

โครงการเลขที่ วศ.คพ. P030/2563

เรื่อง

ระบบตรวจจับไฟไหม้

โดย

นาย รัชพงศ์ ทอหุล รหัส 600610769

โครงการนี้

เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่

ปีการศึกษา 2563

PROJECT No. CPE P030/2563

Automatic Fire Detection System

Rakpong Thorhun 600610769

A Project Submitted in Partial Fulfillment of Requirements
for the Degree of Bachelor of Engineering
Department of Computer Engineering
Faculty of Engineering
Chiang Mai University
2020

หัวข้อโครงการ	: ระบบตรวจจับไฟไหม้
	: Automatic Fire Detection System
โดย	: นาย รัชพงศ์ ทอหุล รหัส 600610769
ภาควิชา	: วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา	: อ.ดร.เกษมสิทธิ์ ตียพันธ์
ปริญญา	: วิศวกรรมศาสตรบัณฑิต
สาขา	: วิศวกรรมคอมพิวเตอร์
ปีการศึกษา	: 2563

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่ได้อนุมัติให้โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต (สาขาวิศวกรรมคอมพิวเตอร์)

..... หัวหน้าภาควิชาวิศวกรรมคอมพิวเตอร์
(รศ.ดร.ศักดิ์กษิต ระมิงค์วงศ์)

คณะกรรมการสอบโครงการ

..... ประธานกรรมการ
(อ.ดร.เกษมสิทธิ์ ตียพันธ์)

..... กรรมการ
(รศ.ดร.ศันสนีย์ เอื้อพันธ์วิริยะกุล)

..... กรรมการ
(ผศ.ดร.กานต์ ปทานุคม)

หัวข้อโครงงาน	: ระบบตรวจจับไฟไหม้	
โดย	: นาย รัชพงศ์ ทอหุล	รหัส 600610769
ภาควิชา	: วิศวกรรมคอมพิวเตอร์	
อาจารย์ที่ปรึกษา	: อ.ดร.เกษมสิทธิ์ ตียพันธ์	
ปริญญา	: วิศวกรรมศาสตรบัณฑิต	
สาขา	: วิศวกรรมคอมพิวเตอร์	
ปีการศึกษา	: 2563	

บทคัดย่อ

ในปัจจุบันปัญหาไฟป่าถือเป็นปัญหาที่มีความสำคัญเป็นอย่างยิ่งเพราะการเกิดไฟป่านั้นหากเกิดขึ้นโดยไม่สามารถแก้ปัญหาได้เร็ว ไฟป่าจะก่อให้เกิดความเสียหายเป็นอย่างมาก การที่เราสามารถตรวจจับไฟป่าที่กำลังเกิดขึ้นได้อย่างรวดเร็วจึงเป็นสิ่งสำคัญในการที่จะสามารถลดความเสียหายจากการเกิดไฟป่าได้

โดยระบบตรวจจับไฟไหม้นั้นจะทำการนำภาพมาตรวจจับไฟและควันไฟโดยใช้โครงข่ายประสาท (neural network) ประเภท CNN (Convolutional Neural Network) เพื่อเป็นตัวจำแนก (classifier) ว่าภาพที่ได้ในขณะนั้นเกิดไฟไหม้ขึ้นหรือไม่ แล้วส่งข้อมูลแจ้งออกมาให้ทราบ โดยภาพที่ระบบตรวจจับไฟไหม้จะรับเข้ามาเป็นวิดีโอที่จะนำไปแยกเป็นเฟรมภาพโดยภาพนั้นจะเป็นภาพลักษณะมุมสูงเพื่อจำลองว่าเป็นการตรวจจับจากกล้องบนหอคอยตรวจจับไฟป่านั้นเอง

หัวข้อโครงการ	: Automatic Fire Detection System
โดย	: Rakpong Thorhun 600610769
ภาควิชา	: Computer Engineering
อาจารย์ที่ปรึกษา	: Kasemsit Teeyapan, Ph.D. Lecturer
ปริญญา	: Bachelor of Engineering
สาขา	: Computer Engineering
ปีการศึกษา	: 2020

ABSTRACT

At present, the problem of forest fire is a very important problem because the occurrence of wildfire if occurs without solving the problem quickly. Wildfires are very damaging. Being able to quickly detect an impending wildfire is essential in order to reduce the damage caused by wildfires.

The fire detection system uses images to detect fire and smoke using a CNN (Convolutional Neural Network) type neural network to classify (classifier) that the image was at that time. Is the fire burning up? And send out the information to inform The image received by the fire detection system is a video that is split into frames, the image is of a high-angle image to simulate detection from a camera on a forest fire detection tower.

กิตติกรรมประกาศ

โครงการนี้จะไม่สำเร็จลงได้ ถ้าไม่ได้รับความกรุณาจาก อ.ดร.เกษมสิทธิ์ ตียพันธ์ อาจารย์ที่ปรึกษา ที่ได้สละเวลาให้ความช่วยเหลือทั้งให้คำแนะนำ ให้ความรู้และแนวคิดต่างๆรวมถึง รศ.ดร.คันสนีย์ เอื้อพันธ์วิริยะกุล และ ผศ.ดร.กานต์ ปทานุคม ที่ให้คำปรึกษาจนทำให้โครงการเล่มนี้เสร็จสมบูรณ์ไปได้

ขอบคุณ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่ ที่เอื้อเพื่อสถานที่ในการทำโครงการ

ขอขอบคุณเพื่อนๆ ที่ให้กำลังใจรวมถึงคำแนะนำที่ดีตลอดการทำโครงการที่ผ่านมา

นอกจากนี้ผู้จัดทำขอขอบพระคุณขอขอบพระคุณบิดา มารดาที่ได้ให้ชีวิต เลี้ยงดูสั่งสอน และส่งเสียให้กระผมได้ศึกษาเล่าเรียนจนจบหลักสูตรปริญญาตรี หลักสูตรวิศวกรรมศาสตรบัณฑิต ซึ่งท่านได้ให้กำลังใจ ในวันที่ท้อแท้ตลอดมา ซึ่งท่านยังเป็นแรงผลักดันให้กระผมสร้างสรรค์และมุ่งมั่นจนทำให้โครงการนี้สำเร็จ รวมทั้งขอขอบพระคุณอีกหลายๆท่านที่ไม่ได้เอ่ยนามมา ณ ที่นี้ ที่ได้ให้ความช่วยเหลือตลอดมา หากหนังสือโครงการเล่มนี้มีข้อผิดพลาดประการใด กระผมขอน้อมรับด้วยความยินดี

นาย รัชพงศ์ ทอหุล

24 มีนาคม 2564

สารบัญ

หน้า

บทคัดย่อ	ง
ABSTRACT	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ.....	ช
สารบัญภาพ	ณ
สารบัญตาราง.....	ญ
บทนำ.....	1
1.1 ที่มาของโครงการ.....	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตของโครงการ.....	1
1.3.1 ขอบเขตด้านฮาร์ดแวร์.....	1
1.3.2 ขอบเขตด้านซอฟต์แวร์.....	2
1.4 ประโยชน์ที่ได้รับ	2
1.5 เทคโนโลยีและเครื่องมือที่ใช้	2
1.5.1 เทคโนโลยีด้านฮาร์ดแวร์.....	2
1.5.2 เทคโนโลยีด้านซอฟต์แวร์	2
1.6 แผนการดำเนินงาน	2
1.7 บทบาทและความรับผิดชอบ	3
1.8 ผลกระทบด้านสังคม สุขภาพ ความปลอดภัย กฎหมาย วัฒนธรรม.....	3
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....	4
2.1 การตรวจจับไฟป่าโดยใช้หอดูดาว (การตรวจหาไฟทางกึ่งอากาศ) [1]	4
2.2 neural network [7].....	8
2.3 Convolutional Neural Network [2]	8
2.4 Transfer Learning [8]	13
2.5 แกรดแคม (grad cam) [9].....	15
2.6 ความรู้ตามหลักสูตรซึ่งถูกนำมาใช้หรือบูรณาการในโครงการ	16
2.7 ความรู้นอกหลักสูตรซึ่งถูกนำมาใช้หรือบูรณาการในโครงการ	16
บทที่ 3 โครงสร้างและขั้นตอนการทำงาน.....	17
3.1 การเตรียมชุดข้อมูลเพื่อฝึกสอนตัวแบบ(model)	17
3.2 การแปลงตัวแบบจำแนกลักษณะมาตรฐานที่มีอยู่แล้วให้เหมาะกับการจำแนกไฟไหม้.	18

3.3 การฝึกสอนตัวแบบ	18
3.4 การสร้างเว็บแอปพลิเคชัน	19
บทที่ 4 การทดลองและผลลัพธ์	21
4.1 การทดสอบกับชุดข้อมูลรูปภาพกับตัวแบบในชุดข้อมูลชุดแรก	22
4.2 การทดสอบกับชุดข้อมูลรูปภาพกับตัวแบบในชุดข้อมูลชุดที่สอง	22
4.3 การทดสอบกับชุดข้อมูลรูปภาพที่ได้มาจากวิดีโออินพุตในเว็บแอปพลิเคชัน	23
บทที่ 5 บทสรุปและข้อเสนอแนะ	25
5.1 สรุปผล.....	25
5.2 ปัญหาที่พบและแนวทางแก้ไข.....	25
5.3 ข้อเสนอแนะและแนวทางการพัฒนาต่อ	25
เอกสารอ้างอิง	26
ภาคผนวก.....	27
ภาคผนวก ก การใช้งานเว็บแอปพลิเคชัน.....	28
ประวัติผู้เขียน.....	32

สารบัญภาพ

หน้า

รูปที่ 2.1.1 การตรวจหาไฟโดยหอคูไฟและเครื่องเล็งทิศทางไฟ.....	5
รูปที่ 2.1.2 ตัวอย่างการหาตำแหน่งที่เกิดไฟป่าโดยใช้เครื่องเล็งทิศทางไฟ.....	6
รูปที่ 2.3.1 Neural Network.....	8
รูปที่ 2.3.1 แสดงการมองของมนุษย์.....	9
รูปที่ 2.3.2 ตัวกรอง 3x3 สำหรับหาเส้นตรงทแยงสีขาว.....	9
รูปที่ 2.3.3 ผังคุณลักษณะ (feature map).....	10
รูปที่ 2.3.4 ผังคุณลักษณะ (feature map) และ Stride.....	10
รูปที่ 2.3.5 ผังคุณลักษณะ (feature map) และ Stride.....	11
รูปที่ 2.3.6 ผังคุณลักษณะ (feature map) และ padding.....	11
รูปที่ 2.3.7 การสเกลภาพ.....	12
รูปที่ 2.3.8 max pooling.....	13
รูปที่ 2.4.1 การทรานเฟอร์เลินนิ่ง.....	14
รูปที่ 2.4.2 แสดงรายละเอียดของตัวแบบที่ใช้ในระบบ.....	15
รูปที่ 2.5.1 แสดงภาพ grad cam ที่แปลงจาก heatmap.....	15
รูปที่ 3.1.1 ตัวอย่างภาพที่ไม่มีควันไฟ.....	17
รูปที่ 3.1.2 ตัวอย่างภาพที่มีควันไฟ.....	17
รูปที่ 3.3.1 รูปแสดงค่าความแม่นยำจากการฝึกสอนตัวแบบ.....	18
รูปที่ 3.4.1 ภาพแสดงหน้าเว็บแอปพลิเคชัน.....	19

รูปที่ 3.4.2 ภาพแสดงหน้าเว็บแอปพลิเคชัน(ต่อ).....	19
รูปที่ 3.4.3 ภาพแสดงเส้นทางการไหลของข้อมูล.....	20
รูปที่ 4.0.1 แสดงตัวอย่างชุดข้อมูลทดสอบที่มีควันไฟ.....	21
รูปที่ 4.0.2 แสดงตัวอย่างชุดข้อมูลทดสอบที่ไม่มีควันไฟ.....	21
รูปที่ 4.2.1 แสดงบางส่วนของรูปภาพที่ไม่มีควันไฟที่เพิ่มเข้ามา.....	22
รูปที่ 4.2.2 แสดงบางส่วนของรูปภาพที่มีควันไฟที่เพิ่มเข้ามา.....	23

สารบัญตาราง

หน้า

ตารางที่ 4.2.1 แสดงรายละเอียดการทำนาย.....	23
ตารางที่ 4.3.1 แสดงรายละเอียดการทำนาย.....	24

บทนำ

1.1 ที่มาของโครงการ

ในปัจจุบันปัญหาไฟป่าถือเป็นปัญหาที่มีความสำคัญเป็นอย่างยิ่งเพราะการเกิดไฟป่านั้นหากเกิดขึ้นโดยไม่สามารถแก้ปัญหาได้เร็ว ไฟป่าจะก่อให้เกิดความเสียหายเป็นอย่างมากทั้งในส่วนของธรรมชาติที่จะถูกทำลายไป ระบบนิเวศในป่าและสิ่งมีชีวิตในป่าที่ตายจากการถูกไฟคลอกและสำคัญกว่านั้นคือควันไฟรวมไปถึงกลุ่มชาวบ้านที่อาศัยอยู่ใกล้ ๆ อีกด้วยอีกทั้งการฟื้นฟูสิ่งที่ถูกทำลายไปจากไฟป่านั้นทำได้ช้าเนื่องจากป่าไม้ต้องใช้เวลาในการเติบโต ดังนั้นการที่เราสามารถตรวจจับไฟป่าที่กำลังเกิดขึ้นได้อย่างรวดเร็วจึงเป็นสิ่งสำคัญในการที่จะสามารถลดความเสียหายจากการเกิดไฟป่าได้เป็นอย่างมาก โดยการตรวจจับไฟป่าในประเทศไทยในปัจจุบันมีหลายประเภทอาทิเช่น การตรวจจับโดยใช้พลเดินเท้าหรือยานยนต์ออกค้นหาตามจุดเสี่ยง การติดตั้งหอดูไฟในการเฝ้าระวังไฟป่า การตรวจหาไฟโดยใช้ดาวเทียม สิ่งที่เราน่าสนใจคือการตรวจจับไฟและควันไฟโดยใช้หอดูไฟซึ่งใช้เจ้าหน้าที่ในการตรวจสอบโดยเจ้าหน้าที่จะต้องอยู่ประจำการบนหอและตรวจสอบอยู่ตลอดเวลาซึ่งมีข้อเสียคือในการประจำการเป็นเวลานานเจ้าหน้าที่อาจจะเบื่อหน่าย หรือผลพลได้้อันจะก่อให้เกิดโอกาสที่จะตรวจจับไฟไหม้ได้ช้าเราจึงคิดว่าสามารถใช้เทคโนโลยีในปัจจุบันมาช่วยเพิ่มประสิทธิภาพของการตรวจจับไฟไหม้ได้โดยติดตั้งกล้องไว้บนหอดูไฟแล้วนำภาพที่ได้จากกล้องมาตรวจจับไฟโดยใช้โครงข่ายประสาท (neural network) ประเภท CNN (Convolutional Neural Network) เพื่อเป็นตัวจำแนก (classifier) ว่าภาพที่ได้จากหอดูไฟในขณะนั้นเกิดไฟไหม้ขึ้นหรือไม่ เพื่อสามารถเพิ่มประสิทธิภาพในการตรวจจับไฟและลดความผิดพลาดจากเจ้าหน้าที่ได้

1.2 วัตถุประสงค์ของโครงการ

1. พัฒนาเว็บแอปพลิเคชันสำหรับการทำนายภาพที่ได้จากวิดีโอในมุมมองสูงว่าเกิดไฟไหม้หรือควันไฟขึ้นหรือไม่ได้

1.3 ขอบเขตของโครงการ

1.3.1 ขอบเขตด้านฮาร์ดแวร์

1. อุปกรณ์สามารถเข้าใช้งานเว็บแอปพลิเคชัน
2. อุปกรณ์สามารถเข้าถึงอินเทอร์เน็ต

1.3.2 ขอบเขตด้านซอฟต์แวร์

1. ระบบใช้สำหรับการตรวจจับไฟไหม้และควันไฟจากรูปในลักษณะมุมสูงจากหอดูไฟเท่านั้น
2. รูปที่สามารถตรวจจับได้เป็นรูปสีเท่านั้น

1.4 ประโยชน์ที่ได้รับ

1. ได้เว็บแอปพลิเคชันที่สามารถจำแนกรูปภาพที่มีไฟไหม้และควันไฟอยู่ในรูปได้

1.5 เทคโนโลยีและเครื่องมือที่ใช้

1.5.1 เทคโนโลยีด้านฮาร์ดแวร์

1. อุปกรณ์ที่สามารถเข้าถึงอินเทอร์เน็ต และใช้งานเว็บแอปพลิเคชันได้

1.5.2 เทคโนโลยีด้านซอฟต์แวร์

1. React: ใช้สำหรับการพัฒนาเว็บแอปพลิเคชันส่วนหน้าใช้งาน
2. Python: ภาษาที่ใช้ในการพัฒนาระบบ
3. Tensorflow[6]: ไบรารี (library) สำหรับการสร้าง computation graph
4. Keras[4]: ไบรารี (library) สำหรับการสร้าง Convolutional Neural Network
5. Flask: ใช้สำหรับการพัฒนาเว็บแอปพลิเคชันส่วนวิเคราะห์ข้อมูล

1.6 แผนการดำเนินงาน

ขั้นตอนการดำเนินงาน	2563						2564		
	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	ม.ค.
ศึกษาทฤษฎีที่เกี่ยวข้อง									
รวบรวมข้อมูลที่ใช้ในการฝึกสอนตัวแบบ									
ศึกษาโมดูล CNN									
ออกแบบตัวแบบ									
ออกแบบ UX และ UI									
พัฒนาเว็บแอปพลิเคชัน									
พัฒนาตัวแบบ									
ทดสอบระบบ									
จัดทำรายงาน									

1.7 บทบาทและความรับผิดชอบ

โครงการนี้เป็นโครงการที่ต้องการเวลาในการศึกษาและลองผิดลองถูก จึงใช้วิธีดำเนินงานแบบ Agile ผสม Waterfall โดยทำการกำหนดแผนงานโดยคร่าว ๆ และนัดหมายกับอาจารย์ที่ปรึกษาทุกสัปดาห์เพื่อส่งมอบความคืบหน้าและสรุปปัญหาที่พบเจอ เพื่อหาแนวทางแก้ไขและกำหนดงานในสัปดาห์ถัดไป

1.8 ผลกระทบด้านสังคม สุขภาพ ความปลอดภัย กฎหมาย วัฒนธรรม

โครงการนี้จะช่วยให้การตรวจจับไฟฟ้าในปัจจุบันมีประสิทธิภาพเพิ่มมากขึ้น โดยเพิ่มความแม่นยำในการตรวจจับไฟฟ้าจากหอดูไฟและลดโอกาสที่จะตรวจจับไฟฟ้าพลาดหรือเข้าไปของเจ้าหน้าที่ประจำหอดูไฟอันเนื่องมาจากการทำงานติดต่อกันเป็นเวลานานทำให้เหนื่อยหรือเหนื่อยล้าได้เมื่อระบบตรวจจับไฟฟ้าดีขึ้น โอกาสที่ไฟฟ้าที่เกิดขึ้นจะลุกลามร้ายแรงก็น้อยลงส่งผลให้คุณภาพชีวิตของประชากร และสัตว์ป่าปลอดภัยมากยิ่งขึ้น

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

การทำโครงงาน เริ่มต้นด้วยการศึกษาค้นคว้า ทฤษฎีที่เกี่ยวข้อง หรือ งานวิจัย/โครงงาน ที่เคยมีผู้นำเสนอไว้แล้ว ซึ่งเนื้อหาในบทนี้จะเกี่ยวกับการอธิบายถึงสิ่งที่เกี่ยวข้องกับโครงงาน เพื่อให้ผู้อ่านเข้าใจเนื้อหาในบทถัดๆไปได้ง่ายขึ้น เนื้อหาในบทนี้จะแบ่งออกเป็นสามส่วนหลักๆคือส่วนที่เป็น การการตรวจจับไฟป่าโดยใช้หอดูไฟ ส่วน neural network และส่วน Convolutional Neural Network ดังนี้

2.1 การตรวจจับไฟป่าโดยใช้หอดูไฟ (การตรวจหาไฟทางกึ่งอากาศ) [1]

เป็นการตรวจหาไฟโดยจุดตรวจการณ์จะอยู่กับที่ ซึ่งจุดตรวจการณ์อาจเป็นหอดูไฟ (ภาพที่ 5.4) ซึ่งสร้างขึ้นโดยตรง หรือดัดแปลงคาบตันไม้เพื่อใช้เป็นหอดูไฟ หรือใช้จุดสังเกตการณ์ที่มีอยู่ตามธรรมชาติ เช่น ยอดเขา และริมหน้าผาสูง เป็นต้น วิธีนี้เหมาะสำหรับการตรวจหาไฟในพื้นที่ที่เป็นที่ราบกว้างใหญ่ โดยปกติแล้วรัศมีระยะตรวจการณ์สูงสุดของหอดูไฟจะอยู่ ระหว่าง 30-40 กิโลเมตร โดยมีอุปกรณ์ช่วย คือกล้องส่องทางไกล (Heikkila et al, 1993) ทั้งนี้โดยมีปัจจัยต่าง ๆ ที่มีอิทธิพลต่อรัศมีการมองเห็น ได้แก่ ช่วงเวลาที่แตกต่างกันของวัน หมอกควัน และทิศทางของแสงอาทิตย์ เป็นต้น

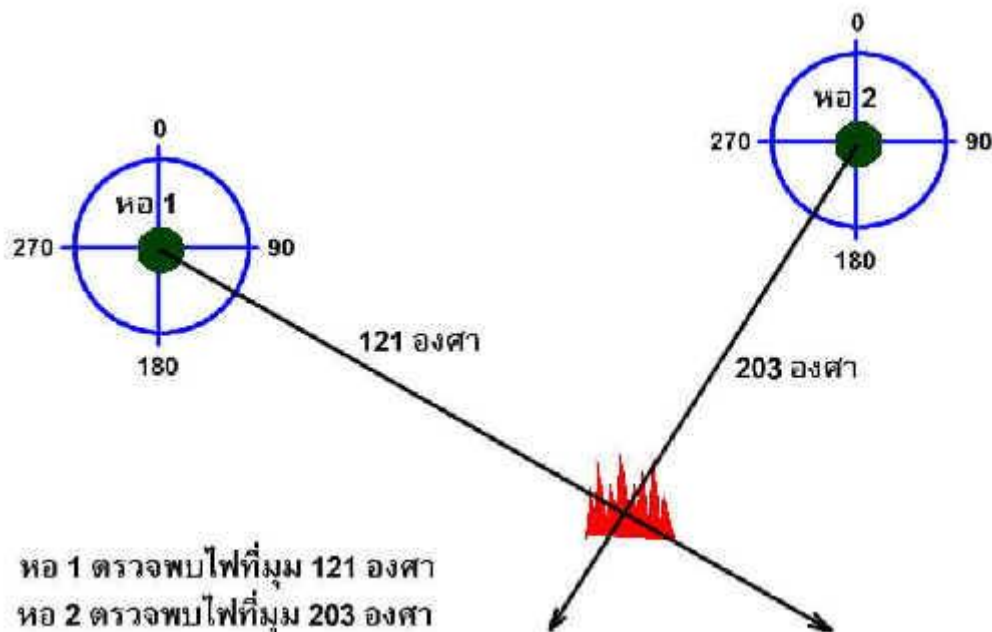
เนื่องจากการสร้างหอดูไฟเป็นการลงทุนที่ใช้ค่าใช้จ่ายสูง ดังนั้นจึงได้มีความพยายามที่จะดัดแปลงเพื่อลดต้นทุน เช่น ในประเทศสหรัฐอเมริกา ได้ทดลองสร้างหอดูไฟเคลื่อนที่โดยใช้รถแทรกเตอร์มาดัดแปลงเป็นหอดูไฟ และต่อมาได้ทดลองใช้หอดูไฟอัตโนมัติ โดยติดตั้งกล้องโทรทัศน์ไว้บนหอดูไฟแทนการใช้เจ้าหน้าที่ประจำหอดูไฟ แต่ปรากฏว่าไม่ประสบความสำเร็จเท่าที่ควร เนื่องจากกล้องโทรทัศน์ไม่สามารถแยกแยะควันไฟได้ดีเท่ากับสายตาของมนุษย์ และค่าใช้จ่ายก็ยังคงสูงอยู่เช่นเดิม สำหรับประเทศอิตาลีได้มีการพัฒนาระบบตรวจหาไฟอัตโนมัติติดตั้งบนหอดูไฟมาเป็นเวลานานแล้ว โดยในปัจจุบันมีระบบที่มีประสิทธิภาพสูง คือใช้เครื่องตรวจจับคลื่นรังสีอินฟราเรด (Infrared Sensor) ประกอบกับกล้องวิดีโอ ทำให้สามารถตรวจพบไฟป่าได้อย่างมีประสิทธิภาพมากกว่าใช้สายตามนุษย์และในขณะเดียวกันก็สามารถส่งภาพบริเวณที่เกิดไฟไหม้ในขณะนั้นมายังห้องควบคุมได้ในเวลาเดียวกัน อย่างไรก็ตามระบบดังกล่าวยังมีราคาแพงมาก โดยประเทศไทยเคยนำระบบดังกล่าวมาทดลองใช้ที่สถานีควบคุมไฟป่าห้วยฮ่องไคร้ จังหวัดเชียงใหม่ ในปี พ.ศ. 2537



รูปที่ 2.1.1 การตรวจหาไฟโดยหอดูไฟและเครื่องเล็งทิศทางไฟ

1. ข้อควรคำนึงในการสร้างหอดูไฟ

1.1 ในพื้นที่รับผิดชอบ จะต้องสร้างหอดูไฟหลายๆ หอในลักษณะเป็นเครือข่ายที่สามารถครอบคลุมการตรวจการณ์ได้ทั่วทั้งพื้นที่ โดยมีพื้นที่ตรวจการณ์ของแต่ละหอเหลื่อมกัน เพื่อว่าเมื่อเกิดไฟป่าขึ้น ณ บริเวณใด ๆ ก็ตาม จะต้องมียอดหอดูไฟอย่างน้อยที่สุด 2 หอ ตรวจพบไฟนั้นได้ในเวลาเดียวกัน ซึ่งจะทำให้สามารถใช้เครื่องเล็งทิศทางไฟ (Fire finder) วัดหาจุดที่เกิดไฟป่าได้อย่างแม่นยำโดยใช้หลักการหาจุดตัด (Interception) (ภาพที่ 2.2)



รูปที่ 2.1.2 ตัวอย่างการหาตำแหน่งที่เกิดไฟป่าโดยใช้เครื่องเล็งทิศทางไฟ

1.2 จุดที่สร้างหอดูไฟต้องเป็นที่สูงของพื้นที่นั้น ทำให้ตรวจการณ์พื้นที่โดยรอบได้เป็นบริเวณกว้างที่สุดเท่าที่จะเป็นไปได้ ความสูงของหอดูไฟโดยทั่วไปจะอยู่ระหว่าง 5-25 เมตร ขึ้นอยู่กับภูมิประเทศของจุดที่ตั้งหอดูไฟ และความสูงของต้นไม้โดยรอบ โดยหอดูไฟจะต้องสูงพ้นยอดต้นไม้หรือพุ่มไม้ที่ขวางการเห็นใด ๆ ในพื้นที่นั้น

1.3 หอดูไฟจะสร้างด้วยเหล็กหรือไม้ก็ได้ แต่จะต้องสร้างอย่างมั่นคงแข็งแรง บันไดทางขึ้นมีที่ยึดเกาะอย่างปลอดภัย และต้องไม่ตั้งในจุดที่มีลมแรงเกินไป

1.4 บริเวณรอบฐานหอดูไฟจะต้องถางวัชพืชให้หมด เพื่อป้องกันไม่ให้ไฟป่าลุกลามมาสร้างความเสียหายให้แก่หอดูไฟ โดยเฉพาะอย่างยิ่งหอดูไฟที่สร้างด้วยไม้

1.5 จุดที่ตั้งหอดูไฟ ต้องสามารถหมายถึงลงในแผนที่ได้อย่างถูกต้องแน่นอน ทั้งนี้เพื่อให้เครื่องเล็งทิศทางไฟสามารถวัดมุมและหาจุดที่เกิดไฟป่าได้อย่างแม่นยำ

1.6 จะต้องมียุทธวิธีประจำหอดูไฟอย่างครบถ้วน ดังนี้คือ

- เครื่องเล็งทิศทางไฟ
- แผนที่ภูมิประเทศ มาตรฐาน 1 : 50,000

- กล้องส่องทางไกล
- อุปกรณ์ดับไฟฟ้า 1 ชุดเล็ก (ที่ดับไฟ ถังฉีดน้ำดับไฟ และครอบไฟฟ้า อย่างละ 2 อัน)
- อุปกรณ์การสื่อสาร
- สมุดจดรายงานไฟฟ้า

2. ข้อดีของการตรวจหาไฟโดยหอดูไฟ

- 2.1 สามารถตรวจการณ์พื้นที่ได้กว้างมากในเวลาเดียวกัน
- 2.2 ประหยัดค่าใช้จ่าย โดยเฉพาะค่าน้ำมันเชื้อเพลิง
- 2.3 สามารถกำหนดตำแหน่งที่เกิดไฟฟ้าได้อย่างแม่นยำ โดยการใช้เครื่องเล็งทิศทางไฟ
- 2.4 ใช้เป็นแม่ข่ายในการติดต่อสื่อสารทางวิทยุ เพื่อประสานงานการดับไฟฟ้าในพื้นที่นั้น ๆ ได้ในตัว
- 2.5 ใช้เป็นที่ติดตั้งอุปกรณ์ตรวจวัดอากาศ โดยเฉพาะอย่างยิ่งเครื่องวัดความเร็วและทิศทางลม เพื่อการคำนวณพฤติกรรมของไฟฟ้า
- 2.5 ใช้เป็นที่ชมวิวดูในช่วงนอกฤดูไฟฟ้า

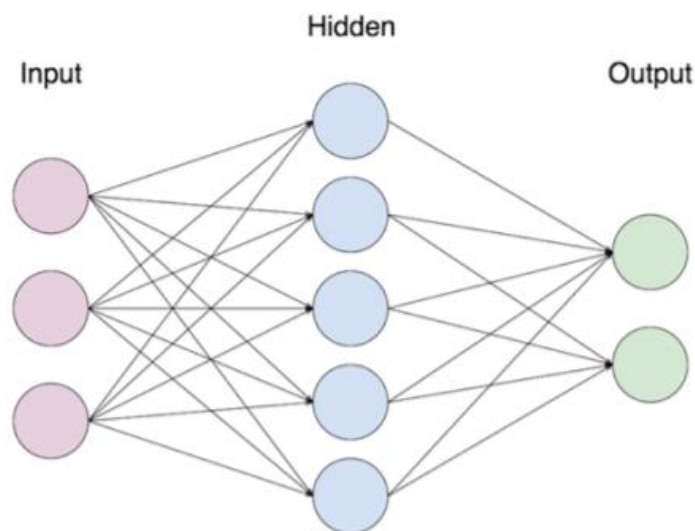
3. ข้อเสียของการตรวจหาไฟโดยหอดูไฟ

- 3.1 มีความยากลำบากในการก่อสร้าง เพราะมักต้องก่อสร้างในพื้นที่ป่าที่ไม่มีถนนหนทางและอยู่ในที่สูง ซึ่งเป็นอุปสรรคสำคัญในการขนส่งวัสดุอุปกรณ์การก่อสร้าง
- 3.2 ค่าใช้จ่ายในการก่อสร้างสูง
- 3.3 ต้องมีค่าใช้จ่ายในการบำรุงรักษาอยู่เสมอ เพื่อให้หอมีความมั่นคงแข็งแรงและปลอดภัยสำหรับผู้ปฏิบัติงาน
- 3.4 ไม่เหมาะสำหรับพื้นที่ที่เป็นภูเขาสลับซับซ้อน เพราะจะเกิดจุดอับที่ตรวจการณ์ไม่เห็นเป็นจำนวนมาก เช่นในร่องเขา ในหุบเขา และด้านหลังเขา เป็นต้น
- 3.5 หากทัศนวิสัยไม่ดี อันเนื่องมาจากเมฆ หรือหมอกควัน จะทำให้ประสิทธิภาพในการตรวจการณ์ลดลงมาก

3.6 เจ้าหน้าที่ประจำหอดูไฟ เกิดความเครียดและความเบื่อหน่ายในการปฏิบัติงานได้ง่าย เนื่องจาก การต้องอยู่บนหอคนเดียวเป็นเวลานาน โดยมีกิจกรรมที่ซ้ำซากจำเจเพียงอย่างเดียวคือการเพ่งมอง พื้นที่รอบตัว จึงมีบ่อยครั้งที่เจ้าหน้าที่เผลอหลับ หรือบางครั้งนั่งอ่านหนังสือแก้เหงาจนลืมตรวจหาไฟ ไปในบางเวลา

2.2 neural network [7]

โครงข่ายประสาท หรือ Neural Network [7] เป็นโมเดลที่อาศัยการเรียนรู้จากชุดข้อมูล ที่เลียนแบบการทำงานของเซลล์ประสาทด้วยพีชคณิตเชิงเส้น สามารถแสดงในรูปของกราฟได้โดยมี โหนดแทนเซลล์ประสาท และเส้นเชื่อมระหว่างโหนดแทนปลายประสาท

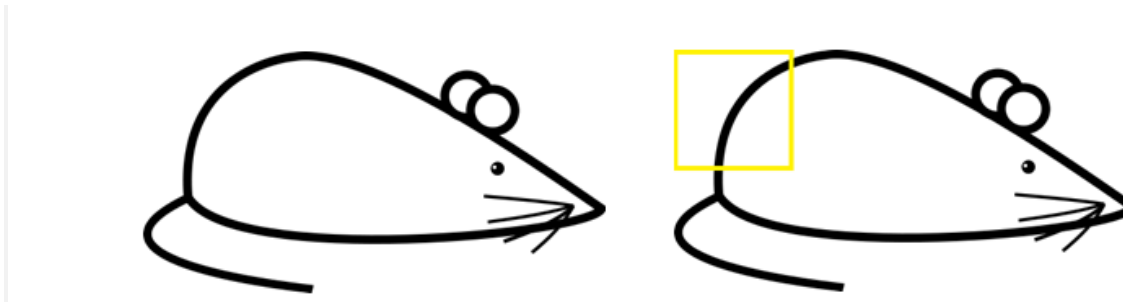


รูปที่ 2.3.1 Neural Network

2.3 Convolutional Neural Network [2]

Convolutional Neural Network (CNN) หรือ โครงข่ายประสาทแบบคอนโวลูชัน เป็น โครงข่ายประสาทเทียมหนึ่งในกลุ่ม bio-inspired โดยที่ CNN จะจำลองการมองเห็นของมนุษย์ที่มอง พื้นที่เป็นที่ย่อย ๆ และนำกลุ่มของพื้นที่ย่อย ๆ มาผสมกัน เพื่อดูว่าสิ่งที่เห็นอยู่เป็นอะไรกันแน่

การมองพื้นที่ย่อยของมนุษย์จะมีการแยกคุณลักษณะ (feature) ของพื้นที่ ย่อยนั้น เช่น ลายเส้น และการตัดกันของสี ซึ่งการที่มนุษย์รู้ว่าพื้นที่ตรงนี้เป็นเส้นตรงหรือสี่เหลี่ยม เพราะมนุษย์ดูทั้งจุดที่สนใจและบริเวณรอบ ๆ ประกอบกัน



รูปที่ 2.3.1 แสดงการมองของมนุษย์

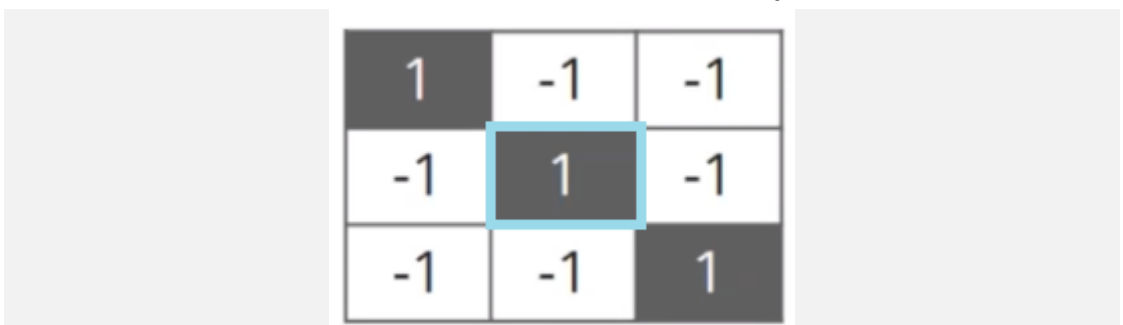
ถ้าเราเปรียบว่ากรอบสี่เหลี่ยมสีเหลืองนั้น คือพื้นที่ที่มนุษย์กำลังให้ความสนใจอยู่ แต่เราสามารถรับรู้ได้ว่าสิ่งนี้คือหนู เพราะเรากวาดสายตามองรอบ ๆ

Feature Extraction

แนวคิดของ CNN นั้นค่อนข้างเป็นแนวคิดที่ดีมาก แต่สิ่งที่ซับซ้อนของมันคือระบบการคำนวณที่สอดคล้องกับแนวคิดของมันเองและต้องมีคณิตศาสตร์มารองรับ โดยการคำนวณตามแนวคิดนี้ใช้หลักการเดียวกันกับ คอนโวลูชันเชิงพื้นที่ (Spatial Convolution) ในการทำงานด้าน อิมเมจโปรเซสซิง (Image Processing) การคำนวณนี้จะเริ่มจากการกำหนดค่าใน **ตัวกรอง (filter)** หรือ **เคอร์เนล (kernel)** ที่ช่วยดึงคุณลักษณะที่ใช้ในการรู้จำวัตถุออก โดยปกติตัวกรอง/เคอร์เนลอันหนึ่งจะดึงคุณลักษณะที่สนใจออกมาได้หนึ่งอย่าง เราจึงจำเป็นต้องตัวกรองหลายตัวกรองด้วย เพื่อหาคุณลักษณะทางพื้นที่หลายอย่างประกอบกัน

ลักษณะของ Filter

สำหรับ Filter ของภาพดิจิทัลนั้น โดยปกติแล้วจะเป็นตารางสองมิติที่มีขนาดตามพื้นที่ย่อย ๆ ที่เราอยากพิจารณา สมมุติว่าถ้าเราต้องการหาเส้นตรงทแยงสีขาว ตัวกรองของเราอาจจะอยู่ในลักษณะนี้



รูปที่ 2.3.2 ตัวกรอง 3x3 สำหรับหาเส้นตรงทแยงสีขาว

ตำแหน่งตรงกลางที่มีกรอบสี่ฟ้าคือ Anchor ที่เอาไว้หาบนพิกเซล(pixel)ของภาพข้อมูลเข้า ตัวกรองจะถูกหาบนพิกเซล(pixel)แรกๆของภาพข้อมูลเข้า จากนั้นจะถูกเลื่อนไปหาบนพิกเซล(pixel)อื่น

ในภาพทีละพิกเซล(pixel)จนครบทุกพิกเซล(pixel)ในภาพ เราอาจจะไม่หาตัวกรองบนพิกเซล (pixel)ที่อยู่ใกล้กรอบภาพ เพราะตัวกรองจะล้นออกไปนอกภาพ เมื่อเราเลื่อนตัวกรองไปเรื่อย ๆ จนครบทุกพิกเซล(pixel)ที่สามารถเลื่อนได้ในภาพ สิ่งที่เราได้นั้นจะเป็นสิ่งที่เรียกว่า **ฟังก์ชันลักษณะ (feature map)**

1x1	1x0	1x1	0	0
0x0	1x1	1x0	1	0
0x1	0x0	1x1	1	1
0	0	1	1	0
0	1	1	0	0

4		

รูปที่ 2.3.3 ฟังก์ชันลักษณะ (feature map)

ซ้าย: filter เลื่อนไปบน input. ขวา: ผลลัพธ์การคำนวณซึ่งออกมาเป็น Feature map

Stride และ Padding

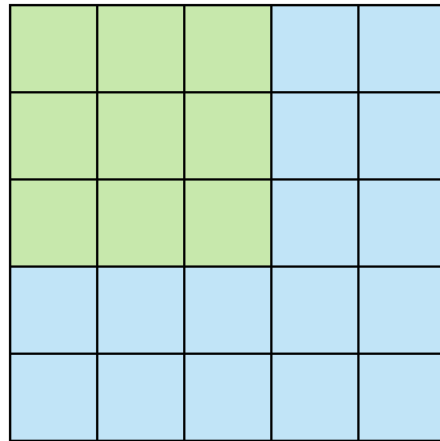
Stride เป็นตัวกำหนดว่าเราจะเลื่อนตัวกรอง (filter) ไปด้วย Step เท่าไร (ตัวอย่างด้านล่างกำหนด Stride เท่ากับ 1)

Stride 1

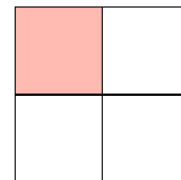
Feature Map

รูปที่ 2.3.4 ฟังก์ชันลักษณะ (feature map) และ Stride

เราสามารถกำหนดค่าของ Stride ให้มากขึ้นก็ได้ ถ้าเราต้องการให้การคำนวณหาคุณลักษณะมีพื้นที่ทับซ้อนกันน้อยขึ้น แต่อย่างไรก็ตามการกำหนดค่าของ Stride ที่มากขึ้นจะทำให้เราได้ฟังก์ชันลักษณะ (feature map) ที่มีขนาดเล็กลง



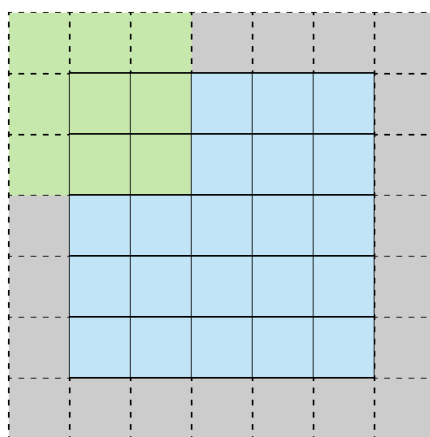
Stride 2



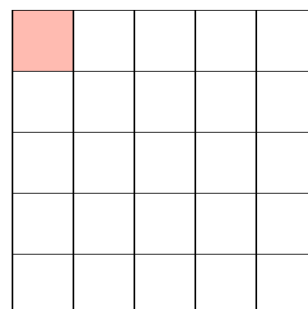
Feature Map

รูปที่ 2.3.5 ฟังก์ชันลักษณะ (feature map) และ Stride

Padding จากรูปด้านล่างเราจะเห็นว่าพื้นที่สีเทารอบ ๆ Input พื้นที่เหล่านี้เป็นพื้นที่ที่เราเพิ่มเข้าไป โดยอาจจะเป็นเต็ม 0 หรือค่าต่าง ๆ เข้าไป เพื่อให้เวลาในการทำ CNN นั้น Feature Map ที่ได้ยังคงมีขนาดเท่ากับ Input



Stride 1 with Padding



Feature Map

รูปที่ 2.3.6 ฟังก์ชันลักษณะ (feature map) และ padding

ในบางปัญหา Input ที่อยู่ตามขอบภาพอาจมีความสำคัญที่ส่งผลต่อการตัดสินใจบางอย่าง เราจึงจำเป็นต้องเก็บคุณลักษณะตามขอบของรูปภาพไว้ด้วย

Max Pooling

ก่อนอื่นเรามองลองดูหนึ่งในปัญหาของการทำ CNN กันก่อน สมมติเราใช้ CNN ด้วยขนาดตัวกรอง 3x3 พิกเซล (pixel) แต่เรารู้ดีว่าเวลาเรามองภาพแล้วเราตอบได้ว่ามันคืออะไร เพราะเรามองไปในบริเวณที่กว้างกว่านั้น



รูปที่ 2.3.7 การสเกลภาพ

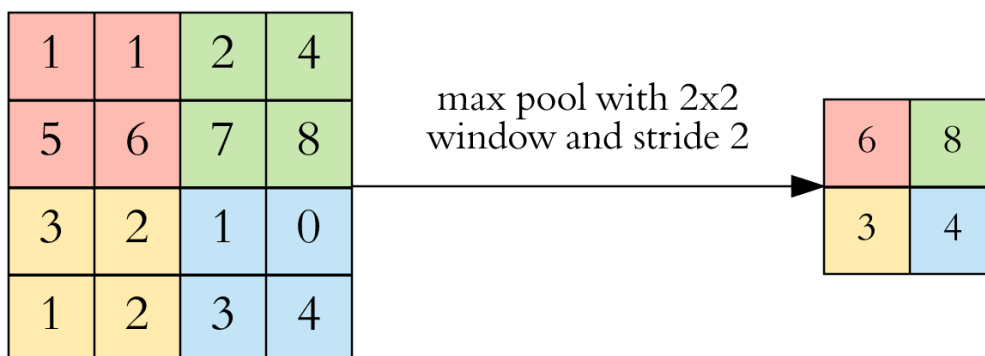
จากภาพ จะเห็นว่าต่อให้รูปภาพมีขนาดสเกลที่เล็กลง แต่เราก็ยังสามารถมองออกว่ามันคือเครื่องปั้นดินเผา แสดงว่า เราจำแนกวัตถุขึ้นนี้ที่ความละเอียดต่ำลง แต่เรากำลังทำ CNN ที่ความละเอียดสูง

Multiscale Analysis จากปัญหาด้านบนเราจะเห็นว่าเป็นไปได้ยากมากหากเราต้องอาศัยข้อมูลที่หายาหรือละเอียดอย่างใดอย่างหนึ่งในการจำแนกวัตถุ ดังนั้นในการฝึกเครื่องเราจึงจะเป็นต้องมีข้อมูลทั้งหายาและละเอียดควบคู่กันไป

ตอนนี้เรารู้แล้วว่าเราจำเป็นต้องคำนวณภาพในหลายสเกล แต่ปัญหาที่สำคัญคือเราจะทำให้การคำนวณอยู่ในรูปหลายสเกลได้อย่างไร หากเราใช้ตัวกรองขนาด 3x3 เรากำลังจะจัดการกับรายละเอียดเล็ก ๆ (ภาพใหญ่มีรายละเอียดมาก จึงถือว่าเป็นสเกลละเอียด) แต่ด้วยตัวกรองขนาดเท่าเดิม หากทำกับภาพที่ขนาดเล็กแล้ว มันจะครอบคลุมพื้นที่วัตถุเดิมมากขึ้น ดังนั้นถ้าโครงข่ายเราควรจะต้องมีการย่อรูปประกอบด้วย เราก็จะสามารถเข้าถึงความสามารถด้านการวิเคราะห์หลายความละเอียดได้

Pooling คือความสามารถในการย่อรูปแบบหนึ่ง ซึ่งมีสองประเภทหลักที่นิยมกันคือ **max pooling** และ **mean pooling**

Max Pooling เป็นตัวกรองแบบหนึ่งที่ทำค่าสูงสุดในบริเวณที่ตัวกรองทาอยู่มาเป็นผลลัพธ์ โดยเราจะเตรียมตัวกรองในลักษณะเดียวกับการทำ Feature Extraction ของ CNN มาทาบบนข้อมูลแล้วเลือกค่าที่สูงที่สุดในตัวกรองนั้นมาเป็นผลลัพธ์ใหม่ และจะเลื่อนตัวกรองไปตาม Stride ที่กำหนดไว้ โดยขนาดตัวกรองของการทำ max pooling จะนิยมเรียกกันว่า **pool size**



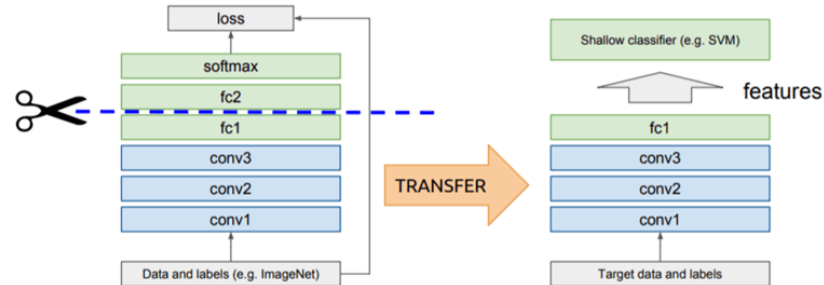
รูปที่ 2.3.8 max pooling

2.4 Transfer Learning [8]

โมเดล Deep Learning หลาย ๆ ตัวที่เราใช้อยู่ มีความซับซ้อน มี Parameter (Weight) จำนวนหลายล้านตัว การเริ่มต้นเทรนโมเดล Deep Learning ที่ซับซ้อนขนาดนี้ ตั้งแต่ต้น (Weight Initialization ด้วยค่า Random) ต้องใช้ทั้งข้อมูล Dataset ขนาดใหญ่ พลังการประมวลผลมหาศาล และเวลาหลายวันจนถึงหลายสัปดาห์ Transfer Learning คือ เทคนิคที่ช่วยลดเวลาการเทรนโมเดล Deep Learning ด้วยการนำบางส่วนของโมเดลที่เทรนเรียบร้อยแล้ว กับงานที่ใกล้เคียงกัน มาใช้เป็นส่วนหนึ่งของโมเดลใหม่ ในทางปฏิบัติ มีคนจำนวนน้อยมากที่เทรน Convolutional Neural Network ตั้งแต่ต้น เนื่องจากไม่มีชุดข้อมูล Dataset ที่ใหญ่พอ ดังนั้นคนส่วนใหญ่จึงใช้วิธีนำโมเดล ConvNet ที่เทรนกับชุดข้อมูล Dataset ขนาดใหญ่ (เช่น ImageNet ที่มีข้อมูลตัวอย่างจำนวน 1.2 ล้านรูป ประกอบด้วย 1000 หมวดหมู่) นำโมเดลนั้นมาเป็นโมเดลตั้งต้นเพื่อเทรนต่อ กับ Dataset ขนาดเล็กในงานเฉพาะทาง หรือ ใช้สกัด Feature สำหรับงานที่ต้องการออกมา

Idea: use outputs of one or more layers of a network trained on a different task as generic feature detectors. Train a new shallow model on these features.

Assumes that $D_S = D_T$



รูปที่ 2.4.1 การทรานเฟอร์เลินนิ่ง

การใช้ Transfer Learning ส่วนใหญ่ แบ่งเป็น 3 แบบดังนี้

- ใช้ ConvNet เป็น Fixed Feature Extractor – นำ ConvNet มาลบ Dense Layer สุดท้ายออกไป เราจะได้ Feature Extractor ที่เราสามารถสร้าง Linear Classifier (Head) เทรนให้ Classify Feature เหล่านี้ สำหรับงานใหม่ กับชุดข้อมูล Dataset ใหม่ที่มีขนาดเล็กกว่ามาก
- Fine-tuning โมเดล ConvNet – แทนที่เราจะเทรนเฉพาะ Head เราสามารถ Fine-tuning ทั้งโมเดล ConvNet ทุก Layer เพื่อให้ได้ประสิทธิภาพที่ดีขึ้น กับงานใหม่ และ Dataset ใหม่
- Pretrained models – เนื่องจาก ConvNet สมัยใหม่ ต้องใช้เวลาเทรนที่ยาวนานประมาณ 2-3 สัปดาห์ บนเครื่อง Server ความเร็วสูง ที่มีหลาย GPU จึงมีผู้นำ Pretrained models โมเดลที่เทรนเรียบร้อยแล้ว มาแชร์กันในอินเทอร์เน็ต ให้ผู้อื่นได้ใช้ เรียกว่า Model Zoo

ซึ่งในโครงการนี้จะใช้การ transfer learning แบบ Fine-tuning จาก Model Zoo ที่ชื่อ MobileNetV2 ดังนี้

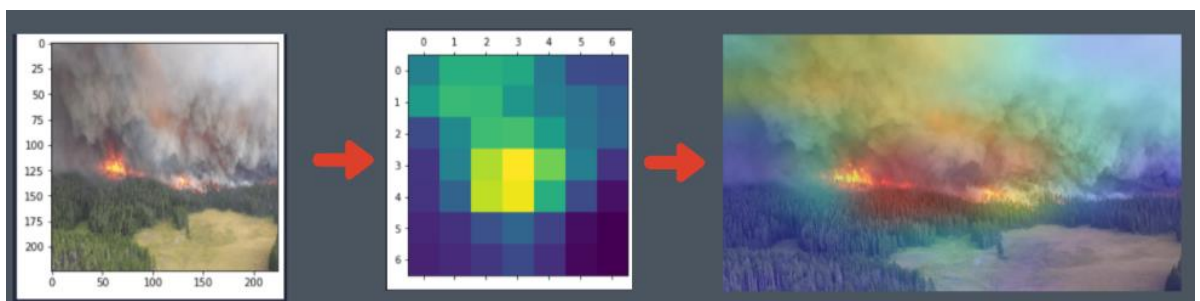
Model: "MobileNetV2"

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
mobilenetv2_1.00_224 (Function)	(None, None, None, 1280)	2257984
avg_pool (GlobalAveragePooling2D)	(None, 1280)	0
batch_normalization (Batch Normalization)	(None, 1280)	5120
top_dropout (Dropout)	(None, 1280)	0
pred (Dense)	(None, 2)	2562
activation (Activation)	(None, 2)	0
Total params: 2,265,666		
Trainable params: 2,228,994		
Non-trainable params: 36,672		

รูปที่ 2.4.2 แสดงรายละเอียดของตัวแบบที่ใช้ในระบบ

2.5 แกรดแคม (grad cam) [9]

หลาย ๆ คนจะมองว่า Neural Network เป็นเหมือนกล่องดำ ข้างในมีแต่ตัวเลข เมตริก เทนเซอร์ Neuron, Activation, Gradient วิ่งไปวิ่งมา โดยที่เราไม่รู้เลยว่า Logic การทำงานภายในของมันเป็นอย่างไร ไม่สามารถเข้าใจได้ แต่เราสามารถใช้เทคนิค Grad-CAM มาช่วยในการตีความ Activation และ Gradient ของโมเดล ทำให้เข้าใจถึงการทำงานภายใน Neural Network มากขึ้นว่าโมเดลพิจารณาจากบริเวณไหน Attention โฟกัสส่วนไหน เป็นพิเศษ โดยจะนำค่าเกรเดียนของโมเดลที่ใช้ในการปรับค่าพารามิเตอร์นำออกมาแปลงเป็น heatmap และนำมา visualize กับรูปภาพ อินพุตเราก็จะได้รูป grad cam ที่แสดงให้เห็นว่าโมเดลมีการให้ความสำคัญกับบริเวณไหนในรูปภาพอินพุตนั่นเอง



รูปที่ 2.5.1 แสดงภาพ grad cam ที่แปลงจาก heatmap

2.6 ความรู้ตามหลักสูตรซึ่งถูกนำมาใช้หรือบูรณาการในโครงการ

ได้นำความรู้เรื่องการจำแนกลักษณะสิ่งต่าง ๆ ทั้งจาก neural network จาก Convolutional Neural Network และจาก Grad cam มาใช้ในโครงการนี้ซึ่งได้เรียนรู้มาจากวิชา major elective ตามหลักสูตร

2.7 ความรู้นอกหลักสูตรซึ่งถูกนำมาใช้หรือบูรณาการในโครงการ

ความรู้ที่ได้ศึกษาเองก็จะเป็นการเขียนเว็บแอปพลิเคชันโดยใช้ react , Flask การใช้ไลบรารี (library) Tensorflow และ Keras ในการพัฒนาระบบ

บทที่ 3

โครงสร้างและขั้นตอนการทำงาน

ในบทนี้จะกล่าวถึงหลักการ และการออกแบบระบบจำแนกลักษณะของรูปที่มีไฟไหม้อยู่ในภาพและไม่มีไฟไหม้อยู่ในภาพโดยอาศัยหลักการ Convolutional Neural Network ในการจำแนกลักษณะของรูปโดยการทำงานของระบบจะมีดังนี้

3.1 การเตรียมชุดข้อมูลเพื่อฝึกสอนตัวแบบ(model)

โดยจะมีชุดข้อมูลที่หามาจากอินเทอร์เน็ตเป็นรูปภาพที่มีควันไฟอยู่ในภาพและรูปภาพที่ไม่มีควันไฟอยู่ในภาพมาทำการฝึกสอนระบบที่ได้จากการ transfer learning แบบ Fine-tuning จาก Model Zoo ที่ชื่อ MobileNetV2 โดยข้อมูลภาพที่ได้รวบรวมมาจะเป็นภาพสี โมเดล RGB นามสกุลไฟล์ jpg โดยจะแบ่งเป็นสองช่วงช่วงแรกเป็นรูปควันไฟและวิวจำนวน 2,723 รูปโดยเป็นรูปควันไฟในลักษณะที่ไม่มีเปลวเพลิงอยู่เลย ภายหลังได้เพิ่มชุดข้อมูลเพิ่มเข้าไปอีกเป็นจำนวน 4,304 รูป แบ่งเป็นรูปภาพที่ไม่มีควันไฟ 2,097 รูปและรูปที่มีควันไฟ 2,207 รูปโดยเป็นรูปควันไฟที่มีเปลวเพลิงและวิวที่มีโทนสีแดงซึ่งภาพที่ได้รวบรวมมาจะมีตัวอย่างดังนี้

- ภาพที่ไม่มีควันไฟ



รูปที่ 3.1.1 ตัวอย่างภาพที่ไม่มีควันไฟ

- ภาพที่มีควันไฟ



รูปที่ 3.1.2 ตัวอย่างภาพที่มีควันไฟ

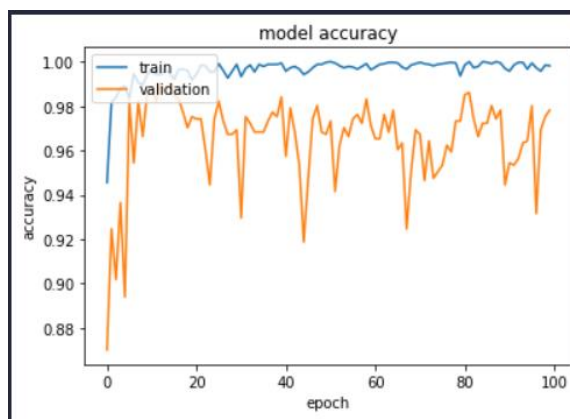
จากนั้นเราก็นำชุดข้อมูลที่ได้นั้นมาทำการเตรียมก่อนจะนำไปฝึกสอนโดยทำการสุมหมุนรูปภาพ สุมชุมรูปภาพ และทำการปรับอัตราส่วนค่าสี (rescale) จากที่ค่าสีมีค่า 0 ถึง 255 เป็น -1 ถึง 1

3.2 การแปลงตัวแบบจำแนกลักษณะมาตรฐานที่มีอยู่แล้วให้เหมาะกับการจำแนกไฟไหม้

โดยเราจะทำการโหลด (load) ตัวแบบสำเร็จรูปที่มีอยู่แล้วในไลบรารี (library) ในที่นี้เราจะใช้ตัวแบบที่ชื่อ MobileNetV2 [5] มาใช้ซึ่งตัวแบบนี้เดิมจะใช้ในการจำแนกลักษณะสิ่งต่าง ๆ 1,000 ลักษณะแต่เราจะนำมาดัดแปลงให้ใช้กับการจำแนกลักษณะของไฟไหม้ คว้นไฟ และรูปปกติเท่านั้น โดยจะนำเฉพาะส่วนเริ่มต้นของตัวแบบมาใช้เพื่อที่จะเอาความสามารถของตัวแบบขั้นต้นในการประเมินขั้นต้นเท่านั้น จากนั้นเราจะทำการสร้างเลเยอร์ (layer) นำมาต่อท้ายตัวแบบ MobileNetV2 [5] ส่วนเริ่มต้นโดยจะมีเลเยอร์ (layer) GlobalAveragePooling2D และ Dense ที่มี activation function เป็น SoftMax โดยมีสองโหนดเพื่อเป็นการจำแนกลักษณะของ คว้นไฟ และปกตินั่นเอง โดยเราเรียกการนำตัวแบบสำเร็จรูปมาดัดแปลงให้ได้การจำแนกลักษณะที่ต้องการนี้ว่า การทรานเฟอร์เลิร์นนิง (Transfer Learning) [3]

3.3 การฝึกสอนตัวแบบ

เมื่อได้ตัวแบบที่จะใช้ในการจำแนกแล้วเราก็นำมาทำการฝึกสอนกับชุดข้อมูลที่เราได้จากข้อที่ 1. แล้วทำการฝึกสอนไปเรื่อย ๆ จนได้ค่าความผิดพลาดที่พอใจ โดยจากการฝึกสอนแล้วพบว่า การฝึกสอนที่ 82 รอบของชุดข้อมูล (epoch) ได้ค่าความแม่นยำในการฝึกสอนมากที่สุดโดยได้ค่าความแม่นยำในการทำนายเท่ากับ 0.9861 ดังภาพประกอบนี้

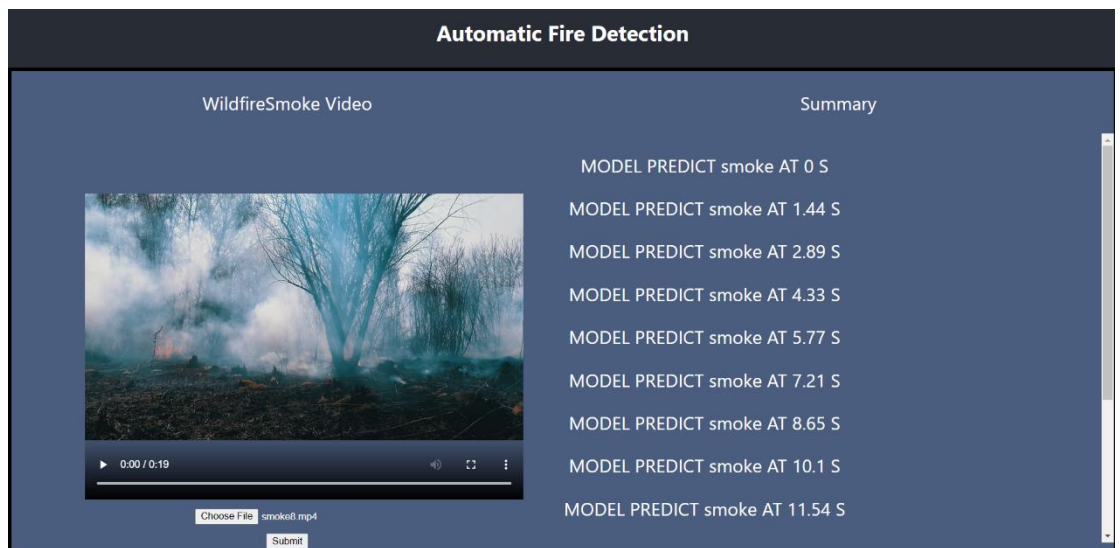


รูปที่ 3.3.1 รูปแสดงค่าความแม่นยำจากการฝึกสอนตัวแบบ

3.4 การสร้างเว็บแอปพลิเคชัน

- การพัฒนาส่วนการแสดงผล (Frontend)

ในส่วนของการพัฒนาส่วนแสดงผลเลือกใช้ React ซึ่งเป็น ไลบรารี JavaScript โดยสามารถใช้งานในรูปแบบเว็บแอปพลิเคชันที่สามารถเข้าได้ทุกอุปกรณ์ที่รองรับเว็บเบราว์เซอร์แต่เหมาะกับอุปกรณ์ pc ที่สุด ซึ่งมีปุ่มอัปโหลดไฟล์วิดีโอที่มีควันไฟ ปุ่ม submit จะมีส่วนแสดงผลการทำนายจากโมเดลจากรูปที่ได้มาจากวิดีโอและแสดงรูปจากแกรดแคม(gradcam)[9] ดังนี้



รูปที่ 3.4.1 ภาพแสดงหน้าเว็บแอปพลิเคชัน

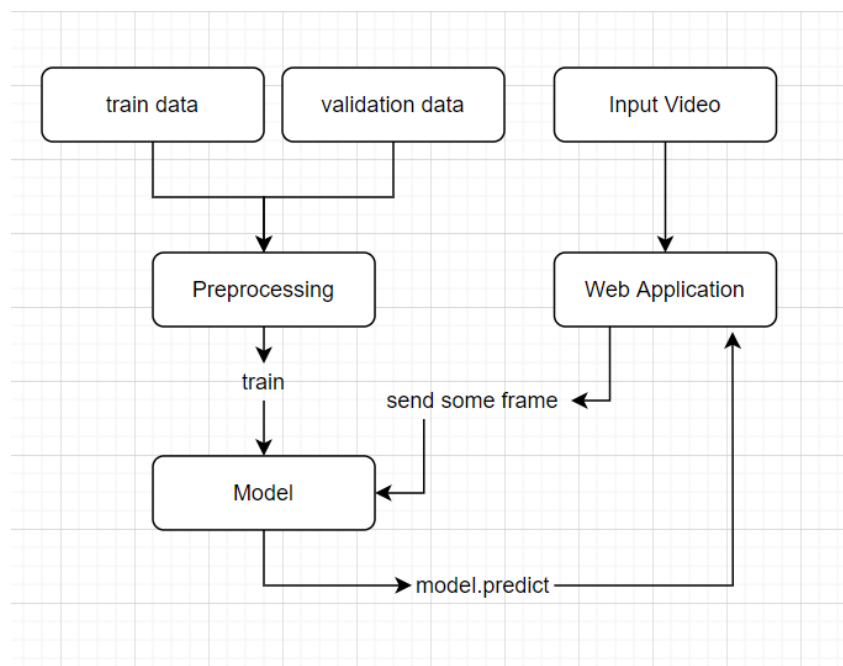


รูปที่ 3.4.2 ภาพแสดงหน้าเว็บแอปพลิเคชัน(ต่อ)

- การพัฒนาส่วนการวิเคราะห์ข้อมูล (Backend)

จะนำโมเดลที่ได้รับการฝึกสอนมาแล้วมาทำการทำนายรูปที่ได้จากเฟรมภาพจาก วิดีโอ อินพุตจากหน้าส่วนการแสดงผลโดยจะทำการแบ่งเฟรมรูปภาพเป็น 14 เฟรมตลอดแล้วนำ ผลการทำนายที่ได้และรูปเกรดแคม(grad cam)[9] ที่ทำจากรูปที่ถูกทำนายเป็นคว้นไฟใน เฟรมแรกส่งกลับไปยังส่วนการแสดงผลโดยใช้เครื่องมือพัฒนาเป็น flask ที่ใช้ภาษา python ในการพัฒนา

- เส้นทางการไหลของข้อมูล (Dataflow)

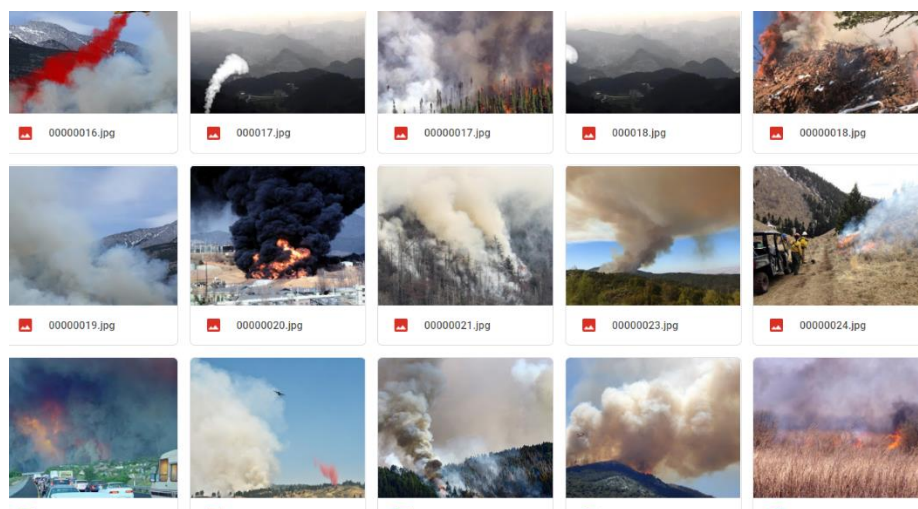


รูปที่ 3.4.3 ภาพแสดงเส้นทางการไหลของข้อมูล

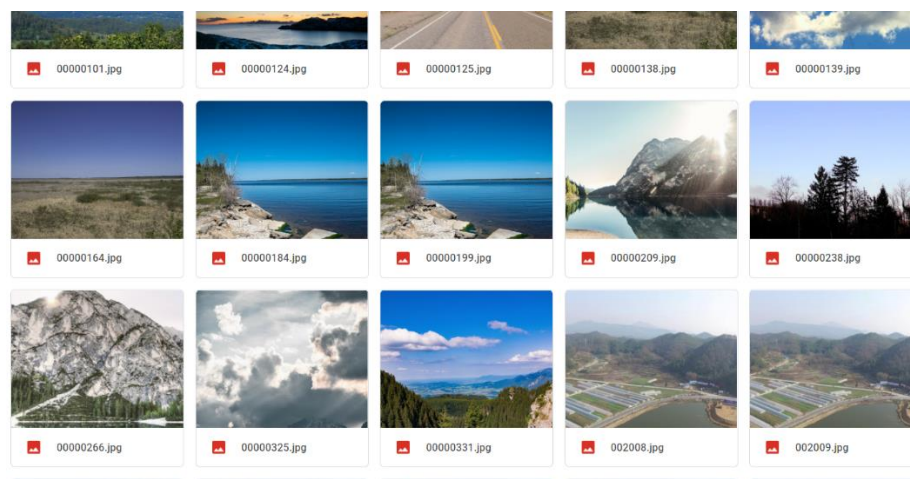
บทที่ 4

การทดลองและผลลัพธ์

ในบทนี้จะทดสอบเกี่ยวกับความถูกต้องในการทำนายของตัวแบบจากข้อมูลที่ไม่เคยเห็นมาก่อน (Blind Test) โดยจะแบ่งเป็นการทดสอบจากรูปภาพที่ทำการรวบรวมมาเพื่อทำการทดสอบ และการทดสอบจากรูปภาพที่ได้จากวิดีโอที่ได้มาจากการทดสอบเว็บแอปพลิเคชันโดยชุดข้อมูลทดสอบมีจำนวน 125 รูปแบ่งเป็น รูปที่ไม่มีควันไฟ 62 รูปและรูปที่มีควันไฟ 63 รูป ตัวอย่างดังนี้



รูปที่ 4.0.1 แสดงตัวอย่างชุดข้อมูลทดสอบที่มีควันไฟ



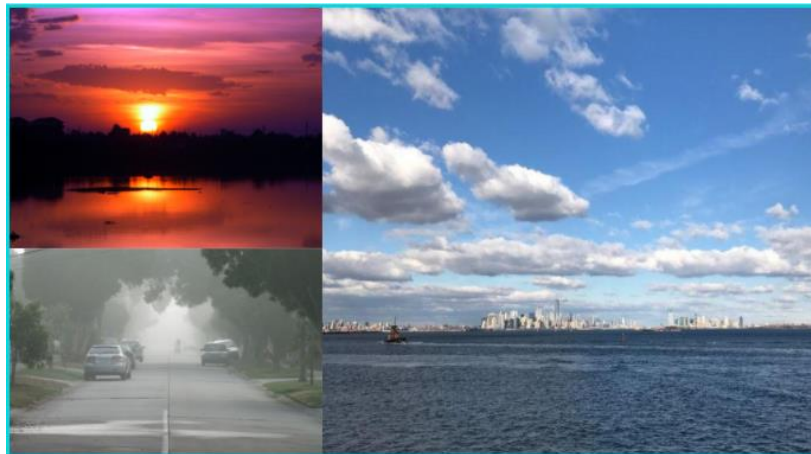
รูปที่ 4.0.2 แสดงตัวอย่างชุดข้อมูลทดสอบที่ไม่มีควันไฟ

4.1 การทดสอบกับชุดข้อมูลรูปภาพกับตัวแบบในชุดข้อมูลชุดแรก

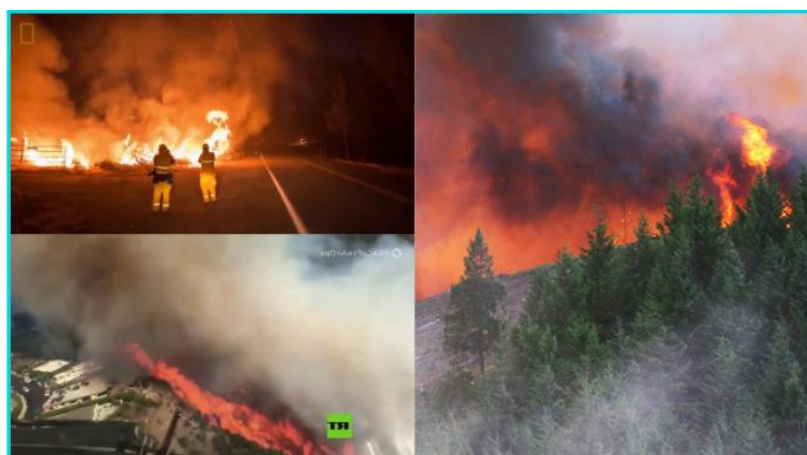
จากบทที่สามได้กล่าวไว้ว่าข้อมูลในการฝึกสอนตัวแบบนั้นมีสองชุดข้อมูลโดยในการทดสอบนี้จะกล่าวถึงการทดสอบตัวแบบที่ได้รับการฝึกสอนจากชุดข้อมูลชุดแรกซึ่งเป็นภาพที่มีลักษณะเป็นควันไฟที่ไม่มีเปลวเพลิงอยู่ในรูปภาพเลย โดยได้ค่าความถูกต้องของการทำนายเพียง 0.77 เท่านั้นโดยการตรวจสอบแต่ละรูปพบว่ารูปที่ตัวแบบทำนายผิดนั้นส่วนใหญ่เป็นรูปควันไฟที่มีเปลวเพลิงอยู่ในรูปทำให้ความน่าเชื่อถือของตัวแบบยังไม่เป็นที่น่าพึงพอใจ

4.2 การทดสอบกับชุดข้อมูลรูปภาพกับตัวแบบในชุดข้อมูลชุดที่สอง

จากข้อที่ 4.1 เราจึงได้ทำการเพิ่มข้อมูลฝึกสอนเข้าไปอีกโดยเน้นรูปควันไฟในลักษณะที่มีเปลวเพลิงอยู่ในรูปด้วย และในส่วนรูปที่ไม่มีควันไฟเราก็เน้นรูปที่มีลักษณะโทนสีแดงและรูปที่มีทัศนวิสัยเป็นหมอกควันเพิ่มเข้าไปอีกด้วยโดยสุดท้ายแล้วเราได้ชุดข้อมูลฝึกสอนทั้งหมด 4,304 รูปแบ่งเป็นรูปภาพที่ไม่มีควันไฟ 2,097 รูปและรูปที่มีควันไฟ 2,207 รูปโดยรูปที่เพิ่มเข้ามาจะมีตัวอย่างดังนี้



รูปที่ 4.2.1 แสดงบางส่วนของรูปภาพที่ไม่มีควันไฟที่เพิ่มเข้ามา



รูปที่ 4.2.2 แสดงบางส่วนของรูปภาพที่มีควันไฟที่เพิ่มเข้ามา

โดยเมื่อนำตัวแบบที่ได้รับการฝึกสอนจากชุดข้อมูลล่าสุดนี้แล้วได้ค่าความถูกต้องจากการทำนายเป็น 0.97 ซึ่งเพิ่มมากขึ้นกว่าเดิมในระดับที่น่าพึงพอใจโดยรายละเอียดแบ่งเป็นดังนี้

ข้อมูลทดสอบทั้งหมด 125 รูปแบ่งเป็น รูปที่ไม่มีควันไฟ 62 รูปและรูปที่มีควันไฟ 63 รูป

True	False	True	False
Positive	Positive	Negative	Negative
62	0	60	3

ตารางที่ 4.2.1 แสดงรายละเอียดการทำนาย

True Positive คือกรณีที่รูปไม่มีควันไฟและโมเดลทำนายว่าไม่มีควันไฟ

False Positive คือกรณีที่รูปไม่มีควันไฟและโมเดลทำนายว่ามีควันไฟ

True Negative คือกรณีที่รูปมีควันไฟและโมเดลทำนายว่ามีควันไฟ

False Negative คือกรณีที่รูปมีควันไฟและโมเดลทำนายว่าไม่มีควันไฟ

4.3 การทดสอบกับชุดข้อมูลรูปภาพที่ได้มาจากวิดีโออินพุตในเว็บแอปพลิเคชัน

เนื่องจากเราได้ทำการพัฒนาเว็บแอปพลิเคชันขึ้นมาเพื่อจำลองการทำงานของระบบเราจึงได้ทำการหาวิดีโอเพื่อนำมาใช้ทดสอบการทำงานของระบบโดยวิดีโอที่นำมาใช้มีทั้งหมด 14 วิดีโอ แบ่งเป็นวิดีโอที่มีควันไฟ 10 วิดีโอ วิดีโอที่ไม่มีควันไฟ 3 วิดีโอ และวิดีโอที่มีทั้งควันไฟและไม่มีควันไฟ

ในวิดีโอเดียวกัน 1 วิดีโอ โดยแต่ละวิดีโอจะได้รูปภาพทั้งหมด 14 รูปจำนวน 14 วิดีโอเราก็จะได้จำนวนรูปภาพทั้งหมด 196 รูปโดยมีรายละเอียดการทำนายดังนี้

True Positive	False Positive	True Negative	False Negative
47	1	104	44

ตารางที่ 4.3.1 แสดงรายละเอียดการทำนาย

True Positive คือกรณีที่รูปไม่มีควันไฟและโมเดลทำนายว่าไม่มีควันไฟ

False Positive คือกรณีที่รูปไม่มีควันไฟและโมเดลทำนายว่ามีควันไฟ

True Negative คือกรณีที่รูปมีควันไฟและโมเดลทำนายว่ามีควันไฟ

False Negative คือกรณีที่รูปมีควันไฟและโมเดลทำนายว่าไม่มีควันไฟ

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 สรุปผล

ในการทำโครงงานนี้สามารถพัฒนาเว็บแอปพลิเคชันที่รับอินพุตเป็นวิดีโอแล้วนำมาทำการแบ่งเฟรมแล้วส่งให้ตัวแบบทำการทำนายและส่งค่ากลับมาแสดงผลได้อย่างถูกต้องในด้านของการทำงานของระบบถือว่าทำงานได้อย่างถูกต้องตามเส้นทางการไหลของข้อมูล แต่ในการด้านของความถูกต้องในการทำนายของตัวแบบนั้นจะเห็นได้จากผลในบทที่ 4 ที่เห็นได้ว่าแม้ในการทดสอบโดยใช้ข้อมูลทดสอบที่ไม่เคยเห็นมาก่อน (Blind Test) จะได้ค่าความถูกต้องในการทำนายถึง 0.97 แต่เมื่อนำมาทดสอบกับวิดีโออินพุตที่ถือว่าเป็นข้อมูลทดสอบที่ไม่เคยเห็นมาก่อน (Blind Test) เช่นกัน จะเห็นว่าค่าความถูกต้องในการทำนายได้เพียง 0.77 (151 จาก 196) ทำให้ตัวแบบที่ใช้ในการทำนายยังมีความไม่แน่นอนในการทำนายอยู่อีกทั้งการทำนายที่ผิดส่วนใหญ่ยังเป็นกรณี False Negative อีกด้วยซึ่งเป็นกรณีที่ควรเกิดขึ้นน้อยที่สุดจึงทำให้ตัวแบบนี้ยังคงต้องรับการฝึกสอนต่อไปเพื่อให้มีความแน่นอนในการทำนายเพิ่มขึ้นมากกว่านี้ แต่จากการตรวจสอบการให้ความสำคัญในการทำนายของตัวแบบจากรูปภาพเกรดแคม (grad cam) ทำให้พบว่าตัวแบบยังคงทำนายรูปภาพจากบริเวณที่มีควันไฟอยู่ในรูปภาพจริง ๆ ทำให้เชื่อได้ว่าการฝึกสอนตัวแบบดำเนินมาอย่างถูกต้องแล้วแต่อาจจะยังคงสับสนกับวัตถุที่คล้ายควันไฟอย่างอื่นอีก อาทิเช่น ก้อนเมฆ จึงต้องมีการค้นคว้าและฝึกสอนต่อไป

5.2 ปัญหาที่พบและแนวทางแก้ไข

ข้อมูลที่นำมาใช้ในการฝึกสอนนั้นในรูปแบบมุมมองที่ต้องการยังคงหาได้ยากที่จะสามารถเข้าถึงและนำมาได้โดยง่าย จึงต้องทำการหาชุดข้อมูลฝึกสอนให้มากกว่านี้เพื่อให้ตัวแบบทำนายได้แน่นอนมากขึ้น

5.3 ข้อเสนอแนะและแนวทางการพัฒนาต่อ

ควรทำการหาชุดข้อมูลฝึกสอนให้มากและหลากหลายกว่านี้เพื่อที่ตัวแบบจะได้สามารถจำแนกลักษณะที่เป็นควันไฟได้โดยไม่สับสนกับวัตถุอื่น ๆ ที่มีลักษณะคล้ายกันเช่นก้อนเมฆ รวมถึงโทนสีของรูปภาพที่ทำนายรูปควันไฟในโทนสีออกน้ำตาลไม่ได้เพราะว่าชุดข้อมูลฝึกสอนเป็นรูปภาพโทนสีสดอาจจะหารูปภาพในโทนสีต่าง ๆ เพิ่มขึ้นเพื่อในกรณีที่เจอรูปควันไฟที่มีหมอกควันบดบังจนออกมาในลักษณะโทนสีน้ำตาล

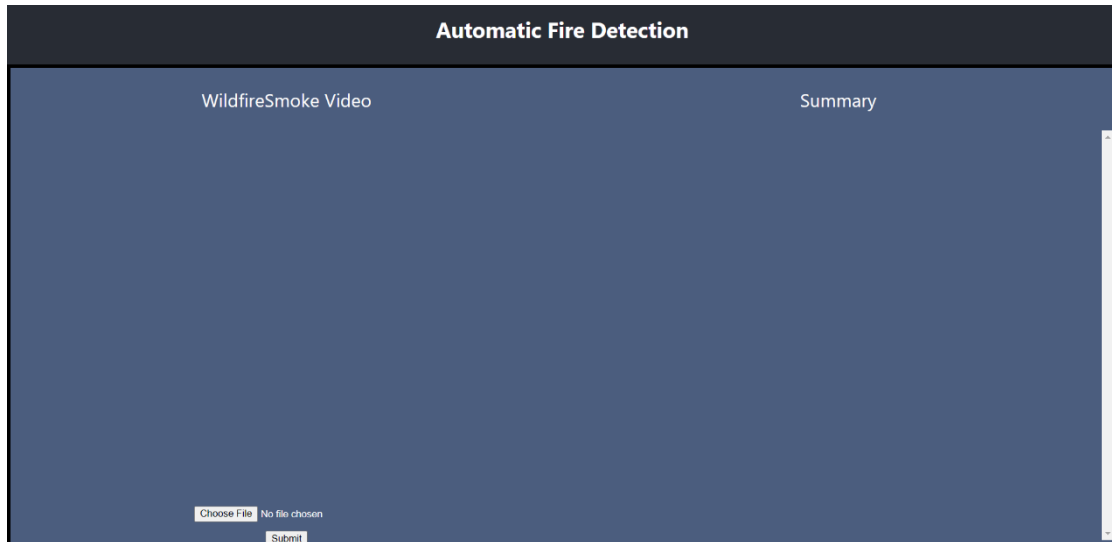
เอกสารอ้างอิง

- [1] ส่วนควบคุมไฟฟ้า. (ม.ป.ป). การตรวจหาไฟทางกิ่งอากาศ.สืบค้น 20 ตุลาคม 2563, เข้าถึงได้จาก http://www.dnp.go.th/forestfire/FIRESCIENCE/lesson%205/lesson5_3.htm
- [2] Natthawat Phongchit. (2018). *Convolutional Neural Network (CNN) คืออะไร*.สืบค้น 20 ตุลาคม 2563, เข้าถึงได้จาก <https://medium.com/@natthawatphongchit/%E0%B8%A1%E0%B8%B2%E0%B8%A5%E0%B8%AD%E0%B8%87%E0%B8%94%E0%B8%B9%E0%B8%A7%E0%B8%B4%E0%B8%98%E0%B8%B5%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B8%84%E0%B8%B4%E0%B8%94%E0%B8%82%E0%B8%AD%E0%B8%87-cnn-%E0%B8%81%E0%B8%B1%E0%B8%99-e3f5d73eebaa>
- [3] Francois Chollet. (2020). *Transfer learning & fine-tuning*.สืบค้น 20 ตุลาคม 2563, เข้าถึงได้จาก https://keras.io/guides/transfer_learning/
- [4] Francois Chollet. (2020). *Keras*.สืบค้น 20 ตุลาคม 2563, เข้าถึงได้จาก <https://keras.io/about/>
- [5] Mark Sandler. (2019). *MobileNetV2*.สืบค้น 20 ตุลาคม 2563, เข้าถึงได้จาก <https://arxiv.org/abs/1801.04381v4>
- [6] Abadi, Martin, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin et al. "Tensorflow: A system for large-scale machine learning." In 12th {USENIX} symposium on operating systems design and implementation (OSDI 16), pp. 265-283. 2016.
- [7] Wikipedia.(2020).neural network.สืบค้น 20 ตุลาคม 2563, เข้าถึงได้จาก https://en.wikipedia.org/wiki/Neural_network
- [8] KENG SURAPONG. (2020). *Transfer Learning คืออะไร*.สืบค้น 10 มีนาคม 2563, เข้าถึงได้จาก <https://www.bualabs.com/archives/3493/what-is-transfer-learning-build-headless-mobilenet-model-transfer-learning-machine-learning-mobilenet-json-retrain-webcam-tensorflow-js-tfjs->
- [9] KENG SURAPONG. (2019). *Visualization เจาะลึกภายใน Neural Network วิเคราะห์ Activation และ Gradient ด้วย Heatmap และ Grad-CAM คืออะไร*.สืบค้น 25 มีนาคม 2563, เข้าถึงได้จาก <https://www.bualabs.com/archives/2008/visualization-inside-neural-network-analyze-activation-gradient-heatmap-gradcam-convnet-ep-4/>

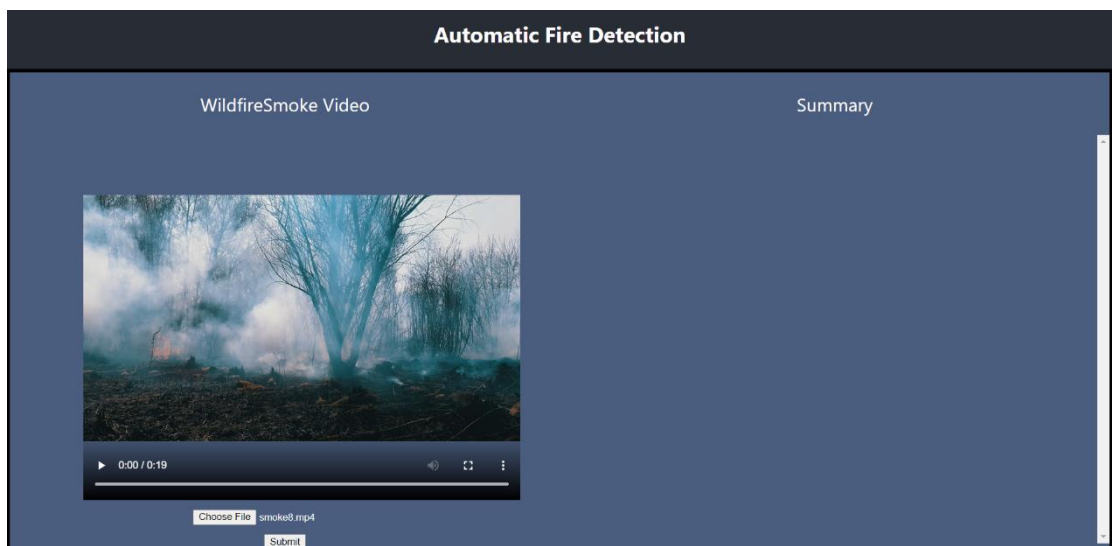
ภาคผนวก

ภาคผนวก ก การใช้งานเว็บแอปพลิเคชัน

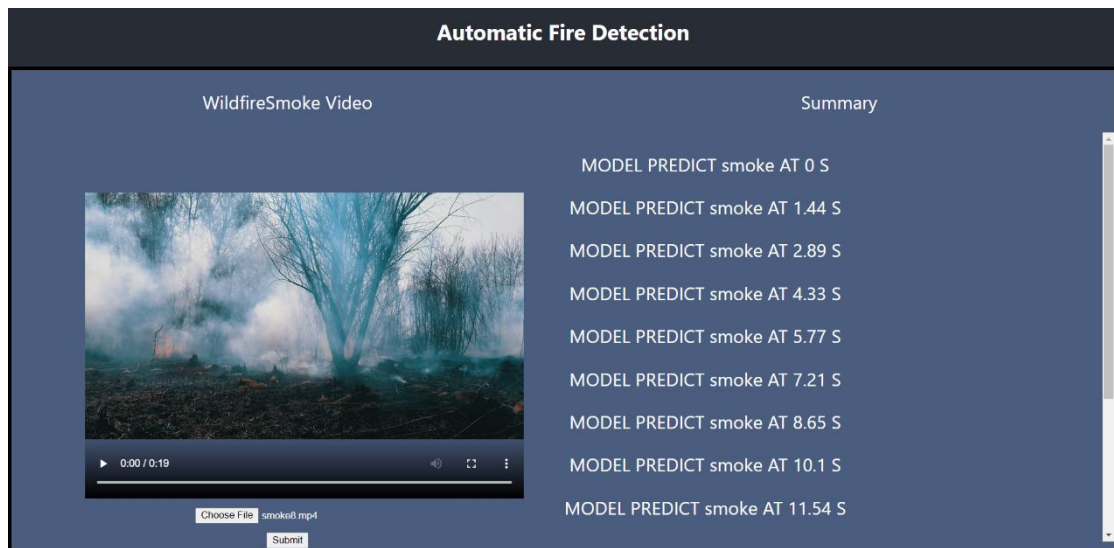
ก.1. รูปภาพเว็บแอปพลิเคชันตอนเริ่มต้น



ก.2. รูปภาพเว็บแอปพลิเคชันตอนอัปโหลดวิดีโอ



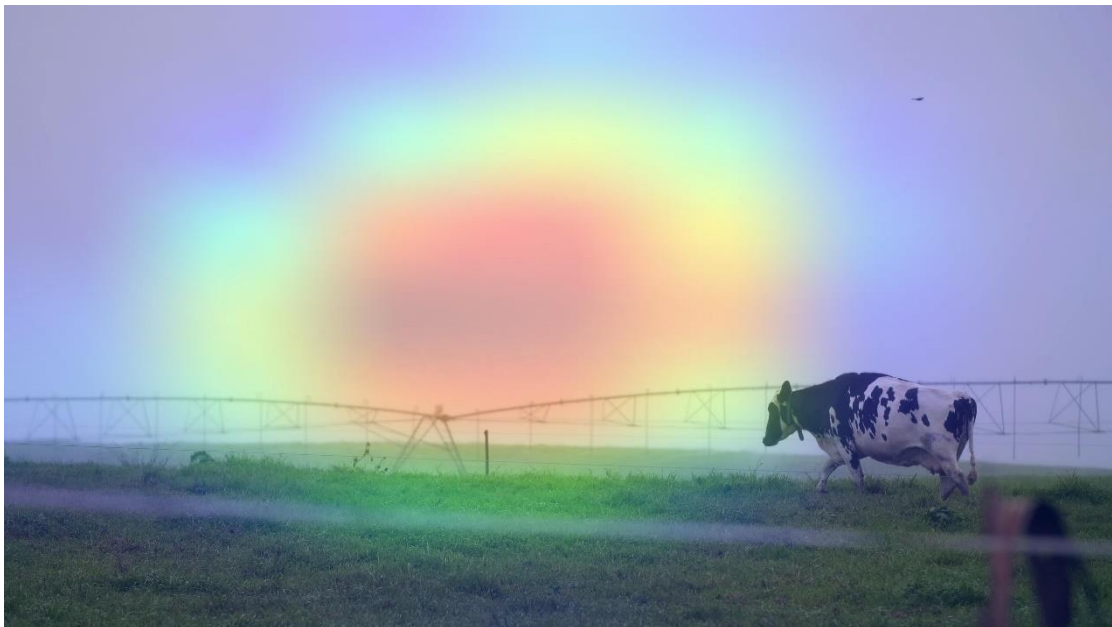
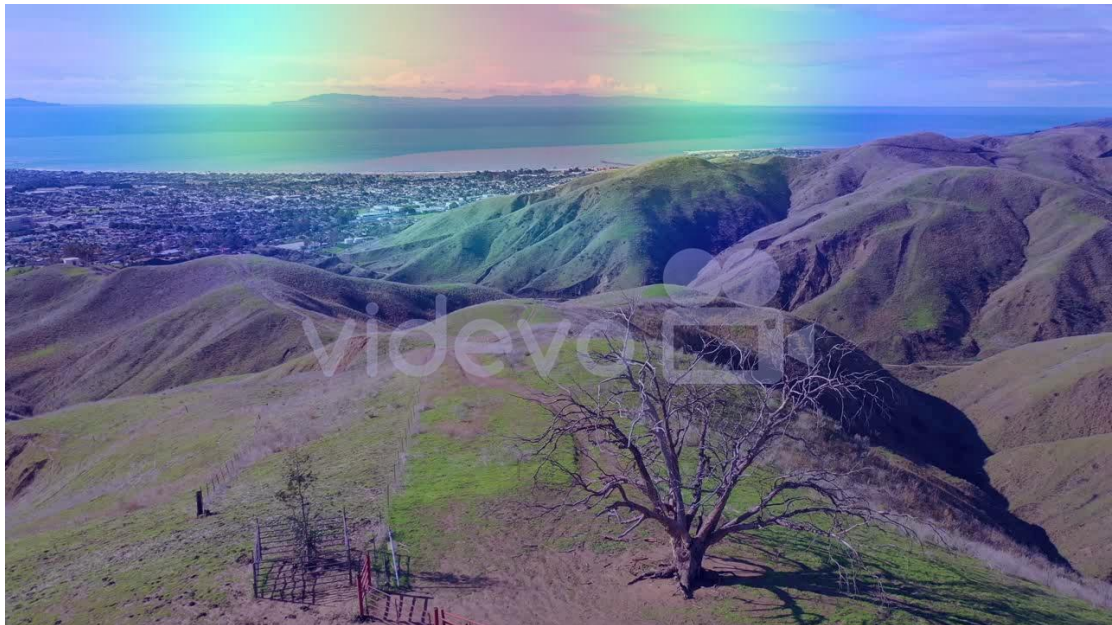
ก.3. รูปภาพเว็บแอปพลิเคชันขั้นตอนแสดงผลการทำนาย

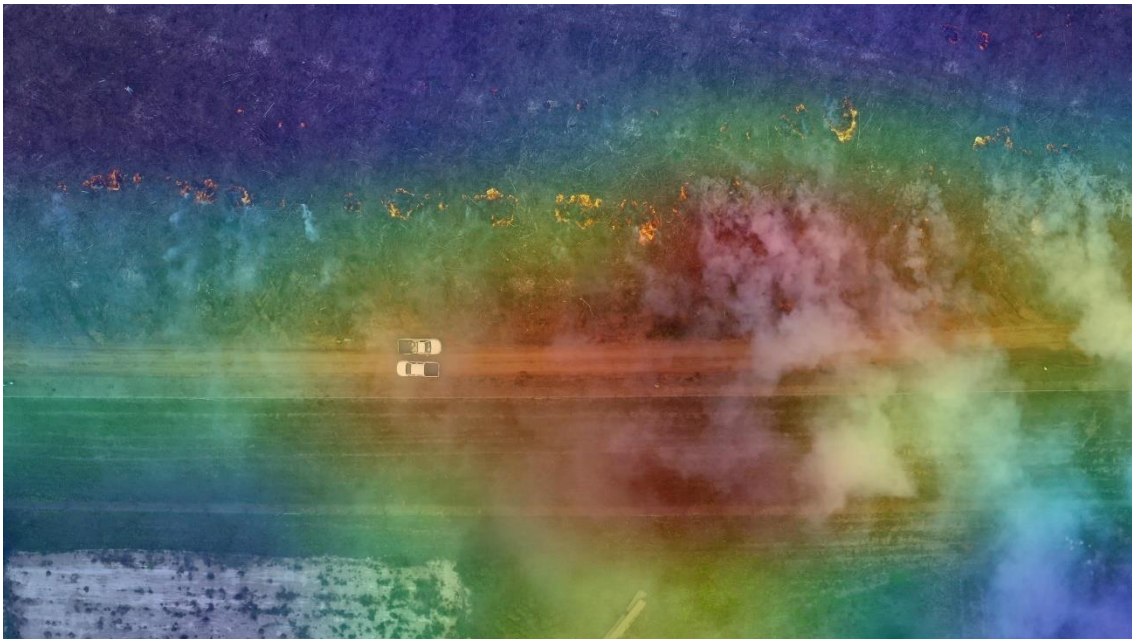


ก.4. รูปภาพเว็บแอปพลิเคชันตอนเลื่อนลงเพื่อดูรูป grad cam



ก.5. รูป grad cam บางส่วนจากวิดีโอต่าง ๆ





ประวัติผู้เขียน



นายรัชพงศ์ ทอหุล เกิดเมื่อวันที่ 17 กุมภาพันธ์ 2542 ณ จังหวัดเชียงใหม่สำเร็จการศึกษาระดับมัธยมจากโรงเรียนจอมทอง เข้าศึกษาที่ภาควิชาวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยเชียงใหม่ เมื่อ สิงหาคม 2560 โดยมีความสนใจในด้าน การเขียนโปรแกรม โดยมีความตั้งใจจะไปทำงานในด้าน data science และ web development