

Introduction

Computer Vision and Pattern Recognition

Byeongjoon Noh

powernoh@sch.ac.kr



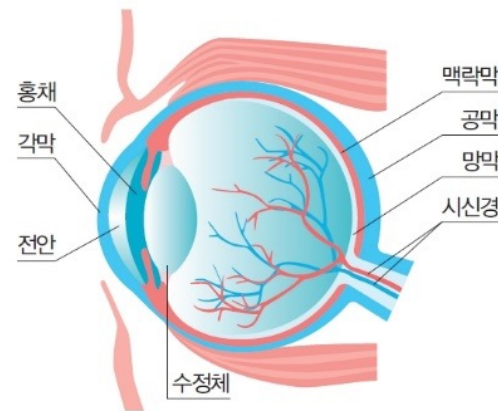
Contents

1. What is computer vision?
2. Python and OpenCV settings

1. What is computer vision?

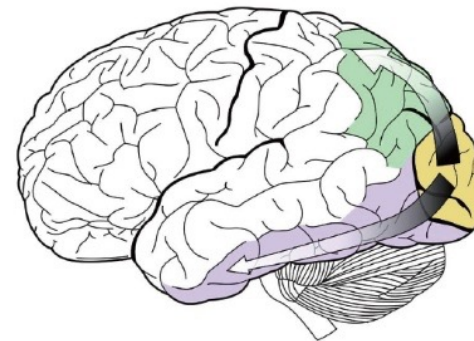
Human vision

- 시각은 인간의 오감 중 가장 뛰어난 감각
- 인간의 눈 구조와 동작
 - 등쪽 경로(녹색부분)는 주로 물체의 움직임 탐지를 담당
 - 배쪽 경로(보라색부분)은 물체의 부류를 담당
 - 매 순간 빠르고 정확하게 그리고 아주 손쉽게 인식



(a) 눈의 구조

그림 1-2 인간의 시각 시스템



(b) 시각 정보 처리를 담당하는 시각 피질

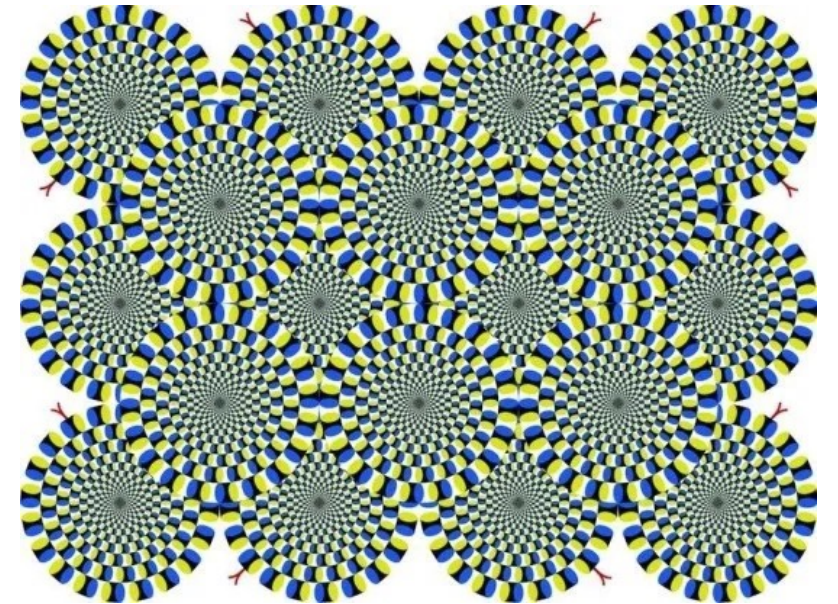
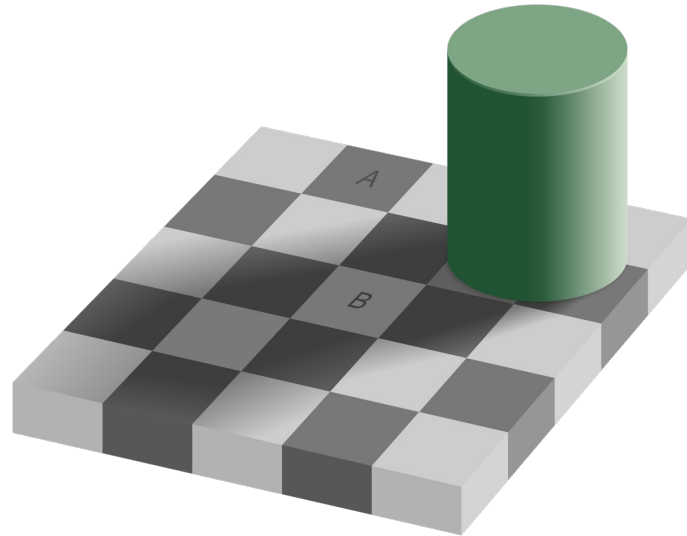
Human vision

- 인간 시각의 강점
 - 분류, 검출, 분할, 추적, 행동 분석에 능숙
 - 3차원 복원 능력
 - 빠르고 강건하며, 사전 행동에 능숙
 - 다른 지능 요소 (지식표현, 추론, 계획 등)과 협력
 - 과업 전환이 매끄럽고 유기적으로 빠름



Human vision

- 인간 시각의 한계
 - 착시 (optical illusion)가 있음
 - 정밀 측정에 오차
 - 시야가 한정됨
 - 피로해지고 퇴화됨



Computer vision

- 컴퓨터 비전은 인간의 시각을 흉내 내는 컴퓨터 프로그램
 - 물체의 단순 탐지 및 인식의 경우 이미 인간의 능력 초월
 - 그러나, 복합상황 인지, 추론, 계획, 지식 표현 부분에서는 명확한 한계

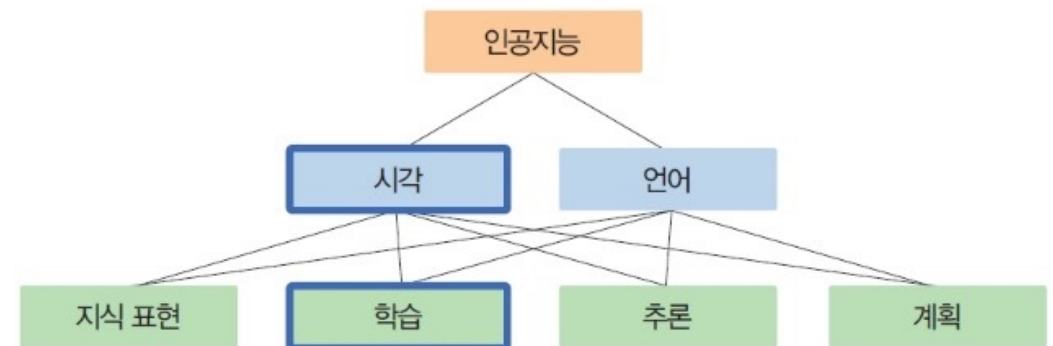
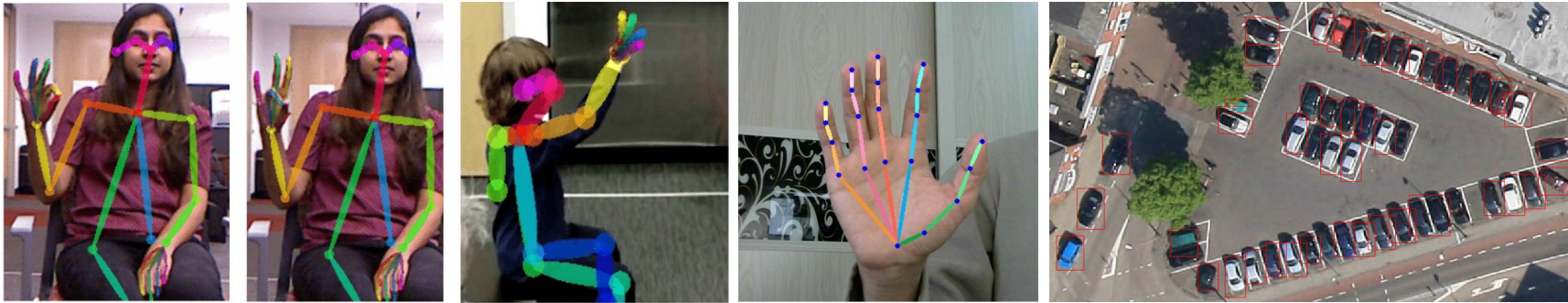


그림 1-4 인공지능의 실현

What makes computer vision challenging?

- 환경 (낮/밤, 날씨 등) 변화
- 보는 위치와 방향의 변화
- 강체와 연성 물체
- 원자~우주 까지 매우 긴 스펙트럼을 다룸

- **넘버 크런처 (number cruncher)**

- 컴퓨터로 수치해석을 하는 사람들
- 대량의 데이터를 처리하거나 빠른 속도로 정밀한 계산을 하는데 사용하는 컴퓨터

- 인공지능의 미숙함

- 지식 표현, 추론, 계획, 학습 등이 유기적으로 동작할 때만 강한 인공지능이 가능
- 강한 인공지능은 영영 불가능?

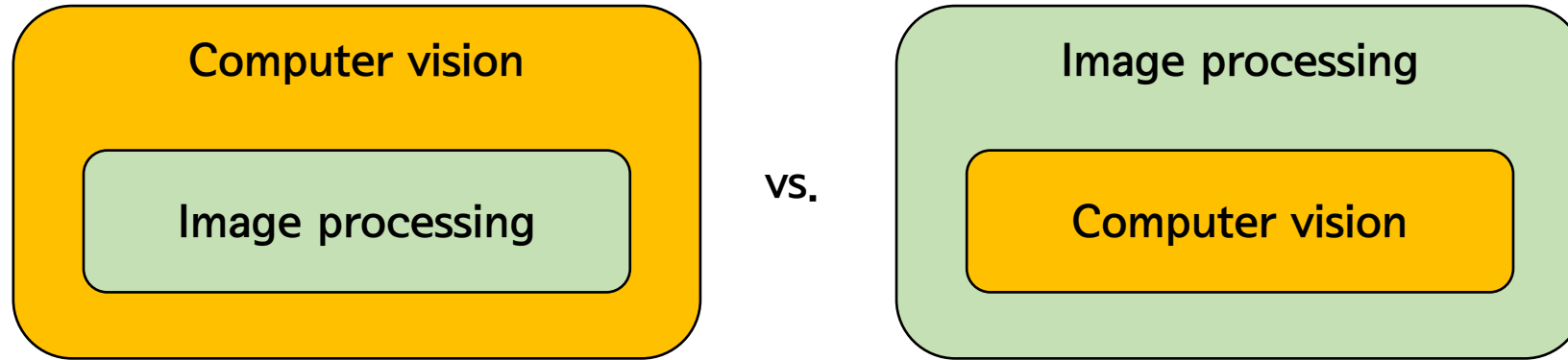
- ...

125	134	125	122	127	127	120	130	139	135	139	140	133	127	127	130	133	135	138	133	137	139	134	130	125	121
117	123	114	116	120	122	118	120	122	117	122	126	124	117	106	100	99	102	105	120	118	113	109	105	106	111
109	110	105	102	112	123	130	135	147	171	191	184	183	174	157	139	124	107	90	92	87	88	92	93	88	89
108	105	100	116	117	129	163	195	210	217	205	215	211	198	185	176	167	143	117	91	80	77	88	91	84	79
107	109	102	120	146	173	200	193	172	165	138	141	135	123	118	125	139	143	137	121	99	84	85	88	82	81
104	107	115	134	159	171	170	136	115	129	107	83	83	82	80	83	90	103	113	125	108	93	91	90	86	83
107	120	137	160	150	125	139	150	167	174	115	99	94	93	98	98	89	87	91	104	103	99	97	95	94	95
111	133	155	134	151	157	189	206	216	212	136	114	92	83	97	110	108	100	98	97	101	101	95	92	103	120
130	145	164	165	185	213	219	210	212	195	158	108	123	137	137	123	111	121	134	145	132	130	147	159	163	171
138	151	170	185	195	215	222	211	214	218	209	160	152	151	157	163	166	167	166	159	155	150	180	193	195	193
142	153	171	190	190	204	218	213	207	214	218	213	204	195	192	189	183	178	173	161	159	163	171	183	189	187
141	151	164	188	178	180	197	204	201	197	196	196	193	190	187	176	163	157	156	156	161	163	166	174	186	192
144	151	160	185	183	176	176	187	192	191	188	193	184	178	177	174	165	156	151	148	163	177	182	188	200	203
152	160	168	176	193	193	182	180	180	174	172	164	161	159	154	146	140	143	149	173	184	190	190	193	199	205
159	188	178	178	202	206	197	194	187	175	175	167	172	179	180	176	176	188	203	215	212	206	204	202	204	205
161	171	185	197	210	204	199	211	210	206	212	219	210	206	215	225	226	220	215	214	209	210	214	216	211	200

그림 1-6 컴퓨터 비전이 인식해야 하는 영상은 아주 큰 숫자 배열



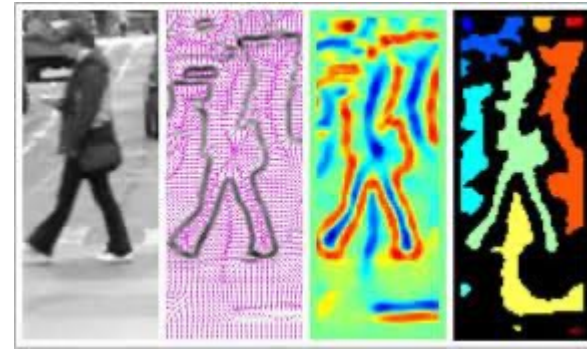
Note: Computer vision vs. Image processing



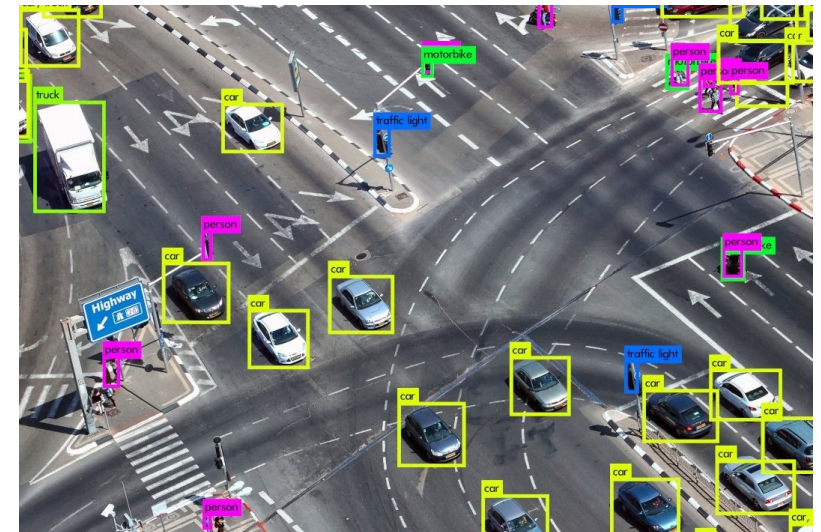
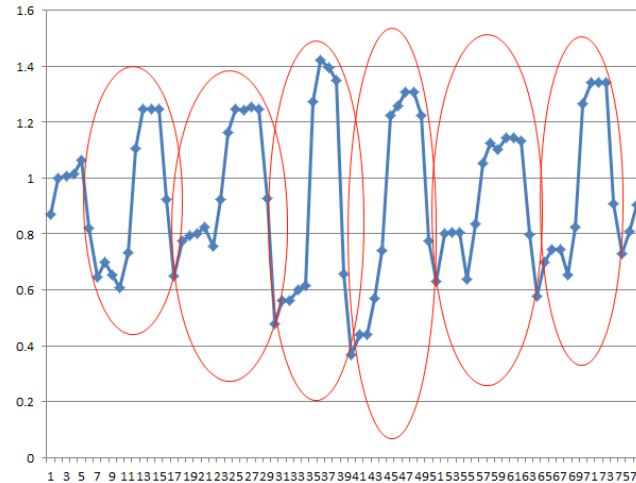
- **Image processing:** 주로 이미지를 개선하거나 변형하는데 집중
 - 이미지의 가시성 개선, 데이터 추출을 위해 이미지에 다양한 연산 적용 등
 - 이미지에서 정보를 추출하고 분석하는 다양한 기술을 포함
- **Computer vision:** 이미지나 비디오에서 복잡한 장면을 해석하고 이해하는 과정에 집중
 - 인간의 시각 시스템을 모방하여 실세계의 객체를 인식, 분류, 추적, 생성 등의 고급 작업을 목표로 함
 - 이미지의 복잡한 해석과 이해에 집중

Pattern recognition in computer vision

- Pattern recognition
 - a data analysis method that uses “something” algorithms to automatically recognize patterns and regularities in data
 - anything data sources: text, image, sound, etc.

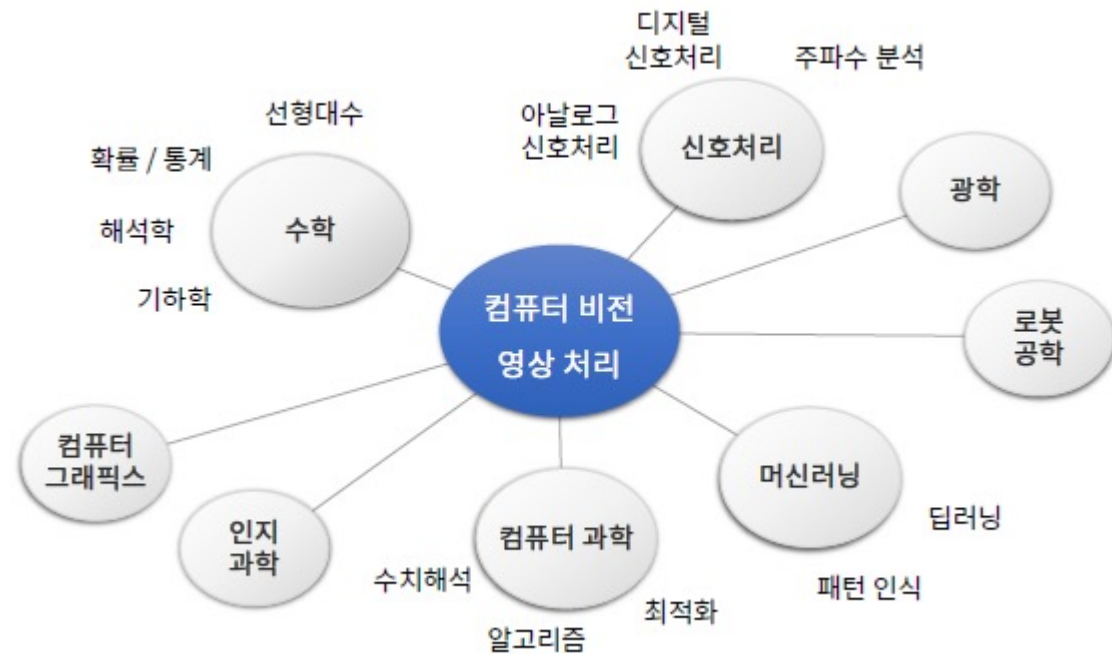
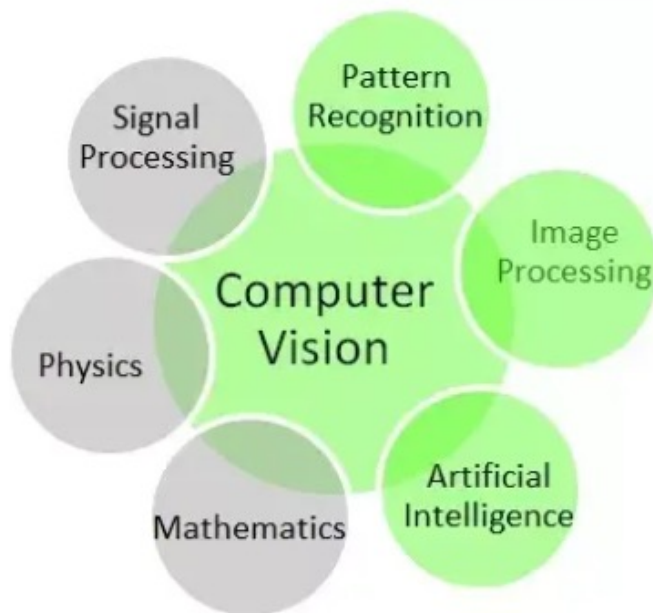


1+4 = 5
2+5 = 12
3+6 = 21
8+11 = ?



Note: Computer vision vs. Image processing

- Image processing
 - smoothing, sharpening, compression, calibration, watermarking, edge detection, etc.
- Computer vision
 - detecting, positioning, identification, action, measurement, projection, etc.



History

- 신문 산업에서 태동한 디지털 영상
 - 1920년 유럽과 북미 간 케이블을 통해 사진을 전송하는 Bartlane 시스템 개통
- 1946년 에니악 탄생
 - 에니악: 세계 최초 범용 전자식 컴퓨터 (초당 3000개의 덧셈 가능)
- 1957년 스캐너를 통해 디지털 영상을 컴퓨터에 저장
 - 5cm*5cm 사진에서 획득한 176*176 디지털 영상 (← 컴퓨터 비전의 태동)



Bartlane 시스템(유럽-북미 해저 케이블로 사진을 전송하는 시스템)으로 전송된 세계 최초의 디지털 영상(1920년)

→ 37년



세계 최초로 컴퓨터에 저장된 디지털 영상 (1957년)

→ 65년



자율주행차(현재)

그림 1-7 컴퓨터 비전의 발전

History

표 1-1 컴퓨터 비전의 역사

연도	사건
1920	• Bartlane 영상 전송 케이블 시스템 구축 [McFarlane1972]
1946	• 세계 최초 전자식 범용 디지털 컴퓨터인 에니악 탄생
1957	• 커쉬가 세계 최초로 디지털 영상을 컴퓨터에 저장
1958	• 로젠블라트의 퍼셉트론 제안(이후 Mark 1 Perceptron에서 문자 인식 실험)
1968	• 소벨의 소벨 에지 연산자 제안
1979	• IEEE Transactions on Pattern Analysis and Machine Intelligence 창간 • ACRONYM 시스템 발표 [Brooks1979]
1980	• 후쿠시마의 네오코그니트론 논문 발표 [Fukushima1980]
1983	• 제1회 CVPR(Computer Vision and Pattern Recognition)이 미국 알링턴에서 개최
1986	• 캐니의 캐니 에지 연산자 논문 발표 [Canny1986] • 루멜하트의 『Parallel Distributed Processing』 출간(다층 퍼셉트론 제안) [Rumelhart1986]

History

1987	<ul style="list-style-type: none">• International Journal of Computer Vision 창간• 런던에서 제1회 ICCV(International Conference on Computer Vision) 개최(홀수 연도)• Marr상 제정(ICCV에서 시상)• 덴버에서 제1회 NIPS(Neural Information Processing Systems) 개최(2018년에 NeurIPS로 개명)
1990	<ul style="list-style-type: none">• 프랑스 안티베에서 제1회 ECCV(European Conference on Computer Vision) 개최(짝수 연도)
1991	<ul style="list-style-type: none">• Eigenface 얼굴 인식 논문 발표 [Turk1991]
1998	<ul style="list-style-type: none">• 르쿤의 컨볼루션 신경망 논문 발표 [LeCun1998]
1999	<ul style="list-style-type: none">• 로우의 SIFT 논문 발표 [Lowe1999]• 엔비디아에서 GPU 발표
2000	<ul style="list-style-type: none">• CVPR에서 OpenCV 알파 버전 공개
2001	<ul style="list-style-type: none">• Viola-Jones 물체 검출 논문 발표 [Viola2001]
2004	<ul style="list-style-type: none">• 그랜드 챌린지(고속도로 자율주행)
2005	<ul style="list-style-type: none">• PASCAL VOC 대회 시작

History

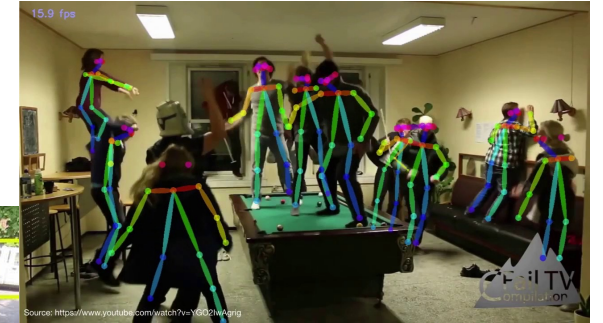
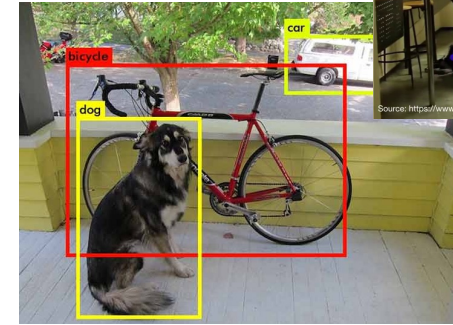
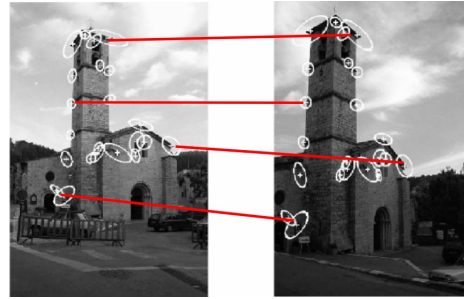
2006	<ul style="list-style-type: none">• OpenCV 1.0 공개
2007	<ul style="list-style-type: none">• 어번 챌린지(도심 자율주행)• Azriel Rosenfeld Lifetime Achievement상 제정
2009	<ul style="list-style-type: none">• 페이페이 리가 CVPR에서 ImageNet 데이터셋 발표• OpenCV 2.0 공개
2010	<ul style="list-style-type: none">• Xbox 360을 위한 Kinect 카메라 시판• 제1회 ILSVRC 대회 개최• MS COCO 데이터셋 발표
2012	<ul style="list-style-type: none">• ILSVRC 대회에서 AlexNet 우승 [Krizhevsky2012]• 시각 장애인을 태운 자율주행차의 시범 운행 성공
2013	<ul style="list-style-type: none">• 아타리 비디오 게임에서 사람 성능 추월 [Mnih2013]• 스코츠데일에서 제1회 ICLR(International Conference on Learning Representations) 개최
2014	<ul style="list-style-type: none">• RCNN 논문 발표 [Girshick2014]• 생성 모델인 GAN 발표 [Goodfellow2014]• ILSVRC에서 GoogLeNet이 우승, VGGNet이 준우승
2015	<ul style="list-style-type: none">• 텐서플로 서비스 시작• ILSVRC에서 ResNet이 우승

History

2016	<ul style="list-style-type: none">• 파이토치 서비스 시작• YOLO 논문 발표 [Redmon2016]
2017	<ul style="list-style-type: none">• 트랜스포머 논문 발표 [Vaswani2017]• Open Images 데이터셋 공개• 구글 렌즈 서비스 시작
2018	<ul style="list-style-type: none">• 인공지능이 그린 에드몽 벨라미가 경매에서 5억 원에 낙찰• 벤지오, 힌튼, 르쿤 교수가 딥러닝으로 튜링상 수상
2019	<ul style="list-style-type: none">• 알파스타가 스타크래프트에서 그랜드마스터 수준 달성• 트랜스포머를 위한 파이썬 라이브러리 transformers 2.0 공개
2020	<ul style="list-style-type: none">• OpenAI 재단의 GPT-3 발표• iPad Pro에 라이다 센서 장착
2021	<ul style="list-style-type: none">• 비전 트랜스포머 발표 [Dosovitskiy2021]• OpenAI 재단의 DALL·E 발표 [Ramesh2021]
2022	<ul style="list-style-type: none">• 구글의 Imagen 발표 [Saharia2022]

History

MIT
"The summer vision project"



1960

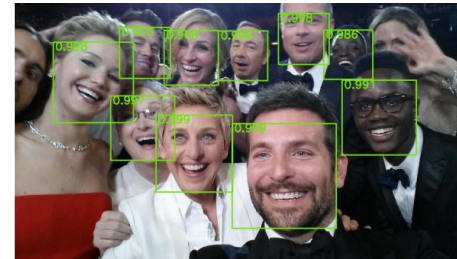
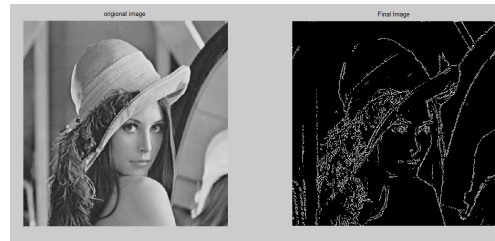
1970

2000

2010

2023

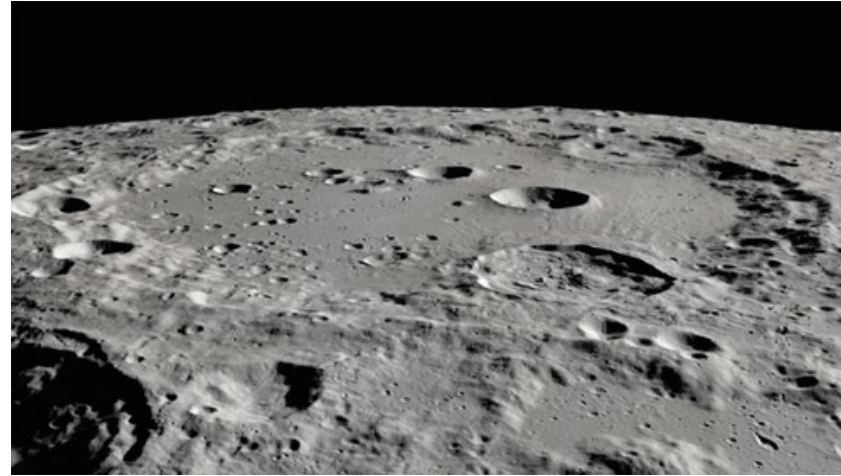
아날로그 방식의
사진 편집



달 표면 사진 화질 복원

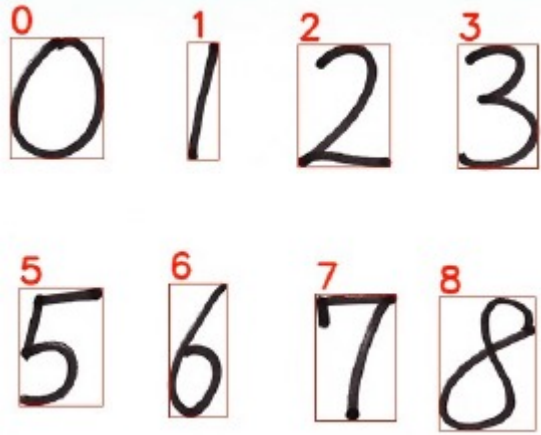
Fields

- 영상 화질 개선

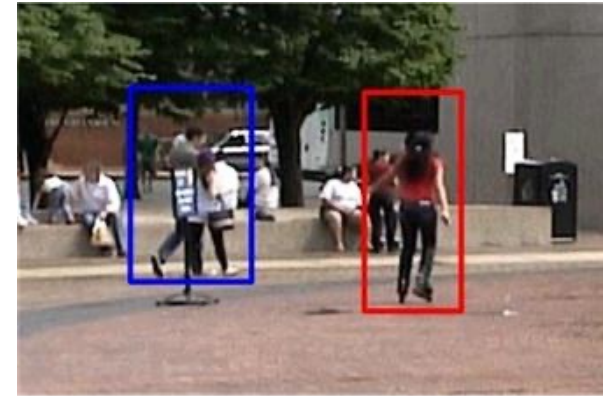


Fields

- 인식 (recognition)과 분류 (classification)



CAT



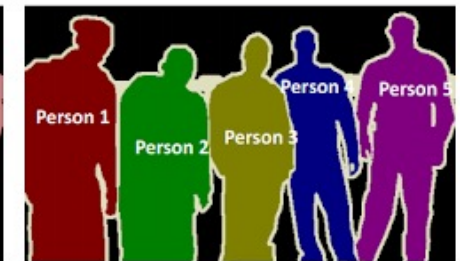
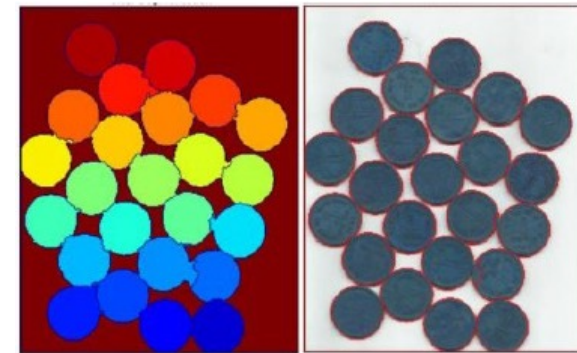
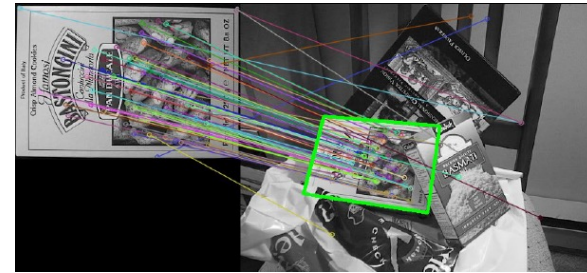
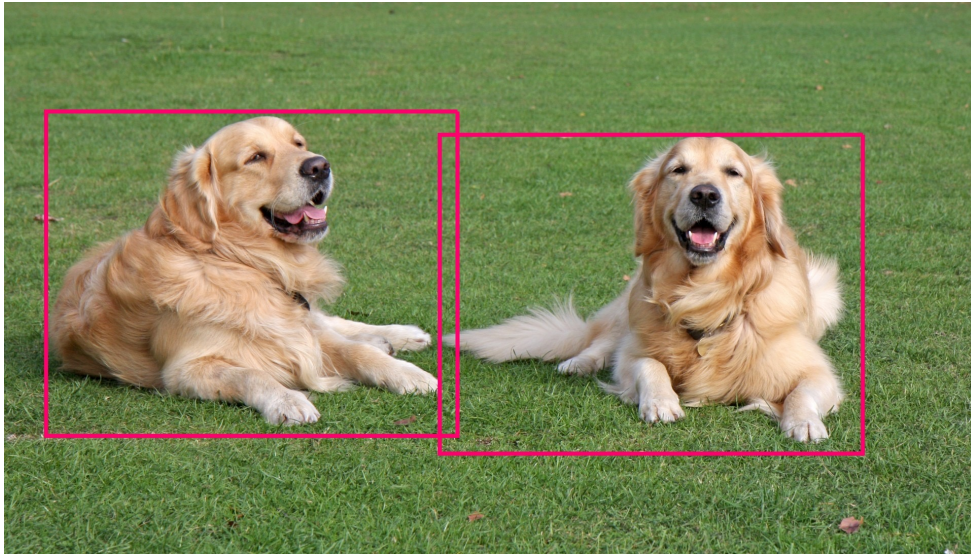
walk

skate



Fields

- 객체 검출 (object detection)과 영상 분할 (segmentation)



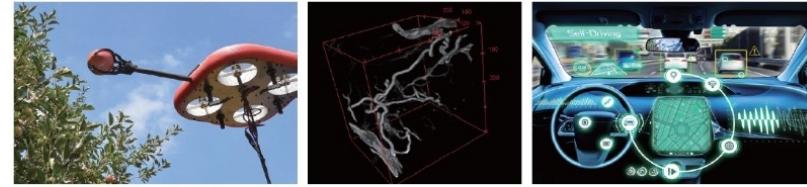
Semantic Segmentation

Instance Segmentation

Scope of computer vision

- 컴퓨터 비전을 활용한 대표적 응용 사례

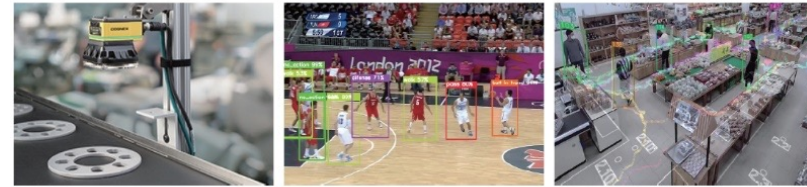
- 농업
- 의료
- 교통
- 스마트 공장
- 스포츠
- 유통
- 보안
- 에너지
- 환경
- 예술
- ...



(a) 과일 수확 드론

(b) 혈관 분할

(c) 자율주행



(d) 불량 검사

(e) 선수의 행동 분석

(f) 고객의 동선 분석



(g) 얼굴 인식 보안

(h) 태양광 모니터링

(i) 게임 플레이(알파스타)



(j) 지형 모니터링

(k) 화성 탐사선

(l) 광장 감시



(m) 에드몽 벨라미

(n) 청소 로봇

(o) 휴머노이드 로봇

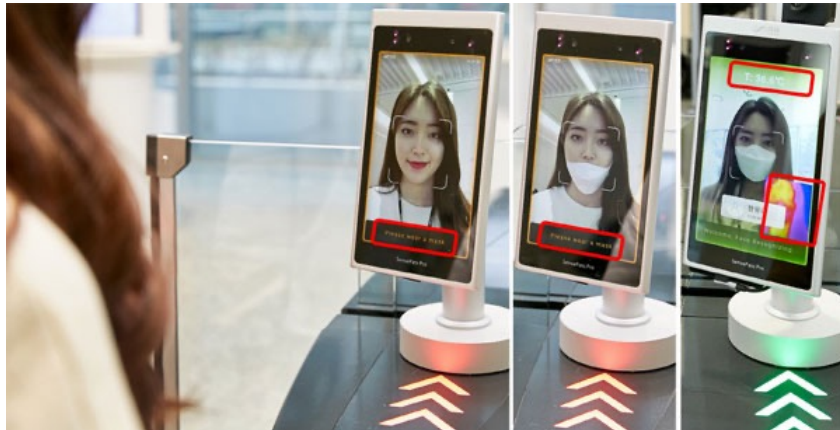
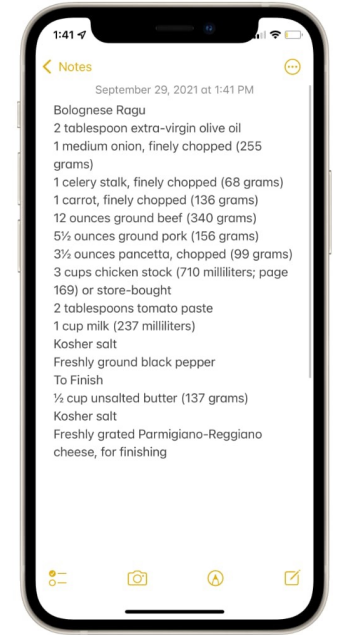
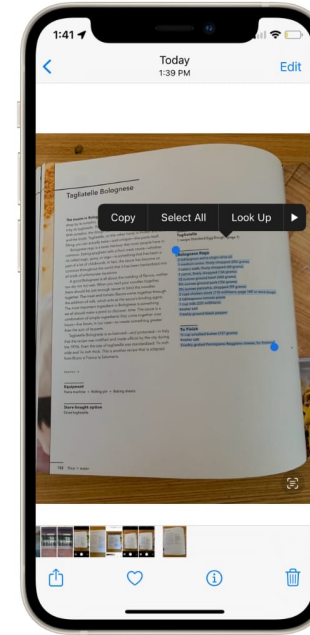
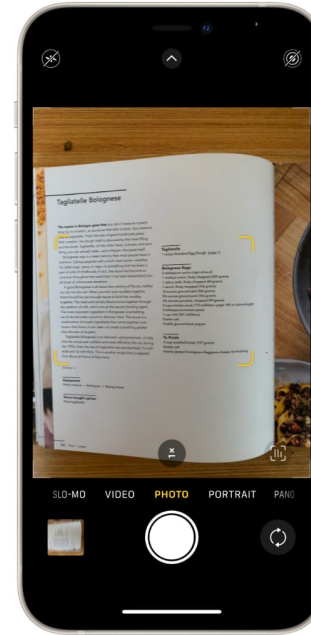
그림 1-5 컴퓨터 비전의 응용 사례

Scope of computer vision

- AI services



ChatGPT



Community

- 컴퓨터 비전 커뮤니티의 공개 문화

- SOTA를 달성한 연구자는 논문 발표와 더불어 github/huggingface 등에 소스코드와 데이터를 공개하는 문화

- 이를 활용한 웹/앱 서비스 활성화

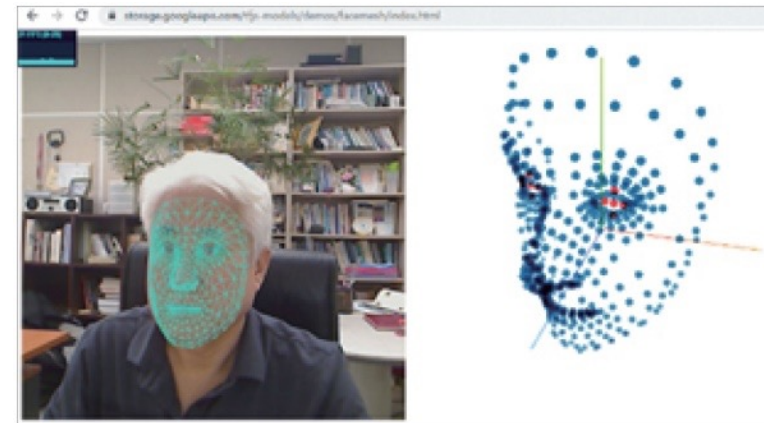
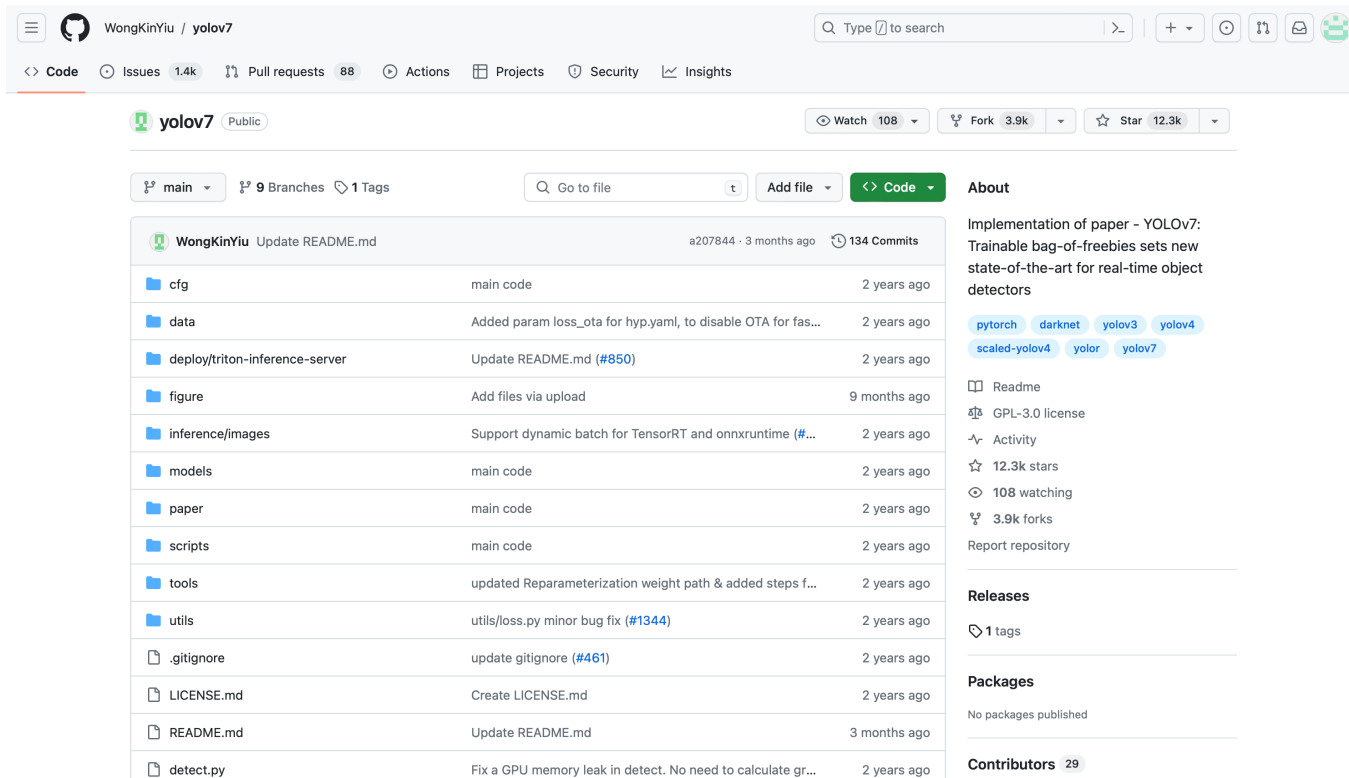


그림 1-10 얼굴 랜드마크 검출

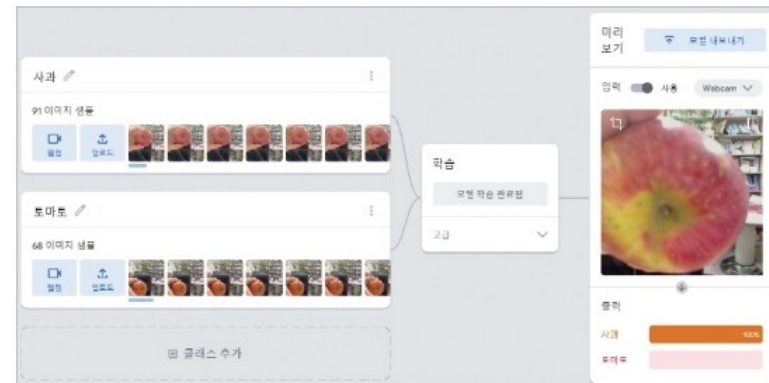


그림 1-12 티처블 머신

Scope

- 궁극적인 목표
 - 일반적인 상황에서 잘 작동하는 인간과 같은 시각 (강한 인공지능)
 - 영영 불가능하거나 먼 미래에 실현
- 현실적인 목표
 - 제한된 환경에서 특정 과업을 높은 성능으로 달성 (약한 인공지능)
 - 컴퓨터 비전 문제를 여러 세부 문제로 구분하고 세부 문제별 알고리즘을 구상

Challenges

- 기본 문제

- 분류

- 검출

- 분할

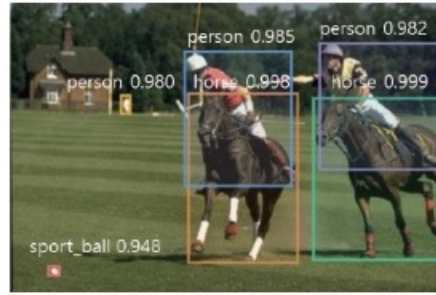
- 추적

- 행동 분석

- ...



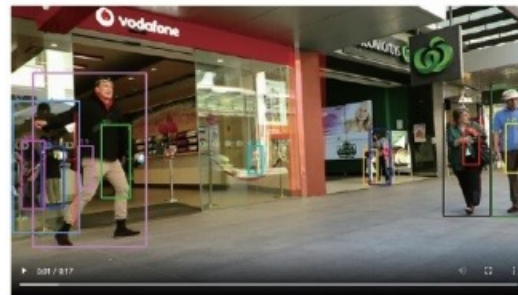
(a) 분류



(b) 검출



(c) 분할



(d) 추적(<https://motchallenge.net/vis/MOT17-09-SDP>)



(e) 행동 분석(<https://github.com/mostafa-saad/deep-activity-rec#dataset>)

그림 1-13 컴퓨터 비전이 풀어야 할 문제

Challenges

- 특정 상황에 따라 다양하게 변형
 - ex) 사과 따는 로봇 비전 → 사과 검출에만 집중 → 로봇 손을 위해 정확한 위치가 중요
 - ex) 방법용 홈캠 → 모든 상황을 인식할 필요가 있을까? → 특정 상황/특정 조건에만 작동하도록
- 다른 지능 요소와 협업
 - 가장 활발한 협업분야 = 자연어 처리
 - ex) 영상 설명하기
 - 지식 표현, 추론, 계획과 협업은 매우 소강 상태
 - 로봇과의 협업은 활발



그림 1-11 영상 설명

컴퓨터 비전 알고리즘과 프로그래밍

- 고전 컴퓨터 비전과 딥러닝 컴퓨터 비전
 - 2010년을 기점으로 방법론의 대전환
 - 고전 방법: 규칙 기반
 - 딥러닝 방법: 데이터 중심
 - 둘 다 이해하는 것이 중요
- 프로그래밍
 - 주로 Python을 사용
 - 산업계에서는 C, C++, Java를 주로 활용



References

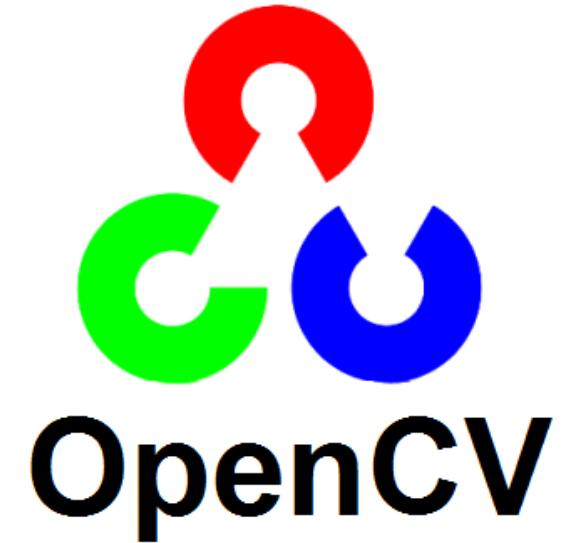
- Richard Szeliski 『computer Vision: Algorithms and Applications』 (2nd Edition, 2022)
 - 최신 컴퓨터 비전 내용을 방대하게 담음
 - 무료 pdf 제공 (<https://szeliski.org/Book>)

- Aston Zhang 『Dive into Deep Learning』(2022)
 - 딥러닝에 대한 최신 내용을 충실하게 전달
 - 무료 온라인 교과서. <https://d21.ai>

2. Python and OpenCV settings

OpenCV 개요

- 1999년 Intel사에서 개발된 IPL (Image Primitive Library)를 기반으로 발전
- 2000년대에 일반인에게 공개 (Open source) → 교육/상업 목적 모두 무료
 - OpenCV version 1.0 정식 배포
 - C 기반 라이브러리
 - IplImage (iplimage) 구조체 (객체)
- 2009년 version 2.0 배포 → Java / Python 지원 시작
- 2015년 version 3.0 배포 → 머신러닝 알고리즘 탑재
- 현재 version 4.x → 딥러닝 관련 알고리즘 일부 탑재
- Windows, Linux, macOS, Android, iOS 모두 지원
 - 교차 플랫폼 지원



History

표 2-1 OpenCV의 역사

연도	사건
1998	• 인텔 직원인 개리 브라드스키(Gary Bradski)가 아이디어 제안
1999	• 오픈 소스로 공개하기로 결정하고 이름을 OpenCV로 정함
2000	• CVPR 컨퍼런스에서 알파 버전 발표
2001-2005	• 5개의 베타 버전 발표
2005	• 스탠퍼드 대학교의 자율주행차인 스탠리의 개발 팀에 합류해 그랜드 챌린지 우승 • OpenCV Korea 출범(https://cafe.naver.com/opencv)

2006	• OpenCV 1.0(C 인터페이스) 공개 •  로고 완성
2009	• OpenCV 2.0(C++ 인터페이스) 공개 • 파이썬과 자바 인터페이스 지원
2012	• 안드로이드와 iOS 지원 시작 • 깃허브로 마이그레이션
2015	• OpenCV 3.0 공개
2016	• 자바스크립트 인터페이스 지원 시작 • 딥러닝을 지원하는 DNN 모듈 추가
2018	• OpenCV 4.0 공개 • 고속 처리를 지원하는 OpenVINO 공개
2020	• Computer Vision and Deep Learning 코스 개설 • 전용 보드인 OpenCV AI Kit 출시
2022	• OpenCV 4.6 공개

History

표 2-1 OpenCV의 역사

연도	사건
1998	• 인텔 직원인 개리 브라드스키(Gary Bradski)가 아이디어 제안
1999	• 오픈 소스로 공개하기로 결정하고 이름을 OpenCV로 정함
2000	• CVPR 컨퍼런스에서 알파 버전 발표
2001-2005	• 5개의 베타 버전 발표
2005	• 스탠퍼드 대학교의 자율주행차인 스탠리의 개발 팀에 합류해 그랜드 챌린지 우승 • OpenCV Korea 출범(https://cafe.naver.com/opencv)

OpenCV 설치 및 사용 메뉴얼

- <https://docs.opencv.org>
 - 프로그래밍 시 가장 많은 도움을 주는 사이트
 - chatGPT는 최신 업데이트 내용 반영 할 수 없음

OpenCV 4.9.0

Open Source Computer Vision

Main Page Related Pages Modules Namespaces Classes Files Examples Java documentation

OpenCV-Python Tutorials Introduction to OpenCV

Install OpenCV-Python in Windows

Goals

In this tutorial

- We will learn to setup OpenCV-Python in your Windows system.

Below steps are tested in a Windows 7-64 bit machine with Visual Studio 2010 and Visual Studio 2012. The screenshots shows VS2012.

Installing OpenCV from prebuilt binaries

1. Below Python packages are to be downloaded and installed to their default locations.
 - a. Python 3.x (3.4+) or Python 2.7.x from here.
 - b. Numpy package (for example, using `pip install numpy` command).
 - c. Matplotlib (`pip install matplotlib`) (*Matplotlib is optional, but recommended since we use it a lot in our tutorials*).
2. Install all packages into their default locations. Python will be installed to `C:/Python27/` in case of Python 2.7.
3. After installation, open Python IDLE. Enter `import numpy` and make sure Numpy is working fine.
4. Download latest OpenCV release from GitHub or SourceForge site and double-click to extract it.
5. Goto `opencv/build/python2.7` folder.
6. Copy `cv2.pyd` to `C:/Python27/lib/site-packages`.
7. Copy the `opencv_world.dll` file to `C:/Python27/lib/site-packages`
8. Open Python IDLE and type following codes in Python terminal.

```
>>> import cv2 as cv
>>> print( cv._version_ )
```

If the results are printed out without any errors, congratulations !!! You have installed OpenCV-Python successfully.

Nightly	Doxygen HTML	Sphinx HTML	Javadoc HTML
<ul style="list-style-type: none">• upcoming 5.x• 4.x• 3.4(3.4.x)• 2.4(2.4.x)	<ul style="list-style-type: none">• 4.9.0 (zip - 93.8 Mb, tar.xz - 65.0 Mb)• 4.8.0 (zip - 92.3 Mb, tar.xz - 63.3 Mb)• 3.4.20 (zip - 75.3 Mb, tar.xz - 53.6 Mb)• 4.7.0 (zip - 86.7 Mb, tar.xz - 53.6 Mb)• 3.4.19 (zip - 75.3 Mb, tar.xz - 53.6 Mb)• 4.6.0 (zip - 85.5 Mb, tar.xz - 59.6 Mb)• 3.4.18 (zip - 75.0 Mb, tar.xz - 53.3 Mb)• 4.5.5 (zip - 86.1 Mb, tar.xz - 60.3 Mb)• 3.4.17 (zip - 75.8 Mb, tar.xz - 54.1 Mb)• 4.5.4 (zip - 86.4 Mb, tar.xz - 60.9 Mb)• 3.4.16 (zip - 76.7 Mb, tar.xz - 54.9 Mb)• 4.5.3 (zip - 85.7 Mb, tar.xz - 59.8 Mb)• 3.4.15 (zip - 76.7 Mb, tar.xz - 54.3 Mb)• 4.5.2 (zip - 85.4 Mb, tar.xz - 59.6 Mb)• 3.4.14 (zip - 76.7 Mb, tar.xz - 54.3 Mb)• 4.5.1 (zip - 84.4 Mb, tar.xz - 59.0 Mb)• 3.4.13 (zip - 76.5 Mb, tar.xz - 54.2 Mb)• 4.5.0 (zip - 82.3 Mb, tar.xz - 57.7 Mb)• 3.4.12 (zip - 76.6 Mb, tar.xz - 54.2 Mb)• 4.4.0 (zip - 80.9 Mb, tar.xz - 57.4 Mb)• 3.4.11 (zip - 75.9 Mb, tar.xz - 54.2 Mb)• 4.3.0 (zip - 80.6 Mb, tar.xz - 57.4 Mb)• 3.4.10 (zip - 75.9 Mb, tar.xz - 54.3 Mb)• 4.2.0 (zip - 80.3 Mb, tar.xz - 57.3 Mb)• 3.4.9 (zip - 75.8 Mb, tar.xz - 54.2 Mb)• 4.1.2 (zip - 80.1 Mb, tar.xz - 57.3 Mb)• 3.4.8 (zip - 76.7 Mb, tar.xz - 53.1 Mb)• 4.1.1 (zip - 79.8 Mb, tar.xz - 56.7 Mb)• 3.4.7 (zip - 77.4 Mb, tar.xz - 55.2 Mb)• 4.1.0 (zip - 78.8 Mb, tar.xz - 56.4 Mb)• 3.4.6 (zip - 76.8 Mb, tar.xz - 55.0 Mb)• 4.0.1 (zip - 78.9 Mb, tar.xz - 56.0 Mb)• 3.4.5 (zip - 77.5 Mb, tar.xz - 55.2 Mb)• 4.0.0 (zip - 78.7 Mb, tar.xz - 56.0 Mb)• 3.4.4 (zip - 77.3 Mb, tar.xz - 55.2 Mb)• 4.0.0-rc (zip - 79.1 Mb, tar.xz - 55.7 Mb)• 4.0.0-beta (zip - 79.1 Mb, tar.xz - 55.5 Mb)• 4.0.0-alpha (zip - 77.7 Mb, tar.xz - 54.9 Mb)• 3.4.3 (zip - 77.2 Mb, tar.xz - 54.9 Mb)• 3.4.2 (zip - 76.9 Mb, tar.xz - 54.7 Mb)	<ul style="list-style-type: none">• 2.4.13.7 (zip - 41.3 Mb, tar.xz - 26.6 Mb)• 2.4.13.6 (zip - 41.3 Mb, tar.xz - 26.6 Mb)• 2.4.13.5 (zip - 41.3 Mb, tar.xz - 26.6 Mb)• 2.4.13.4 (zip - 41.3 Mb, tar.xz - 26.6 Mb)• 2.4.13.3 (zip - 41.3 Mb, tar.xz - 26.5 Mb)• 2.4.13.2 (zip - 41.3 Mb, tar.xz - 26.7 Mb)• 2.4.13 (zip - 40.5 Mb, tar.xz - 25.9 Mb)• 2.4.12 (zip - 40.5 Mb, tar.xz - 25.9 Mb)• 2.4.11 (zip - 57.8 Mb, tar.xz - 37.9 Mb)• 3.0-last-rc1 (zip - 65.2 Mb, tar.xz - 43.9 Mb)• 2.4.10 (zip - 57.8 Mb, tar.xz - 37.9 Mb)• 3.0-beta (zip - 64.0 Mb, tar.xz - 43.2 Mb)• 3.0-alpha (zip - 60.7 Mb, tar.xz - 38.8 Mb)• 2.4.9 (zip - 56.8 Mb, tar.xz - 36.4 Mb)• 2.4.8 (zip - 56.4 Mb, tar.xz - 34.4 Mb)• 2.4.7 (zip - 54.3 Mb, tar.xz - 34.0 Mb)• 2.4.6 (zip - 47.7 Mb, tar.xz - 30.1 Mb)• 2.4.0 (zip - 33.0 Mb, tar.xz - 22.6 Mb)• 2.4.5 (zip - 42.8 Mb, tar.xz - 23.2 Mb)• 2.4.4 (zip - 41.9 Mb, tar.xz - 22.7 Mb)• 2.4.4-beta (zip - 31.7 Mb, tar.xz - 19.7 Mb)• 2.4.3.2 (zip - 25.2 Mb, tar.xz - 14.7 Mb)• 2.4.3 (zip - 29.5 Mb, tar.xz - 15.3 Mb)• 2.4.3-rc (zip - 39.9 Mb, tar.xz - 26.8 Mb)• 2.4.2 (zip - 40.4 Mb, tar.xz - 26.1 Mb)• 2.4.1 (zip - 27.0 Mb, tar.xz - 19.5 Mb)• 2.4-beta (zip - 20.5 Mb, tar.xz - 9.7 Mb)• 2.3 (zip - 31.5 Mb, tar.xz - 22.0 Mb)	<ul style="list-style-type: none">• Note: Javadoc is embedded into doxygen pages since OpenCV 3.4.1+• 3.0.0 (zip - 0.5 Mb, tar.xz - 0.2 Mb)• 3.0-rc1 (zip - 0.4 Mb, tar.xz - 0.1 Mb)• 2.4.11 (zip - 0.6 Mb, tar.xz - 0.2 Mb)• 2.4.9 (zip - 0.6 Mb, tar.xz - 0.2 Mb)• 2.4.8 (zip - 0.6 Mb, tar.xz - 0.2 Mb)• 2.4.7.1 (zip - 0.6 Mb, tar.xz - 0.2 Mb)• 2.4.7 (zip - 0.6 Mb, tar.xz - 0.2 Mb)• 2.4.6 (zip - 0.6 Mb, tar.xz - 0.2 Mb)• 2.4.5 (zip - 0.6 Mb, tar.xz - 0.2 Mb)• 2.4.4 (zip - 0.6 Mb, tar.xz - 0.2 Mb)• 2.4.4-beta (zip - 0.6 Mb, tar.xz - 0.2 Mb)• 2.4.3.2 (zip - 0.6 Mb, tar.xz - 0.2 Mb)• 2.4.3.1 (zip - 0.6 Mb, tar.xz - 0.2 Mb)• 2.4.3 (zip - 0.6 Mb, tar.xz - 0.2 Mb)• 2.4.2 (zip - 0.5 Mb, tar.xz - 0.2 Mb)

Python development environment

- 기본적인 실습: Windows (실습실 PC 또는 개인 Laptop 활용)
- 서버 제공: 모델 학습 시 활용
 - NVIDIA GeForce RTX 4060 or 4070 (랜덤)
 - **반드시 종강 후 데이터 백업할 것** (방학 중 초기화되도록 설정)
- IDE: VSCode
 - Colab 추천하지 않음
- Compile env.: Anaconda
 - 실습실 PC 활용하는 경우 **anaconda 환경 필수 적용**
 - 서버활용 시 **anaconda 환경 필수 적용** (로컬 환경 구동 시 타인과의 버전 충돌로 인한 불이익 no excuse)

Assignment

- 실습실 PC 또는 개인 laptop (windows)에 Python 개발 환경 세팅 및 OpenCV 설치

- 콘솔화면에서 OpenCV 설치 버전 캡처한 이미지 제출

```
(bj) C:\Users\user\LectureSourceGit\lecture_computervision\2024_cv_lecture\Chap01>python opencv_install.py
OpenCV is already installed. Current version: 4.7.0

(bj) C:\Users\user\LectureSourceGit\lecture_computervision\2024_cv_lecture\Chap01>█
```

End of slide
