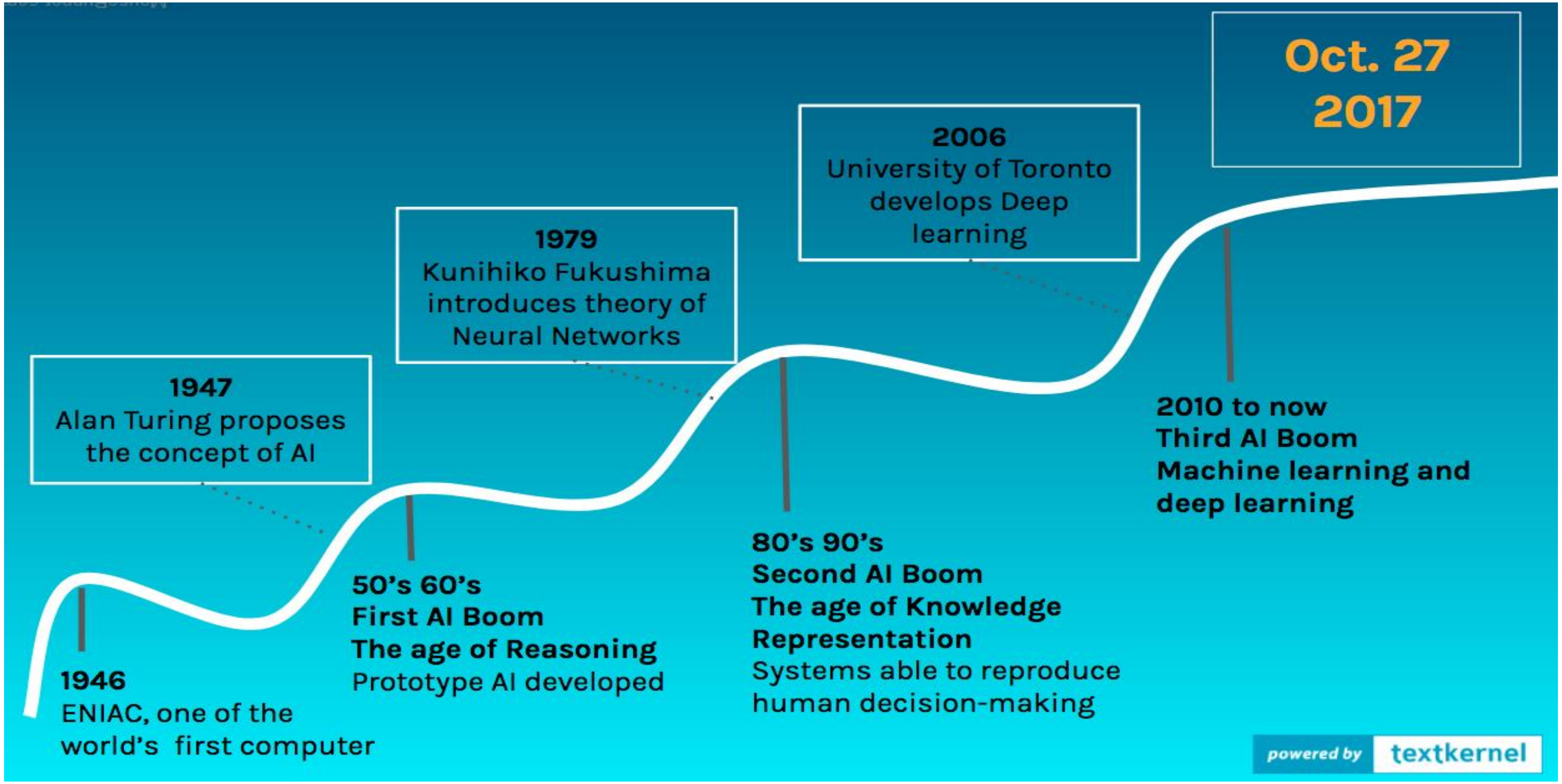
Deep Learning Seminar

1. Overview

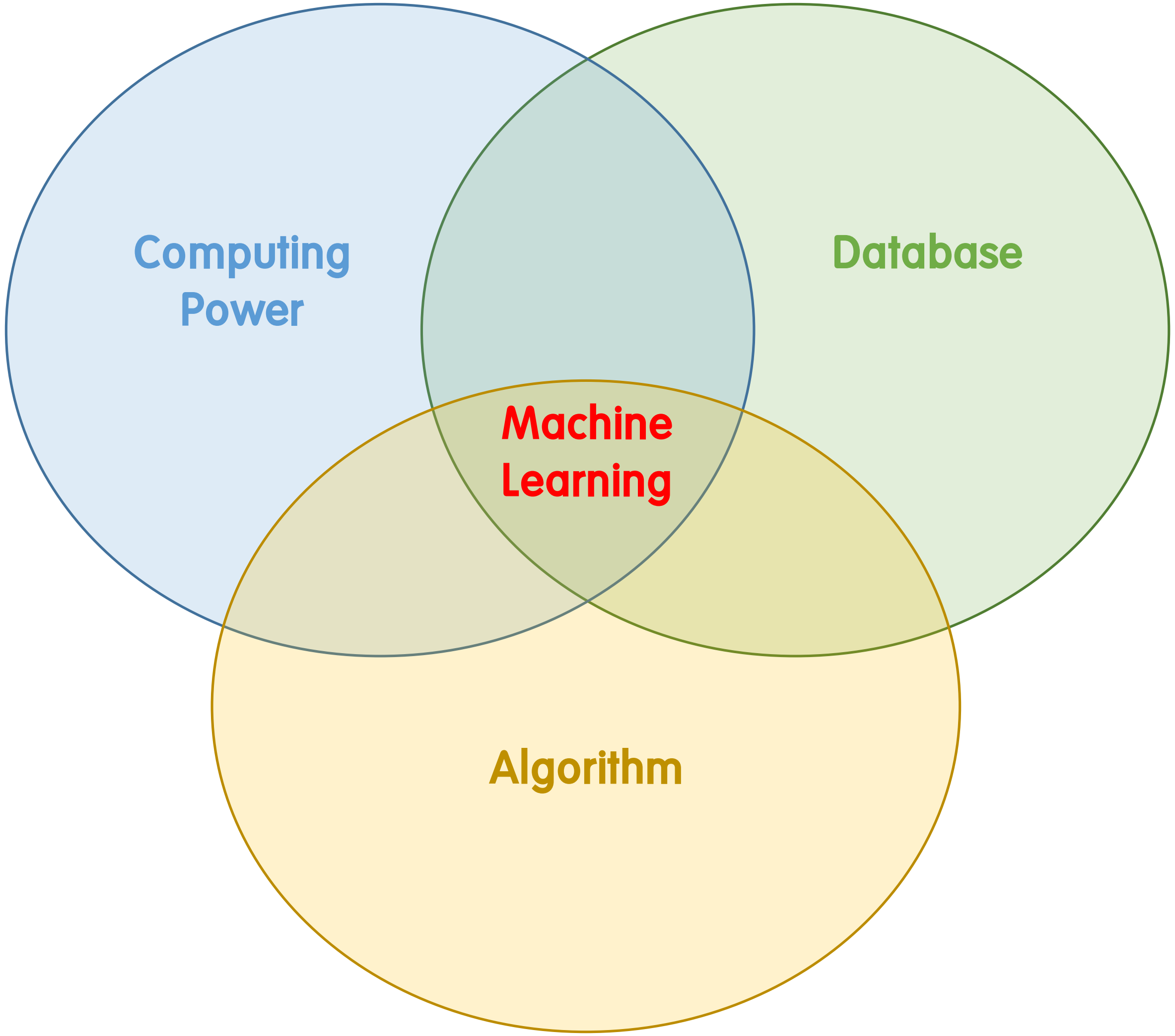
Contents

Overview

AI History


Oct. 27
2017
2006
University of Toronto develops Deep learning
1979
Kunihiko Fukushima introduces theory of Neural Networks
1947
Alan Turing proposes the concept of AI
2010 to now
Third AI Boom
Machine learning and deep learning
80's 90's
Second AI Boom
The age of Knowledge Representation
Systems able to reproduce human decision-making
50's 60's
First AI Boom
The age of Reasoning
Prototype AI developed
1946
ENIAC, one of the world's first computer
powered by textkernel

Machine Learning ?



Current revolution of AI

Current revolution of AI

Current revolution of AI

Current revolution of AI

Human: 95% vs. FaceNet of Google: **99.6%**

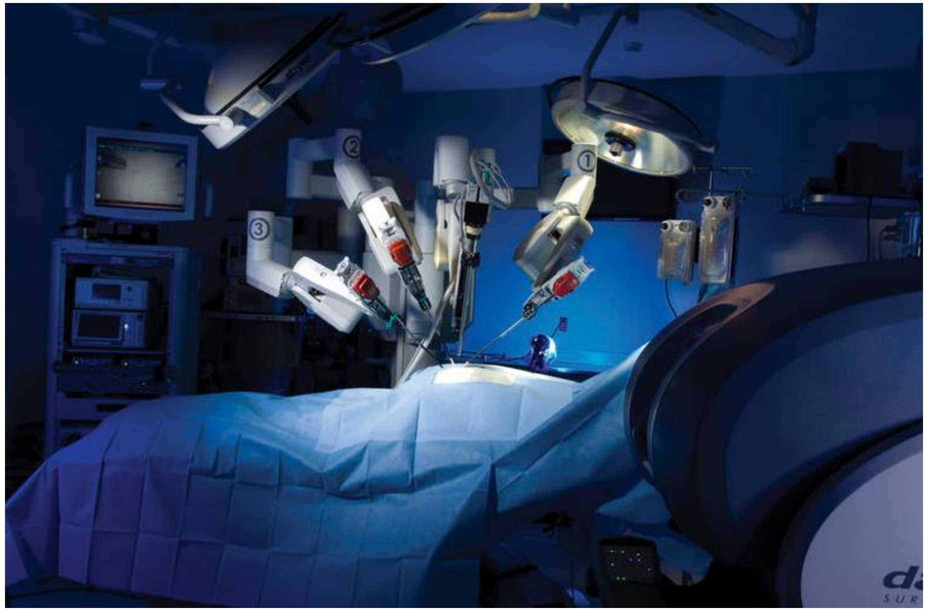
Current revolution of AI

- Watson learned
  - 600,000 medical evidence, 42 medical journals and clinical trials (2millon pages), 69 guidelines, 61,540 clinical trials
  - 1,500 lung cancer cases, physician notes, lab results and clinical research
  - 14,700 hrs of hand-on training

Current revolution of AI

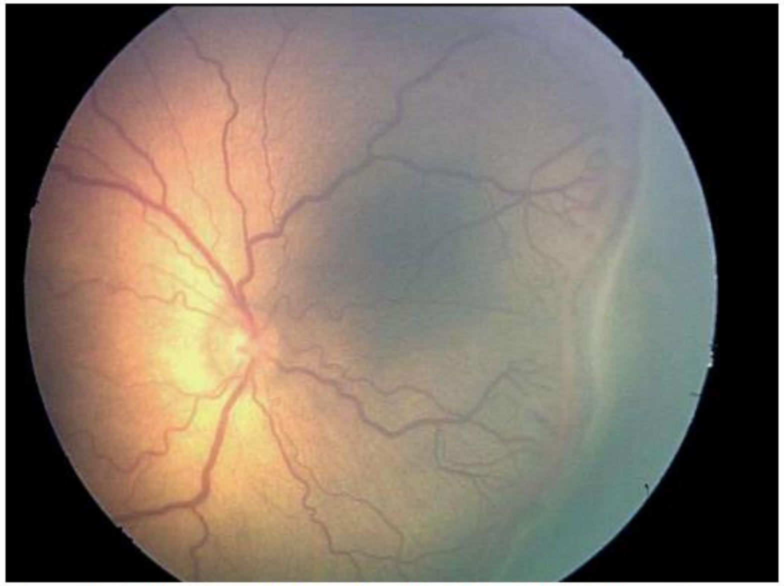
- 46 billion data points
- Predict medical outcomes
  - die in the hospital
  - be discharged and readmitted
  - final diagnosis

Current revolution of AI



- Robotic surgery
  - Smaller scars, Shorter hospitalization time, lower risk for adverse side effects, faster overall recovery time, less pain, no blood donation prior to surgery, less blood loss, quicker return to normal activity

Current revolution of AI



This image of an eye shows how twisted and dilated vessels of the retina can indicate retinopathy of prematurity, or ROP, the leading cause of childhood blindness. Credit: Michael Chiang/OHSU

- Retina scanning is used
  - to help diabetic patients
  - To detect preemie blindness
  - To diagnose heart disease

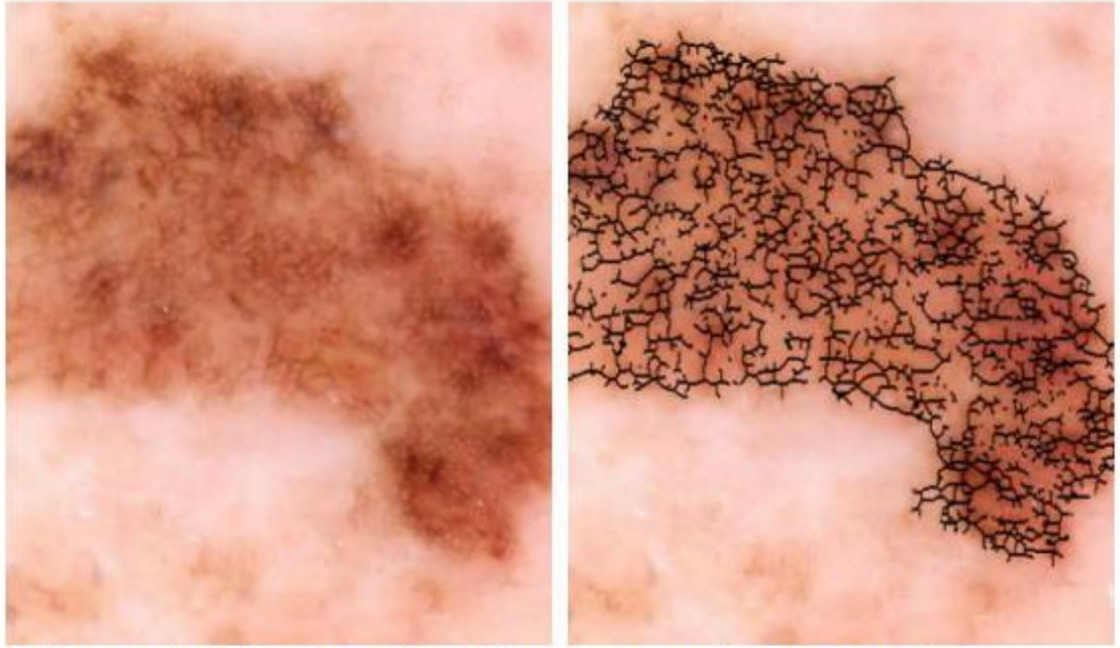
Current revolution of AI

Cataract diagnosis

Current revolution of AI

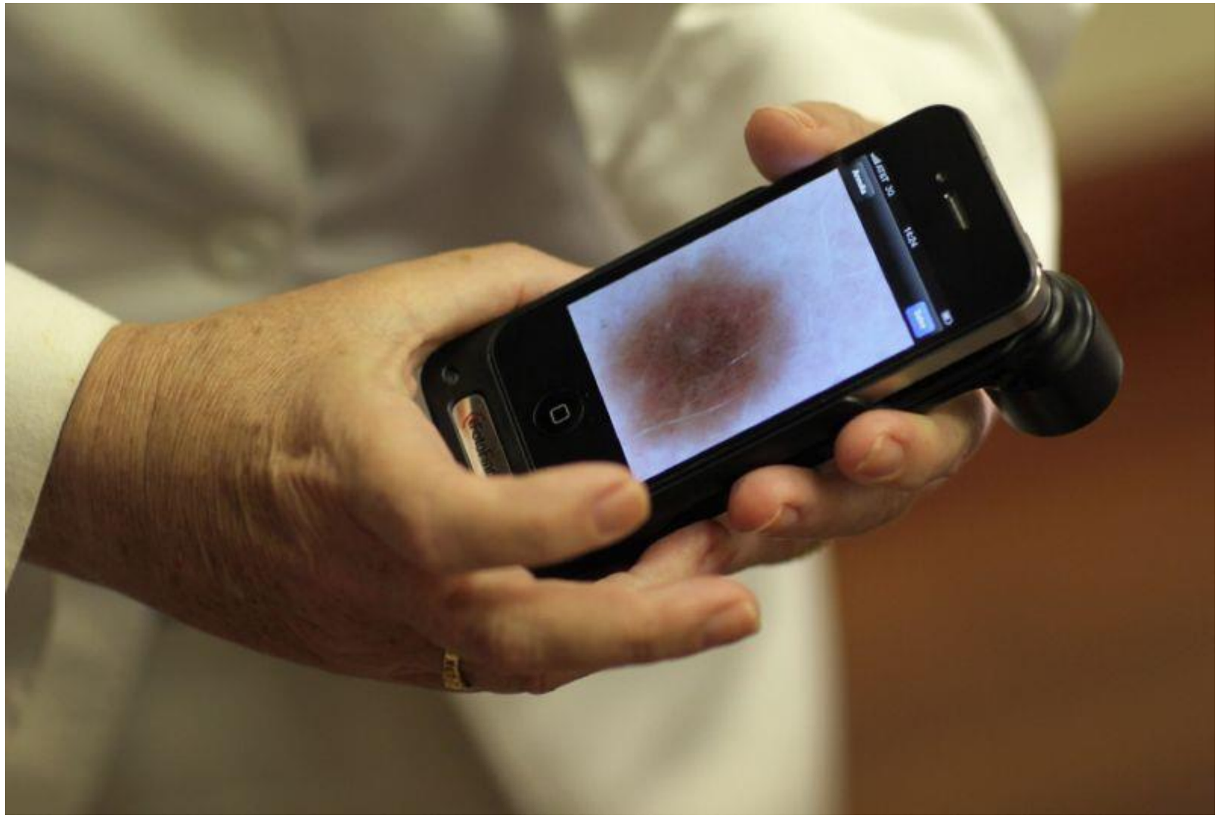
Researchers develop automated melanoma detector for skin cancer screening

December 25, 2016 in Medicine & Health / Cancer



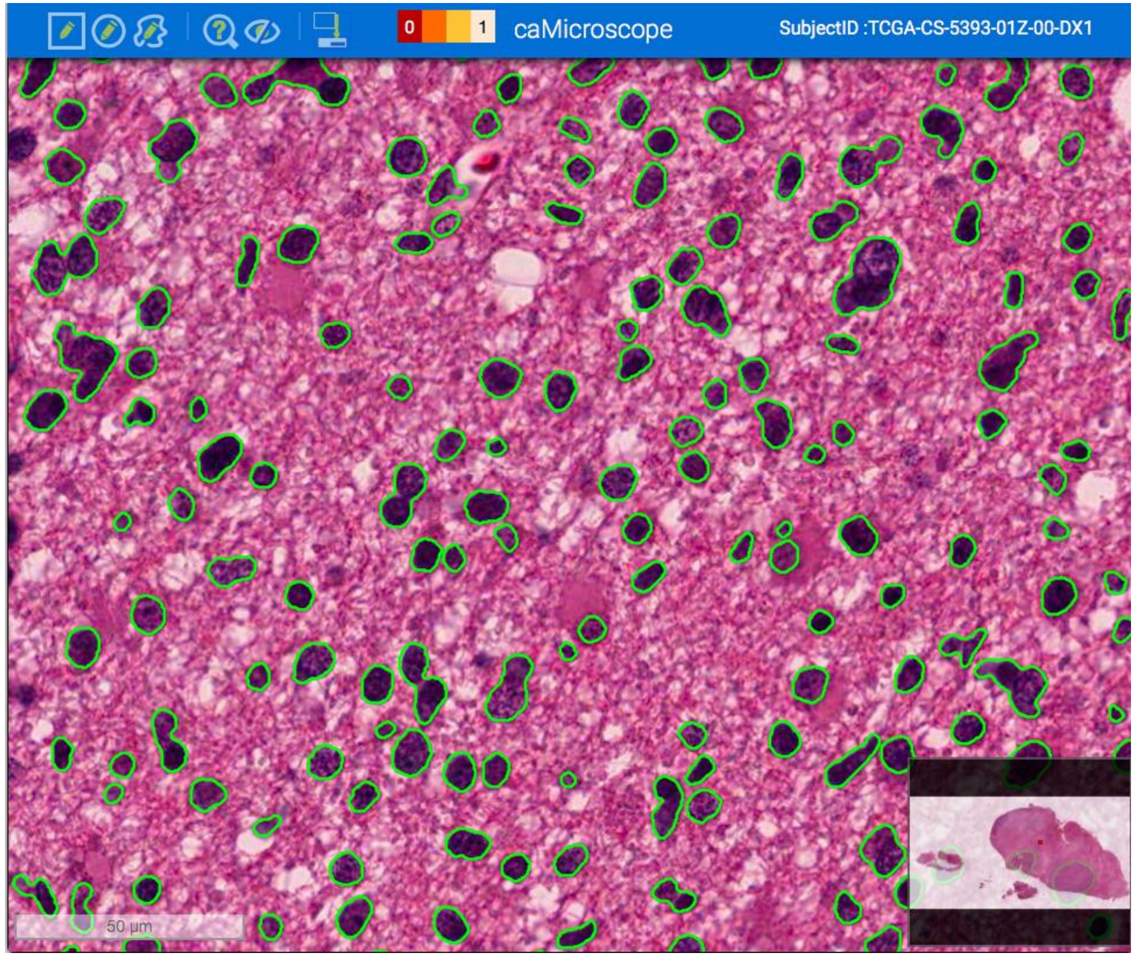
Malignant or benign?: An image of a skin lesion is processed by a new technology to extract quantitative data, such as irregularities in the shape of pigmented skin, which could help doctors determine if the growth is cancerous. Credit: Rockefeller University

Current revolution of AI



A computer was better than human dermatologists at detecting skin cancer in a study that pitted human against machine in the quest for better, faster diagnostics, researchers said (AFP Photo/JOE RAEDLE)

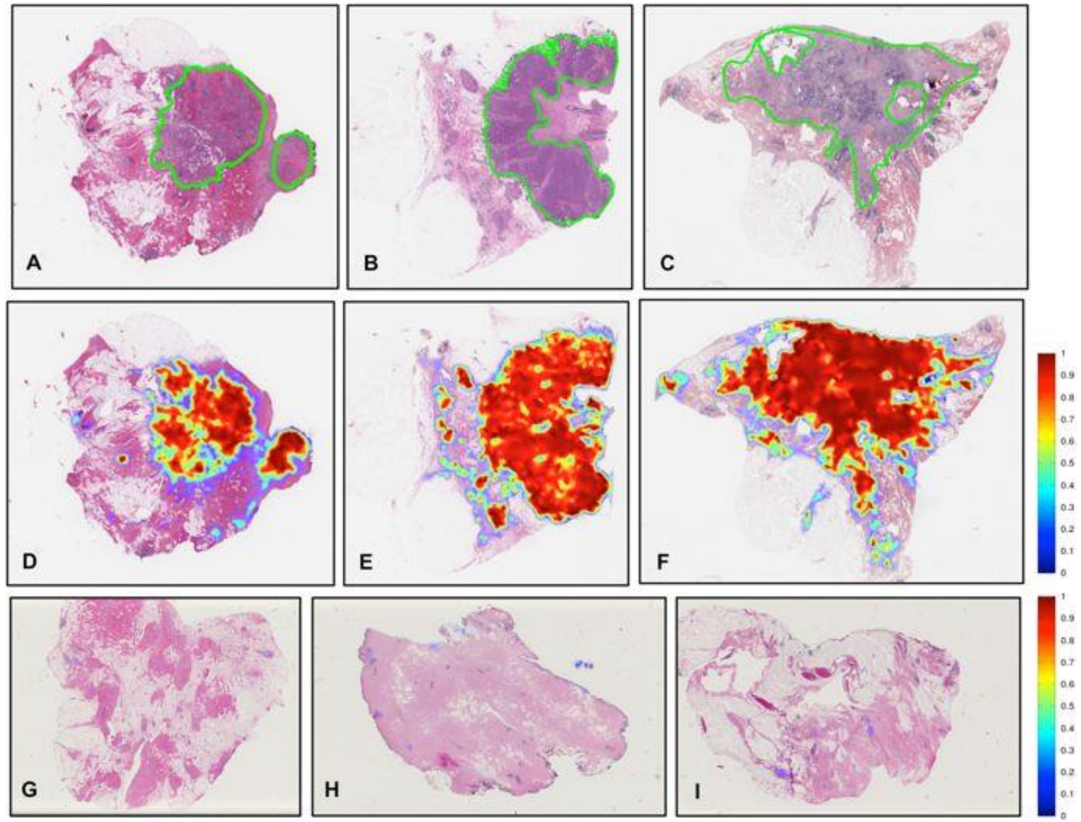
Current revolution of AI



- Pathology

Tumor segmentation

Current revolution of AI

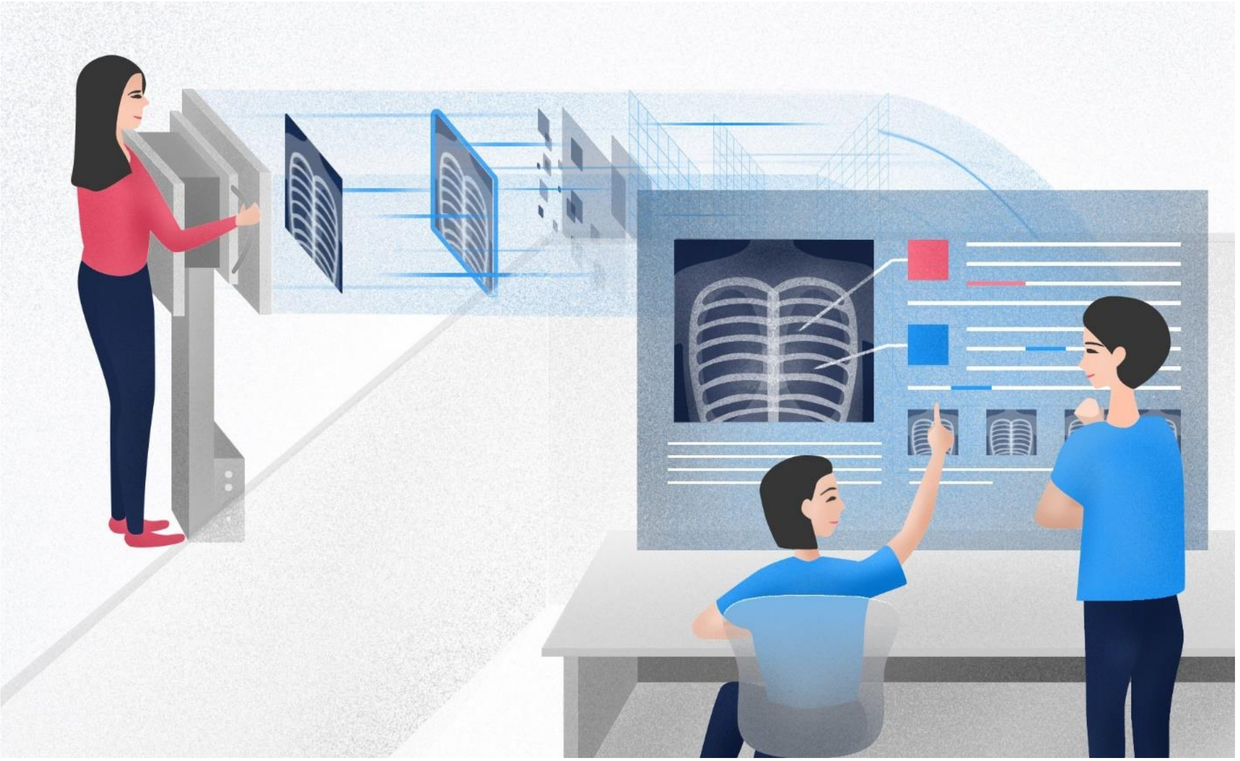


Prostate Cancer Pathological Slide Segmentation

Cruz-Roa. "Accurate and reproducible invasive breast cancer detection in whole-slide images: A Deep Learning approach for quantifying tumor extent"
From: https://www.nature.com/articles/srep46450

Current revolution of AI

Current revolution of AI



Automatic Generation of Medical Imaging Reports

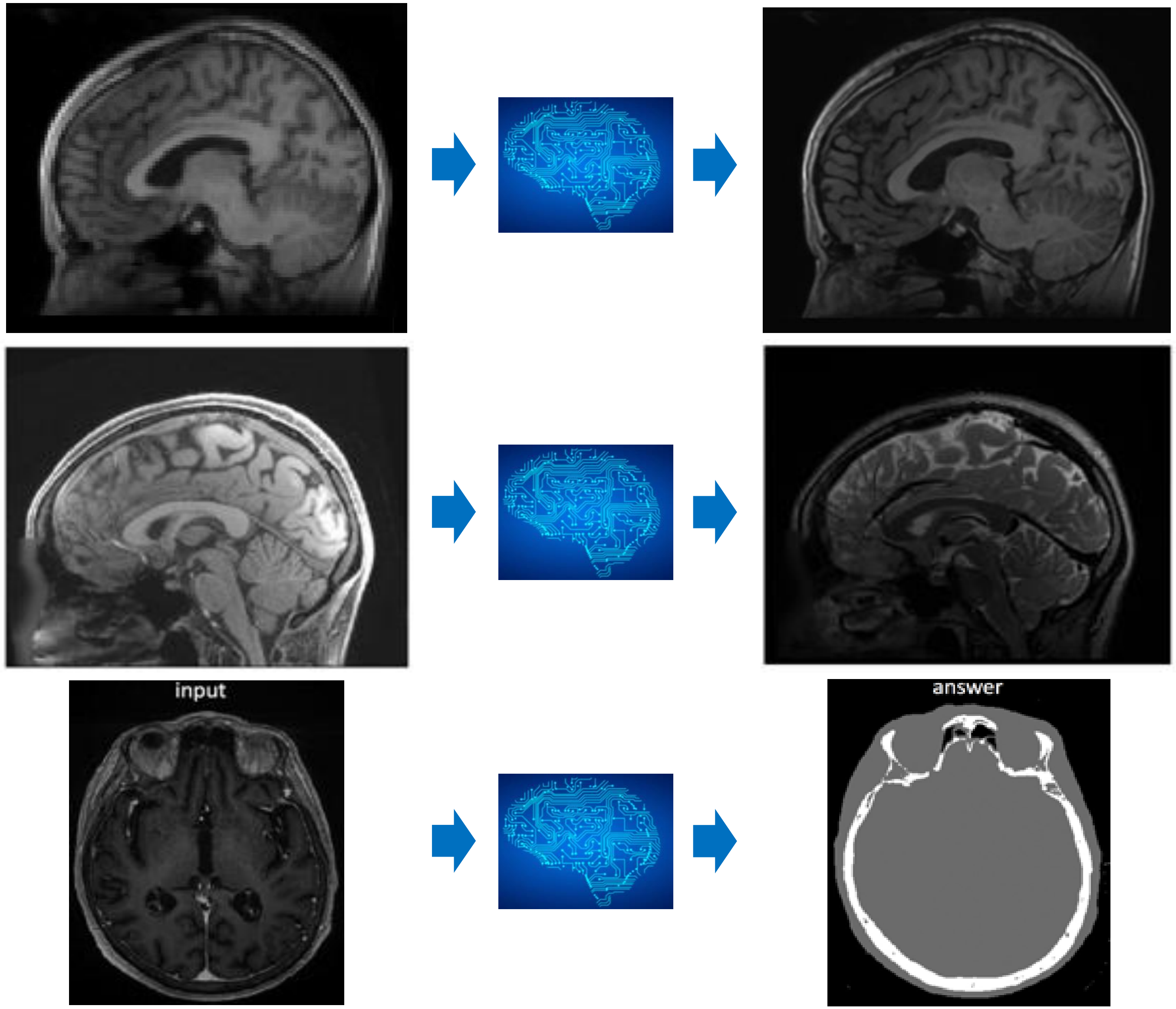
https://medium.com/@Petuum/on-the-automatic-generation-of-medical-imaging-reports-7d0a7748fe3d

Current revolution of AI

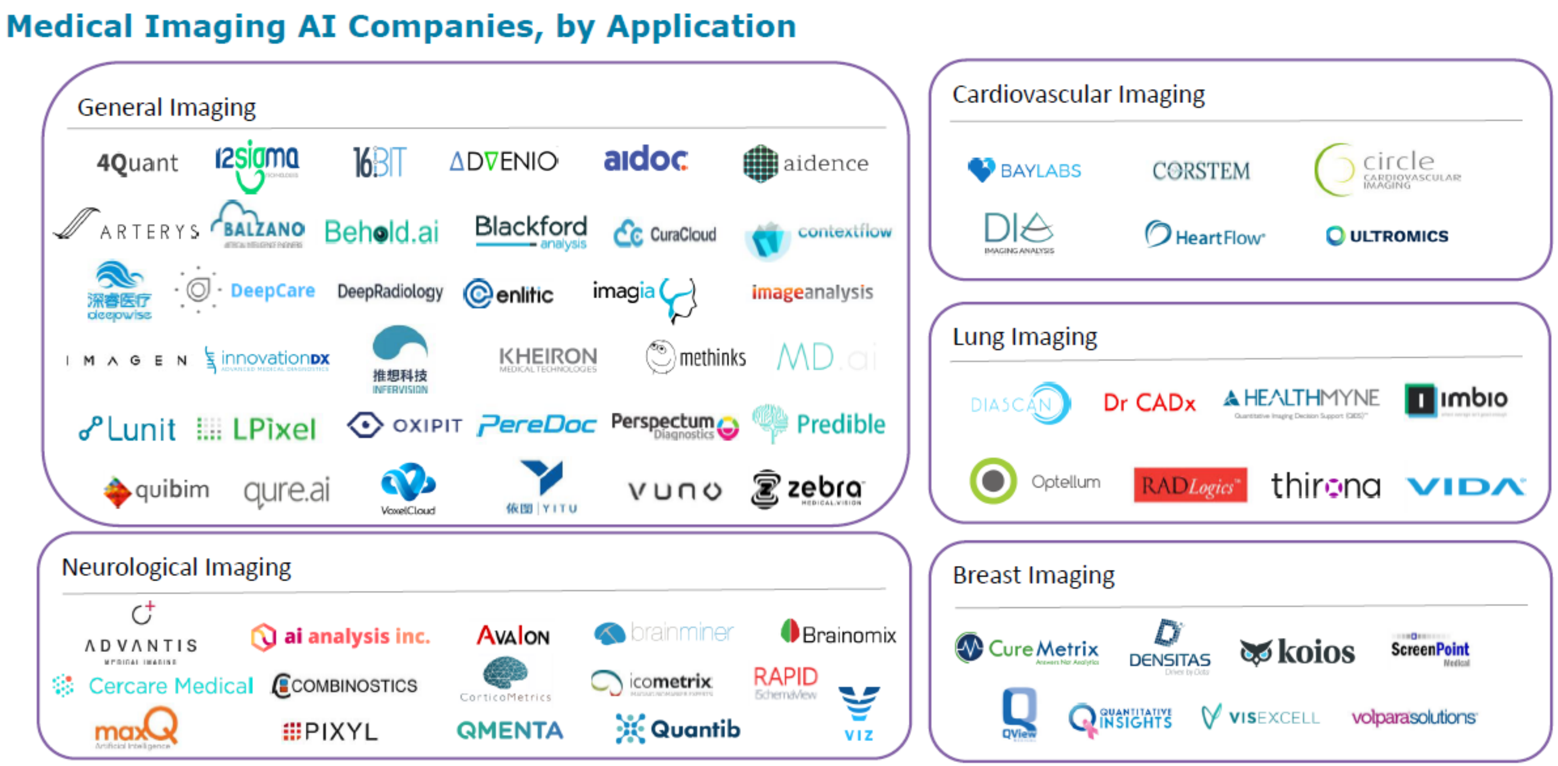
Quality/Speed Improvement

Contrast Conversion

Modality Conversion

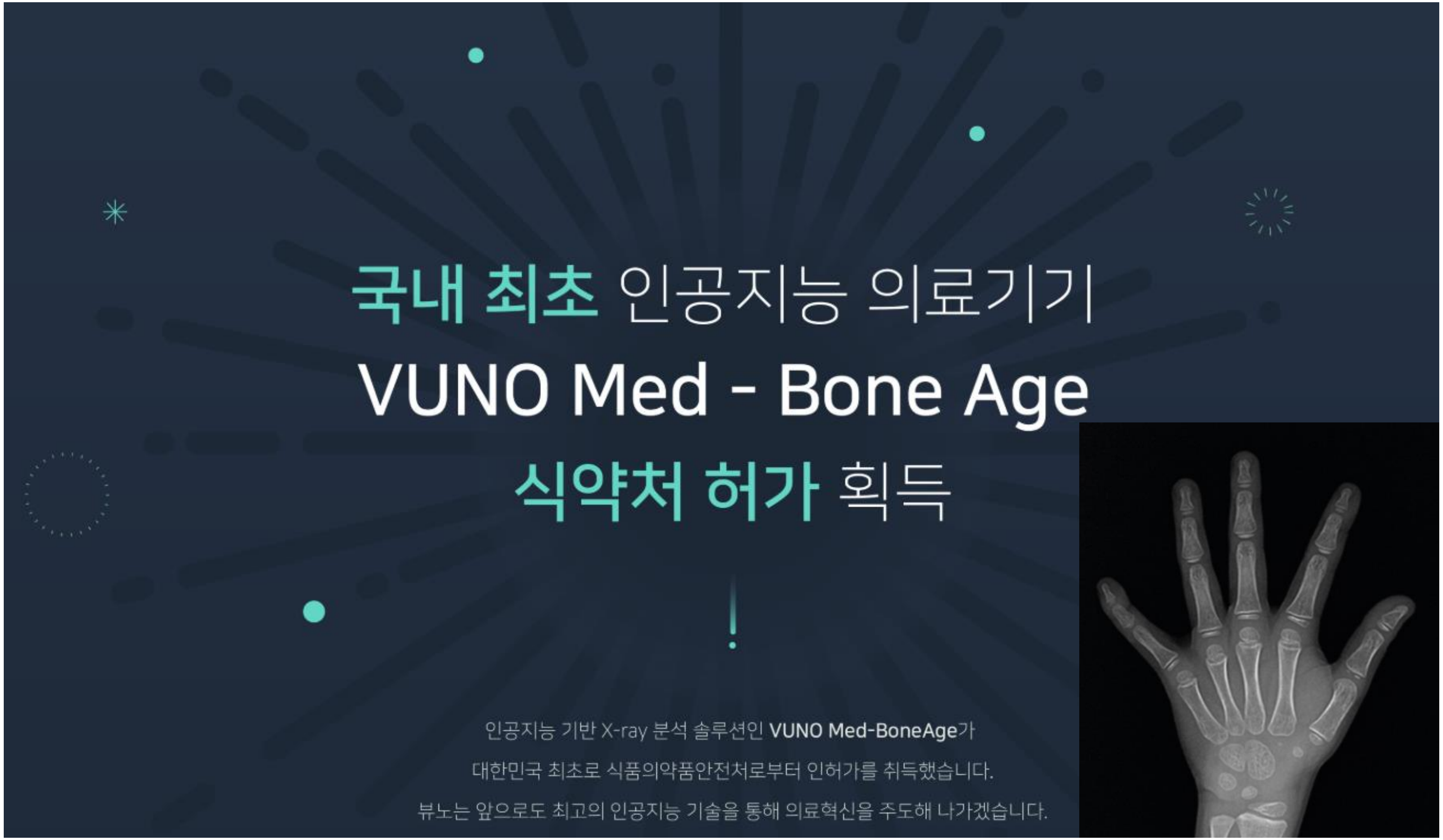


Medical AI Companies



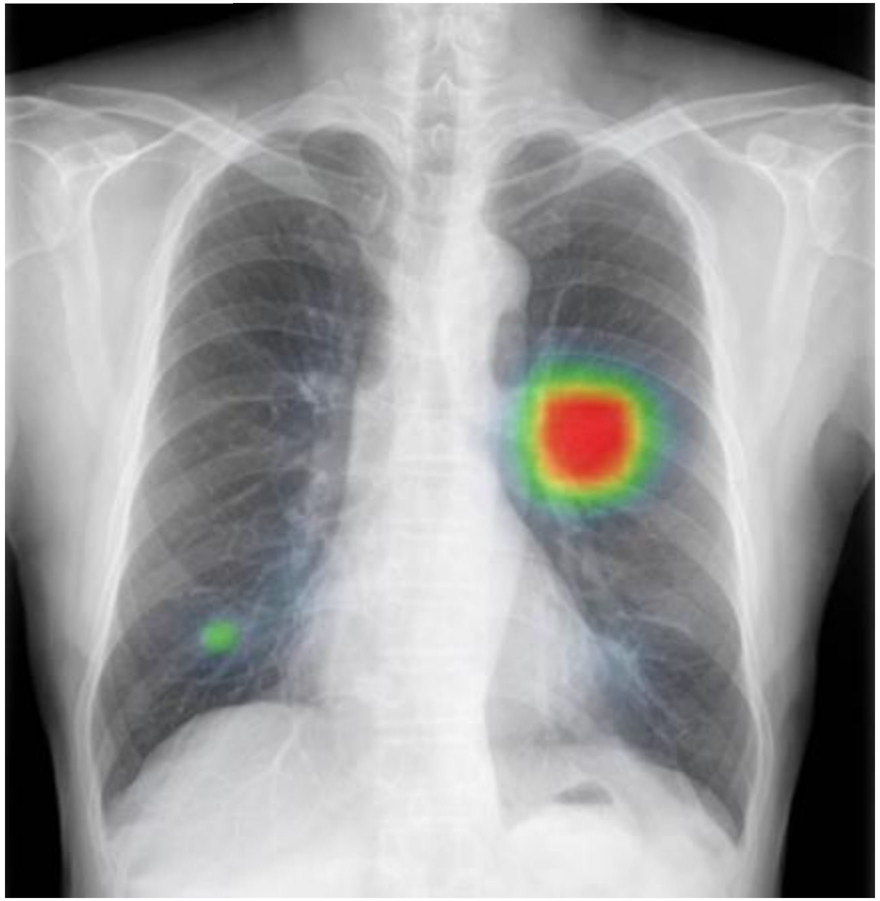
- Over 80 start-ups in medical imaging AI
- Services: diagnosis (lesion detection and segmentation)
  report writing/workflow optimization
  reconstruction
- Total investment in 2017 $234M

Medical AI Companies



Medical AI Companies

Lunit



- Chest x-ray nodule detection
- 97% AUC for lunc cancer
- 99% AUC for Pneumothorax
- 99% AUC for tuberculosis
- 98% AUC for Pneumonia
- 의사의 판독정확도 20% 향상
- 식약처허가

Medical AI Companies

- Big revolution for diagnosis, prognosis, segmentation, identification, report generating, workflow optimization…

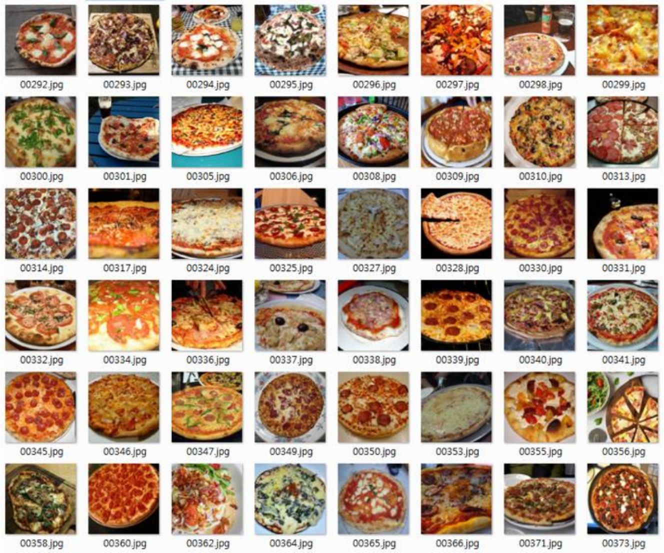
- Still questions on generation, regulation issues, billing to insurance, workflow integration

Are you a believer?

MRI (1,000)

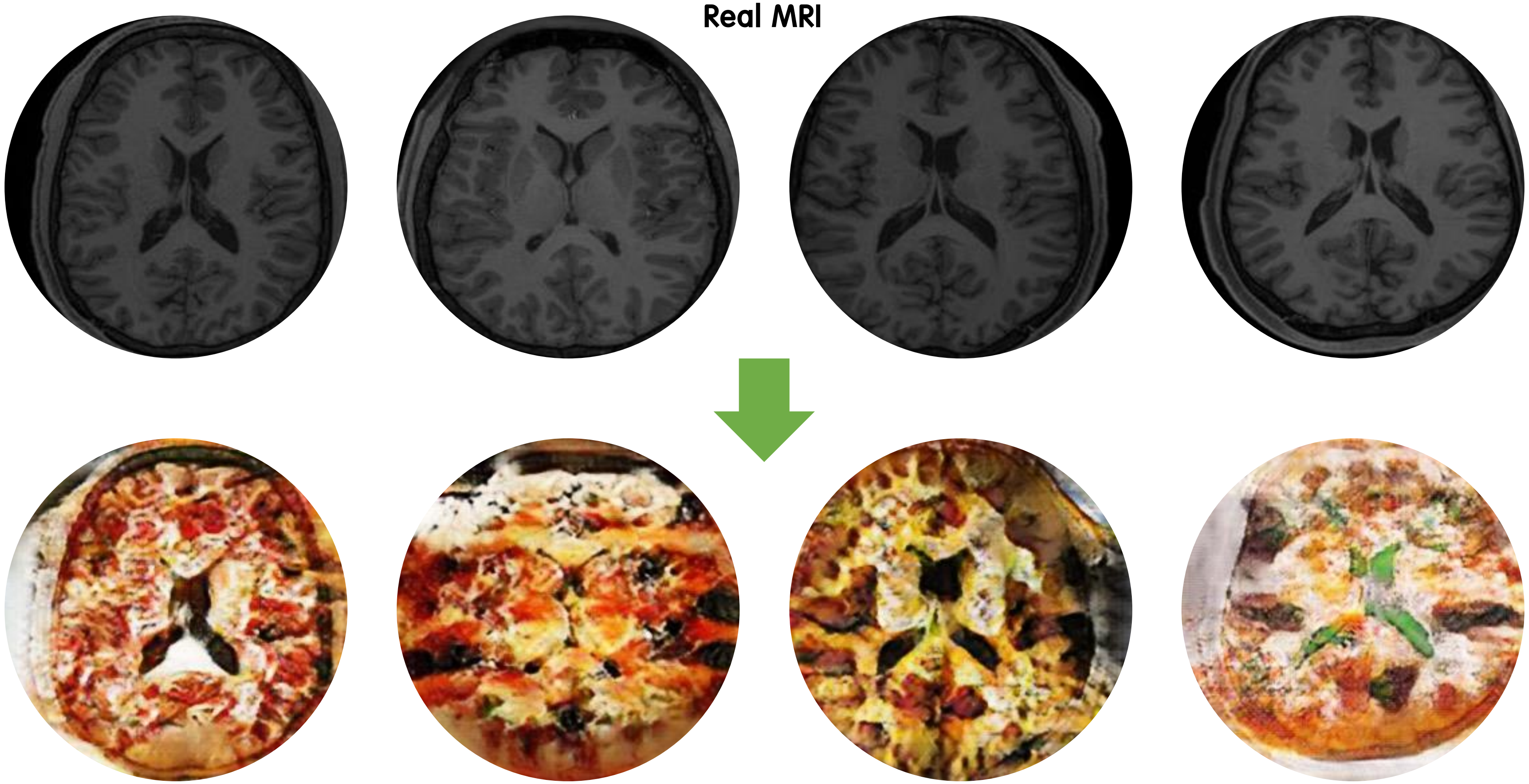
ML

Pizza (1,000)



Are you a believer?

Real MRI

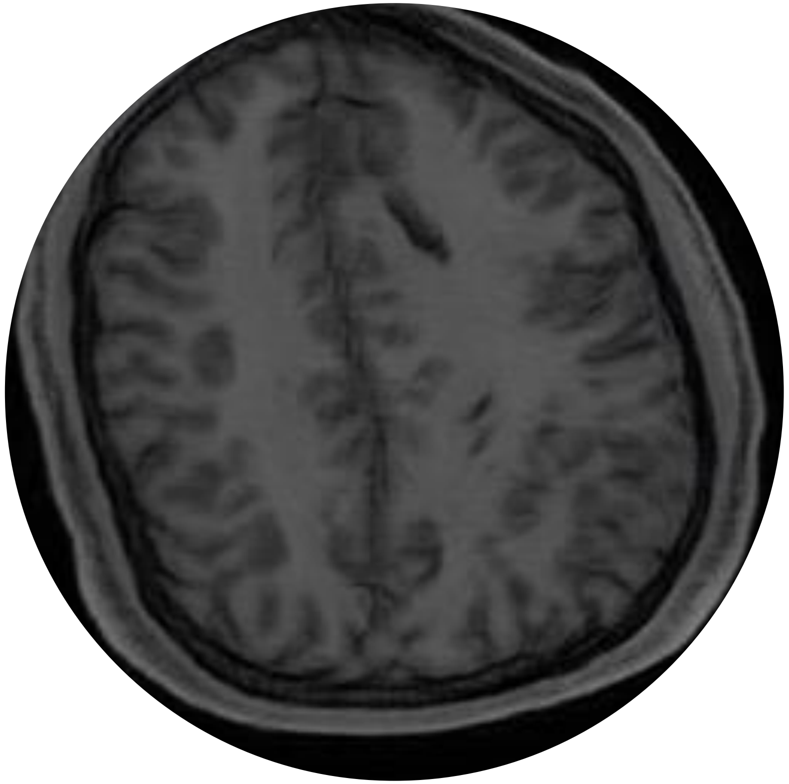


Fake Pizza (generated from MRI!)

Are you a believer?

Real Pizza

Fake MRI



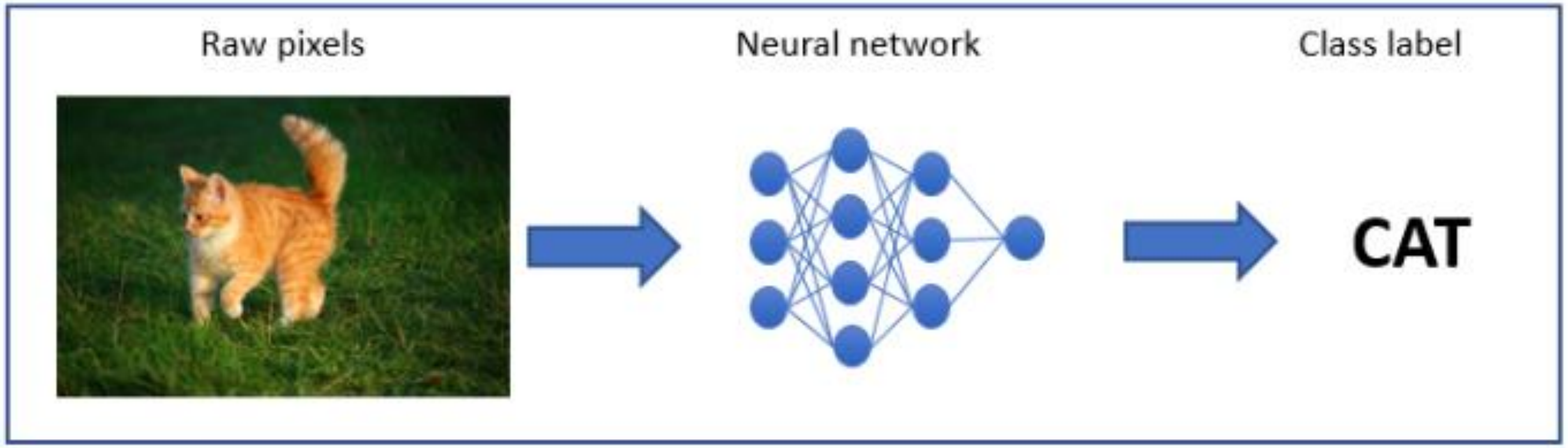
2. What can we do

using deep learning ?

What can we do?

- Classification

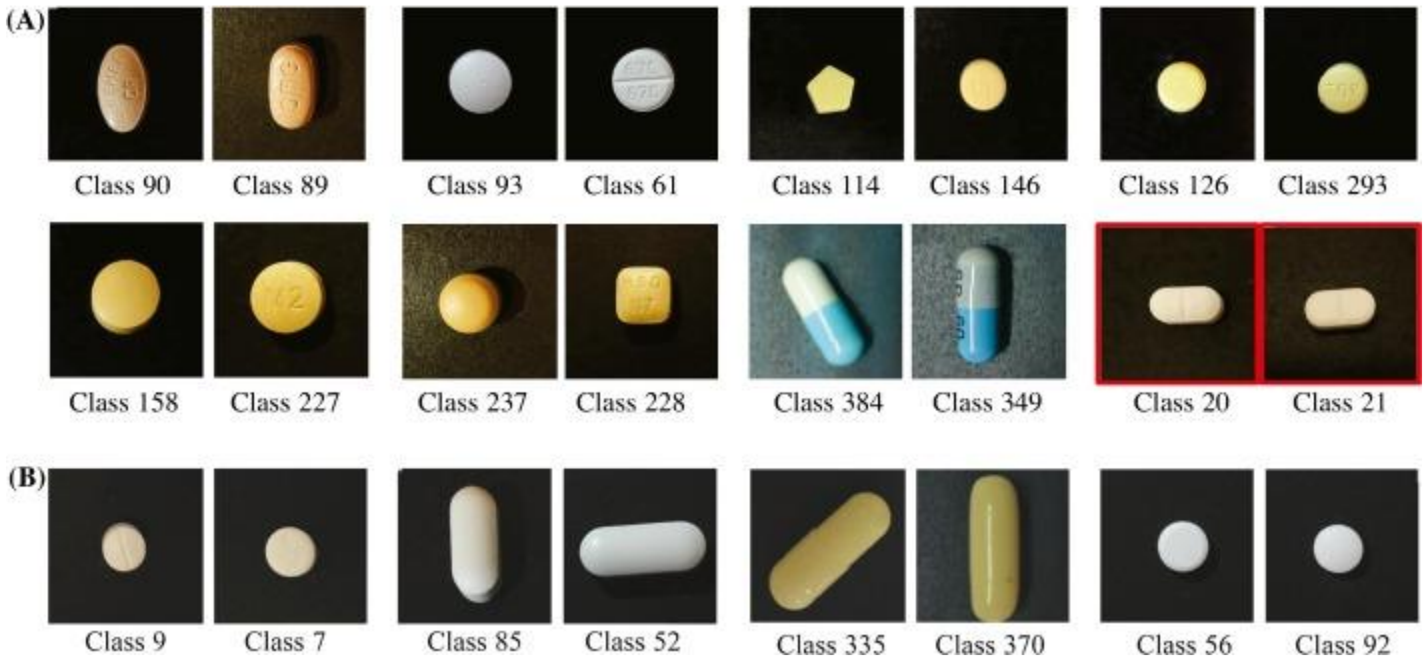
Example



What can we do?

- Classification

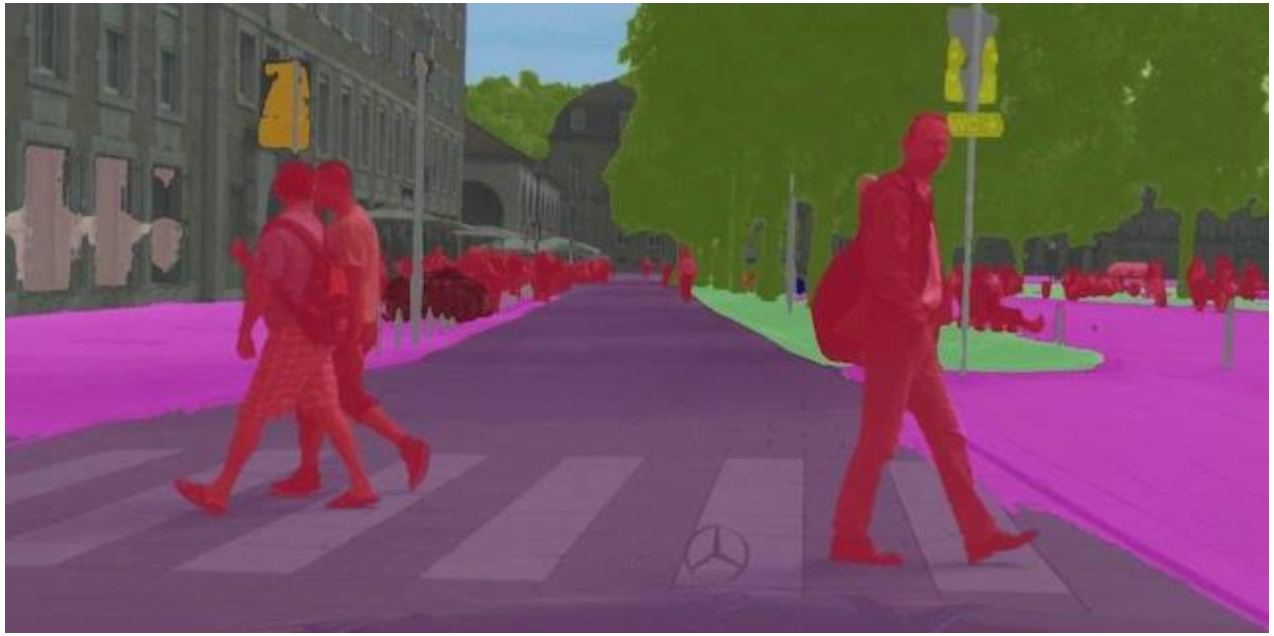
Example



What can we do?

- Segmentation

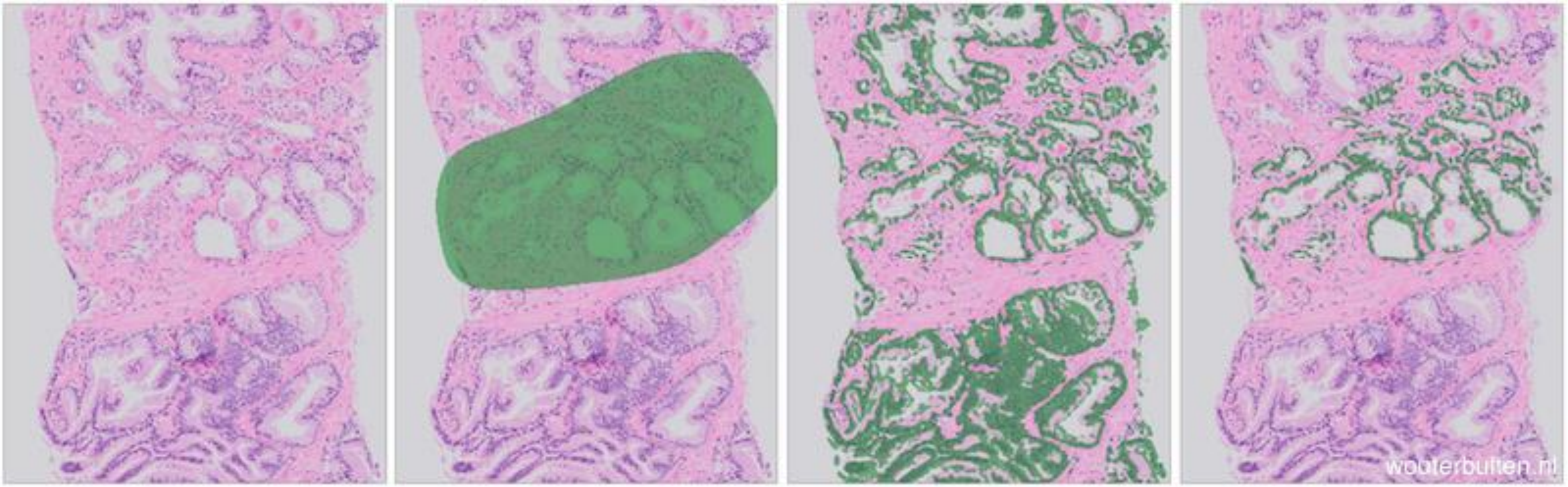
Example



What can we do?

- Segmentation

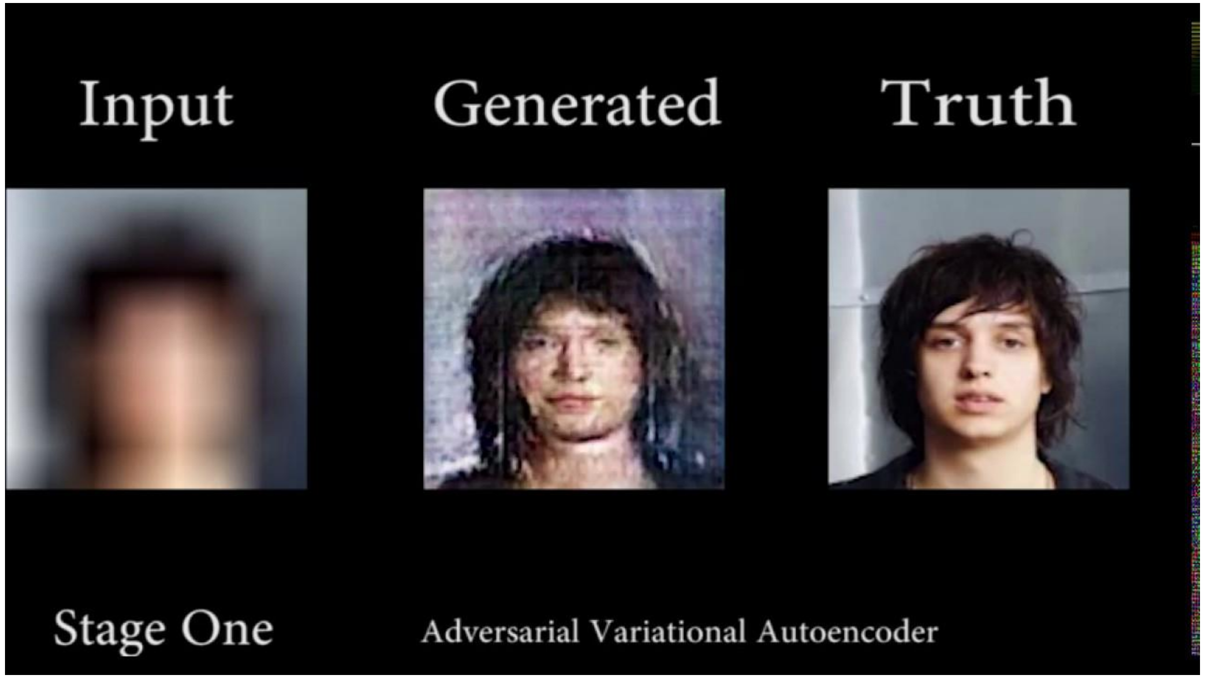
Example



What can we do?

- Image Generation

Example



What can we do?

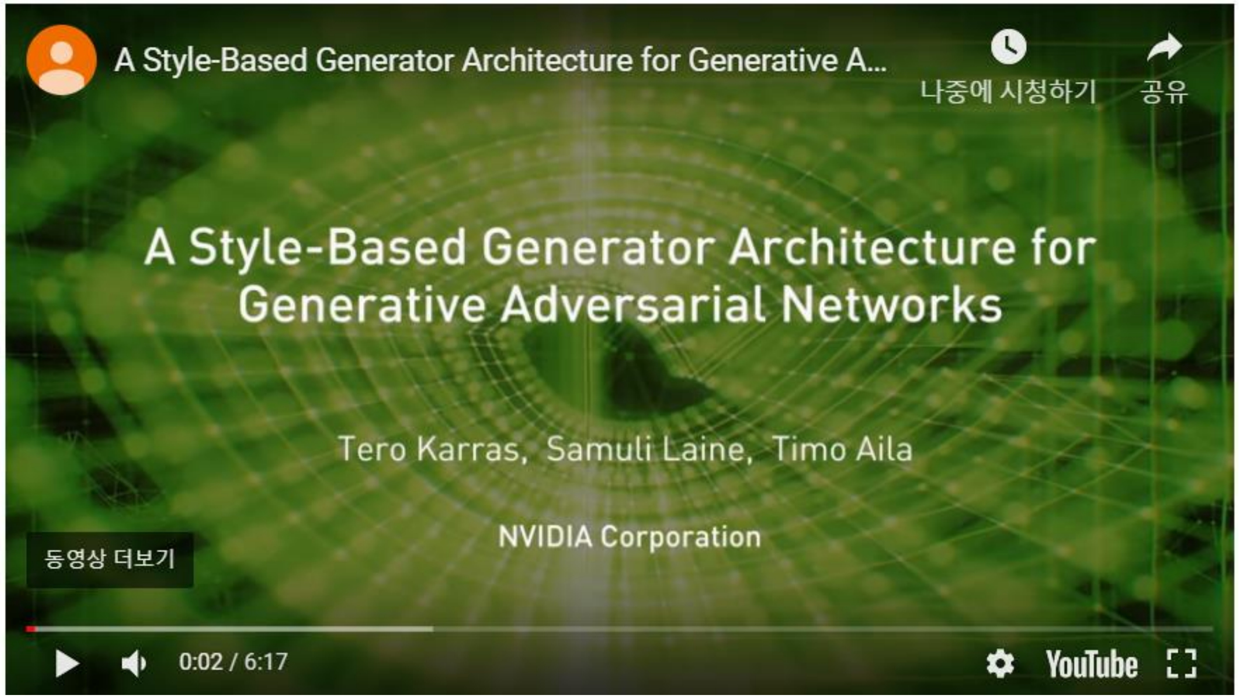
- Image Generation

Example

What can we do?

- Image Generation

Example

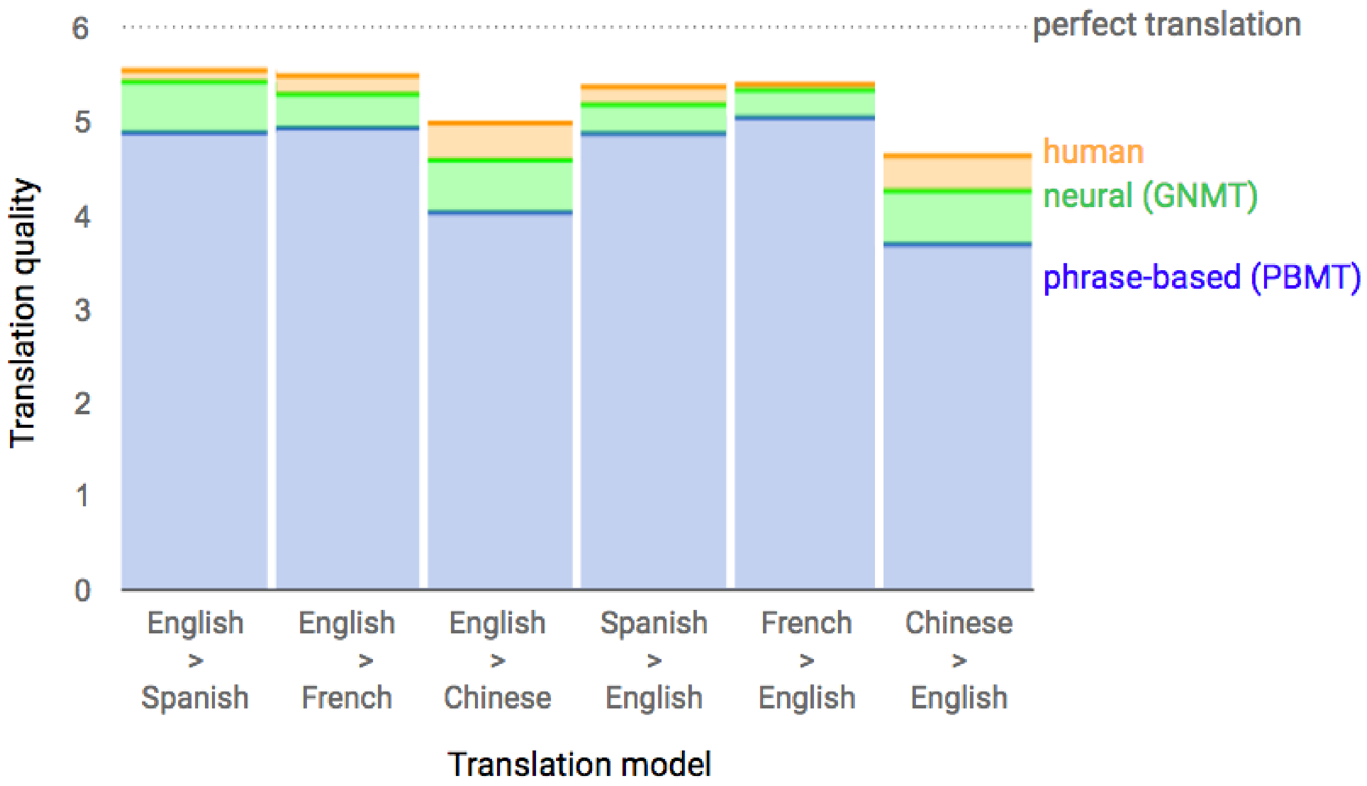


https://youtu.be/kSLJriaOumA?t=92

What can we do?

- Natural Language Processing (NLP)

Example



3. Project Example

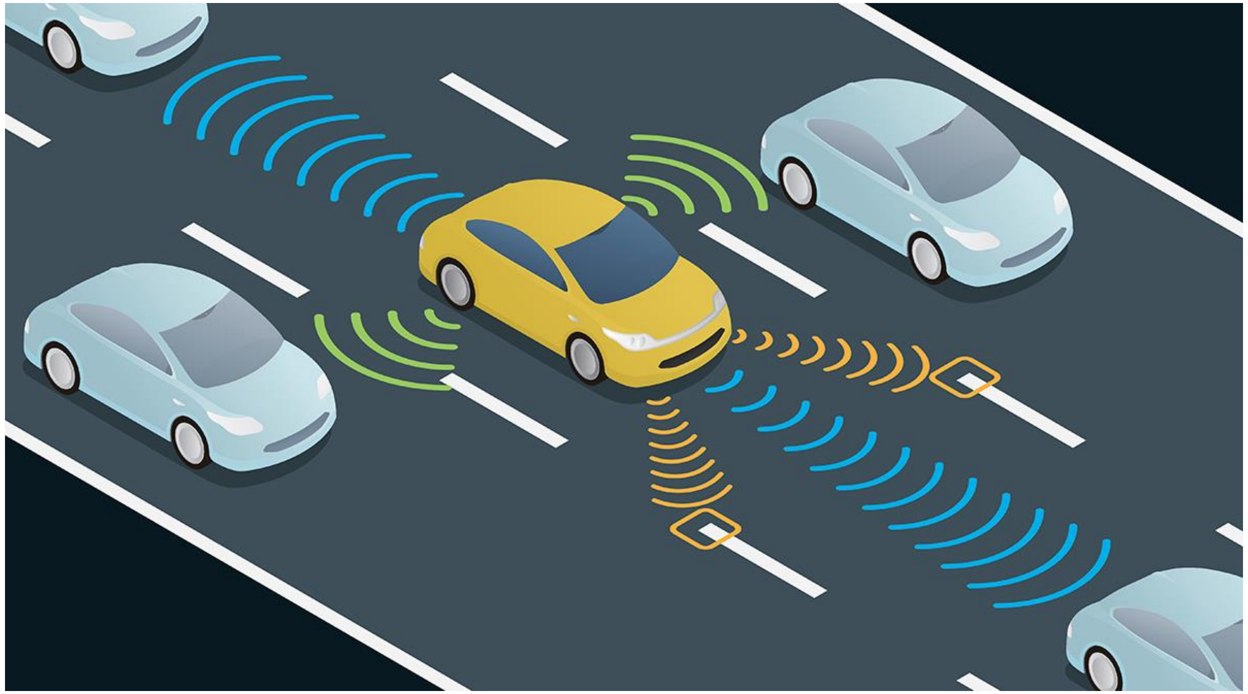
(Self – Driving Car)

Overview

1. 상황인식 (눈)
2. 상황판단 (뇌)
3. 행동결정 (뇌)
4. 행동실행 (손, 발)

http://autotimes.hankyung.com/apps/news.sub_view?nkey=201803091618161

Overview



1. 상황인식 (센서)
2. 상황판단 (알고리즘)
3. 행동결정 (알고리즘)
4. 행동실행 (기계제어)

https://www.ttnews.com/articles/analysis-self-driving-cars-need-standardized-software-avoid-crashes

Overview

1. 상황인식 (센서)

    전자공학 / 기계공학

2. 상황판단 (**알고리즘**)

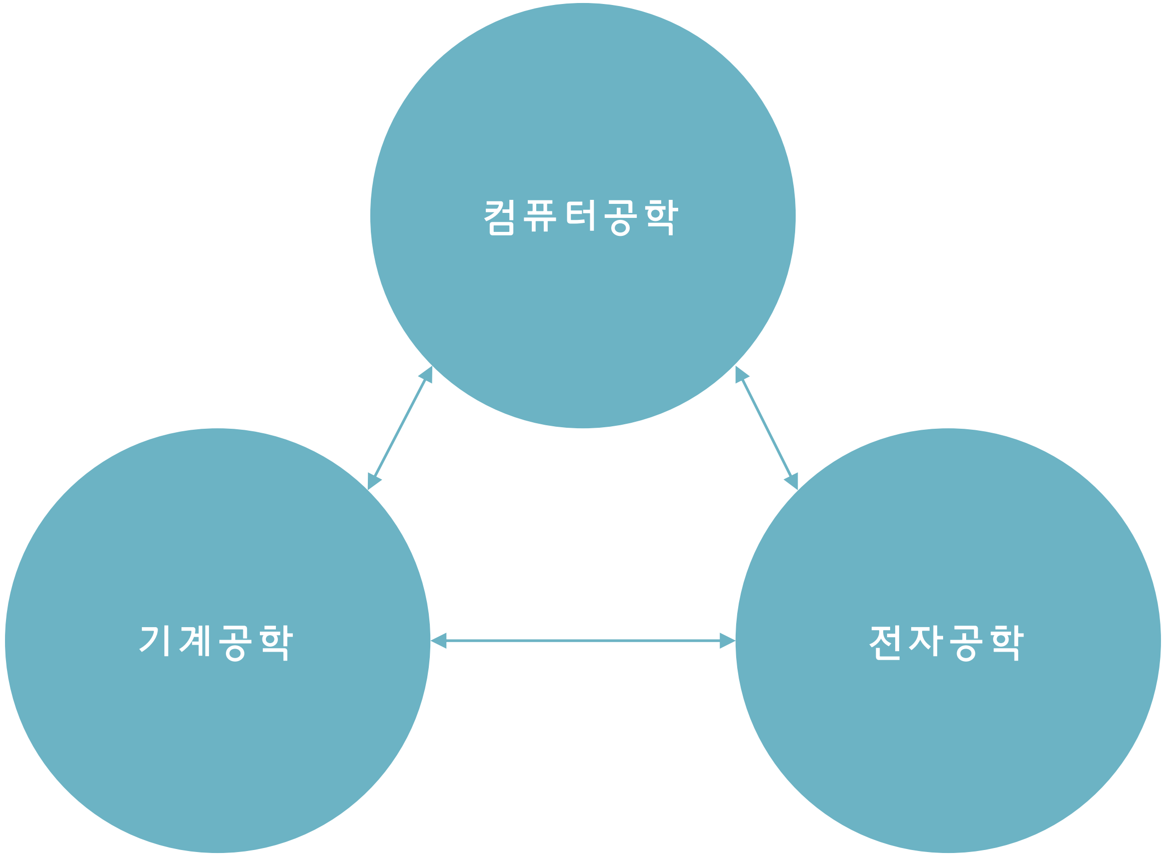
    컴퓨터공학

3. 행동결정 (**알고리즘**)

    컴퓨터공학

4. 행동실행 (기계제어)

    기계공학



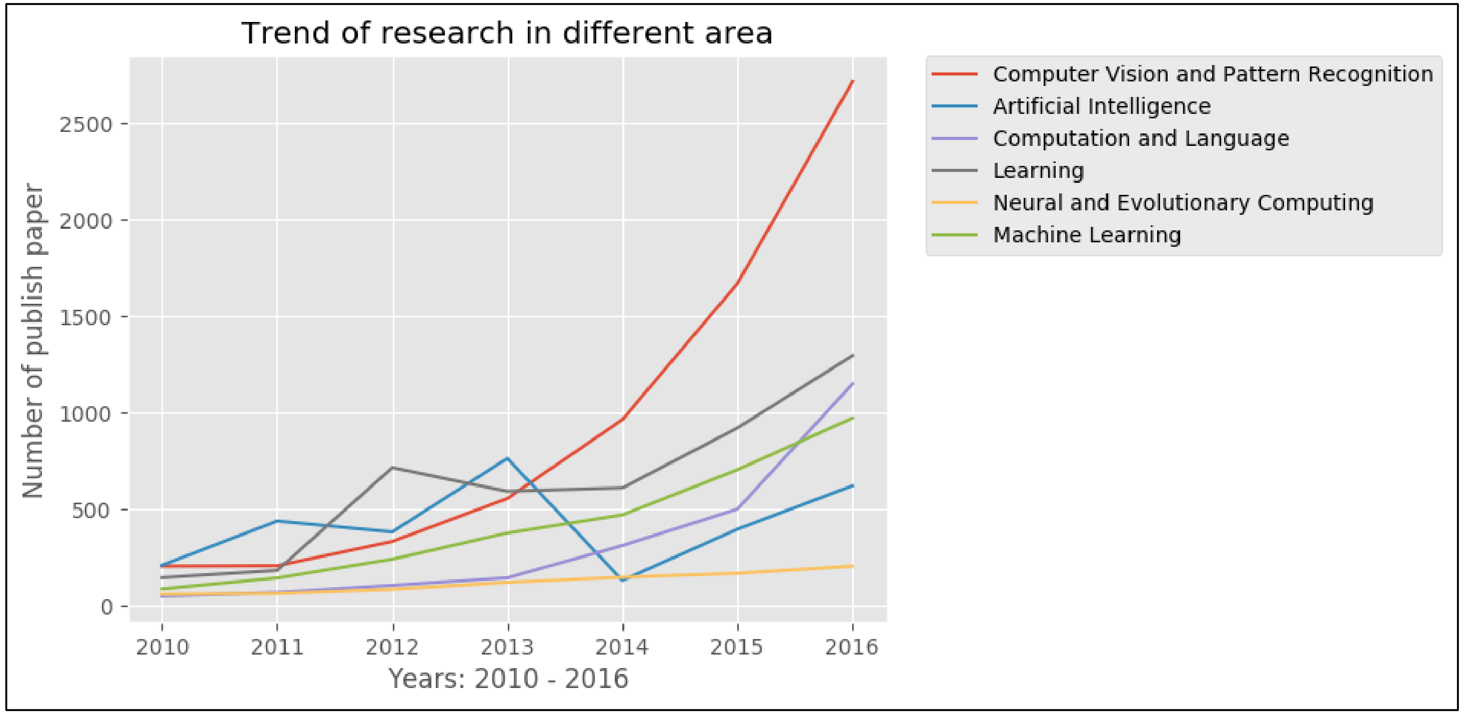
Overview

- 상황판단

    Computer Vision / Artificial Intelligence

- 행동결정

    Artificial Intelligence / Algorithm



https://towardsdatascience.com/getting-started-with-reading-deep-learning-research-papers-the-why-and-the-how-dfd1ac15dbc0

Overview

- 자동주행 자동차 등급

0단계: 비 자율주행

1단계: 특정 기능/ 한 가지 기능만 수행 (ex. 오토크루즈)

2단계: 조합기능 / 두 가지 이상 기능 조합(ex. 장애물보고 정지)

3단계: 조건부 자율 / 일정 구간 자동차가 알아서 운전하되, 운전자는 항시 대기 및 특정상황 대처
→ Google, Uber 제외한 대부분 나머지 기업들 (고속도로 주행가능)

4단계: 고도 자율 / 자동차가 혼자 모든 주행을 담당하되 운전석에 운전자는 탑승해 있어야 함.
→ Google, Uber (국도 주행가능)

5단계: 완전 자율 / 운전자 불필요

Method (1. 상황인식)

- 자율주행에 사용되는 센서

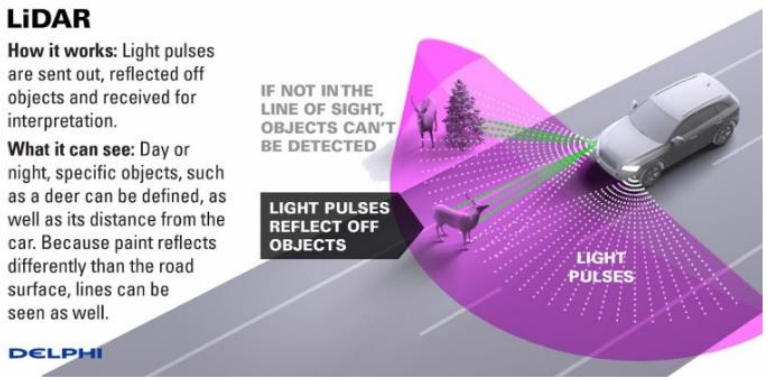
인지		
센서 (차선 및 차량에 대한 정보 인지)	카메라	영상을 통해 도로 주행 환경을 인식 (차선, 신호등 정보 등)
	레이더	1. 사물에 대한 정보는 알 수 없음 2. 밤 혹은 악천후에도 사용이 가능하며, 측정거리가 최소 60m에서 최대 250m까지로 김
	라이더	1. 측정거리가 150m이상이며, 360도 모두 측정 가능하나 비쌈 2. 환경에 영향을 많이 받음
	초음파	1. 주로 차량 후방 사물 인식에 사용되며, 주차 등에 이용됨 2. 값이 싸지만, 15m 내에서 인식 가능
정밀지도 (경로, 차로 선택)	-	1. 50cm이하의 지도 정확성이 필요 → 따라서, 비용 및 시간이 많이 들며, 데이터 축적이 중요함 2. 데이터 축적 단계는 센서로 데이터 수집 → 구분 → 후처리의 과정을 거침 3. 데이터 용량이 매우 큼 → 저장, 활용, 업데이트에 어려움이 큼 4. 구글, HERE, 탐탐이 주요 업체이며, 구글의 경우 3D인지 가능한 라이더를 이용해 HD급 지도 제작 중
측위지도 (차량 위치 파악)		1. 측위 정보를 획득하여 내 차량에 대한 위치를 파악 2. 위치 파악을 위한 주요 기술로 GPS, DR, DGPS가 있으나, 범용적인 GPS의 경우 10m 이상 오차 발생
V2X	-	1. 차량센서로는 감지 못하는 다른 차량 정보(V2V), 사고정보(V2I) 확인 2. 외부 인프라, 차와 통신하는 기술이며 모든 차량에 동일한 통신 기술 적용이 중요함 → 완성 차 업계가 주도하고 있으며, 미국/유럽은 WAVE 기술이 Main이 되어 감

Method (1. 상황인식)

- 라이다(LiDAR) = 눈

센서의 핵심이며, 레이저를 이용하여 물체와 거리와 지형을 측정하며, 회전을 통해 전 방향을 감시한다.
거리를 측정하기 때문에 3D 지형 데이터를 얻을 수 있다. 레이저로 찍기 때문에 빛의 영향을 덜 받는다.

가격이 매우 비싸다. 2017년까지는 구글에서 약 4,000,000km의 주행 테스트를 할 동안 사용된 라이다 가격이 9,000만원이라고 하였다. 2018년부터 구글이 가격을 대폭 절감시켜 900만원까지 줄였다고 발표하였다. 그러나 문제는 센서로 라이다 하나만 쓰는 것이 아니다.



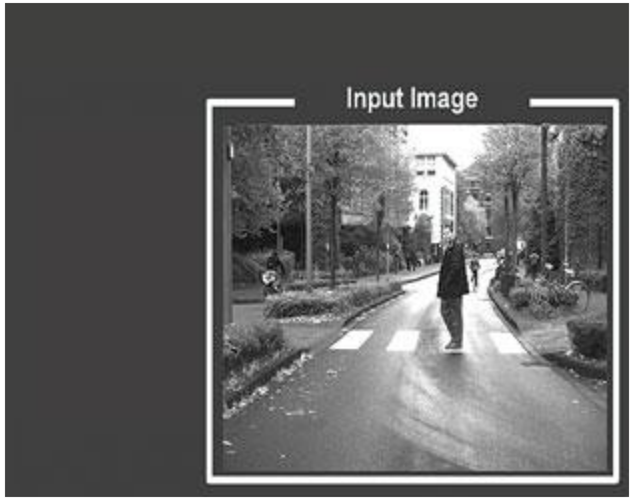
LiDAR로 사물을 관측하는 모습

좌 : Velodyne사의 LiDAR들 / 우 : Velodyne사의 PUCK VLP-16 LiDAR

Method (2. 상황판단)

- 전방사진 및 상황 판단

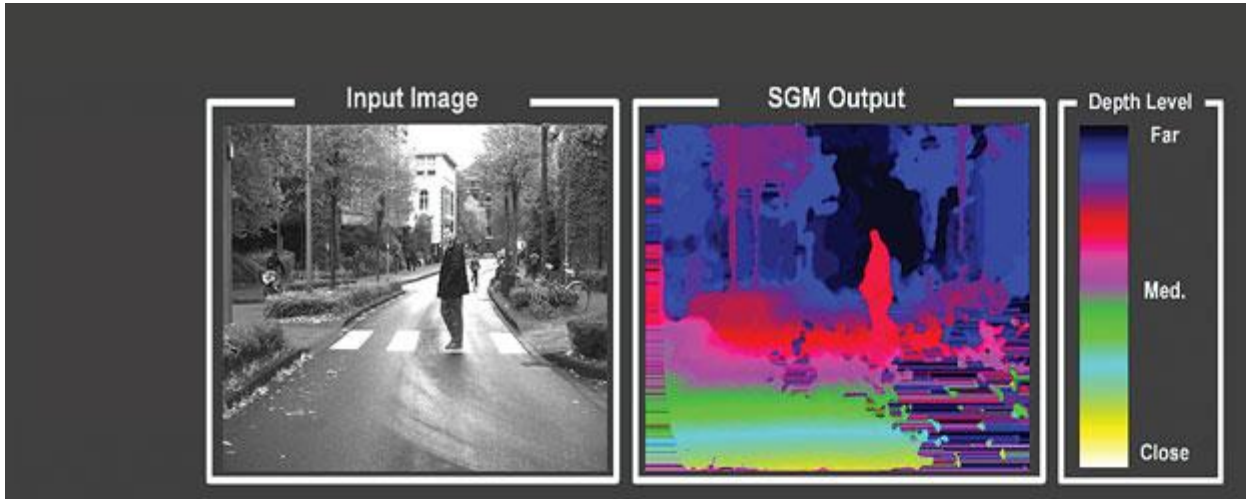
센서를 통해 얻은 이미지



Method (2. 상황판단)

- 전방사진 및 상황 판단

센서를 통해 얻은 이미지

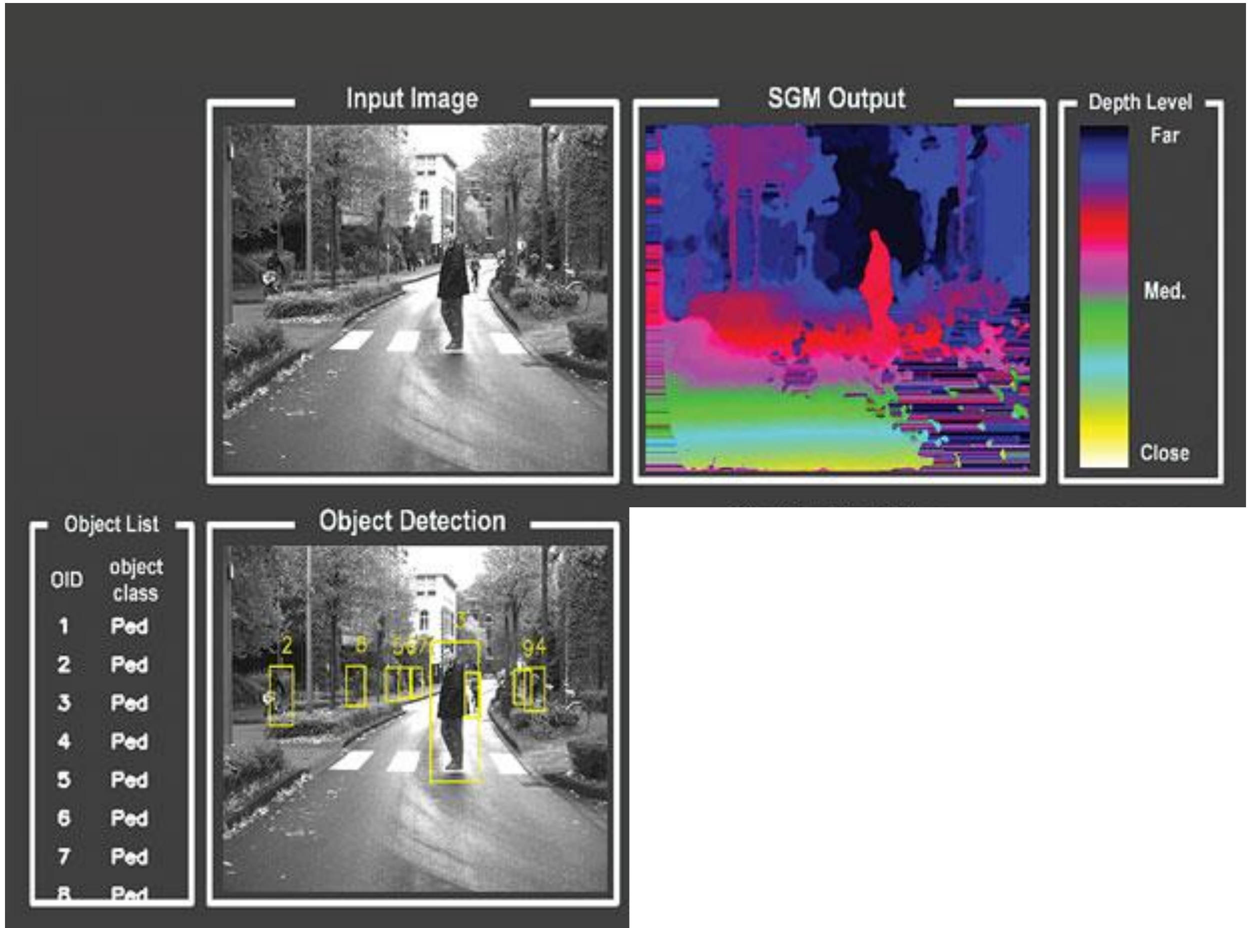


각 픽셀들의 거리 측정

Method (2. 상황판단)

- 전방사진 및 상황 판단

센서를 통해 얻은 이미지



각 픽셀들의 거리 측정

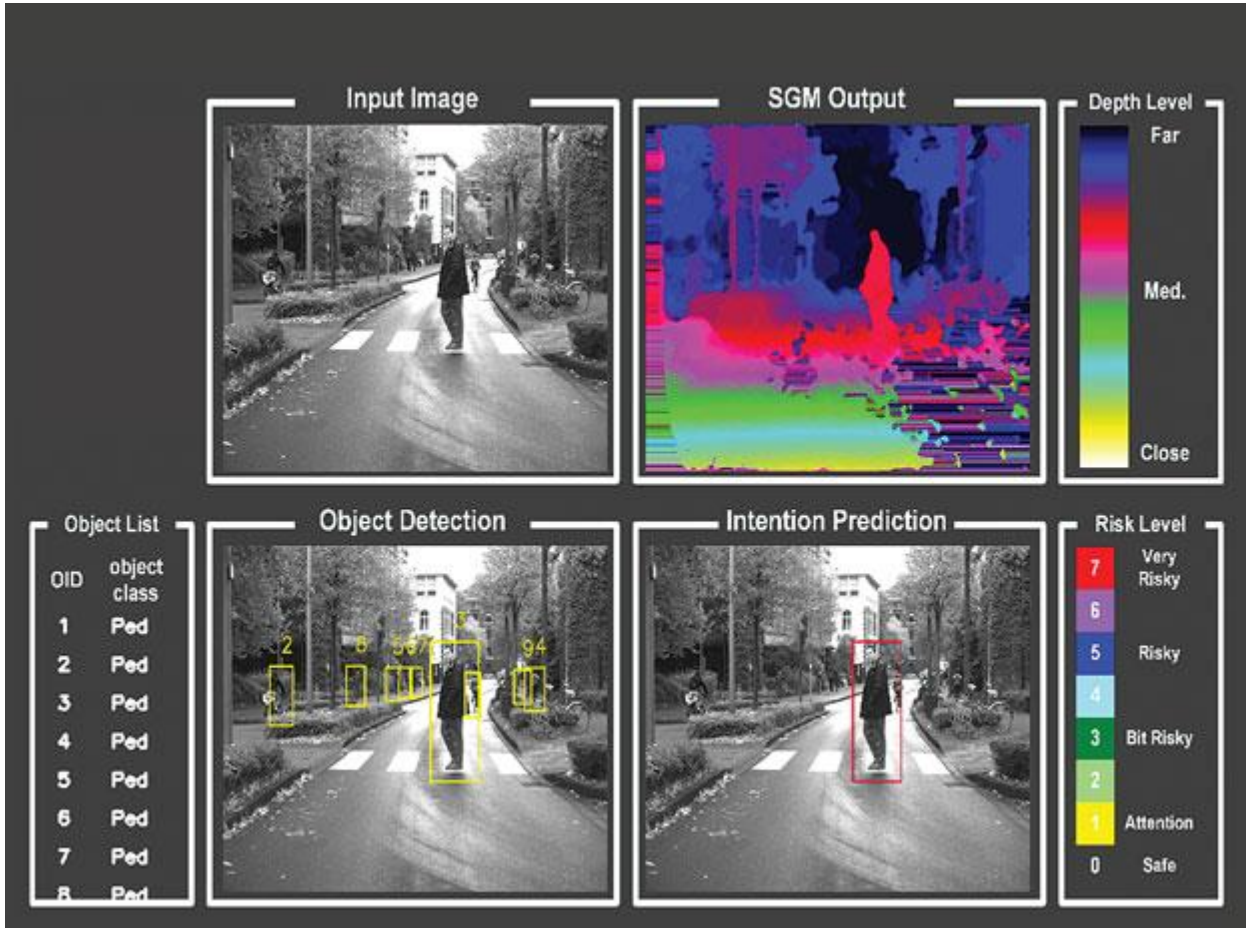
각 물체의 종류 판단

Method (2. 상황판단)

- 전방사진 및 상황 판단

센서를 통해 얻은 이미지

각 픽셀들의 거리 측정



각 물체의 종류 판단

각 물체 별 위험도 판단

Method (2. 상황판단)

- 전방사진 및 상황 판단

HOW ?

Method (2. 상황판단)

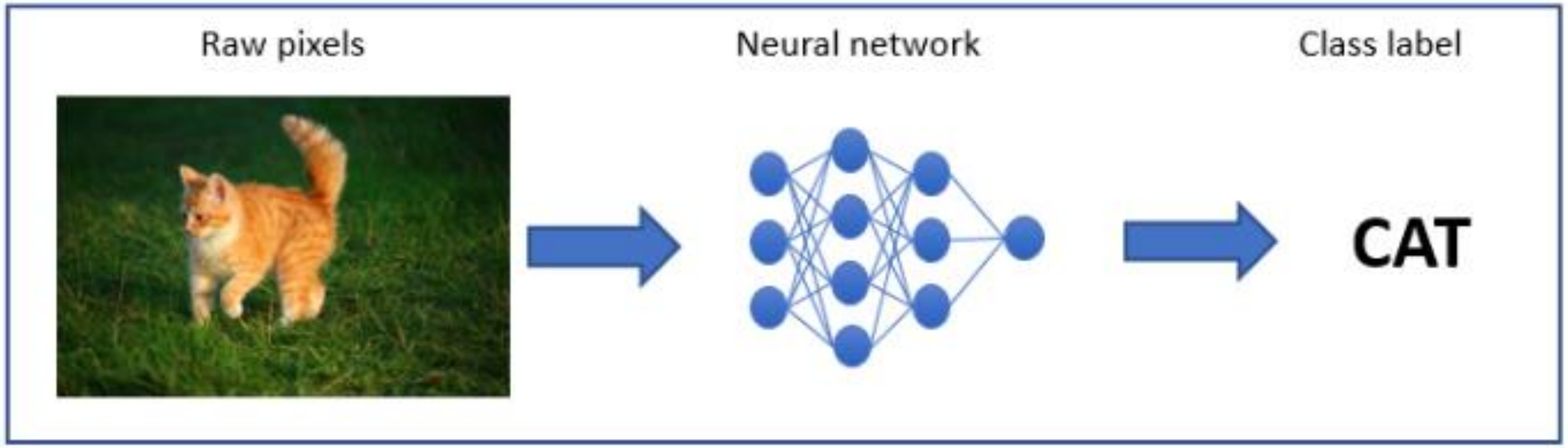
- 전방사진 및 상황 판단

Deep Learning

Method (2. 상황판단)

- Classification

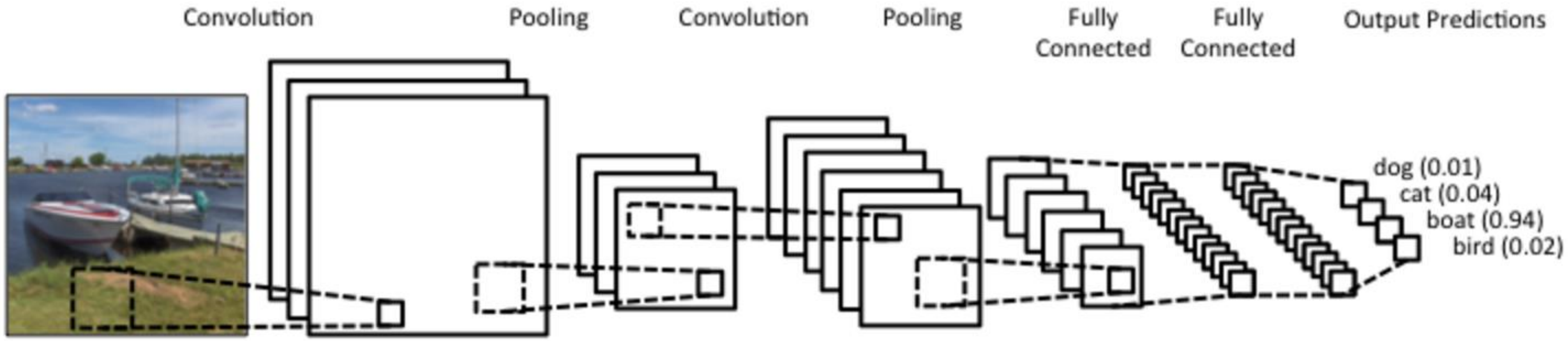
Example



Method (2. 상황판단)

- Classification

Classification Model



Convolutional Neural Network (CNN)

Method (2. 상황판단)

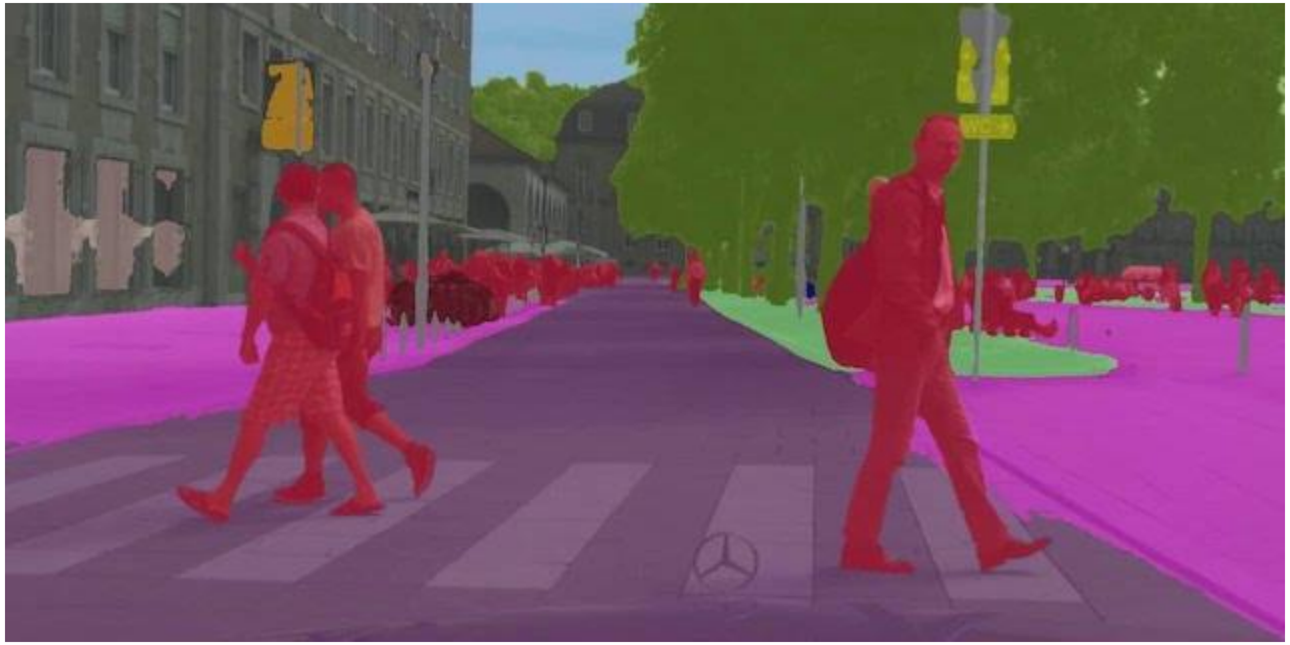
- Segmentation

Example

Method (2. 상황판단)

- Segmentation

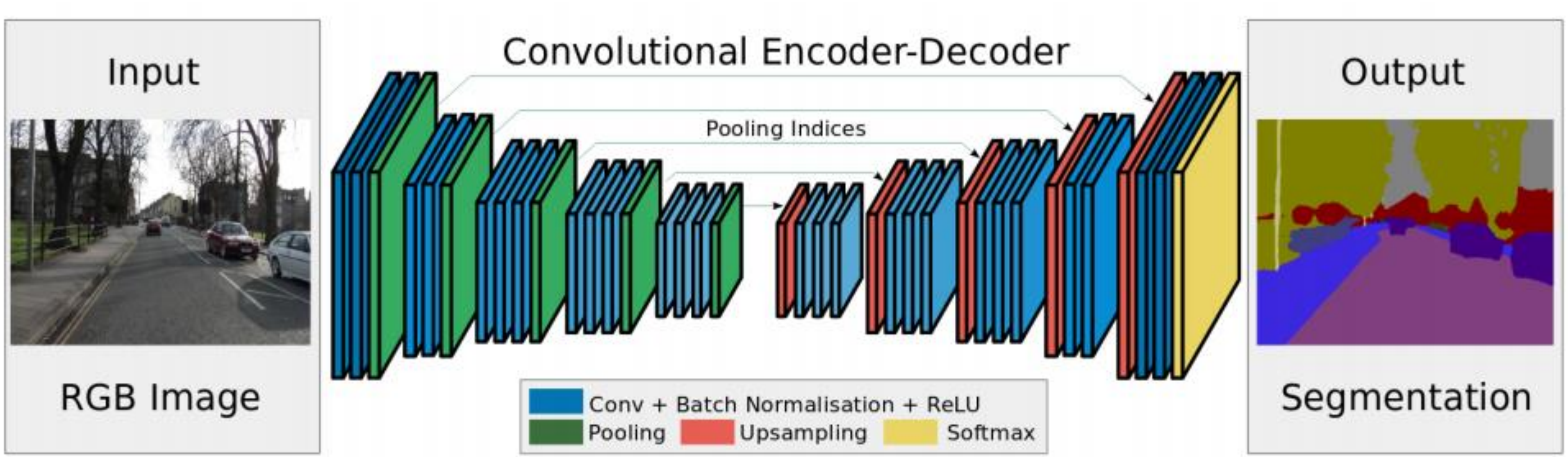
Example



Method (2. 상황판단)

- Segmentation

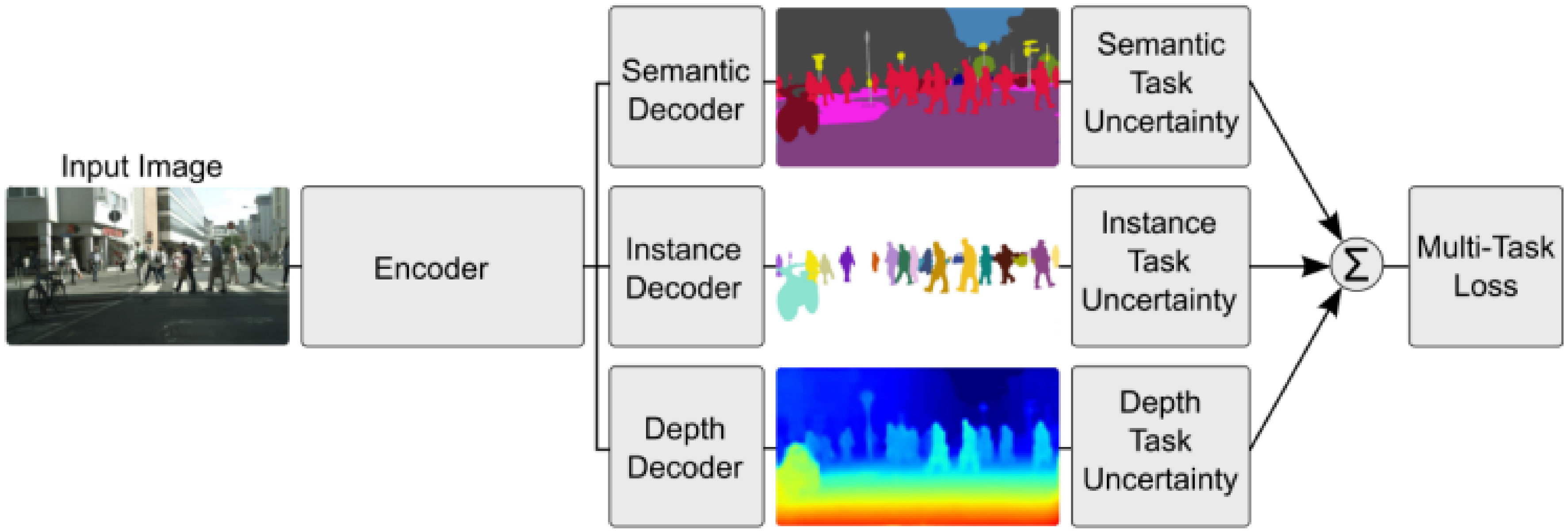
Segmentation Model



Convolutional Neural Network (CNN)

Method (2. 상황판단)

- 상황판단을 위한 종합적인 모델



Project Example

ndrply/self-driving-car (https://github.com/ndrplz/self-driving-car)

Project Example

	Task	Keywords	Language
1.	**Lane Finding**	Computer Vision	Python
2.	**Traffic Sign Classifier**	Deep Learning	Python
3.	**Behavioral Cloning**	Deep Learning	Python
4.	**Advanced lane finiding**	Computer Vision	Python
5.	**Vehicle detection**	Deep Learning, Computer Vision	Python
6.	Extended Kalman Filter	Kalman Filter	C++
7.	Unscented Kalman Filter	Kalman Filter	C++
8.	Kidnapped Vehicle	Particle Filter	C++
9.	PID Control	PID Controller	C++
10.	MPC Control	MPC Controller	C++
11.	Path Planning	Path Planning	C++
12.	**Road Segmentation**	Deep Learning	Python

Project Example

1. Lane Finding

- Summary

  Detected highway lane lines on a video stream. Used OpencV image analysis techniques to identify lines, including **Hough Transforms and Canny edge detection.**

- Main Code

  lane_detection.py

Project Example

1. Lane Finding

- Method

Input: Frame from the frontal / Output: Frame marked on Lines

1. Color Image

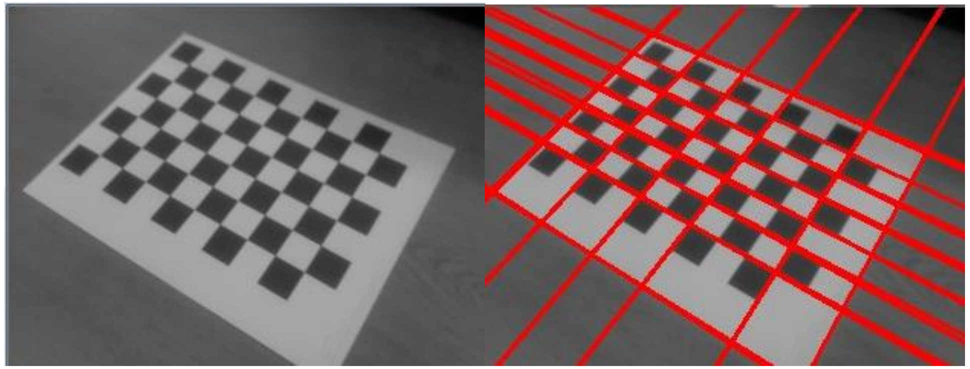
2. Gray Image

3. Blur Image

4. Edge Detection (Canny Edge)

5. Hough Line Detection (Hough Transform)

Canny Edge Detection



Hough Transform

Project Example

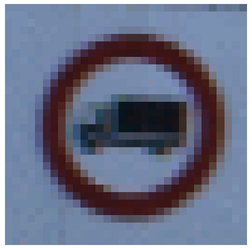
2. Traffic Sign Classifier

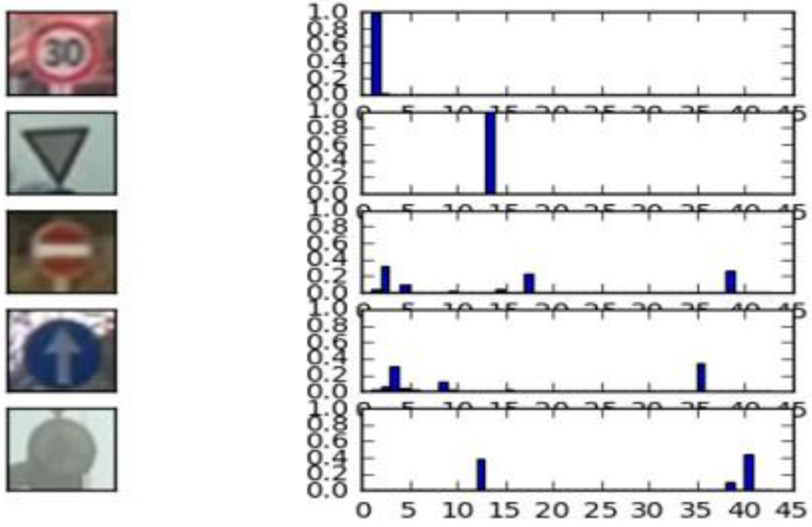
- Summary

Built and trained a **deep neural network (CNN)** to classify traffic signs, using TensorFlow. Experimented with different network architectures. Performed image pre-processing and validation to guard against overfitting.

- Main Code

Traffic_Sign_Classifier.ipynb





Project Example

2. Traffic Sign Classifier

- Method

Input: Traffic Sign Image   */ Output: Traffic Sign Label*

1. Training Data Loading

2. Training Data Augmentation

3. Deep Neural Network Training with Training Data

4. Deep Neural Network Testing with Testing Data

5. Inference with verified Deep Neural Network

Project Example

3. Behavioral Cloning

- Summary

  Built and trained a convolutional neural network (CNN) for end-to-end driving in a simulator, using TensorFlow and Keras. Used optimization techniques such as regularization and dropout to generalize the network for driving on multiple tracks.

- Main Code

  Model.py

Project Example

3. Behavioral Cloning

- Method

Input: Frontal/left/right Frame / Output: Corresponding Steering Direction

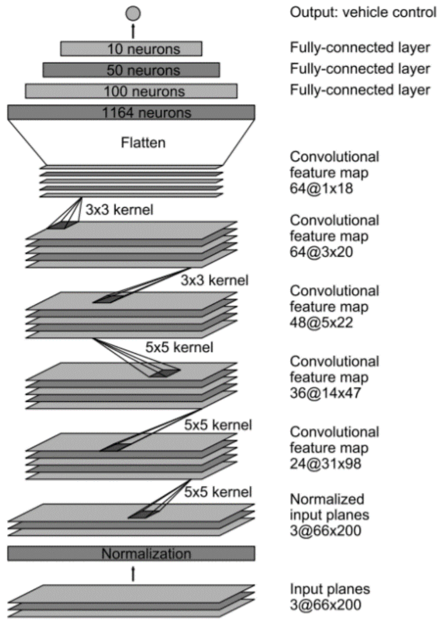
1. Training Data Loading

2. Training Data Augmentation

3. Deep Neural Network Training with Training Data

4. Deep Neural Network Testing with Testing Data

5. Inference with verified Deep Neural Network



Network Architecture

Project Example

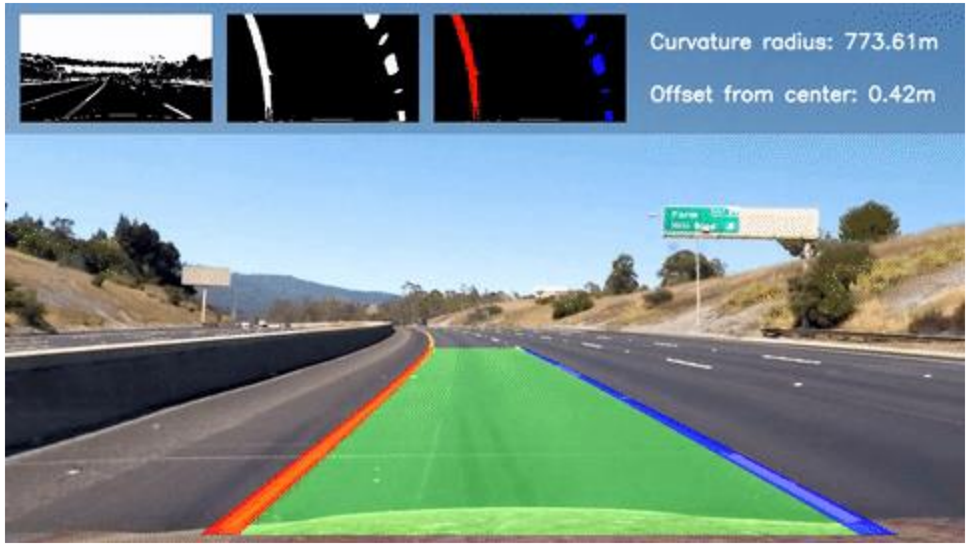
4. Advanced lane finiding

- Summary

Built an advanced lane-finding algorithm using **distortion correction, image rectification, color transforms, and gradient thresholding**. Identified lane curvature and vehicle displacement. Overcame environmental challenges such as shadows and pavement changes.

- Main Code

main.py



Project Example

4. Advanced lane finiding

- Method

Input: Frame from the frontal   / Output: Frame marked on three types of Lines

1. Compute the camera calibration matrix and distortion coefficients given a set of chessboard images.
2. Apply a distortion correction to raw images.
3. Use color transforms, gradients, etc., to create a thresholded binary image.
4. Apply a perspective transform to rectify binary image ("birds-eye view").
5. Detect lane pixels and fit to find the lane boundary.
6. Determine the curvature of the lane and vehicle position with respect to center.
7. Warp the detected lane boundaries back onto the original image.
8. Output visual display of the lane boundaries and numerical estimation of lane curvature and vehicle position.

Project Example

5. Vehicle detection

- Summary

  Created a vehicle detection and tracking pipeline with OpenCV, histogram of oriented gradients (HOG), and support vector machines (SVM). Implemented the same pipeline using a **deep network** to perform detection. Optimized and evaluated the model on video data from a automotive camera taken during highway driving.

- Main Code

  SSD.py

- Recommandation Model

  Mask R-CNN (light ver.)

Project Example

5. Vehicle detection

- Method

Input: Frontal Frame / Output: Frontal Frame marked on cars

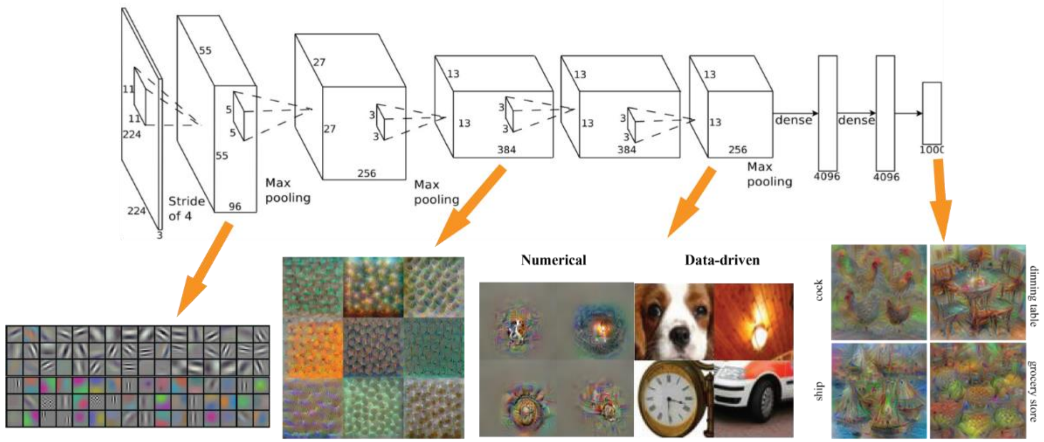
1. Training Data Loading

2. Training Data Augmentation

3. Deep Neural Network Training with Training Data

4. Deep Neural Network Testing with Testing Data

5. Inference with verified Deep Neural Network



Conv 1: Edge+Blob
Conv 3: Texture
Conv 5: Object Parts
Fc8: Object Classes

Project Example

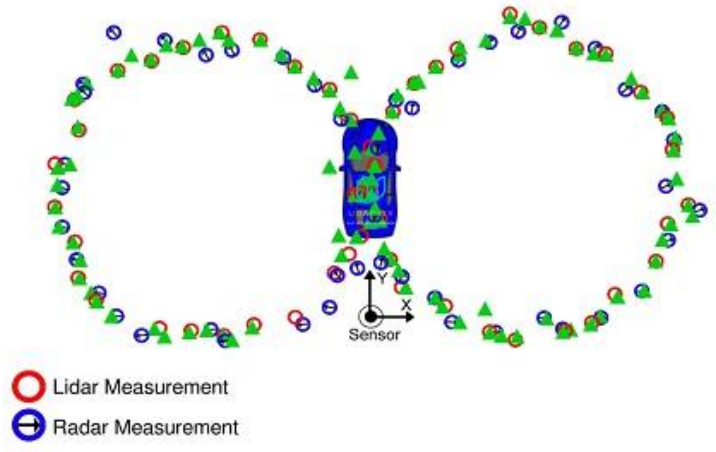
6. Extended Kalman Filter

Summary

Implement the extended **Kalman filter** in C++. Simulated lidar and. radar measurements are used to detect a bicycle that travels around your vehicle. Kalman filter, lidar measurements and radar measurements are used to track the bicycle's position and velocity

Kalman Filter

오차를 가지는 관측치로부터 시스템의 상태를 추정하거나 제어하기 위한 알고리즘

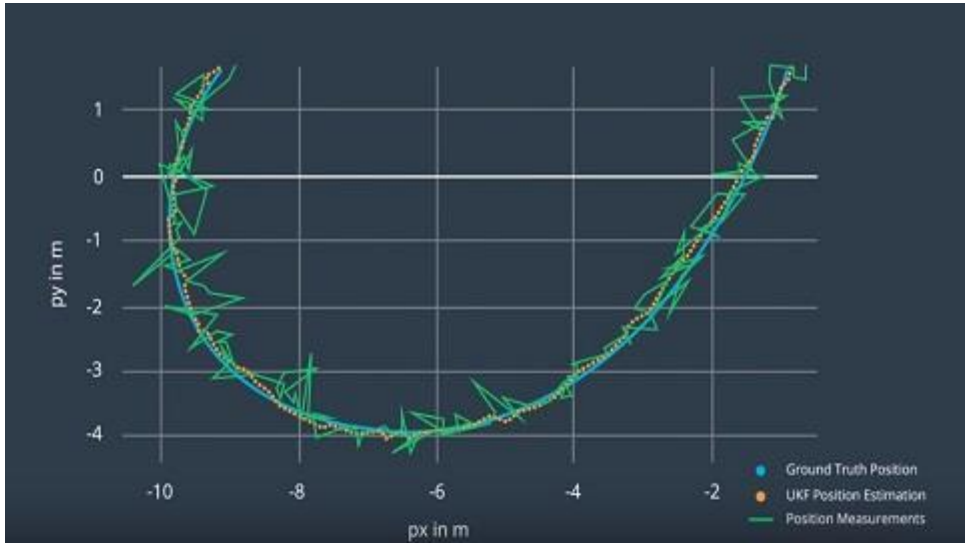


Project Example

7. Unscented Kalman Filter

Summary

Utilize an **Unscented Kalman Filter** to estimate the state of a moving object of interest with noisy lidar and radar measurements. Kalman filter, lidar measurements and radar measurements are used to track the bicycle's position and velocity.



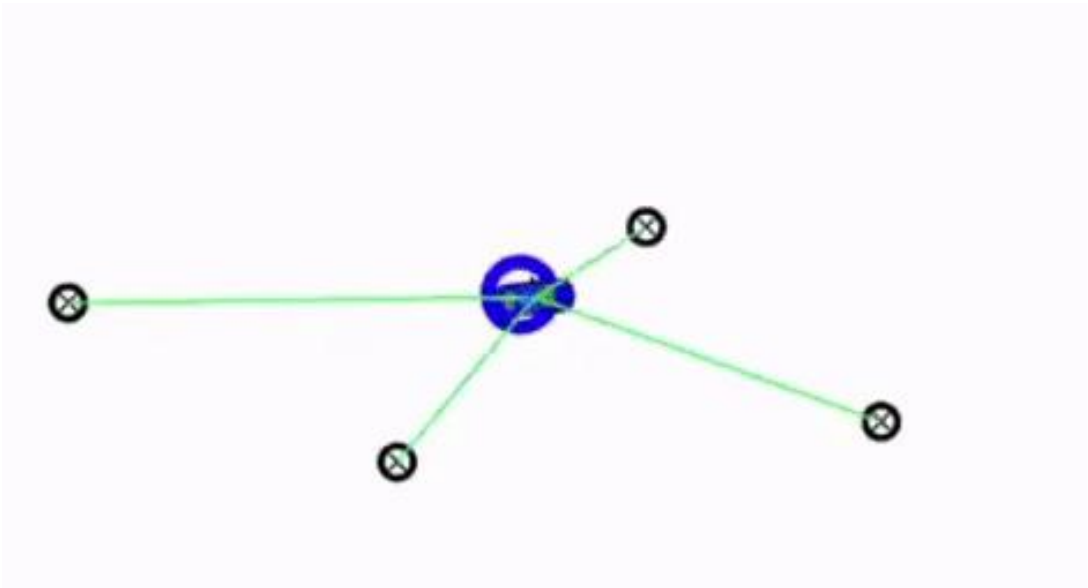
Project Example

8. Kidnappped Vehicle

- Summary

Your robot has been kidnapped and transported to a new location! Luckily it has a map of this location, a (noisy) GPS estimate of its initial location, and lots of (noisy) sensor and control data. In this project you will implement a 2 dimensional particle filter in C++. Your particle filter will be given a map and some initial localization information (analogous to what a GPS would provide). At each time step your filter will also get observation and control data.

(현위치에서의 최소 거리를 계산하여 목적지 도달 / GPS)



Project Example

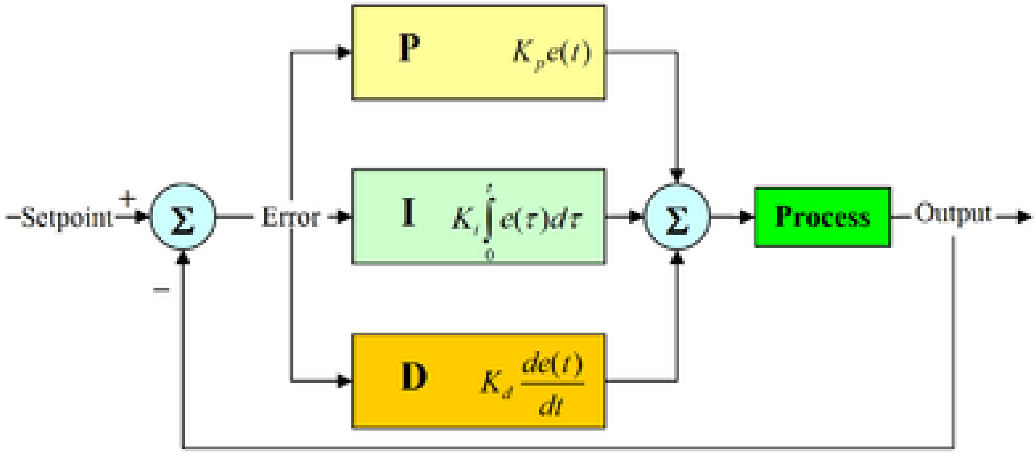
9. PID Control

- Summary

Implement a PID controller for keeping the car on track by appropriately adjusting the steering angle.

- PID Controller (proportional-integral-derivative controller) – 제어공학 이론

PID Controller란 피드백 제어기의 형태 중 하나로 제어하고자 하는 대상의 출력을 측정하여 이를 원하고자 하는 참조 값으로 설정하여 오차를 계산하고 오차를 반영하여 최종 결정을 하는 시스템 구조



PID 제어기

Project Example

9. PID Control

Project Example

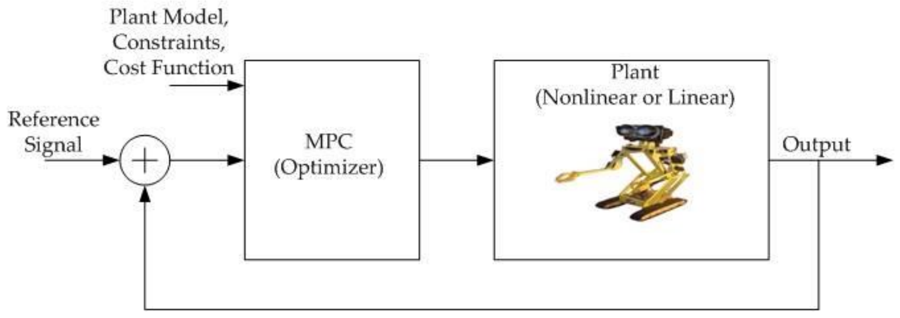
10. MPC Control

- Summary

Implement an **MPC controller** for keeping the car on track by appropriately adjusting the steering angle. Differently from previously implemented PID controller, MPC controller has the ability to anticipate future events and can take control actions accordingly. Indeed, future time steps are taking into account while optimizing current time slot.

- Model Predictive Control (MPC)

모델예측제어(MPC)에서는 제어대상의 모델을 이용해 상태변수나 출력을 예측하고, 이를 바탕으로 적절한 비용함수와 제약조건을 이용해 최적화를 실행한다. 최적화 과정에서 입력이나 상태변수가 부등식 형태의 제약조건을 만족하도록 할 수 있다. 예를들면 물리적으로 음수가 될 수 없는 제어입력이 실제로 그렇게 결정되도록 한다든가, 중요한 공정의 특정 부분 온도가 어떤 범위 안에 있도록 한다든가 하는 게 가능하다.



Project Example

10. MPC Control

Project Example

11. Path Planning

- Summary

The goal in this project is to build a **path planner** that is able to create smooth, safe trajectories for the car to follow. The highway track has other vehicles, all going different speeds, but approximately obeying the 50 MPH speed limit. The car transmits its location, along with its sensor fusion data, which estimates the location of all the vehicles on the same side of the road.

Project Example

12. Road Segmentation

- Summary

Implement the road segmentation using a **fully-convolutional network.**

- Segmentation

Segmentation [segmen'teɪʃn] 분할, 분할된 부분

Project Example

12. Road Segmentation

- Method

Input: Frontal Frame / Output: Frontal Frame marked on Road

1. Training Data Loading

2. Training Data Augmentation

3. Deep Neural Network Training with Training Data
   Deep Neural Network validation with validationing Data

4. Deep Neural Network Testing with Testing Data
5. Inference with verified Deep Neural Network