

การอบรมเชิงปฏิบัติการ

การใช้อุปกรณ์ ESP32-CAM และ Deep Learning สำหรับ CUBESAT

ศูนย์ภูมิภาคเทคโนโลยีอวกาศและภูมิสารสนเทศ
(ภาคเหนือ)

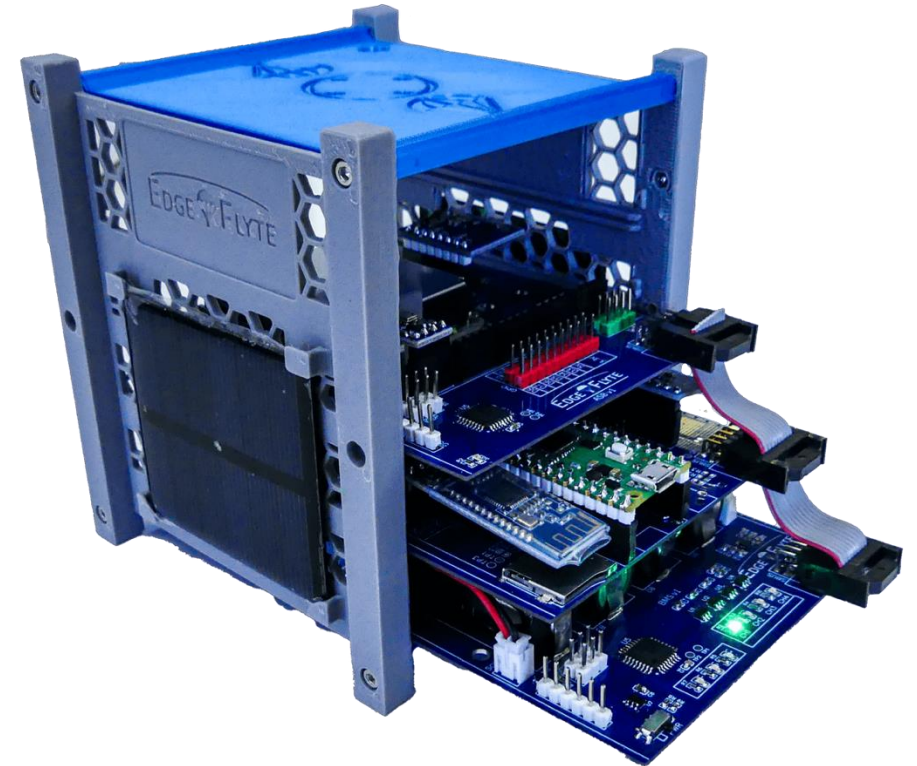


เนื้อหา

- รู้จักกับ ESP32-CAM
- ทำความเข้าใจเทคนิค Deep Learning สำหรับประมวลผลข้อมูลภาพ
- การออกแบบระบบการถ่ายภาพอัตโนมัติร่วมกับ Deep learning

ภาพรวม CUBESAT

- **CubeSat** คืออะไร?
- ดาวเทียมขนาดเล็กมาตรฐาน 1U = $10 \times 10 \times 10 \text{ cm}^3$
- น้ำหนักไม่เกิน 1.33 กิโลกรัม
- ต้นทุนต่ำเมื่อเทียบกับดาวเทียมขนาดใหญ่
- ใช้สำหรับการวิจัย การศึกษา และภารกิจเฉพาะทาง



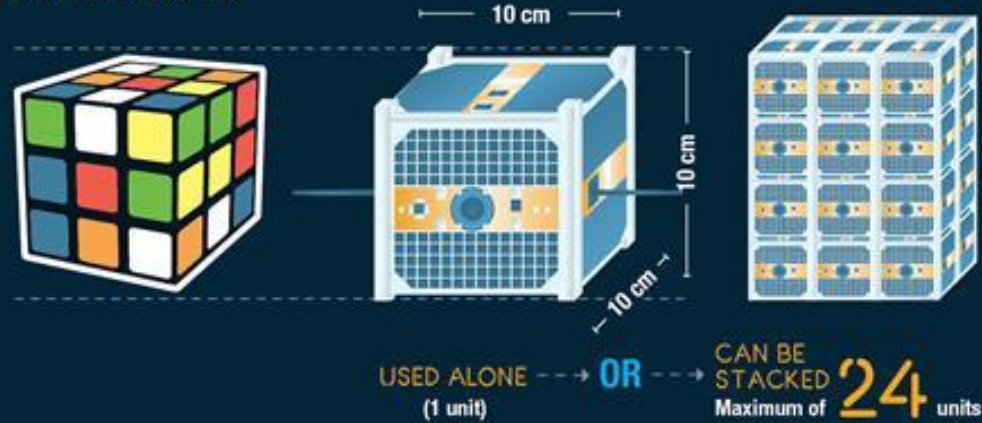


CUBESAT

IT'S HIP TO BE SQUARE!

A
CUBESAT
is a
**MINIATURE
CUBE-
SHAPED
SATELLITE.**

DIMENSIONS



ADVANTAGES

- BUILT RAPIDLY**
(within 24 months)
- SIMPLE TECHNOLOGY**
purchased off-the-shelf
- SIMPLE TO DESIGN**
- NO SPACE DEBRIS**
they burn up in the atmosphere upon reentry
- LOW COST**

4

TYPES OF MISSIONS



Technology
demonstration



Scientific
research

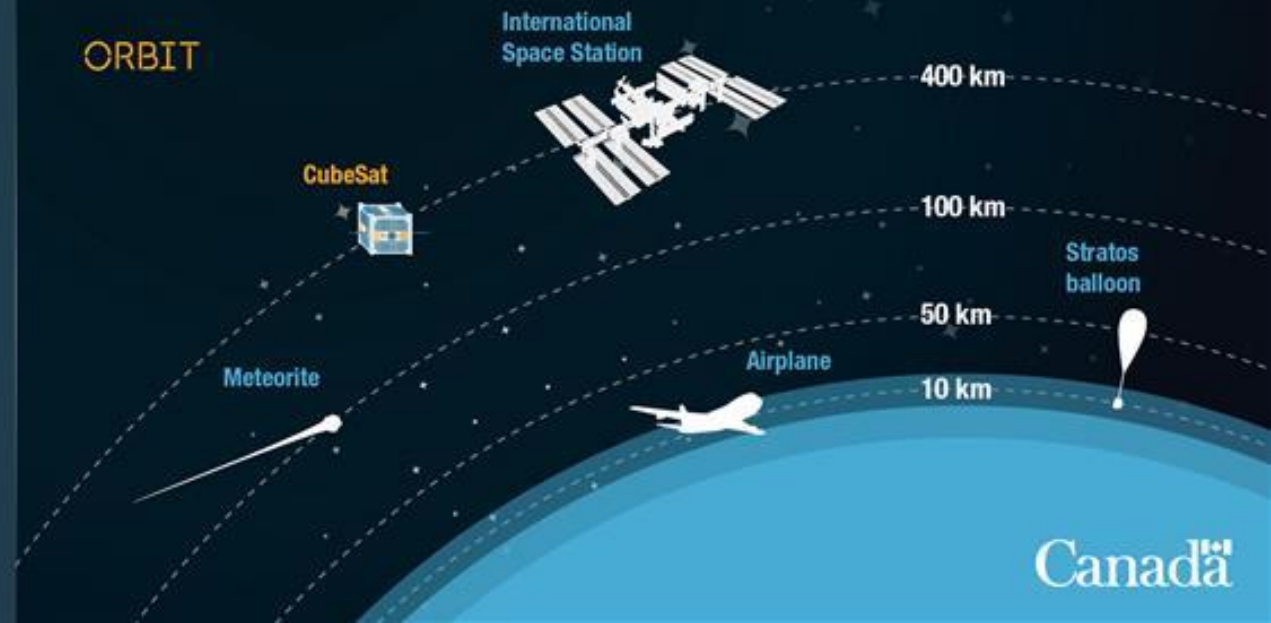


Educational
project



Commercial

ORBIT



Canadian Space
Agency

Agence spatiale
canadienne

Canada

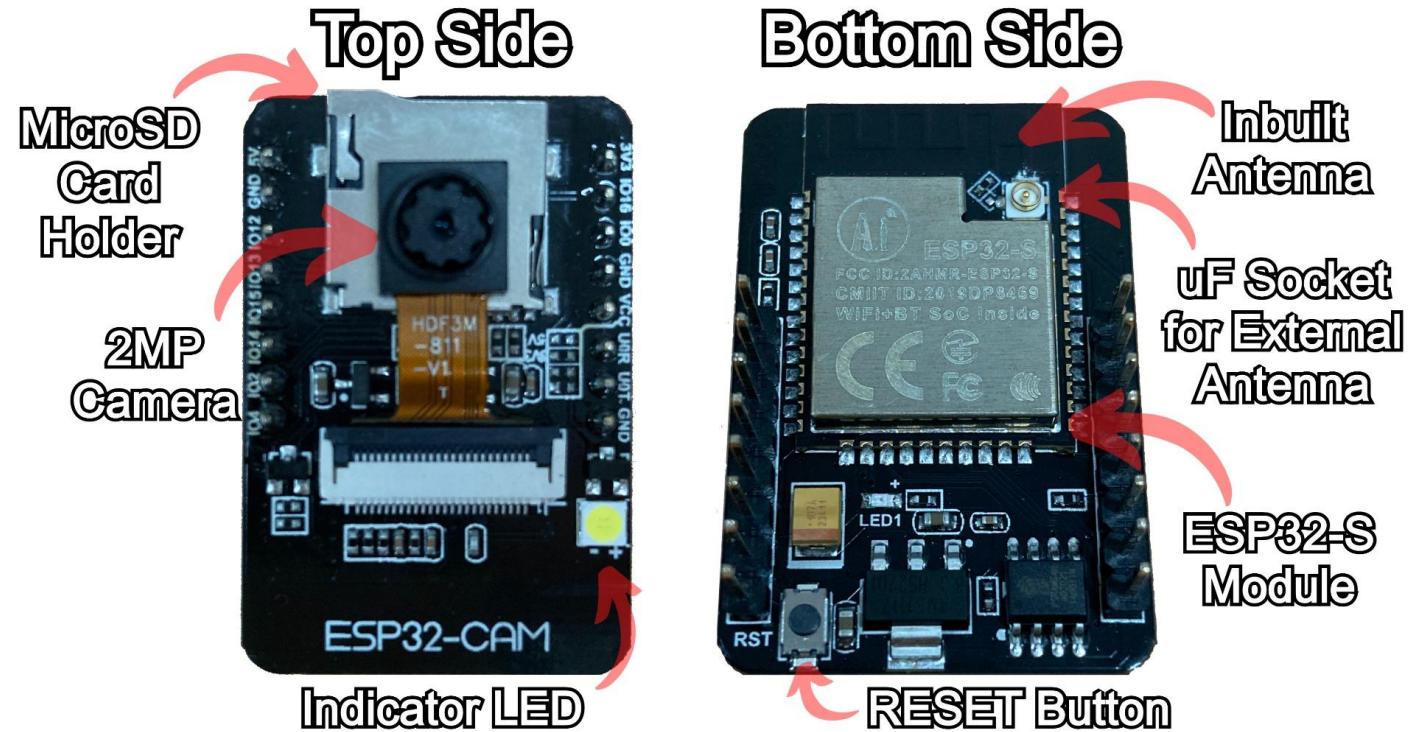
ประเภทภารกิจ CUBESAT

- **Earth Observation:** การสำรวจพื้นผิวโลก
- **Space Weather Monitoring:** ตรวจสอบสภาพอวกาศอวกาศ
- **Technology Demonstration:** ทดสอบเทคโนโลยีใหม่
- **Educational Missions:** การศึกษาและการฝึกอบรม
- **Communication:** การสื่อสารและการถ่ายทอดข้อมูล

ESP32-CAM

Module Overview

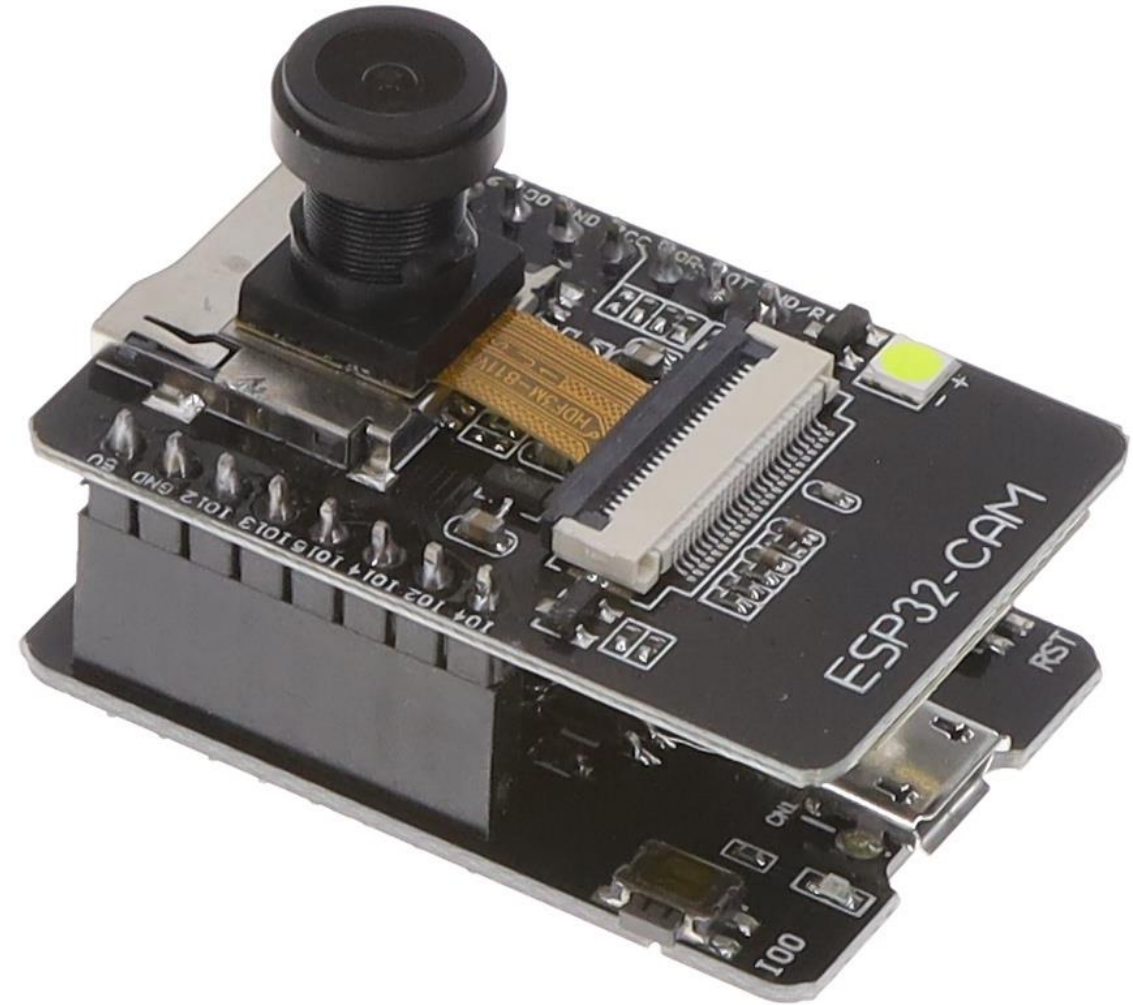
- **Microcontroller:** ESP32 dual-core processor
- **Camera:** OV2640 2MP camera sensor
- **Connectivity:** Wi-Fi, Bluetooth
- **Storage:** MicroSD card slot
- **Power:** ประหยัดพลังงาน



ESP32-CAM

Technical Specifications

- **CPU:** Xtensa dual-core 32-bit LX6 240MHz
- **Memory:** 320KB SRAM, 4MB Flash
- **Camera Resolution:** Up to 2MP (1600×1200)
- **Video:** MJPEG, H.264 encoding
- **Operating Voltage:** 3.3V
- **Power Consumption:** < 160mA (active mode)



ESP32-CAM

Advantages for Space Applications

- ขนาดเล็กกะทัดรัด - เหมาะกับข้อจำกัดพื้นที่
- ประหยัดพลังงาน - ลดการใช้แบตเตอรี่
- ราคาถูก - ลดต้นทุนภารกิจ
- ประมวลผลในตัว - ไม่ต้องส่งข้อมูลดิบทั้งหมด
- ความยืดหยุ่น - โปรแกรมได้ตามต้องการ

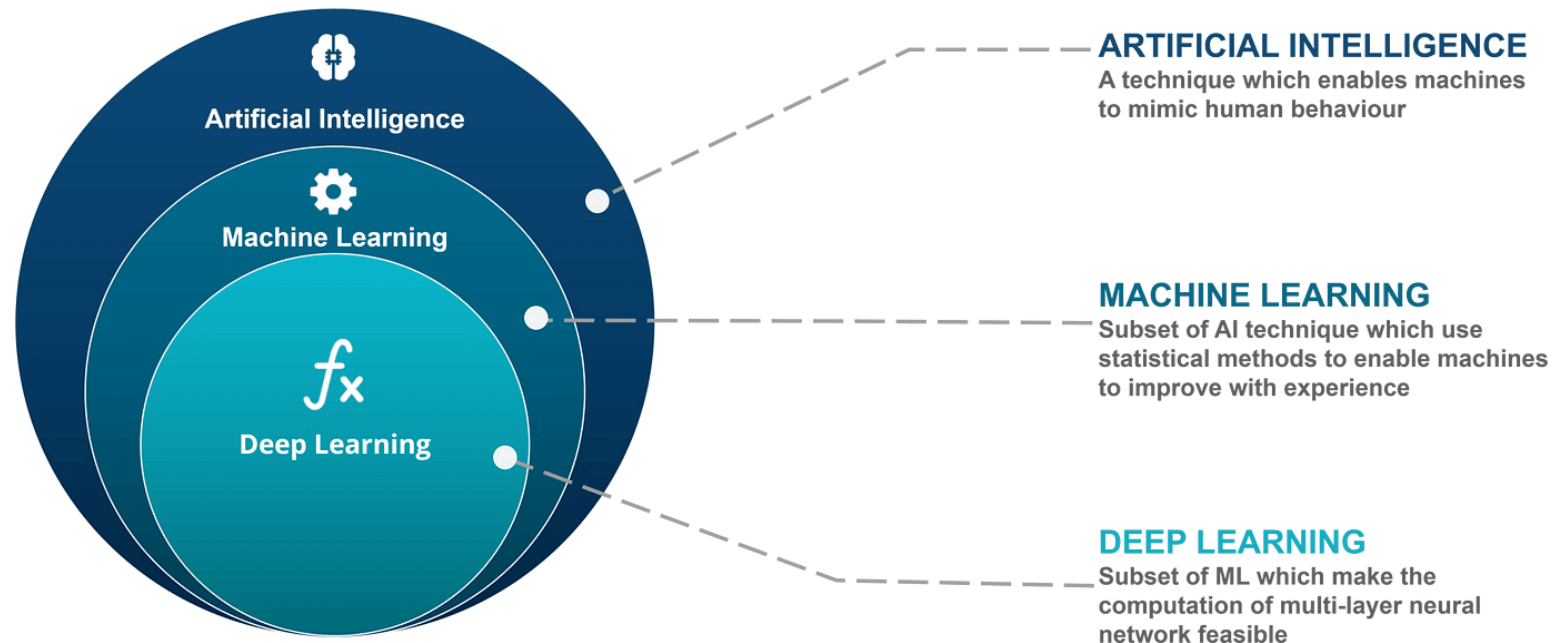
ความท้าทายในสภาพแวดล้อมนอกโลก

- **Radiation:** รังสีคอสมิกส่งผลต่อเซ็นเซอร์
- **Temperature Extremes:** -40°C ถึง $+85^{\circ}\text{C}$
- **Vacuum:** ความดันต่ำ
- **Power Constraints:** พลังงานจำกัด
- **Communication Limits:** แบนด์วิธจำกัด

AI, Machine Learning และ Deep learning

ความหมายของ AI, Machine Learning และ Deep Learning

- AI (Artificial Intelligence)
เป็นการพัฒนาเทคโนโลยีที่สามารถเลียนแบบการคิดและการตัดสินใจของมนุษย์
- Machine Learning (ML)
เป็นแขนงหนึ่งของ AI ที่ใช้ข้อมูลในการฝึกระบบให้เรียนรู้และตัดสินใจ
- Deep Learning (DL)
เป็นวิธีการเรียนรู้ที่ใช้โครงข่ายประสาทเทียม (Neural Networks) ที่ซับซ้อน

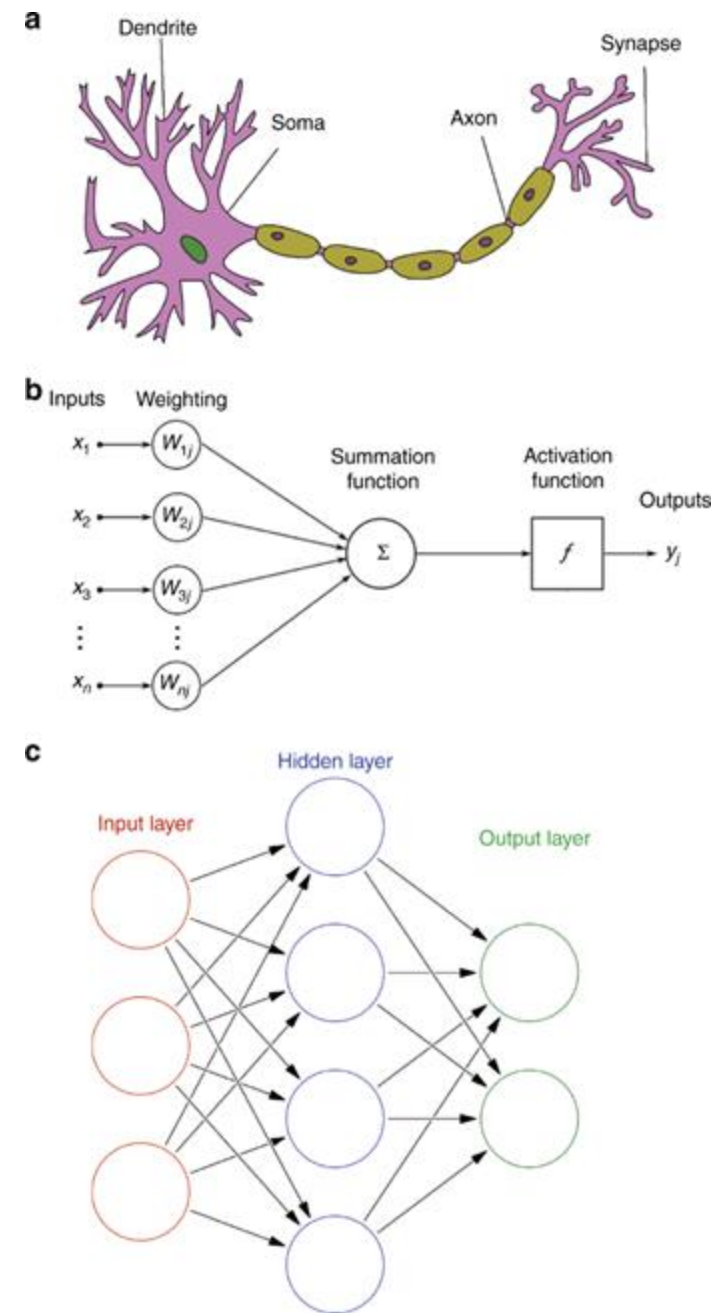


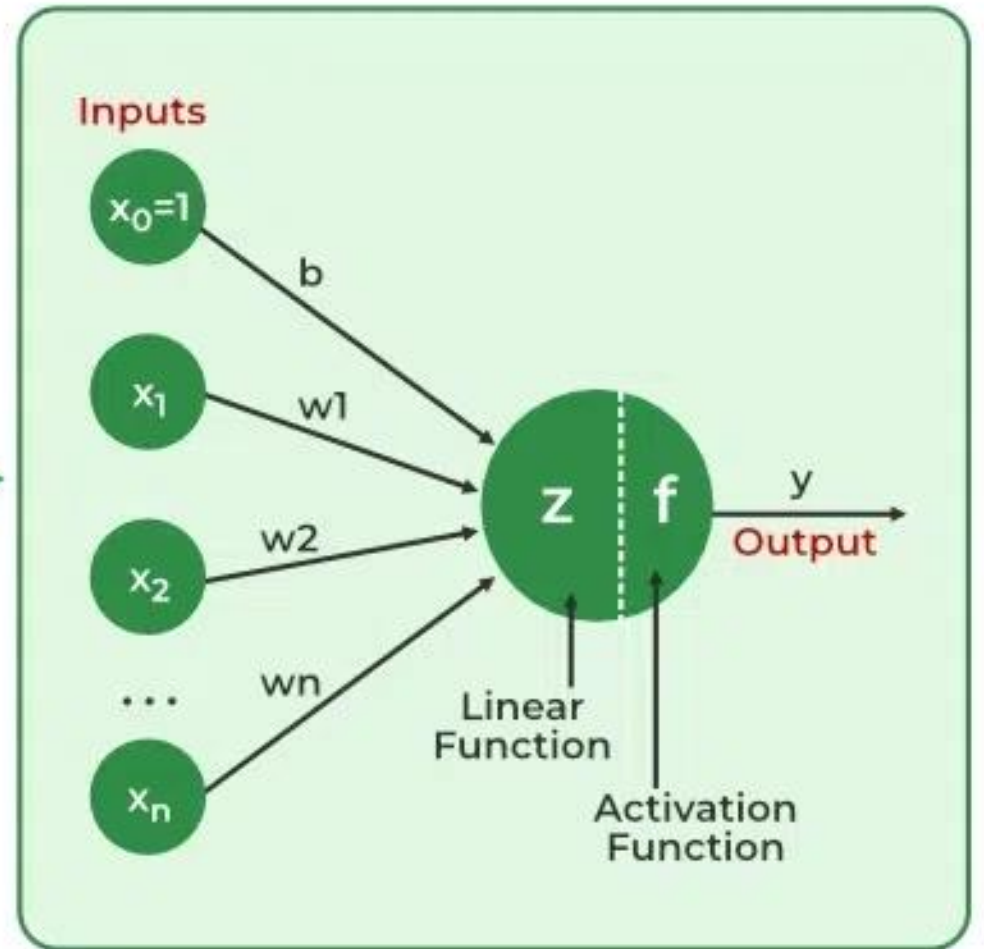
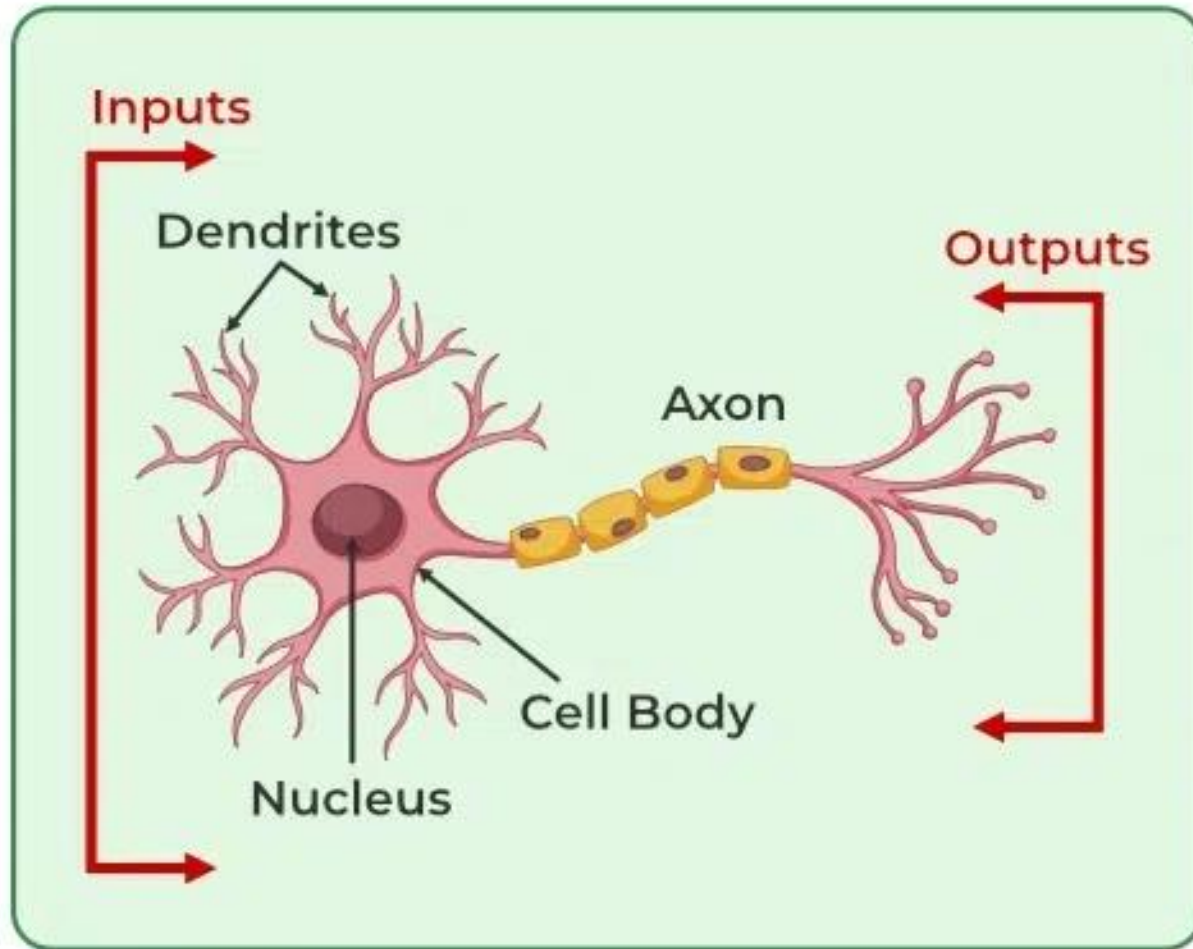
Artificial Neural Network (ANN)

โครงข่ายประสาทเทียม (ANN) คือแบบจำลองทางคอมพิวเตอร์ที่มีโครงสร้างและการทำงานคล้ายกับโครงข่ายประสาทในสมองของมนุษย์

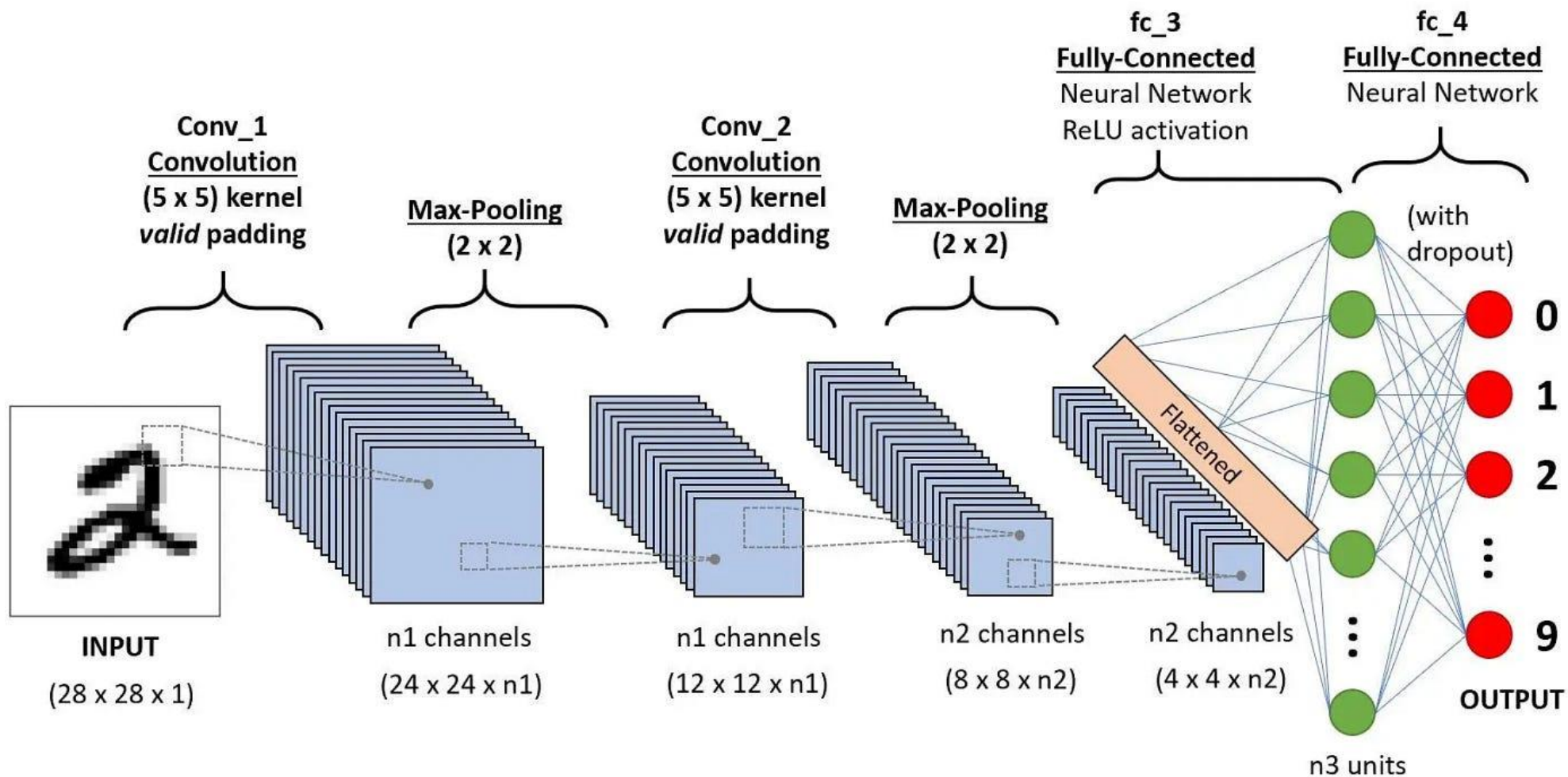
- **Neurons/Nodes:** เป็นหน่วยพื้นฐานที่รับข้อมูลเข้า (Input) ประมวลผล และส่งข้อมูลออก (Output) ไปยังเซลล์ประสาทอื่น ๆ
- **Layers:** เซลล์ประสาทจะถูกจัดเรียงเป็นชั้น ๆ
 - **ชั้นอินพุต (Input Layer)** รับข้อมูลเริ่มต้นเข้าสู่ระบบ เช่น พิกเซลของรูปภาพหรือคำในประโยค
 - **ชั้นซ่อน (Hidden Layers)** เป็นชั้นที่อยู่ระหว่างชั้นอินพุตและเอาต์พุต ที่นี้คือส่วนที่การประมวลผลและการเรียนรู้ส่วนใหญ่เกิดขึ้น แบบจำลองที่มีชั้นซ่อนหลายชั้นคือสิ่งที่เรียกว่า "Deep" ใน Deep Learning
 - **ชั้นเอาต์พุต (Output Layer)** สร้างผลลัพธ์สุดท้ายของแบบจำลอง เช่น การจำแนกประเภทของภาพหรือการคาดการณ์ตัวเลข

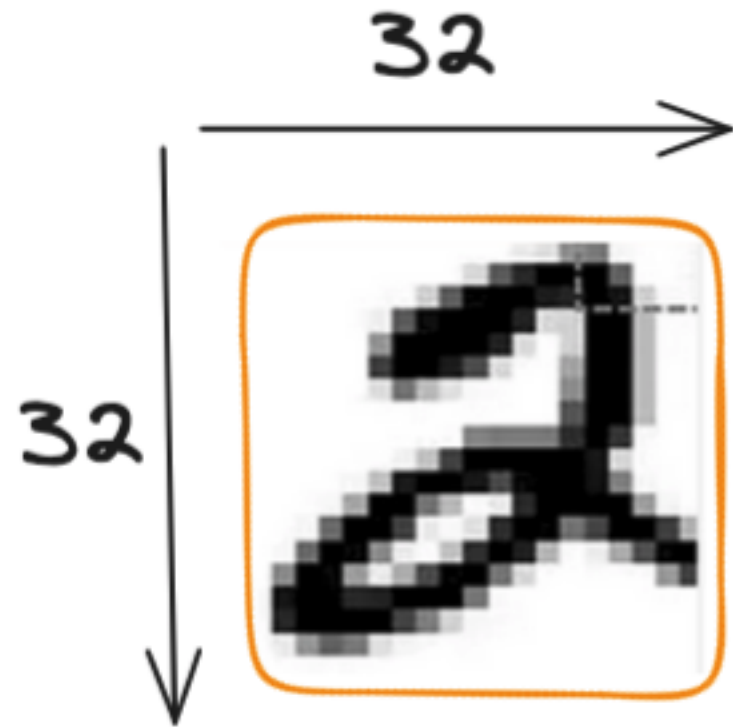
- **Connections and Weights:** การเชื่อมต่อจะมี "ค่าน้ำหนัก" กำกับอยู่ การเรียนรู้ของ ANN คือการหาค่าที่เหมาะสมที่สุดสำหรับน้ำหนักเหล่านี้





การทำงานของ CNN





Input Image



0	0	0	0	0.2	0
0	0	0	0	0.2	0.3
0	0	0.1	0	0.5	0.2
0	0.2	0	0.2	0.2	0.6
0	0.2	0	0	0.9	0.7
0	0	0	0	0	0

Pixel representation

Zoumana K.

Convolution Sign

0	0	0	0	0.2	0
0	0	0	0	0.2	0.3
0	0	0.1	0	0.5	0.2
0	0.2	0	0.2	0.2	0.6
0	0.2	0	0	0.9	0.7
0	0	0	0	0	0

Pixel representation



1	0	1
0	1	0
1	0	1

3x3 Kernel



0.1	0	1.0	0.4
0	0.5	0.4	1.6
0.3	0.2	1.7	1.1
0.2	0.4	1.1	1.7

Convolved Matrix

$$\begin{aligned} &0 \times 1 + 0 \times 0 + 0 \times 1 + \\ &0 \times 0 + 0 \times 1 + 0 \times 0 + \\ &0 \times 1 + 0 \times 0 + 0 \times 1 = 0.1 \end{aligned}$$

$$\begin{aligned} &0.2 \times 1 + 0 \times 0 + 0.2 \times 1 + \\ &0.2 \times 0 + 0 \times 1 + 0 \times 0 + \\ &0 \times 1 + 0 \times 0 + 0 \times 1 = 0.4 \end{aligned}$$

Zoumana K.

- Edge Detection Kernels:

- Sobel Filter
- Laplacian Filter

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$
$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

- Sharpening Kernels

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

- Smoothing (Blurring) Kernels

$$\begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$

Max Pooling

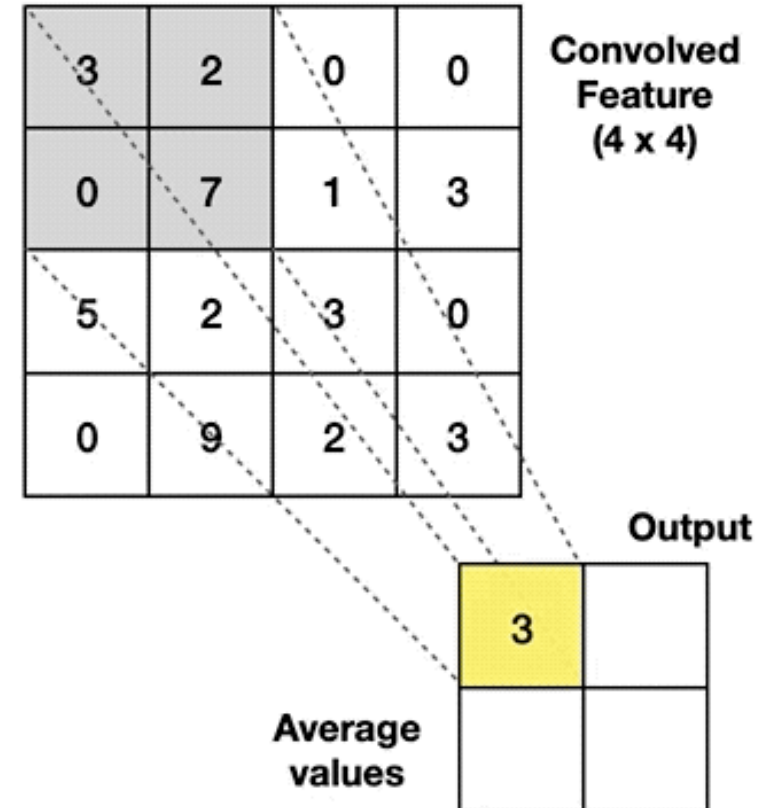
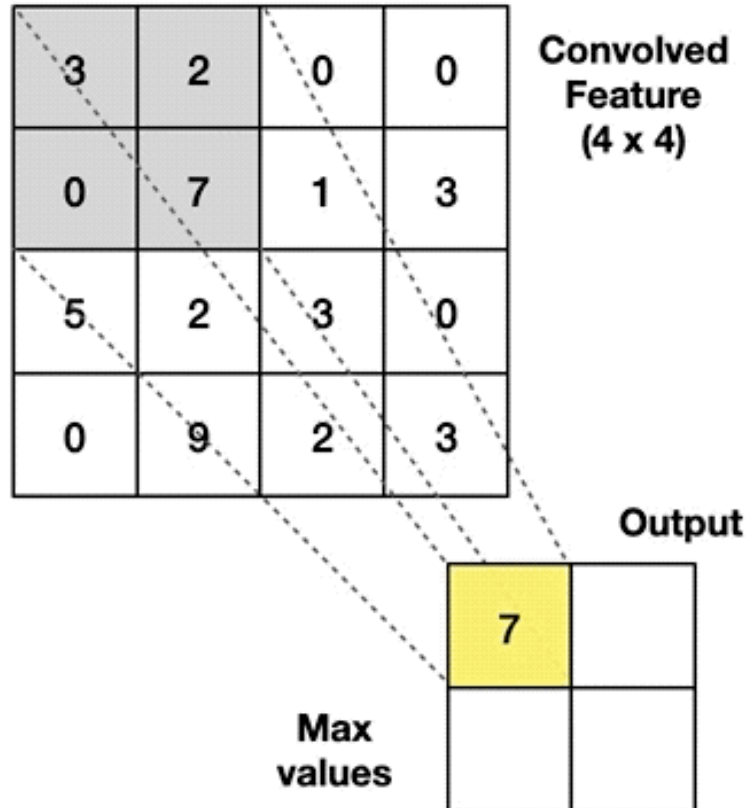
Take the **highest** value from the area covered by the kernel

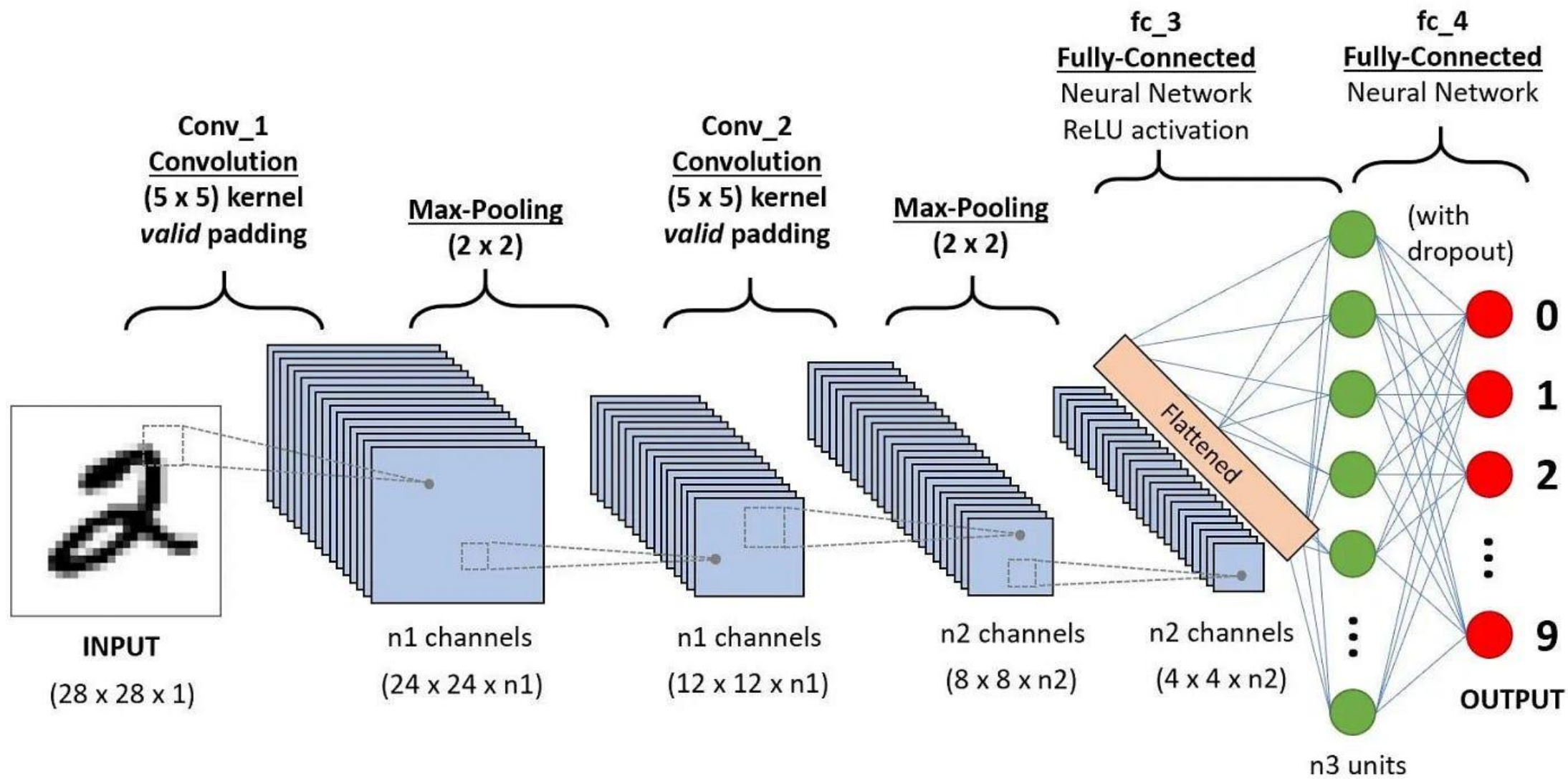
Average Pooling

Calculate the **average** value from the area covered by the kernel

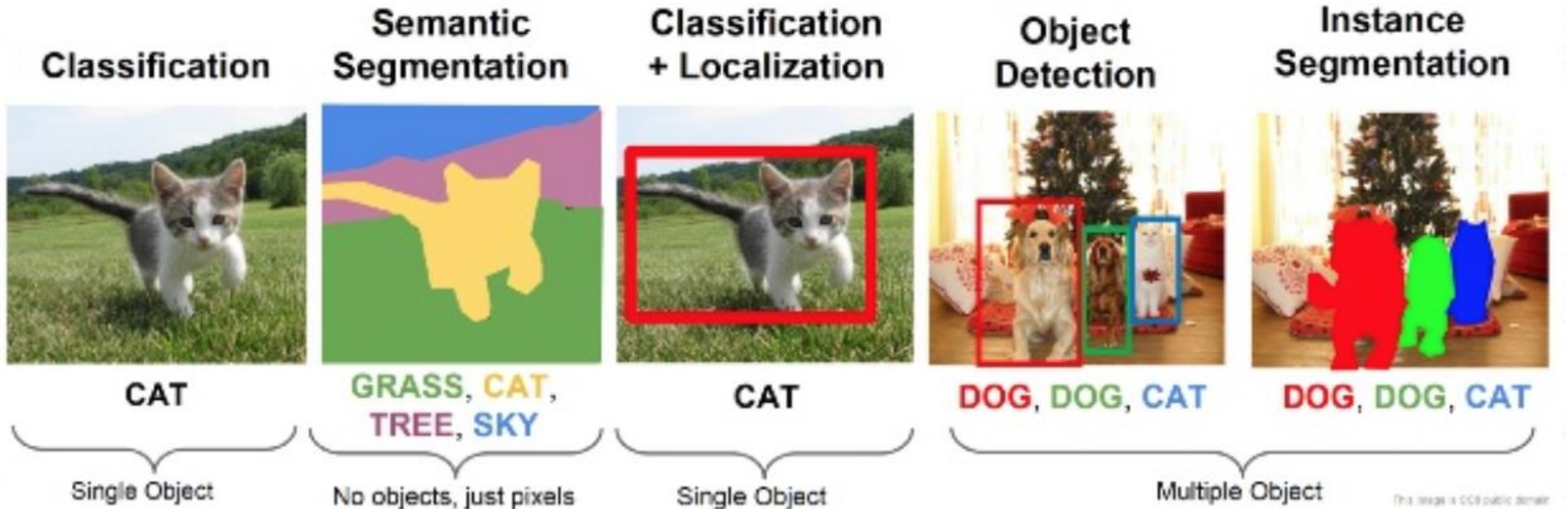
- Pooling

Example: Kernel of size 2 x 2; stride=(2,2)





Deep Learning สำหรับการประมวลผลภาพ



YOLO

J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91.

YOLO (ย่อมาจาก "You Only Look Once") เป็นอัลกอริทึมสำหรับตรวจจับวัตถุในภาพที่ได้รับความนิยมอย่างมากเนื่องจากความเร็วและความแม่นยำสูง โดยหลักการทำงานของ YOLO มีดังนี้

1. การแบ่งภาพออกเป็นตาราง (Grid)

ภาพที่ป้อนเข้ามาจะถูกแบ่งออกเป็นตาราง (เช่น 13×13 , 19×19 หรืออื่น ๆ ขึ้นอยู่กับรุ่น) แล้วแต่ละ cell ของตารางจะรับผิดชอบการตรวจจับวัตถุที่อยู่ในบริเวณนั้น

2. การทำนาย bounding boxes และความน่าจะเป็น (Confidence)

แต่ละ cell จะทำนาย bounding boxes หลายอันพร้อมกับความน่าจะเป็นที่บอกว่าใน bounding box นั้นมีวัตถุอยู่จริงหรือไม่ รวมถึงการทำนายประเภทของวัตถุ (class probabilities)

3. การลดการทับซ้อน (Non-Maximum Suppression)

เมื่อมีการทำนาย bounding boxes หลายอันที่มีการซ้อนทับกัน YOLO จะใช้เทคนิค Non-Maximum Suppression เพื่อลด bounding box ที่ซ้ำซ้อนและเลือกเฉพาะอันที่มีคะแนนสูงสุด

4. การประมวลผลแบบ end-to-end

ทั้งหมดนี้ทำในขั้นตอนเดียว (one forward pass) ซึ่งทำให้ YOLO มีความเร็วสูงและสามารถนำไปใช้งานแบบ real-time ได้

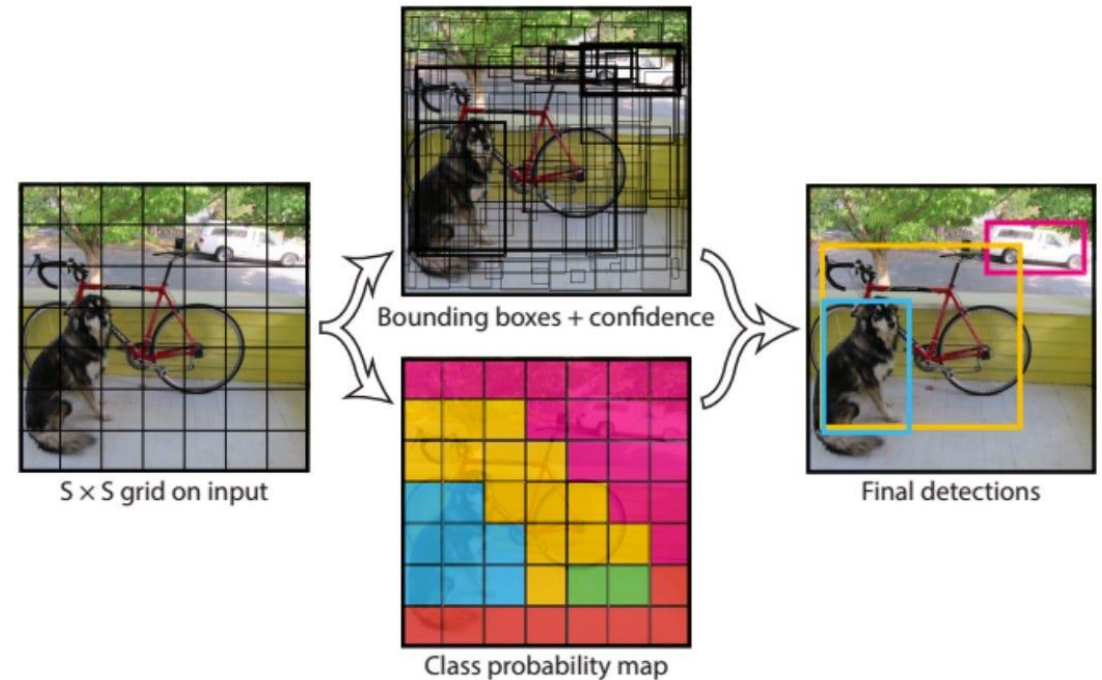
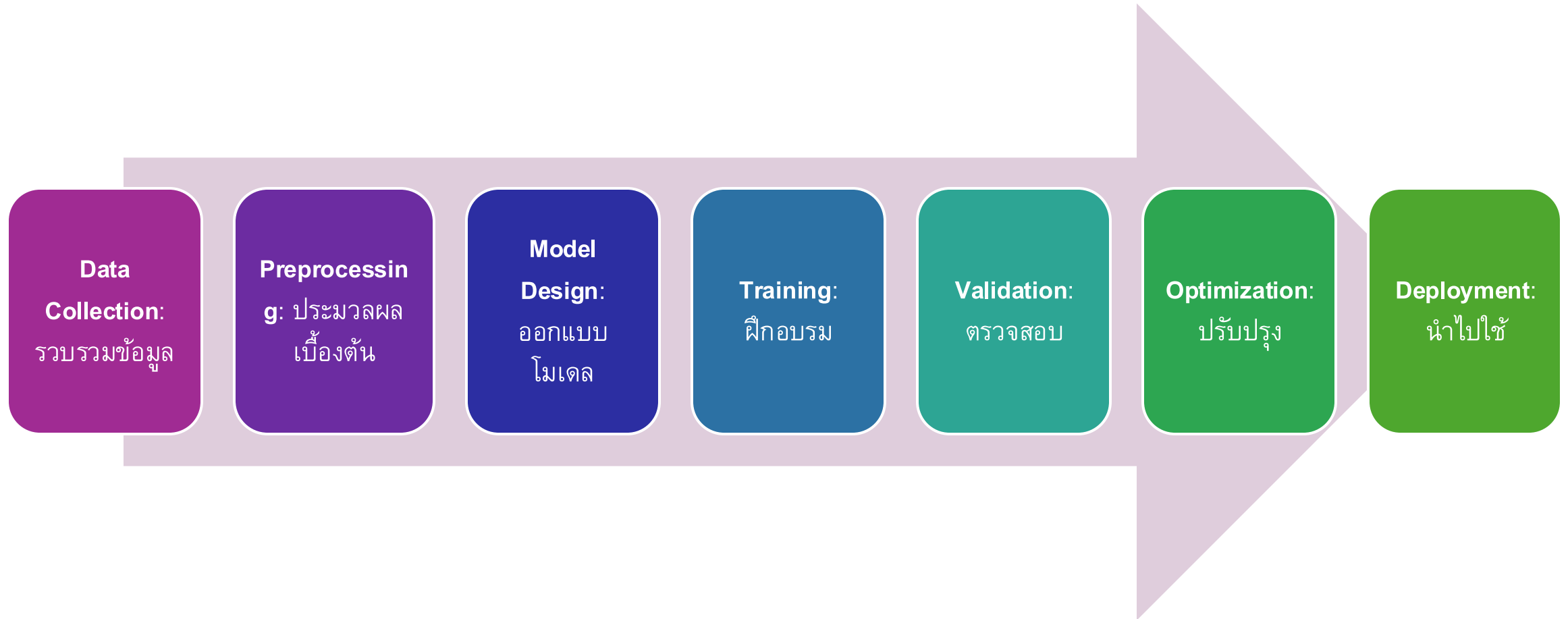


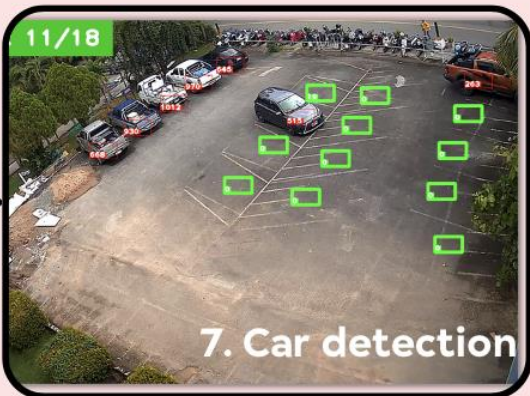
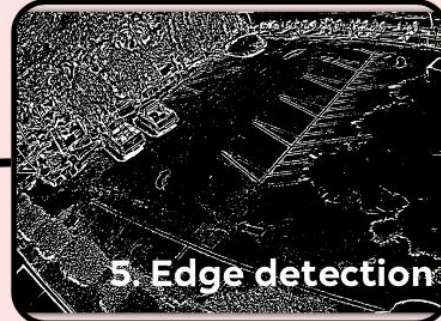
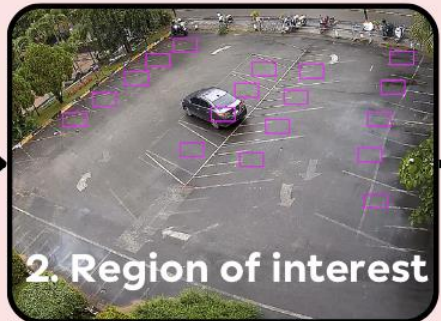
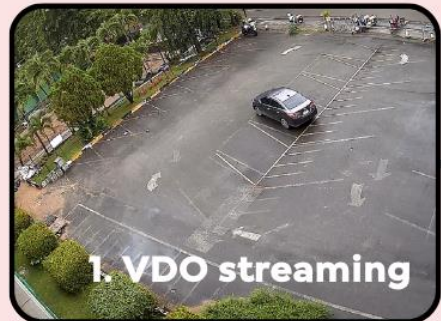
Figure 2: The Model. Our system models detection as a regression problem. It divides the image into an $S \times S$ grid and for each grid cell predicts B bounding boxes, confidence for those boxes, and C class probabilities. These predictions are encoded as an $S \times S \times (B * 5 + C)$ tensor.

ขั้นตอนการพัฒนาโมเดล



ตัวอย่างการประยุกต์ใช้

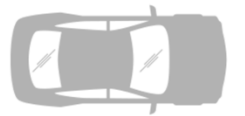
- **Cloud Detection:** ตรวจจับเมฆ
- **Land Use Classification:** จำแนกการใช้ประโยชน์ที่ดิน
- **Disaster Monitoring:** ตรวจสอบภัยพิบัติ
- **Vegetation Analysis:** วิเคราะห์พืชพรรณ
- **Ocean Monitoring:** ตรวจสอบมหาสมุทร



[จอด] ดี

ระบบสนับสนุนการตัดสินใจสำหรับช่วยหาช่องจอดรถ
A Decision Support System for Available Parking Lot

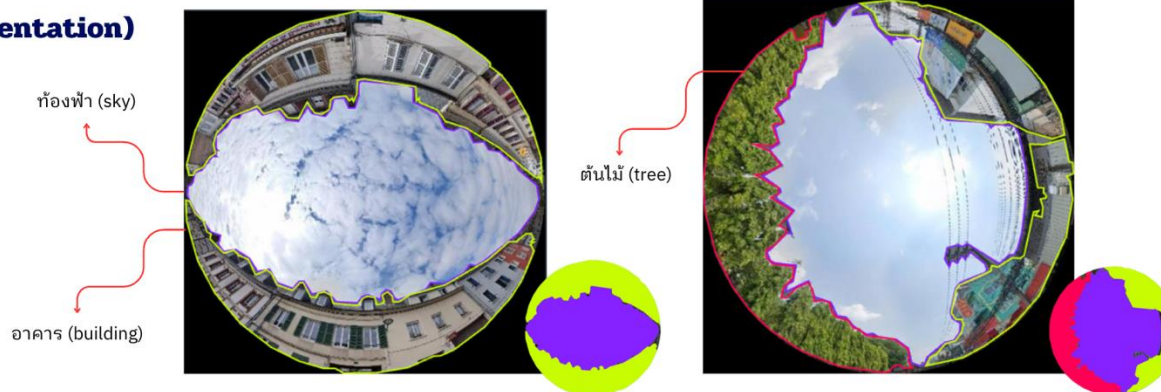
โดย ชัชวาล คำคุณา*
และ ศักดิ์ธิดา หอมหวล





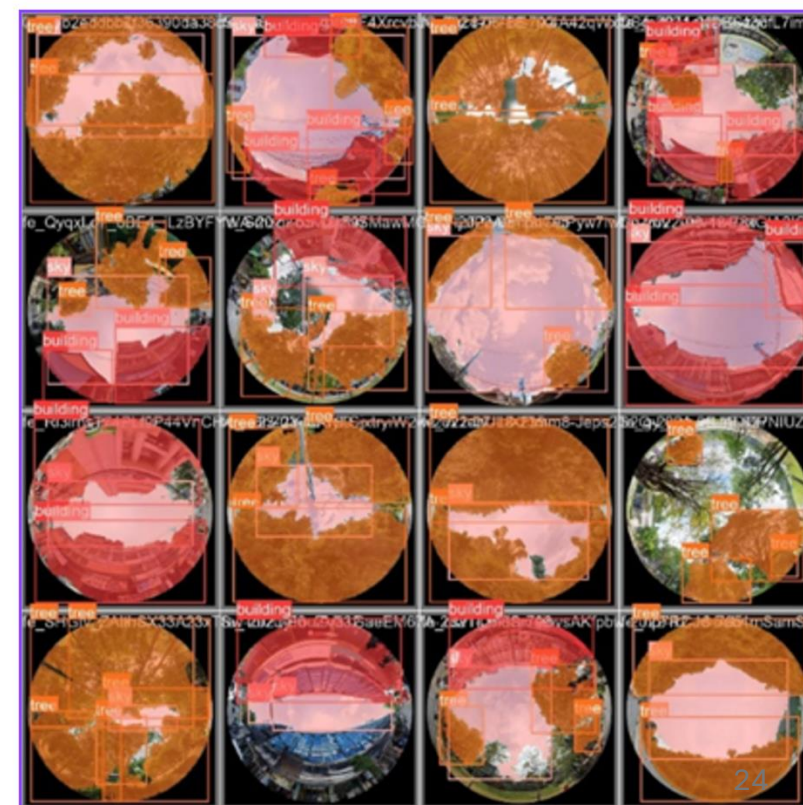
Label (image segmentation)

- ท้องฟ้า (sky)
- ต้นไม้ (tree)
- อาคาร (building)



“เธอเห็นท้องฟ้านั้นไหม” การทำนายอุณหภูมิเมืองด้วยสัดส่วนการมองเห็นท้องฟ้าจากข้อมูล Google Street View (GSV) และ deep learning

กัญญาพัชร ดวงใจ





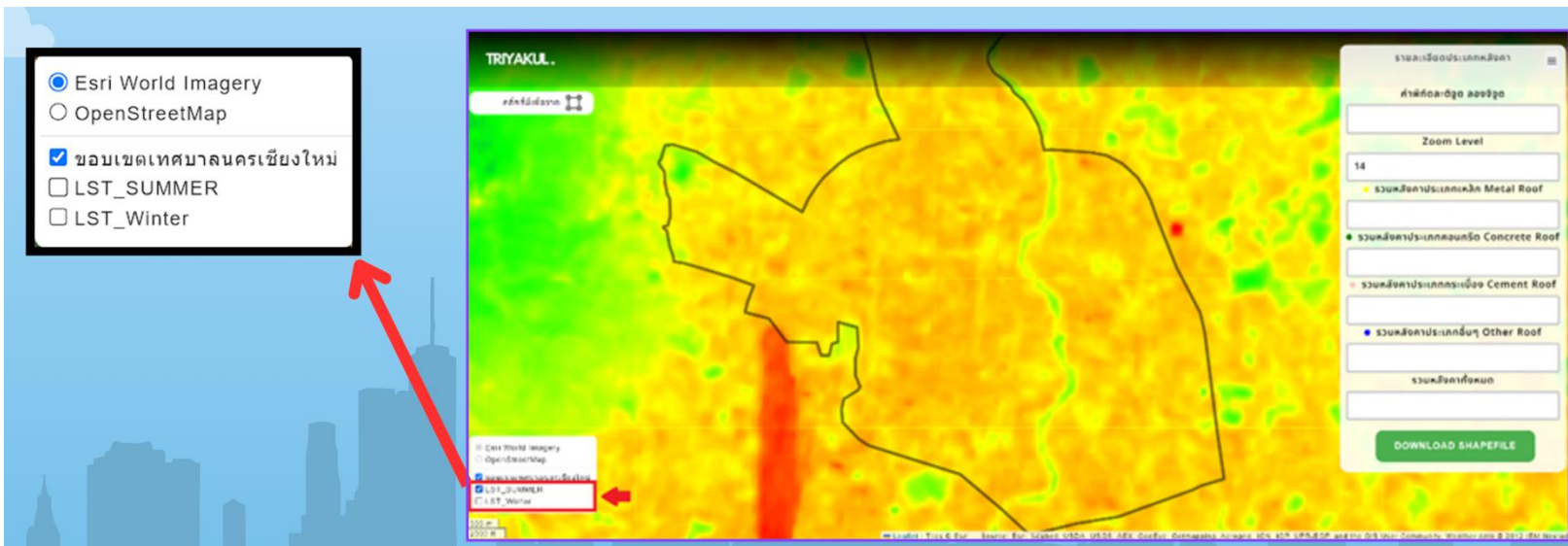
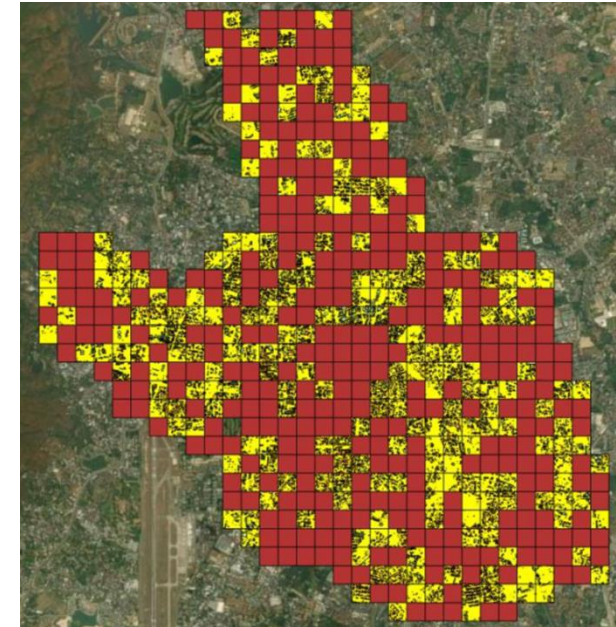
Model_Google



Model_Bing

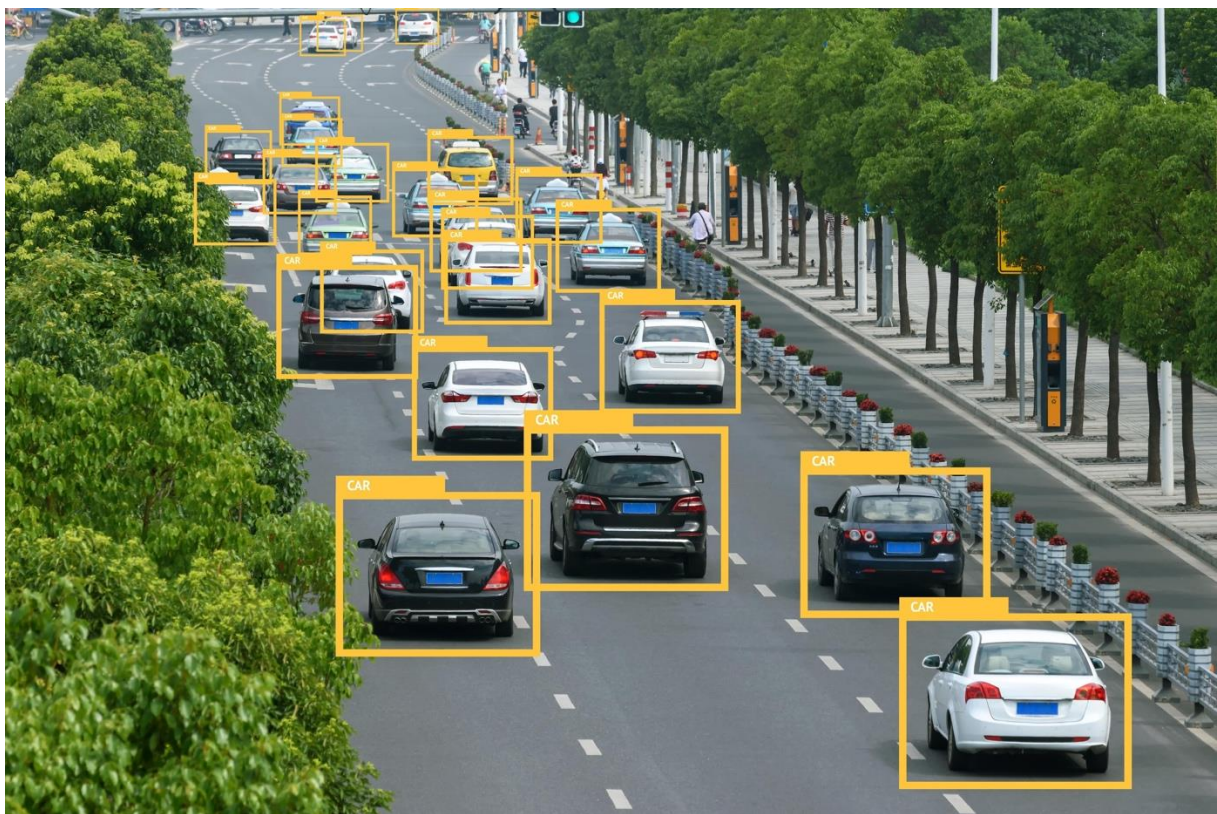


Model_MIX

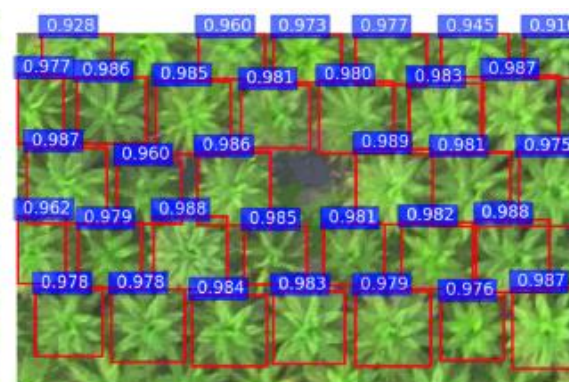


“หลังคากับความร้อน” การวิเคราะห์วัสดุหลังคาและอุณหภูมิด้วย Deep Learning

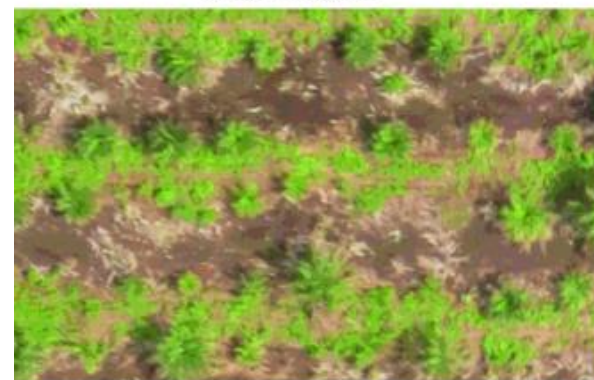
กษิตศ ตริยกุล



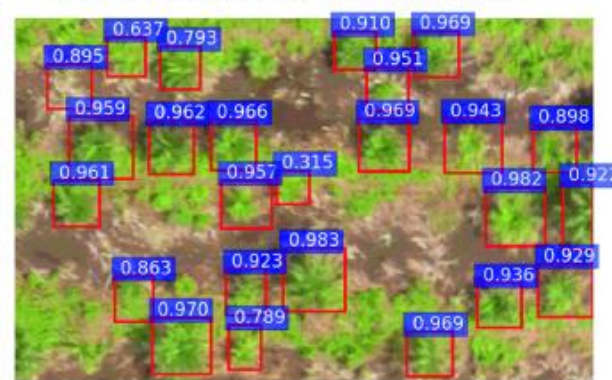
(a) Image a



(b) Detection result of image (a)

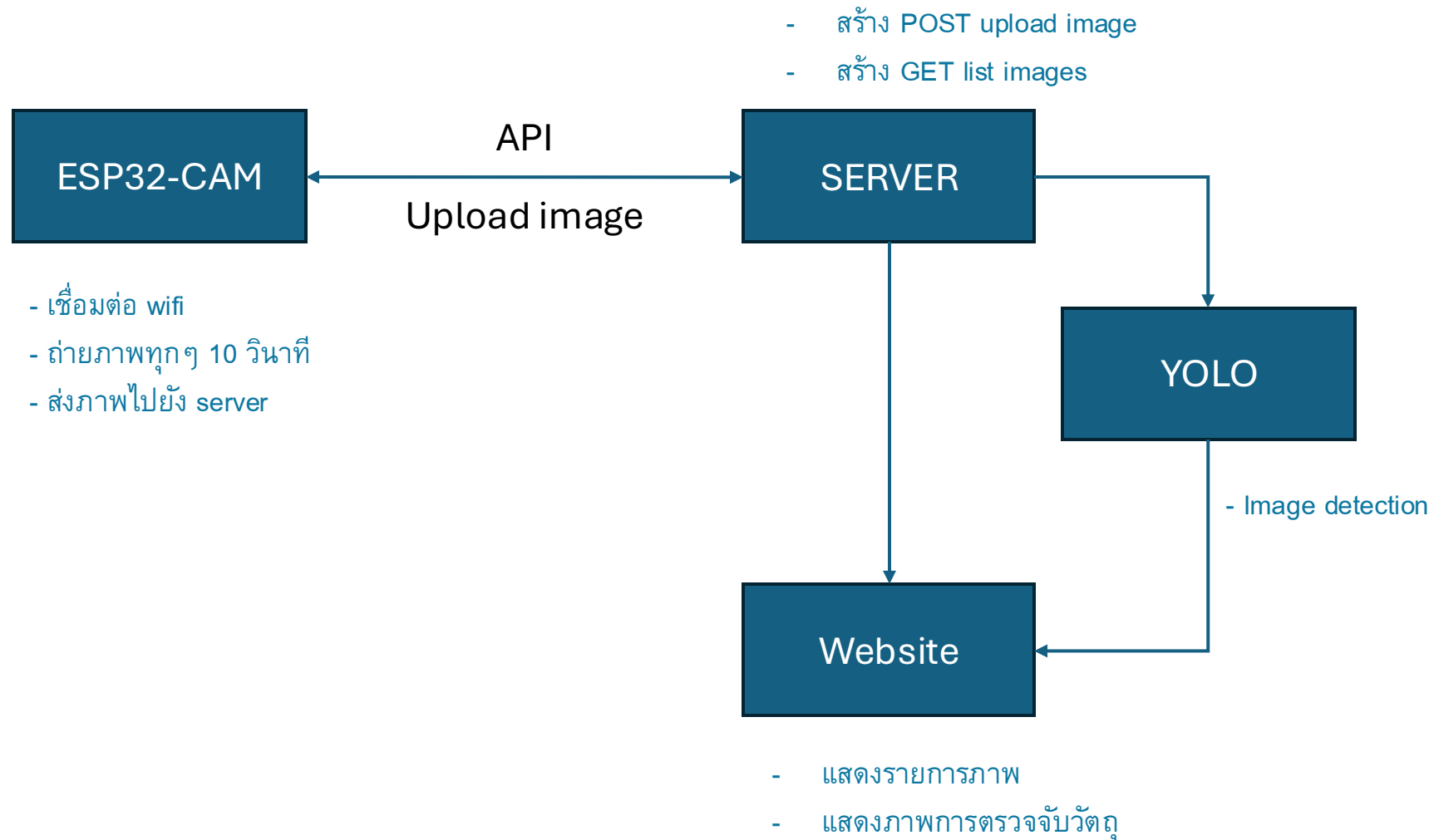


(c) Image c



(d) Detection result of image (c)

ขั้นตอนการทำงานของ ESP32-CAM กับ YOLO



สรุป

- แนวทางการใช้ ESP32-CAM สำหรับ CUBESAT
- Deep Learning เพิ่มความสามารถในการวิเคราะห์ข้อมูลภาพถ่าย
- การทำงานร่วมกันของ ESP32-CAM และ Deep Learning
- ESP32-CAM และ Deep Learning สามารถนำไปใช้งานด้านต่างๆ

เอกสารปฏิบัติการ

- <https://cmu.to/Y8Cie>

