

Your Art Painter
(จิตรกรผู้สร้างศิลปะของคุณ)

Mr. Nawarit Longkhum
Ms. Pataraphorn Tanutsiriteeradet
Ms. Sukasama Chitakson

A Project Submitted in Partial Fulfillment
of the Requirements for
the Degree of Bachelor of Engineering (Computer Engineering)
Faculty of Engineering
King Mongkut's University of Technology Thonburi
2020

Project Committee

..... (Assoc.Prof. My main advisor name , Ph.D.)	Project Advisor
..... (Assoc.Prof. My Co-advisor name, Ph.D.)	Project Co-Advisor
..... (Asst.Prof. Committee2, Ph.D.)	Committee Member
..... (Asst.Prof. Committee3, Ph.D.)	Committee Member

Project Title	Your Art Painter (จิตรกรผู้สร้างศิลปะของคุณ)
Credits	3
Member(s)	Mr. Nawarit Longkhum Ms. Pataraphorn Tanutsiriteeradet Ms. Sukasama Chitakson
Project Advisor	Assoc.Prof. My main advisor name , Ph.D.
Co-advisor	Assoc.Prof. My Co-advisor name, Ph.D.
Program	Bachelor of Engineering
Field of Study	Computer Engineering
Department	Computer Engineering
Faculty	Engineering
Academic Year	2020

Abstract

In a multihop ad hoc network, the interference among nodes is reduced to maximize the throughput by using a smallest transmission range that still preserve the network connectivity. However, most existing works on transmission range control focus on the connectivity but lack of results on the throughput performance. This paper analyzes the per-node saturated throughput of an IEEE 802.11b multihop ad hoc network with a uniform transmission range. Compared to simulation, our model can accurately predict the per-node throughput. The results show that the maximum achievable per-node throughput can be as low as 11% of the channel capacity in a normal set of α operating parameters independent of node density. However, if the network connectivity is considered, the obtainable throughput will reduce by as many as 43% of the maximum throughput.

Keywords: Multihop ad hoc networks / Topology control / Single-Hop Throughput

หัวข้อปริญญานิพนธ์	หัวข้อปริญญานิพนธ์บรรทัดแรก หัวข้อปริญญานิพนธ์บรรทัดสอง
หน่วยกิต	3
ผู้เขียน	นายสมศักดิ์ คอมพิวเตอร์ นางสาวสมศรี คอมพิวเตอร์2 นางสาวสมปอง คอมพิวเตอร์3
อาจารย์ที่ปรึกษา	รศ.ดร.ที่ปรึกษา วิทยานิพนธ์ รศ.ดร.ที่ปรึกษา วิทยานิพนธ์ร่วม
หลักสูตร	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมคอมพิวเตอร์
คณะ	วิศวกรรมศาสตร์
ปีการศึกษา	2563

บทคัดย่อ

การวิจัยครั้งนี้มีวัตถุประสงค์ เพื่อศึกษาความพึงพอใจในการให้บริการงานทั่วไปของสำนักวิชา พื้นฐานและภาษา เพื่อเปรียบเทียบระดับความพึงพอใจต่อการให้บริการงานทั่วไปของสำนักวิชาพื้นฐานและภาษา ของนักศึกษาที่มาใช้บริการสำนักวิชาพื้นฐานและภาษา สถาบัน เทคโนโลยีไทย-ญี่ปุ่น จาแนกตามเพศ คณะ และชั้นปีที่ศึกษา เพื่อศึกษาปัญหาและข้อเสนอแนะของ นักศึกษามาเป็นแนวทางในการพัฒนาและปรับปรุงการให้บริการของสำนักวิชาพื้นฐานและภาษา

คำสำคัญ: การชูปเคลือบด้วยไฟฟ้า / การชูปเคลือบผิวเหล็ก / เคลือบผิวรังสี

กิตติกรรมประกาศ

ขอบคุณอาจารย์ที่ปรึกษา กรรมการ พ่อแม่พี่น้อง และเพื่อนๆ คนที่ช่วยให้งานสำเร็จ ตามต้องการ

สารบัญ

หน้า

ABSTRACT	ii
บทคัดย่อ	iii
กิตติกรรมประกาศ	iv
สารบัญ	vi
สารบัญตาราง	vii
สารบัญรูปภาพ	viii
สารบัญสัญลักษณ์	ix
สารบัญคำศัพท์ทางเทคนิคและคำย่อ	x
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญ	1
1.2 ประเภทของโครงงาน	1
1.3 วิธีการนำเสนอ	1
1.4 วัตถุประสงค์ของโครงการ	2
1.5 ขอบเขตของโครงงาน	2
1.6 เนื้อหาทางวิศวกรรมที่เป็นต้นฉบับ	2
1.7 การแยกย่อยงาน และวางแผนการดำเนินงาน	2
1.8 ผลการดำเนินการ	5
1.8.1 ผลการดำเนินการในภาคการศึกษาที่1	5
1.8.2 ผลการดำเนินการในภาคการศึกษาที่2	6
บทที่ 2 ทฤษฎีความรู้และงานที่เกี่ยวข้อง	7
2.1 บทนำ	7
2.2 แนวความคิดทางทฤษฎี	7
2.2.1 Neural style transfer	7
2.2.2 Overview of machine learning	7
2.2.3 Classical artificial neural networks	8
2.2.3.1 Neuron unit ใน hidden layer	8
2.2.3.2 Backward propagation	9
2.2.3.3 Cost function	9
2.2.4 Convolutional Neural Network (CNN)	9
2.2.4.1 Input data	9
2.2.4.2 Convolution layer	10
2.2.4.3 Pooling layer	10
2.2.4.4 Fully connected layer	10
2.2.5 Transfer Learning	10
2.2.5.1 VGGNet	11
2.2.6 Web Application	12
2.2.7 Development Tools	13
2.2.7.1 Python	13
2.2.7.2 Hypertext Markup Language (HTML)	13
2.2.7.3 Cascading Style Sheets (CSS)	13
2.2.7.4 JavaScript	13
2.2.7.5 Pytorch	13
2.2.7.6 Django	13

บทที่ 3	วิธีการดำเนินงาน	15
3.1	ข้อกำหนดและความต้องการของระบบ	15
3.2	สถาปัตยกรรมระบบ	15
3.3	Hardware Module 1	15
3.3.1	Component 1	15
3.3.2	Logical Circuit Diagram	15
3.4	Hardware Module 2	15
3.4.1	Component 1	15
3.4.2	Component 2	15
3.5	Path Finding Algorithm	15
3.6	Database Design	15
3.7	UML Design	15
3.8	GUI Design	15
3.9	การออกแบบการทดลอง	15
3.9.1	ตัวชี้วัดและปัจจัยที่ศึกษา	15
3.9.2	รูปแบบการเก็บข้อมูล	15
บทที่ 4	ผลการดำเนินงาน	16
4.1	ประสิทธิภาพการทำงานของระบบ	16
4.2	ความพึงพอใจการใช้งาน	16
4.3	การวิเคราะห์ข้อมูลและผลการทดลอง	16
บทที่ 5	บทสรุป	17
5.1	สรุปผลโครงการ	17
5.2	ปัญหาที่พบและการแก้ไข	17
5.3	ข้อจำกัดและข้อเสนอแนะ	17
บรรณานุกรม		18
APPENDIX		19
A	ชื่อภาคผนวกที่ 1	20
B	ชื่อภาคผนวกที่ 2	22

สารบัญตาราง

ตารางที่	หน้า
1.1 ตารางแผนการทำงานภาคการศึกษาที่ 1	4
1.2 ตารางแผนการทำงานภาคการศึกษาที่ 2	5
3.1 test table x1	15

สารบัญรูปภาพ

รูปที่		หน้า
2.1	แสดงหลักการทำงานของ Neural Style Transfer ที่ใช้ Convolutional Neural Network (CNN) [2]	7
2.2	แสดงส่วนประกอบหลักของ Artificial neural networks	8
2.3	แสดงการทำงานภายใน neuron unit [?]	8
2.4	แสดงการทำงานของ feed forward (forwardpass) กับ back propagation (backwardpass)	9
2.5	แสดงให้เห็นถึงกระบวนการทำงานของ CNN ที่มีการเรียนรู้ 2 ส่วนคือ feature learning ซึ่งมี convolution layer และ pooling layer ประกอบกัน และ classification ซึ่งมี fully connected layer มาเกี่ยวข้อง	9
2.6	แสดงให้เห็นถึงลักษณะของ input data ที่มีการแบ่งชั้นสีเป็น 3 channels ซึ่งจากรูปนี้จะมีขนาดของรูปเป็น 4*4*3 RGB Image	10
2.7	แสดงให้เห็นถึงการเปรียบเทียบการทำงานระหว่าง traditional ML กับ transfer learning ที่มีรูปแบบการทำงานที่ต่างกัน [1]	11
2.8	แสดงสถาปัตยกรรมของ VGG16 และ VGG19	12
2.9	แสดงหลักการทำงานของ web application	13

สารบัญสัญลักษณ์

SYMBOL

α	Test variable
λ	Interarival rate
μ	Service rate

UNIT

m^2
jobs/ second
jobs/ second

สารบัญคำศัพท์ทางเทคนิคและคำย่อ

ABC	=	Adaptive Bandwidth Control
MANET	=	Mobile Ad Hoc Network

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญ

กว่า 5,000 ปีมาแล้ว ตั้งแต่ยุคก่อนประวัติศาสตร์ ที่มนุษย์รู้จักการวาดรูป โดยเริ่มแรกมนุษย์วาดภาพครั้งแรกโดยการขีดขูดบนผนังถ้ำ หรือเพิงผา โดยรูปภาพที่วาดออกมาจะเป็นภาพคน สัตว์ หรือการล่าสัตว์ ซึ่งแสดงให้เห็นถึงชีวิตความเป็นอยู่ของคนในยุคสมัยนั้น และหลังจากนั้นมนุษย์ได้มีการพัฒนาการวาดรูปอยู่ตลอดเวลา จากการวาดรูปตามผนังถ้ำ ตามผนังโบสถ์ หรือราชวังที่สำคัญต่างๆ จนมาเป็นวาดรูปตามผืนผ้า ตลอดจนมาเป็นในกระดาษ ภาพวาดนั้นนอกจากจะวาดเพื่อถ่ายทอดเรื่องราวในอดีตแล้ว ยังแสดงให้เห็นถึงเอกลักษณ์ของศิลปินอีกด้วย ซึ่งศิลปินของโลกทางตะวันตกและตะวันออกก็มีลักษณะที่แตกต่างกัน และในแต่ละประเทศหรือศิลปินแต่ละท่านก็จะมีเทคนิควิธีการวาดที่ไม่เหมือนกัน ศิลปินไทยที่มีชื่อในการวาดภาพ เช่น อ.เฉลิมชัย โฆษิตพิพัฒน์ อ.ชอุ่ม นิ่มเสมอ และอ.จักรพันธ์ โปษยกฤต เป็นต้น style ในการวาดของศิลปินแต่ละท่านก็มีความแตกต่างกัน ทำให้ผู้คนชื่นชอบงานของท่าน และอยากมีภาพที่มีสไตล์แบบนั้น

จนกระทั่งปี 2015 Gatys et al. ได้ตีพิมพ์ผลงาน A Neural Algorithm of Artistic Style ซึ่งเป็นการศึกษาเกี่ยวกับการนำเอาภาพ style ที่มีชื่อเสียงไปรวมกับภาพวิวเมือง Tubingen, ประเทศเยอรมัน โดยใช้แนวคิดของ Convolutional Neural Network (CNN) ได้สำเร็จ ออกมาเป็นรูปภาพที่เป็นรูปภาพวิวเมือง Tubingen แต่ได้ใช้เทคนิคการวาดรูปจากศิลปินชื่อดัง ทำให้ภาพที่ได้มาใหม่นั้นดูเหมือนศิลปินที่เสียชีวิตไปแล้วกลับมาวาดรูปขึ้นใหม่ จากความสำเร็จครั้งนั้น ส่งผลให้มีคนจำนวนมากสนใจในการทำ Neural Style Transfer (NST) และ CNN มากขึ้น

ดังนั้นทางคณะผู้จัดทำจึงต้องการศึกษาวิธีการทำ Neural Style Transfer ที่มี Convolutional Neural Network (CNN) เป็นทฤษฎีเบื้องหลัง และนำความรู้ที่ได้มาพัฒนาโมเดลที่สามารถทำ Neural Style Transfer รูปที่วาดโดยอ.จักรพันธ์ โปษยกฤตและอ.ชอุ่ม นิ่มเสมอ ซึ่งเป็นศิลปินไทยได้ อีกทั้งเพื่อเพื่อพัฒนา web application ที่มาสามารถสร้างรูปภาพจากที่มีลักษณะของศิลปินผสมผสานได้

1.2 ประเภทของโครงการ

นำเสนอความต้องการของผู้มีส่วนได้ส่วนเสียเฉพาะกลุ่ม ผลิตภัณฑ์ทางการค้าที่มีศักยภาพ และ เว็บแอปพลิเคชันสำหรับคอมพิวเตอร์

1.3 วิธีการนำเสนอ

ในการบรรลุวัตถุประสงค์ของโครงการนั้น ใช้วิธีการดังต่อไปนี้

1. เตรียมชุดข้อมูลเพื่อนำไปใช้ train โมเดล

รวบรวมรูปภาพที่สร้างจากศิลปินที่มีชื่อเสียงและลักษณะภาพสามารถบอกได้ถึงเอกลักษณ์ของศิลปินท่านนั้นได้ โดยทางผู้จัดทำได้รวบรวมรูปภาพของศิลปิน จำนวน 2 ท่านได้แก่ อ.จักรพันธ์ โปษยกฤต และ อ. ชอุ่ม นิ่มเสมอ เพื่อใช้เป็น style image และรวบรวมรูปภาพคนหรือสัตว์ทั่ว ๆ ไป เพื่อใช้เป็น content image

2. จัดการรูปภาพให้อยู่ในรูปแบบที่ใช้ในโมเดล (Data preprocessing)

จัดการรูปภาพทั้ง style image และ content image ในเหมาะสมโดยทำการเพิ่มความละเอียด เพิ่มความคมชัด เป็นต้น และแปลงรูปภาพให้อยู่ในลักษณะที่สามารถใช้กับโมเดลได้ คือ ขนาด 224*224 พิกเซล

3. สร้างโมเดลและปรับค่า parameter

นำโมเดลของ VGGNet ที่มีอัลกอริทึม convolutional neural network (CNN) อยู่เบื้องหลังมาใช้ในการค้นหารูปแบบและลักษณะเด่นของภาพออกมา (Feature Extraction) จากนั้นเอาลักษณะเด่นของ style image และ content image มารวมกัน เพื่อสร้างรูปภาพใหม่ที่เป็นแบบ content image แต่มีลักษณะของ style image ผสมผสานเข้าด้วยกัน โดยปรับค่า weight และ bias จนได้โมเดลที่มีค่า loss function เหมาะสม

4. นำรูปภาพที่สร้างจากศิลปินท่านอื่นหรือ style image รูปอื่น ๆ มา train ร่วมกับ content image เดิม และปรับค่าต่าง ๆ ให้โมเดลสามารถทำงานกับรูปอื่นได้ทั่วไป

5. ทดลองใช้งานโมเดล โดยนำรูปภาพคนหรือสัตว์ รูปอื่น ๆ มาใช้งานโมเดลแล้วแก้ไขข้อผิดพลาดและพัฒนาให้โมเดลดียิ่งขึ้นจากเดิม

6. สร้าง Web Application ตามที่ได้ออกแบบไว้

1.4 วัตถุประสงค์ของโครงการ

1. เพื่อศึกษาการทำ Neural Style Transfer ที่มี Convolutional Neural Network (CNN) เป็นทฤษฎีเบื้องหลัง
2. เพื่อพัฒนาโมเดลที่สามารถทำ Neural Style Transfer รูปที่วาดโดยศิลปินไทยได้
3. เพื่อพัฒนา web application ที่สามารถสร้างรูปภาพจากที่มีลักษณะของศิลปินผสมผสานได้

1.5 ขอบเขตของโครงการ

1. สร้าง Model ที่สามารถใช้งานกับรูปภาพของศิลปินไทยได้ โดยใช้ CNN
2. สร้างรูปภาพใหม่ที่เกิดจากการผสมผสานระหว่างรูปภาพที่สร้างโดยศิลปินไทยกับภาพถ่ายคนหรือภาพสัตว์ โดยใช้ภาษา python
3. สร้าง web application ด้วย Django framework ได้
4. ผู้ใช้สามารถเลือกแบบรูปภาพของแต่ละศิลปิน โดยศิลปินที่จะเอามาเป็น style image ได้แก่
 - อ.จักรพันธ์ โปษยกฤษ
 - อ.ชลุต นิมเสมอ

1.6 เนื้อหาทางวิศวกรรมที่เป็นต้นฉบับ

ในส่วนที่คณะผู้จัดทำจะทำการรวมรูปภาพที่สร้างโดยศิลปินหรือ Style image เข้ากับรูปภาพคนหรือสัตว์หรือ content image เข้าด้วยกันเป็นภาพใหม่ที่มีรูปแบบ content image แต่มีการวาดแบบ style image ผสมเข้าด้วยกัน โดยได้เลือกรูปภาพของศิลปินที่สนใจมา 2 ท่านได้แก่ อ.จักรพันธ์ โปษยกฤษ และ อ. ชลุต นิมเสมอ มาใช้เป็น style image และเลือกใช้ภาษา python ในการพัฒนาโมเดลซึ่งมี library และโมเดลที่หลายหลายในเลือกใช้ จึงเลือกใช้ VGGNet ที่มีอัลกอริทึม Convolutional neural network (CNN) อยู่เบื้องหลังมาใช้งาน และได้มีการพัฒนาวิธีการปรับค่า weight และ bias ของโมเดล เพื่อให้ได้โมเดลที่มีค่า loss function ที่เหมาะสมและเหมาะกับรูปภาพที่ทางคณะผู้จัดทำนำมา train

ในส่วนของการใช้งานหลังจากสร้างโมเดลเรียบร้อยแล้ว ทางคณะผู้จัดทำได้ทำการพัฒนา Web application โดยใช้ภาษา Html, CSS และ JavaScript ในส่วนติดต่อกับผู้ใช้ frontend และใช้ Django ที่เป็น framework library ชนิดหนึ่งในภาษา Python ในส่วน backend

1.7 การแยกย่อยงาน และวางแผนการดำเนินงาน

1. กำหนดหัวข้อโครงการที่ต้องการทำ
 - 1 หาข้อมูลเกี่ยวกับสิ่งที่สนใจจะทำ
 - 2 ปรึกษาสมาชิกในกลุ่มและกำหนดหัวข้อโครงการ
 - 3 นำเสนอและขอคำแนะนำจากอาจารย์ที่ปรึกษา
2. ศึกษาและรวบรวมข้อมูลเกี่ยวกับวิธีการและเทคโนโลยีที่ใช้ในโครงการ
 - 1 รวบรวมข้อมูลเบื้องต้นเกี่ยวกับโครงการ
 - 2 ศึกษาวิธีการทำงานและงานวิจัยที่เกี่ยวข้อง
3. ประเมินความเป็นไปได้และกำหนดขอบเขตของโครงการ
4. จัดทำข้อเสนอโครงการและวางแผนดำเนินการ
 - 1 เขียน proposal report
 - 2 วางแผนเวลาในการทำงานพร้อมแบ่งงานย่อยและมอบหมายให้กับสมาชิกในกลุ่ม

5. ออกแบบโครงสร้างและส่วนประกอบของโครงการ

- 1 ออกแบบ flow ของการทำงาน
- 2 ศึกษาโมเดล อัลกอริทึมและ library ที่เกี่ยวข้อง
- 3 ออกแบบ User Interface
- 4 ออกแบบ Database
- 5 ออกแบบโมเดลที่ใช้ทำงาน
- 6 ศึกษาการ Optimization และการประเมินผลประสิทธิภาพของโมเดล

6. รวบรวมข้อมูลเพื่อใช้เป็นในการ train โมเดล

- 1 รวบรวมรูปภาพที่สร้างจากศิลปินเพื่อเป็น style image
- 2 รวบรวมรูปภาพคนหรือสัตว์เพื่อเป็น content image

7. จัดการรูปภาพให้อยู่ในรูปแบบที่ใช้ในโมเดล (Data preprocessing)

8. ทดลองสร้างโมเดลและปรับแต่งให้มีประสิทธิภาพดีที่สุด

- 1 ทดลองสร้างโมเดลโดยใช้ parameter พื้นฐาน
- 2 ปรับแต่งโมเดลด้วยการปรับค่าของ parameter ต่าง ๆ และวัดผลที่ได้ออกมา
- 3 เลือกโมเดลที่ได้ผลลัพธ์ที่ดีที่สุด

9. ทำให้โมเดลให้สามารถใช้งานได้ทั่วไปมากขึ้นโดยนำรูปภาพจากศิลปินอื่น ๆ มา train โมเดลเพิ่มเติม

10. ทดลองใช้งานโมเดลโดยนำรูปภาพคนหรือสัตว์ มา test โมเดลเพิ่มเติม

11. ประเมินผลโมเดลที่ได้สร้างขึ้นมา

12. พัฒนา Web Application ตามที่ได้ออกแบบไว้

- 1 Frontend
- 2 Backend

13. นำเสนอโครงการ

14. จัดทำรายงานแสดงความคืบหน้า

15. นำเสนอรายงานประจำการศึกษา

จากขั้นตอนการดำเนินงานข้างต้น สามารถแบ่งออกเป็น 2 ภาคการศึกษา โดยภาคการศึกษาที่ 1 นั้นมุ่งเน้นไปที่การศึกษาหาข้อมูลและทดลองพัฒนาซอฟต์แวร์ตัวต้นแบบ ซึ่งสามารถแสดงรายการและระยะเวลาออกมาได้เป็นแผนภูมิแกนต์ดังตามตารางที่ 1.1 และในส่วนภาคการศึกษาที่ 2 นั้นมุ่งเน้นไปที่การพัฒนาต่อยอดซอฟต์แวร์ตัวต้นแบบให้สามารถทำงานได้อย่างถูกต้อง ครบทุกฟังก์ชันและแก้ไขปัญหาเมื่อซอฟต์แวร์ได้ทำงานในสภาพแวดล้อมจริง ตลอดจนถึงขั้นตอนสุดท้ายสามารถเขียนเป็นแผนภูมิแกนต์ ดังตารางที่ 1.2

Test/Week	Aug			Sep				Oct				Nov				Dec		ผู้รับผิดชอบ
	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	
1. กำหนดหัวข้อโครงการที่ต้องการทำ																		
1.1 หาข้อมูลเกี่ยวกับสิ่งที่สนใจจะทำ	■	■																ทุกคน
1.2 ปรึกษามาคิกในกลุ่มและกำหนดหัวข้อโครงการ	■	■	■															ทุกคน
1.3 นำเสนอและขอคำแนะนำจากอาจารย์ที่ปรึกษา		■	■	■														ทุกคน
2. ศึกษาและรวบรวมข้อมูลเกี่ยวกับวิธีการและเทคโนโลยีที่ใช้ในโครงการ																		
2.1 รวบรวมข้อมูลเบื้องต้นเกี่ยวกับโครงการ		■	■	■														นฤฤทธิ์
2.2 ศึกษาวิธีการทำงานและงานวิจัยที่เกี่ยวข้อง			■	■	■													ทุกคน
3. ประเมินความเป็นไปได้และกำหนดขอบเขตของโครงการ			■	■	■													ภัทรพร
4. จัดทำข้อเสนอโครงการและร่างแผนดำเนินการ																		
4.1 เขียน proposal report			■	■	■	■												ทุกคน
4.2 วางแผนเวลาในการทำงานพร้อมแบ่งงานย่อยและมอบหมายให้กับสมาชิกในกลุ่ม				■	■	■												ภัทรพร
5. ออกแบบโครงสร้างและส่วนประกอบของโครงการ																		
5.1 ออกแบบ flow ของการทำงาน					■	■	■											สุกษมา
5.2 ศึกษาโมเดล อัลกอริทึมและ Library ที่เกี่ยวข้อง						■	■	■	■									ภัทรพร นวฤทธิ์
5.3 ออกแบบ User Interface						■	■	■	■	■								ภัทรพร
5.4 ออกแบบ Database						■	■	■	■									สุกษมา
5.5 ออกแบบโมเดลที่ใช้ทำงาน								■	■	■	■	■						ภัทรพร นวฤทธิ์
5.5 ศึกษาการ Optimization และการประเมินผลประสิทธิภาพของโมเดล										■	■	■	■	■				ภัทรพร
6. รวบรวมข้อมูลเพื่อใช้ในการ train โมเดล																		
6.1 รวบรวมรูปภาพที่สร้างจากศิลปินเพื่อเป็น style image												■	■	■	■			ภัทรพร นวฤทธิ์
6.2 รวบรวมรูปภาพคนหรือสัตว์เพื่อเป็น content image												■	■	■	■			ภัทรพร นวฤทธิ์
13. นำเสนอโครงการ															■	■	■	ทุกคน
14. จัดทำรายงานแสดงความคืบหน้า											■	■	■	■				ทุกคน
15. นำเสนอรายงานประจำการศึกษา															■	■	■	ทุกคน

ตารางที่ 1.1 ตารางแผนการทำงานภาคการศึกษาที่ 1

Test/Week	Dec		Jan				Feb				Mar				Apr				ผู้รับผิดชอบ
	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	
7. จัดการรูปภาพให้อยู่ในรูปแบบที่ใช้ในโมเดล (Data preprocessing)																			ภัทรพร นวฤทธิ์
8. ทดลองสร้างโมเดลและปรับแต่งให้มีประสิทธิภาพที่สุด																			
8.1 ทดลองสร้างโมเดลโดยใช้ parameter พื้นฐาน																			ภัทรพร นวฤทธิ์
8.2 ปรับแต่งโมเดลด้วยการปรับค่าของ parameter ต่าง ๆ และวัดผลที่ได้ออกมา																			ภัทรพร
8.3 เลือกโมเดลที่ได้ผลลัพธ์ที่ดีที่สุด																			ภัทรพร
9. ทำให้โมเดลสามารถใช้งานได้ทั่วไปมากขึ้นโดยนำรูปภาพจากศิลปินอื่นๆมา train โมเดลเพิ่มเติม																			ภัทรพร นวฤทธิ์
10. ทดลองใช้งานโมเดลโดยนำรูปภาพคนหรือสัตว์ มา test โมเดลเพิ่มเติม																			ภัทรพร นวฤทธิ์
11. ประเมินผลโมเดลที่ได้สร้างขึ้นมา																			ทุกคน
12. พัฒนา Web Application ตามที่ได้ออกแบบไว้																			
12.1 Frontend																			ภัทรพร นวฤทธิ์
12.2 Backend																			สุกษมา
13. นำเสนอโครงการ																			ทุกคน
14. จัดทำรายงานแสดงความคืบหน้า																			ทุกคน
15. นำเสนอรายงานประจำการศึกษา																			ทุกคน

ตารางที่ 1.2 ตารางแผนการทำงานภาคการศึกษาที่ 2

1.8 ผลการดำเนินการ

1.8.1 ผลการดำเนินการในภาคการศึกษาที่1

1. ศึกษาและรวบรวมข้อมูลเกี่ยวกับวิธีการและเทคโนโลยีที่ใช้ในโครงการ
2. ประเมินความเป็นไปได้และกำหนดขอบเขตของโครงการ
3. จัดทำข้อเสนอโครงการและวางแผนดำเนินการ
4. ออกแบบโครงสร้างและส่วนประกอบของโครงการ
5. รวบรวมข้อมูลเพื่อใช้ในการ train โมเดล (style image และ content image)
6. จัดการรูปภาพให้อยู่ในรูปแบบที่ใช้ในโมเดล (Data preprocessing)
7. สร้างโมเดลโดยใช้ parameter พื้นฐาน
8. จัดทำรายงานแสดงความคืบหน้า

1.8.2 ผลการดำเนินการในภาคการศึกษาที่2

1. ทดลองสร้างโมเดลและปรับแต่งให้มีประสิทธิภาพดีที่สุด
2. ทำโมเดลให้สามารถใช้งานได้ทั่วไปมากขึ้น
3. ทดลองใช้งานโมเดล
4. จัดทำ Web application ตามที่ออกแบบไว้
5. นำเสนอโครงการ
6. จัดทำรายงานแสดงความคืบหน้า
7. นำเสนอรายงานประจำการศึกษา

บทที่ 2 ทฤษฎีความรู้และงานที่เกี่ยวข้อง

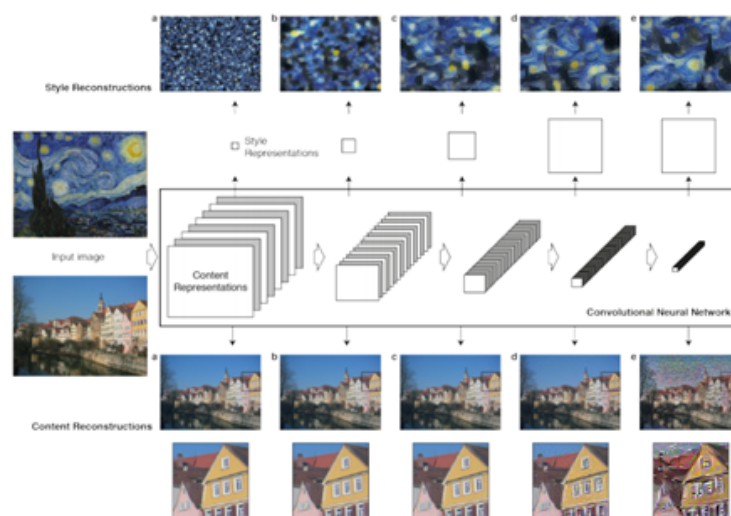
2.1 บทนำ

ในบทนี้ อธิบายเกี่ยวกับแนวคิดที่นำมาประยุกต์ใช้ในโครงงาน อัลกอริทึมที่ใช้ในการทำโครงงานคือ VGGNet และมีการพัฒนาโดยใช้ภาษา python นอกจากนี้มีการทบทวนวรรณกรรม ที่จะกล่าวถึงโครงงานที่มีลักษณะเนื้อหาที่เกี่ยวข้องกับการทำโครงงานนี้

2.2 แนวความคิดทางทฤษฎี

2.2.1 Neural style transfer

Neural Style Transfer [4] เป็นกระบวนการที่ใช้ CNNs ในการประมวลผลรูปภาพให้ได้รูปภาพที่เหมือนกับรูปภาพที่ใช้เป็น content image แต่มีรูปแบบหรือลักษณะที่ต่างกันไปตามรูปภาพที่ใช้เป็น style image ดังรูปที่ 2.1



รูปที่ 2.1 แสดงหลักการทำงานของ Neural Style Transfer ที่ใช้ Convolutional Neural Network (CNN) [2]

ดังรูปที่ 2.1 จะเห็นได้ว่า มีการป้อนข้อมูลเป็น content image และ style image จากนั้นจะทำการประมวลผลรูปที่ป้อนเข้ามา โดยจะมี filters ที่ reconstruct รูปภาพทั้ง content และ style โดยแสดงให้เห็นกระบวนการต่าง ๆ ใน CNN layer ต่าง ๆ ได้แก่ conv1-1(a), conv2-1(b), conv3-1(c), conv4-1(d) และ conv5-1(e) ซึ่งจะเห็นว่า ใน layer หลัง ๆ รูปภาพที่ได้รูปภาพที่ได้มีการลดทอนของรูปภาพที่เป็น content และมีการเพิ่มส่วนของรูปภาพที่เป็น style เข้าด้วยกัน จนได้รูปภาพที่เหมือนกับรูปภาพที่ใช้เป็น content image แต่มีรูปแบบหรือลักษณะที่ต่างกันไปตามรูปภาพที่ใช้เป็น style image

2.2.2 Overview of machine learning

Machine learning (ML) เป็นการทำให้คอมพิวเตอร์สามารถเรียนรู้ได้ด้วยตนเอง โดยเริ่มต้นจะต้องสอนให้คอมพิวเตอร์เข้าใจก่อน โดยมีรูปแบบการเรียนรู้ 3 แบบ คือ Supervised Learning เป็นการเรียนรู้ที่สอนให้คอมพิวเตอร์รู้จักทั้งข้อมูลที่เป็น input และ output ตัวอย่างเช่น การทำนายรูปภาพว่าเป็นหมาหรือแมว ซึ่งต้องสอนให้คอมพิวเตอร์รู้ว่ารูปภาพนี้เป็นหมาหรือแมวก่อนถึงจะเอาไปใช้ได้ เป็นต้น, Unsupervised Learning เป็นการเรียนรู้ที่สอนคอมพิวเตอร์เฉพาะข้อมูลที่เป็น input ไม่มี output ให้โดยจะให้คอมพิวเตอร์เรียนรู้ input ไปเรื่อย ๆ ตัวอย่างเช่น การจัดกลุ่มข้อมูล โดยจะป้อนเฉพาะข้อมูลเข้าไปแล้วคอมพิวเตอร์จะเรียนรู้เองว่า ข้อมูลแบบนี้ควรอยู่กับข้อมูลแบบไหน เป็นต้นและ Reinforcement learning ที่เป็นการเรียนรู้โดยการปฏิสัมพันธ์กับสิ่งแวดล้อมรอบ ๆ โดยถ้าทำถูกต้องจะได้คะแนนเพิ่ม ถ้าทำผิดคะแนนจะลดลง ตัวอย่างเช่น การเดินไปข้างหน้าของหุ่นยนต์ที่จะต้องรู้ก่อนว่า อยู่ตำแหน่งไหน แล้วในตำแหน่งควรเดินไปตำแหน่งไหนต่อไปถึงจะดี เป็นต้น

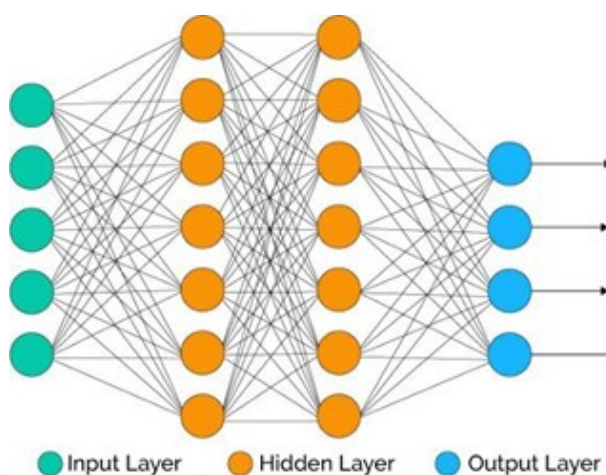
ในขั้นตอนการสร้างโมเดล machine learning จะมีข้อมูล input ที่เกี่ยวข้อง 2 ส่วนคือ training set เป็นชุดข้อมูลที่ใช้สำหรับการสอนโมเดลโดยจะฝึกฝนโมเดลให้ได้ผลลัพธ์ตามชุดข้อมูลที่ป้อนเข้ามา อีกส่วนคือ testing set เป็นชุดข้อมูลที่ใช้ทดสอบการทำงานของ

โมเดล โดยโมเดลที่ดีจะต้องไม่ทำให้เกิดการ overfitting หรือสามารถทำนายผลลัพธ์ใน training set ได้แม่นยำ แต่ทำนายผลลัพธ์ใน test-ing set ได้แย่

เมื่อสอนคอมพิวเตอร์เรียบร้อยแล้ว เมื่อมีข้อมูลใหม่ ๆ เข้า คอมพิวเตอร์ก็สามารถจัดการหรือทำนายข้อมูลนั้นได้เลย ไม่จำเป็นต้องเขียนโปรแกรมใหม่ทุกครั้งที่มีข้อมูลเข้ามา

2.2.3 Classical artificial neural networks

Artificial neural networks(ANN) คือโมเดลทางคณิตศาสตร์หรือโมเดลทางคอมพิวเตอร์ที่จำลองการทำงานของเซลล์ประสาทในสมองมนุษย์ที่มีเซลล์ประสาท(neurons) และจุดประสาทประสาท ดึงเกิดเป็น ANN ซึ่งมีส่วนประกอบในการทำงาน 3 ส่วนหลัก ๆ ดังรูปที่ 2.1 คือ input layer, hidden layer และ output layer ซึ่งการทำงานของ artificial neural networks จะมี 2 ส่วนคือ กระบวนการทำงานไปข้างหน้าหรือ feed forward คือป้อน input เข้าไป แล้วเข้า hidden layer ทำการประมวลผลแล้วได้ค่าออกมาส่งไปที่ output layer และ back propagation หรือกระบวนการทำย้อนกลับโดยจะเป็นการทำย้อนกลับจาก output layer ไป input layer

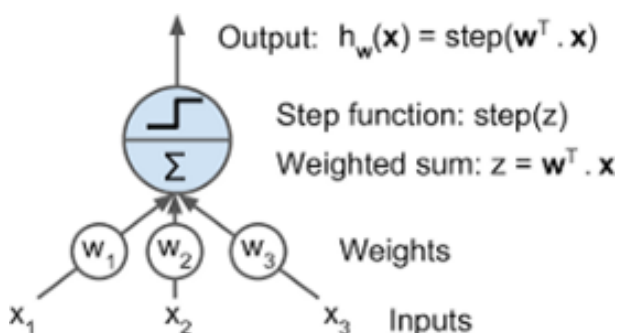


รูปที่ 2.2 แสดงส่วนประกอบหลักของ Artificial neural networks

ในแต่ละ layer จะมีก้อนกลม ๆ อยู่หรือก็คือ neuron unit อยู่ดังรูปที่ 2.2 ถ้าใน input layer ภายในจะเก็บข้อมูลที่รับเข้ามา แต่ถ้าเป็น hidden layer ภายในจะมีการคำนวณเกิดขึ้นโดยสามารถเพิ่ม hidden layer ได้หลาย layer ตามที่ผู้พัฒนาต้องการได้ ส่วนใน output layer จะมีการทำนายผลออกมาว่า ผลที่ได้เป็นคลาสอะไร

2.2.3.1 Neuron unit ใน hidden layer

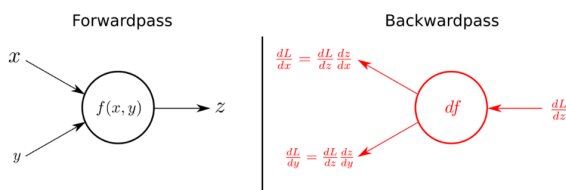
ภายใน neuron unit แต่ละตัวใน hidden layer จะมีการคำนวณโดยหลังจากผ่าน input layer เข้ามาจะมีการเพิ่ม weight ให้กับ input neuron unit แต่ละตัว จากนั้นเมื่อเข้า hidden layer ภายในจะรวมผลคูณระหว่าง weight กับ input neuron unit แต่ละตัวเข้าด้วยกันแล้วเข้า step function หรือส่วนใหญ่เรียกว่า activation function จากนั้นจะได้ output ออกมา ดังรูปที่ 2.3



รูปที่ 2.3 แสดงการทำงานภายใน neuron unit [?]

2.2.3.2 Backward propagation

Backward propagation เป็นกระบวนการทำงานย้อนกลับจาก output layer มา input layer เพื่อปรับค่า weight แต่ละตัวให้ดีขึ้น โดยจะทำการไปเรื่อง ๆ จนกว่าจะได้ cost function หรือ error ที่น้อยตามที่ต้องการ โดยลักษณะการทำงานของ feed forward กับ back propagation มีลักษณะต่างกันดังรูปที่ 2.4



รูปที่ 2.4 แสดงการทำงานของ feed forward (forwardpass) กับ back propagation (backwardpass)

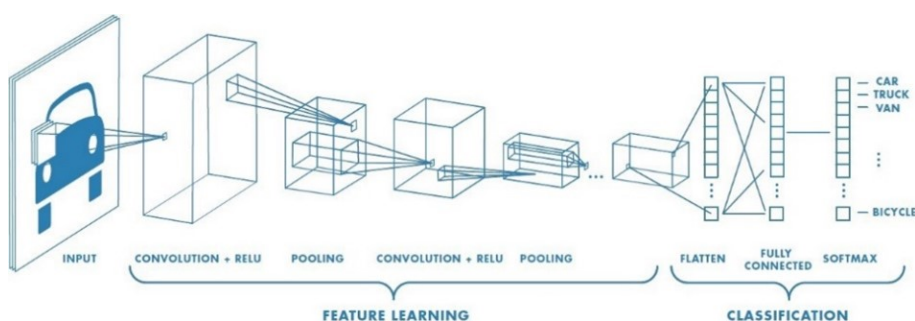
ในการทำ back propagation จะใช้อัลกอริทึม optimization เพื่อเพิ่มประสิทธิภาพตัวหนึ่งที่ชื่อว่า Gradient descent algorithm [13] ที่จะทำให้การหา weight ที่ต่ำที่สุดที่จะทำให้โมเดลสามารถทำนายผลได้ดีที่สุด เกิด error น้อย

2.2.3.3 Cost function

Cost function หรือ loss function หรือค่า error ที่เกิดขึ้น ซึ่งจุดมุ่งหมายของโมเดลต่าง ๆ จะเหมือนกันคือ ต้องการให้เกิดค่า error ที่น้อยที่สุดที่เป็นไปได้ ซึ่งการคำนวณค่า error ที่เกิดขึ้นจะใช้การเปรียบเทียบระหว่างผลลัพธ์จริง ๆ ที่ได้มาจากข้อมูลจริงกับผลลัพธ์ที่เกิดจากการทำนายโดยโมเดลมาเทียบกันโดยใช้วิธีต่าง ๆ ตัวอย่างเช่น การคำนวณค่า Mean Squared Error Loss, Mean Absolute Error Loss, Binary Crossentropy Loss, Categorical Crossentropy Loss, Sparse Categorical Crossentropy Loss เป็นต้น ซึ่งการคำนวณแบบต่าง ๆ ก็เหมาะกับปัญหาคนละแบบ ตัวอย่างเช่น Binary Crossentropy Loss ที่เหมาะกับข้อมูลที่มีเพียง 2 คลาส เป็นต้น

2.2.4 Convolutional Neural Network (CNN)

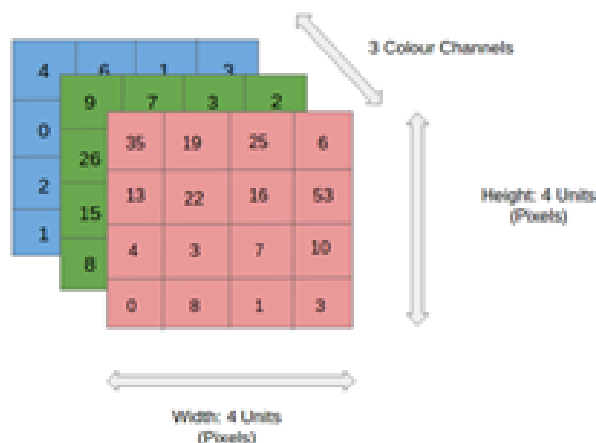
Convolutional Neural Network (CNN) หรือ ConvNet [2,11] เป็นอัลกอริทึม Deep Learning ชนิดหนึ่งที่ใช้ประมวลผลรูปภาพ โดยเรียนรู้คุณลักษณะต่าง ๆ ของรูปภาพที่ป้อนเข้าผ่านการใช้ filters ซึ่งเป็นการเรียนรู้ลักษณะของรูปภาพ เพื่อแยกแยะว่า รูปภาพที่ป้อนเข้ามาเป็นรูปอะไรดังรูปที่ 2.5



รูปที่ 2.5 แสดงให้เห็นถึงกระบวนการทำงานของ CNN ที่มีการเรียนรู้ 2 ส่วนคือ feature learning ซึ่งมี convolution layer และ pooling layer ประกอบกัน และ classification ซึ่งมี fully connected layer มาเกี่ยวข้อง

2.2.4.1 Input data

ข้อมูลที่เข้ามาประมวลใน CNN เป็นรูปภาพซึ่งมีหลากหลายรูปแบบที่เป็นขาวดำหรือ Grayscale, รูปภาพสีแบบ RGB, HSV หรือ CMYK เป็นต้น ซึ่งโดยส่วนใหญ่เป็นรูปภาพสีแบบ RGB ซึ่งเมื่อรับ input เข้ามาจะถูกแบ่งสืออกเป็น channels คือ Red(R) , Green(G) และ Blue(B) และแปลงขนาดความกว้างและความยาวของรูปภาพให้เป็น pixels ทำให้อยู่ในรูป heights*weights*จำนวนของ channels ดังรูปที่ 2.6



รูปที่ 2.6 แสดงให้เห็นถึงลักษณะของ input data ที่มีการแบ่งชั้นสีเป็น 3 channels ซึ่งจากรูปนี้จะมีขนาดของรูปเป็น 4*4*3 RGB Image

2.2.4.2 Convolution layer

ในส่วนของ convolutional layer จะมี filter หรือ Kernel (K) ซึ่งเป็น matrix ขนาดต่าง ๆ ใช้สำหรับ map กับข้อมูลที่เข้ามา และมี stride ที่ใช้สำหรับบอกว่าต้องขยับ filter ไปกี่ pixels โดย filters จะเคลื่อนที่ไปเรื่อย ๆ จนผ่านครบทุก pixels ในข้อมูลที่เข้ามา และนอกจากนี้จะมีการใช้ non-linear functions เพื่อ scale ข้อมูลที่เข้ามาไม่ให้ติดลบซึ่งเหมาะกับการเรียนรู้ใน CNNs โดยตัวที่นิยมใช้คือ Rectified Linear Unit (ReLU) ที่มีฟังก์ชันคือ $f(x) = \max(0, x)$

2.2.4.3 Pooling layer

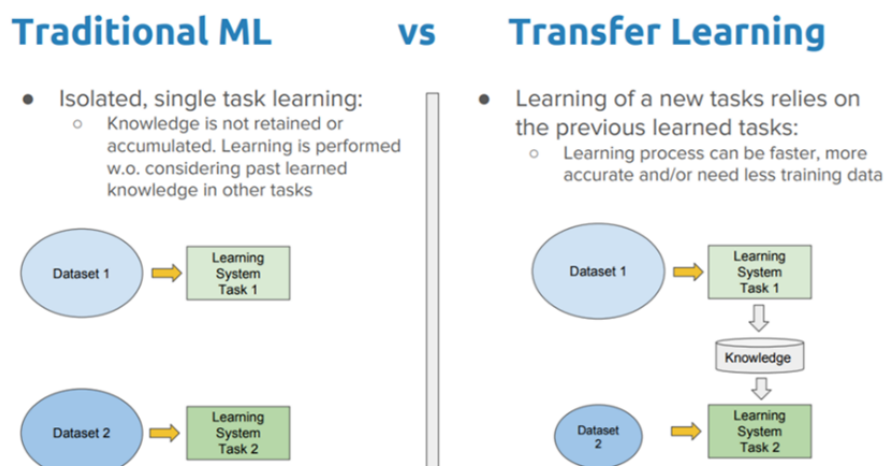
ในส่วนของ pooling layer จะทำงานต่อจาก convolution layer โดยจะทำการดึงข้อมูลออกมาหลังจากที่มีการทำ filter แต่ละครั้ง ออกมาเป็นข้อมูลที่ออกมา ณ ตำแหน่งนั้น ๆ ที่มีการทำ filter ทำให้สามารถดึงลักษณะสำคัญของข้อมูลที่ได้ ตัวอย่างเช่น ขอบ สี รูปทรง เป็นต้น และยังทำให้ขนาดข้อมูลที่ได้ออกมามีขนาดเล็กลงเมื่อเทียบกับข้อมูลที่เข้ามา โดย pooling มี 3 แบบ คือ min pooling, max pooling และ average pooling ซึ่ง convolution layer กับ pooling layer จะทำงานคู่กัน ซึ่งใน CNN จะมี layer นั้นขึ้นอยู่กับความซับซ้อนของรูปภาพ

2.2.4.4 Fully connected layer

เมื่อทำ feature learning โดยใช้ convolution layer กับ pooling layer แล้ว output ที่ได้จะอยู่ในรูปแบบที่เป็น 3 ชั้นหรือมี 3 มิติอยู่ จะมีการทำ flatten ก่อนเพื่อให้อยู่ในรูปแบบที่เป็น 1 มิติ เพื่อที่จะได้สามารถเข้าไปใน fully connected layer ได้ โดยใน fully connected layer มีการใช้ softmax classification ที่ใช้แยกว่าข้อมูล input ที่เข้ามาเป็นรูปอะไรโดยแสดงผลเป็นความเป็นไปได้ที่เป็นรูปต่าง ๆ

2.2.5 Transfer Learning

Transfer Learning เป็นการเรียนรู้ที่เป็นที่นิยมในการทำ machine learning (ML) โดยเป็นการนำโมเดลที่มีคนสร้างมาก่อนที่เหมาะสมกับงานหนึ่ง ๆ มาประยุกต์ใช้กับงานอื่นโดยไม่จำเป็นต้องสอนโมเดลใหม่ทั้งหมด ทำให้สามารถลดเวลาประมวลผลและเวลาในการสอนโมเดลได้ ดังรูปที่ 2.7



รูปที่ 2.7 แสดงให้เห็นถึงการเปรียบเทียบการทำงานระหว่าง traditional ML กับ transfer learning ที่มีรูปแบบการทำงานที่ต่างกัน [1]

โดยในการทำ transfer learning มีอยู่ 2 รูปแบบ คือ develop model ซึ่งเป็นการนำโมเดลนั้นมาแก้ไขและพัฒนาให้ดีกว่าเดิมให้เหมาะกับงานที่โมเดลนั้นสร้างมาใช้และจึงนำไปใช้กับงานอื่น ๆ ส่วนอีกรูปแบบคือ pre-trained model ซึ่งเป็นการนำโมเดลไปใช้กับงานอื่นเลย โดยส่วนใหญ่จะใช้ pre-trained model เป็นหลัก

สำหรับการทำ transfer learning ที่เกี่ยวกับรูปภาพ[6][5]มีการสร้างโมเดลสำหรับ ImageNet ที่มีรูปภาพกว่า 1.4 ล้านรูปและมากกว่า 1,000 คลาสที่เป็นความท้าทายสำหรับการทำ image classification ทำให้มีองค์การวิจัยต่าง ๆ มีการพัฒนาโมเดลขึ้นมาสำหรับแก้ปัญหา นี้ และได้เปิดให้คนทั่วไปสามารถเอาไปใช้ได้ ตัวอย่างเช่น โมเดล VGG ของ Oxford, โมเดล Inception ของ Google และโมเดล ResNet ของ Microsoft เป็นต้น

2.2.5.1 VGGNet

Visual Geometry Group (VGG) เป็น CNN สำหรับจดจำรูปภาพ ซึ่งสถาปัตยกรรมของ VGG จะประกอบไปด้วยบล็อก แต่ละบล็อกจะเป็น 2 มิติ คือ Convolution layer และ Pooling layer แนวความคิดนี้ถูกพัฒนาโดย Simonyan & Zisserman จาก Oxford Robotics Institute ซึ่ง VGG นั้นมีโมเดล 2 ตัวคือ VGG16 และ VGG19 ซึ่งเลข 16 และ เลข 19 นั้นมาจากคือจำนวนของ hidden layer สำหรับ VGG16 จะประกอบไปด้วยบล็อกของ convolutional layer และ Pooling ทั้งหมด 13 layers และ Fully connected layer 3 layers ส่วน VGG 19 นั้นคือ VGG16 ที่มีบล็อกของ convolutional layer เพิ่มมาอีก 3 layers ดังรูปที่ 2.8

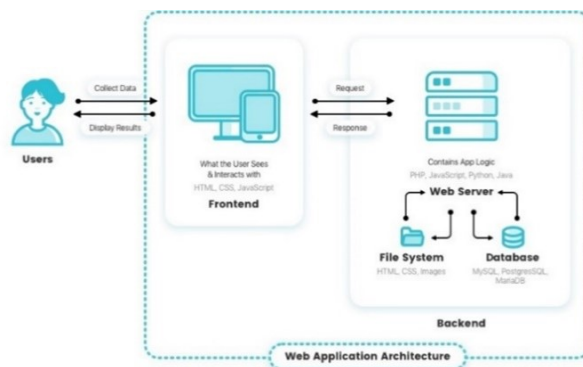
ConvNet configuration	
VGG16	VGG19
16 weight layers	19 weight layers
Input (224 x 224 RGB image)	
Conv3-64	Conv3-64
Conv3-64	Conv3-64
Max pool	
Conv3-128	Conv3-128
Conv3-128	Conv3-128
Max pool	
Conv3-256	Conv3-256
Conv3-256	Conv3-256
Conv3-256	Conv3-256
Max pool	
Conv3-512	Conv3-512
Conv3-512	Conv3-512
Conv3-512	Conv3-512
Max pool	
Conv3-512	Conv3-512
Conv3-512	Conv3-512
Conv3-512	Conv3-512
Max pool	
FC-4096	
FC-4096	
FC-4096	
Soft-max	

รูปที่ 2.8 แสดงสถาปัตยกรรมของ VGG16 และ VGG19

2.2.6 Web Application

Web Application เป็นซอฟต์แวร์ที่สามารถเข้าถึงผ่าน browser ใดก็ได้ โดยที่ผู้ใช้ไม่ต้องดาวน์โหลดก่อนใช้งานเพราะสามารถเข้าถึงได้ผ่าน network ซึ่ง Web Application นั้นออกแบบมาให้สามารถมีปฏิสัมพันธ์กับผู้ใช้ได้ ไม่ใช่แค่การดูข้อมูลต่าง ๆ แต่สามารถจัดการหรือใช้งานได้ ซึ่งผู้ใช้หลายคนสามารถเข้าถึงพร้อมกัน

หลักการทำงานของ web Application คือ ผู้ใช้งานติดต่อสื่อสารกับ web application ผ่าน frontend หรือ client-side ซึ่ง frontend จะมีการเรียกใช้หรือดึงข้อมูลออกมาจาก backend หรือ server-side ผ่าน http request ที่ถูกเรียกใช้โดยผู้ใช้งาน ภายใน backend จะมี web server, database และไฟล์อื่น ๆ เก็บไว้ ดังรูปที่ 2.9



รูปที่ 2.9 แสดงหลักการทำงานของ web application

2.2.7 Development Tools

2.2.7.1 Python

Python เป็นภาษาที่ใช้ในการเขียนโปรแกรม ซึ่งได้รับความนิยมเป็นอย่างมาก เนื่องจากเป็นภาษาที่เรียนรู้ง่าย ใช้งานง่ายและมีประสิทธิภาพสูง ถูกพัฒนาขึ้นโดย Guido van Rossum ในปี ค.ศ. 1991 ซึ่งสามารถทำงานบนแพลตฟอร์มต่าง ๆ ได้หลากหลาย มีไวยากรณ์ที่เข้าใจง่าย สามารถเขียนได้ทั้งแบบเชิงวัตถุ object-oriented หรือแบบฟังก์ชัน functional และรันบน interpreter system ซึ่งประมวลผลได้อย่างรวดเร็วและมี library หลากหลายให้ใช้งาน

2.2.7.2 Hypertext Markup Language (HTML)

Hypertext Markup Language (HTML) เป็นภาษาคอมพิวเตอร์ที่ใช้ในการแสดงผลของข้อความ รูปภาพหรือวัตถุอื่น ๆ บนหน้าเว็บไซต์ โดยมีโครงสร้างการเขียนที่อาศัย tag ควบคุมซึ่งถูกพัฒนาโดย World Wide Web Consortium (W3C) โดยออกแบบมาให้สามารถทำความเข้าใจและเรียนรู้ง่าย โดยใน tag ต่าง ๆ จะแตกต่างกันไปแต่มีส่วนประกอบหลักที่เหมือนกันคือ เครื่องหมาย "<>" และมีชื่ออยู่ตรงกลางเครื่องหมายนั้น ตัวอย่างเช่น <head> </head> เป็นต้น

2.2.7.3 Cascading Style Sheets (CSS)

Cascading Style Sheets (CSS) หรือ Style Sheets เป็นภาษาคอมพิวเตอร์ตัวหนึ่ง ที่ใช้สำหรับกำหนดรูปแบบการแสดงผลของเอกสาร html อันได้แก่ สีข้อความ ขนาดข้อความ ประเภทตัวอักษร ขนาดรูปภาพ การจัดวางข้อความ สีพื้นหลัง เป็นต้น

2.2.7.4 JavaScript

JavaScript เป็นภาษาคอมพิวเตอร์ชนิดหนึ่งที่ใช้ร่วมกับเอกสาร html เพื่อให้เว็บไซต์มีการเคลื่อนไหวเพื่อตอบสนองต่อการใช้งานของผู้ใช้มากขึ้น โดยมีลักษณะการทำงานคือ แปลความและดำเนินการไปที่คำสั่ง หรือ object-oriented programming ซึ่งดำเนินการโดย browser เป็น client-side script ทำให้สามารถใช้งานกับ server ได้ โดย JavaScript นี้สามารถเขียนเพื่อเปลี่ยนแปลง html element ได้ สามารถตอบสนองกับผู้ใช้และตรวจสอบข้อมูล รวมไปถึงสร้าง cookies ที่ใช้เก็บข้อมูลในคอมพิวเตอร์ของผู้ใช้ได้

2.2.7.5 Pytorch

Pytorch เป็น Deep learning library ที่ Facebook เป็นคนพัฒนาขึ้นด้วยภาษา python โดยดัดแปลงมาจาก library torch ที่ถูกใช้ในภาษา Lua มาก่อน โดยจะมองข้อมูลให้อยู่ในรูปของ tensor ซึ่งเป็นการเก็บข้อมูลหลายมิติแบบ arrays ทำให้สามารถใช้ numpy ในการคำนวณได้เลย และมีการเตรียม neural network แบบต่าง ๆ ไว้ให้ผู้พัฒนาสามารถนำไปพัฒนาต่อได้อย่างรวดเร็ว และยังสามารถใช้ GPU เข้ามาช่วยในการคำนวณได้ ทำให้สามารถประมวลผลได้เร็วขึ้น

2.2.7.6 Django

Django เป็น high-level python web framework ที่ช่วยให้ผู้พัฒนาสามารถสร้าง web application ได้อย่างรวดเร็ว ซึ่งจุดเด่นของ Django คือ มีระบบ admin มาให้ทำให้ผู้พัฒนาไม่จำเป็นต้องสร้างใหม่เอง สามารถใช้งานกับโปรแกรม editor ได้หลากหลายตามที่ผู้พัฒนานัด มีเครื่องมือที่ช่วยให้ผู้พัฒนาสามารถขึ้นโครงโปรแกรมได้อย่างรวดเร็ว ตัวอย่างเช่น ระบบการจัดการข้อมูล(models), ระบบ

แสดงผล(vIEWS), ระบบการเข้าใช้งานของผู้ใช้, ระบบการจัดการผู้ใช้งาน เป็นต้น รวมทั้งมี Django REST Framework ที่ช่วยให้ผู้พัฒนาสามารถสร้าง API เพื่อให้ในส่วน frontend สามารถติดต่อกับส่วน backend ได้อย่างสะดวก รวดเร็ว

หลักการของ Django คือ 1 Project มี 1 app โดย 1 app คือ 1 module ในเว็บไซต์ ในการสร้าง Django Project [9] ผู้ใช้จะได้ไฟล์หลายไฟล์ดังรูปที่ 2.10 ซึ่งไฟล์หลัก ๆ ประกอบไปด้วย

บทที่ 3 วิธีการดำเนินงาน

Explain the design (how you plan to implement your work) of your project. Adjust the section titles below to suit the types of your work. Detailed physical design like circuits and source codes should be placed in the appendix.

3.1 ข้อกำหนดและความต้องการของระบบ

3.2 สถาปัตยกรรมระบบ

ตารางที่ 3.1 test table x1

SYMBOL		UNIT
α	Test variable	m ²
λ	Interarrival rate	jobs/ second
μ	Service rate	jobs/ second

3.3 Hardware Module 1

3.3.1 Component 1

3.3.2 Logical Circuit Diagram

3.4 Hardware Module 2

3.4.1 Component 1

3.4.2 Component 2

3.5 Path Finding Algorithm

3.6 Database Design

3.7 UML Design

3.8 GUI Design

3.9 การออกแบบการทดลอง

3.9.1 ตัวชี้วัดและปัจจัยที่ศึกษา

3.9.2 รูปแบบการเก็บข้อมูล

บทที่ 4 ผลการดำเนินงาน

You can title this chapter as **Preliminary Results** ผลการดำเนินงานเบื้องต้น or **Work Progress** ความก้าวหน้าโครงการ for the progress reports. Present implementation or experimental results here and discuss them. ใส่เฉพาะหัวข้อที่เกี่ยวข้องกับงานที่ทำ

- 4.1 ประสิทธิภาพการทำงานของระบบ
- 4.2 ความพึงพอใจการใช้งาน
- 4.3 การวิเคราะห์ข้อมูลและผลการทดลอง

บทที่ 5 บทสรุป

This chapter is optional for proposal and progress reports but is required for the final report.

5.1 สรุปผลโครงการ

สรุปว่าโครงการบรรลุตามวัตถุประสงค์ที่ตั้งไว้หรือไม่ อย่างไร

5.2 ปัญหาที่พบและการแก้ไข

State your problems and how you fixed them.

5.3 ข้อจำกัดและข้อเสนอแนะ

ข้อจำกัดของโครงการ What could be done in the future to make your projects better.

บรรณานุกรม

1. Jason Brownlee, 2017, “A gentle introduction to transfer learning for deep learning,” **Machine Learning Mastery**, 2017.
2. Leon A Gatys, Alexander S Ecker, and Matthias Bethge, 2015, “A neural algorithm of artistic style,” **arXiv preprint arXiv:1508.06576**, 2015.
3. Aurélien Géron, 2018, “Neural networks and deep learning,” 2018.
4. Sumit Saha, 2018, “A comprehensive guide to convolutional neural networks—the ELI5 way,” **Towards Data Science**, vol. 15, 2018.
5. Dipanjan Sarkar, 2018, “A comprehensive hands-on guide to transfer learning with real-world applications in deep learning,” **Towards Data Science**. Saatavissa: <https://towardsdatascience.com/a-comprehensive-handson-guide-to-transfer-learning-with-real-world-applications-in-deep-learning-212bf3b2f27a>. Hakupäivä, vol. 20, pp. 2020, 2018.
6. C. Zheng and Y. Zhang, 2018, “Two-Stage Color ink Painting Style Transfer via Convolution Neural Network,” in **2018 15th International Symposium on Pervasive Systems, Algorithms and Networks (I-SPAN)**, 2018, pp. 193–200.

ภาคผนวก A

ชื่อภาคผนวกที่ 1

ใส่หัวข้อตามความเหมาะสม

This is where you put hardware circuit diagrams, detailed experimental data in tables or source codes, etc..

This appendix describes two static allocation methods for fGn (or fBm) traffic. Here, λ and C are respectively the traffic arrival rate and the service rate per dimensionless time step. Their unit are converted to a physical time unit by multiplying the step size Δ . For a fBm self-similar traffic source,

$$C = \lambda + (\kappa(H)\sqrt{-2\ln \epsilon})^{1/H} a^{1/(2H)} x^{-(1-H)/H} \lambda^{1/(2H)} \quad (\text{A.1})$$

where $\kappa(H) = H^H(1-H)^{(1-H)}$. Simplicity in the calculation is the attractive feature of (A.1).

The MVA technique developed in [?] so far provides the most accurate estimation of the loss probability compared to previous bandwidth allocation techniques according to simulation results. Consider a discrete-time queueing system with constant service rate C and input process λ_n with $\mathbb{E}\{\lambda_n\} = \lambda$ and $\text{Var}\{\lambda_n\} = \sigma^2$. Define $X_n \equiv \sum_{k=1}^n \lambda_k - Cn$. The loss probability due to the MVA approach is given by

$$\epsilon \approx \alpha e^{-m_x/2} \quad (\text{A.2})$$

where

$$m_x = \min_{n \geq 0} \frac{((C - \lambda)n + B)^2}{\text{Var}\{X_n\}} = \frac{((C - \lambda)n^* + B)^2}{\text{Var}\{X_{n^*}\}} \quad (\text{A.3})$$

and

$$\alpha = \frac{1}{\lambda\sqrt{2\pi\sigma^2}} \exp\left(\frac{(C - \lambda)^2}{2\sigma^2}\right) \int_C^\infty (r - C) \exp\left(\frac{(r - \lambda)^2}{2\sigma^2}\right) dr \quad (\text{A.4})$$

For a given ϵ , we numerically solve for C that satisfies (A.2). Any search algorithm can be used to do the task. Here, the bisection method is used.

Next, we show how $\text{Var}\{X_n\}$ can be determined. Let $C_\lambda(l)$ be the autocovariance function of λ_n . The MVA technique basically approximates the input process λ_n with a Gaussian process, which allows $\text{Var}\{X_n\}$ to be represented by the autocovariance function. In particular, the variance of X_n can be expressed in terms of $C_\lambda(l)$ as

$$\text{Var}\{X_n\} = nC_\lambda(0) + 2 \sum_{l=1}^{n-1} (n-l)C_\lambda(l) \quad (\text{A.5})$$

Therefore, $C_\lambda(l)$ must be known in the MVA technique, either by assuming specific traffic models or by off-line analysis in case of traces. In most practical situations, $C_\lambda(l)$ will not be known in advance, and an on-line measurement algorithm developed in [?] is required to jointly determine both n^* and m_x . For fGn traffic, $\text{Var}\{X_n\}$ is equal to $\sigma^2 n^{2H}$, where $\sigma^2 = \text{Var}\{\lambda_n\}$, and we can find the n^* that minimizes (A.3) directly. Although λ can be easily measured, it is not the case for σ^2 and H . Consequently, the MVA technique suffers from the need of prior knowledge traffic parameters.

ภาคผนวก B

ชื่อภาคผนวกที่ 2

ใส่หัวข้อตามความเหมาะสม

Next, we show how $\text{Var}\{X_n\}$ can be determined. Let $C_\lambda(l)$ be the autocovariance function of λ_n . The MVA technique basically approximates the input process λ_n with a Gaussian process, which allows $\text{Var}\{X_n\}$ to be represented by the autocovariance function. In particular, the variance of X_n can be expressed in terms of $C_\lambda(l)$ as

$$\text{Var}\{X_n\} = nC_\lambda(0) + 2 \sum_{l=1}^{n-1} (n-l)C_\lambda(l) \quad (\text{B.1})$$

Add more topic as you need

Therefore, $C_\lambda(l)$ must be known in the MVA technique, either by assuming specific traffic models or by off-line analysis in case of traces. In most practical situations, $C_\lambda(l)$ will not be known in advance, and an on-line measurement algorithm developed in [?] is required to jointly determine both n^* and m_x . For fGn traffic, $\text{Var}\{X_n\}$ is equal to $\sigma^2 n^{2H}$, where $\sigma^2 = \text{Var}\{\lambda_n\}$, and we can find the n^* that minimizes (A.3) directly. Although λ can be easily measured, it is not the case for σ^2 and H . Consequently, the MVA technique suffers from the need of prior knowledge traffic parameters.