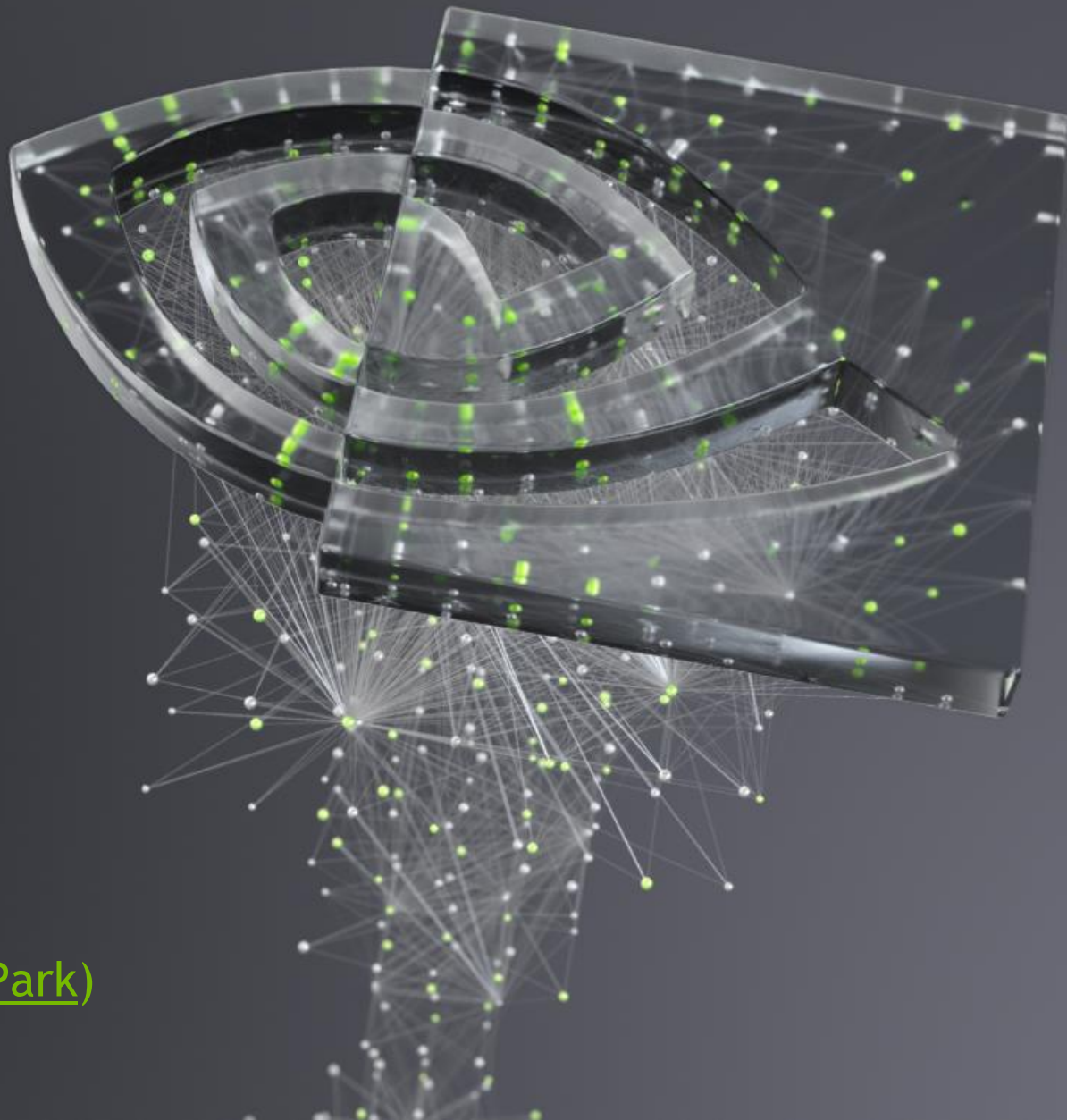




DEEP  
LEARNING  
INSTITUTE

# Introduction to the DeepStream SDK

Building a DeepStream Application  
(Material and presentation: [Jeiyoon Park](#))



# 오늘 목표

우와...



1. 딥러닝 기초 지식, 딥러닝이 어려운 이유
2. DeepStream SDK에 대한 설명
3. DeepStream을 활용한 Video Object Detection 간단한 예제

# 목차 – I 부: Deep Learning

- AI의 역사
- 딥러닝이란 무엇인가
- 딥러닝 혁명
- 딥러닝이 어떻게 세상을 바꾸는가
- 딥러닝이 어려운 이유
- DeepStream

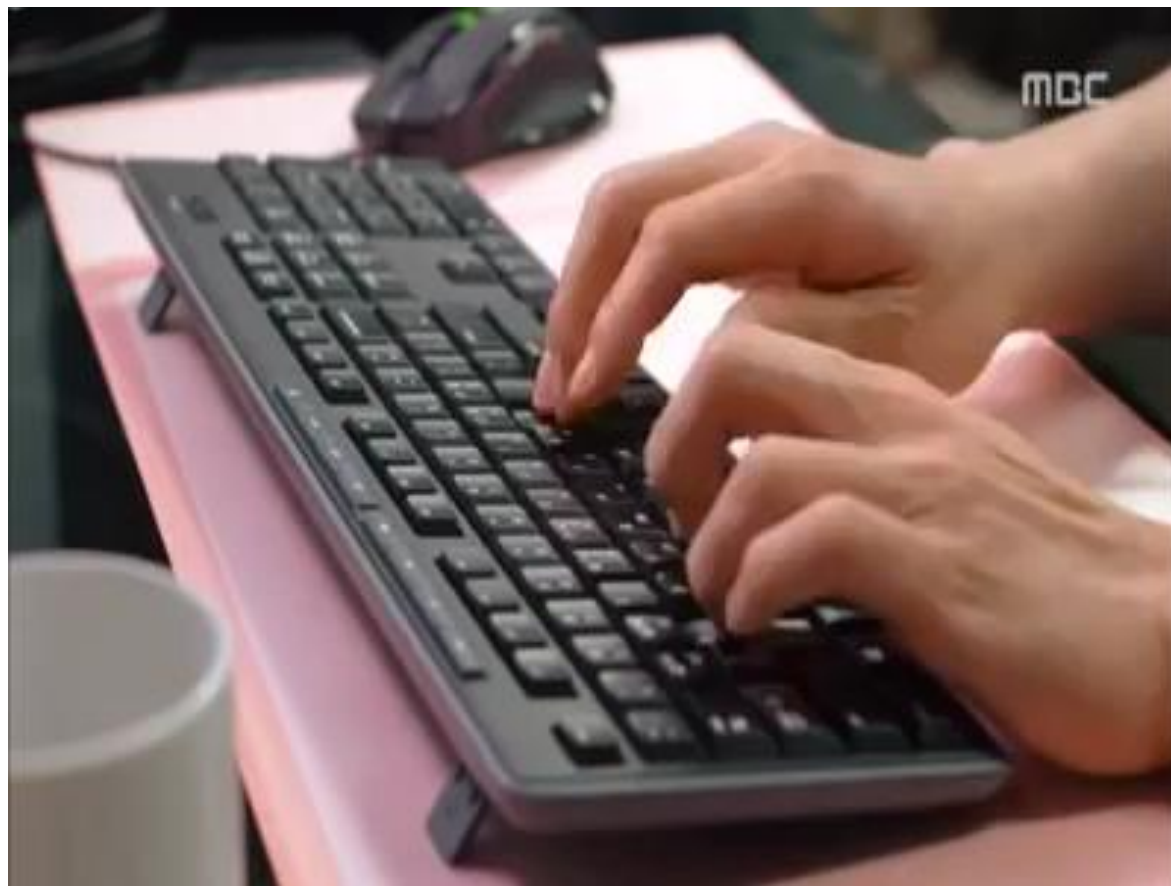
# 목차 – I 부: Deep Learning

- AI의 역사
- 딥러닝이란 무엇인가
- 딥러닝 혁명
- 딥러닝이 어떻게 세상을 바꾸는가
- 딥러닝이 어려운 이유
- DeepStream



# AI의 역사

# 전문가 시스템 (a.k.a. 프로그램)





# 전문가 시스템 (a.k.a. 프로그램)

## 1. 매우 복잡함

(e.g. 구구단 출력 함수)

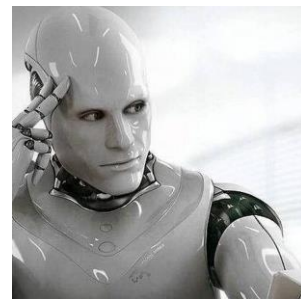
```
174 //구구단 출력
175 public class GuguClass {
176     public static void main(String[] args) {
177         System.out.format("%n");
178         for (int j = 2; j <= 9; j++) {
179             System.out.println();
180             for (int i = 2; i <= 9; i++) { // 2*3
181                 System.out.format("%d X %d = %d\t", i, j, i * j);
182             }
183             System.out.println(); // 줄바꿈
184             for (int j = 2; j <= 9; j++) {
185                 System.out.print("\t");
186             }
187         }
188     }
189 }
```

## 2. 수백 명의 엔지니어에 의해 프로그래밍됨



## 3. 다수의 규칙으로 이루어진 까다로운 프로그래밍

```
174 //구구단 출력
175 public class GuguClass {
176     public static void main(String[] args) {
177         System.out.format("%n");
178         for (int j = 2; j <= 9; j++) {
179             System.out.println();
180             for (int i = 2; i <= 9; i++) { // 2*3
181                 System.out.format("%d X %d = %d\t", i, j, i * j);
182             }
183             System.out.println(); // 줄바꿈
184             for (int j = 2; j <= 9; j++) {
185                 System.out.print("\t");
186             }
187         }
188     }
189 }
```



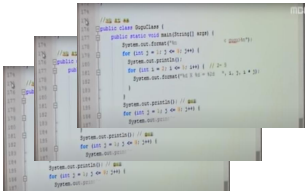
# 전문가 시스템 - 한계

Q) 이 세 개의 이미지는 무엇인가요?





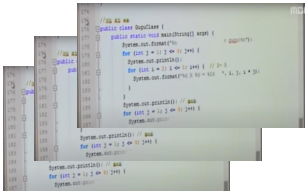
# 전문가 시스템 - 한계



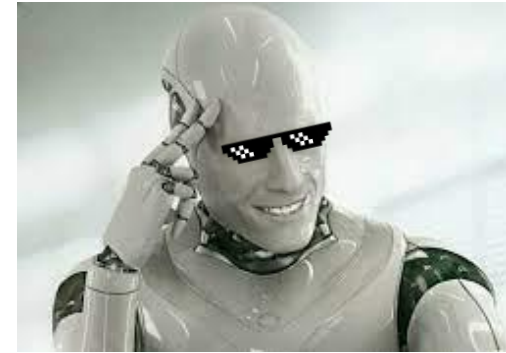
입력 데이터



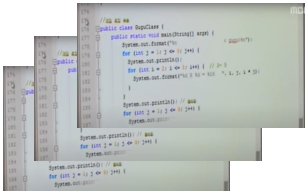
# 전문가 시스템 - 한계



입력 데이터

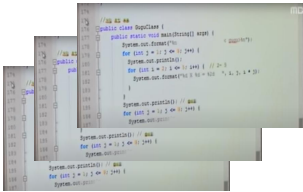


# 전문가 시스템 - 한계



다른 데이터 (이미지 크기, 배경, 대상 등)

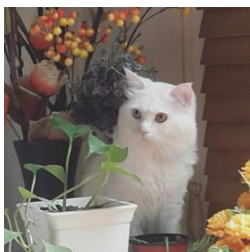
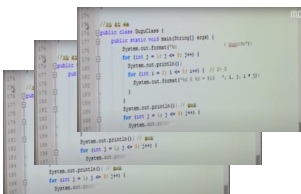
# 전문가 시스템 - 한계



다른 데이터 (이미지 크기, 배경, 대상 등)



# 전문가 시스템 - 한계



→ “규칙에 없는 새로운 입력도 잘 분류할 수 있는 방법은 없을까?”

다른 데이터 (이미지 크기, 배경, 대상 등)

# 아이는 어떻게 학습할까요?



- 다량의 데이터에 노출
- 어린이에게 '정답' 제공
- 스스로 중요한 패턴 습득

→ “아이들 처럼 규칙을 정해주지 않아도 스스로 학습하면서 배우는 프로그램은 없을까?”



# 인공 지능의 시작



- 인간은 컴퓨터가 발명된 이후로 작업을 수행하도록 가르쳐왔음
- 컴퓨터의 목적은 인간이 원하는 작업을 인간의 능력 이상을 수행하는 것



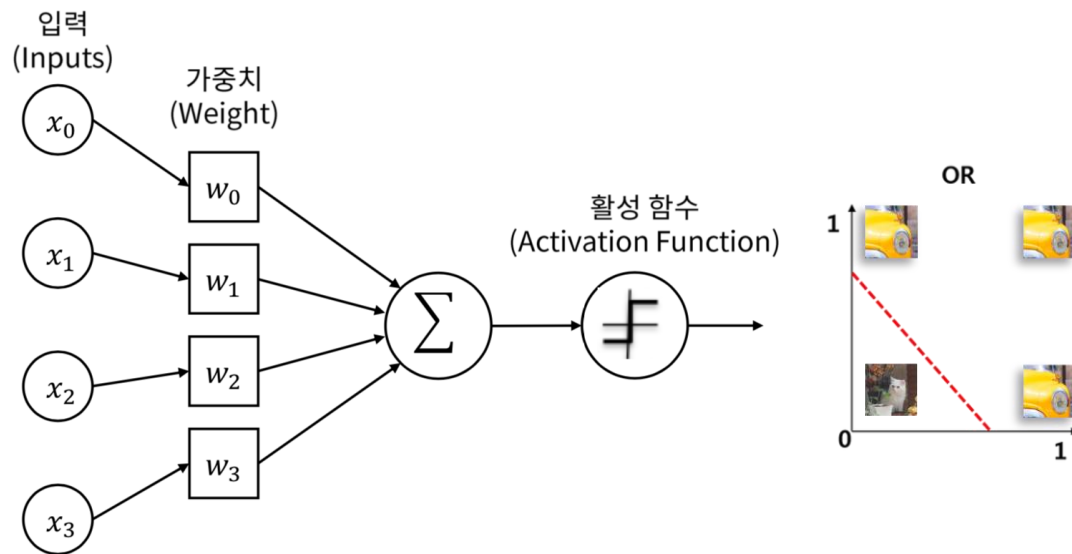
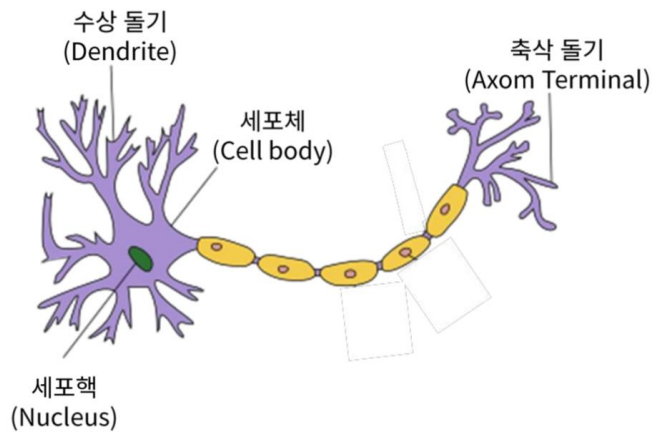
- 컴퓨터가 처음 개발되었을 당시에는, 컴퓨터가 몇 십 년 안에 인간 수준의 지능을 달성할 거라고 믿는 이들이 많았음



- 하지만 결과적으로는 당시의 컴퓨터가 일반화된 인간 지능에 도달하기에는 무리가 있었음

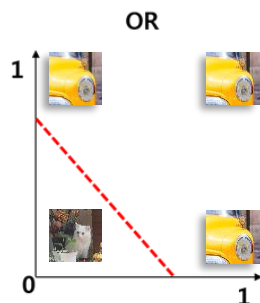
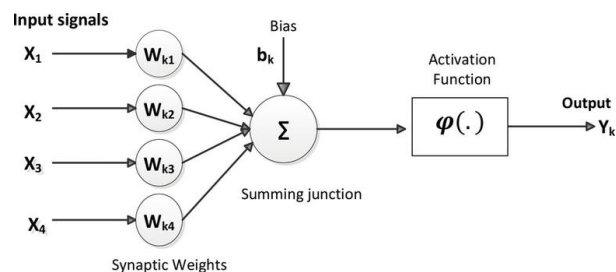
# 인공 지능의 시작

## - 신경 세포 → 인공신경망

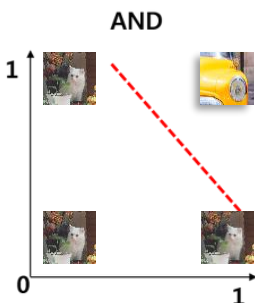


# 인공 지능의 시작...?

e.g.) XOR 문제



$x_1$	$x_2$	$y$
0	0	0
0	1	1
1	0	1
1	1	1



$x_1$	$x_2$	$y$
0	0	0
0	1	0
1	0	0
1	1	1



$x_1$	$x_2$	$y$
0	0	0
0	1	1
1	0	1
1	1	0

과학자들이 예상한 인공지능



“클래스 100개? 거뜰하지”

실제 인공지능



“이거 XOR이야...?”  
“나 죽어버릴거야...”

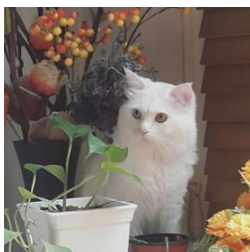
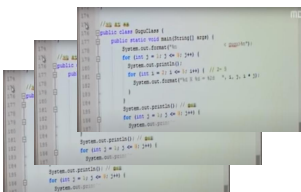
# 목차 – I 부: Deep Learning

- AI의 역사
- 딥러닝이란 무엇인가
- 딥러닝 혁명
- 딥러닝이 어떻게 세상을 바꾸는가
- 딥러닝이 어려운 이유
- DeepStream



딥러닝이란 무엇인가?

# 기존 프로그래밍: 분류기 구축



입력 데이터



- 1) 분류를 위한 규칙 세트 정의
- 2) 규칙을 컴퓨터에 프로그래밍
- 3) 예시가 주어지면 프로그램이 규칙을 통한 분류 수행



# 딥러닝을 선택해야 하는 경우

## 기존 프로그래밍

규칙이 명확하고,  
직관적인 경우  
→ 단순히 프로그래밍

e.g.) 구구단 출력

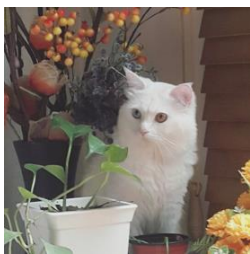
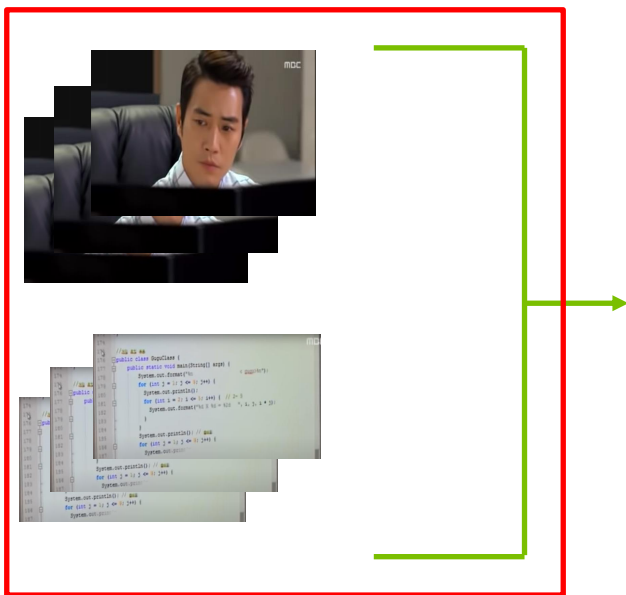
## 딥러닝

규칙이 복잡하며  
식별하기 어려운 경우  
→ 딥러닝 사용

e.g.) 강아지 vs 빵

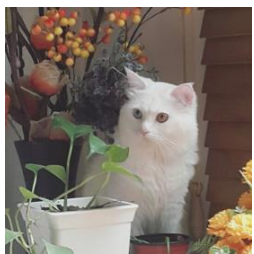
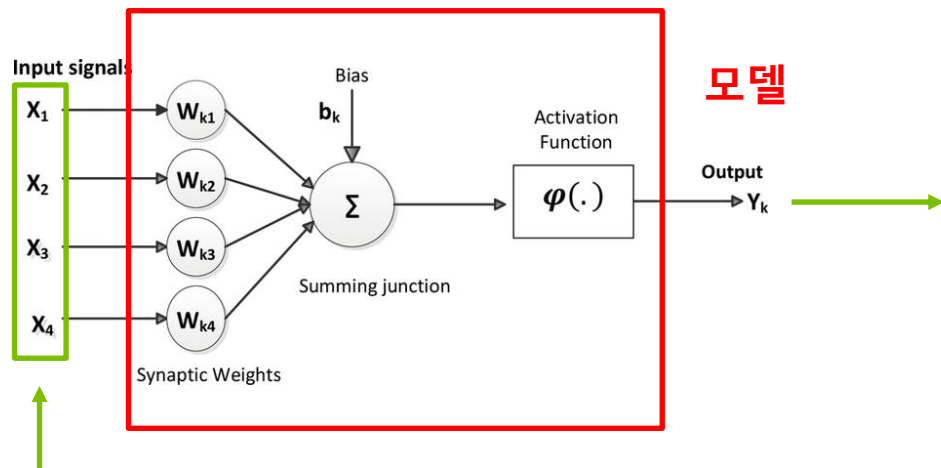


# 기존 프로그래밍 → 딥러닝



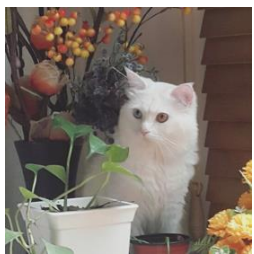
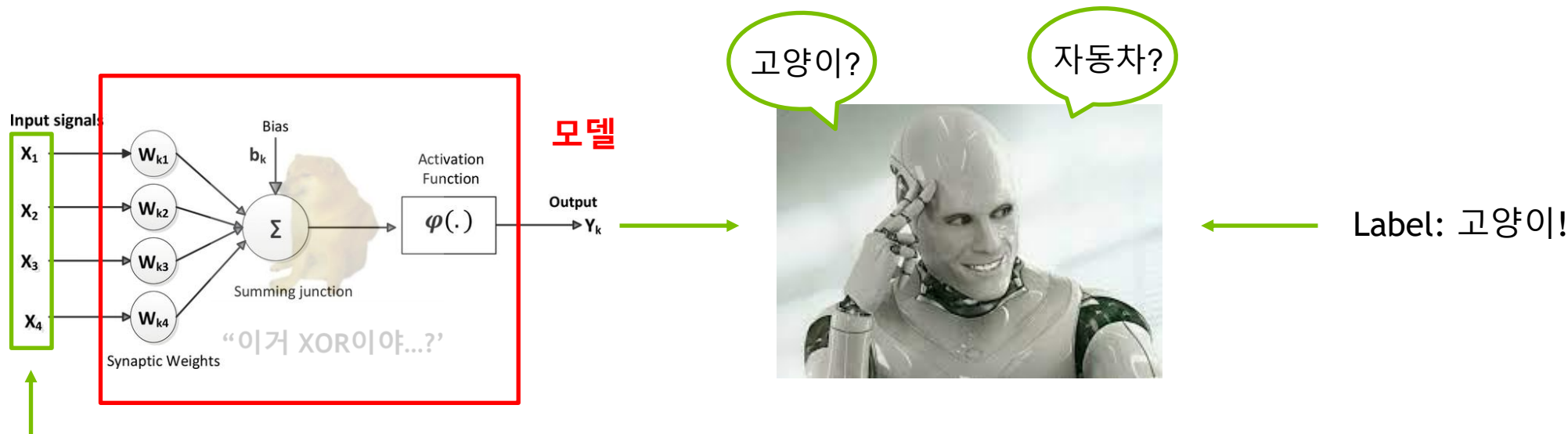
입력 데이터

# 기존 프로그래밍 → 딥러닝



입력 데이터

# 딥러닝 (Deep Learning)

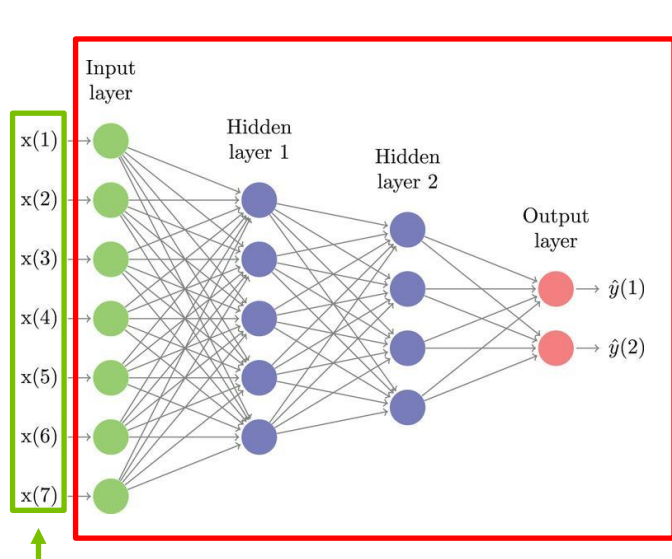


입력 데이터

데이터

- 1) 분류 방법에 대한 답과 예시를 모델에 제공
- 2) 모델이 추측을 수행하고 인간이 모델에 옳고 그름을 알려줌
- 3) 모델이 올바르게 분류하는 법을 스스로 학습

# 딥러닝 (Deep Learning)



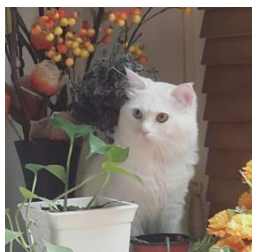
모델

고양이?

자동차?



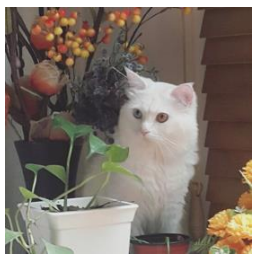
Label: 고양이!



입력 데이터

데이터

# 딥러닝 (Deep Learning)



입력 데이터

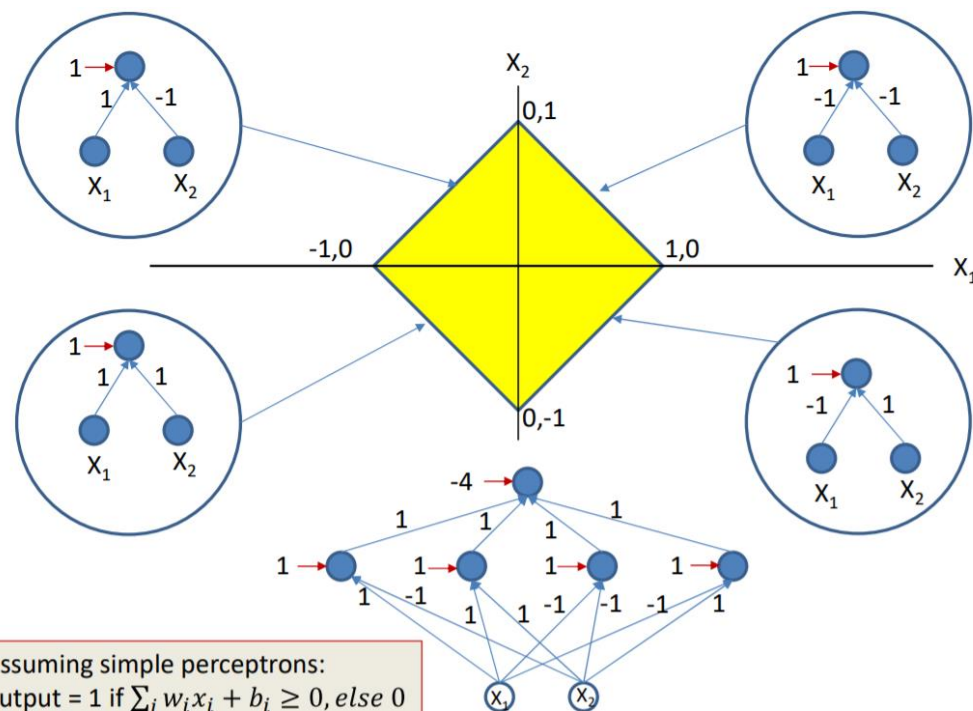
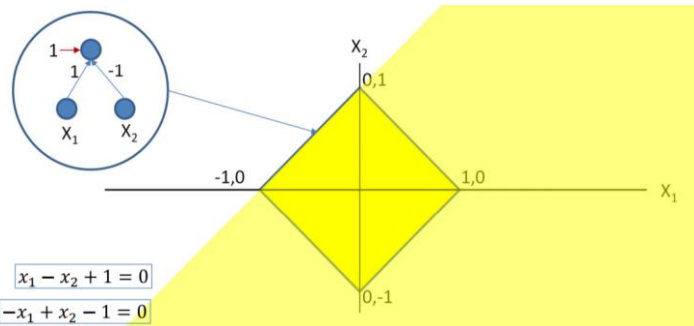
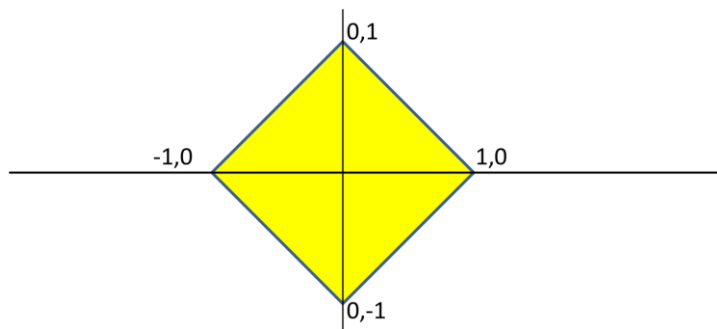
데이터

“왜 깊게 쌓는거지?”



# 딥러닝 (Deep Learning)

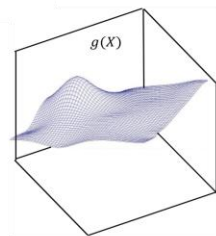
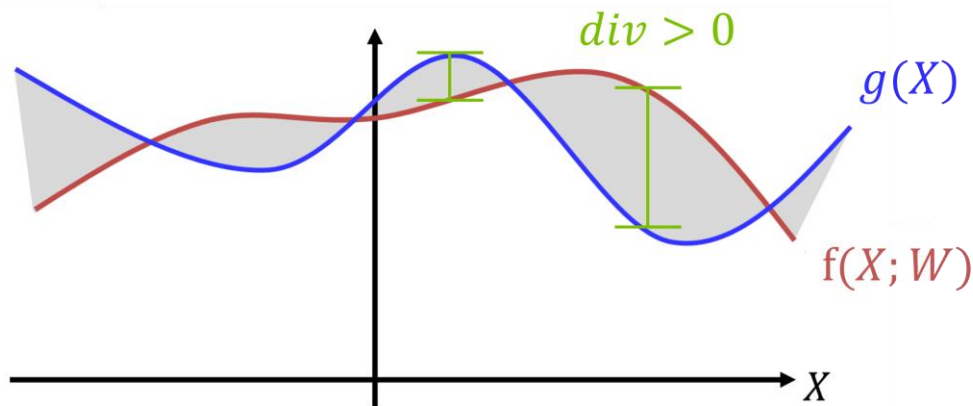
## 1. 퍼셉트론 >> decision boundary



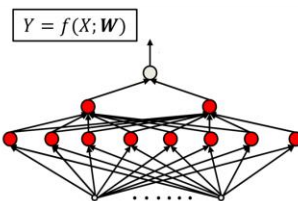
# 딥러닝 (Deep Learning)

## 2. Decision boundary 수식을 어떻게 알지? → 딥러닝의 학습

- $\hat{W} = \underset{W}{\operatorname{argmin}} \int_X \operatorname{div}(f(X; W), g(X)) dX,$
- where  $f(X; W)$  has the capacity to exactly represent  $g(X)$ ,  $\operatorname{div}()$  is a positive divergence function that goes to zero when  $f(X; W) = g(X)$ .



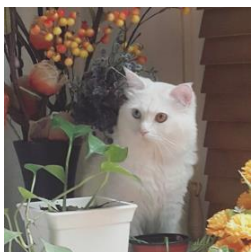
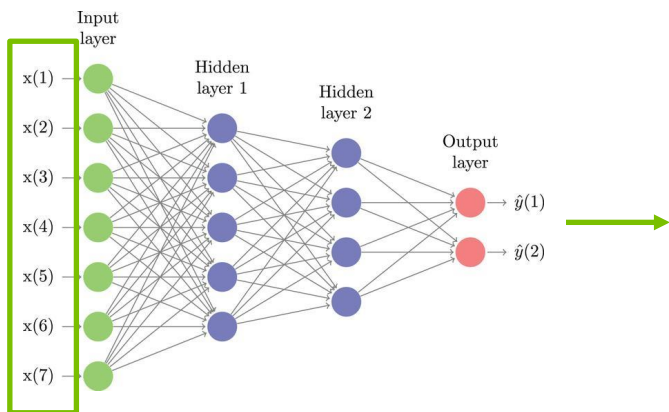
: 실제 정답 함수



: 우리가 학습시키는 함수

# 딥러닝의 구성요소

- 딥러닝: (1) 데이터, (2) 모델 (Neural Networks), (3) 손실함수 (Loss Function)



입력 데이터

모델

손실함수

# 목차 – I 부: Deep Learning

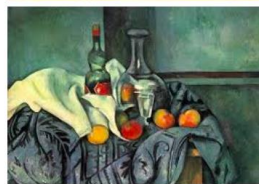
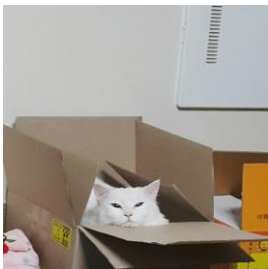
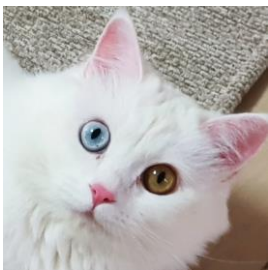
- AI의 역사
- 딥러닝이란 무엇인가
- **딥러닝 혁명**
- 딥러닝이 어떻게 세상을 바꾸는가
- 딥러닝이 어려운 이유
- DeepStream



딥러닝 혁명

# 1. 데이터

- Neural network가 고양이가 무엇인지 이해하기 위해서는 수많은 고양이 사진 데이터와 고양이가 아닌 사진 데이터에 노출되어야함
- 디지털 시대와 인터넷이 이러한 데이터를 공급해주었음





## 2. 컴퓨팅 성능

인공 '두뇌'가 현실적인 시간 내에 다량의 데이터를 관찰할 수 있는 방법이 필요함



### 3. GPU의 중요성

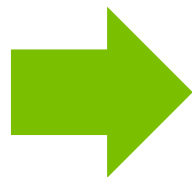
- e.g.) NVIDIA Omniverse (3D simulation)



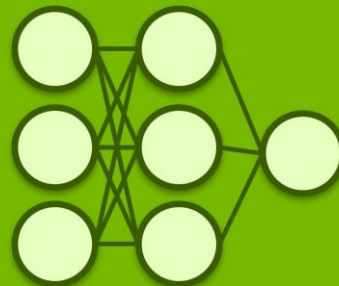
### 3. GPU의 중요성

- 평균적인 CPU에는 병렬 처리를 위한 코어가 4개, 또는 8개가 있음
- GPU는 수천개의 병렬 처리를 위한 코어를 가지고 있기 때문에 컴퓨터 그래픽스 같은 분야에서 엄청난 성능을 보여줄 수 있음
- Neural network를 training 하는 연산은 간단하지만 많게는 수십억, 수조 회의 병렬 계산이 수행되어야 함

렌더링된 이미지



뉴럴 네트워크



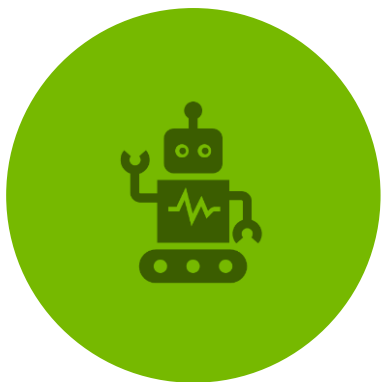
# 목차 – I 부: Deep Learning

- AI의 역사
- 딥러닝이란 무엇인가
- 딥러닝 혁명
- 딥러닝이 어떻게 세상을 바꾸는가
- 딥러닝이 어려운 이유
- DeepStream



딥러닝이 어떻게 세상을 바꾸는가

# 1. 컴퓨터 비전



로보틱스 및 제조



물체 검출



자율 주행 자동차

## 2. 자연어 처리



실시간 번역



음성 인식



가상 어시스턴트



### 3. 추천 시스템



콘텐츠 큐레이션



타깃 광고



쇼핑 추천

## 4. 강화 학습



알파고



AI 봇



주식 거래 로봇

# 목차 – I 부: Deep Learning

- AI의 역사
- 딥러닝이란 무엇인가
- 딥러닝 혁명
- 딥러닝이 어떻게 세상을 바꾸는가
- **딥러닝이 어려운 이유**
- DeepStream

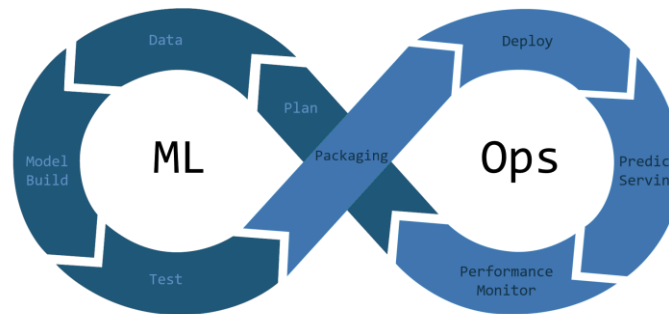


딥러닝이 어려운 이유

# 딥러닝이 어려운 이유

## 1. 방대한 전문지식을 가진 전문가들이 많이 필요함

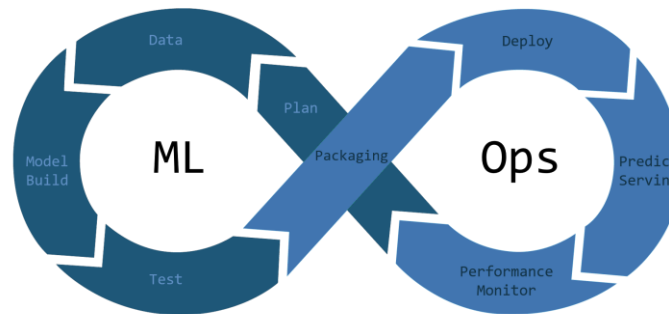
- 딥러닝 기본 이론
- Tensorflow, Pytorch framework를 능숙하게 다룰 줄 알아야함
- 딥러닝 모델을 최적화 하고 배포하기 위한 엔지니어링 지식
- 새로운 모델을 연구하고 개발하기 위한 리서치 능력



# 딥러닝이 어려운 이유

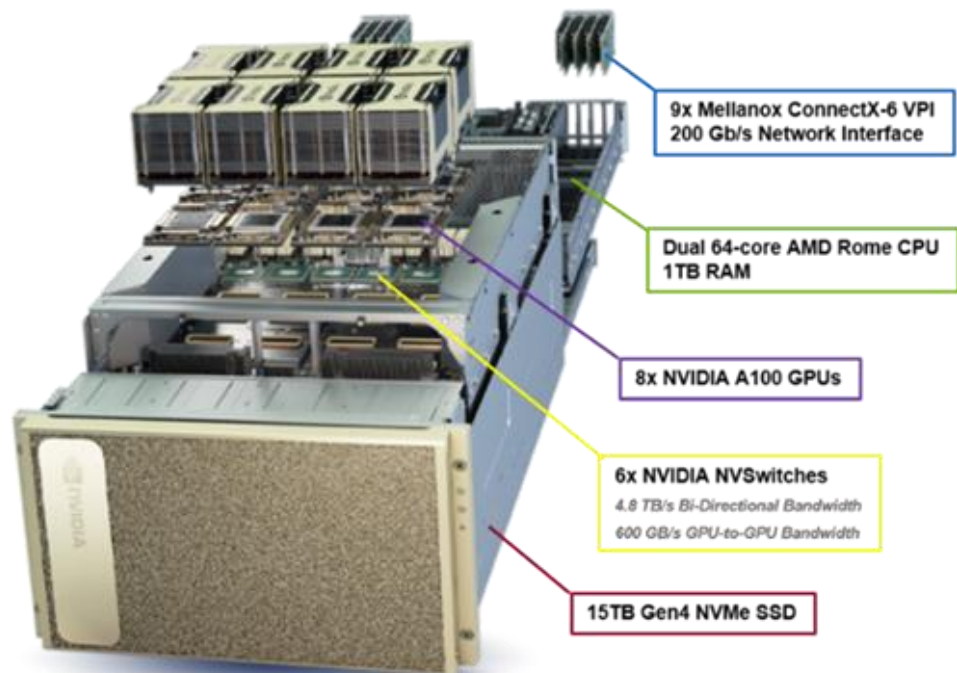
1. 방대한 전문지식을 가진 전문가들이 많이 필요함

- 딥러닝 기본 이론
- Tensorflow, Pytorch framework를 능숙하게 다룰 줄 알아야함
- 딥러닝 모델을 최적화 하고 배포하기 위한 엔지니어링 지식
- 새로운 모델을 연구하고 개발하기 위한 리서치 능력



# 딥러닝이 어려운 이유

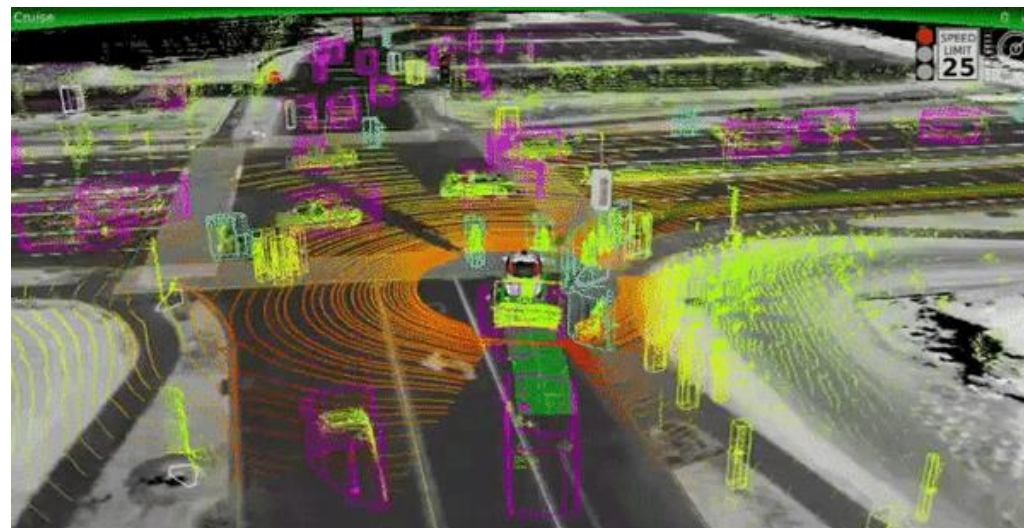
2. 하드웨어 (e.g., GPUs)와 하드웨어를 움직이는 소프트웨어를 능숙하게 다룰줄 알아야함





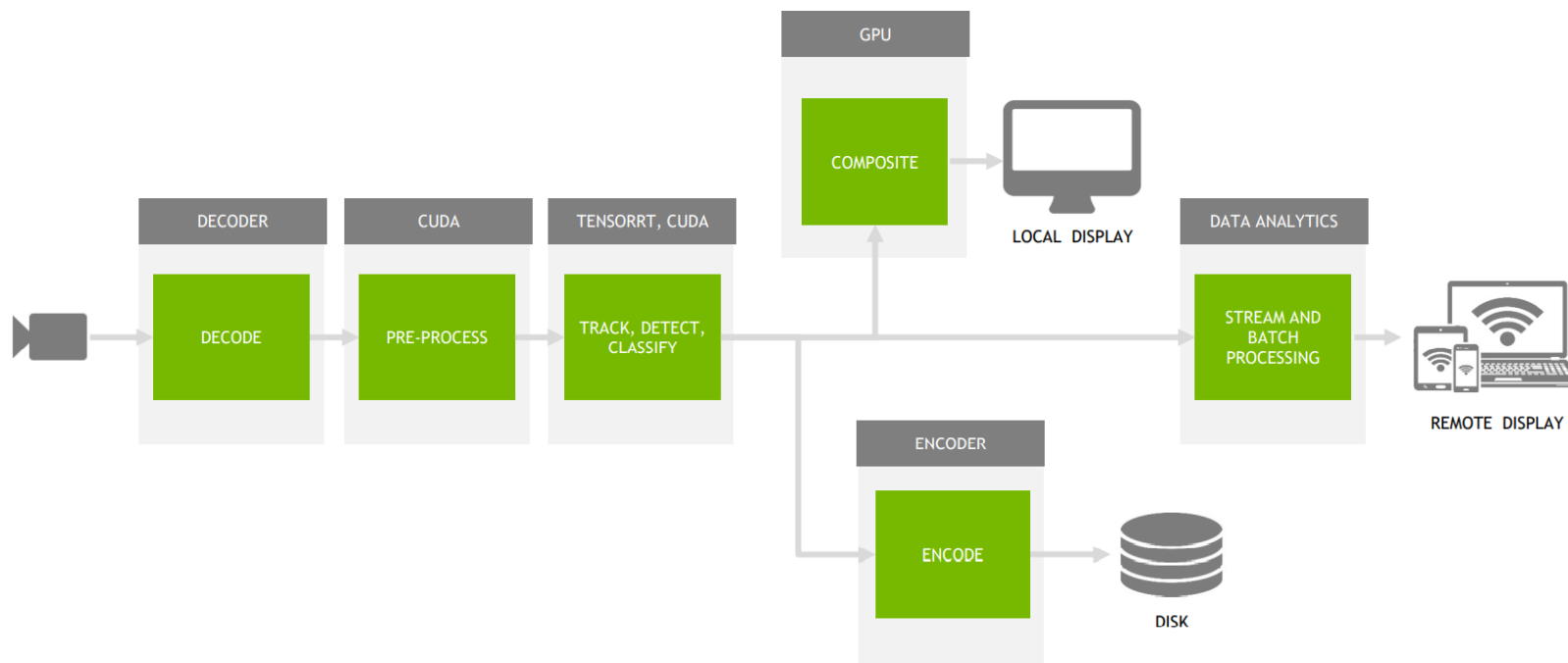
# 딥러닝이 어려운 이유

3. Real-time application (e.g., 무인차)들은 실시간으로 데이터를 처리하는데 어려움이 있음



# 딥러닝이 어려운 이유

4. 보통 딥러닝 모델 개발시 Small edge device (e.g., 휴대폰)부터 Public cloud까지 다양한 플랫폼에서 사용 가능하도록 고려해야한다



# 딥러닝이 어려운 이유

4. 보통 딥러닝 모델 개발시 Small edge device (e.g., 휴대폰)부터 Public cloud까지 다양한 플랫폼에서 사용 가능하도록 고려해야한다

아이디어와 데이터만 있으면 딥러닝 모델을  
쉽게 만들고 배포할 수 있는 방법이 없을까?

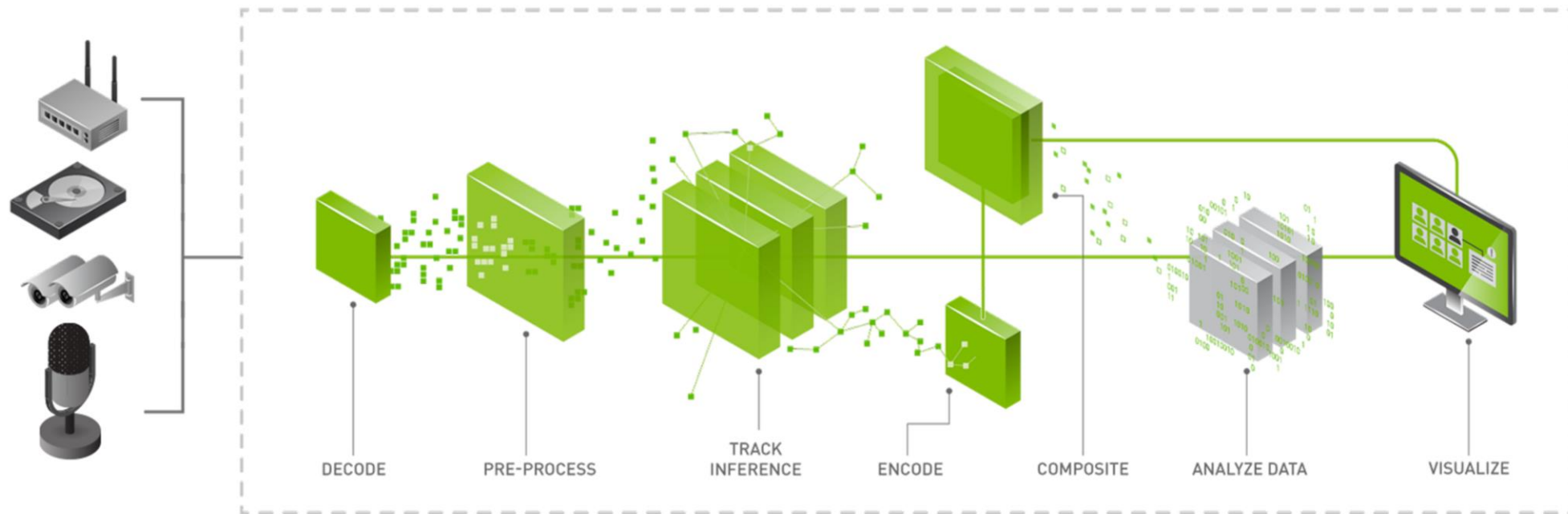


# 목차 – I 부: Deep Learning

- AI의 역사
- 딥러닝이란 무엇인가
- 딥러닝 혁명
- 딥러닝이 어떻게 세상을 바꾸는가
- 딥러닝이 어려운 이유
- **DeepStream**

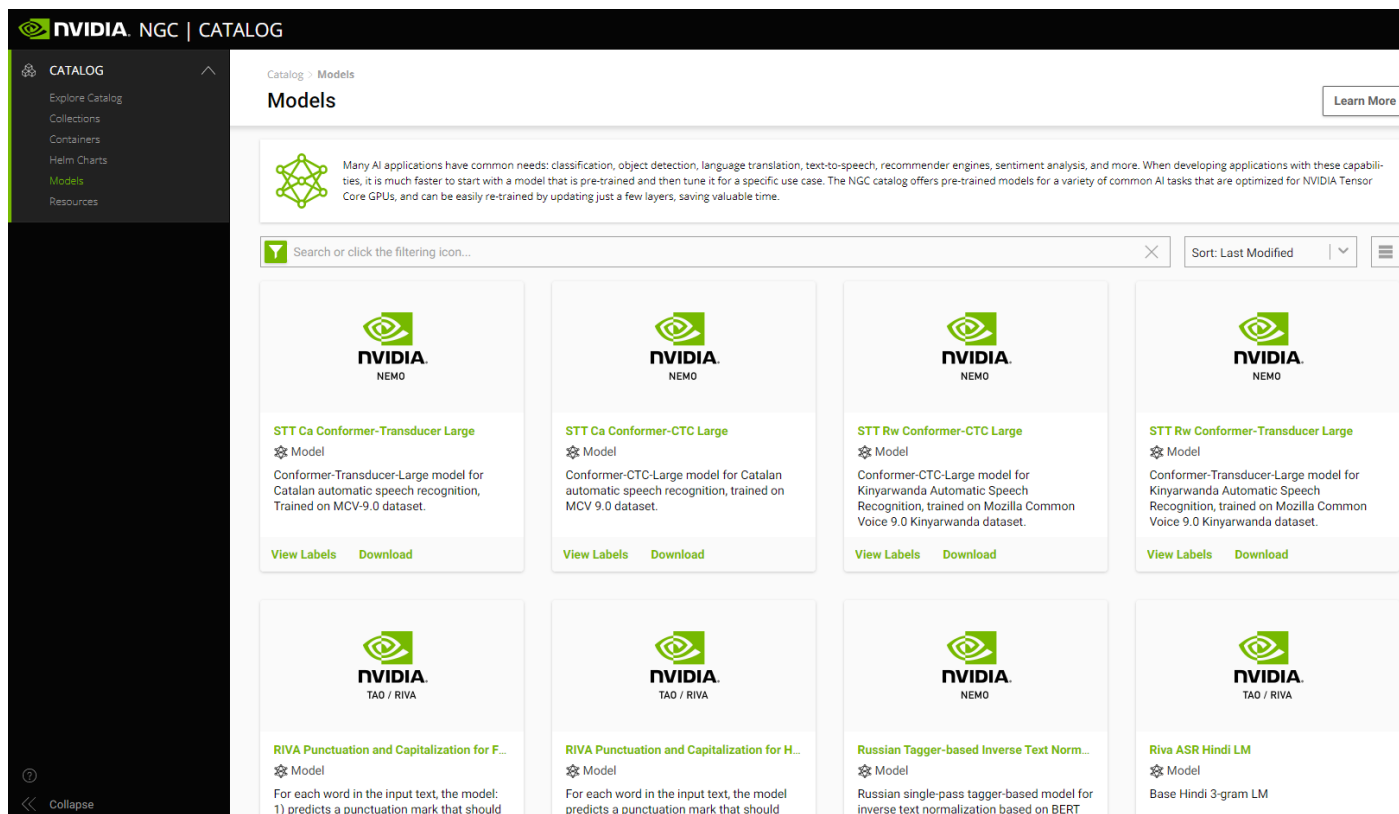
# DeepStream

- DeepStream이란? 실시간 비디오 처리를 위한 모델 파이프라인을 쉽게 만들 수 있는 framework.



# DeepStream

- 별도의 학습과정 없이 Cloud에서 모델을 불러서 쓰기만 하면 된다.





10분 쉬고  
바로 2부 시작하겠습니다!



# 목차 – 2부: DeepStream

- API Key Setup
- DeepStream이 좋은 이유
- DeepStream Pipeline 구축 (Part 1)
- DeepStream Pipeline 구축 (Part 2)
- DeepStream 활용을 위한 참고 사이트

# 목차 – 2부: DeepStream

- **API Key Setup**
- DeepStream이 좋은 이유
- DeepStream Pipeline 구축 (Part 1)
- DeepStream Pipeline 구축 (Part 2)
- DeepStream 활용을 위한 참고 사이트

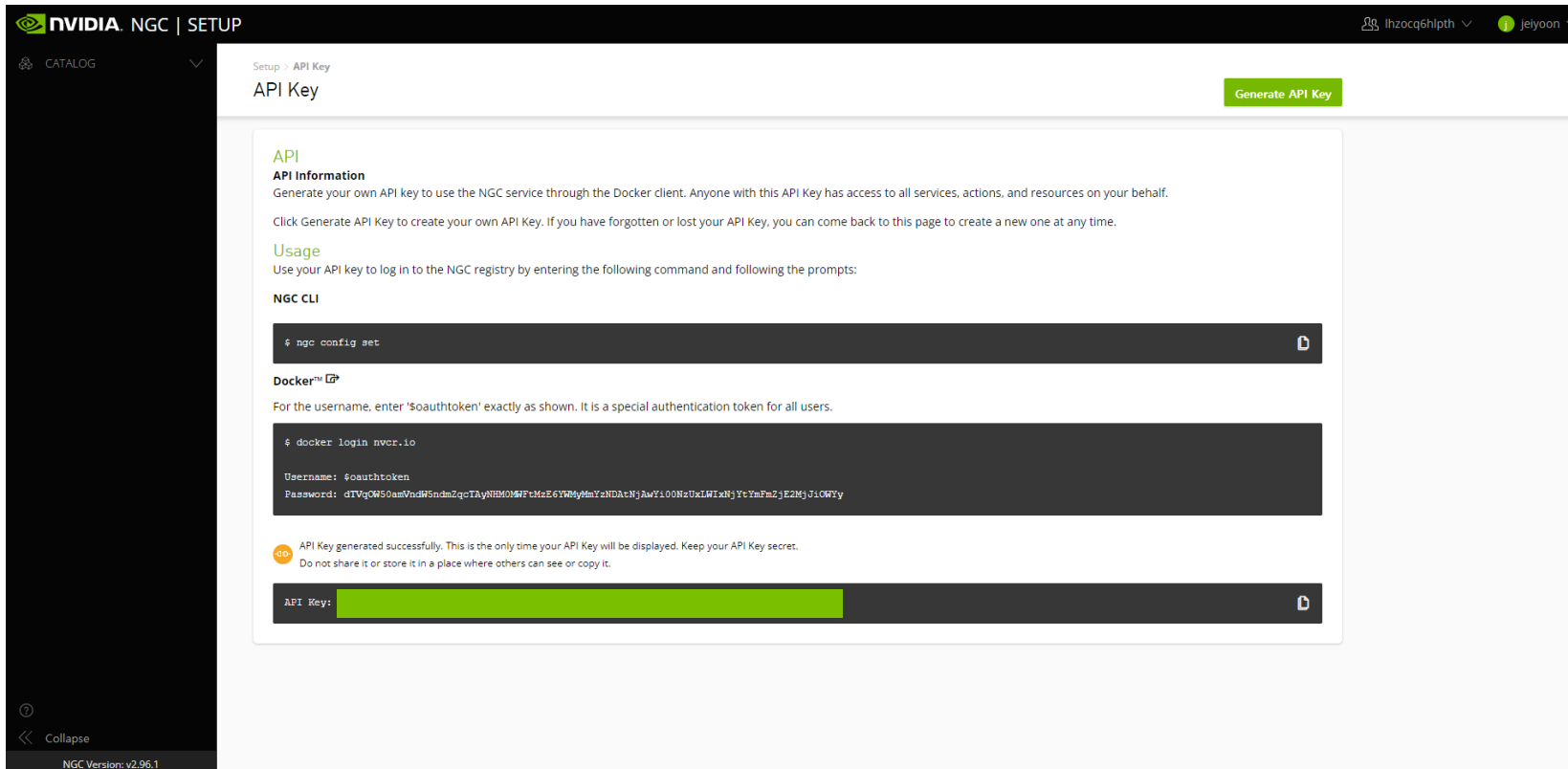
# API Key Setup

- DeepStream을 사용하기 위해 Nvidia GPU Cloud (NGC)에서 API Key를 받아옴
- <https://catalog.ngc.nvidia.com/>

The screenshot displays the NVIDIA NGC Setup page. On the left, there is a dark sidebar with the NVIDIA logo and 'NGC | SETUP' text. Below it, a 'CATALOG' link is visible. The main content area is titled 'Setup' and contains two primary sections. The first section, 'Generate API Key', is highlighted with a red rectangular box. It features a key icon, the heading 'Generate API Key', a descriptive paragraph about using the NGC service, and a green 'Get API Key' button. The second section, 'CLI', includes a terminal icon, the heading 'CLI', a paragraph about the NGC command line interface, and two buttons: 'Documentation' and 'Downloads'. On the right side of the page, a user profile dropdown menu is open, showing options like 'My Account Settings', 'Setup' (which is highlighted with a red rectangular box), 'Terms of Use', 'Privacy Policy', and 'Sign Out'. The top right of the page shows the user's email 'lhzcq6hlpth' and a profile picture.

# API Key Setup

- DeepStream을 사용하기 위해 Nvidia GPU Cloud (NGC)에서 API Key를 받아옴



# API Key Setup

- DeepStream을 사용하기 위해 Nvidia GPU Cloud (NGC)에서 API Key를 받아옴

```
%%writefile config
;WARNING - This is a machine generated file. Do not edit manually.
;WARNING - To update local config settings, see "ngc config set -h"

[CURRENT]
apikey=<<<<MY_NGC_API_KEY>>>>
format_type=json
org=nvidia
```

```
# DO NOT CHANGE THIS CELL
!mkdir -p ~/.ngc & mv config ~/.ngc/
```

# 목차 – 2부: DeepStream

- API Key Setup
- **DeepStream**이 좋은 이유
- DeepStream Pipeline 구축 (Part 1)
- DeepStream Pipeline 구축 (Part 2)
- DeepStream 활용을 위한 참고 사이트

# DeepStream이 좋은 이유

- 세상에는 정말 다양한 곳에서 Video AI가 필요하다.



Access Control



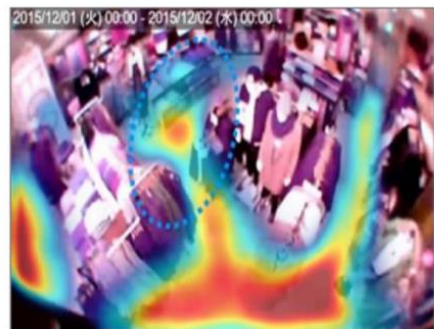
Managing operations



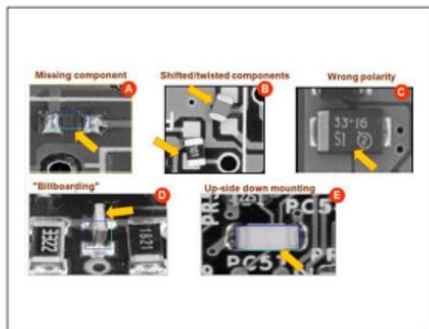
Parking Management



Traffic Engineering



Retail Analytics



Optical Inspection



Managing Logistics



Content Filtering

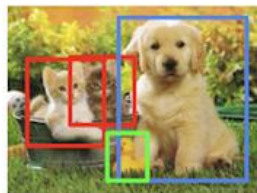


# DeepStream이 좋은 이유

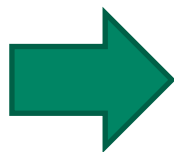
- 다양한 Real-Time Video AI application들을 쉽게 만들 수 있다.



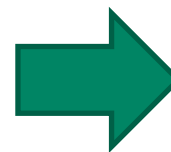
CAT



CAT, DOG, DUCK



DeepStream  
파이프 라인



Classification



CAT

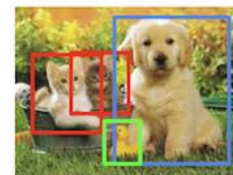
Classification  
+ Localization



CAT

Single object

Object Detection



CAT, DOG, DUCK

Instance  
Segmentation



CAT, DOG, DUCK

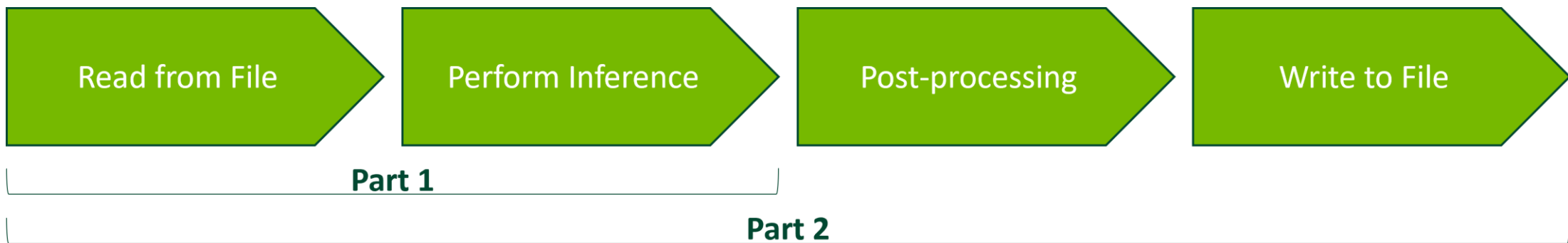
Multiple objects

# 목차 – 2부: DeepStream

- API Key Setup
- DeepStream이 좋은 이유
- **DeepStream Pipeline 구축 (Part 1)**
- DeepStream Pipeline 구축 (Part 2)
- DeepStream 활용을 위한 참고 사이트

# DeepStream Pipeline 구축

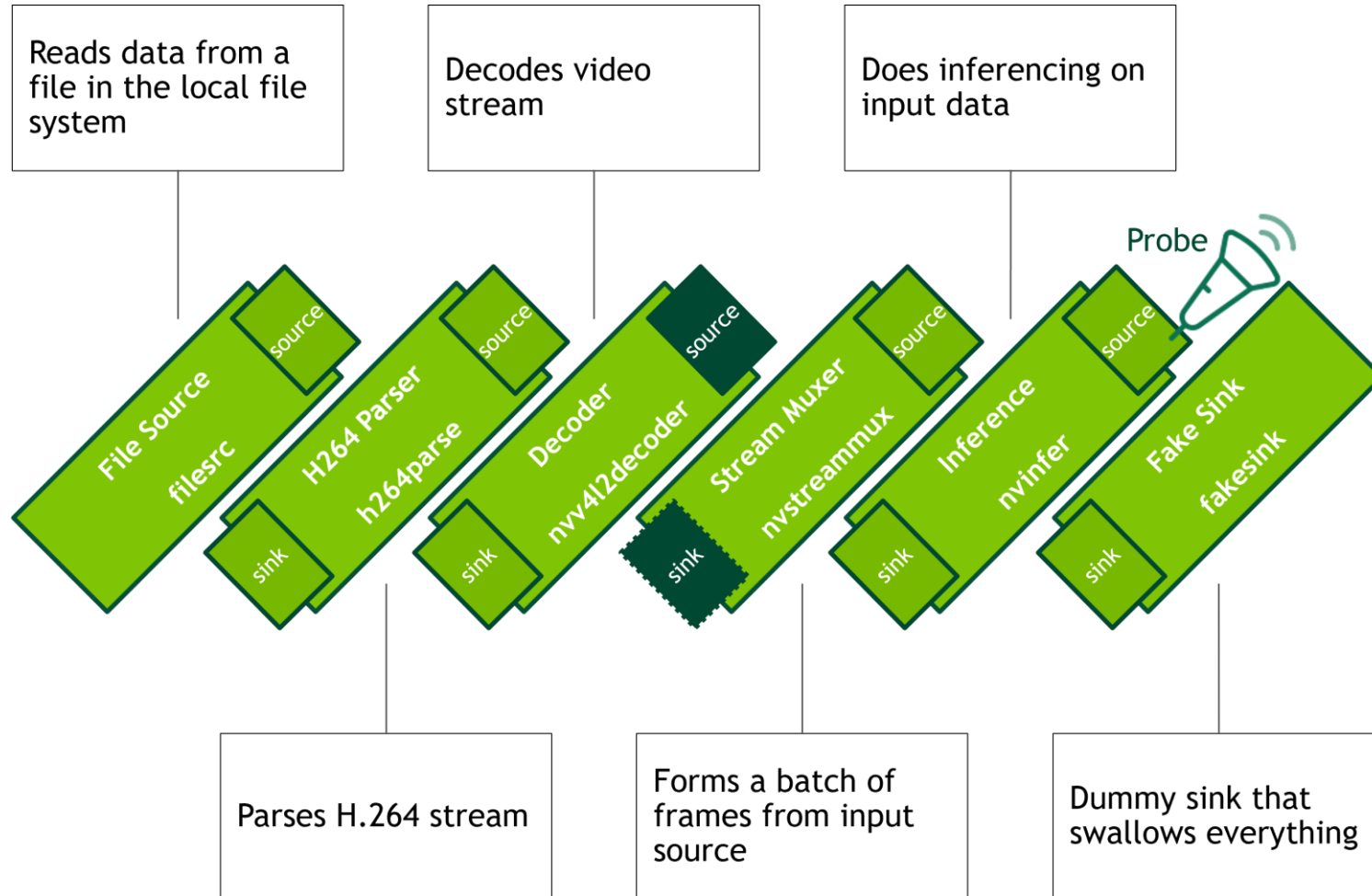
- Part 1: 데이터를 읽고 파이프라인에 넣고 출력 결과 확인
- Part 2: 파이프라인 출력결과에 Bounding box, text label등을 추가, 저장



An abstract graphic on the left side of the slide, consisting of a complex network of white lines connecting small white and green dots, resembling a molecular structure or a data network. The dots are arranged in a way that suggests a hierarchical or interconnected system, with some dots being highlighted in green.

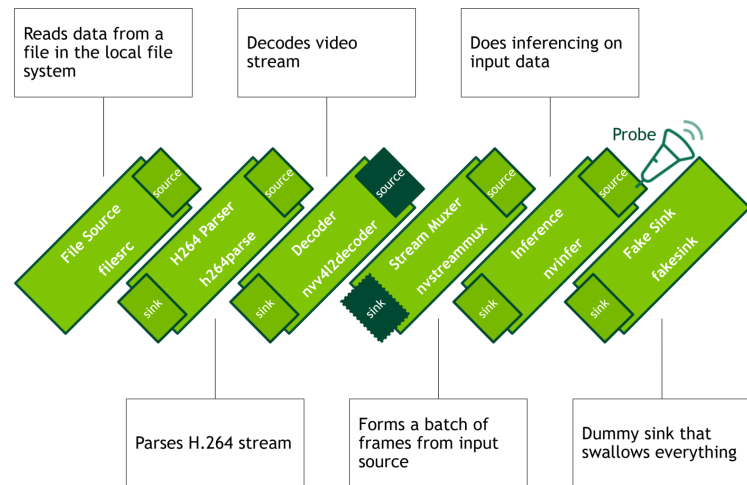
# Part1 Code Review

# Part 1 요약



# Part 1 요약

- NGC 설치
- TrafficCamNet Object Detection Model 불러오기
- GStreamer and Pipeline 초기화 (`Gst.init(None)`)
- Pipeline 요소 생성 (`pipeline.add()`)
- Configuration 설정 (`pgie.set_property()`)
- 파이프라인 내 요소들 링킹 (`link()`)
- Probe를 이용하여 metadata 확인 (e.g., 프레임 당 예측된 차량 수)
  - Data가 pad (i.e., source or sink pad)를 지나갈때 probe로 확인
  - `gst_buffer_get_nvds_batch_mata()` / `pyds.NvDsFrameMeta.cast()`
- 파이프라인 시작
  - Bus: 메시지 모니터 하는곳 / `get_bus()` / `add_signal_watch()`
  - `loop = Glib.MainLoop()`
- Inference
  - `loop.run()`



# 목차 – 2부: DeepStream

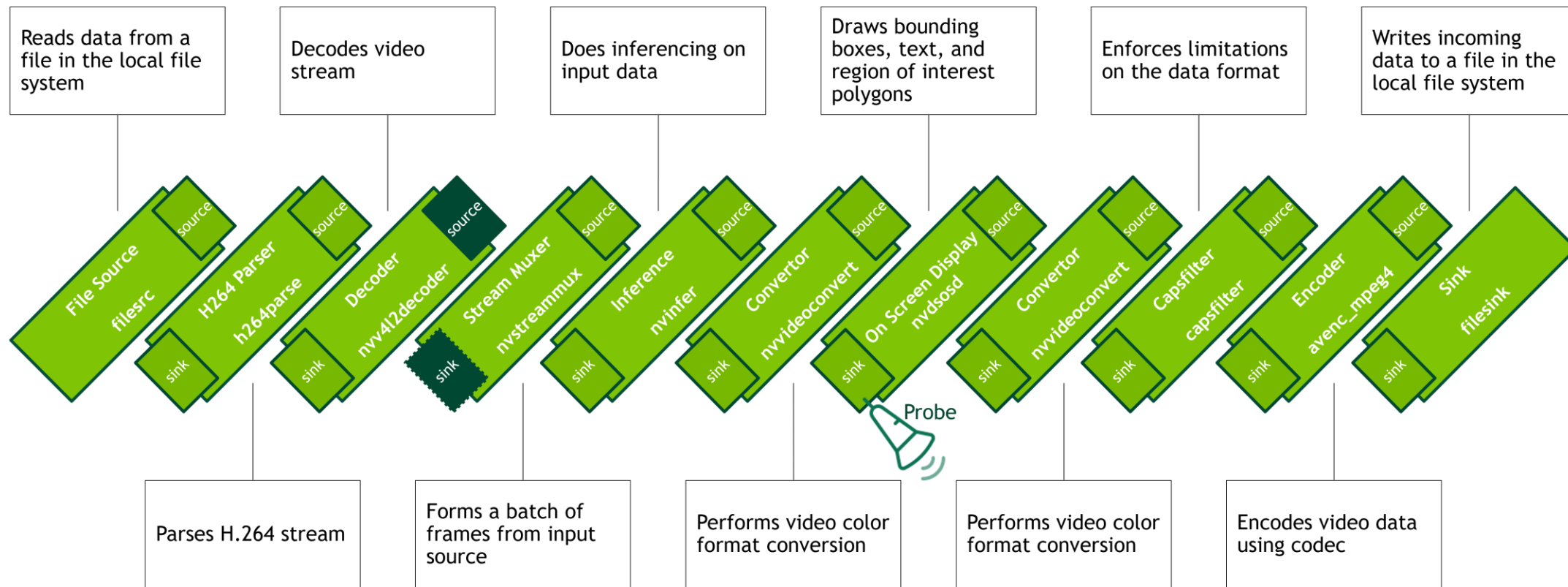
- API Key Setup
- DeepStream이 좋은 이유
- DeepStream Pipeline 구축 (Part 1)
- **DeepStream Pipeline 구축 (Part 2)**
- DeepStream 활용을 위한 참고 사이트



An abstract graphic on the left side of the slide, consisting of a complex network of white lines connecting small white and green dots, resembling a molecular structure or a data network. The dots are arranged in a way that suggests a hierarchical or interconnected system, with some dots being highlighted in green.

## Part2 Code Review

# Part 2 요약



## Part 2 요약

- Part1에서 했던거
- fakesink 제거 (*pipeline.get\_by\_name('fakesink')*)
- nvdsosd: bounding box랑 text label 그려주는 모듈
- nvvideoconvert: video frame format 설정 (e.g., RGBA to I420 (YUV))
- capsfilter: 출력 데이터 포맷 설정 (e.g., I420)
- avenc\_mpeg4: MPEG4 codec을 이용하여 I420 formatted frame으로 encode
- filesink: 데이터 저장

# 목차 – 2부: DeepStream

- API Key Setup
- DeepStream이 좋은 이유
- DeepStream Pipeline 구축 (Part 1)
- DeepStream Pipeline 구축 (Part 2)
- **DeepStream** 활용을 위한 참고 사이트

# DeepStream 활용을 위한 참고 사이트

- DeepStream github:  
([https://github.com/NVIDIA-AI-IOT/deepstream\\_python\\_apps](https://github.com/NVIDIA-AI-IOT/deepstream_python_apps))
- DeepStream SDK:  
(<https://developer.nvidia.com/deepstream-sdk>)
- GST-python github:  
(<https://github.com/GStreamer/gst-python>)
- DeepStream Plugin Guide:  
(<https://docs.nvidia.com/metropolis/deepstream/dev-guide/index.html#plugins-development-guide>)
- GStreamer Plugin Guide:  
([https://gstreamer.freedesktop.org/documentation/plugins\\_doc.html?gi-language=c](https://gstreamer.freedesktop.org/documentation/plugins_doc.html?gi-language=c))

# DeepStream 활용을 위한 참고 사이트

- e.g., DeepStream github:

The screenshot shows the GitHub repository for 'NVIDIA-AI-IOT / deepstream\_python\_apps'. The repository is public and has 57 watchers. The main branch is 'master'. The repository contains a file tree with folders like '3rdparty', 'apps', 'bindings', 'docs', 'notebooks', 'tests', and files like '.gitmodules', '.python-app-pipeline.png', '.test3-app.png', 'FAQ.md', 'HOWTO.md', 'LICENSE', 'README.md', and 'THIRD\_PARTY\_LICENSE'. The repository is updated to 1.1.3 release. The repository is described as 'DeepStream SDK Python bindings and sample applications'. The repository has 833 stars, 57 watching, and 319 forks. The repository has 9 releases, with the latest release being 'DeepStream\_Python\_Apps\_Bind...' on 17 Jun. The repository has no packages published. The repository has 3 contributors: nv-zhiliu, nv-rpaliwal, and aparnachhajer.

Download the Deepstream Python bindings file from here:

[https://developer.nvidia.com/deepstream\\_python\\_v0.5](https://developer.nvidia.com/deepstream_python_v0.5) Untar the file in the following folder: tar xf ds\_pybind\_0.5.tbz2 -C <DeepStream 4.0.1 ROOT>/sources

```
tar xf ds_pybind_0.5.tbz2 -C /home/nvidia/Downloads/deepstream/sources
```

## Package Contents

The DeepStream Python package includes:

- Python bindings for DeepStream Metadata libraries These bindings are provided as a compiled module, available for x86\_64 and Jetson platforms. Find them at: bindings | - pyds.so | - x86\_64 | - jetson | - pyds.so
- DeepStream test apps in Python Four test apps are available: deepstream-test1/deepstream\_test\_1.py deepstream-test2/deepstream\_test\_2.py deepstream-test3/deepstream\_test\_3.py deepstream-test3/deepstream\_test\_4.py

## Run code:

Note - Make sure the folder path of where pyds.so file is located is in PATH.

## Importing necessary libraries

```
import sys
sys.path.append('.')
sys.path.append('/home/nvidia/Downloads/deepstream/sources/python/bindings/x86_64')
sys.path.append('/usr/lib/python3/dist-packages')
from common.is_aarch64 import is_aarch64
import gi
gi.require_version('Gst', '1.0')
from gi.repository import GObject, Gst
from common.is_aarch64 import is_aarch64
from common.bus_call import bus_call

import pyds
```

## Declaring class label ids

# 감사합니다

<https://jeiyoong.github.io/>



DEEP  
LEARNING  
INSTITUTE