TAIWAN ONLINE COURSES

AOI01-AOI Basics

Hsueh-Ting Chu

朱學亭老師

- 現職：亞洲大學資工系副教授(兼AI研究組長)

- EMAIL: htchu.taiwan@gmail.com
- FB: https://www.facebook.com/htchu.taiwan

Outlines

01
- Ways of optical inspection
- Overkill and underkill problems

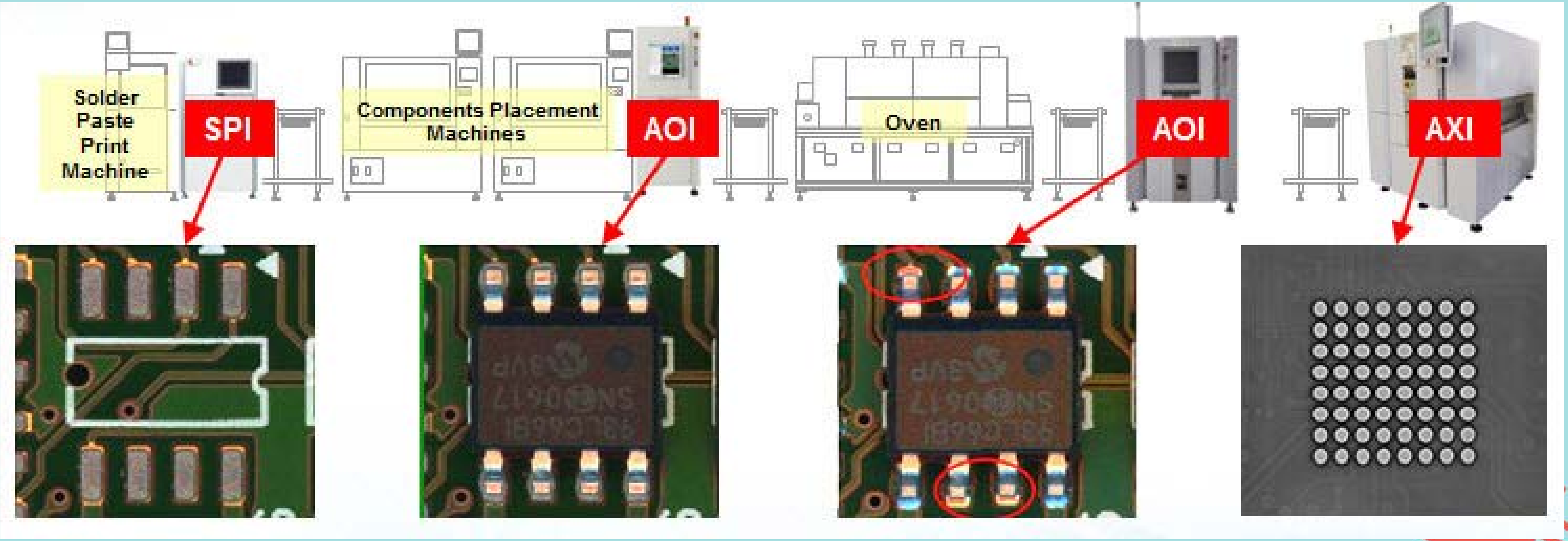
02
- Deep Learning & CNN
- Transfer Learning

03
- AIdea AOI dataset
- Jupyter notebook and Colab

01-Automated Optical Inspection

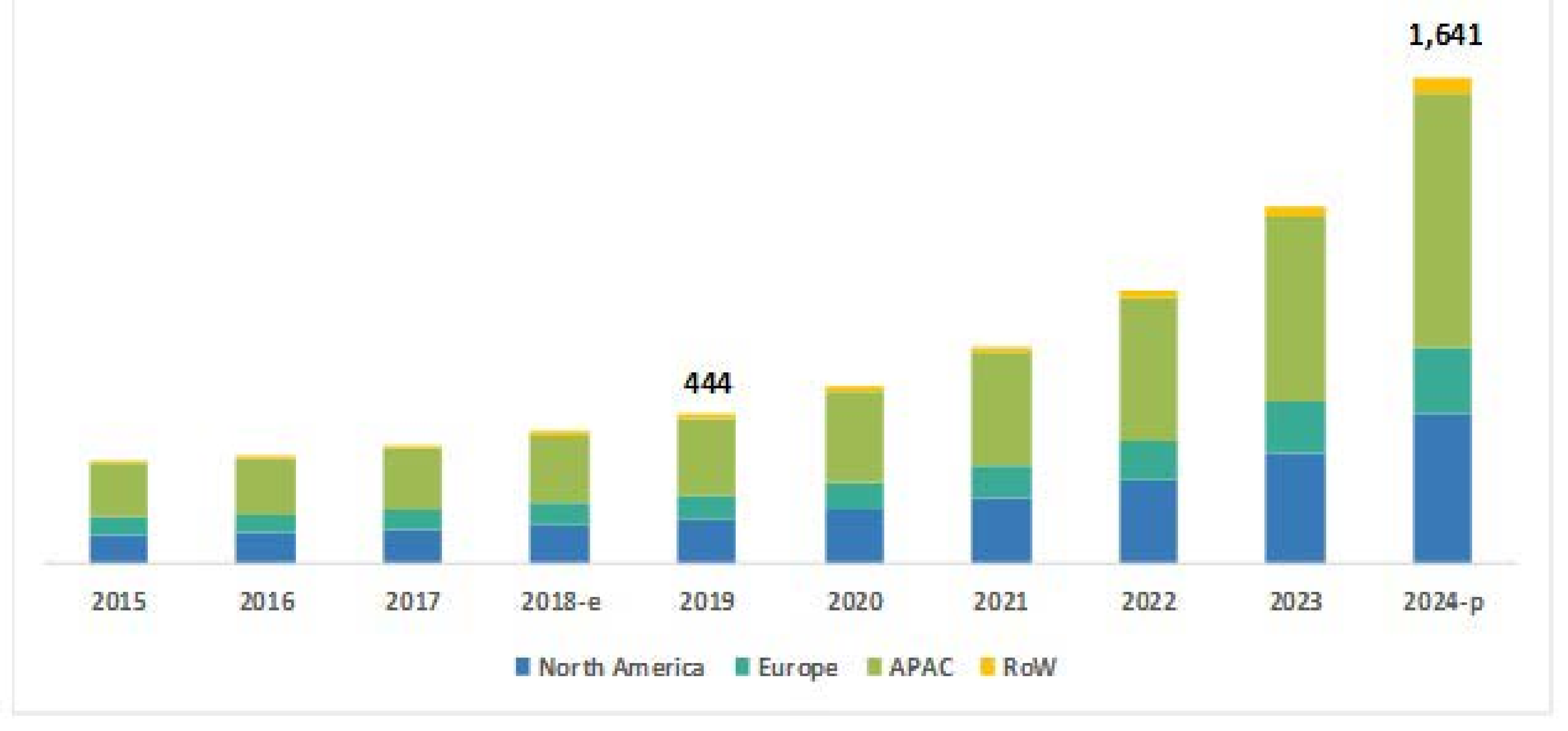
AOI Machine

Types of Automated Optical Inspection



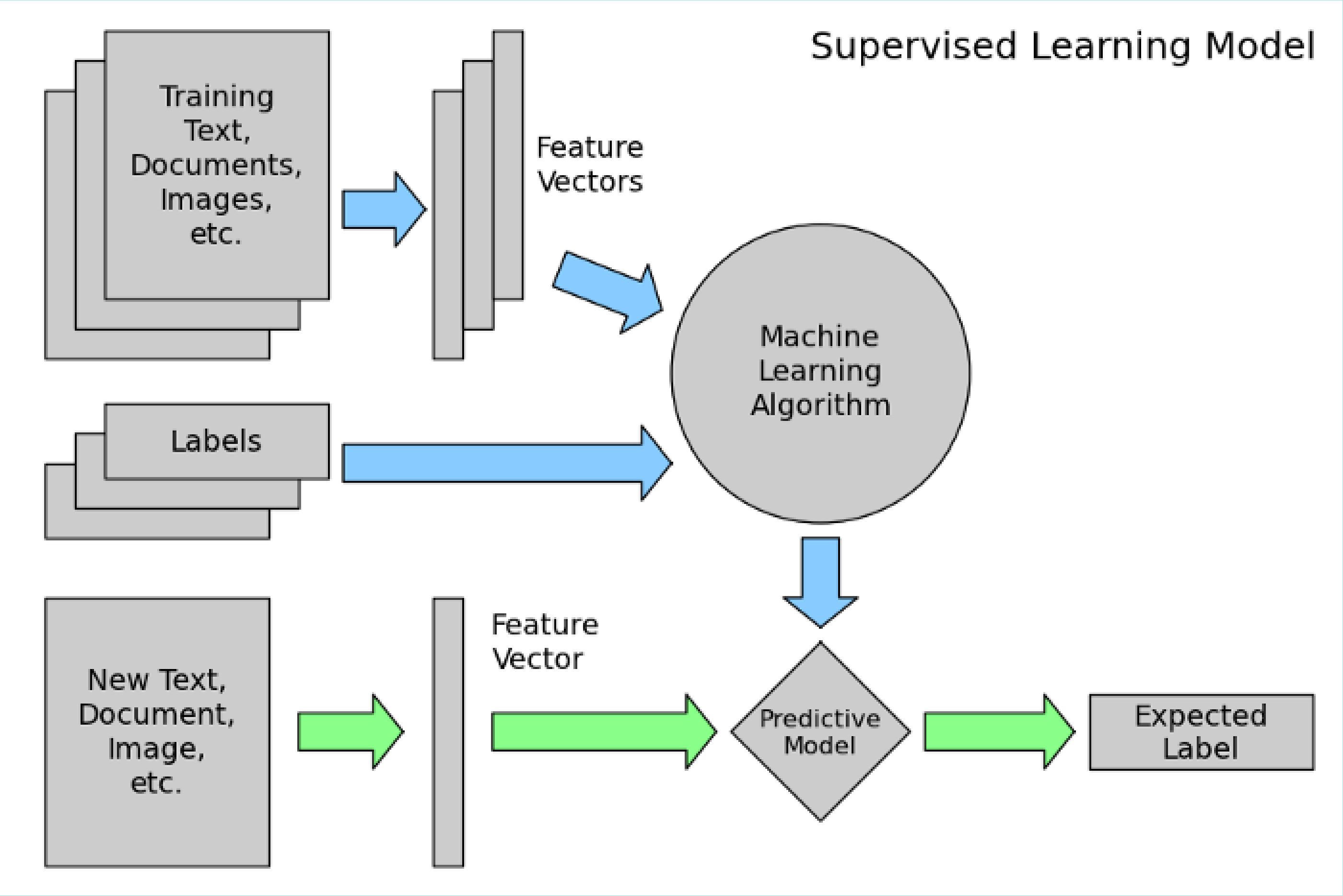
SPI (Solder Paste Inspection)
AOI (Auto Optical Inspection)
AXI (Automatic X-ray Inspection)

Automated Optical Inspection System Market

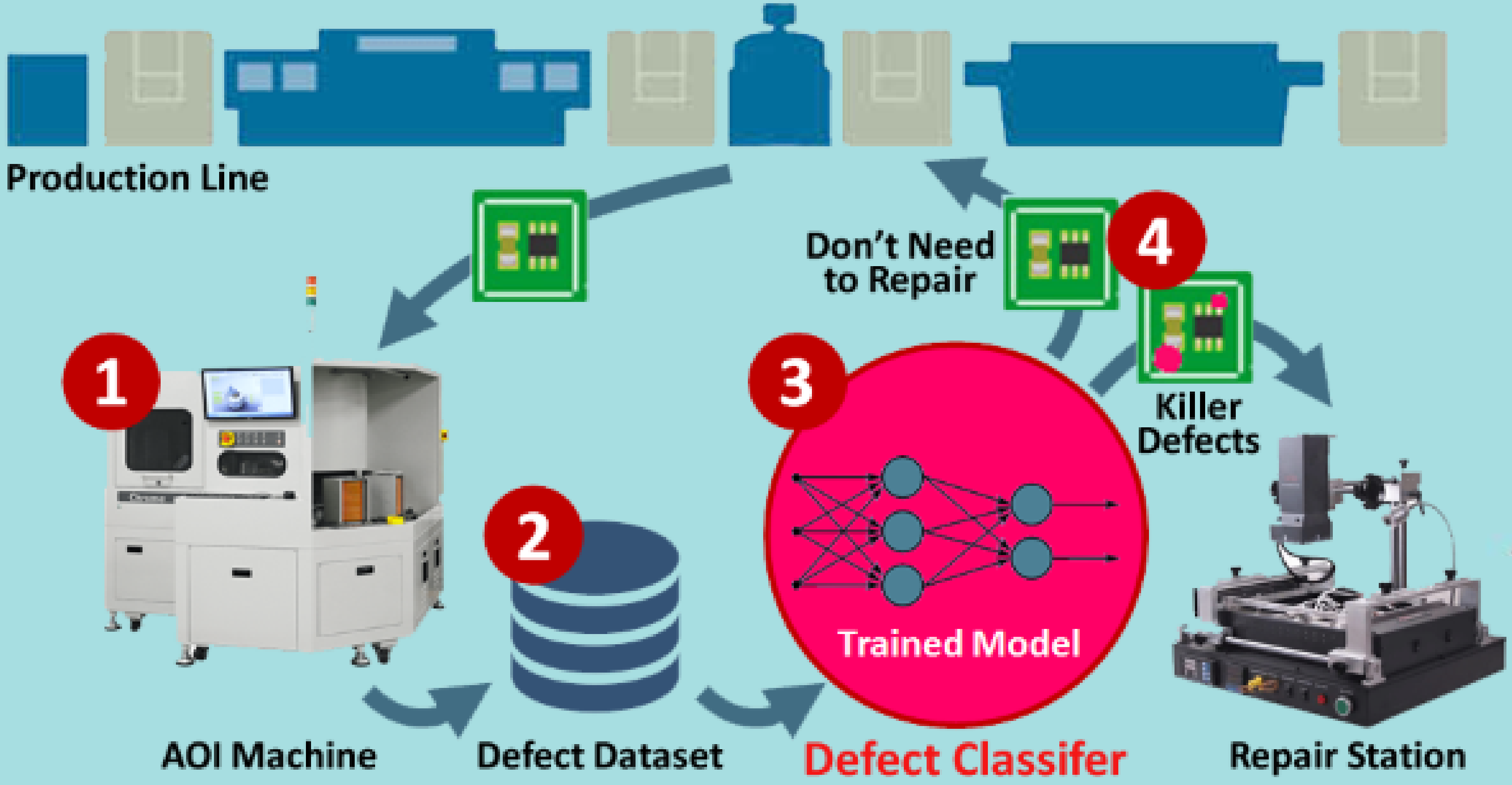


http://semiconductorreports.over-blog.com/2019/01/automated-optical-inspection-system-market-emerging-trends-analysis-and-forecasts-2018-2023.html

Machine Learning model



AOI with Deep Learning models



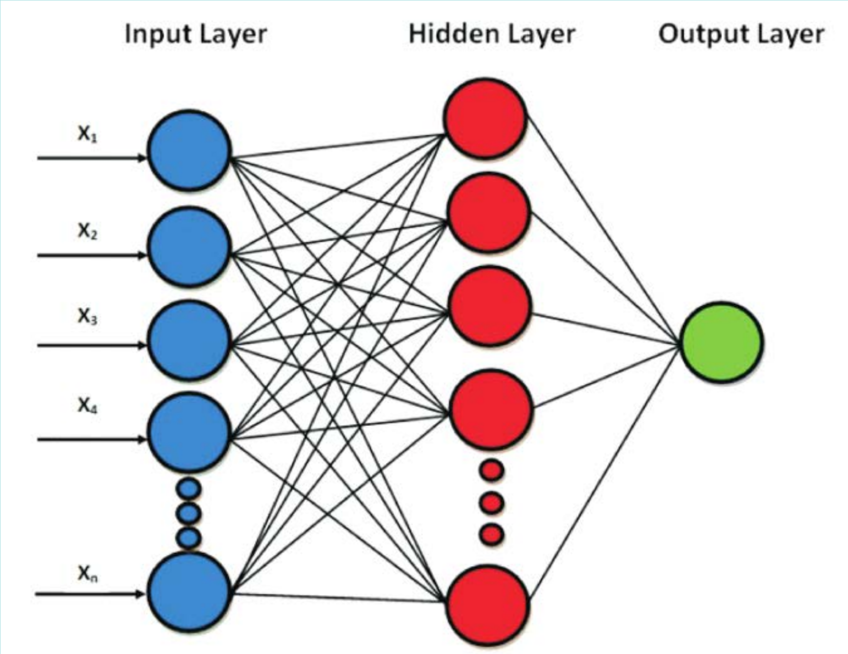
AOI

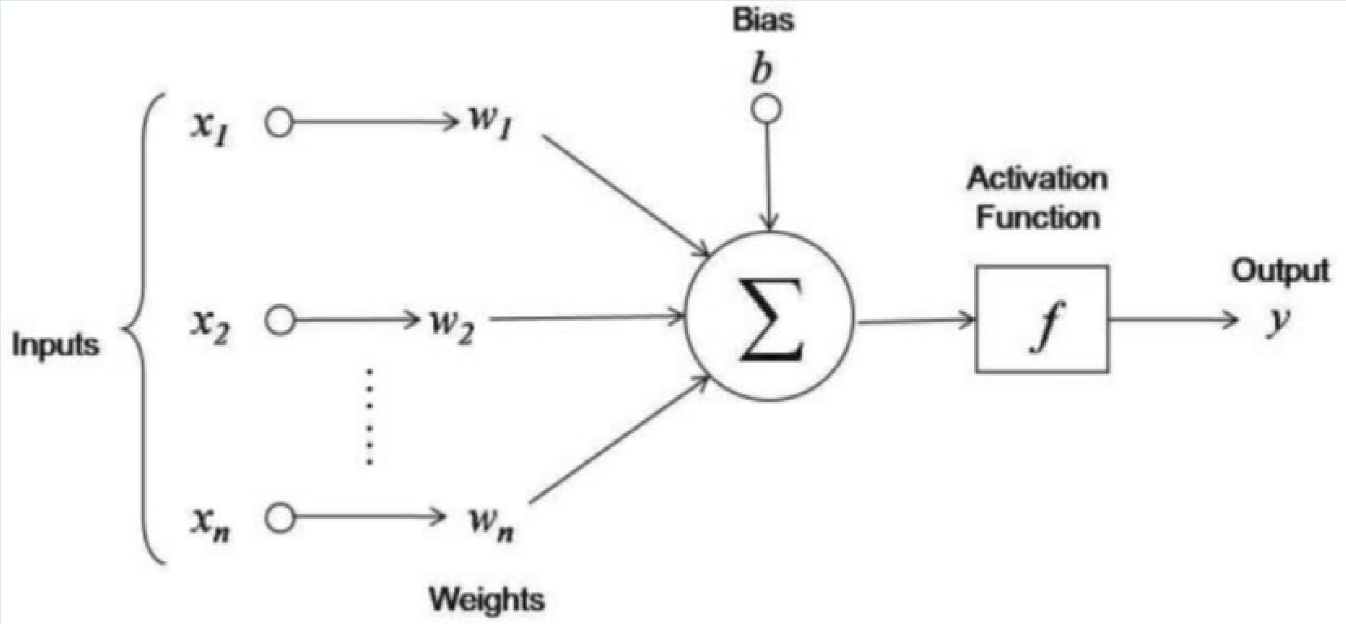
- 檢錯率（underkill rate） Bad->Good
- 誤判率（overkill rate） Good->Bad

- 傳統AOI檢測作業上，為了讓不讓瑕疵物件被放過，通常會嚴格設定條件，導致許多誤殺（overkill）的情形，因此，還是必須要設有複檢人員再次檢測，這也是許多製造業廠商想導入AI來提升檢測效率的原因。

02-Deep Learning for AOI

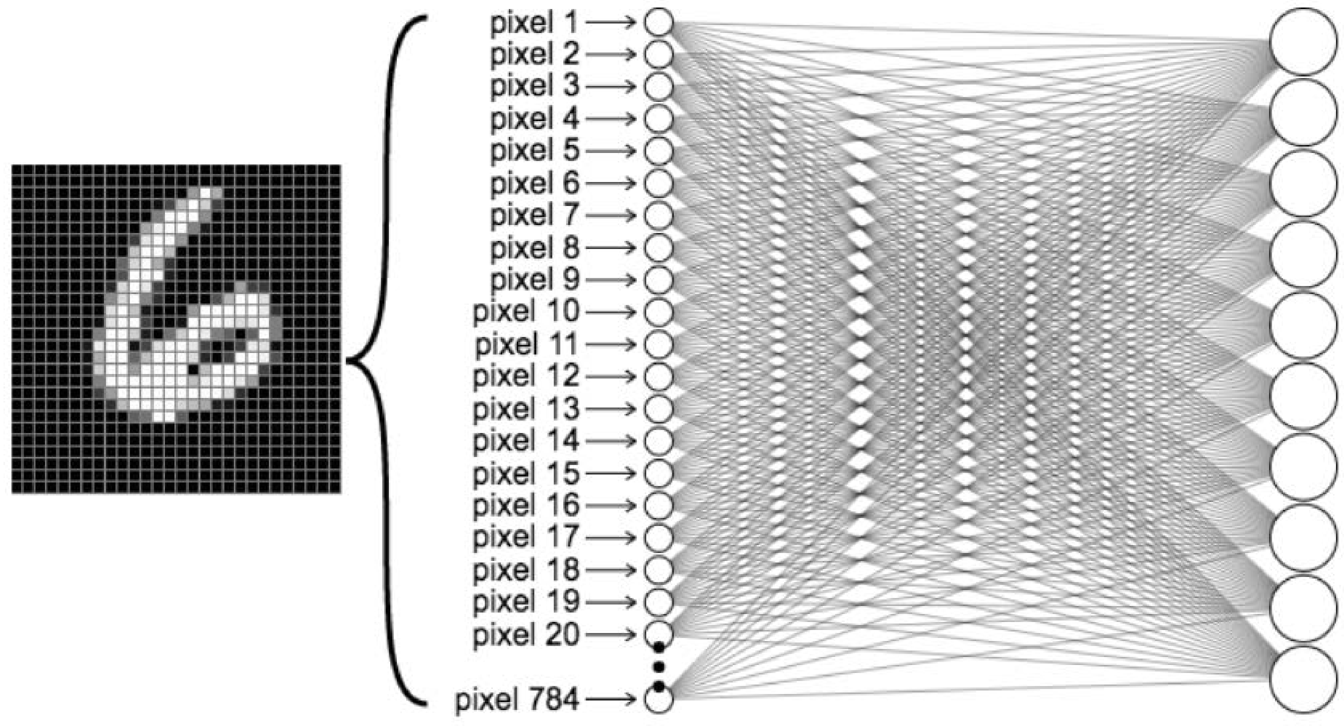
Feed Forward Neural Networks





MNIST handwritten digit dataset

Feed Forward Neural Networks

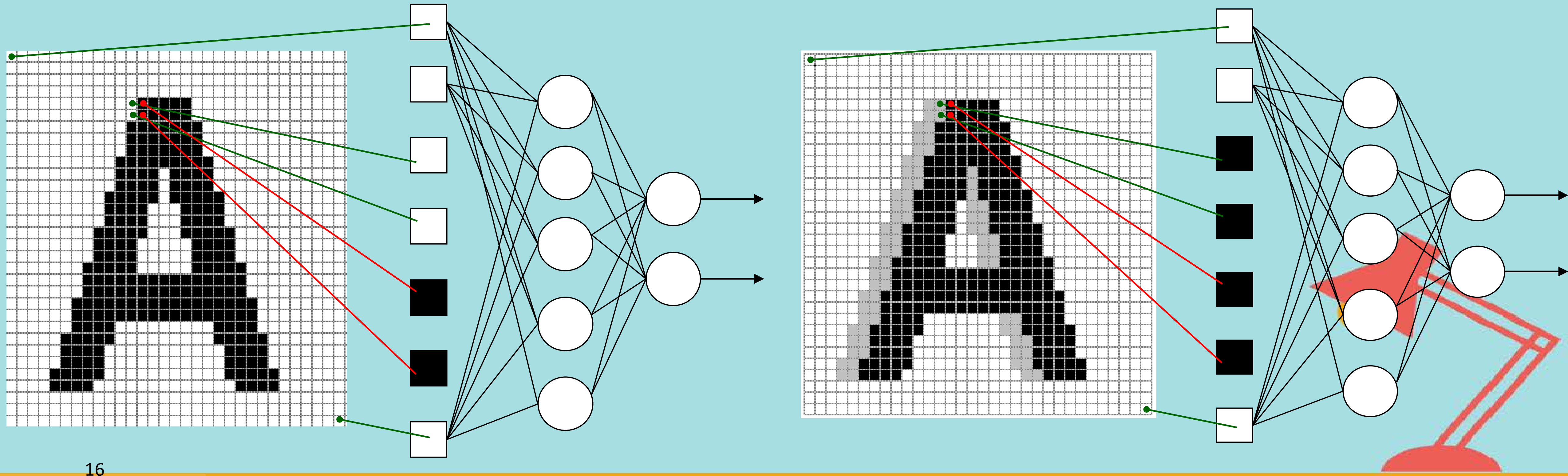


NN with input and output layer, hidden layer is hidden :)

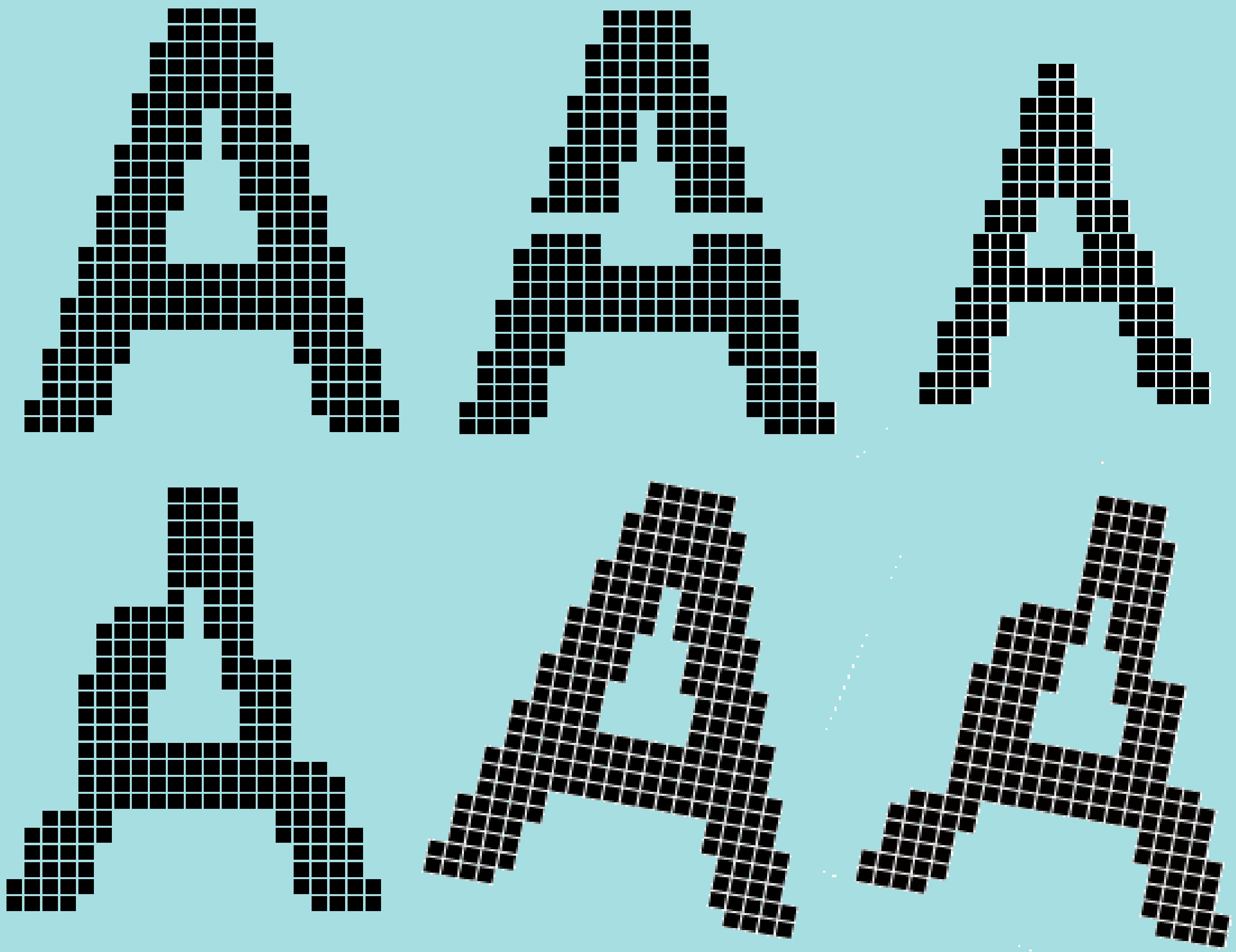
```
model = keras.models.Sequential([
  keras.layers.Flatten(input_shape=(28, 28)),
  keras.layers.Dense(128, activation='relu'),
  keras.layers.Dropout(0.2),
  keras.layers.Dense(10, activation='softmax')
])

model.compile(optimizer='adam',
              loss='sparse_categorical_crossentropy',
              metrics=['accuracy'])

model.fit(x_train, y_train, epochs=5)

model.evaluate(x_test, y_test)
```

Drawbacks of Feed Forward neural networks

Little or no invariance to shifting, scaling, and other forms of distortion



Scaling, and other forms of distortion

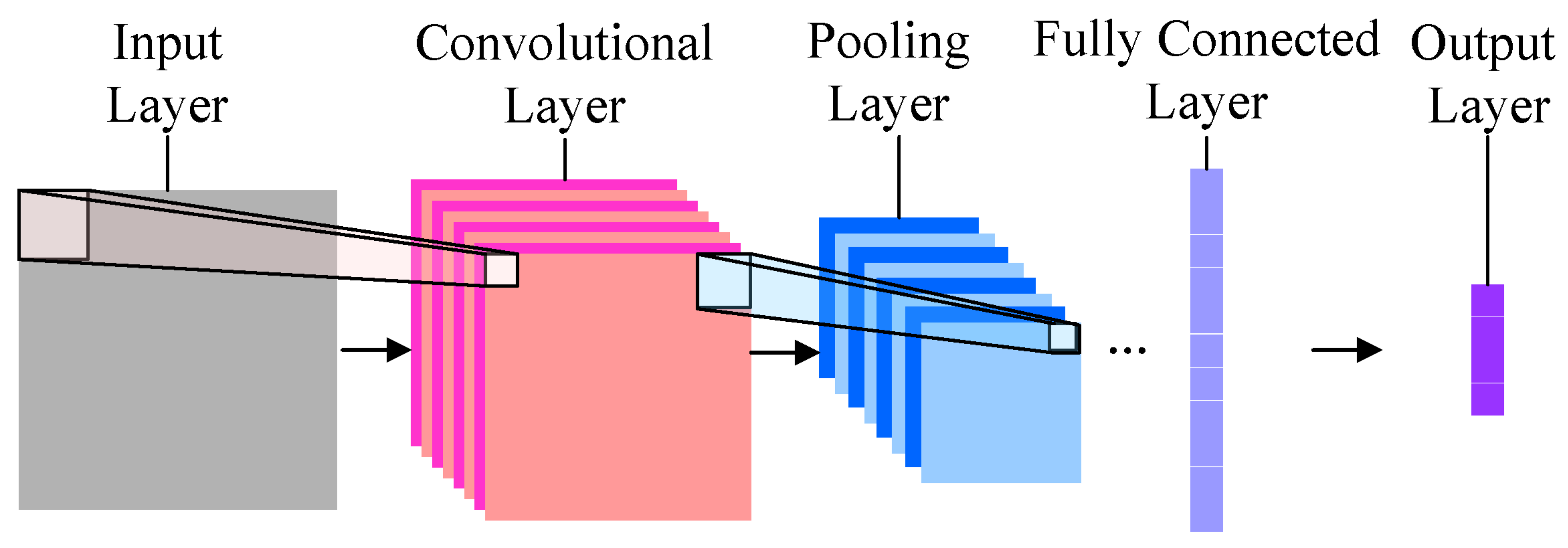


CNN History

Yann LeCun, Professor of Computer Science
The Courant Institute of Mathematical Sciences
New York University
Room 1220, 715 Broadway, New York, NY 10003, USA.
(212)998-3283 yann@cs.nyu.edu

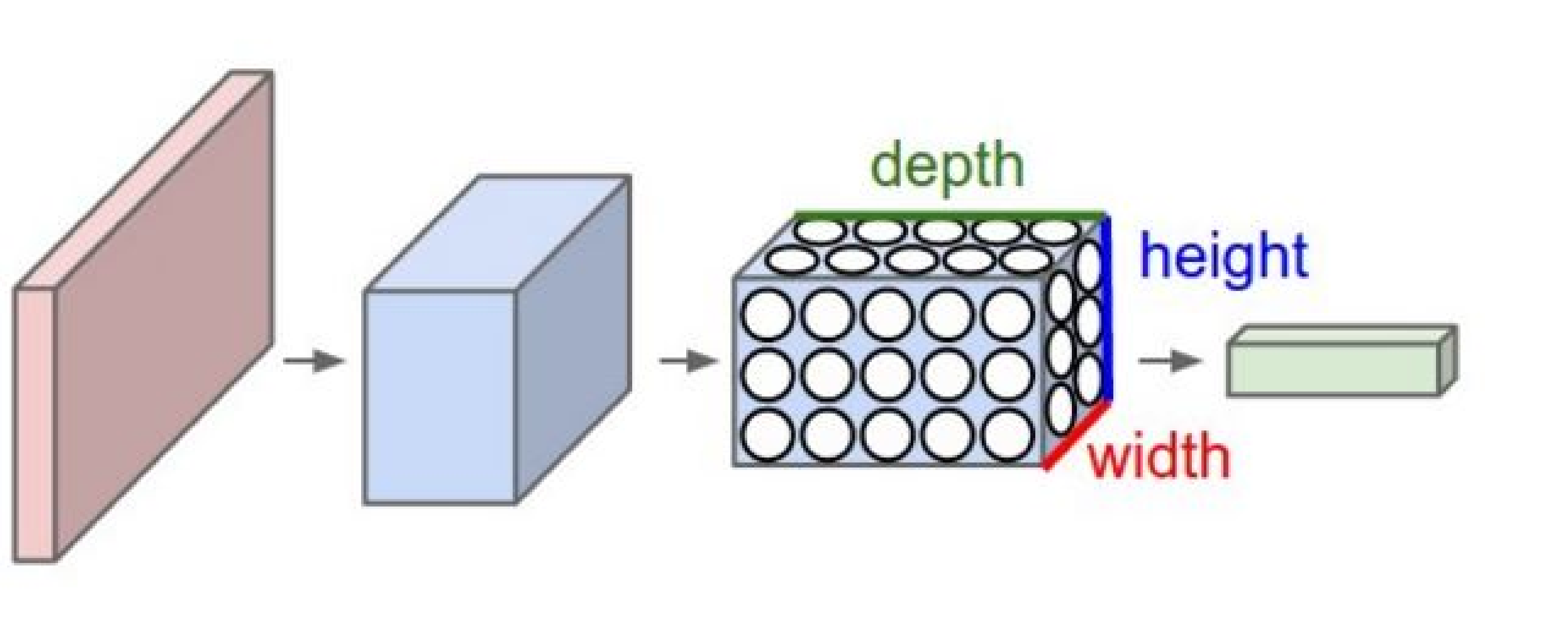
In 1995, Yann LeCun and Yoshua Bengio introduced the concept of convolutional neural networks.

CNN

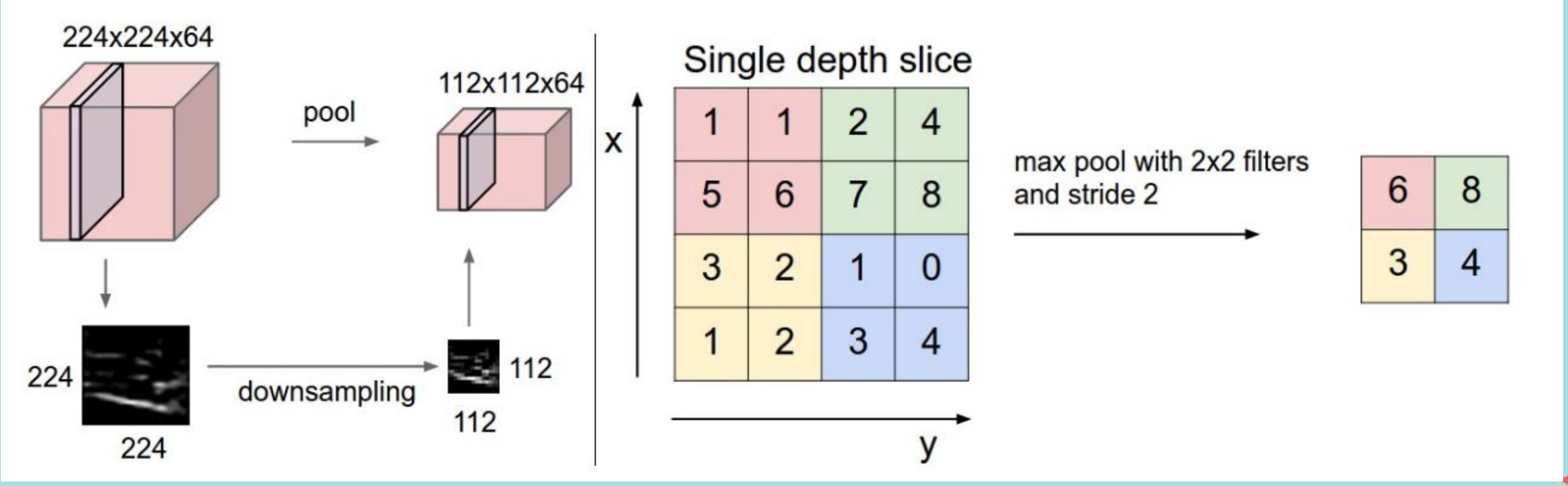


Input Layer
Convolutional Layer
Pooling Layer
Fully Connected Layer
Output Layer

Convolutional layer



Pooling layer



Color image: RGB 3 channels

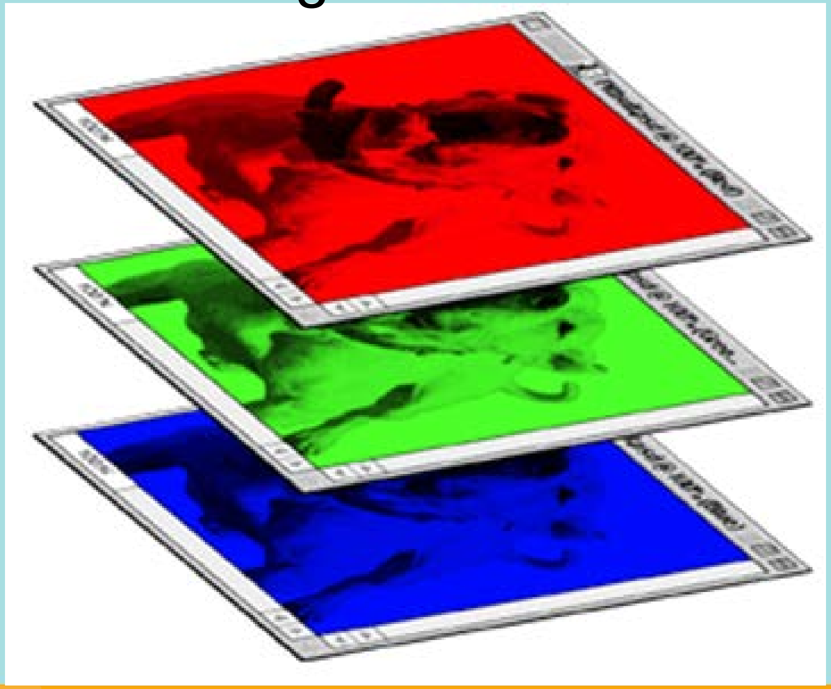
Filter 1

1	-1	-1
-1	1	-1
-1	-1	1

Filter 2

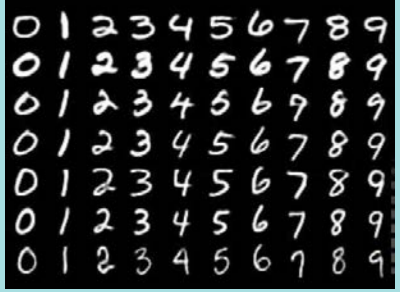
-1	1	-1
-1	1	-1
-1	1	-1

Color image



1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

The whole CNN



Convolution

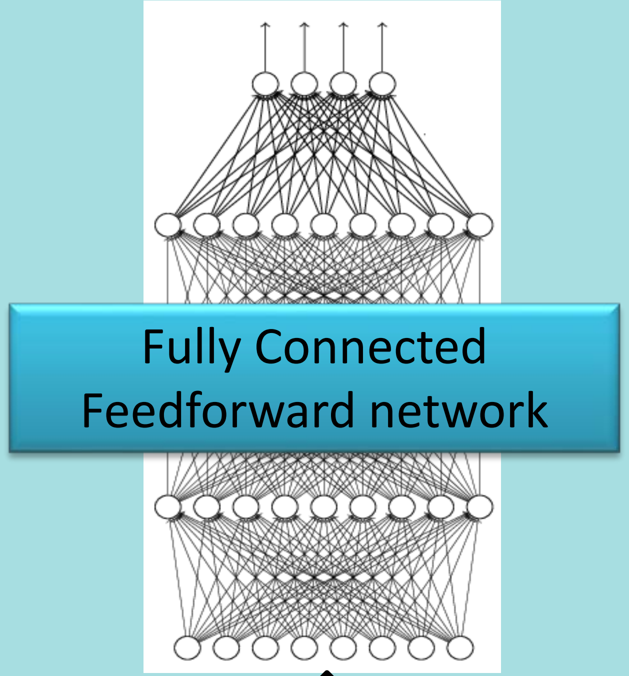
Max Pooling

Convolution

Max Pooling

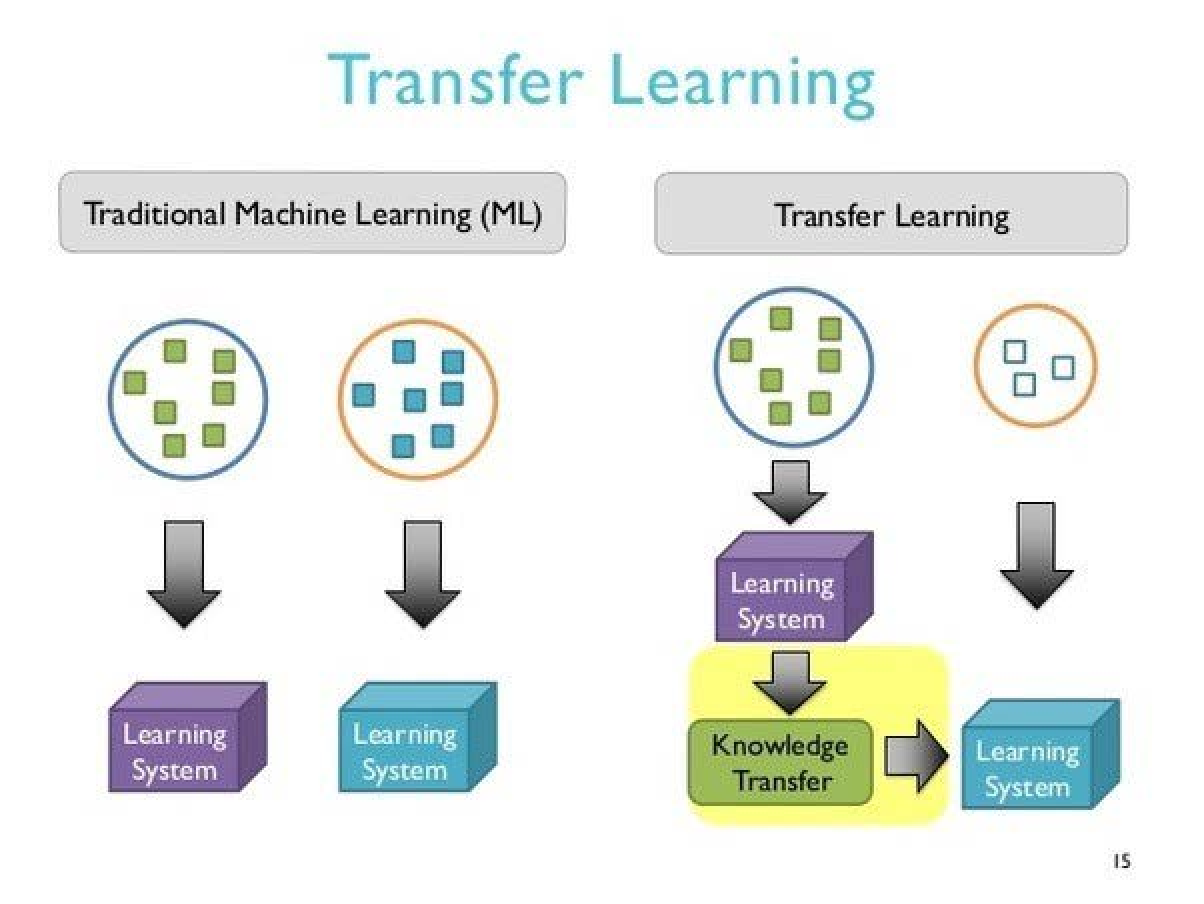
Can repeat many times

Flattened



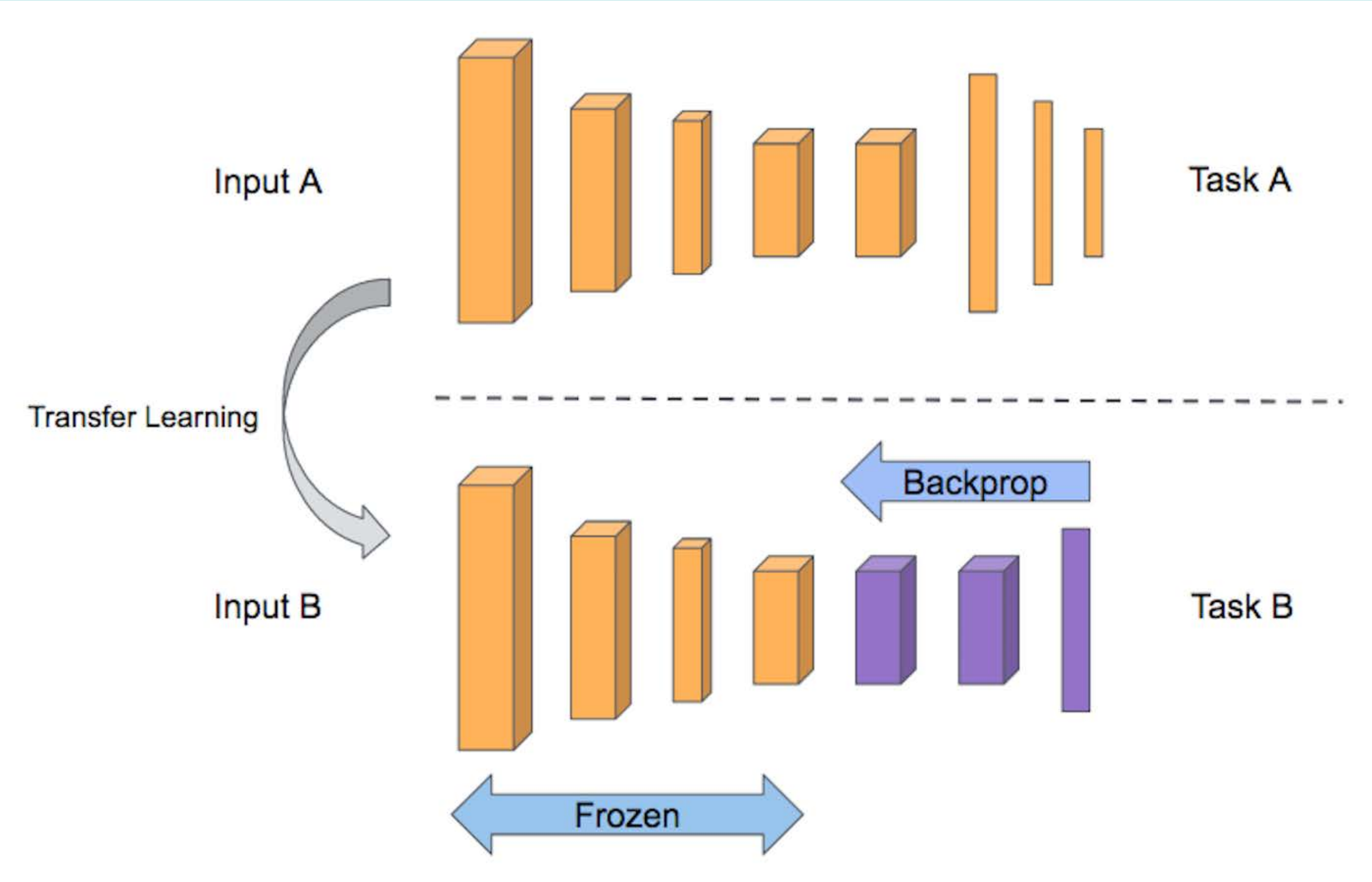
0, 1, 2, 3……

Concept of Transfer learning



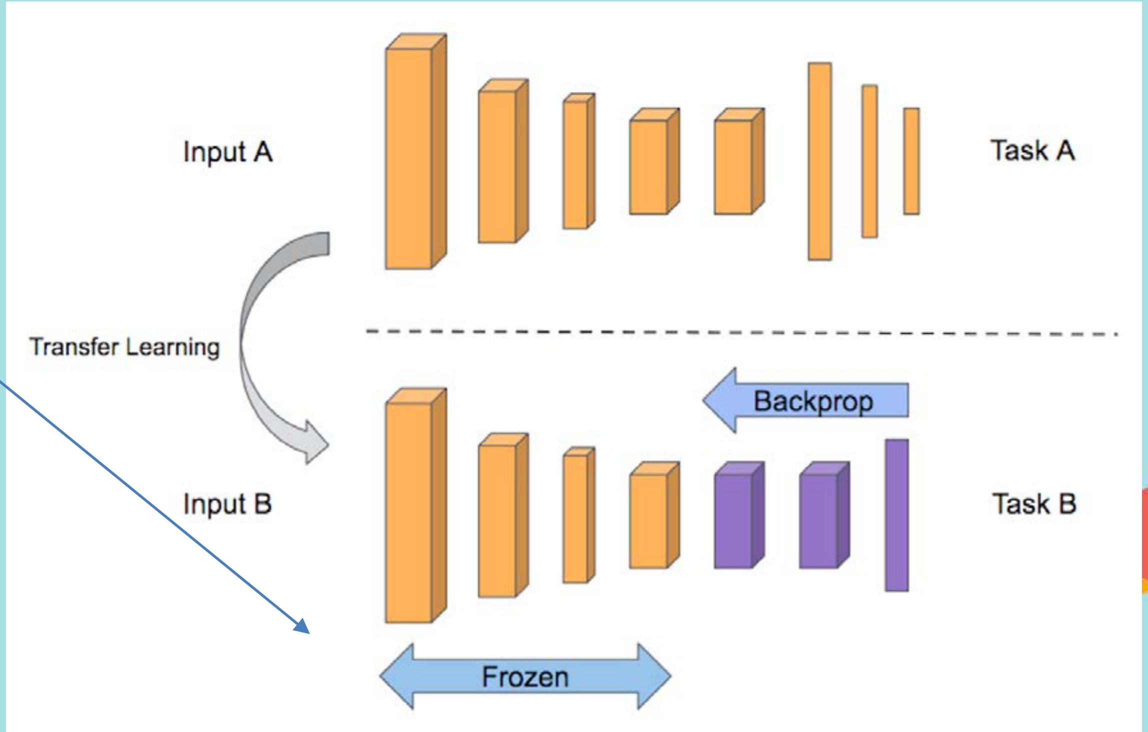
ImageNet CNN Models

Transfer learning model



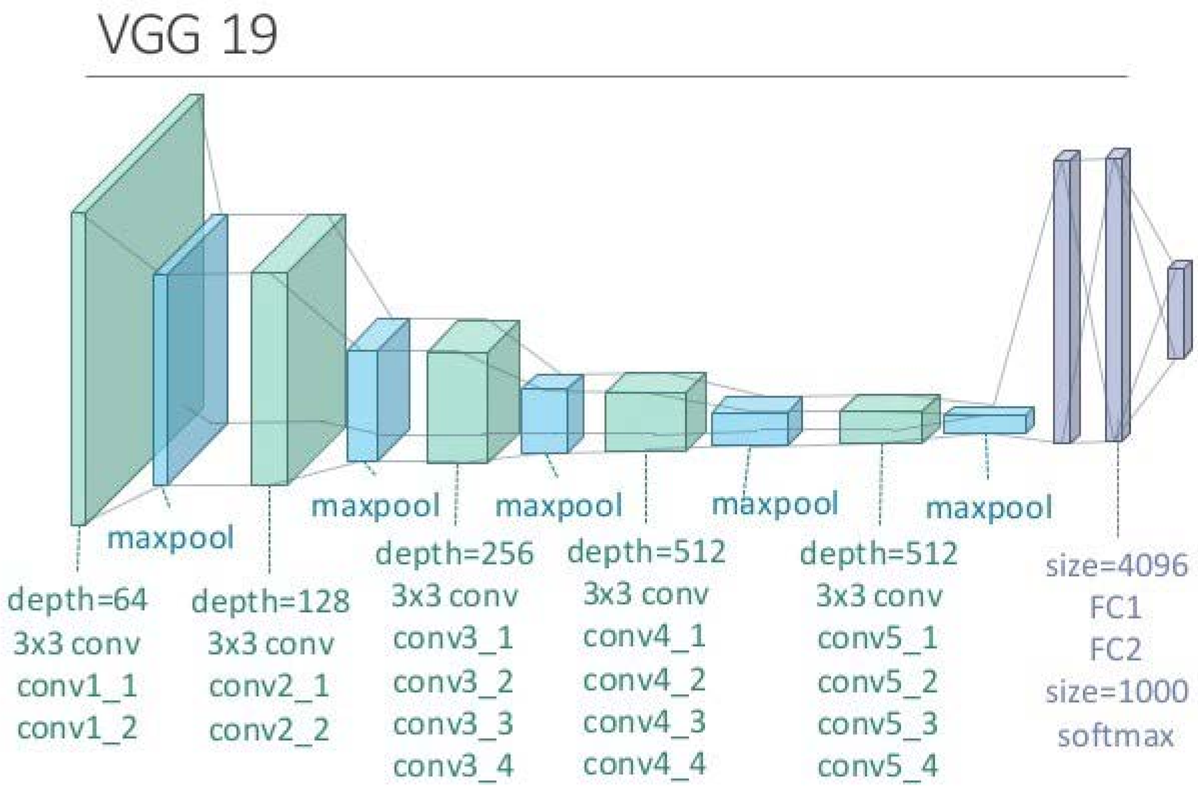
Frozed layers

- layer.trainable = False



CNN models

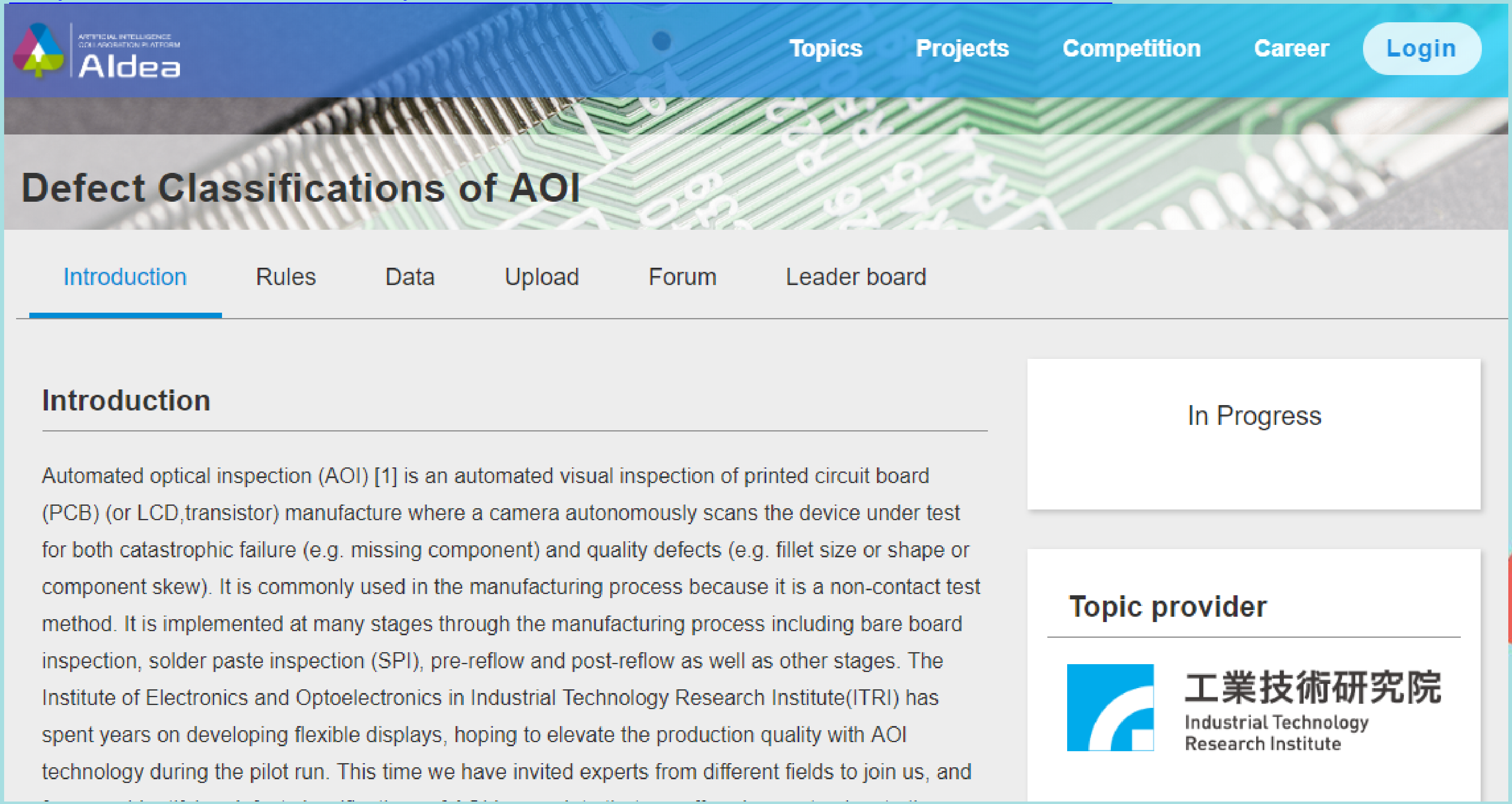
- VGG
- Resnet
- InceptionNet
- XceptionNet



03-Learning Deep Learning-based AOI

AIdea AOI project

https://aidea-web.tw/topic/a49e3f76-69c9-4a4a-bcfc-c882840b3f27



ARTIFICIAL INTELLIGENCE COLLABORATION PLATFORM
AIdea

Topics Projects Competition Career Login

Defect Classifications of AOI

Introduction Rules Data Upload Forum Leader board

Introduction

Automated optical inspection (AOI) [1] is an automated visual inspection of printed circuit board (PCB) (or LCD,transistor) manufacture where a camera autonomously scans the device under test for both catastrophic failure (e.g. missing component) and quality defects (e.g. fillet size or shape or component skew). It is commonly used in the manufacturing process because it is a non-contact test method. It is implemented at many stages through the manufacturing process including bare board inspection, solder paste inspection (SPI), pre-reflow and post-reflow as well as other stages. The Institute of Electronics and Optoelectronics in Industrial Technology Research Institute(ITRI) has spent years on developing flexible displays, hoping to elevate the production quality with AOI technology during the pilot run. This time we have invited experts from different fields to join us, and

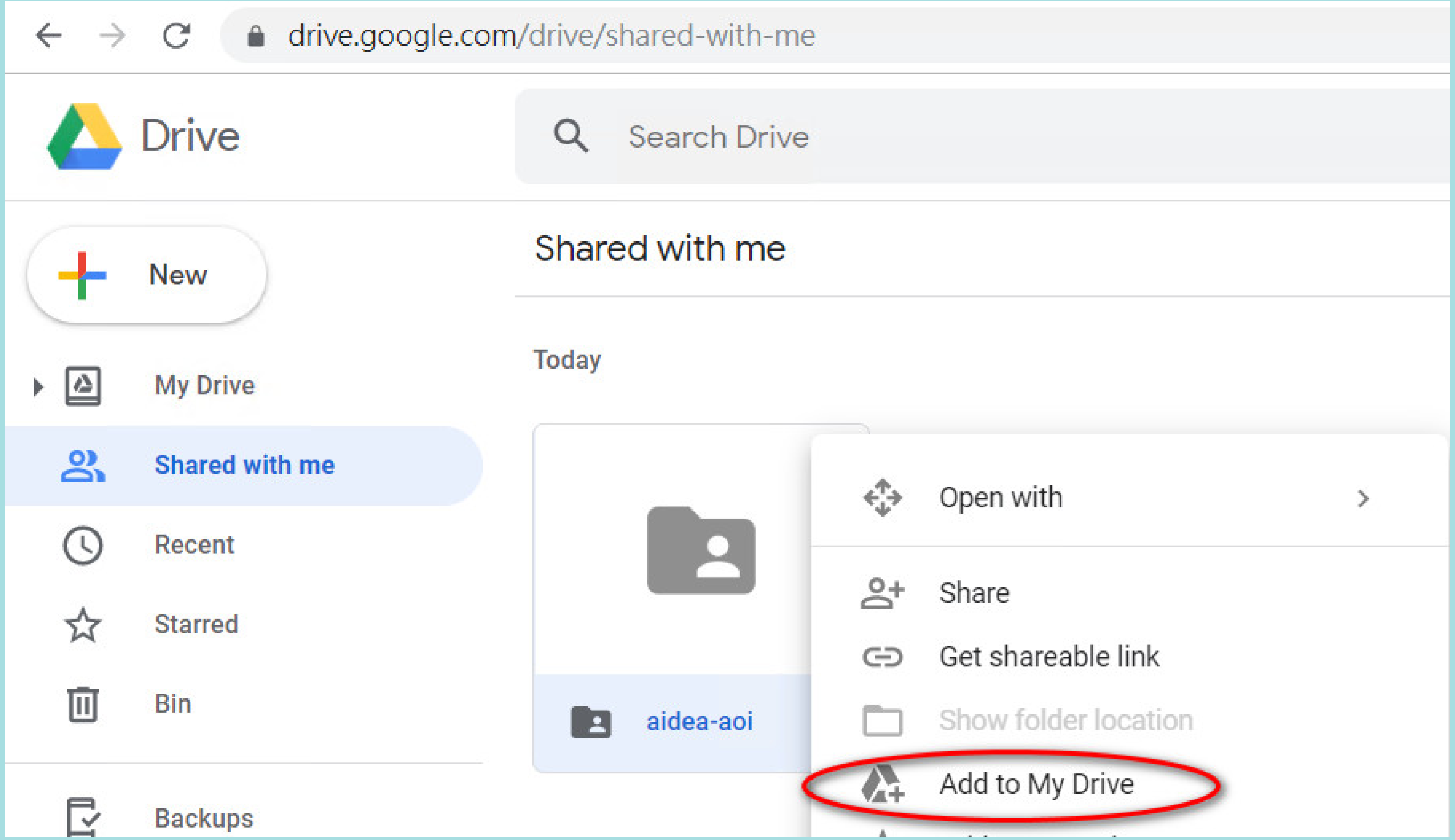
In Progress

Topic provider

工業技術研究院
Industrial Technology
Research Institute

Google Drive

Click https://drive.google.com/open?id=15tGIHAPAatgdB8iZh_m80jCBPa-CrI_P



Google Colab

(2) Transfer learning

AOI-Tutorial-1-20200114.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

AOI Course (part 1) by Hsueh-Ting Chu, Asia University, Jan., 2020

- 這個教程使用工研院AIdea人工智慧共創平台的AOI資料集做為練習的標的。
- 介紹撰寫深度學習的程式來進行自動光學檢查的瑕疵分類。
- 本notebook程式可以在雲端使用Google Colab或使用個人電腦上的Jupyter執行。

AIdea人工智慧共創平台 https://aidea-web.tw/topic/76f9ec46-cb90-4aa8-82f2-ebfed54cecfb

朱學亭老師

- EMAIL: htchu.taiwan@gmail.com
- FB: https://www.facebook.com/htchu.taiwan

Tutorial 1: training a full CNN model for AOI

(1) Training a full model

AOI-Tutorial-2-20200114.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

AOI Course (part 2) by Hsueh-Ting Chu, Asia University, Jan., 2020

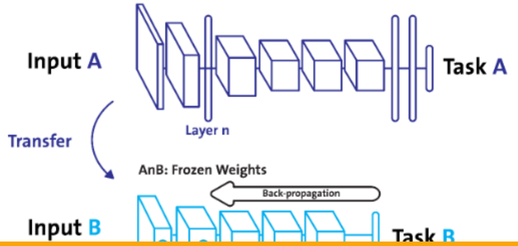
- 這個教程使用工研院AIdea人工智慧共創平台的AOI資料集做為練習的標的。
- 介紹撰寫深度學習的程式來進行自動光學檢查的瑕疵分類。
- 本notebook程式可以在雲端使用Google Colab或使用個人電腦上的Jupyter執行。

AIdea人工智慧共創平台 https://aidea-web.tw/topic/76f9ec46-cb90-4aa8-82f2-ebfed54cecfb

朱學亭老師

- EMAIL: htchu.taiwan@gmail.com
- FB: https://www.facebook.com/htchu.taiwan

Tutorial 2: Transfer a CNN model for AOI



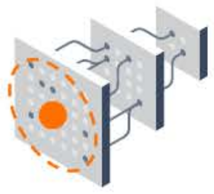
TensorFlow

TensorFlow   Install   Learn   API   Resources   Community   Why TensorFlow   Search   Language

Why TensorFlow

TensorFlow is an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML powered applications.

About →



Easy model building

Build and train ML models easily using intuitive high-level APIs like Keras with eager execution, which makes for immediate model iteration and easy debugging.

Robust ML production anywhere

Easily train and deploy models in the cloud, on-prem, in the browser, or on-device no matter what language you use.

Powerful experimentation for research

A simple and flexible architecture to take new ideas from concept to code, to state-of-the-art models, and to publication faster.

TensorFlow 2.x

Announcement of TensorFlow 2.0 September 30, 2019

TensorFlow | Install | Learn | API | Resources | Community | Why TensorFlow | Search

TensorFlow guide

TensorFlow 2
- Effective TensorFlow
- Migrate from TF1 to TF2
- Convert with the upgrade script
- Performance with tf.function
- Community testing FAQ

Keras
- Keras overview
- Keras functional API
- Train and evaluate
- Write custom layers and models
- Save and serialize models
- Keras Recurrent Neural Networks
- Masking and padding
- Write custom callbacks
- Mixed precision

Estimators
- Estimator overview

TensorFlow > Learn > TensorFlow Core > Guide

Effective TensorFlow 2

There are multiple changes in TensorFlow 2.0 to make TensorFlow users more productive. TensorFlow 2.0 removes redundant APIs, makes APIs more consistent (Unified RNNs, Unified Optimizers), and better integrates with the Python runtime with Eager execution.

Many RFCs have explained the changes that have gone into making TensorFlow 2.0. This guide presents a vision for what development in TensorFlow 2.0 should look like. It's assumed you have some familiarity with TensorFlow 1.x.

A brief summary of major changes

API Cleanup

Many APIs are either gone or moved in TF 2.0. Some of the major changes include removing `tf.app`, `tf.flags`, and `tf.logging` in favor of the now open-source absl-py, rehoming projects that lived in `tf.contrib`, and cleaning up the main `tf.*` namespace by moving lesser used functions into subpackages like `tf.math`. Some APIs have been replaced with their 2.0 equivalents - `tf.summary`, `tf.keras.metrics`, and `tf.keras.optimizers`. The easiest way to automatically apply these renames is to use the v2 upgrade script.

tf.Keras Layers

- Core Layers
  - **Dense**
  - **Activation**
  - **Dropout**
  - **Flatten**
  - **Input**
  - **Reshape**
- Convolutional Layers
  - **Conv2D**
  - **ZeroPadding2D**
- Pooling Layers
  - **MaxPooling2D**

- Recurrent Layers
  - **RNN**
  - **SimpleRNN**
  - **GRU**
  - **LSTM**
- Embedding Layers
  - **Embedding**
- Layer wrappers
  - **Bidirectional**

Thanks!

Q&A