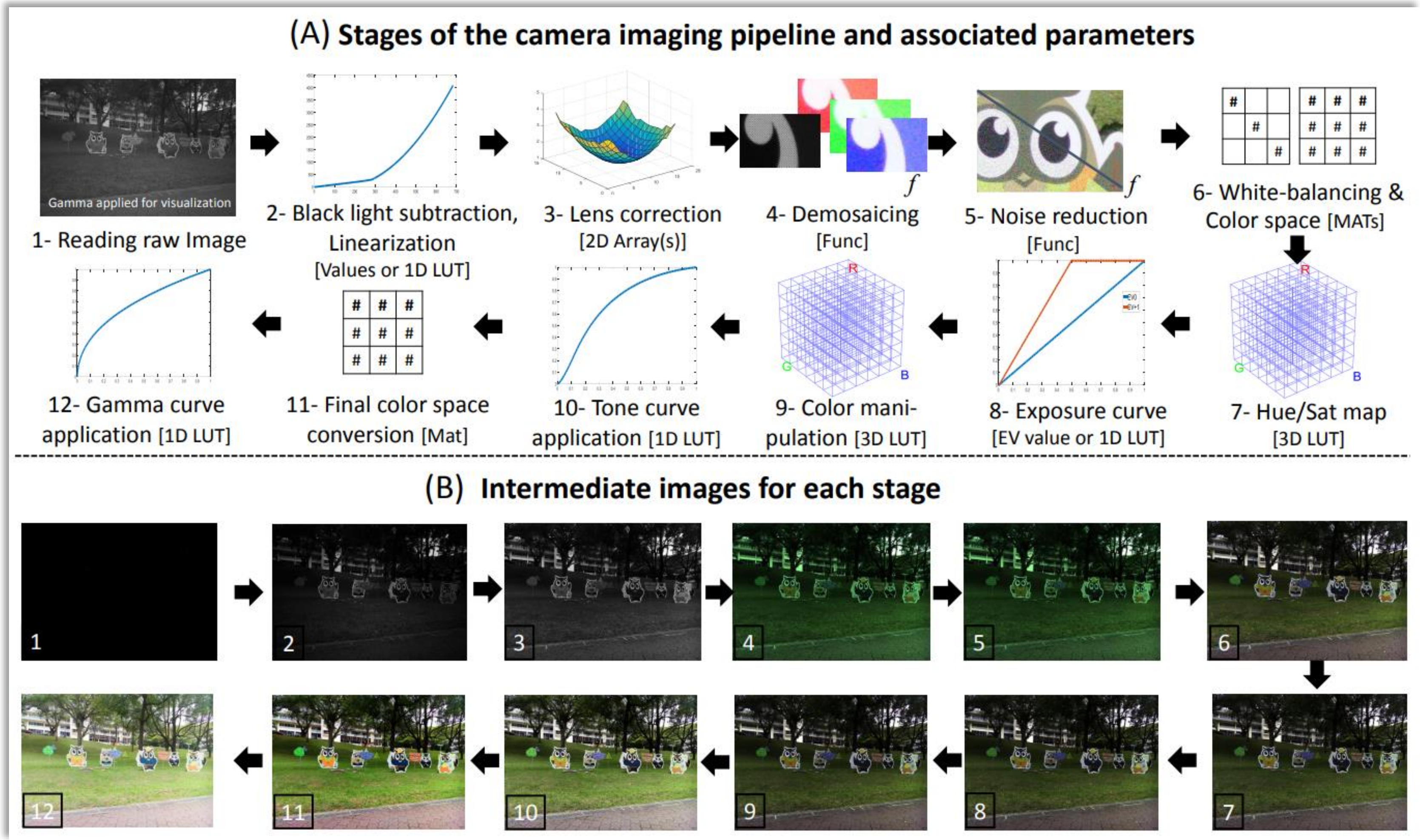


Image sensor pipeline

https://karaimer.github.io/camera-pipeline/paper/Karaimer_Brown_ECCV16.pdf



- **3A algorithm**
  - Auto focus
  - Auto exposure
  - Auto white balance



Vector and Matrix (1/3)

Vector

$$a = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \in \mathbb{R}^3$$

Matrix

$$C = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \\ c_{41} & c_{42} & c_{43} \end{bmatrix} \in \mathbb{R}^{4\times 3}$$

$$= \begin{bmatrix} c^{(1)} & c^{(2)} & c^{(3)} \end{bmatrix}$$

$$= \begin{bmatrix} r^{(1)} \\ r^{(2)} \\ r^{(3)} \\ r^{(4)} \end{bmatrix}$$

Identity matrix

$$I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{3\times 3}$$

$$I \cdot A = A \cdot I = A$$

$$I^T = I$$

$$I^{-1} = I$$

Vector and Matrix (2/3)

Matrix and vector multiplication (Column-wise)

$$Ca = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \\ c_{41} & c_{42} & c_{43} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

$$= \begin{bmatrix} c_{11}a_1 + & c_{12}a_2 + & c_{13}a_3 \\ c_{21}a_1 + & c_{22}a_2 + & c_{23}a_3 \\ c_{31}a_1 + & c_{32}a_2 + & c_{33}a_3 \\ c_{41}a_1 + & c_{42}a_2 + & c_{43}a_3 \end{bmatrix}$$

$$= a_1 c^{(1)} + a_2 c^{(2)} + a_3 c^{(3)}$$

Vector and Matrix multiplication (Row-wise)

$$a^T C^T = \begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix} \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \end{bmatrix}$$

$$= \begin{bmatrix} a_1c_{11} + a_2c_{21} + a_3c_{31} & a_1c_{12} + a_2c_{22} + a_3c_{32} & a_1c_{13} + a_2c_{23} + a_3c_{33} & a_1c_{14} + a_2c_{24} + a_3c_{34} \end{bmatrix}$$

$$= a_1 r^{(1)} + a_2 r^{(2)} + a_3 r^{(3)}$$

Transpose

$$(Ca)^T = a^T C^T$$

Vector and Matrix (3/3)

Vector norm

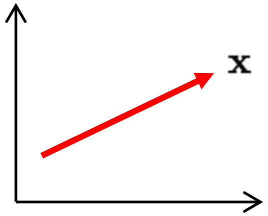
$$\mathbf{x} = [x_1, x_2, \dots, x_n]^T$$

$$\|\mathbf{x}\|_1 = \sum_{i=1}^{n} |x_i|$$

$$\|\mathbf{x}\|_2 = \left(\sum_{i=1}^{n} x_i^2\right)^{\frac{1}{2}}$$

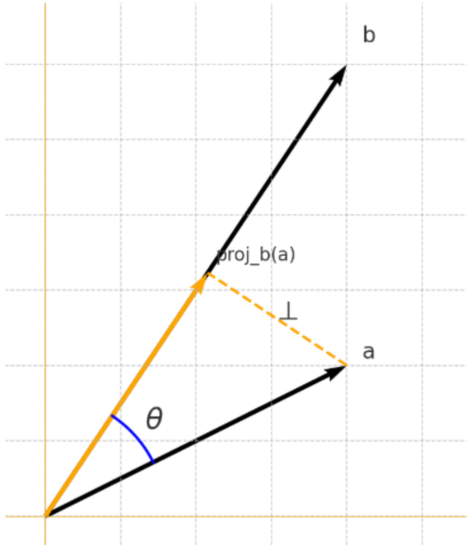
$$\|\mathbf{x}\|_\infty = \max_i |x_i|$$

$$\|\mathbf{x}\|_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}}$$



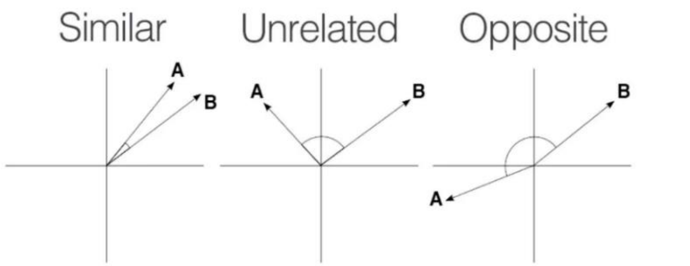
Inner product between vectors

$$a \cdot b = a^T b = a_1 b_1 + a_2 b_2 + a_3 b_3$$
$$= \|a\| \, \|b\| \cos\theta$$



Cosine similarity

$$\cos\theta = \frac{a \cdot b}{\|a\| \|b\|}$$



	Inner Product	Cosine Similarity
Aspect	Magnitude & Direction	Direction
Range	Unbounded	[-1, 1]

Probability and Statistics (1/5)

Random variable

$$X \in [0, 10]$$

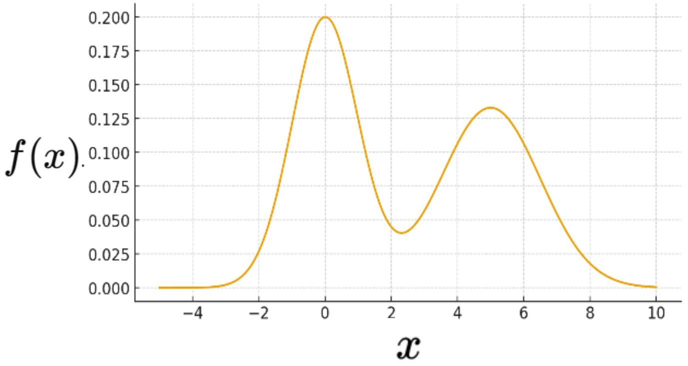
Sampling

$$x \sim X$$

PDF (Probability Density Function)

$\longrightarrow$ Continuous

$\longrightarrow$ Density (i.e., Probability over an interval)



$$\boxed{\mathbb{E}[X] = \int_{-\infty}^{\infty} x\, f(x)\, dx}$$

$$f(x) \geq 0$$

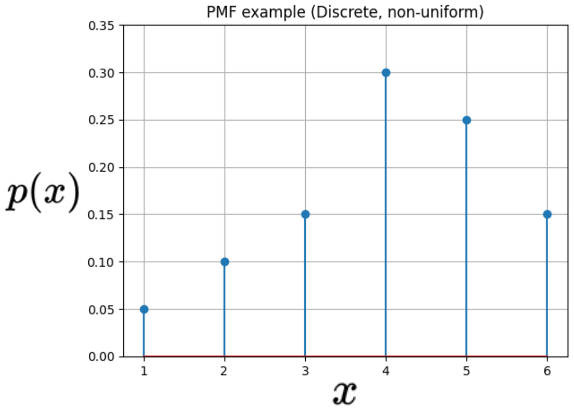
$$\begin{cases} P(a \leq X \leq b) = \int_a^b f(x)\, dx \\ P(X = x) = 0 \end{cases}$$

$$\int_{-\infty}^{\infty} f(x)\, dx = 1$$

PMF (Probability Mass Function)

$\longrightarrow$ Discrete

$\longrightarrow$ Probability



$$\boxed{\mathbb{E}[X] = \sum_x x\, p(x)}$$

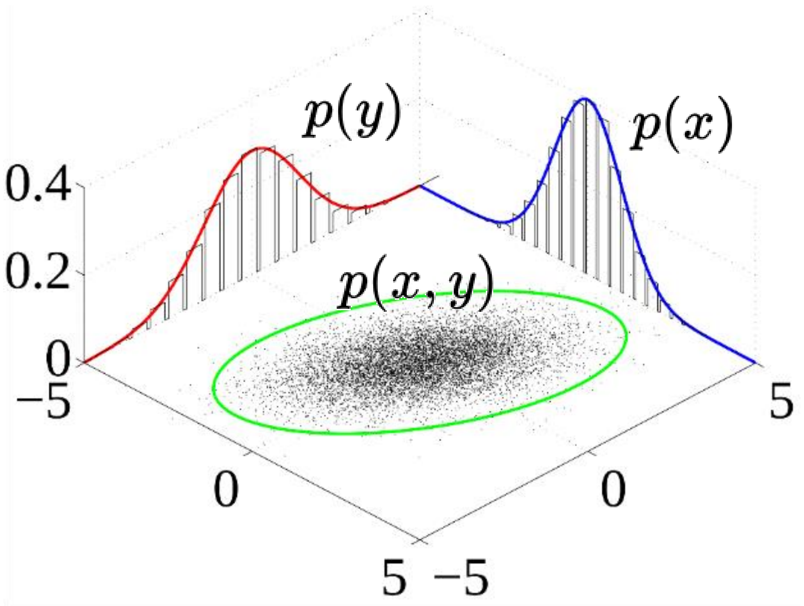
$$0 \leq p(x) \leq 1$$

$$p(x) = P(X = x)$$

$$\sum_x p(x) = 1$$

Probability and Statistics (2/5)

Joint probability distribution



Marginal distribution

$$p(x) = \int p(x,y)dy$$

$$p(y) = \int p(x,y)dx$$

Marginalization

Conditional probability

$$p(x|y) = \frac{p(x,y)}{p(y)}$$

$$p(y|x) = \frac{p(x,y)}{p(x)}$$

Independence

$$p(x,y) = p(x)\,p(y)$$

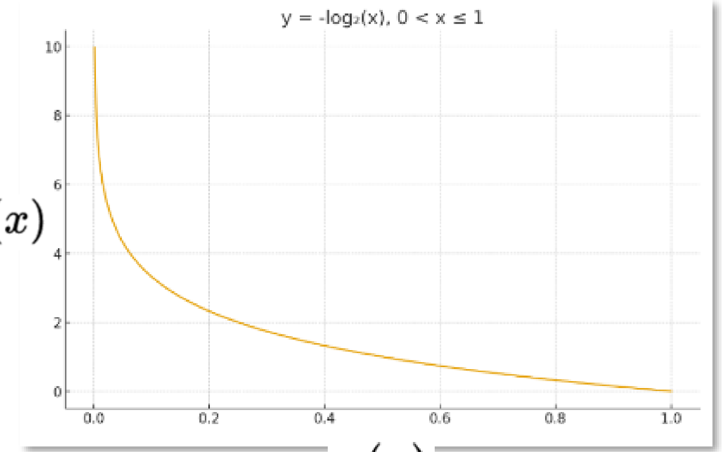
$$p(x) = p(x|y)$$

Probability and Statistics (3/5)

Entropy (→ Uncertainty)

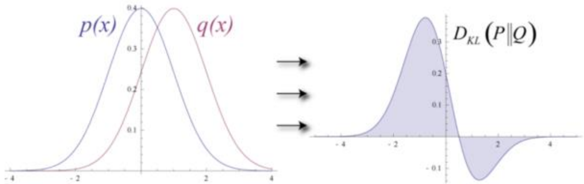
- The minimum average number of bits required to encode the outcomes of the variable
  - High probability → short bit length
  - Low probability → long bit length

$-\log_2 p(x)$



$p(x)$

$$H(p) = \sum_{x\in X} p(x)\,(-\log_2 p(x))$$
$$= -\sum_{x\in X} p(x)\log_2 p(x)$$



Cross entropy

- A measure of the difference between p and q probability distributions

$$H(p,q) = -\sum_{x\in X} p(x)\log_2 q(x)$$

KL Divergence

- A measure of how one probability distribution p differs from another reference probability distribution q.

$$D_{KL}(p||q) = H(p,q) - H(p)$$
$$= -\sum_{x\in X} p(x)\log_2 q(x) + \sum_{x\in X} p(x)\log_2 p(x)$$
$$= \sum_{x\in X} p(x)\log_2 \frac{p(x)}{q(x)} \geq 0$$

Probability and Statistics (4/5)

Bayes' rule

$$\overset{\textit{Posterior}}{p(\theta|x)} = \frac{\overset{\textit{Likelihood}}{p(x|\theta)}\ \overset{\textit{Prior}}{p(\theta)}}{\underset{\textit{Evidence}}{p(x)}}$$

Markov process

$$P(X_{t+1} \mid X_t, X_{t-1}, \dots, X_0) = P(X_{t+1} \mid X_t)$$

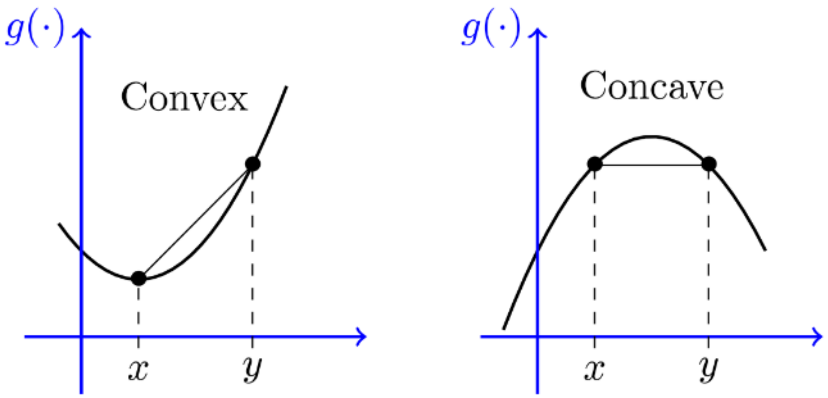
$$P(\text{Future} \mid \text{Present}, \text{Past}) = P(\text{Future} \mid \text{Present})$$

MAP (Maximum A Posteriori Estimation)

$$\hat{\theta}_{\text{MAP}} = \arg\max_{\theta} p(\theta \mid x)$$

$$\hat{\theta}_{\text{MAP}} = \arg\max_{\theta} \left[\log p(x \mid \theta) + \log p(\theta)\right]$$

Convex vs. Concave



MLE (Maximum Likelihood Estimation)

$$\hat{\theta}_{\text{MLE}} = \arg\max_{\theta} p(x \mid \theta)$$

https://www.probabilitycourse.com/chapter6/6_2_5_jensen%27s_inequality.php

Probability and Statistics (5/5)

ELBO (Evidence Lower Bound)

$$p(z|x) = \frac{p(x|z)p(z)}{p(x)}$$

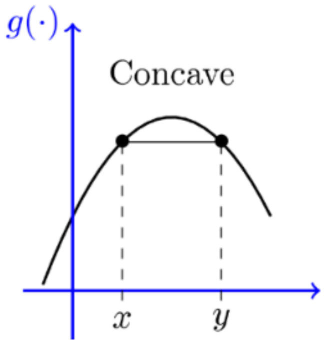
Marginalization → $p(x) = \int p(x,z)\,dz = \int p(x|z)p(z)\,dz$

$$\log p(x) = \log \int p(x,z)\,dz = \log \int q(z|x)\frac{p(x,z)}{q(z|x)}\,dz$$

Expectation → $= \log \mathbb{E}_{q(z|x)}\left[\frac{p(x,z)}{q(z|x)}\right]$

Jensen's inequality → $\geq \mathbb{E}_{q(z|x)}\left[\log \frac{p(x,z)}{q(z|x)}\right]$

ELBO → $\mathcal{L}(q) \;:=\; \mathbb{E}_{q(z|x)}\left[\log \frac{p(x,z)}{q(z|x)}\right]$



$$= \mathbb{E}_{q(z|x)}[\log p(x|z) + \log p(z) - \log q(z|x)]$$

$$= \underbrace{\mathbb{E}_{q(z|x)}[\log p(x|z)]}_{\text{reconstruction term}} \uparrow - \underbrace{\mathbb{E}_{q(z|x)}\left[\log \frac{q(z|x)}{p(z)}\right]}_{D_{\mathrm{KL}}(q(z|x)\|p(z))} \downarrow$$

Image dataset

- Core Image Classification Datasets (CNN / ViT)

Dataset	Task	#Classes	#Images	Image Size	Typical Usage	Official Link
MNIST	Classification	10	70K	28×28	Toy benchmark, sanity check	http://yann.lecun.com/exdb/mnist/
Fashion-MNIST	Classification	10	70K	28×28	MNIST replacement	https://github.com/zalandoresearch/fashion-mnist
CIFAR-10	Classification	10	60K	32×32	CNN & ViT baseline	https://www.cs.toronto.edu/~kriz/cifar.html
CIFAR-100	Classification	100	60K	32×32	Fine-grained classification	https://www.cs.toronto.edu/~kriz/cifar.html
SVHN	Classification	10	600K+	32×32	Domain shift test	http://ufldl.stanford.edu/housenumbers/
STL-10	Classification / SSL	10	113K	96×96	Low-label SSL benchmark	https://cs.stanford.edu/~acoates/stl10/
Tiny ImageNet	Classification	200	100K	64×64	Lightweight ImageNet proxy	https://www.kaggle.com/c/tiny-imagenet
ImageNet-1K	Classification	1,000	1.28M	~224×224	Standard vision benchmark	https://www.image-net.org/
ImageNet-21K	Classification	21K	14M	~224×224	Large-scale pretraining	https://www.image-net.org/
Places365	Scene Classification	365	1.8M	~224×224	Scene understanding	http://places2.csail.mit.edu/
iNaturalist	Fine-grained Cls	5K+	3M+	Variable	Long-tail evaluation	https://www.inaturalist.org/

Image dataset

- Large-Scale & Pretraining Datasets (ViT-focused)

Dataset	Purpose	Scale	Notes	Official Link
JFT-300M	Pretraining	300M images	Internal Google dataset (ViT)	Not public
OpenImages	Classification / Detection	9M+ images	Large-scale, noisy labels	https://storage.googleapis.com/openimages/web/index.html
YFCC100M	SSL / VLM	100M images	Flickr-based, weak labels	https://multimediacommons.wordpress.com/yfcc100m-core-dataset/
LAION-400M	Vision-Language	400M pairs	CLIP-style training	https://laion.ai/blog/laion-400-open-dataset/
LAION-5B	Vision-Language	5B pairs	Foundation model scale	https://laion.ai/blog/laion-5b/

Image dataset

- Detection / Segmentation Benchmarks

Dataset	Task	#Classes	#Images	Typical Usage	Official Link
PASCAL VOC	Detection / Segmentation	20	~11K	Classical benchmark	http://host.robots.ox.ac.uk/pascal/VOC/
MS COCO	Detection / Seg / Keypoints	80	330K	Standard detection benchmark	https://cocodataset.org/
Cityscapes	Segmentation	19	25K	Autonomous driving	https://www.cityscapes-dataset.com/
ADE20K	Segmentation	150	25K	Complex scenes	https://groups.csail.mit.edu/vision/datasets/ADE20K/
LVIS	Detection	1,200+	164K	Long-tail detection	https://www.lvisdataset.org/