# Computer Vision

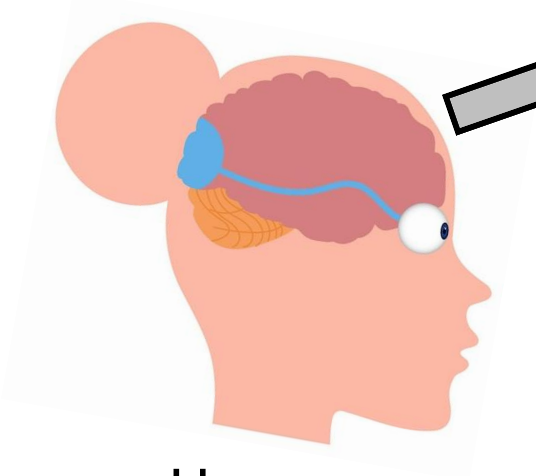
Dr. Rebeka Sultana
Specially appointed assistant professor
Tokyo University of Agriculture and Technology
Email: sultana@go.tuat.ac.jp

Women in Science Japan

July 26th, 2025, Tokyo

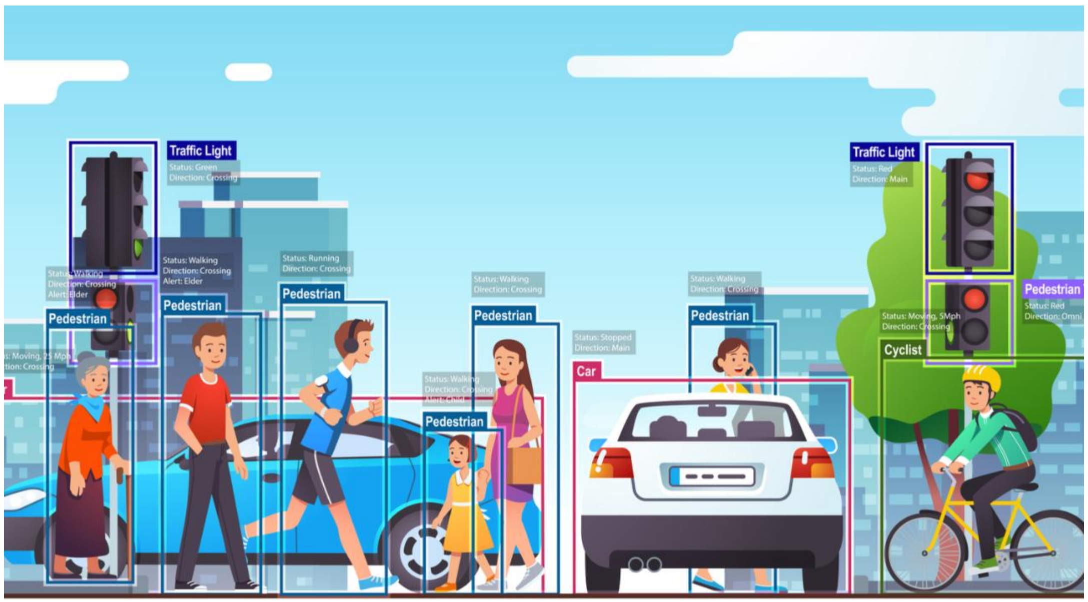
# Contents

- Looking at picture and visual interpretation

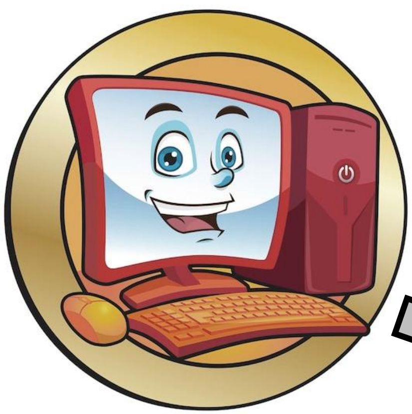


Complex functions in brain

Human




Picture



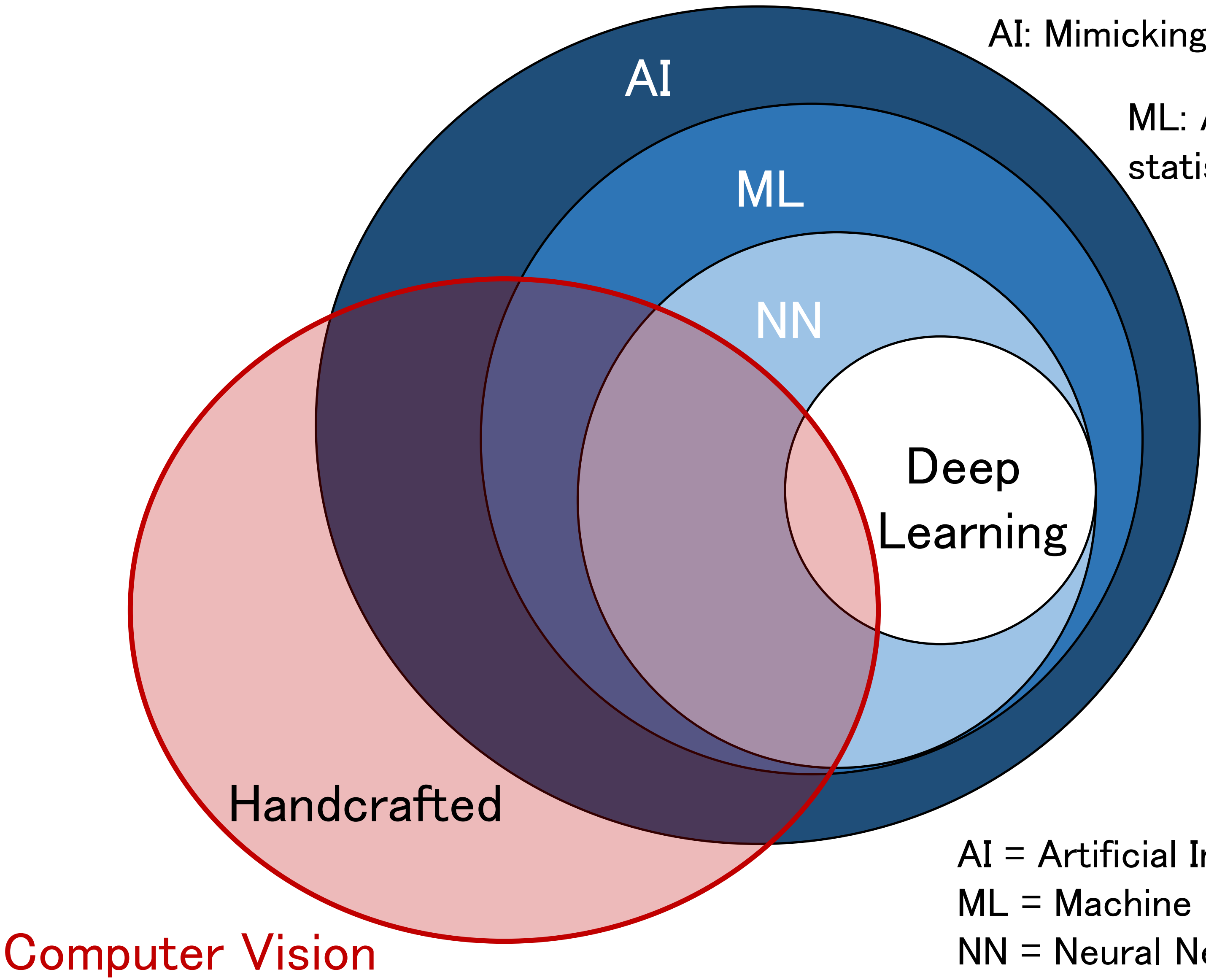
Machine

Various image processing algorithms

Computer Vision

Image source: https://www.augmentedstartups.com/blog/8-real-world-applications-of-computer-vision-a-comprehensive-guide

❑ Algorithms




AI: Mimicking human behavior

ML: Automatic learning by statistical techniques

DL: Automatic learning by neural network

AI = Artificial Intelligence
ML = Machine Learning
NN = Neural Network

# Contents

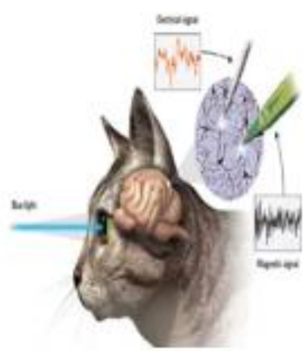
❑Years of groundbreaking works in computer vision

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1700S – 1900S | 1957 | 1962 | 1963 | 1966 | 1967 | 1968 | 1972 | 1973 | 1980 | 1981 | 1983 | 1986 | 1987 | 1988 | 1989 | 1991 | 1993 | 1999 | 2000 | 2001 | 2004 | 2005 | 2006 | 2007 | 2009 | 2010 | 2012 | 2014 | 2015 | 2017 | 2018 | 2019 | 2020 - present |

**1700S–1900S**

Early development in field of light and vision

1962



Research on visual cortex

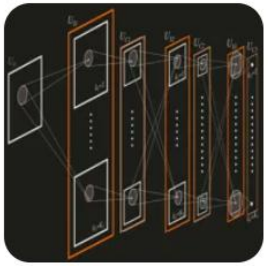
1966

Summer vision project

1972

Hough Transform

**1980**

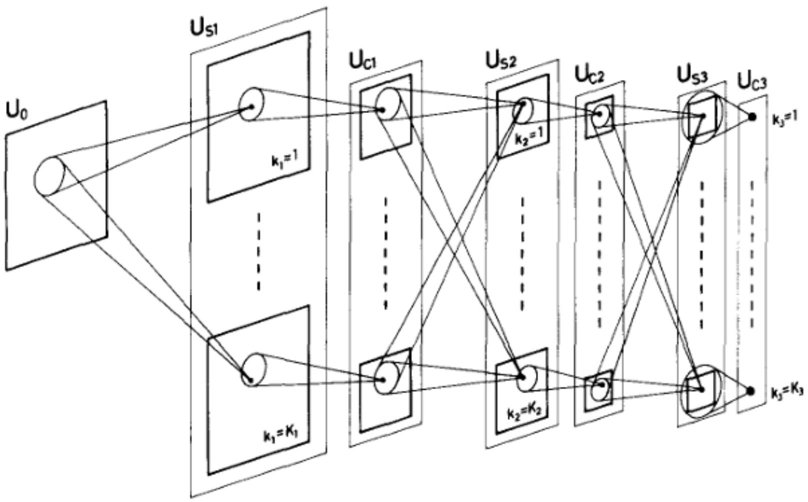


NeoCognitron

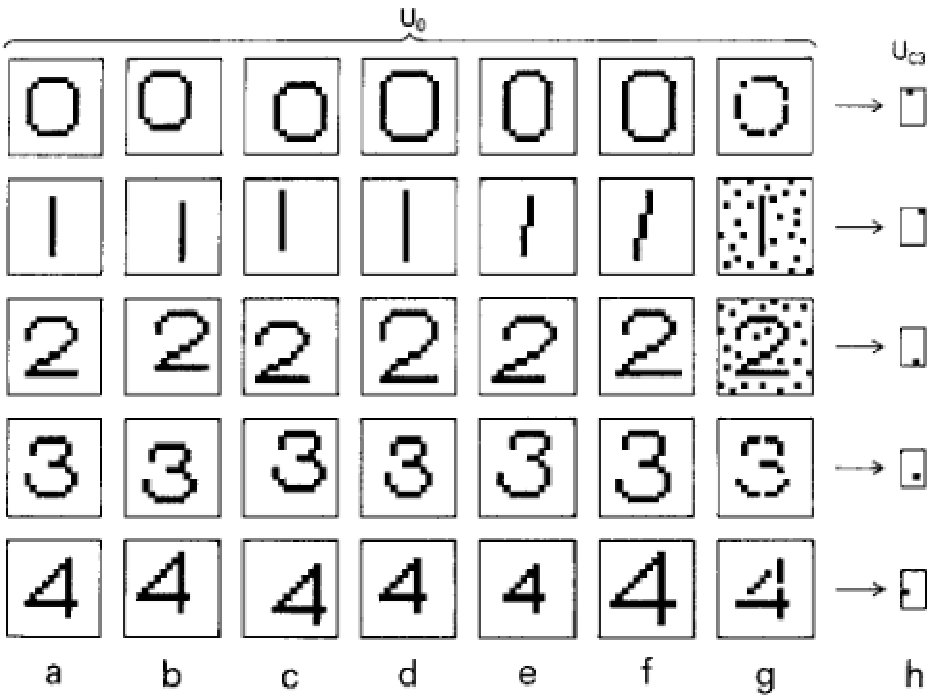
Source: https://letsdatascience.com/learn/history/history-of-computer-vision/

❑NeoCognitron

A mechanism of visual pattern recognition by neural network



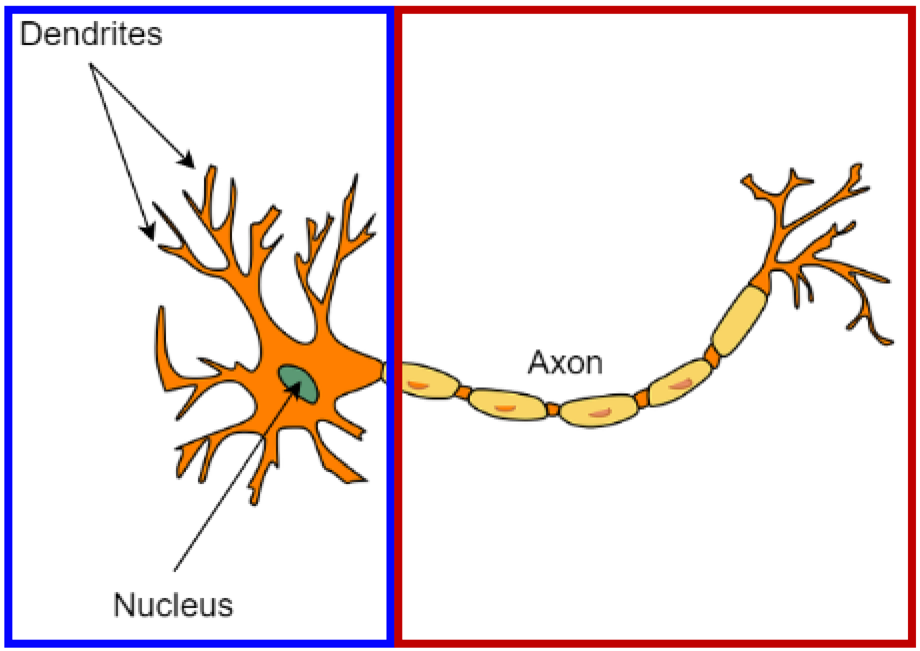

Network architecture



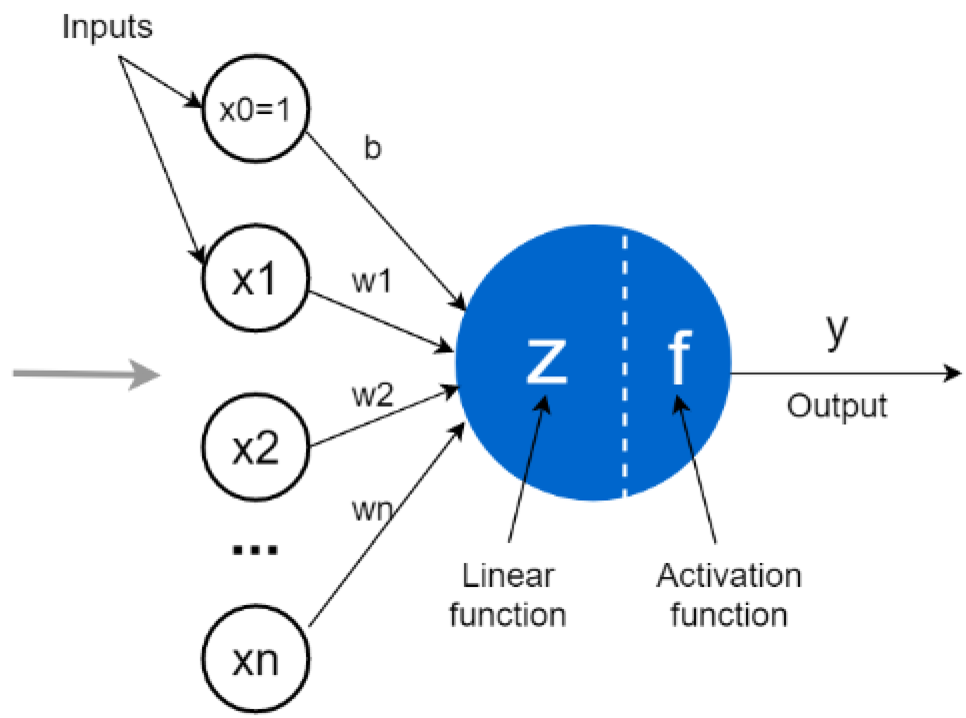

Example of correctly recognized pattern

Image source: Kunihiko Fukushima, "Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position", Biol. Cybernetics 36, 193 202 (1980)

- Concept of neuron (perceptron)




Linear function    Activation and output




Neuron in living body

Neuron in neural network

Image source: https://towardsdatascience.com/the-concept-of-artificial-neurons-perceptrons-in-neural-networks-fab22249cbfc/

# History



❑Years of groundbreaking works in computer vision

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1700S – 1900S | 1957 | 1962 | 1963 | 1966 | 1967 | 1968 | 1972 | 1973 | 1980 | 1981 | 1983 | 1986 | 1987 | 1988 | 1989 | 1991 | 1993 | 1999 | 2000 | 2001 | 2004 | 2005 | 2006 | 2007 | 2009 | 2010 | 2012 | 2014 | 2015 | 2017 | 2018 | 2019 | 2020 - present |

1981

Lucas–Kanade Optical Flow algorithm

1983


Vision By David Marr

1986

3D shape recovery from images

1991

Eigenfaces algorithm

1999

SIFT

2000

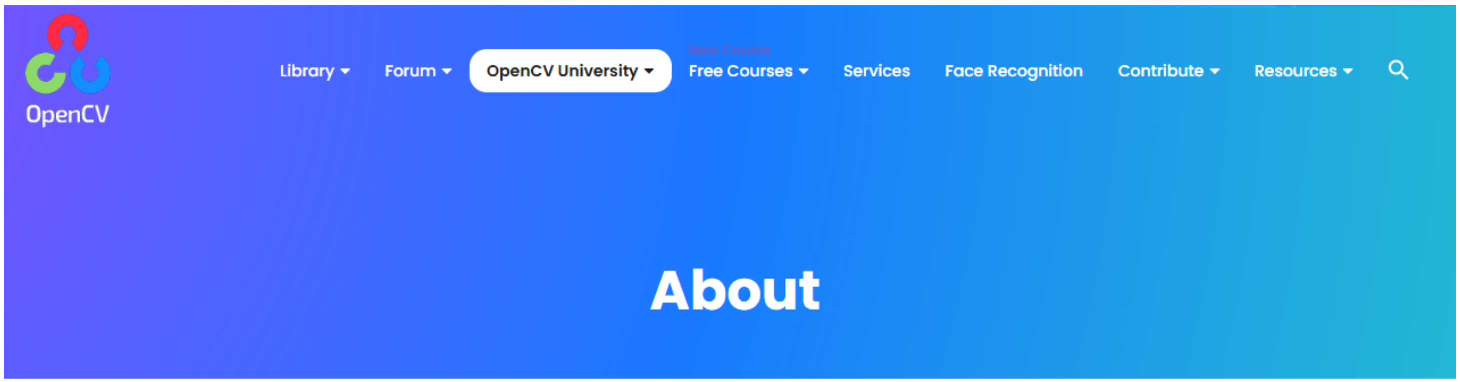
OpenCV

OpenCV

Source: https://letsdatascience.com/learn/history/history-of-computer-vision/

## ❑OpenCV

Open-source computer vision software contains 2500 optimized algorithms




OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being an Apache 2 licensed product, OpenCV makes it easy for businesses to utilize and modify the code.

The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces, identify objects, classify human actions in videos, track camera movements, track moving objects, extract 3D models of objects, produce 3D point clouds from stereo cameras, stitch images together to produce a high resolution image of an entire scene, find similar images from an image database, remove red eyes from images taken using flash, follow eye movements, recognize scenery and establish markers to overlay it with augmented reality, etc. OpenCV has more than 47 thousand people of user community and estimated number of downloads exceeding 18 million. The library is used extensively in companies, research groups and by governmental bodies.

Along with well-established companies like Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda, Toyota that employ the library, there are many startups such as Applied Minds, VideoSurf, and Zeitera, that make extensive use of OpenCV. OpenCV's deployed uses span the range from stitching streetview images together, detecting intrusions in surveillance video in Israel, monitoring mine equipment in China, helping robots navigate and pick up objects at Willow Garage, detection of swimming pool drowning accidents in Europe, running interactive art in Spain and New York, checking runways for debris in Turkey, inspecting labels on products in factories around the world on to rapid face detection in Japan.

It has C++, Python, Java and MATLAB interfaces and supports Windows, Linux, Android and Mac OS. OpenCV leans mostly towards real-time vision applications and takes advantage of MMX and SSE instructions when available. A full-featured CUDAand OpenCL interfaces are being actively developed right now. There are over 500 algorithms and about 10 times as many functions that compose or support those algorithms. OpenCV is written natively in C++ and has a templated interface that works seamlessly with STL containers.

Image source: https://opencv.org/about/

- Years of groundbreaking works in computer vision

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1700S – 1900S | 1957 | 1962 | 1963 | 1966 | 1967 | 1968 | 1972 | 1973 | 1980 | 1981 | 1983 | 1986 | 1987 | 1988 | 1989 | 1991 | 1993 | 1999 | 2000 | 2001 | 2004 | 2005 | 2006 | 2007 | 2009 | 2010 | 2012 | 2014 | 2015 | 2017 | 2018 | 2019 | 2020 - present |

| 2004 | 2012 | 2012 | 2014 | 2018 | 2020– |
|---|---|---|---|---|---|
| Voila-Jones Face Detection Model | ILSVRC | AlexNet | GAN's, VGG | YOLO v3 | GPT-3, Vision-Language Models |

**Dataset**

Deep learning models

Source: https://letsdatascience.com/learn/history/history-of-computer-vision/

## ❑ILSVRC: A pivotal benchmark in computer vision

IMAGENET

14,197,122 images, 21841 synsets indexed

Home Download Challenges About

Not logged in. Login | Signup

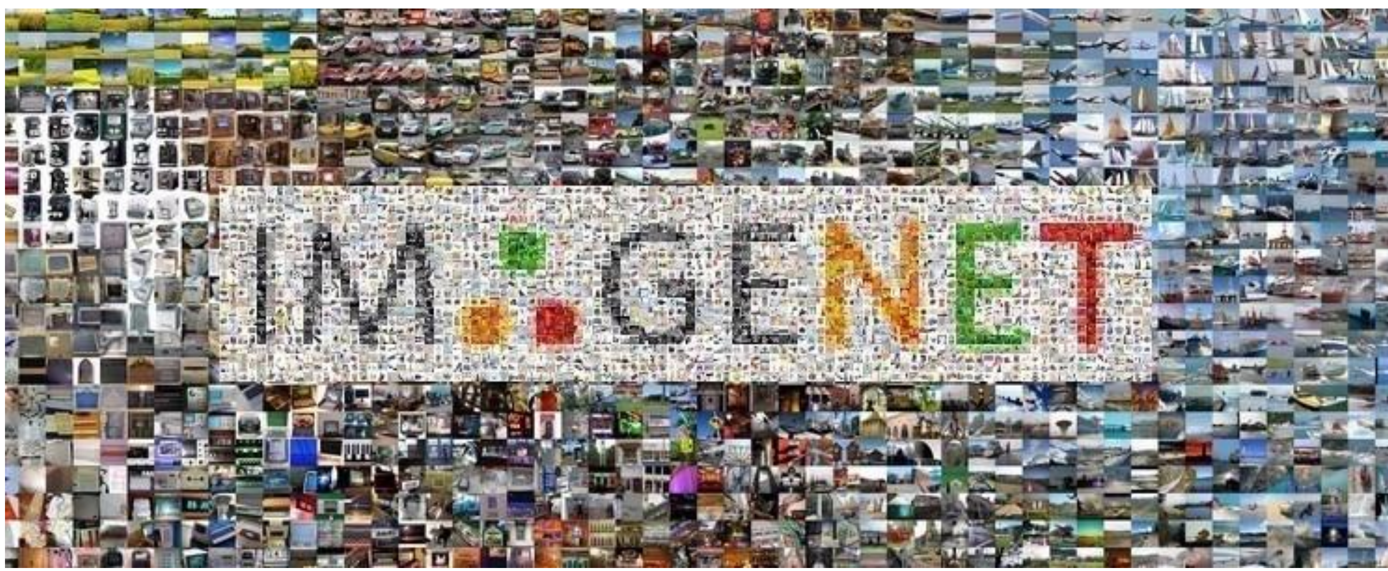
### ImageNet Large Scale Visual Recognition Challenge (ILSVRC)

#### Competition

The ImageNet Large Scale Visual Recognition Challenge (ILSVRC) evaluates algorithms for object detection and image classification at large scale. One high level motivation is to allow researchers to compare progress in detection across a wider variety of objects -- taking advantage of the quite expensive labeling effort. Another motivation is to measure the progress of computer vision for large scale image indexing for retrieval and annotation.

For details about each challenge please refer to the corresponding page.

- ILSVRC 2017
- ILSVRC 2016
- ILSVRC 2015
- ILSVRC 2014
- ILSVRC 2013
- ILSVRC 2012
- ILSVRC 2011
- ILSVRC 2010

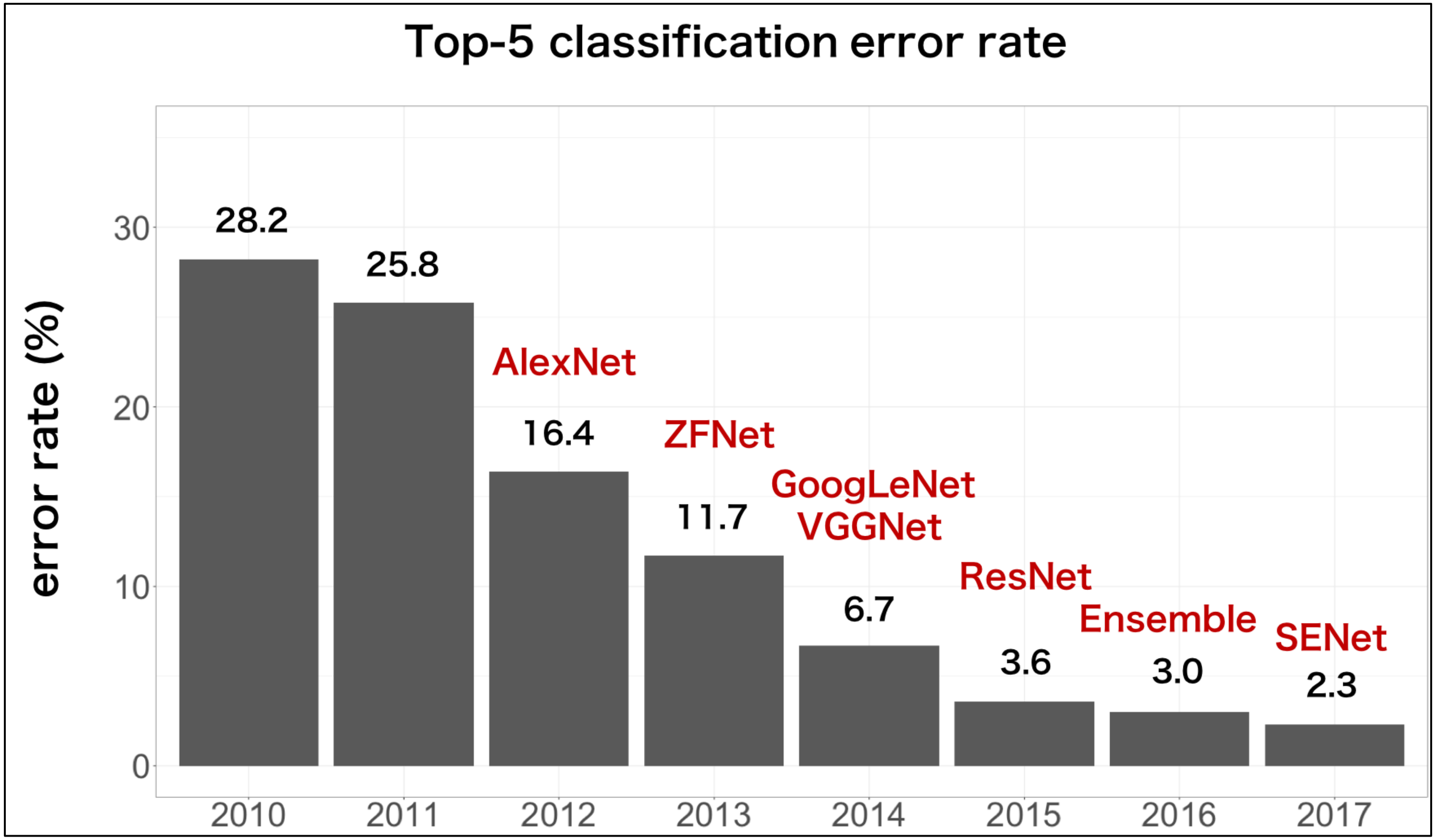


Sources:
https://www.image-net.org/challenges/LSVRC/index.php
https://deargen.blog/2016/12/23/deargen-won-at-ilsvrc-2016-imagenet-large-scale-visual-recognition-challenge/

# History



## ❑Performance of models on ILSVRC challenge




- **ImageNet dataset** is widely used for pre-training deep learning networks, particularly in **computer vision** tasks
- **Dataset plays a crucial roles in training deep learning models**

Image source: https://axa.biopapyrus.jp/media/ILSVRC_challenges.png

## ❑Timeline of Deep learning

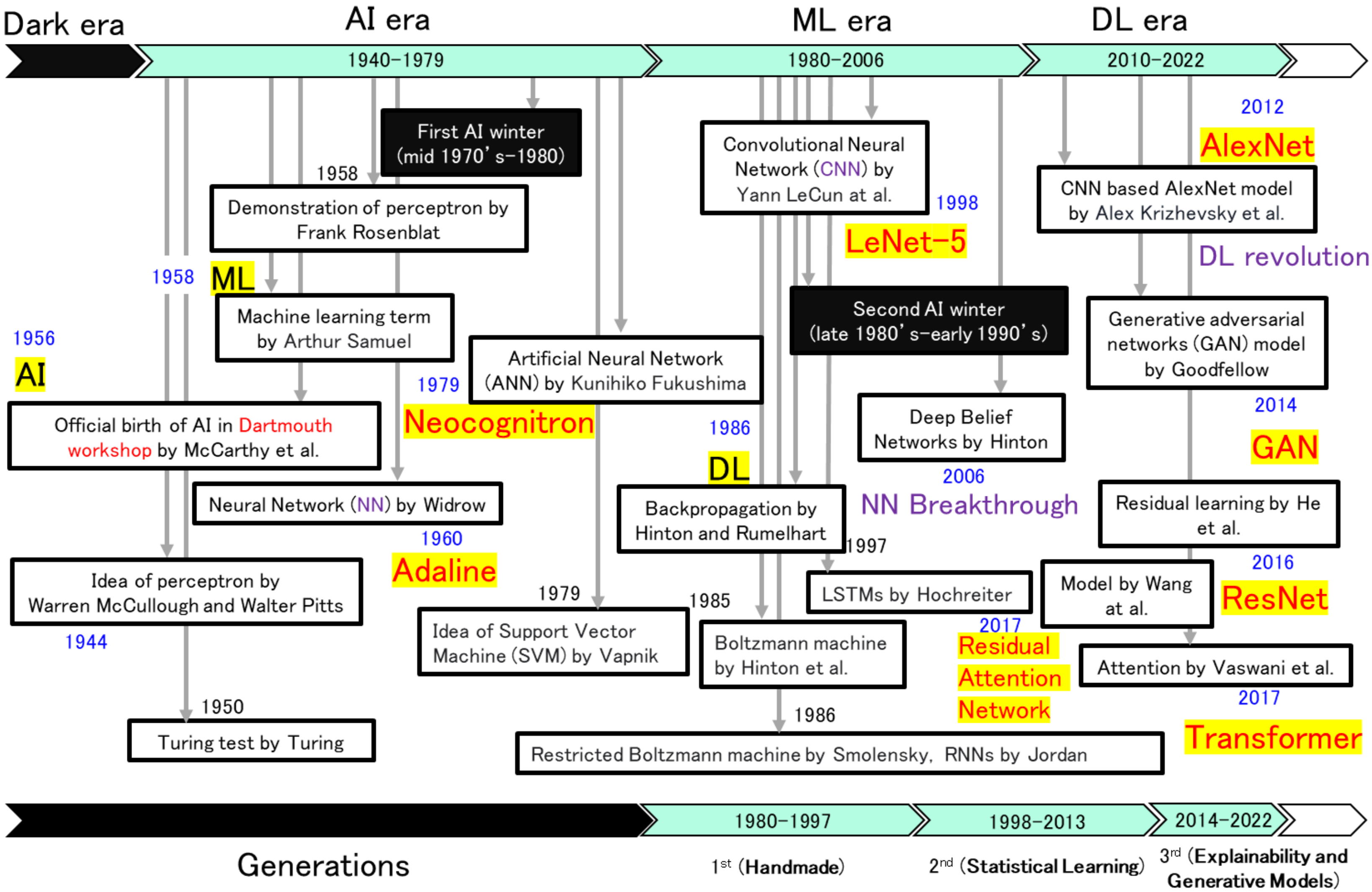



Image source: Rebeka Sultana, "**Deep Learning with Attention Mechanism for Driver's Visual Attention Prediction and Image Translation-based Data Augmentation**", PhD thesis, 2023.

# Contents

## ❑Popular computer vision tasks

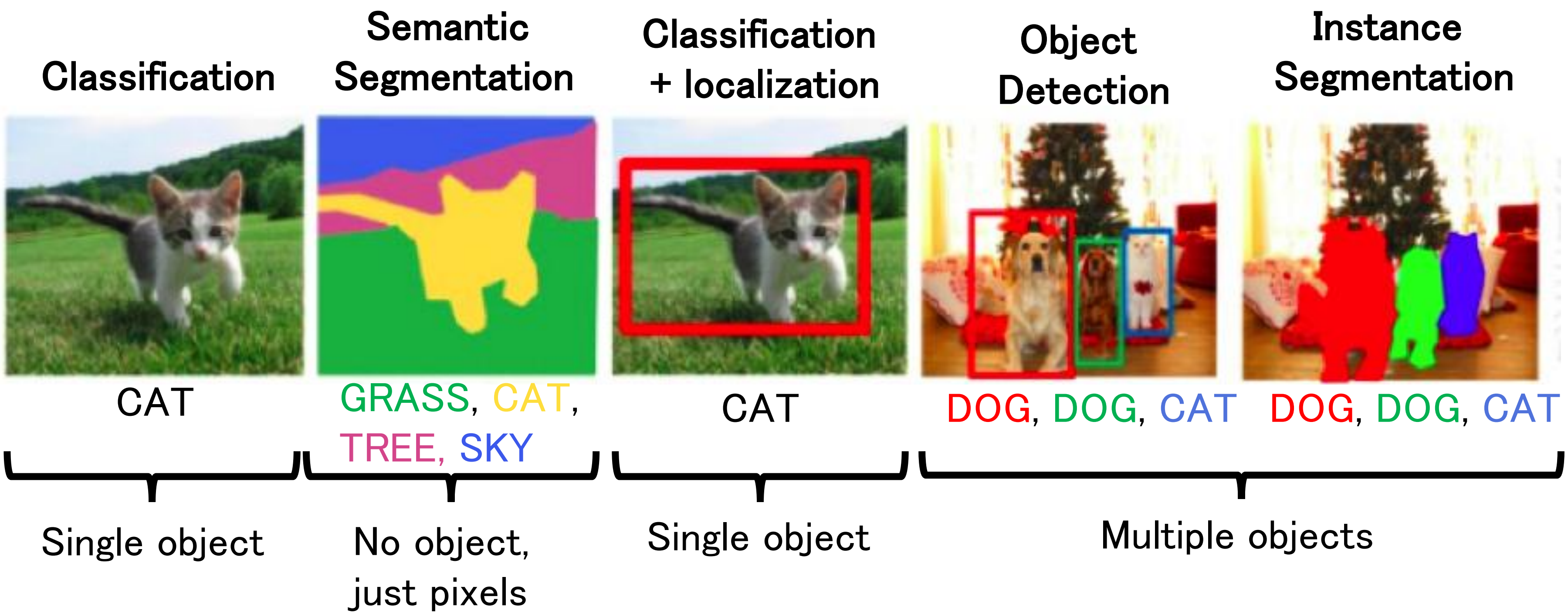


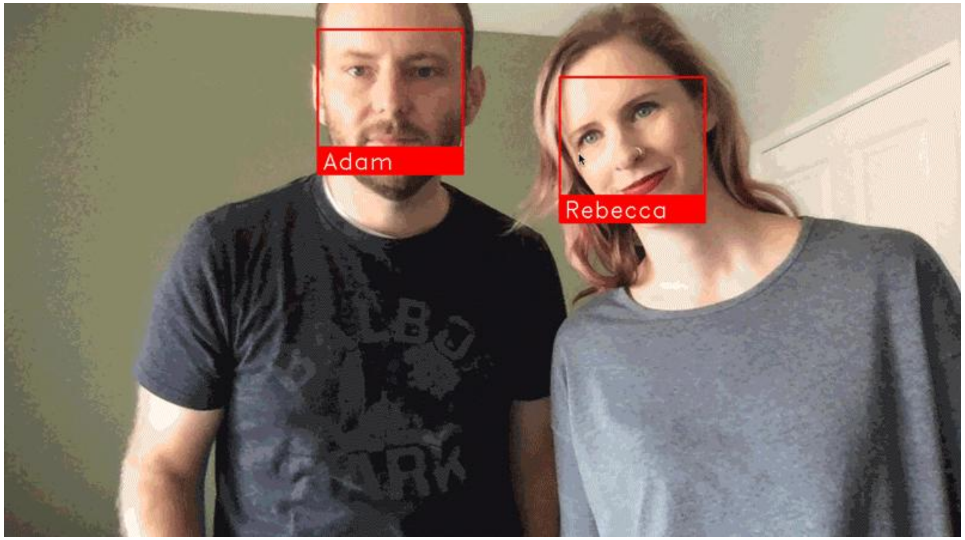

Image sources: https://www.superannotate.com/blog/image-classification-basics

# Tasks



## ❑Popular computer vision tasks

Face and person recognition




Video motion analysis


## ❑Others

- Edge detection
- Image restoration
- Matching scene reconstruction

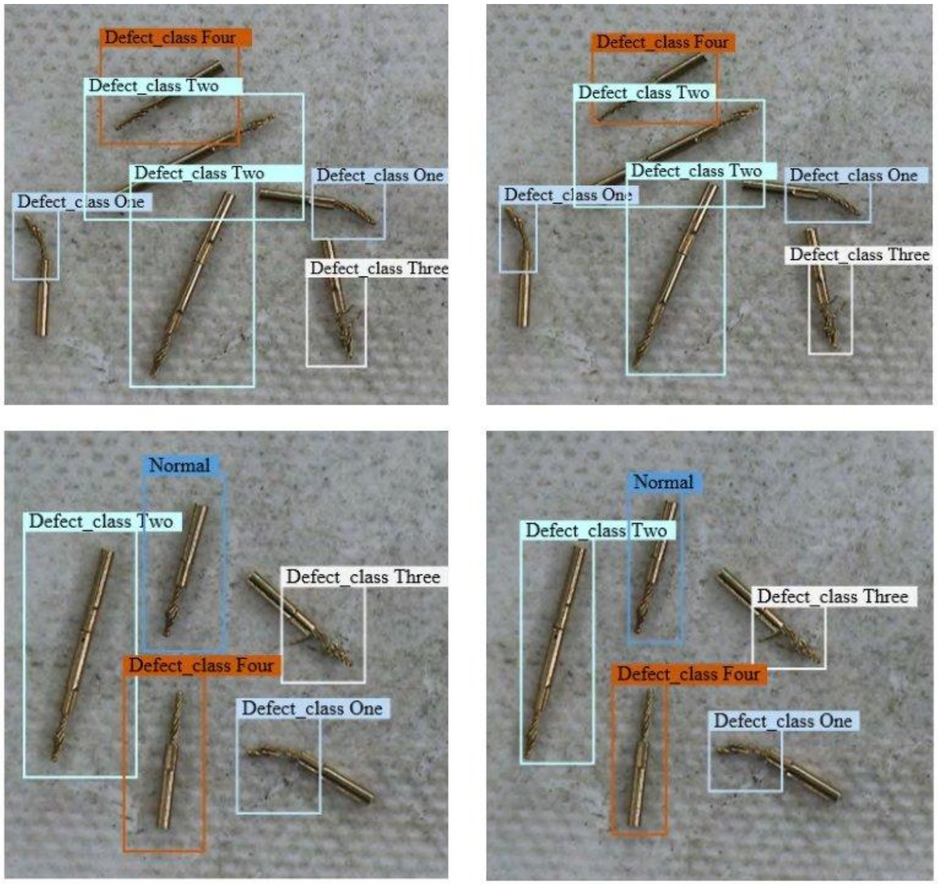
Image source: https://github.com/ageitgey/face_recognition
Video source: https://www.youtube.com/watch?v=oA_C040vLZs&t=1s&ab_channel=OpenCVAI

# Contents

# Applications



## Manufacture




## Agriculture

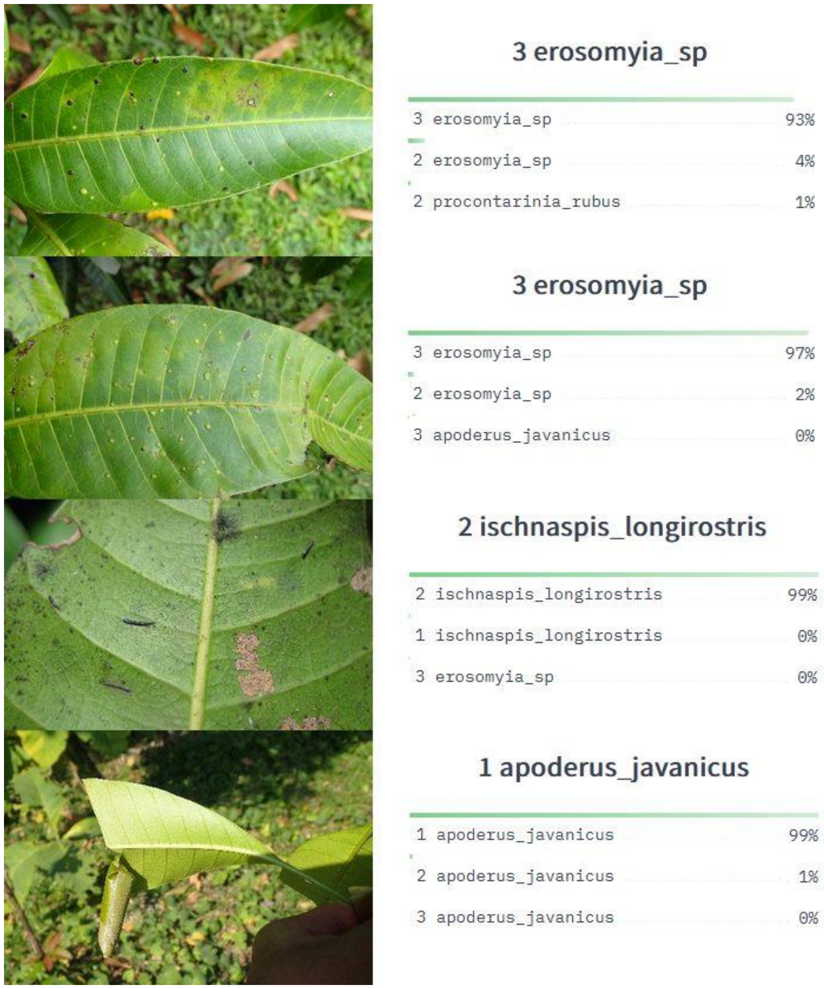
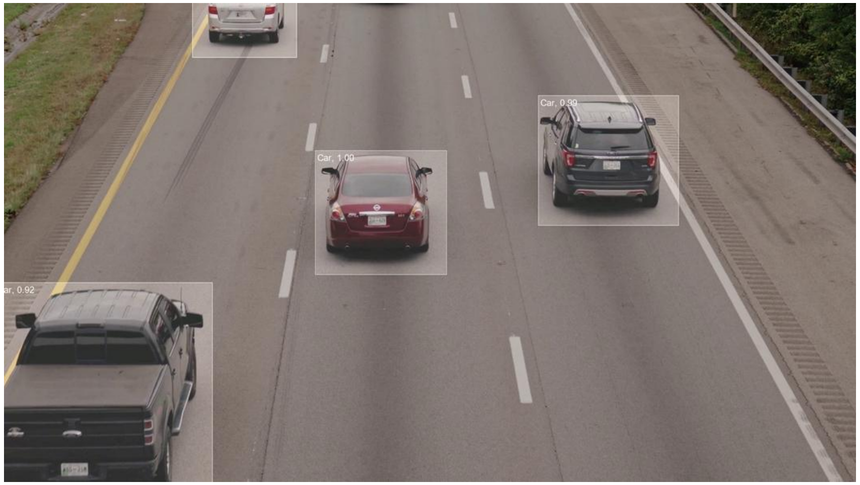



Image source: https://viso.ai/applications/computer-vision-applications/

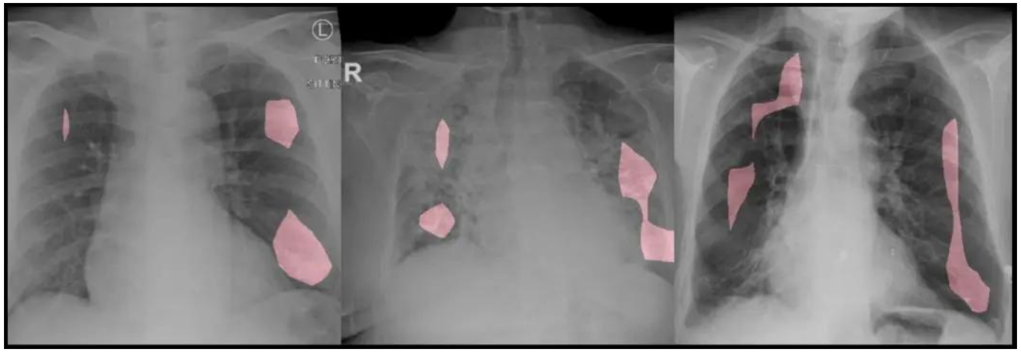
# Applications



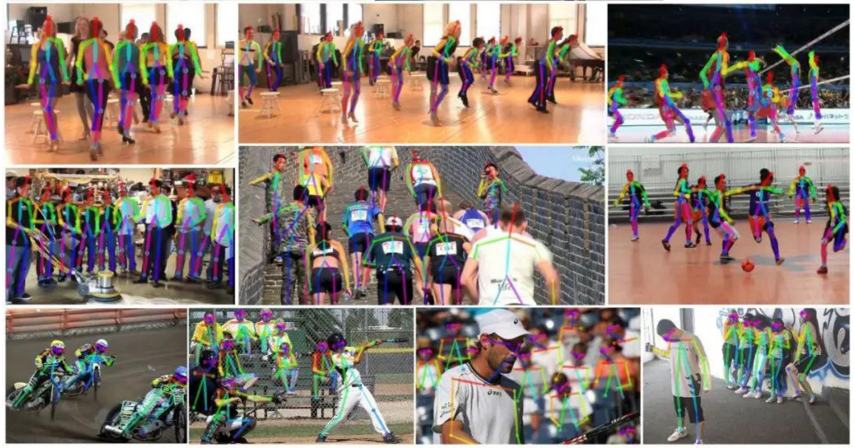
## Autonomous



## Healthcare



## Sports



## Surveillance & Security

Image sources:
https://viso.ai/applications/computer-vision-applications/
https://viso.ai/applications/computer-vision-in-healthcare/
https://www.superannotate.com/blog/computer-vision-in-security-and-surveillance

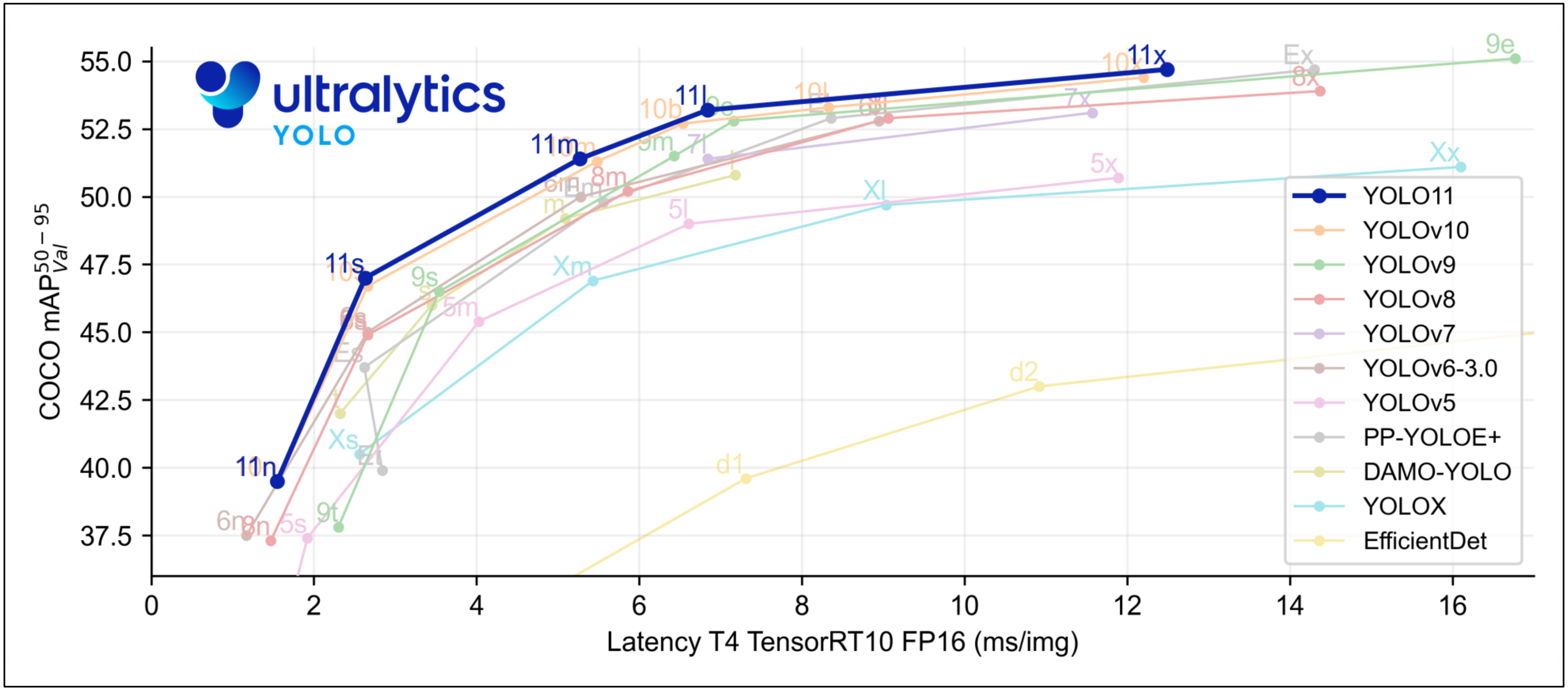
# Contents

# Current models



❑ Deep Learning-Powered Algorithms: Object detection

➢ YOLO (You Only Look Once)



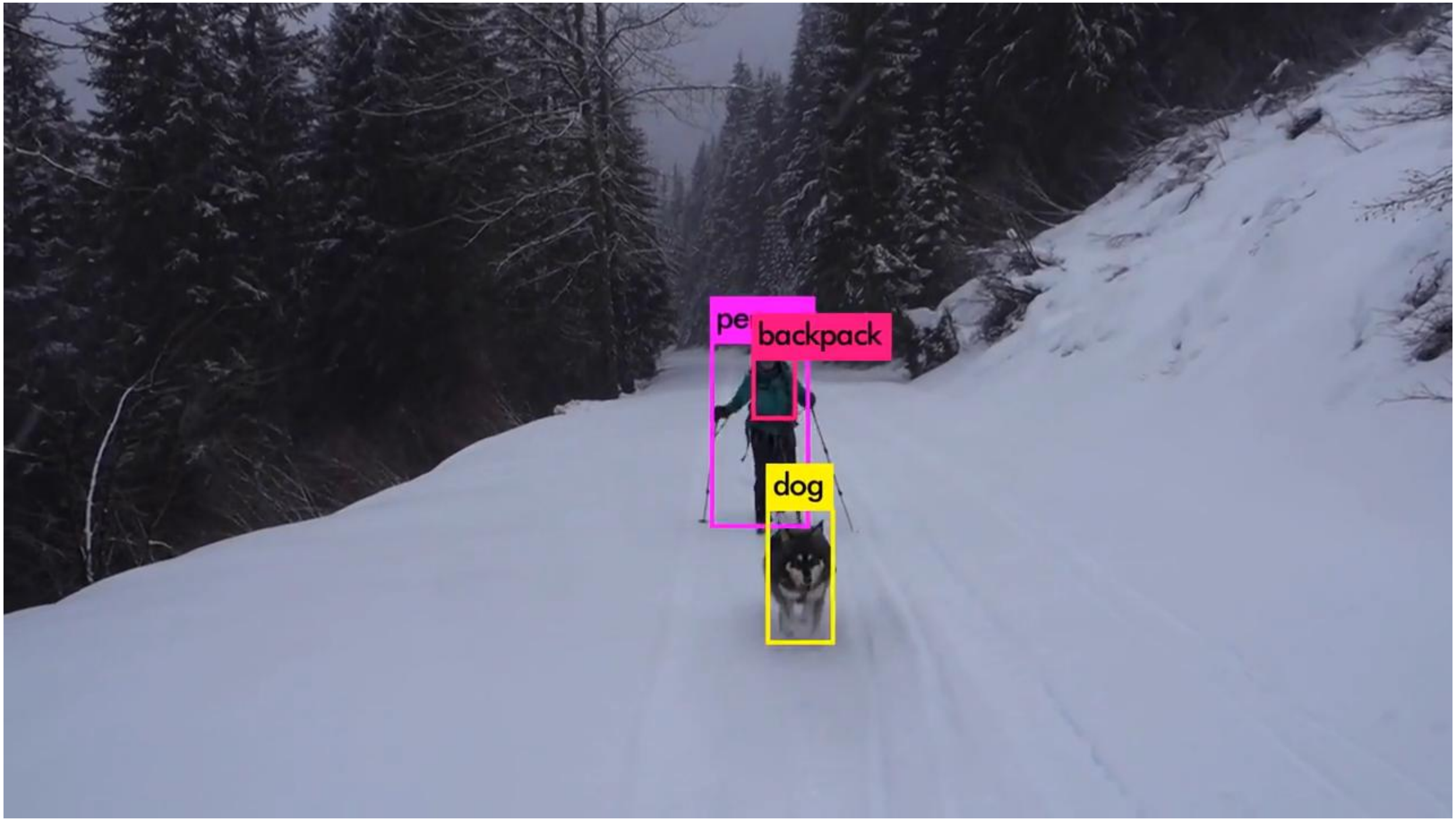

Source: https://docs.ultralytics.com/models/

# Current models



❑Deep Learning-Powered Algorithms: Object detection

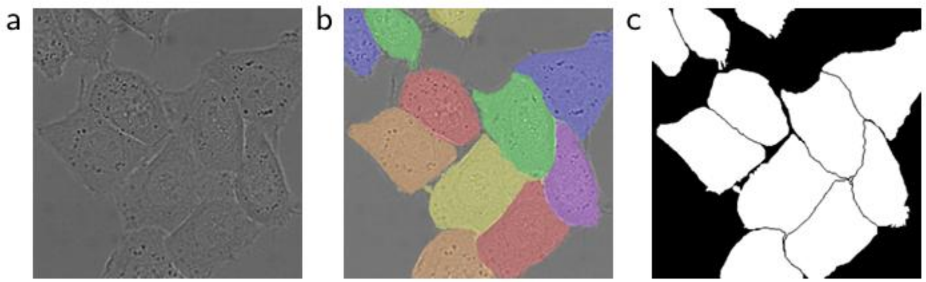
- YOLO (You Only Look Once)



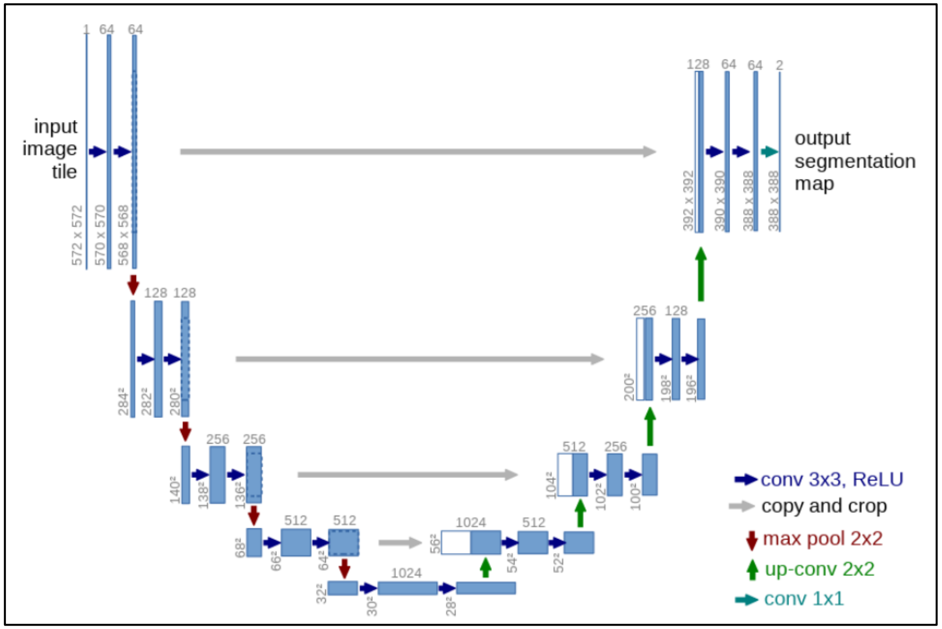

Source: https://www.youtube.com/watch?v=svn9-xV7wjk&ab_channel=DeepBean

## ❑Deep Learning-Powered Algorithms: Segmentation

- U-net (2012)




Input | Ground truth | Segment by U-net



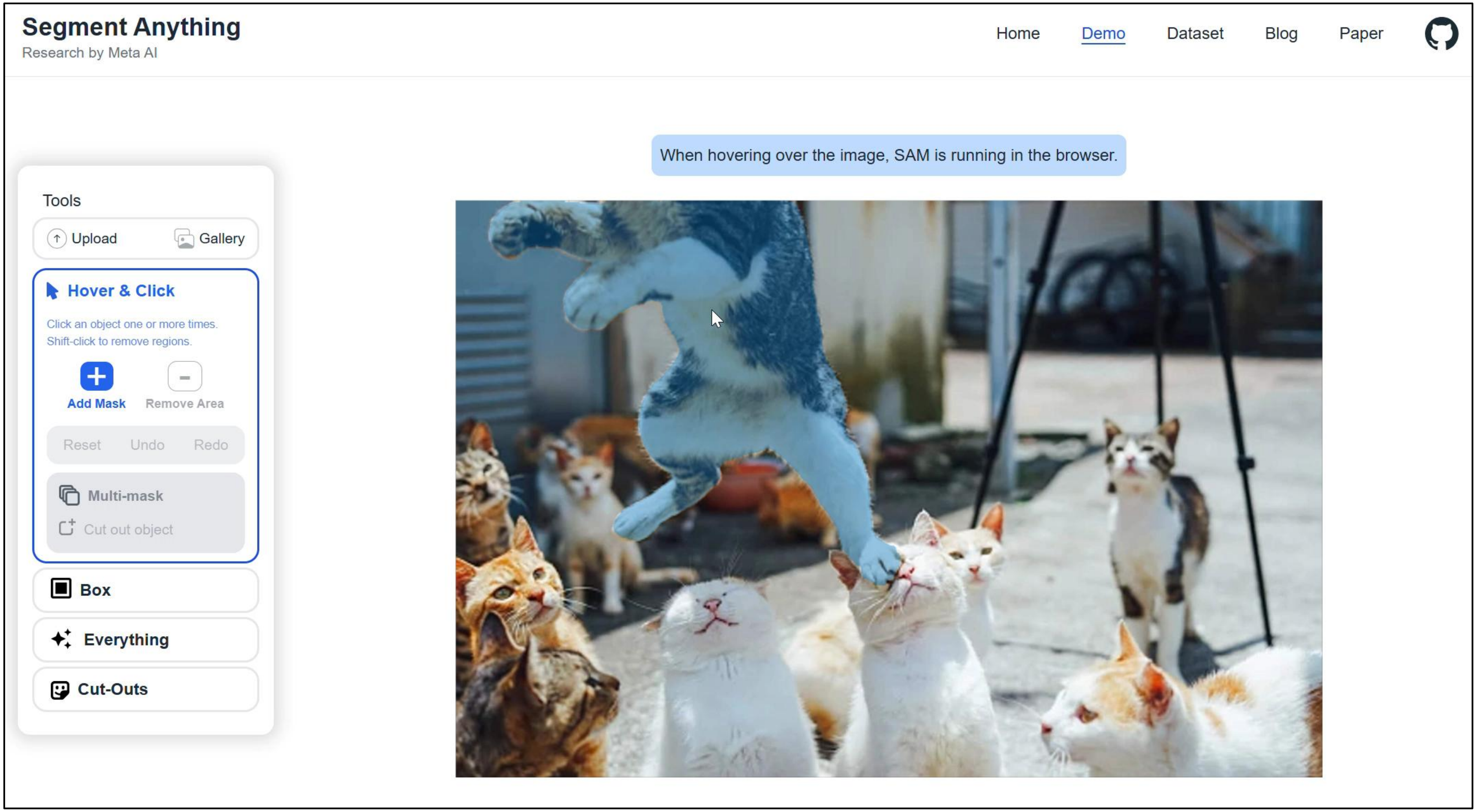

HeLa cell segmentation

Architecture

Source: Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." Medical image computing and computer-assisted intervention-MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18. Springer international publishing, 2015.

## ❑Deep Learning-Powered Algorithms: Segmentation

➢ SAM (Segment Anything Model)



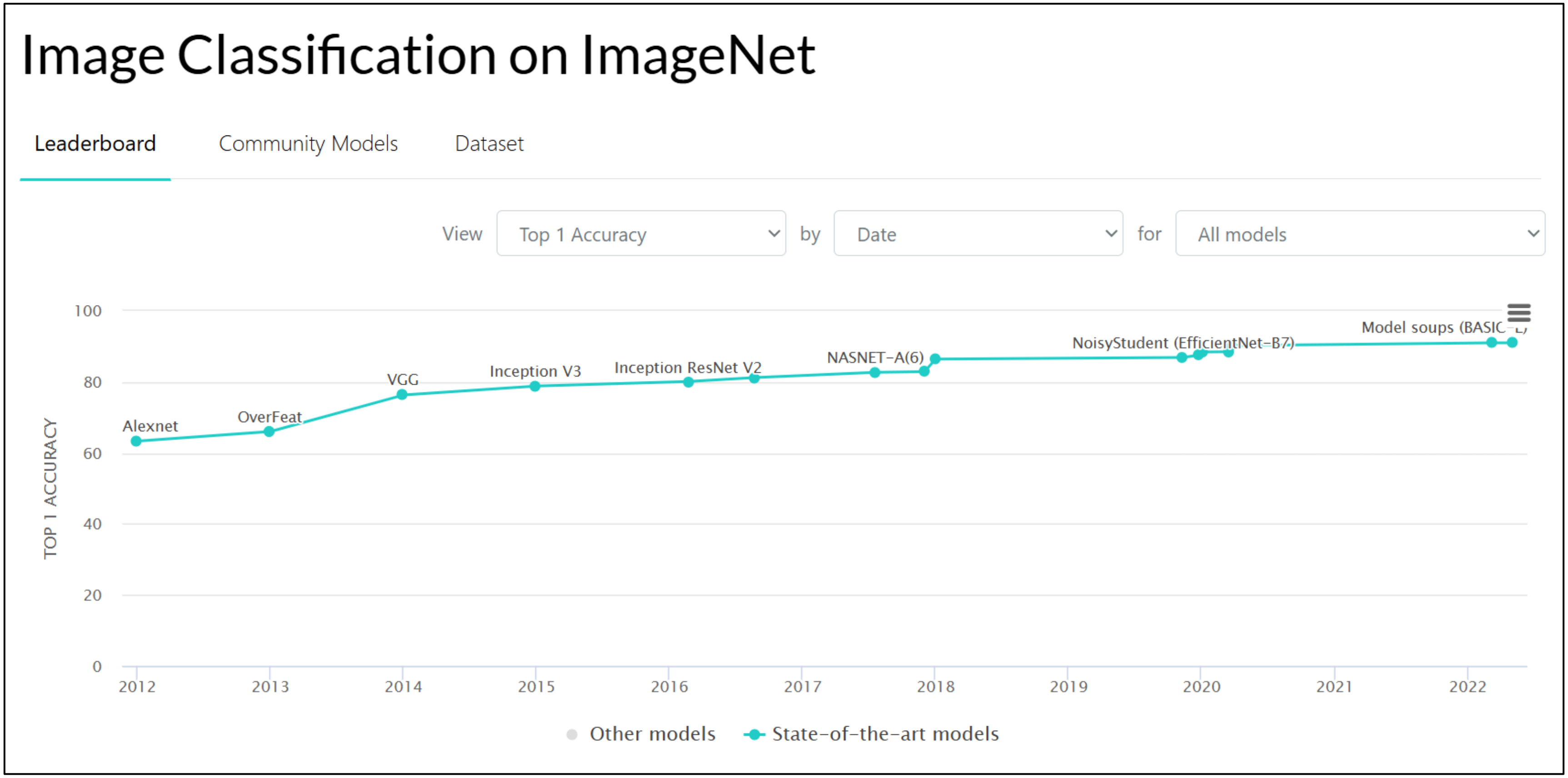

Source: https://segment-anything.com/demo

# Current models



## ❑Deep Learning-Powered Algorithms: Classification

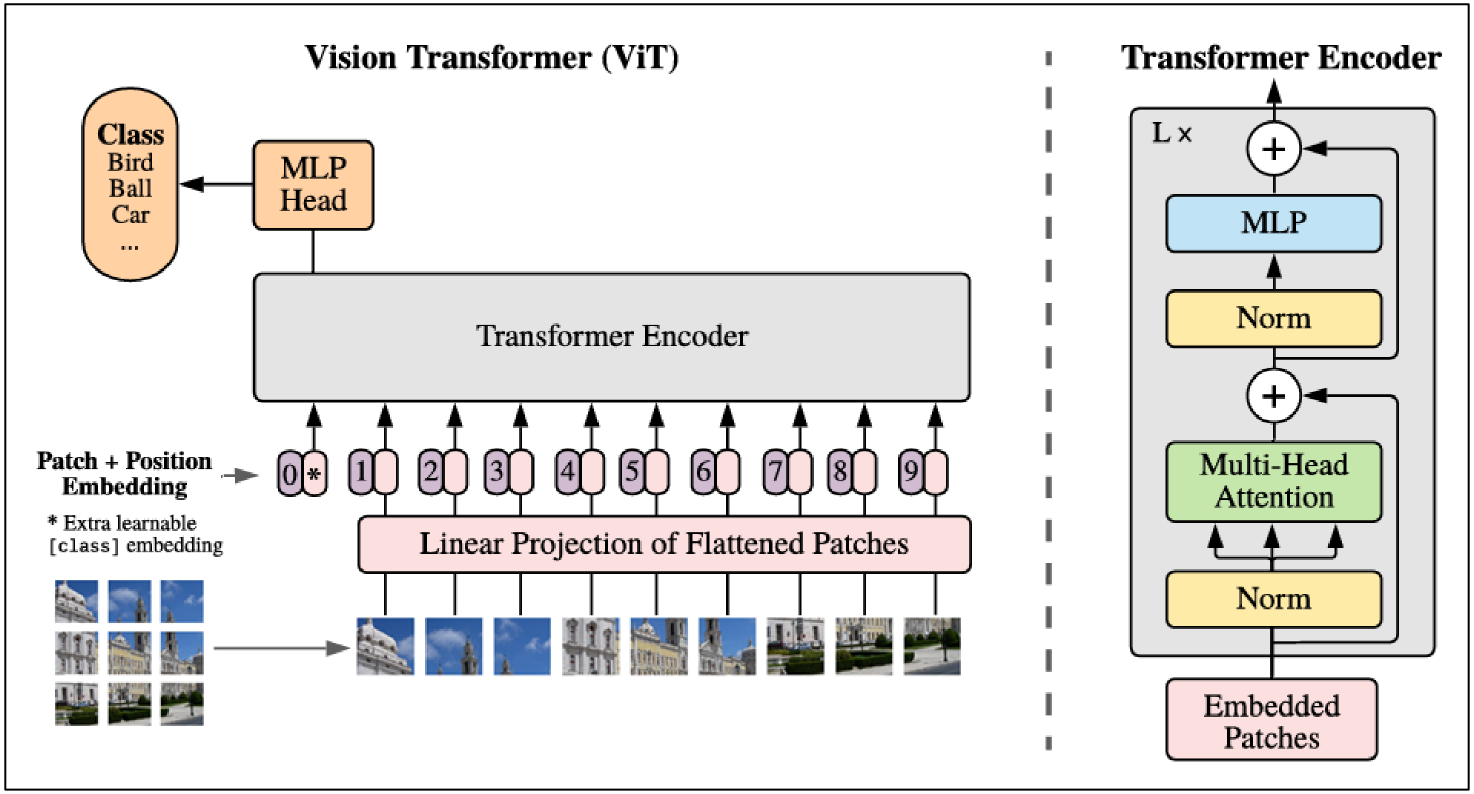
➢ State-of-the-art-models




Source: https://paperswithcode.com/sota/image-classification-on-imagenet?metric=Top%201%20Accuracy

❑Deep Learning–Powered Algorithms: Classification

➢ Vision Transformer (ViT)




Architecture

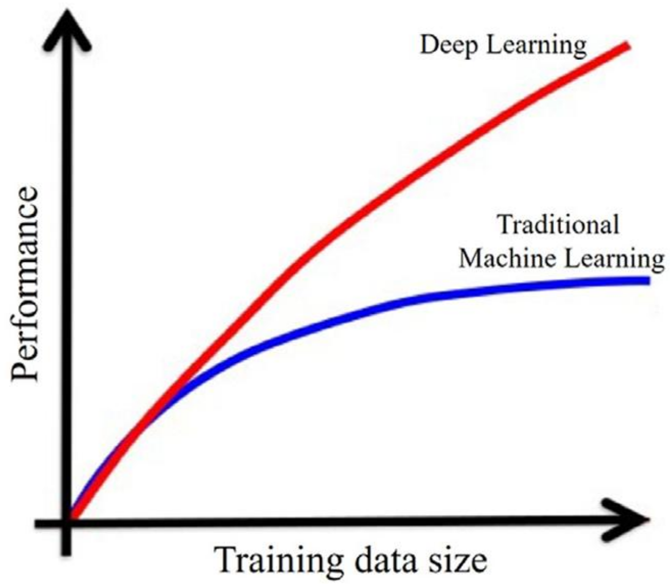
Source: Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.

# Contents

# Limitation of deep learning

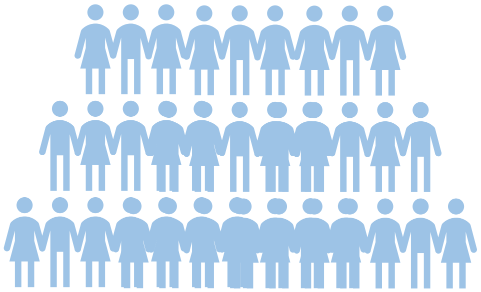


1. Dataset size: Requires large scale dataset




[Source]

2. Data annotation: Laborious and time-consuming



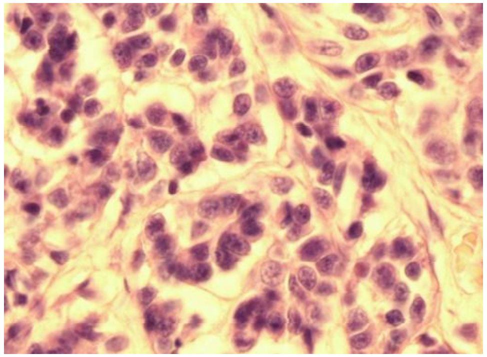
Source: Alyafeai, Z., & Ghouti, L. (2020). A fully-automated deep learning pipeline for cervical cancer classification. Expert Systems with Applications, 141, 112951.

# Contents

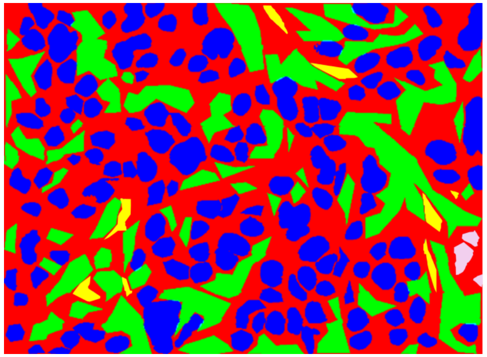
# Our recent research



1. Cancer cytoplasm segmentation[Source 1]



Image



Segmentation

1. Train detection[Source 2]

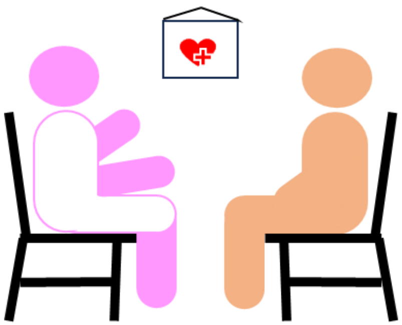

Source 1: R. Sultana, H. Horibe, T. Murakami, and I. Shimizu, ″Cancer Cytoplasm Segmentation from Cell Image Based on Transfer Learning″, 2024 6th International Conference on Image, Video and Signal Processing, 2024.
Source 2: Y. Inoue, R. Sultana, Y. Nishino, and I. Shimizu, ″Enhancing Daytime Train Image Detection with YOLOv8 through Data Augmentation Techniques″, 2024 6th Asia Digital Image Processing Conference (ADIP 2024), 2024.

# 1. Cancer cytoplasm segmentation

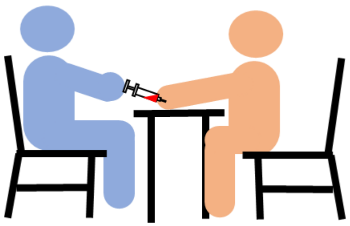


## ❑ Pathology diagnosis

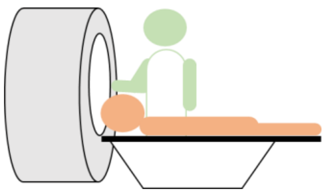
① Consult with doctor



② Various clinical tests

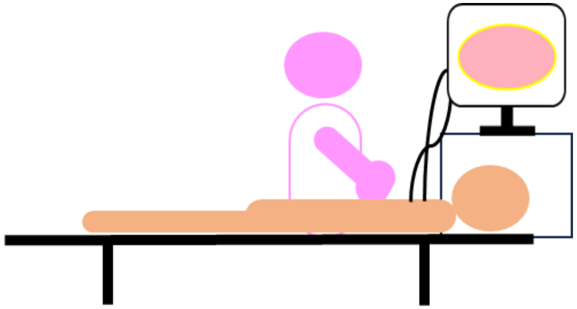


Blood sample

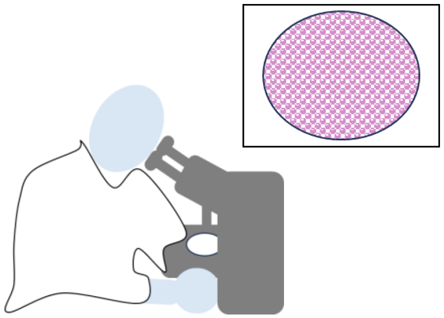


MRI

③ Biopsy in critical situation



④ Pathological examination



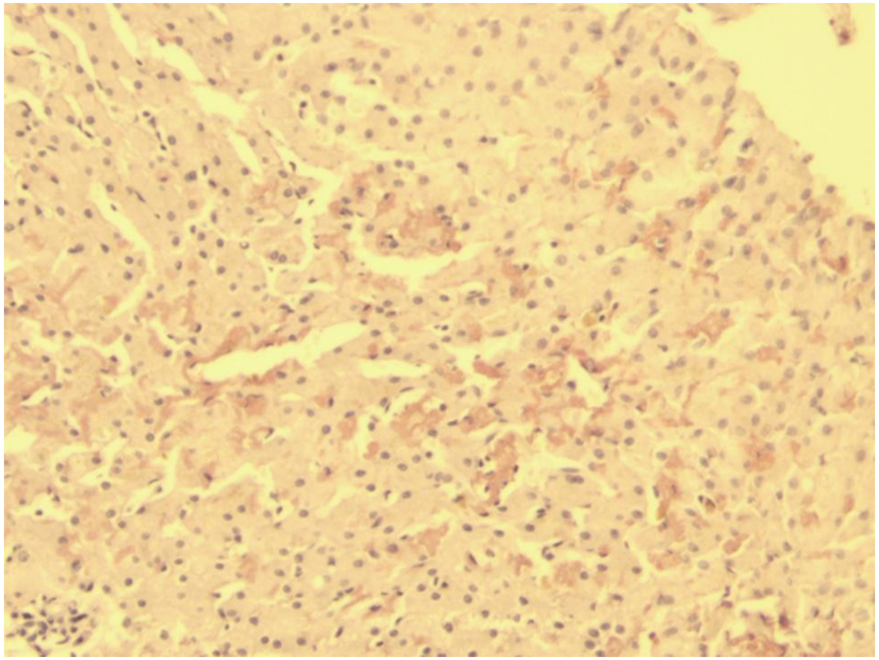
➢ Delaying the decision-making can cause significant loss to patient

# 1. Cancer cytoplasm segmentation



## ❑ Issues with pathology examination system

1. Shortage of pathologists (Two pathologist for 100 thousand people in Japan)[Source]
2. Difficult to identify cancer due to irregular shape of cancer cell with naked eye



Cancer cell image

➢ Digital pathology could support early-stage decision-making to reduce the burden on pathologists

Source: https://pathology.or.jp/ippan/outpatient-04.html

# 1. Cancer cytoplasm segmentation



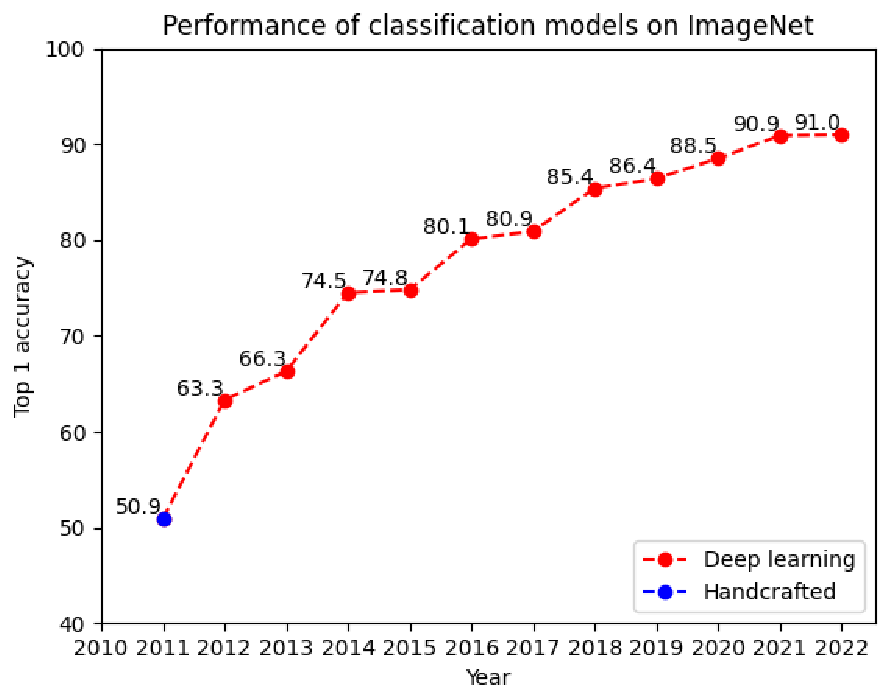
## ❑Methods used in digital pathology

- ❑Digital image analysis[Ruifrok+, 2001]
- ❑Machine learning algorithm[Ludovic+, 2013]
- ❑Deep learning models[Arbelle+, 2012]

Recent works due to outperformance of deep learning over conventional models

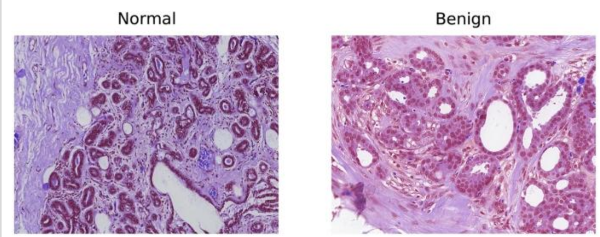
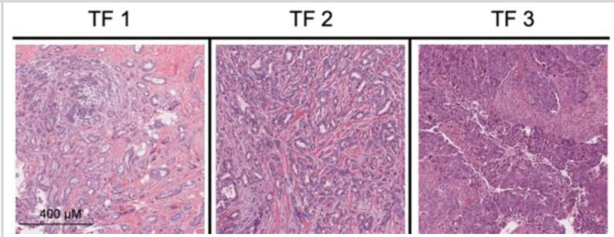
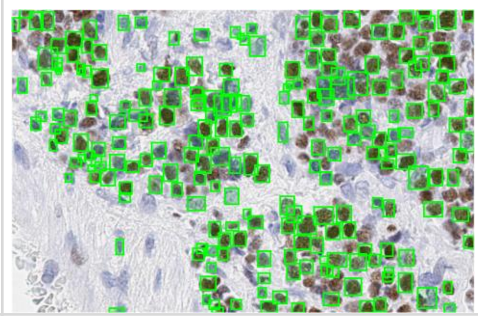
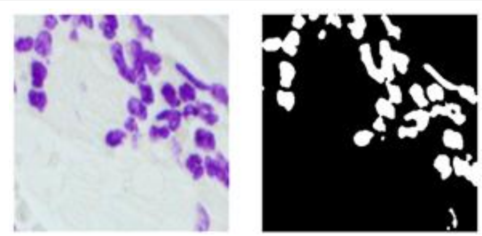
ImageNet dataset[Krizhevsky+, 2012]

Classification of 1000 categories



# 1. Cancer cytoplasm segmentation

## ❑Tasks in pathology image using deep learning

| | |
|---|---|
| Cell image classification |   [Hameed+ 2022] |
| Cancer grading<br>(Nottingham grading system) |   [Jaroensri+ 2022] |
| Nuclear detection |  [Source] |
| Nuclear segmentation |  Image Segmented map [Aydın+2022] |

Source: https://www.kitware.com/cell-nuclei-detection-on-whole-slide-histopathology-images-using-histomicstk-and-faster-r-cnn-deep-learning-models/

# 1. Cancer cytoplasm segmentation



## ❑ Challenge in pathology image analysis task using deep learning

- Lack of dataset for various tasks

Example of histopathological dataset

| Dataset | Task |
|---|---|
| MoNuSeg | Nuclear segmentation |
| TNBC | |
| UCSB | |
| BreaKHis | Cancer classification |
| ARCH | Multiple instance captioning |

Cancer cytoplasm is one of the important elements for cancer type identification in pathology

↓

To the best of our knowledge, no dataset is available for cancer cytoplasm detection research

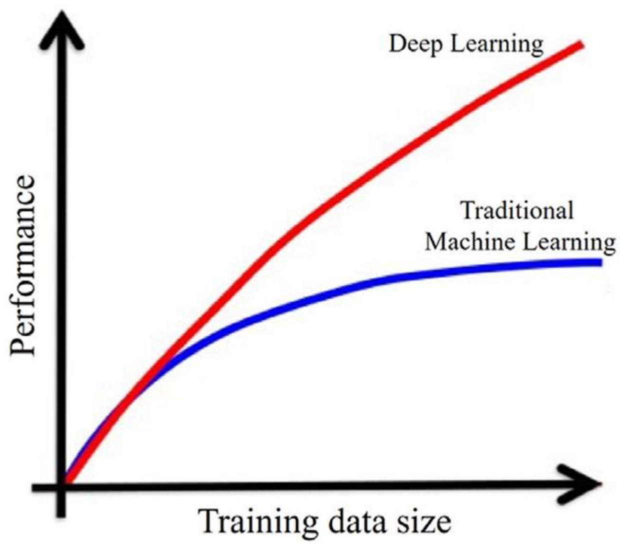
↓

We introduce deep learning to the cancer cytoplasm segmentation of cell images

## ❑Important factors in training deep learning

Performance depends on scale of dataset

[Source]



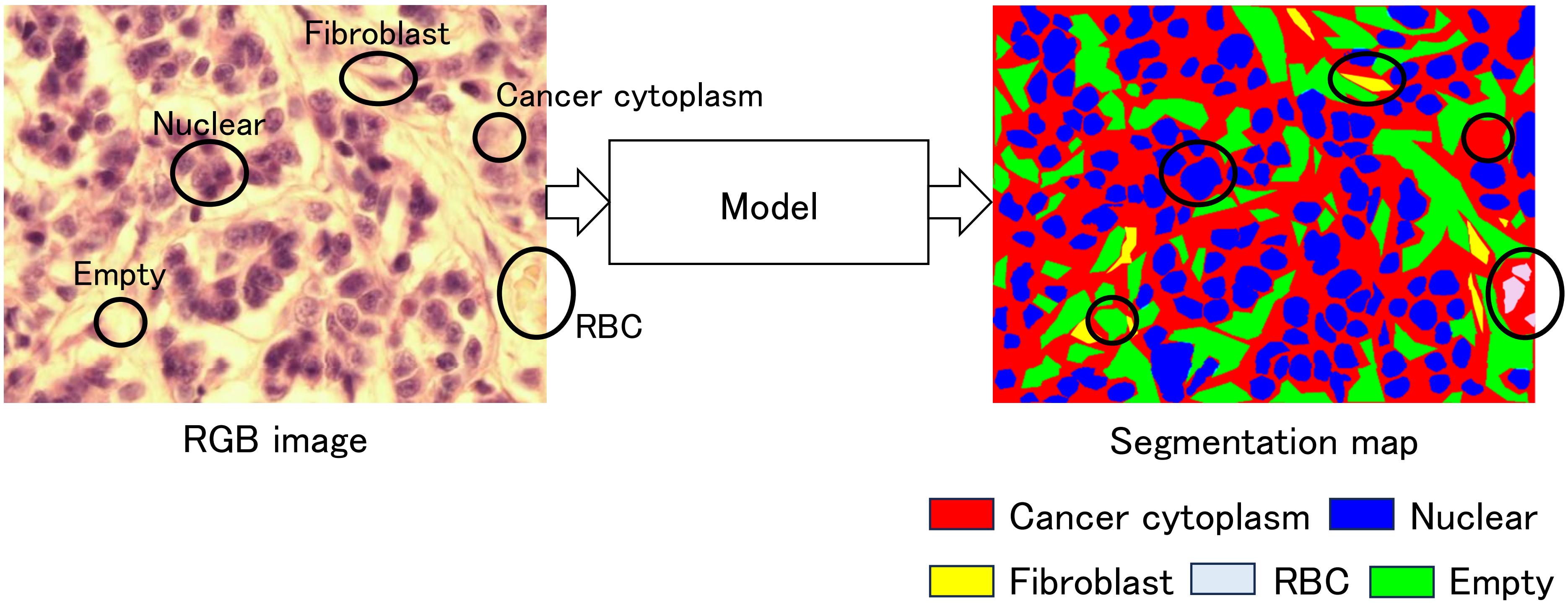

- How to learn deep learning model with small dataset?
  1. Pre-training/ fine-tuning → Also known as transfer learning
  2. Data augmentation

Source: Alyafeai, Z., & Ghouti, L. (2020). A fully-automated deep learning pipeline for cervical cancer classification. Expert Systems with Applications, 141, 112951.

# 1. Cancer cytoplasm segmentation

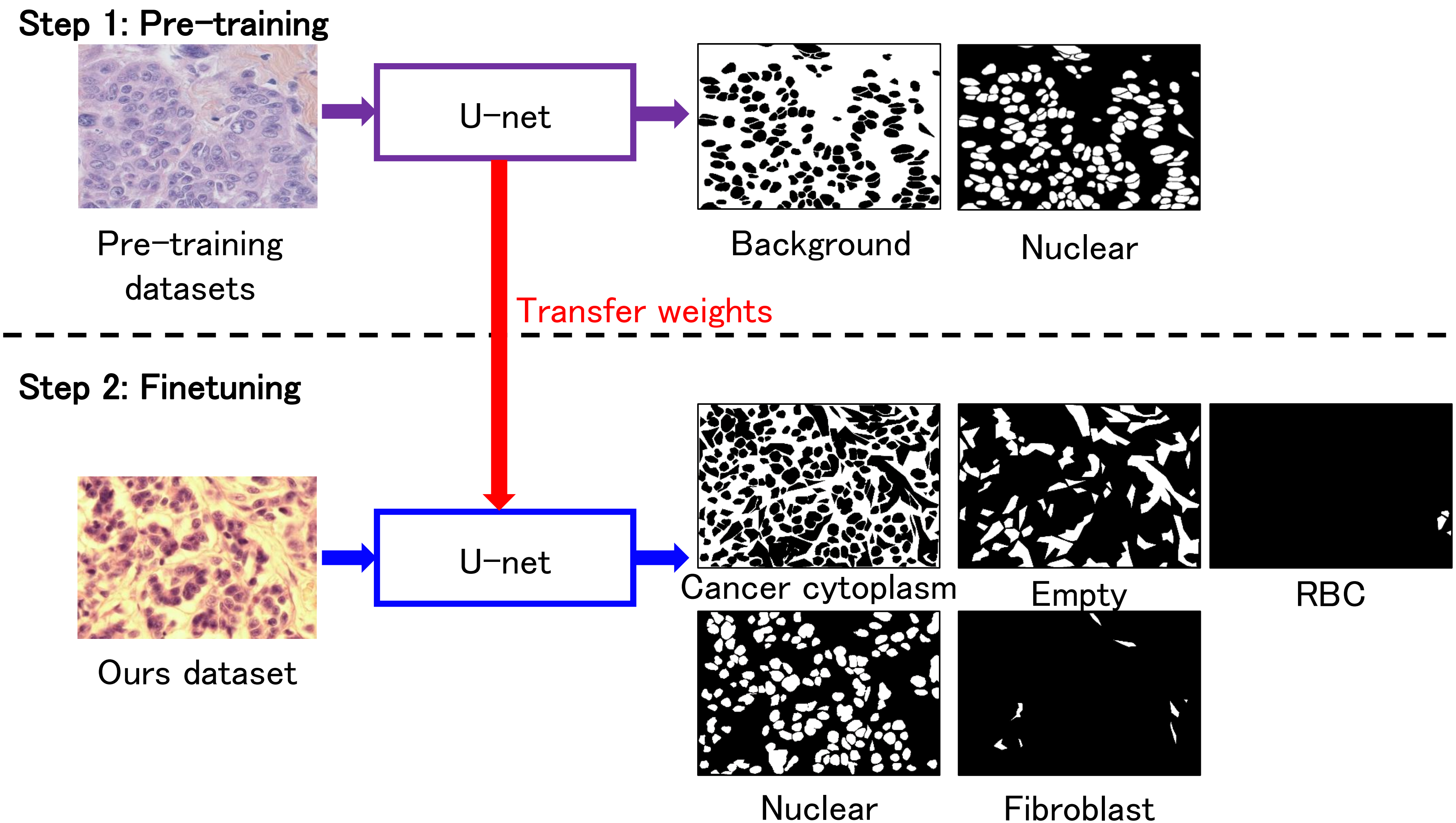
## ❑Purpose

Segmentation of cancer cytoplasm in RGB cell images using deep learning model with transfer learning approach



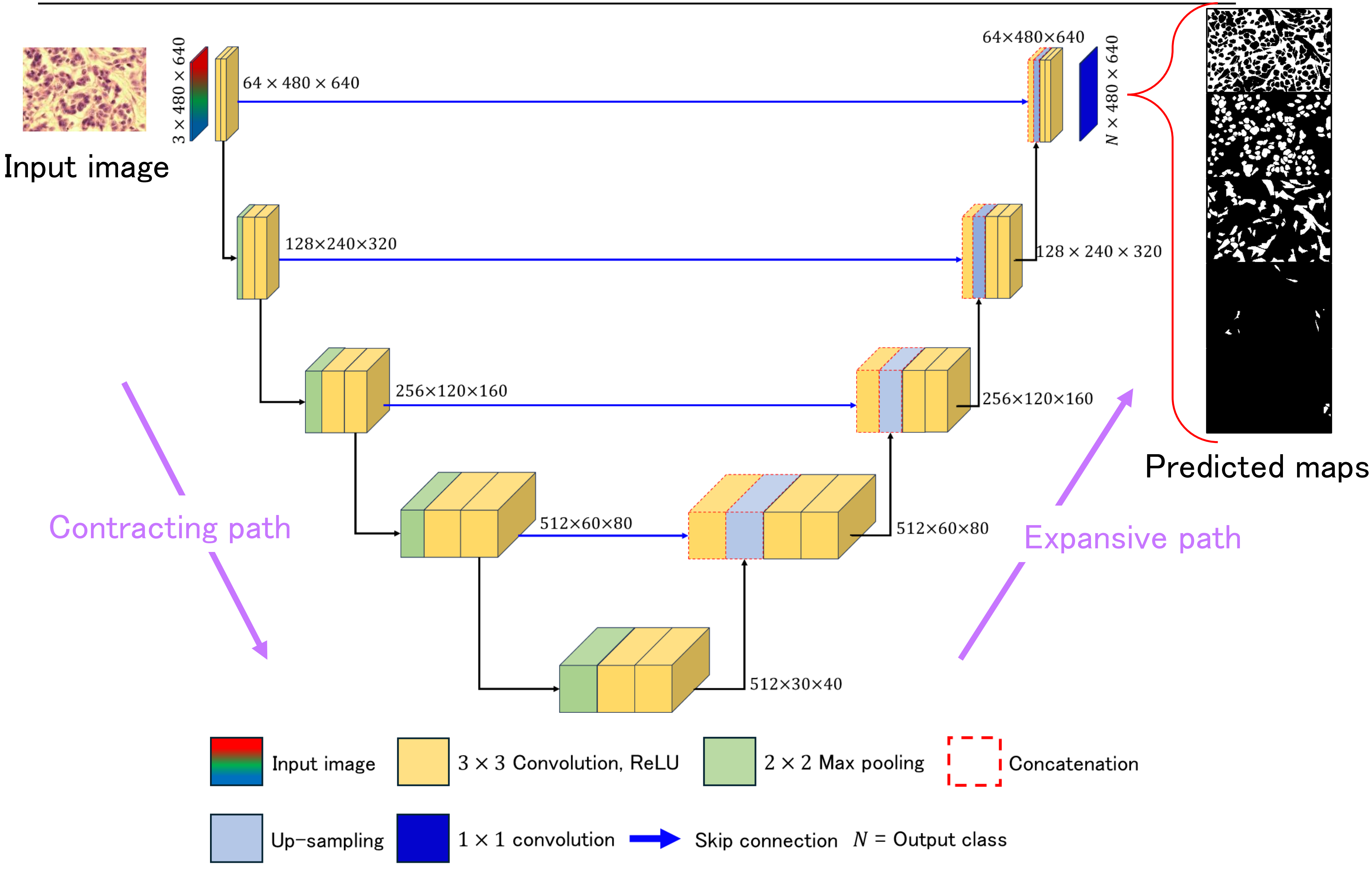
# 1. Cancer cytoplasm segmentation

## ❑Proposed method



# 1. Cancer cytoplasm segmentation

❑ <u>U-net architecture is selected due to it's effectiveness in different tasks</u>

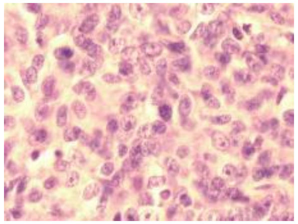


# 1. Cancer cytoplasm segmentation



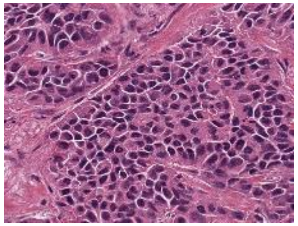
## ❑Dataset

| Dataset | Images | Segment class | Type | Capture technique |
|---|---|---|---|---|
| Ours | 44 | 5 (cytoplasm, nuclear, fibroblast, RBC, empty) | Cancer | H&E stained |
| MoNuSeg | 37 | 2 (nuclear, background) | | |
| TNBC | 50 | | | |
| UCSB | 58 | | | |
| | | | | |
| Fluo-N2DH-GOWT1 | 184 | 2 (nuclear, background) | - | Fluorescence |
| Fluo-C2DL-MSC | 96 | | Tropism | |
| fluocells | 283 | | - | |
| | | | | |
| DIC-C2DH-HeLa | 168 | 2 (nuclear, background) | Cancer | Differential Interference Contrast |
| PhC-C2DH-U373 | 230 | | | Phase contrast |

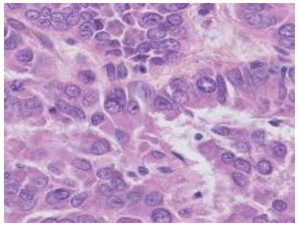


Ours

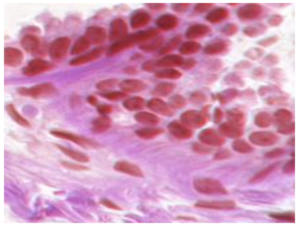
Fine-tuning dataset



MoNuSeg



TNBC



UCSB

Pre-training datasets

# 1. Cancer cytoplasm segmentation



## ❑ <u>Training condition</u>

- Dataset

| Dataset | Images | Segmented objects | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | Cancer cytoplasm | Nuclear | Background/ Empty | Fibroblast | RBC | Total | |
| Ours | Train: 24<br>Valid: 6<br>Test: 14 | 343 | 10526 | 5344 | 260 | 34 | 16507 | |
| MoNuSeg | 37 | – | 34774 | 147 | – | – | 34921 | 47029 |
| TNBC | 50 | – | 8052 | 110 | – | – | 8162 | |
| UCSB | 58 | – | 3783 | 163 | – | – | 3946 | |

# 1. Cancer cytoplasm segmentation



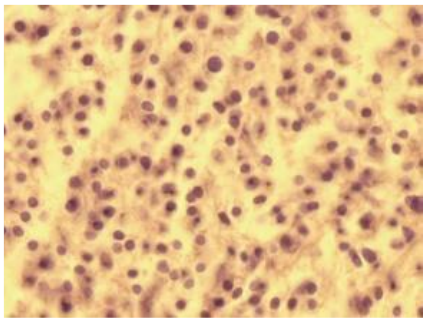
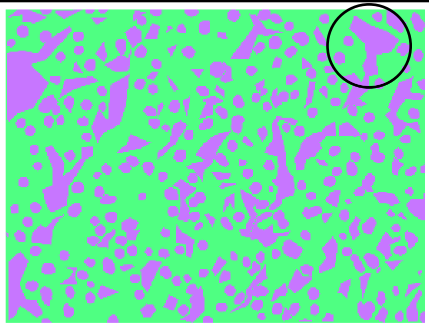
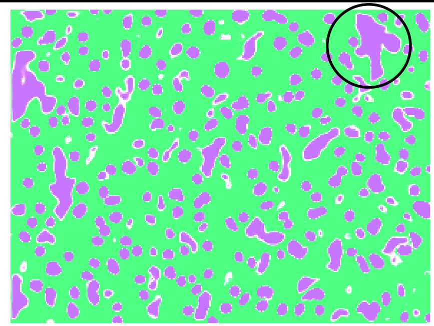
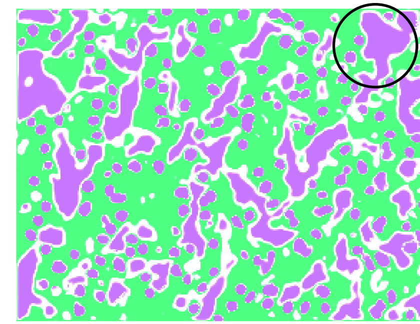
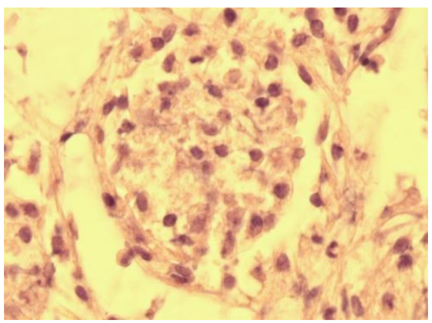
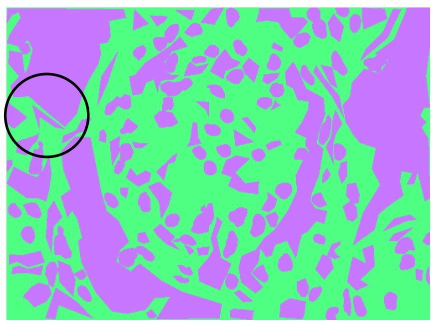
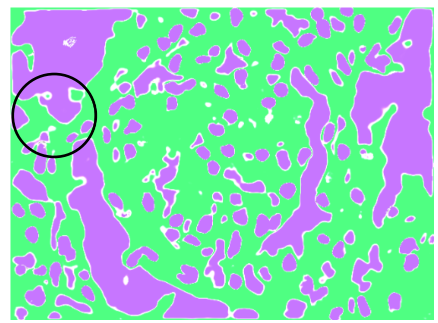
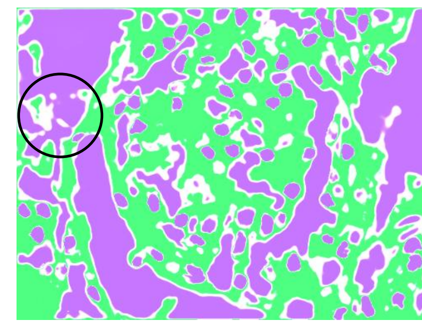
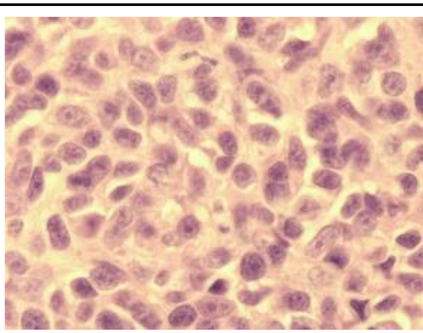
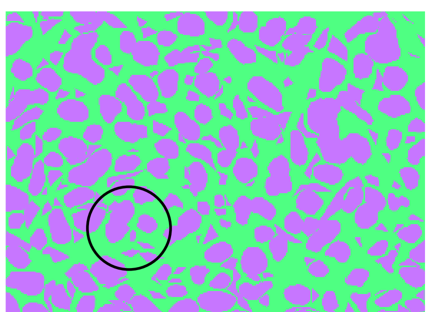
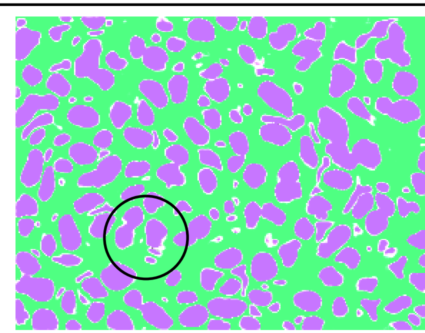
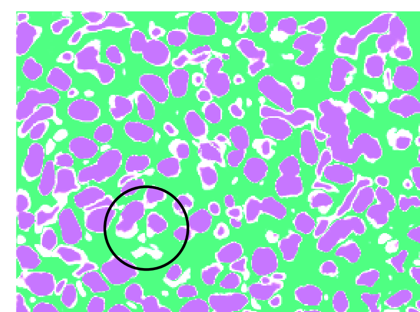
## ❑ Training condition

➢ Environment

| | |
|---|---|
| CPU | Intel® CoreTM i9-10900K CPU @ 3.70GHz 32GB |
| GPU | NVIDIA GeForce RTX 3090 24GB GPU |
| Image size | 640×480 |
| Epoch | Pre-training: 400<br>Finetuning: 100 |
| Evaluation metrics | IoU, dice coefficient |
| Optimizer | RMSprob |
| Learning rate | 0.0001 |
| Framework | PyTorch |
| Programming language | Python |

# 1. Cancer cytoplasm segmentation

## ❑Qualitative results

| Image | Ground truth | Proposed method (with pre-training) | Comparison method (without pre-training) |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Cancer cytoplasm Other area Overlapped area of cancer cytoplasm and other area

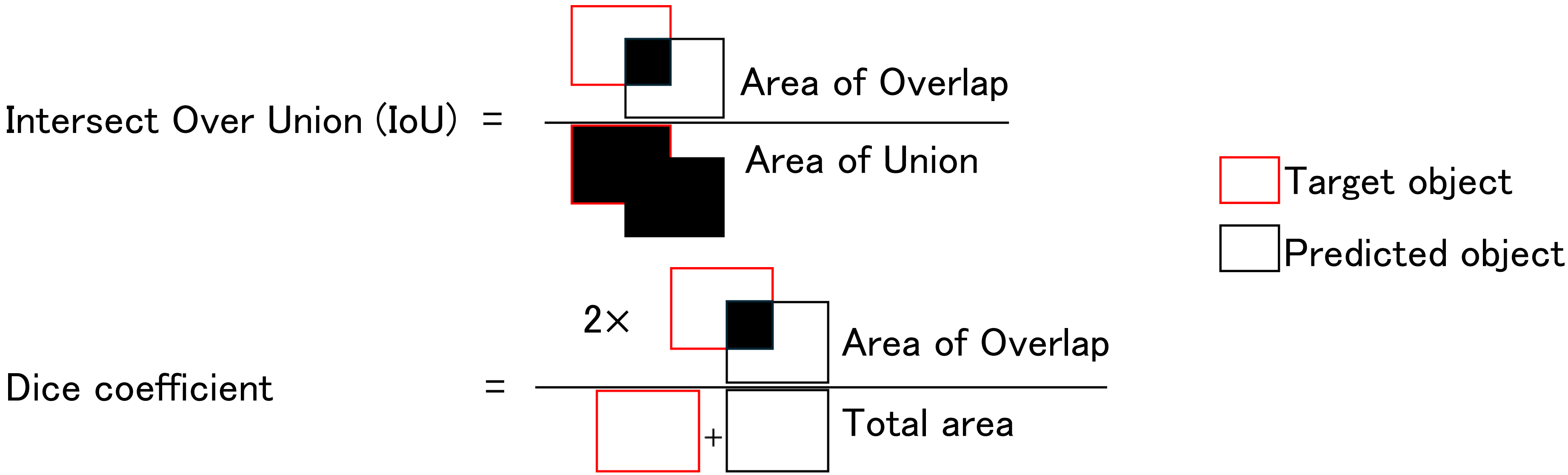
# 1. Cancer cytoplasm segmentation



## ❑Quantitative results

| Method | Transfer learning | IoU (↑) | Dice coefficient (↑) |
|---|---|---|---|
| Proposed | ✓ | **69.02 (+14%)** | **80.45 (+2%)** |
| Comparison | ✘ | 60.56 | 78.85 |

IoU = Measure of segmentation accuracy
Dice coefficient = Balanced evaluation of segmentation performance
↑ = the higher the score is the better the model performance is



# 1. Cancer cytoplasm segmentation



## ❑Ablation analysis

➢ Evaluation class

| Method | Classes | | With pre-training | | Without pre-training | |
|---|---|---|---|---|---|---|
| | Training | Evaluation | IoU (↑) | Dice coefficient (↑) | IoU (↑) | Dice coefficient (↑) |
| 1 | 5 | 2 (proposed) | **69.02** | **80.45** | 60.56 | 71.85 |
| | | 5 | 44.58 | 50.93 | 43.95 | 50.30 |
| 2 | 2 | 2 | 67.31 | 79.10 | 61.77 | 74.32 |

➢ Image pattern matching

| Method | Train set | Test set | Pre-training | IoU (↑) | Dice coefficient (↑) |
|---|---|---|---|---|---|
| 1 | Ours | Ours | × | **61.77** | **74.32** |
| 2 | UCSB + MoNuSeg+ TNCB | Ours | × | 10.85 | 17.24 |

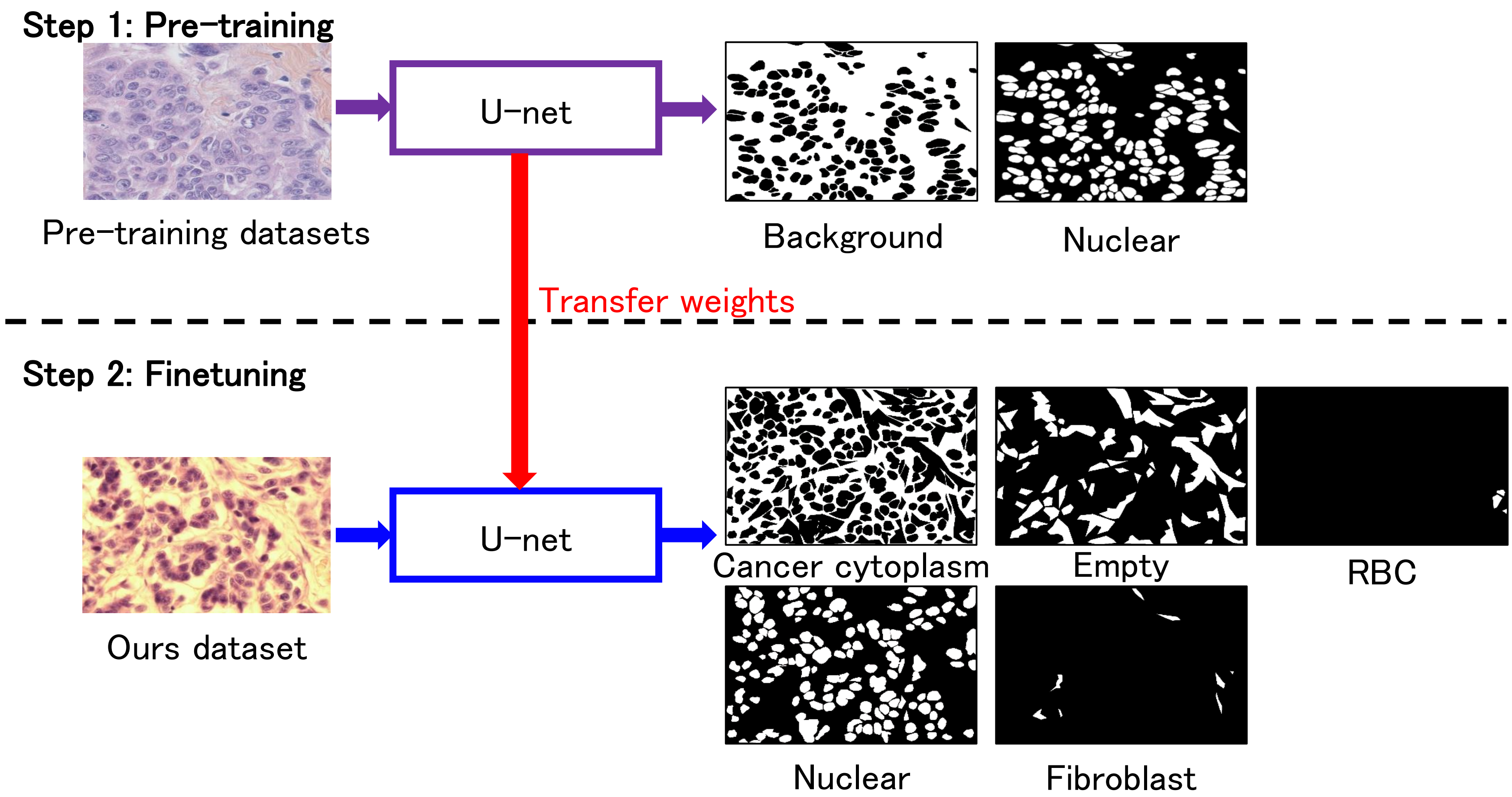
↑ indicates the higher the score is the better the model performance is

# 1. Cancer cytoplasm segmentation



## ❑Task overview

- Cancer cytoplasm area is segmented in cancer cell image using transfer learning.



1. Cancer cytoplasm segmentation



Image



Segmentation

2. Train detection

# 2. Train detection



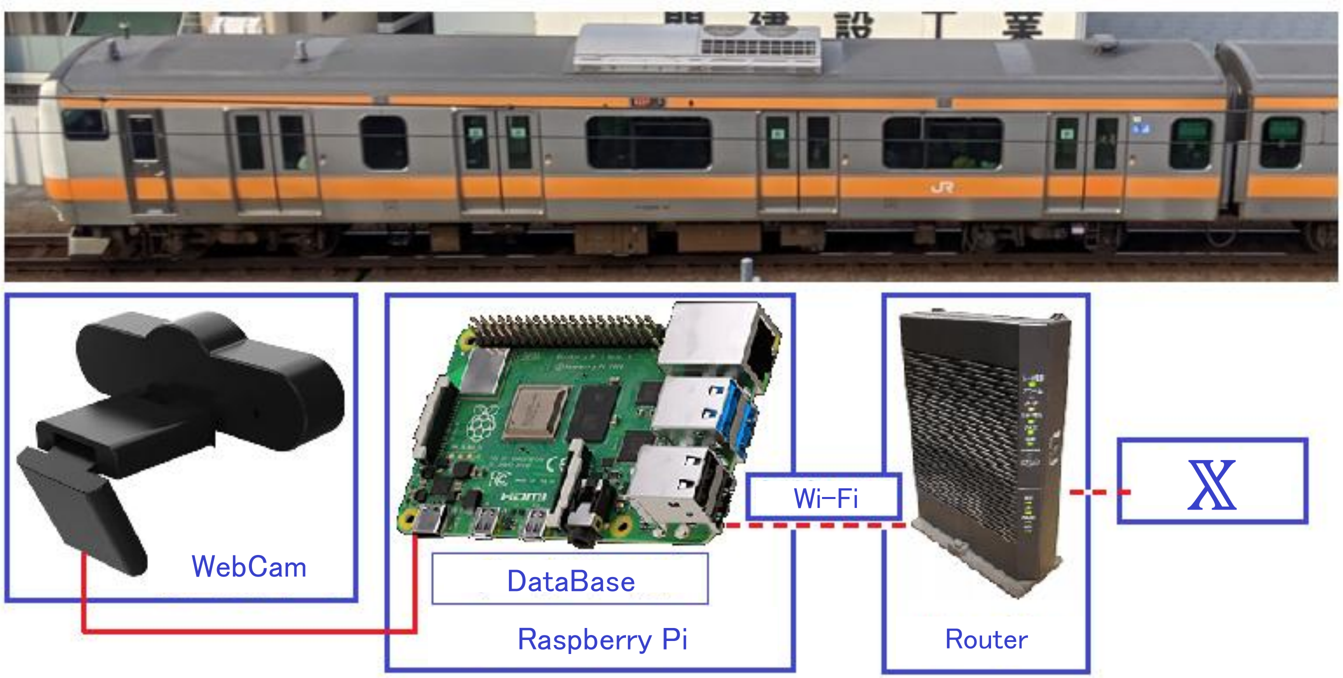
## ❑ <u>Research background</u>

- In Japan: Train rarely delays.
  →If the train is a little late…

Public transportation delays ⇨ Significant Disruption

## ❑ Previous research

Camera-based train delay management and information system




Source:(https://www.ipsj.or.jp/event/taikai/85/85PosterSession/ipsj_poster/pdf/8021.pdf

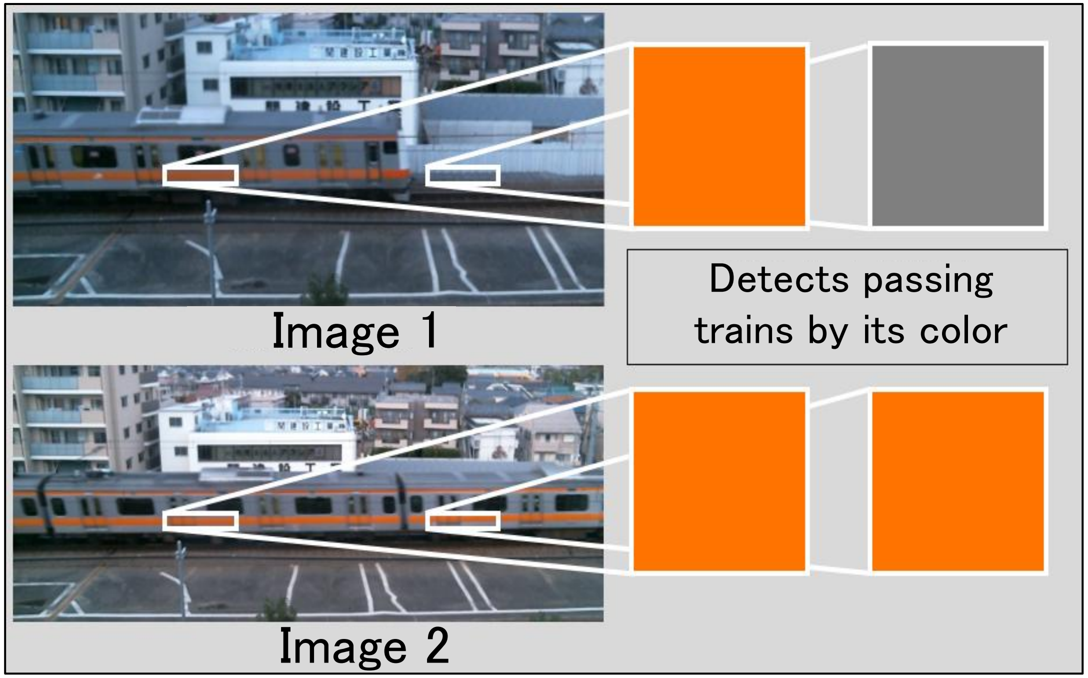
## ❑ Problem in previous research

Method of Previous Research

↓

Color-based masking

Significant drop in detection accuracy
during evening and night



## ❑ Image recognition models

- Example of image detection model
  - RetinaNet
  - CenterNet
  - EfficientNet
  - CornerNet
  - YOLO

## ❑ Purpose

- Image Recognition models

**Utilize YOLOv8**
**for train detection**

**YOLOv8:** Able to process 45 frames per second → **suitable for real-time object detection**

# 2. Train detection

## ❑ Proposed method

Data Augmentation

Original Image

- Treating 5 kinds of editing

1. RandomRain
2. Blur
3. MedianBlur
4. CLAHE
5. ImageCompression



5 Augmented Images

## ❑ Proposed method

Data Augmentation

RandomRain

Original Image

MedianBlur

CLAHE

Augment data by processing one photo into multiple versions.

## ❑ Proposed method

**RandomRain**

create an ideal representation of rainy weather conditions

**MedianBlur**

addresses the issue of image blurriness, which can result from high train speeds

**CLAHE**

enhancing the structural details of the train in the image

# 2. Train detection



## ❑Experiment

Datasets

**For Training: Annotated**

Original Image: 30

Data Augmentation

Augmented Image: 150

# 2. Train detection



## ❑Experiment

**For Prediction: Not Annotated**

- Taken in November, evening
- Total: 701

# 2. Train detection



## ❑Experiment

- Utilize YOLOv8

- Training Condition

Epoch: 500
Batch size: 4
GPU: NVIDIA GeForce RTX 2060 × 1

## ❑Experiment

- Vary the number of training data
- Compare those models each other

| Method | Data augmentation | Original image | Augmented image |
|---|---|---|---|
| Proposed | ✓ | 30 | 150 |
| Comparison | × | 30 | × |
| Handcrafted | × | × | × |

## ❑Experiment

- Ablation models

| Method | Images for training | | | | | |
|---|---|---|---|---|---|---|
| | Original image | Data augmentation | | | | |
| | | Random Rain | Blur | Median blur | CLAHE | Image compression |
| Proposed | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| ① | ✓ | ✓ | ✓ | ✓ | ✓ | × |
| ② | ✓ | ✓ | ✓ | ✓ | × | ✓ |
| ③ | ✓ | ✓ | ✓ | × | ✓ | ✓ |
| ④ | ✓ | ✓ | × | ✓ | ✓ | ✓ |
| ⑤ | ✓ | × | ✓ | ✓ | ✓ | ✓ |

## ❑<u>Experiment</u>

- Proposed method marked the highest accuracy.

| Method | Data augmentation | Original image | Augmented image | **Accuracy (%)** |
|---|---|---|---|---|
| Proposed | ✓ | 30 | 150 | <u>**99.672**</u> |
| Comparison | × | 30 | × | 74.893 |
| Handcrafted | × | × | × | 9.985 |

## ❑ <u>Experiment</u>

- Compare those models each other

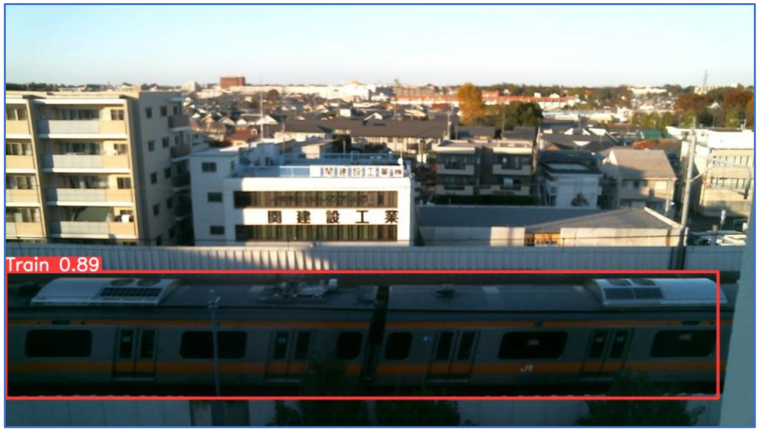
| Method | Images for training | | | | | | Accuracy (%) |
|---|---|---|---|---|---|---|---|
| | Original image | Data augmentation | | | | | |
| | | Random Rain | Blur | Median blur | CLAHE | Image compression | |
| Proposed | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | 99.672 |
| ① | ✓ | ✓ | ✓ | ✓ | ✓ | × | 97.860 |
| ② | ✓ | ✓ | ✓ | ✓ | × | ✓ | <u>**84.736**</u> |
| ③ | ✓ | ✓ | ✓ | × | ✓ | ✓ | 99.287 |
| ④ | ✓ | ✓ | × | ✓ | ✓ | ✓ | 99.429 |
| ⑤ | ✓ | × | ✓ | ✓ | ✓ | ✓ | <u>**75.749**</u> |

# ❑Experiment

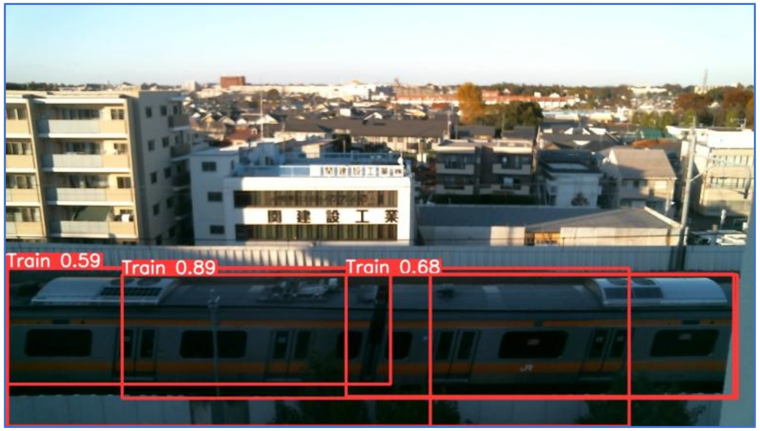
## Comparison of Proposed & ②

| Method | Images for training | | | | | | Accuracy (%) |
|---|---|---|---|---|---|---|---|
| | Original image | Data augmentation | | | | | |
| | | Random Rain | Blur | Median blur | CLAHE | Image compression | |
| Proposed | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | **99.672** |
| ② | ✓ | ✓ | ✓ | ✓ | × | ✓ | 84.736 |

- **CLAHE significantly improves recognition accuracy**




Proposed




②

# 2. Train detection



## ❑Experiment

### Comparison of Proposed & ②

| Method | Images for training | | | | | | Accuracy (%) |
|---|---|---|---|---|---|---|---|
| | Original image | Data augmentation | | | | | |
| | | Random Rain | Blur | Median blur | CLAHE | Image compression | |
| Proposed | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | **99.672** |
| ② | ✓ | ✓ | ✓ | ✓ | × | ✓ | 84.736 |

- **CLAHE significantly improves recognition accuracy**

Proposed


②

# ❑Experiment

## Comparison of Proposed & ⑤

| Method | Images for training | | | | | | Accuracy (%) |
|---|---|---|---|---|---|---|---|
| | Original image | Data augmentation | | | | | |
| | | Random Rain | Blur | Median blur | CLAHE | Image compression | |
| Proposed | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | **99.672** |
| ⑤ | ✓ | × | ✓ | ✓ | ✓ | ✓ | 75.749 |

- **RandomRain enhances recognition accuracy considerably**


Proposed


⑤

# ❑Experiment

## Comparison of Proposed & ⑤

| Method | Images for training | | | | | | Accuracy (%) |
|---|---|---|---|---|---|---|---|
| | Original image | Data augmentation | | | | | |
| | | Random Rain | Blur | Median blur | CLAHE | Image compression | |
| Proposed | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | **99.672** |
| ⑤ | ✓ | × | ✓ | ✓ | ✓ | ✓ | 75.749 |

- **RandomRain enhances recognition accuracy considerably**


Proposed

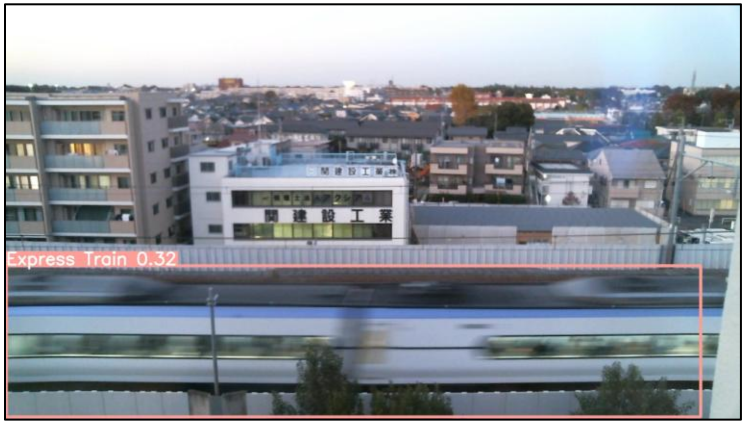
⑤

## ❑Experiment

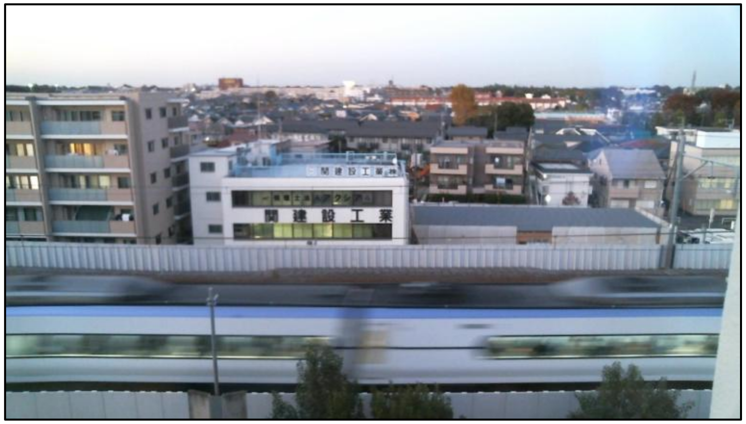
### Comparison of Proposed & ⑤

| Method | Images for training | | | | | | Accuracy (%) |
|---|---|---|---|---|---|---|---|
| | Original image | Data augmentation | | | | | |
| | | Random Rain | Blur | Median blur | CLAHE | Image compression | |
| Proposed | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | **99.672** |
| ⑤ | ✓ | × | ✓ | ✓ | ✓ | ✓ | 75.749 |

- **RandomRain enhances recognition accuracy considerably**




Proposed



⑤

## ❑Experiment

### Comparison of Proposed & ②,⑤

| Method | Images for training | | | | | | Accuracy (%) |
|---|---|---|---|---|---|---|---|
| | Original image | Data augmentation | | | | | |
| | | Random Rain | Blur | Median blur | CLAHE | Image compression | |
| Proposed | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | <u>99.672</u> |
| ② | ✓ | ✓ | ✓ | ✓ | × | ✓ | 84.736 |
| ⑤ | ✓ | × | ✓ | ✓ | ✓ | ✓ | 75.749 |

Random Rain

CLAHE

Accuracy can be improved by training on photos with **different brightness**

YOLOv8

+

Data Augmentation

⬇

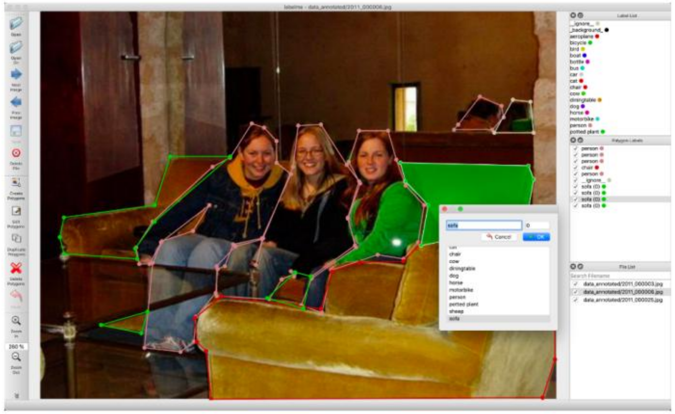
**Improve train detection accuracy**
**A practical train delay information provision system was realized.**

# Important links for data annotation



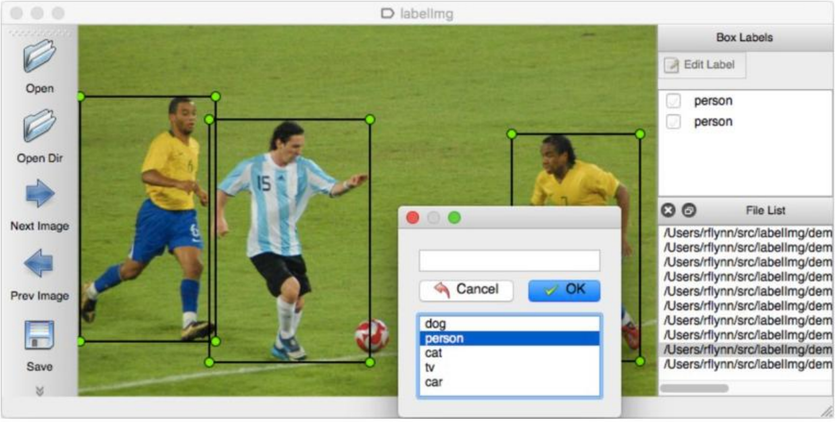
❑Software for segmentation

https://github.com/wkentaro/labelme



❑Software for detection

https://github.com/HumanSignal/labelImg



Thank you very much for your kind attention.