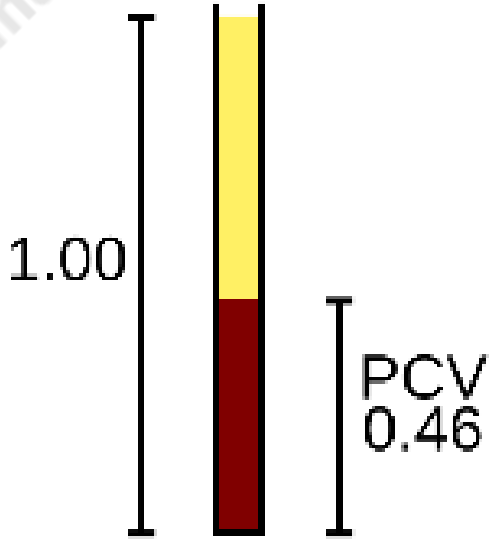
บทที่ 2

ทฤษฎีและระบบงานที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1.1 ค่า PCV (Packed Cell Volume)

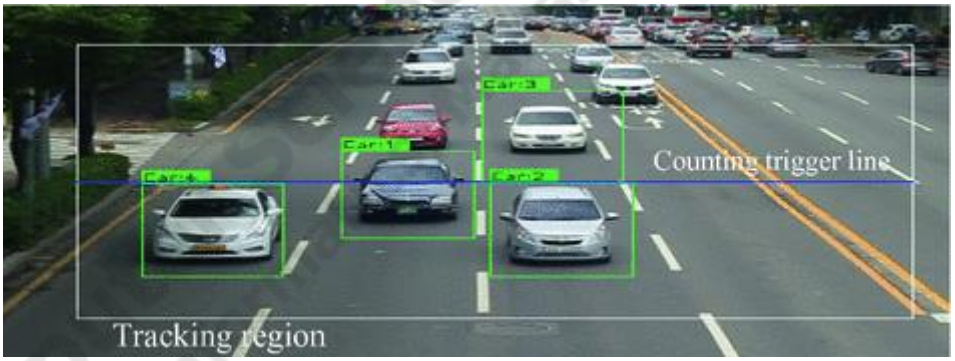
ค่า Packed Cell Volume (PCV) [8] สามารถหาได้โดยการปั่นตกของเลือดที่ใส่สารเฮพารินในหลอด capillary tube (หรือเป็นที่รู้จักในชื่อ Micro Hematocrit Tube) ที่ความเร็ว 10,000 RPM เป็นเวลา 5 นาที[2] ซึ่งจะทำให้เลือดแบ่งออกเป็น 2 ชั้น ปริมาณของเม็ดเลือดแดงหารด้วยปริมาณทั้งหมดของเลือด คือ ค่า PCV เนื่องจากเราทำการวิเคราะห์โดยใช้หลอดทดลอง ดังนั้น เราจึงสามารถคำนวณโดยอาศัยการวัดความยาวของชั้นได้สำหรับเครื่องมือสมัยใหม่ ค่า HCT สามารถคำนวณโดยเครื่องอัตโนมัติซึ่งเป็นการวัดแบบทางอ้อม โดยค่า HCT หาได้จากการเอาจำนวนเม็ดเลือดแดงคูณปริมาตรของเม็ดเลือดแดงโดยเฉลี่ย ค่า %HCT มักจะมีค่าเป็นสามเท่าของความเข้มข้นของฮีโมโกลบิน [3] ค่า HCT อาจจะมีความคลาดเคลื่อนได้หากมีการให้ของเหลว เช่น เลือด น้ำเกลือ ผ่านทางหลอดเลือดดำ ยกตัวอย่างเช่น ถ้าหากมีการให้ packed red cells กับผู้ป่วย จะทำให้ปริมาณของเซลล์เม็ดเลือดแดงมีความเข้มข้นมากขึ้น ดังนั้น ค่า HCT จึงมีค่าที่สูงกว่าความเป็นจริง หรือในกรณีที่ผู้ป่วยให้น้ำเกลือหรือของเหลวอื่น ๆ ก็จะทำให้เลือดถูกเจือจางส่งผลให้ค่า HCT ที่วัดได้นั้นมีค่าต่ำกว่าความเป็นจริง



ภาพประกอบที่ 2.1 Packed cell volume diagram
ที่มา : https://th.wikipedia.org/wiki/ฮีมาโทคริต

2.1.1.2 **การประมวลผลภาพ**

การประมวลผลภาพ (Image Processing) [1] คือ เป็นการประยุกต์ใช้งานการประมวลผลสัญญาณบนสัญญาณ 2 มิติ เช่น ภาพนิ่ง (ภาพถ่าย) หรือภาพวีดิทัศน์(วิดีโอ)และยังรวมถึงสัญญาณ 2 มิติอื่น ๆ ที่ไม่ใช่ภาพด้วย แนวความคิดและเทคนิค ในการประมวลผลสัญญาณ สำหรับสัญญาณ 1 มิตินั้น สามารถปรับมาใช้กับภาพได้ไม่ยาก แต่นอกเหนือจาก เทคนิคจากการประมวลผลสัญญาณแล้ว การประมวลผลภาพก็มีเทคนิคและแนวความคิดที่เฉพาะ (เช่น Connectivity และ Rotation Invariance) ซึ่งจะมีความหมายกับสัญญาณ 2 มิติเท่านั้น แต่อย่างไรก็ตามเทคนิคบางอย่าง จากการประมวลผลสัญญาณใน 1 มิติ จะค่อนข้างซับซ้อนเมื่อนำมาใช้กับ 2 มิติ เมื่อหลายสิบปีมาแล้ว การประมวลผลภาพนั้น จะอยู่ในรูปของการประมวลผลสัญญาณแอนะล็อก (Analog) โดยใช้อุปกรณ์ปรับแต่งแสง (Optics) ซึ่งวิธีเหล่านั้นก็ไม่ได้หายสาบสูญ หรือเลิกใช้ไป ยังมีใช้เป็นส่วนสำคัญ สำหรับการประยุกต์ใช้งานบางอย่าง เช่น ฮอโลกราฟี (Holography) แต่เนื่องจากอุปกรณ์คอมพิวเตอร์ในปัจจุบัน ราคาถูกลง และเร็วขึ้นมาก การประมวลผลภาพดิจิทัล (Digital Image Processing) จึงได้รับความนิยมมากกว่า เพราะการประมวลผลที่ทำได้ซับซ้อนขึ้น แม่นยำ และง่ายในการลงมือปฏิบัติ



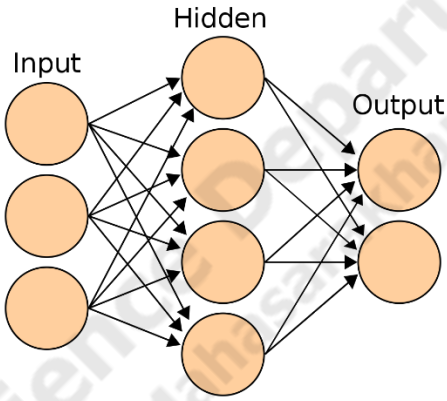
ภาพประกอบที่ 2.2 image processing-trafic

ที่มา : https://medium.com/tni-university/image-processing-981c65c26289

2.1.1.3 **โครงข่ายประสาทเทียม**

โครงข่ายประสาทเทียม (Artificial Neural Networks: ANN) หรือ ข่ายงานประสาทเทียม (Connectionist Systems) [2] คือระบบคอมพิวเตอร์จากโมเดลทางคณิตศาสตร์ เพื่อจำลองการทำงานโครงข่ายประสาทชีวภาพที่อยู่ในสมองของสัตว์ โครงข่ายประสาทเทียมสามารถเรียนรู้ที่จะทำงานที่มอบหมายได้ จากการเรียนรู้ผ่านตัวอย่าง โดยไม่ถูกโปรแกรมด้วยกฎเกณฑ์ตายตัวแบบระบบอัตโนมัติ ยกตัวอย่างเช่น ในการประมวลผลภาพ คอมพิวเตอร์ที่ทำงานด้วยระบบโครงข่ายประสาทเทียมจะเรียนรู้การจำแนกรูปภาพแมวได้จากการให้ตัวอย่างรูปภาพที่กำกับโดยผู้เขียนโปรแกรมว่า “เป็นแมว” หรือ “ไม่เป็นแมว” จากนั้นนำผลลัพธ์ที่ได้ไปใช้ระบุภาพแมวในตัวอย่างรูปภาพอื่น ๆ โปรแกรมโครงข่าย

ประสาทเทียมสามารถแยกแยะรูปภาพแมวได้โดยปราศจากการความรู้ก่อนหน้า ว่า ”แมว” คืออะไร (อาทิ แมวมีขน มีหูแหลม มีเขี้ยว มีหาง) แทนที่จะใช้ความรู้ดังกล่าว โครงข่ายประสาทเทียมทำการระบุตัวแมวโดยอัตโนมัติด้วยการระบุลักษณะเฉพาะ จากชุดตัวอย่างที่เคยได้ประมวลผล
การประมวลผลต่าง ๆ ของโครงข่ายประสาทเทียมเกิดขึ้นในหน่วยประมวลผลย่อย เรียกว่า โหนด (node) ซึ่งโหนดเป็นการจำลองลักษณะการทำงานมาจากเซลล์การส่งสัญญาณ ระหว่างโหนดที่เชื่อมต่อกัน จำลองมาจากการเชื่อมต่อของใยประสาท และแกนประสาทในระบบประสาทของสมองมนุษย์ ภายในโหนด จุดเชื่อมต่อแต่ละจุด มีความคล้ายคลึงกับจุดประสานประสาท (Synapses) ในสมอง มีความสามารถในการส่งสัญญาณไปยังเซลล์ประสาทเซลล์อื่น ๆ ที่เชื่อมต่อกับมันได้



ภาพประกอบที่ 2.3 ข่ายงานประสาทเทียมมีการเชื่อมต่อกันผ่านกลุ่มโหนด
ที่มา : https://th.wikipedia.org/wiki/โครงข่ายประสาทเทียม

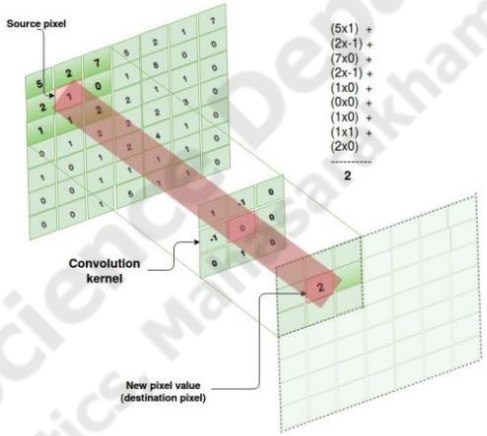
2.1.1.4 **การเรียนรู้เชิงลึก**

Deep Learning [3] หรือการเรียนรู้เชิงลึก เป็นหนึ่งในฟังก์ชันของปัญญาประดิษฐ์ Artificial intelligence(AI) ที่เรียนแบบการทำงานของสมองมนุษย์ในกระบวนการประมวลผลข้อมูลและเป็นการสร้างรูปแบบ สำหรับใช้ในการตัดสินใจ นอกจากนี้ เป็นจักรของ Machine ซึ่งการเรียนรู้เชิงลึก หรือการเรียนรู้ด้วยเครื่อง ซึ่งเป็นลำดับชั้นของเครือข่ายประสามเทียม (Artificial Neural Network) โดยดำเนินการด้วย Matcine Learning มี Nodes เชื่อมต่อกันเหมือนเว็บไซต์ แม้ว่าโปรแกรมแบบเก่าจะสร้างการวิเคราะห์ข้อมูลเชิงเส้น โดยฟังก์ชันลำดับชั้นของระบบ Deep Learning ช่วยให้เครื่องประมวลผลข้อมูลด้วยวิธีการไม่ใช่เชิงเส้น ซึ่งการเรียนรู้เชิงลึก เพื่อกำจัดสิ่งแปลกปลอมในการทำธุรกรรมที่เสี่ยงต่อการหลอกลวงจะประกอบด้วยเวลา

2.1.1.5 **โครงข่ายประสาทแบบคอนโวลูชันนอล**

โครงข่ายประสาทแบบคอนโวลูชัน (Convolutional Neural Network: CNN) [4] โครงข่าย architecture สถาปัตยกรรม Feed-Forward Neural Networks จัดเป็นการเรียนรู้เชิงลึก (Deep Learning) เช่นกัน โดยจะจำลองการมองเห็นของมนุษย์ในพื้นที่ย่อย จากการแยกแยะคุณลักษณะเชิง

ภาพ เช่น สี ลายเส้น และอื่น ๆ จากนั้นนำมาผสานกันเพื่อที่จะทำนายว่าภาพนั้นคือภาพอะไร ในการทำงานของ Convolutional Neural Network (CNN) จุดประสงค์ของการทำคอนโวลูชันเพื่อต้องการระบุคุณลักษณะที่สำคัญและเกี่ยวข้องกับภาพ โดยขั้นตอนนี้จะทำการสร้าง Sliding window (Filter) มาสแกนรูปภาพเพื่อแยก องค์ประกอบต่าง ๆ เช่น รูปทรงของเส้นขอบ สี โดยปกติภาพจะมี สีหลักๆ 3 สี คือ สีแดง สีน้ำเงิน และสีเขียว แบ่งเป็น 3 แชนแนล (Channel) ซึ่งแต่ละพิกเซลสามารถแทนค่าด้วยตัวเลขเพื่อบอก ความเข้มของสี ตั้งแต่ 0-255ในการทำภาพขาวดำจะใช้เพียง 1 แชนแนล สีดำแทนด้วยเลข 1 และสีขาวแทนด้วยเลข 0 จากนั้นจะใช้ Filter เป็นตัวกรอง จากรูปที่ 1 (เมทริกซ์สีเหลือง ขนาด 3x3) ทำการ Convolution กับภาพขาวด เพื่อเก็บค่าไว้ในเมทริกซ์ชุดใหม่ ที่เรียกว่าคอนโวลฟีเจอร์ (Convolved Feature) หรือ ฟีเจอร์แมพ (Feature Map)



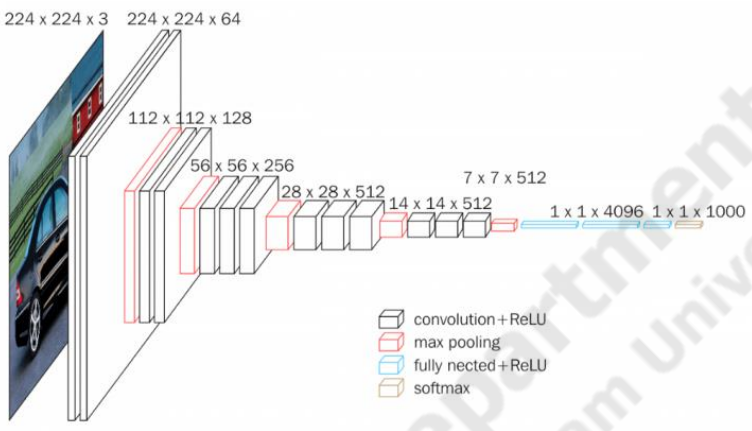
ภาพประกอบที่ 2.4 Convolution model

ที่มา : https://medium.com/olarik/โครงข่ายประสาทเทียมแบบคอนโวลูชัน-4e9e3a7a39bf

2. VGG-16

VGG16 [13] เป็นสถาปัตยกรรมหนึ่งของ Convolutional Neural Network (CNN) ที่เรียบง่ายและใช้กันอย่างแพร่หลายซึ่งใช้สำหรับ ImageNet ซึ่งเป็นโครงการฐานข้อมูลภาพขนาดใหญ่ที่ใช้ในการวิจัยซอฟต์แวร์การรู้จำวัตถุภาพ สถาปัตยกรรม VGG16 ได้รับการทดสอบ 5 อันดับสูงสุด 92.7% ใน ImageNet ซึ่งเป็นชุดข้อมูลมากกว่า 14 ล้านภาพที่อยู่ใน 1000 คลาส เป็นหนึ่งในโมเดลที่มีชื่อเสียงที่ส่งไปยัง ImageNet Large Scale Visual Recognition Challenge (ILSVRC) ในปี 2014 ได้ทำการปรับปรุงสถาปัตยกรรม AlexNet โดยแทนที่ตัวกรองขนาดเคอร์เนลขนาดใหญ่ (11 และ 5 ในเลเยอร์ convolutional ที่หนึ่งและสองตามลำดับ) ด้วยตัวกรองขนาดเคอร์เนลสาม x สามตัวทีละ

ตัว VGG16 ได้รับการฝึกฝนมาเป็นเวลาหลายสัปดาห์โดยใช้ NVIDIA Titan Black GPUs VGG16 ใช้ในเทคนิคการจำแนะรูปภาพการเรียนรู้เชิงลึกจำนวนมากและเป็นที่นิยมเนื่องจาก ใช้งานง่าย



ภาพประกอบที่ 2.5 VGG-16

ที่มา https://ichi.pro/th/vgg16-khux-xari-bthna-su-vgg16-267001881294357

2.1.2.1 การทำงานของ VGG-16

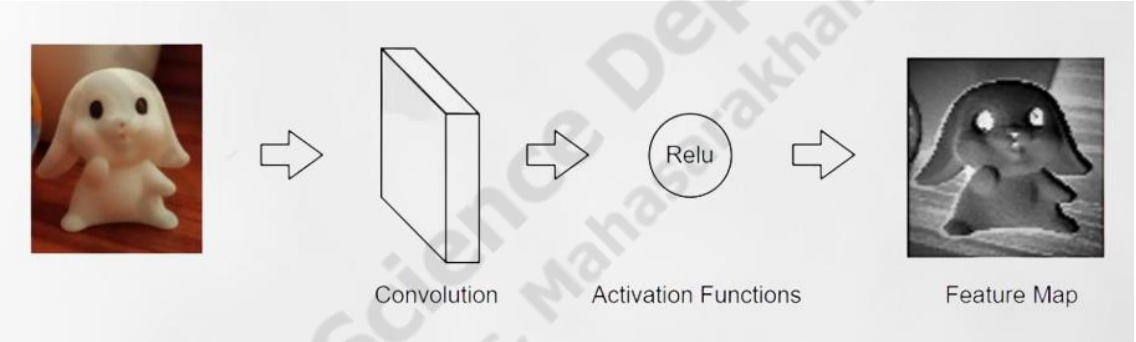
ตารางที่ 2.1 ตารางของ VGG-16

Layer		Feature Map	Size	Kernel Size	Stride	Activation
Input	Image	1	224 x 224 x 3	-	-	-
1	2 X Convolution	64	224 x 224 x 64	3x3	1	relu
	Max Pooling	64	112 x112 x 64	3x3	2	relu
3	2 X Convolution	128	112 x 112 x 128	3x3	1	relu
	Max Pooling	128	56 x 56 x 128	3x3	2	relu
5	2 X Convolution	256	56 x 56 x 256	3x3	1	relu
	Max Pooling	256	28 x 28 x 256	3x3	2	relu
7	3 X Convolution	512	28 x 28 x 512	3x3	1	relu
	Max Pooling	512	14 x 14 x 512	3x3	2	relu

ตารางที่ 2.1 ตารางของ VGG-16 (ต่อ)

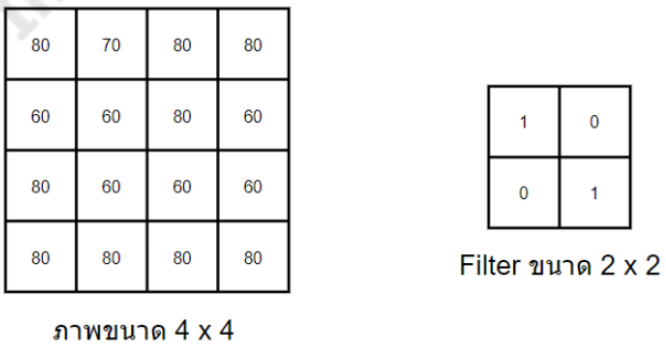
Layer		Feature Map	Size	Kernel Size	Stride	Activation
10	3 X Convolution	512	14 x 14 x 512	3x3	1	relu
	Max Pooling	512	7 x 7 x 512	3x3	2	relu
13	FC	-	25088	-	-	relu
14	FC	-	4096	-	-	relu
15	FC	-	4096	-	-	relu
output	FC	-	1000	-	-	Softmax

2.1.2.2 ค้นหาคุณลักษณ์เด่นของภาพออกมา (Convolution)



ภาพประกอบที่ 2.6 Convolution

การทำงานของในขั้นตอนค้นหาคุณลักษณ์เด่นของภาพออกมา (Convolution) จะทำการ Sliding Windows (Filter) เพื่อค้นหาองค์ประกอบของภาพเช่น สี หรือรูปร่าง

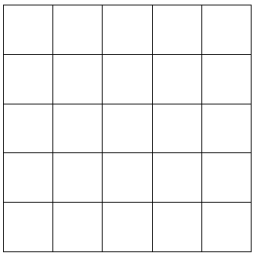


ภาพประกอบที่ 2.7 ขนาดภาพนำเข้าและขนาดของ Filter

กำหนดภาพนำเข้าขนาดเป็น 4 x 4 Filter เป็น 2 x 2 Stride เป็น 1 และ Padding เป็น 1

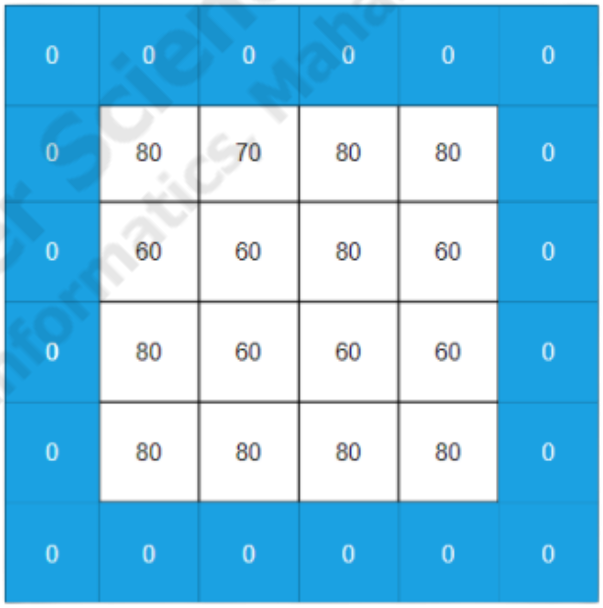
สามารถแทนค่าสมการได้ดังนี้

$$\text{output of size} = \frac{4-2+2(1)}{1} + 1 = 5 \quad (1)$$



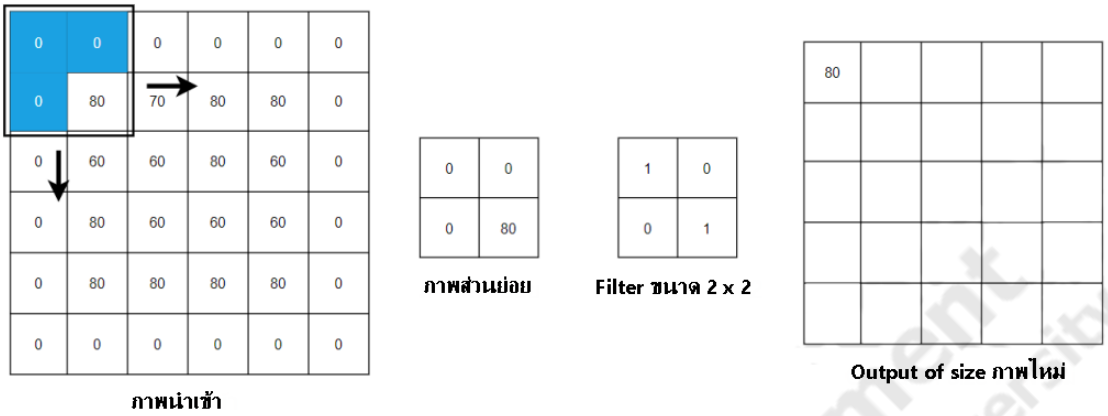
ภาพประกอบที่ 2.8 ภาพ Convolution

กระบวนการ Convolution อาจทำให้ภาพมีขนาดที่เล็กลง ถ้าทำ Convolution หลาย ๆ ชั้น ภาพสุดท้ายที่ออกมาก็จะเล็กลงมาก นอกจากนั้น Convolution ยังมีโอกาสทำให้ข้อมูลที่อยู่ตามขอบภาพไม่ถูกคำนวณเพราะ Filter มีโอกาสจับข้อมูลตามขอบภาพน้อยกว่าตรงกลางภาพแก้ปัญหาด้วยการ Padding ขยายภาพออกทุกด้านเท่าๆ กันโดยตั้ง Padding เป็น 1



ภาพประกอบที่ 2.9 Padding ขยายขอบภาพ

จากนั้นนำ Filter คูณกับภาพที่ทำ Padding เริ่มที่ตำแหน่งแรกของภาพ และนำค่าทั้งสองคูณกันตามตำแหน่งที่ตรงกัน ผลลัพธ์ที่ได้นำมาบวกกันและเก็บผลลัพธ์ไว้ที่ Output of size และทำให้ครบทั่วทั้งภาพใหม่ที่เล็กลง



ภาพประกอบที่ 2.10 การ Convolutional

แสดงตัวอย่างการคำนวณภาพส่วนย่อยคูณกับ Filter ขนาด 2 x 2 ในรอบที่ 1 ได้ดังนี้

ตำแหน่งที่ 1 0 x 1 = 0

ตำแหน่งที่ 2 0 x 0 = 0

ตำแหน่งที่ 3 0 x 0 = 0

ตำแหน่งที่ 4 80 x 1 =80

จากนั้น นำผลลัพธ์ของทุกตำแหน่งมาบวกกัน (0 + 0 + 0 + 80) = 80 และเก็บผลลัพธ์ไว้ที่ Output of size ภาพใหม่ และเลื่อนตำแหน่งไปให้ทั่วทั้งภาพผลลัพธ์ทั้งหมดจากคำนวณได้ดังนี้

80	70	80	80	0
60	140	150	140	80
80	120	120	140	60
80	160	140	140	60
0	80	80	80	80

ภาพประกอบที่ 2.11 Output of Size

2.1.2.3 ขั้นตอนการตรวจจับ (ReLU)

การตรวจจับ (Detector) ในขั้นตอนนี้จะทำหน้าที่รับข้อมูลที่ได้จากขั้นตอน Convolution มาแปลงให้อยู่ในรูปแบบที่ไม่เป็นเชิงเส้น (Nonlinear) โดยใช้ฟังก์ชันการกระตุ้น (Activation function) เช่น Rectified Linear Units (ReLU) โดยผลลัพธ์ที่ได้จากการทำ Convolution ในแต่ละตำแหน่งจะผ่านการแปลงค่าด้วยฟังก์ชัน ReLU ที่เป็นการแปลงแบบไม่เป็นเชิงเส้น เพื่อความง่ายในการคำนวณและประสิทธิภาพของผลลัพธ์ สามารถคำนวณด้วยสมการดังนี้

$$\text{ReLU}= \begin{cases} 0 \text{ if } x<0 \\ x \text{ if } x\geq 0 \end{cases} \quad (2)$$

โดยที่ x คือจุดพิกเซลของ Output of size ขัดตอนการทำงานโดยเงื่อนไขดังนี้

(1) ให้ x เป็น 0 ก็ต่อเมื่อ x น้อยกว่า 0

(2) ให้ x เป็น x ก็ต่อเมื่อ x มากกว่าหรือเท่ากับ 0

การทำงาน ReLU กับภาพตัวอย่างแสดงได้ดังนี้

-50	80
60	80

ภาพตัวอย่าง

U_{00}	U_{01}
U_{10}	U_{11}

ReLU

ภาพประกอบที่ 2.12 ตัวอย่างการทำงานของ ReLU

$$U_{01}= \max(0,-50)$$

$$U_{00}=0$$

$$U_{01}=\max(0,80)$$

$$U_{01}=80$$

$$U_{10}=\max(0,60)$$

$$U_{10}=60$$

$$U_{11}=\max(0,80)$$

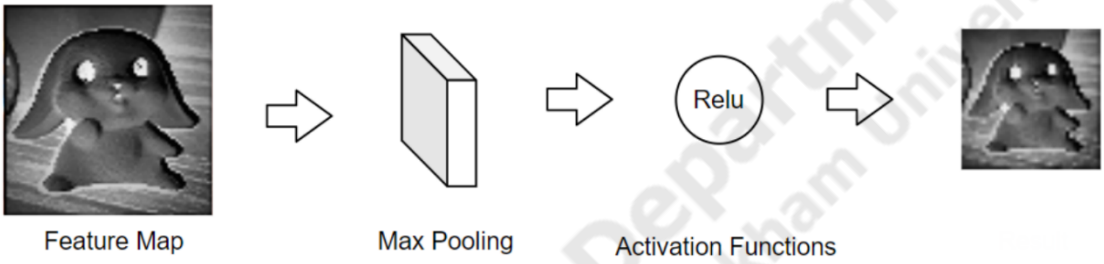
$$U_{11}=80$$

สรุปภาพตัวอย่างหลังจากทำ ReLU ได้ผลลัพธ์ดังนี้

0	80
60	80

ภาพประกอบที่ 2.13 ภาพตัวอย่างหลังจากทำ ReLU

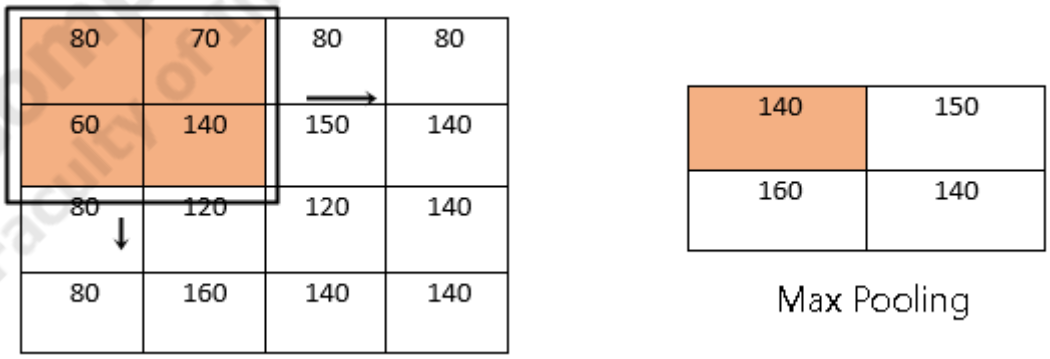
2.1.2.4 ลดขนาดภาพด้วยการเลือกจุดเด่นสุด (Max Pooling)



ภาพประกอบที่ 2.14 ภาพรวมของการ Max Pooling

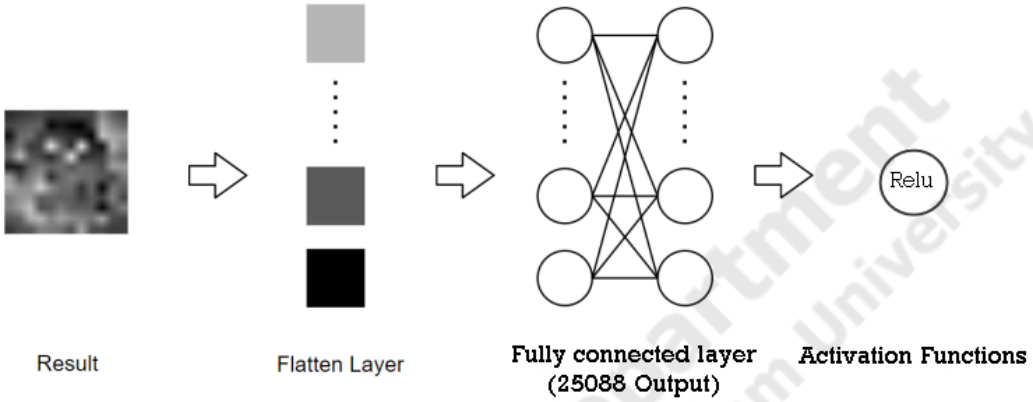
ลดขนาดของภาพด้วยการเลือกจุดเด่นสุด (Max Pooling) เพื่อลดขนาดข้อมูลภาพให้เล็กลงแต่รายละเอียดยังคงลักษณะที่เด่นไว้ และยังเพิ่มความเร็วในการคำนวณในขั้นตอนถัดไป ในการคำนวณหาค่าสูงสุด (Max Pooling) วิธีการทำงานคล้ายกับ Convolution โดยเพิ่มขั้นตอนการหาค่าที่มีกที่สุด และก่อนจะได้ผลลัพธ์ต้องนำภาพผ่านขั้นตอน ReLu อีกครั้ง และแสดงขั้นตอนการทำงานได้ดังนี้

Stride จะเลื่อน Filter ตามขนาดที่กำหนด เช่น ขนาด 2 x 2 ซึ่งจะทำให้มีการลดขนาดของภาพลงได้ครึ่งหนึ่งดังตัวอย่าง

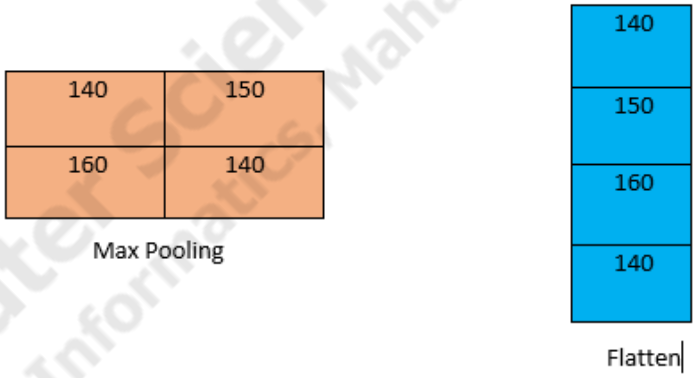


ภาพประกอบที่ 2.15 การหาค่าสูงสุด (Max Pooling)

2.1.2.5 แผ่ภาพให้เป็นแนวตั้ง (Flatten)



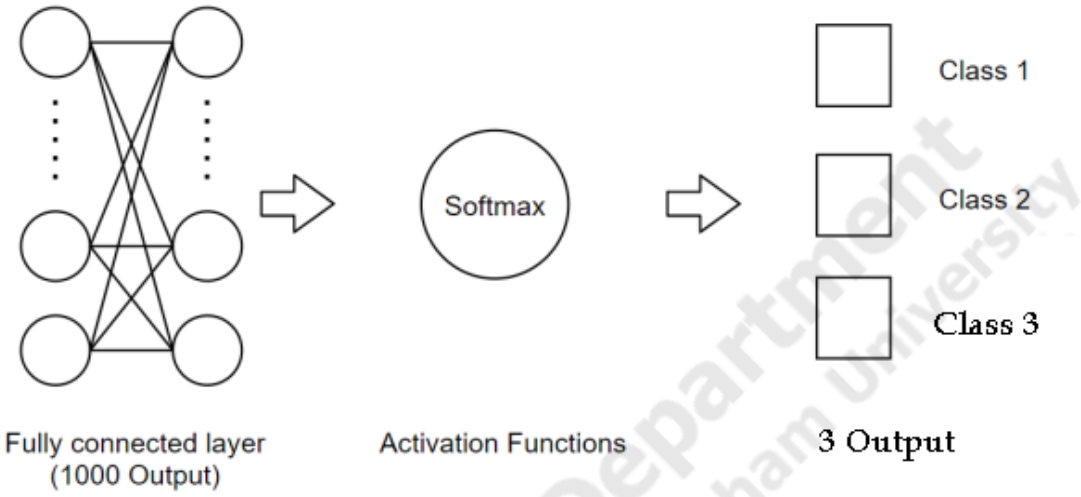
ภาพประกอบที่ 2.16 ภาพรวมของการ Flatten



ภาพประกอบที่ 2.17 ผลลัพธ์การเปลี่ยนโครงสร้างชุดข้อมูล

เมื่อทำการแผ่ผลลัพธ์ด้วยฟังก์ชัน Flatten ในขั้นตอนต่อไปจะเป็นการนำผลลัพธ์ทั้งหมดเข้าไปทำงานในขั้นตอน Full connection เป็นขั้นตอนของโครงข่ายประสาทเทียม (Artificial Neural Network) ที่เป็นการรับ Input แบบแนวตั้ง

2.1.2.6 ปรับ Input ให้เหลือเท่ากับ Output (Full Connection)



ภาพประกอบที่ 2.18 ภาพรวมของการ Full connection

เมื่อผ่านขั้นตอน Full Connection จนเหลือ 1000 ความเป็นไปได้ของ Output สุดท้ายจะต้องนำค่าที่ได้ทั้งหมดเข้าฟังก์ชัน Activation Functions แบบ Softmax ก่อนแล้วจะได้คำตอบ Output จริง ๆ ตัวอย่างการคำนวณ Softmax แสดงตัวอย่างด้วยชุดมูลที่กำหนดให้ต่อไปนี้

Full connection = [-1,0,3] แสดงวิธีการคำนวณด้วยตำแหน่งที่ 1 ดังนี้

$$s(-1) = \frac{e^{-1}}{e^{-1}+e^{0}+e^{3}} \quad (3)$$

$$s(-1) = \frac{0.367}{0.367+1+20.08} \quad (4)$$

$$s(-1) = \frac{0.367}{21.44} \text{ หรือ } S(-1)=0.017 \quad (5)$$

เมื่อคำนวณครบทุกค่าผลลัพธ์จะได้ตั้งแต่ 0 ถึง 1 ถ้าทั้งหมดรวมกันแล้วจะเท่ากับ 1

ตารางที่ 2.2 การคำนวณ Softmax

x	e^{-1}	ความน่าจะเป็น
-1	0.367	0.01714782554552
0	1	0.046612622577974
3	20.08	0.93623955187651
รวม		1

3. Region proposal network

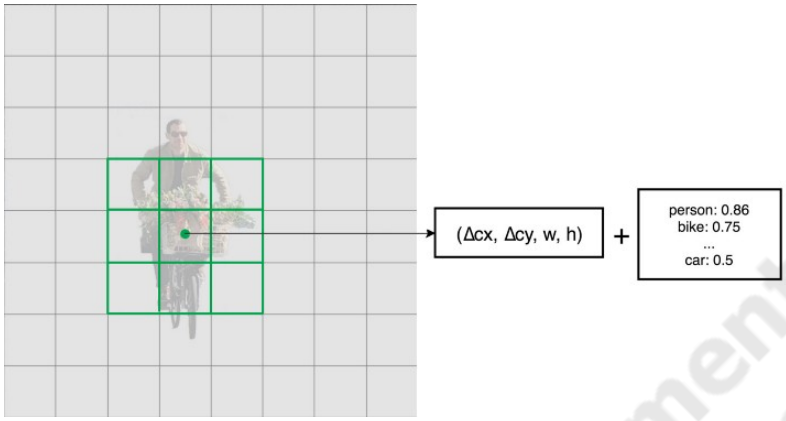
Single Shot MultiBox Detector [6] ออกแบบมาสำหรับการตรวจจับวัตถุแบบเรียลไทม์ R-CNN ที่เร็วขึ้นใช้เครือข่ายข้อเสนอระดับภูมิภาคเพื่อสร้างกล่องขอบเขตและใช้กล่องเหล่านั้นเพื่อจัดประเภทวัตถุ แม้ว่าจะถือว่าเป็นจุดเริ่มต้นของความแม่นยำ แต่กระบวนการทั้งหมดจะทำงานที่ 7 เฟรมต่อวินาที ต่ำกว่าความต้องการของการประมวลผลแบบเรียลไทม์ SSD เร่งกระบวนการโดยไม่ต้องใช้เครือข่ายข้อเสนอของภูมิภาค ในการกู้คืนความแม่นยำที่ลดลง SSD ใช้การปรับปรุงบางอย่างรวมถึงคุณสมบัติหลายมาตราส่วนและกล่องเริ่มต้น การปรับปรุงเหล่านี้ช่วยให้ SSD จับคู่ความแม่นยำของ R-CNN ได้เร็วขึ้นโดยใช้ภาพที่มีความละเอียดต่ำกว่าซึ่งจะทำให้ความเร็วสูงขึ้น จากการเปรียบเทียบดังต่อไปนี้ทำให้ได้ความเร็วในการประมวลผลแบบเรียลไทม์และยังสามารถเอาชนะความแม่นยำของ Faster R-CNN ได้อีกด้วย (วัดความแม่นยำเป็นแผนที่ความแม่นยำเฉลี่ยเฉลี่ย: ความแม่นยำของการคาดการณ์)

ภาพประกอบที่ 2.19 การตรวจจับวัตถุ SSD

ที่มา : https://ichi.pro/th/kar-trwc-cab-watthu-ssd-single-shot-multibox-detector-sahrab-kar-pramwl-phl-baeb-rei-yl-thim-171355751058950

2.1.3.1 **Convolutional Predictors สำหรับการตรวจจับวัตถุ**

SSD ไม่ใช้ RPN ที่ได้รับมอบหมาย แต่จะแก้ไขเป็นวิธีที่ง่ายมาก ระดับการใช้ฟิลเตอร์บิดขนาดเล็ก หลังจากแยกแผนที่คุณลักษณะแล้ว SSD จะใช้ตัวกรอง Convolution 3 × 3 สำหรับแต่ละเซลล์เพื่อทำการคาดคะเน (ตัวกรองเหล่านี้จะคำนวณผลลัพธ์เช่นเดียวกับตัวกรอง CNN ทั่วไป) ตัวกรองแต่ละตัวจะแสดงผล 25 ช่อง: คะแนน 21 คะแนนสำหรับแต่ละชั้นเรียนพร้อมด้วยกล่องขอบเขตหนึ่งกล่อง เช่นใน Conv4_3 เราใช้ตัวกรอง 3 × 3 จำนวน 4 ตัวเพื่อจับคู่ช่องอินพุต 512 ช่องกับช่องสัญญาณเอาต์พุต 25 ช่อง

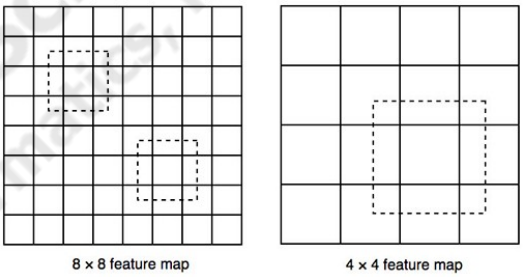


ภาพประกอบที่ 2.20 ตัวกรอง Convolution 3x3 เพื่อทำการคาดคะเนตำแหน่ง

ที่มา : https://ichi.pro/th/kar-trwc-cab-watthu-ssd-single-shot-multibox-detector-sahrab-kar-pramwl-phl-baeb-rei-yl-thim-171355751058950

2.1.3.2 **แผนที่คุณสมบัติหลายมาตราส่วนสำหรับการตรวจจับ**

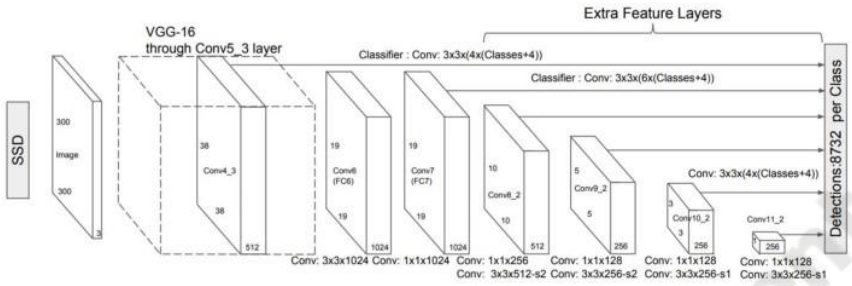
ในตอนแรกเราจะอธิบายว่า SSD ตรวจจับวัตถุจากเลเยอร์เดียวได้อย่างไร จริง ๆ แล้วมันใช้หลายเลเยอร์ (แผนที่คุณลักษณะหลายมาตราส่วน)เพื่อตรวจจับวัตถุ เนื่องจาก CNN ลดขนาดเชิงพื้นที่ลงเรื่อย ๆ ความละเอียดของแผนที่ฟีเจอร์ก็ลดลงเช่นกัน SSD ใช้เลเยอร์ความละเอียดต่ำเพื่อตรวจจับวัตถุขนาดใหญ่ ตัวอย่างเช่นแผนที่คุณลักษณะ 4 × 4 ใช้สำหรับวัตถุขนาดใหญ่



ภาพประกอบที่ 2.21 แผนที่คุณลักษณะความละเอียดต่ำกว่า (ขวา) ตรวจจับวัตถุขนาดใหญ่

ที่มา : https://ichi.pro/th/kar-trwc-cab-watthu-ssd-single-shot-multibox-detector-sahrab-kar-pramwl-phl-baeb-rei-yl-thim-171355751058950

SSD เพิ่มเลเยอร์ Convolution เสริมอีก 6 ชั้นหลังจาก VGG16 ห้าในนั้นจะถูกเพิ่มสำหรับการตรวจจับวัตถุ ในสามเลเยอร์เหล่านั้นเราทำการทำนาย 6 ครั้งแทนที่จะเป็น 4 โดยรวมแล้ว SSD ทำการทำนาย 8732 รายการโดยใช้ 6 เลเยอร์



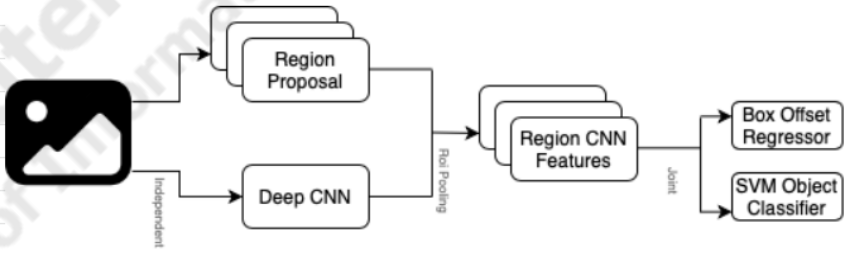
ภาพประกอบที่ 2.22 Single Shot MultiBox Detector

ที่มา : https://ichi.pro/th/kar-trwc-cab-watthu-ssd-single-shot-multibox-detector-sahrab-kar-pramwl-phl-baeb-rei-yl-thim-171355751058950

แผนที่คุณสมบัติหลายมาตราส่วนช่วยเพิ่มความแม่นยำอย่างมาก นี่คือความแม่นยำของเลเยอร์แผนที่คุณลักษณะต่าง ๆ ที่ใช้สำหรับตรวจจับวัตถุ

4. Faster R-CNN

Faster R-CNN [7] เป็นโครงข่ายที่แบ่งออกเป็น 2 สเตจ คือส่วนเสนอพื้นที่ (RPN) และส่วนรู้จำวัตถุ (Recognition) การฝึกตัวแบบให้ทำหน้าที่เป็นสองชุดเช่นนี้เราเรียกว่าเป็นการสร้าง Two-stage Object-Detection Network. ข้อเสียของมันก็คือความเร็วที่ได้นั้นน้อยกว่าระบบที่ทำการเลือกกรอบพื้นที่อย่างรวดเร็วและทำการรู้จำวัตถุทันที



ภาพประกอบที่ 2.23 สถาปัตยกรรมของ Fast R-CNN

ที่มา : https://ichi.pro/th/fuk-r-cnn-di-rew-khun-doy-chi-tensorflow-object-detection-api-phrxm-chud-khxmul-thi-kahnd-xeng-150484150544125

2.2 ระบบงานที่เกี่ยวข้อง

นายพิษณุรัตน์ วงสีเทา และ นายชลวัฒน์ เดโพธิ์ [9] วิทยานิพนธ์นี้ได้เสนอโปรแกรมการรู้จำและตรวจจับความเร็วรถบนท้องถนนเป็นการประเมินประสิทธิภาพ ส่วนที่ 1 การคำนวณความจากข้อมูลรถจากวิดีโอจำนวน 10 คัน สามารถวันความเร็วได้ และ สามารถนำไปพัฒนาต่อได้ ส่วนที่ 2 การระบุสีของรถยนต์จากภาพของรถยนต์ 100 ภาพ และภาพจากวิดีโอ 70 ภาพ ผลการระบุสีจากวิดีโอมีค่าความถูกต้อง 58% ค่าประสิทธิภาพโดยรวม 49% ถือว่ายังมีข้อผิดในการระบุสี และจากไฟล์วิดีโอผลการระบุสีจากวิดีโอมีค่าความถูกต้อง 35% ค่าประสิทธิภาพโดยรวม 46% ถือว่า การระบุจากไฟล์วิดีโอ มีประสิทธิภาพสูงกว่าการระบุสีจากภาพของรถยนต์ ส่วนที่ 3 การจำแนกประเภทของรถยนต์จากภาพของรถยนต์ 260 ภาพ และภาพจากวิดีโอ 70 ภาพ ผลการจำแนกด้วยภาพของรถยนต์มีความถูกต้อง 71% ค่า 48% พบว่ายังเกิดข้อผิดพลาดอยู่ แต่ยังถือว่ายอมรับได้ในการใช้งานอัลกอริทึม และการจำแนกจากภาพมี ประสิทธิภาพกว่ามีประสิทธิภาพกว่าการจำแนกจากภาพวิดีโอ ทั้งนี้จำนวนดาต้าและความคล้ายคลึงยังมีผลต่อการจำแนกได้

โยษิตา คำบุญมี และ สุขสวัสดี ณัฏฐวุฒิสิทธิ์ และ ปราลี มณีรัตน์ [10] วิทยานิพนธ์นี้ได้เสนอวิธีแสดงผลลัพธ์ธ์ตามวัตถุประสงค์ที่วางไว้ เพื่อนำเทคโนโลยีนี้มาใช้ทดทนแรงงานมนุษย์โดย การศึกษาการวิเคราะห์ค่าพารามิเตอร์สีด้วยเทคนิคการ เรียนรู้ของโครงข่ายนิวรอลเน็ตเวิร์ค ในงานวิจัยนี้ใช้มะเขือเทศพันธุ์โทมัสเป็นกลุ่มตัวอย่าง เกณฑ์การ จำแนกสีแบ่งออกเป็น 3 กลุ่ม คือ Green, Red และ Mature Red ซึ่งผลการทดลองมีความถูกต้องในการ จำแนกสีเท่ากับ 94.07% แต่อย่างไรก็ดีหากตัวอย่าง มะเขือเทศที่อยู่ในระหว่างการเปลี่ยนสีทำให้เกิดเฉดสีที่ ใกล้เคียงกันก็สามารถทำให้เกิดความผิดพลาดขึ้นได้ ดังนั้นการนำองค์ความรู้จากงานวิจัยนี้ไปใช้พัฒนา สำหรับการสร้างเครื่องมือคัดแยกสีของมะเขือเทศใน อุตสาหกรรมอาหารในอนาคต จึงจำเป็นต้องเพิ่ม จำนวนตัวอย่างในการเรียนรู้สีของมะเขือเทศในนิวรอล เน็ตเวิร์คต่อไป

ณัฐวุฒิ ชัยพิมล [11] วิทยานิพนธ์นี้ได้เสนอวิธีที่จะช่วยเพิ่มประสิทธิภาพของโปรแกรมเล่นเพลงอัตโนมัติจากโน้ต โดยใช้คอมพิวเตอร์มาช่วยในการจำแนกด้วยวิธี Convolutional Neural Network (CNN) และจัดเก็บข้อมูลในระบบดิจิทัล ซึ่งจะทำให้อ่านโน้ตเพลงนั้นง่ายยิ่งขึ้น จากการทดลองวัดประสิทธิภาพในการจำแนกโน้ตเพลงโดยใช้ข้อมูลฝึกฝนทั้งหมด 20 ประเภท คือ โน้ตตัวกลม โน้ตตัวขาว น้ตตัวดำ โน้ตตัวเขบ็ด 1 ชั้น โน้ตตัวเขบ็ต 2 ชั้น โน้ตตัวเขบ็ต 3 ชั้น ตัวกลมประจุด โน้ตตัวกลมประจุด โน้ตตัวขาวประจุด โน้ตตัวดำประจุด โน้ตตัวเขบ็ต 1 ชั้นประจุด โน้ตตัวขเบ็ต 2 ชั้นประจุด โน้ตตัวขเบ็ต 3 ชั้น ประจุด เทมโป จังหวะ ตัวหยุดตัวกลม ตัวหยุดตัวขาว ตัวหยุดตัวดำ ตัวหยุดตัวเขบ็ต 1 ชั้น ตัวหยุดขบ็ต 2 ชั้น ตัว หยุดตัวเขบ็ต 3 ชั้น ในการฝึกฝนข้อมูลและทดสอบโมเดลเพื่อใช้ในการแยกประเภทด้วยวิธี CNN ทั้งหมด 315,478 ภาพ ซึ่งผลจากการทดลองผลการจำแนกสัญลักษณ์ชนิดต่างสามารถ

จำแนกได้ถูกต้องประมาณ 86 % และจากการทดลองวัดประสิทธิภาพในการจำแนก Tempo โดยการใช้ Tesseract-OCR ดึงข้อความจากรูปภาพ นำมาเทียบกับผลเฉลยโดยใช้ข้อมูลภาพทั้งหมด 4,496 ภาพโดยผลลัพท์ที่ได้ภาพที่ถูกต้อง 3413 ภาพ ส่วนที่ผิด 1082 ภาพ จากการวัดประสิทธิภาพการจำแนก Tempo ด้วย Tesseract-OCR ได้ค่าความถูกต้อง 75.91 %