



YOUR ART PAINTER

(จิตกรผู้สร้างศิลปะของคุณ)

MR. NAWARIT LONGKHAM

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

Your Art Painter  
(ຈິຕຣກຣູສ່ວັງສີລະປະຂອງຄຸນ)

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

.....  
(Dr.Nuttanart Facundes)

Project Advisor

.....  
(Dr.Natasha Dejdumrong)

Committee Member

.....  
(Dr.Unchalisa Taetragool)

Committee Member

.....  
(Dr.Thumrongrat Amornraksa)

Committee Member

Project Title	Your Art Painter (จิตกรผู้สร้างศิลปะของคุณ)
Credits	3
Member(s)	Mr. Nawarit Longkhum Ms. Pataraphorn Tanutsiriteeradet Ms. Sukasama Chitakson
Project Advisor	Dr.Nuttanart Facundes
Program	Bachelor of Engineering
Field of Study	Computer Engineering
Department	Computer Engineering
Faculty	Engineering
Academic Year	2020

### Abstract

blank

**Keywords:** Neural Style Transfer / Convolutional Neural Network

หัวข้อปริญญา呢นนธ์	จิตรกรผู้สร้างศิลปะของคุณ Your Art Painter
หน่วยกิต	3
ผู้เขียน	นายนวฤทธิ์ หล่องคุ้ม
	นางสาวภารพร ชนชีริรัตน์
	นางสาวสุกanya จิตอักษร
อาจารย์ที่ปรึกษา	ผศ.ดร.ณัฐนาถ ฟากุนเดช
หลักสูตร	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมคอมพิวเตอร์
คณะ	วิศวกรรมศาสตร์
ปีการศึกษา	2563

บทคัดย่อ

ว่าง

## กิตติกรรมประกาศ

โครงการนี้จะไม่สามารถเกิดขึ้นได้หากปราศจากความกรุณาจาก ผศ.ดร.ณัฐนาถ พาคุณเดช ออาจารย์ที่ปรึกษาโครงการที่ได้ให้คำเสนอแนะ-แนวคิดตลอดจนตอบข้อสงสัยและแก้ไขข้อบกพร่องมากมายมากโดยตลอดโครงการเล่มนี้เสร็จสมบูรณ์

นวฤทธิ์ หล่อคุ้ม  
ภัทรพร ธนาศิริรีเดช  
สุกษมา จิตอักษร

## สารบัญ

หน้า

ABSTRACT	ii
บทคัดย่อ	iii
กิตติกรรมประกาศ	iv
สารบัญ	vii
สารบัญตาราง	viii
สารบัญรูปภาพ	ix
 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 และร่างแผนการดำเนินงาน	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
2.2.4.1    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
2.2.7.7    MongoDB	14

2.3	การทบทวนวรรณกรรม	15
<b>บทที่ 3 วิธีการดำเนินงาน</b>		<b>16</b>
3.1	บทนำ	16
3.2	รายละเอียดของโครงการ	16
3.2.1	การทบทวนวรรณกรรม	16
3.2.2	Modeling	17
3.2.3	System Requirements สำหรับเว็บแอปพลิเคชัน	18
3.2.4	แผนผังแสดงความสัมพันธ์ (Use Case Diagram)	19
3.2.5	แผนภาพกิจกรรม (Activity Diagram)	20
3.2.5.1	ระบบลงชื่อเข้าใช้งาน	20
3.2.5.2	ระบบบัญชีผู้ใช้	21
3.2.5.3	ระบบวิธีการใช้งาน	22
3.2.5.4	ระบบเกี่ยวกับเรา	23
3.2.5.5	ระบบสร้างรูปภาพ	24
3.2.5.6	ระบบชุมชน	25
3.3	โดยรวมสถาปัตยกรรมและคำอธิบาย	26
3.3.1	User interface design	26
3.3.2	Database	30
3.3.3	Data Management	32
3.3.4	Data structures/Class diagrams/ etc.	32
3.3.4.1	Software Architectural Design	32
3.3.5	Evaluation plans	32
3.4	ภาคการเรียนที่ 2	33
3.4.1	แผนดำเนินงาน	33
3.4.2	modeling	34
<b>บทที่ 4 ผลการดำเนินงาน</b>		<b>35</b>
4.1	Modeling	35
4.1.1	ปรับเปลี่ยนโมเดล VGG19 จาก max pooling เป็น average pooling	35
4.1.2	ทำการทดลองโดยใช้ target image เป็น noise image และ target image เป็นภาพที่มาจากการคัดลอก content image	35
4.1.3	ปรับเปลี่ยนขนาดของ style image	36
4.1.4	ปรับเปลี่ยนจำนวนรอบที่ใช้ในการ train เพื่อสร้างรูปภาพใหม่	37
4.1.5	ปรับเปลี่ยน style weight และ content weight	37
4.2	WEB APPLICATION	38
4.2.1	User Interface	38
4.2.1.1	Homepage	38
4.2.1.2	Create your art Page	38
4.2.1.3	Community Page	38
4.2.1.4	Login Page	38
4.2.1.5	Register Page	38
4.2.2	Backend และ Database	41
4.2.2.1	Module assets: เก็บข้อมูล static (image, css) ที่ถูกรวบรวมมาจาก directory ต่าง ๆ	41
4.2.2.2	Module authentication: เป็น module ที่สำหรับการ login, register	41
4.2.2.3	Module community: ใช้สำหรับแสดงผลภาพที่มีคนโพสต์	41
4.2.2.4	Module create_your_art: ใช้สำหรับเก็บประมวลผล Neural style transfer	41
4.2.2.5	Module Homepage: ใช้สำหรับแสดงผลหน้า homepage	41
4.2.2.6	Module template: ใช้สำหรับเก็บ fronten ที่เป็น .html	41

4.2.2.7	Module user_info: ใช้สำหรับหน้า profile	41
4.2.2.8	Module your_art_painter: เป็น module หลักในการ setting ค่าต่าง ๆ	41
<b>บทที่ 5 บทสรุป</b>		<b>42</b>
5.1	ผลการดำเนินงานในภาคการศึกษาที่ 1 (Midterm)	42
5.2	ผลการดำเนินงานในภาคการศึกษาที่ 2 (Final)	42
<b>หนังสืออ้างอิง</b>		<b>43</b>
<b>APPENDIX</b>		<b>44</b>
A	ชื่อภาคผนวกที่ 1	45
B	ชื่อภาคผนวกที่ 2	47

## สารบัญตาราง

ตารางที่	หน้า
1.1 ตารางแผนการทำงานภาคการศึกษาที่ 1	4
1.2 ตารางแผนการทำงานภาคการศึกษาที่ 2	5
3.1 ตารางอธิบายระบบลงชื่อเข้าใช้	20
3.2 ตารางอธิบายระบบบัญชีผู้ใช้งาน	21
3.3 ตารางอธิบายระบบวิธีการใช้งาน	22
3.4 ตารางอธิบายระบบเกี่ยวกับเรา	23
3.5 ตารางอธิบายระบบสร้างรูปภาพ	24
3.6 ตารางอธิบายระบบข้อมูล	25
3.7 แสดงรายละเอียดของตาราง user_info	30
3.8 แสดงรายละเอียดของตาราง file_upload	30
3.9 แสดงรายละเอียดของตาราง style_info	31
3.10 แสดงรายละเอียดของตาราง style_info	31
3.11 แสดงรายละเอียดของตาราง rating	31
3.12 แสดงรายละเอียดของตาราง like_info	31
3.13 ตารางแผนการทำงานภาคการศึกษาที่ 2 ใหม่	33

## สารบัญ

รูปที่	หน้า
2.1 แสดงหลักการทำงานของ Neural Style Transfer ที่ใช้ Convolutional Neural Network (CNN) [1]	7
2.2 แสดงส่วนประกอบหลักของ Artificial neural networks	8
2.3 แสดงการทำงานภายใน neuron unit [2]	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 [3]	11
2.8 แสดงสถาปัตยกรรมของ VGG16 และ VGG19	12
2.9 แสดงหลักการทำงานของ web application	13
2.10 แสดงโครงสร้างฟล็อเดอร์ของ Django project	14
2.11 แสดงกระบวนการทำงานของ Django	14
3.1 (bn) แสดงผลการประมวลผล (ล่างซ้าย) content image (ล่างขวา) style image: ชื่อภาพ ใบวัด ของชุด นิ่มเสมอ	16
3.2 (bn) แสดงผลการประมวลผล (ล่างซ้าย) content image (ล่างขวา) style image: ชื่อภาพ ผูกอุบะของจักรพันธุ์ โปษยกฤต	17
3.3 แสดงภาพรวม Use Case Diagram	19
3.4 แสดงระบบลงชื่อเข้าใช้งาน	20
3.5 แสดงระบบบัญชีใช้งาน	21
3.6 แสดงระบบบริการใช้งาน	22
3.7 แสดงระบบเกี่ยวกับเรา	23
3.8 แสดงระบบเกี่ยวกับเรา	24
3.9 แสดงระบบบัญชี	25
3.10 แสดงหน้า Homepage	27
3.11 แสดงหน้า create your art ในขั้นตอนการเลือก content image และ style image	27
3.12 แสดงหน้า create your art ในขั้นตอนการเลือก content image และ style image	28
3.13 แสดงหน้า profile	28
3.14 แสดงหน้า Community	29
3.15 แสดงหน้า Entity Relationship diagram ของ database	30
3.16 แสดงโครงสร้างพื้นฐานของระบบซอฟต์แวร์ภายนอกเว็บไซต์ Your Art Painter	32
3.17 แสดงขั้นตอนการทำงานของ Style Transfer	34
4.1 แสดงผลลัพธ์ที่ได้จากการเปลี่ยน pooling layer ใน VGG19	35
4.2 แสดงผลลัพธ์ที่เกิดจากการใช้ target image ที่แตกต่างกัน	36
4.3 แสดงผลลัพธ์ที่เกิดจากการปรับขนาดรูปภาพ style image	36
4.4 แสดงผลลัพธ์ที่เกิดจากการเพิ่มจำนวนรอบในการ train เพื่อสร้างรูปภาพใหม่	37
4.5 แสดงผลลัพธ์ที่เกิดจากการปรับ weight โดยใช้ style image เป็นรูปภาพของ อ.ชลุด นิ่มเสมอ	38
4.6 แสดงผลลัพธ์ที่เกิดจากการปรับ weight โดยใช้ style image เป็นรูปภาพของ อ.จักรพันธุ์ โปษยกฤต	38
4.7 แสดงหน้า Homepage	39
4.8 แสดงหน้า Create your art	39
4.9 แสดงหน้า Community	40
4.10 แสดงหน้า Login	40
4.11 แสดงหน้า Register	40
4.12 แสดง App (module) ในส่วนของ Backend	41

# บทที่ 1 บทนำ

## 1.1 ที่มาและความสำคัญ

กว่า 5,000 ปีมาแล้ว ตั้งแต่ยุคก่อนประวัติศาสตร์ ที่มนุษย์รู้จักการวาดรูป โดยเริ่มแรกมนุษย์วาดภาพครั้งแรกโดยการขีดขูดบนผนังถ้ำ-หรือเพิงพา โดยรูปภาพที่ widać ก็จะเป็นภาพคน สัตว์ หรือการล่าสัตว์ ซึ่งแสดงให้เห็นถึงชีวิตความเป็นอยู่ของคนในยุคสมัยนั้น และหลังจากนั้นมนุษย์ได้มีการพัฒนาการวาดรูปอย่างต่อเนื่อง เจ้าของการวาดรูปตามผนังถ้ำ ตามผนังโบสถ์ หรือราชวังที่สำคัญต่างๆ จนมาเป็นวาระรูปตามผนังฝ้า ตลอดจนมาเป็นในกระดาษ ภาพวดนั้นจะมีความเพื่อถ่ายทอดเรื่องราวในอีดี้แล้ว ยังแสดงให้เห็นถึงเอกลักษณ์ของศิลปินอีกด้วย ซึ่งศิลปินของโลกทางตะวันตกและตะวันออกมีลักษณะที่แตกต่างกัน และในแต่ละประเทศหรือศิลปินแต่ละท่านก็จะมีเทคนิคหรือการวาดที่ไม่เหมือนกัน ศิลปินไทยที่มีชื่อในการวาดภาพ เช่น อ.เฉลิมชัย โชเชิดพิพัฒน์ อ.ชลุต นิ่มเสมอ และ อ.จักรพันธุ์ โปษยกฤต เป็นต้น style ในการวาดของศิลปินแต่ละท่านก็มีความแตกต่างกัน ทำให้ผู้คนชื่นชอบงานของท่าน และอย่างมีภาพที่มีสีสันแบบนั้น

จนกระทั่งปี 2015 Gatys et al. ได้ตีพิมพ์ผลงาน A Neural Algorithm of Artistic Style ซึ่งเป็นการศึกษาเกี่ยวกับการนำภาพ style ที่มีชื่อเสียงไปรวมกับภาพวิวเมือง Tubingen, ประเทศเยอรมัน โดยใช้แนวคิดของ Convolutional Neural Network (CNN) ได้สำเร็จ ออกแบบเป็นรูปภาพที่เป็นรูปภาพวิวเมือง Tubingen แต่ได้ใช้เทคนิคการวาดรูปจากศิลปินชื่อดัง ทำให้ภาพที่ได้มามีน้ำตาลุமื่อนศิลปินที่เสียชีวิตไปแล้วกลับมาราดรูปเข้าใหม่ จากความสำเร็จครั้งนั้น สร้างผลให้มีคนจำนวนมากสนใจในการทำ Neural Style Transfer (NST) และ CNN มากขึ้น

ดังนั้นทางคณะผู้จัดทำจึงต้องการศึกษาวิธีการทำ Neural Style Transfer ที่มี Convolutional Neural Network (CNN) เป็นทฤษฎีเบื้องหลัง และนำความรู้ที่ได้มาพัฒนาโมเดลที่สามารถทำ Neural Style Transfer รูปที่วาดโดย อ.จักรพันธุ์ โปษยกฤตและ อ.ชลุต นิ่มเสมอ ซึ่งเป็นศิลปินไทยได้อีกทั้งเพื่อเพื่อพัฒนา web application ที่สามารถสร้างรูปภาพจากที่มีลักษณะของศิลปินผู้สนใจได้

## 1.2 ประเภทของโครงงาน

นำเสนอความต้องการของผู้มีส่วนได้ส่วนเสียเฉพาะกลุ่ม ผลิตภัณฑ์ทางการค้าที่มีศักยภาพ และ เว็บแอปพลิเคชันสำหรับคอมพิวเตอร์

## 1.3 วิธีการนำเสนอ

ในการบรรยายที่จะนำเสนอ ใช้วิธีการตั้งต่อไปนี้

### 1. เตรียมข้อมูลเพื่อนำไปใช้ train โมเดล

รวบรวมรูปภาพที่สร้างจากศิลปินที่มีชื่อเสียงและลักษณะภาพสามารถบอกได้ถึงเอกลักษณ์ของศิลปินท่านนั้นได้ โดยทางผู้จัดทำได้รวมรูปภาพของศิลปิน จำนวน 2 ท่าน ได้แก่ อ.จักรพันธุ์ โปษยกฤต และ อ.ชลุต นิ่มเสมอ เพื่อใช้เป็น style image และ รวบรวมรูปภาพคนหรือสัตว์ทั่ว 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) อยู่เบื้องหลังมาใช้เพื่อค้นหารูปแบบและลักษณะเด่นของภาพอุปกรณ์ (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 มาใช้งานโมเดลแล้วแก้ไขผิดพลาดและพัฒนาให้โมเดลดียิ่งขึ้นจากเดิม

### 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 ที่สามารถสร้างรูปภาพจากที่มีลักษณะของศิลปินผู้สมพسانได้

#### 1.5 ขอบเขตของโครงการ

1. สร้าง Model ที่สามารถใช้งานกับรูปภาพของศิลปินไทยได้ โดยใช้ CNN
2. สร้างรูปภาพใหม่ที่เกิดจากการผสมผสานระหว่างรูปภาพที่สร้างโดยศิลปินไทยกับภาพถ่ายคนหรือภาพสัตว์ โดยใช้ภาษา python
3. สร้าง web application ด้วย Django framework ได้
4. ผู้ใช้สามารถเลือกแบบรูปภาพของแต่ละศิลปิน โดยศิลปินที่จะนำมาเป็น style image ได้แก่  
อ.จักรพันธุ์ โปษยกฤต  
อ.ชลุด นิ่มเสมอ

#### 1.6 เนื้อหาทางวิศวกรรมที่เป็นต้นฉบับ

ในส่วนที่คณผู้จัดทำจะทำการรวมรูปภาพที่สร้างโดยศิลปินหรือ Style image เข้ากับรูปภาพคนหรือสัตว์หรือ content image เข้าด้วยกันเป็นภาพใหม่ที่มีรูปหลักแบบ content image และมีการคาดแบบ style image ผสมเข้าด้วยกัน โดยได้เลือกรูปภาพของศิลปินที่สนใจมา 2 ท่านได้แก่ อ.จักรพันธุ์ โปษยกฤต และ อ. ชลุด นิ่มเสมอ มาใช้เป็น style image และเลือกใช้ภาษา python ในการพัฒนาโมเดลซึ่งมี library และโมเดลที่หลายหลายในเลือกใช้ จึงเลือกใช้ VGGNet ที่มีอัลกอริทึม Convolutional neural network (CNN) อยู่เบื้องหลังมาใช้งาน และได้มีการพัฒนาวิธีการปรับค่า weight และ bias ของโมเดล เพื่อให้ได้โมเดลที่มีค่า loss function ที่เหมาะสมและเหมาะสมกับรูปภาพที่ทางคณผู้จัดทำนำมา train

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

#### 1.7 การแยกย่อยงาน และร่างแผนการดำเนินงาน

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

5. ออกรอบโค้งสร้างและส่วนประกอบของโครงการ

- 1 ออกรอบ flow ของการทำงาน
- 2 ศึกษาโมเดล อัลกอริทึมและ library ที่เกี่ยวข้อง
- 3 ออกรอบ User Interface
- 4 ออกรอบ Database
- 5 ออกรอบโมเดลที่ใช้ทำงาน
- 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 ตามที่ได้ออกรอบไว้

- 1 Frontend
- 2 Backend

13. นำเสนอโครงการ

14. จัดทำรายงานแสดงความคืบหน้า

15. นำเสนอรายงานประจำการศึกษา

จากขั้นตอนการดำเนินงานข้างต้น สามารถแบ่งออกเป็น 2 ภาคการศึกษา โดยภาคการศึกษาที่ 1 นั้นมุ่งเน้นไปที่การศึกษา-หาข้อมูลและทดลองพัฒนาซอฟต์แวร์ตัวต้นแบบ ซึ่งสามารถแสดงรายการและระยะเวลาอุบัติได้เป็นแผนภูมิแกนต์ตั้งตามตารางที่ [1.1](#) และในส่วนภาคการศึกษาที่ 2 นั้นมุ่งเน้นไปที่การพัฒนาต่ออยอดซอฟต์แวร์ตัวต้นแบบให้สามารถทำงานได้อย่างถูกต้อง ครบถ้วนฟังก์ชันและแก้ไขปัญหาเมื่อซอฟต์แวร์ได้ทำงานในสภาพแวดล้อมจริง ตลอดจนถึงขั้นตอนสุดท้ายสามารถเขียนเป็นแผนภูมิแกนต์ ตั้งตารางที่ [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	3	4	
1. กำหนดหัวข้อโครงการที่ต้องการทำ																				
1.1 หาข้อมูลเกี่ยวกับสิ่งที่สนใจของท่า																				ทุกคน
1.2 ปรึกษาสมาชิกในกลุ่มและกำหนดหัวข้อโครงการ																				ทุกคน
1.3 นำเสนอผลของค่าน้ำจากอาจารย์ที่ปรึกษา																				ทุกคน
2. ศึกษาและรวบรวมข้อมูลเกี่ยวกับวิธีการและเทคโนโลยีที่ใช้ในโครงการ																				
2.1 รวบรวมข้อมูลเบื้องต้นเกี่ยวกับโครงการ																				นวศุลี
2.2 ศึกษาวิธีการท่องานและงานวิจัยที่เกี่ยวข้อง																				ทุกคน
3. ประเมินความเป็นไปได้และกำหนดขอบเขตของโครงการ																				ภัทรพร
4. จัดทำข้อเสนอโครงการและร่างแผนดำเนินการ																				
4.1 เขียน proposal report																				ทุกคน
4.2 วางแผนเวลาในการทำงานพร้อมแบ่งงานย่อยและมอบหมายให้กับสมาชิกในกลุ่ม																				ภัทรพร
5. ออกแบบโครงสร้างและส่วนประกอบของโครงการ																				
5.1 ออกแบบ flow ของการทำงาน																				สุกanya
5.2 ศึกษาโมเดล ชัลโกริฟฟ์ และ library ที่เกี่ยวข้อง																				ภัทรพร นว ฤทธิ์
5.3 ออกแบบ User Interface																				ภัทรพร
5.4 ออกแบบ Database																				สุกanya
5.5 ออกแบบโมเดลที่ใช้ทำงาน																				ภัทรพร นว ฤทธิ์
5.6 ศึกษาการ Optimization และการประนีดគัปประจิทวิภาคของโมเดล																				ภัทรพร
6. รวบรวมข้อมูลเพื่อใช้เป็นในการ train โมเดล																				
6.1 รวบรวมรูปภาพที่สร้างจากคิลป์เพื่อเป็น style image																				ภัทรพร นว ฤทธิ์
6.2 รวบรวมรูปภาพคนหรือสัตว์เพื่อเป็น content image																				ภัทรพร นว ฤทธิ์
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	
7. จัดการรูปภาพให้อยู่ในรูปแบบที่ใช้ในโมเดล (Data preprocessing)															ภัทรพร นวฤทธิ์
8. ทดลองสร้างโมเดลและปรับแต่งให้มีประสิทธิภาพที่สุด															
8.1 ทดลองสร้างโมเดลโดยใช้ parameter พื้นฐาน															ภัทรพร นวฤทธิ์
8.2 ปรับแต่งโมเดลตัวอย่างการปรับค่าของ parameter ต่าง ๆ และวัดผลที่ได้岀มา															ภัทรพร
8.3 เลือกโมเดลที่ได้ผลลัพธ์ที่ดีที่สุด															ภัทรพร
9. ทำให้โมเดลให้สามารถใช้งานได้ทั่วไปมากขึ้นโดยนำรูปภาพจาก ศิลปินชื่อดังมา train โมเดลเพิ่มเติม															ภัทรพร นวฤทธิ์
10. ทดลองใช้งานโมเดลโดยนำรูปภาพคนหรือสัตว์ มา test โมเดล เพิ่มเติม															ภัทรพร นวฤทธิ์
11. ประเมินผลโมเดลที่ได้สร้างขึ้นมา															ทุกคน
12. พัฒนา Web Application ตามที่ได้ออกแบบไว้															
12.1 Frontend															ภัทรพร นวฤทธิ์
12.2 Backend															อุษณียา
13. นำเสนอโครงการ															ทุกคน
14. จัดทำรายงานแสดงความคืบหน้า															ทุกคน
15. นำเสนอรายงานประจำการศึกษา															ทุกคน

ตารางที่ 1.2 ตารางแผนการทำงานภาคการศึกษาที่ 2

## 1.8 ผลการดำเนินการ

### 1.8.1 ผลการดำเนินการในภาคการศึกษาที่ 1

- ศึกษาและรวบรวมข้อมูลเกี่ยวกับวิธีการและเทคโนโลยีที่ใช้ในโครงการ
- ประเมินความเป็นไปได้และกำหนดขอบเขตของโครงการ
- จัดทำข้อเสนอโครงการและร่างแผนดำเนินการ
- ออกแบบโครงสร้างและส่วนประกอบของโครงการ
- รวบรวมข้อมูลเพื่อใช้เป็นในการ train โมเดล (style image และ content image)
- จัดการรูปภาพให้อยู่ในรูปแบบที่ใช้ในโมเดล (Data preprocessing)
- สร้างโมเดลโดยใช้ parameter พื้นฐาน
- 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

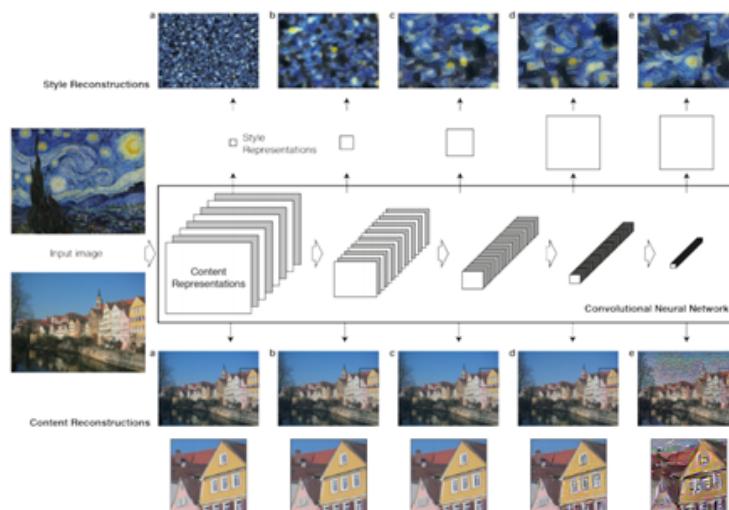
### 2.1 บทนำ

ในบทนี้ อธิบายเกี่ยวกับแนวคิดที่นำมาประยุกต์ใช้ในโครงการ อัลกอริทึมที่ใช้ในการทำโครงการคือ VGGNet และมีการพัฒนาโดยใช้ภาษา python นอกจากนี้มีการทบทวนวรรณกรรม ที่จะกล่าวถึงโครงการที่มีลักษณะเนื้อหาที่เกี่ยวข้องกับการทำโครงการนี้

### 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) [1]

ดังรูปที่ 2.1 จะเห็นได้ว่า มีการป้อนข้อมูลเป็น content image และ style image จากนั้นจากการประมวลผลรูปที่ป้อนเข้ามา 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 จนได้ได้รูปภาพที่เหมือนกับรูปภาพที่ใช้เป็น 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 ไปเรื่อยๆ ตัวอย่างเช่น การจัดกลุ่มข้อมูล โดยจะป้อนเฉพาะข้อมูลเข้าไปแล้วคอมพิวเตอร์จะเรียนรู้เองว่า ข้อมูลแบบนี้ควรอยู่กับข้อมูลแบบไหน เป็นต้นและ Reinforcement learning ที่เป็นการเรียนรู้โดยการปฏิสัมพันธ์กับสิ่งแวดล้อมรอบ ๆ โดยถ้าทำถูกต้องจะได้คะแนนเพิ่ม ถ้าทำผิดคะแนนจะลดลง ตัวอย่างเช่น การเดินไปข้างหน้าของหุ่นยนต์ที่จะต้องรู้ก่อนว่า อยู่ตำแหน่งไหน แล้วในตำแหน่งนั้นควรเดินไปตำแหน่งไหนต่อไปจะจะได้เป็นต้น

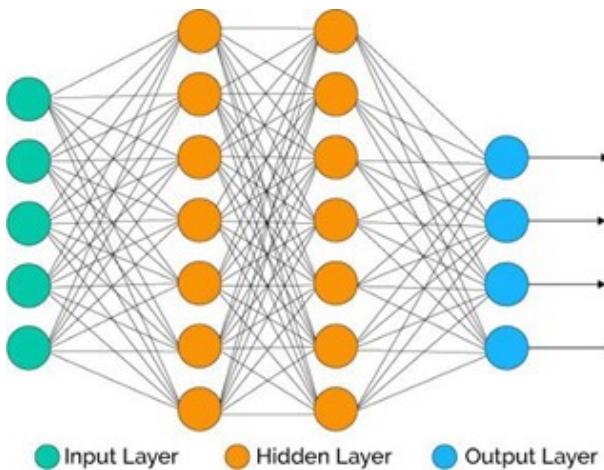
ในขั้นตอนการสร้างโมเดล machine learning จะมีข้อมูล input ที่เกี่ยวข้อง 2 ส่วนคือ training set เป็นชุดข้อมูลที่ใช้สำหรับการสอนโมเดลโดยจะฝึกฝนโมเดลให้ได้ผลลัพธ์ตามมาตรฐานของข้อมูลที่ป้อนเข้ามา อีกส่วนคือ testing set เป็นชุดข้อมูลที่ใช้ทดสอบการทำงานของโม-

เดล โดยโมเดลที่ดีจะต้องไม่ทำให้เกิดการ overfitting หรือสามารถทำงานอย่างผลลัพธ์ใน training set ได้แม่นยำ แต่ทำงานอย่างผลลัพธ์ใน testing set ได้ดี

เมื่อสอนคอมพิวเตอร์เรียนรู้อย่างแล้ว เมื่อวันข้อมูลใหม่ ๆ เข้า คอมพิวเตอร์ก็สามารถจัดการหรือทำงานข้อมูลนั้นได้โดย "ไม่จำเป็นต้องเขียนโปรแกรมใหม่ทุกรอบ" ที่มีข้อมูลเข้ามา

### 2.2.3 Classical artificial neural networks

Artificial neural networks(ANN) คือโมเดลทางคณิตศาสตร์หรือโมเดลทางคอมพิวเตอร์ที่จำลองการทำงานของเซลล์ประสาทในสมองมนุษย์ที่มีเซลล์ประสาทประสาท 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 ทำการประมวลผลแล้วได้ค่าอออกมาส่งไปที่ output layer และ back propagation หรือกระบวนการทำงานที่ย้อนกลับโดยจะเป็นการที่ย้อนกลับจาก output layer ไป input layer

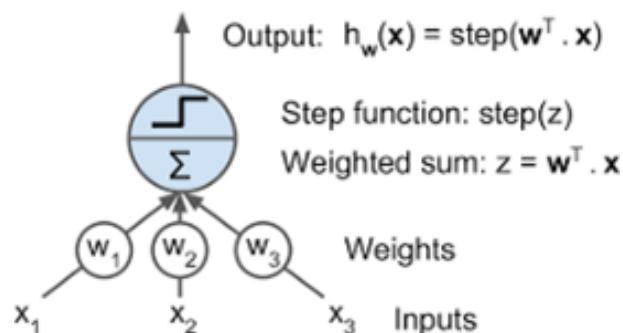


รูปที่ 2.2 แสดงส่วนประกอบหลักของ Artificial neural networks

ในแต่ละ layer จะมีก้อนกลม ๆ อยู่หรือคือ neuron unit อยู่ดังรูปที่ 2.2 ถ้าใน input layer ภายในจะเก็บข้อมูลที่รับเข้ามา แต่ถ้าเป็น hidden layer ภายในจะมีการคำนวณเกิดขึ้นโดยสามารถเพิ่ม hidden layer ได้หลาย layer ตามที่ผู้พัฒนาต้องการได้ ส่วนใน output layer จะมีการทำนายผลลัพธ์ที่มากกว่า ผลที่ได้เป็นคลาสอย่างไร

#### 2.2.3.1 Neuron unit ใน hidden layer

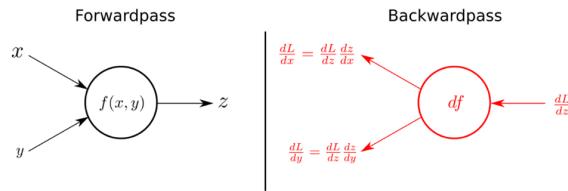
ภายใน neuron unit แต่ละตัวใน hidden layer จะมีการคำนวณโดยหลังจากผ่าน input layer เข้ามาจะมีการเพิ่ม weight ให้กับ input neuron unit แต่ละตัว จากนั้นเมื่อเข้า hidden layer ภายในจะรวมผลคูณระหว่าง weight กับ input neuron unit แต่ละตัวเข้าด้วยกันแล้วเข้า step function หรือส่วนใหญ่เรียกว่า activation function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



รูปที่ 2.3 แสดงการทำงานภายใน neuron unit [2]

### 2.2.3.2 Backward propagation

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



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

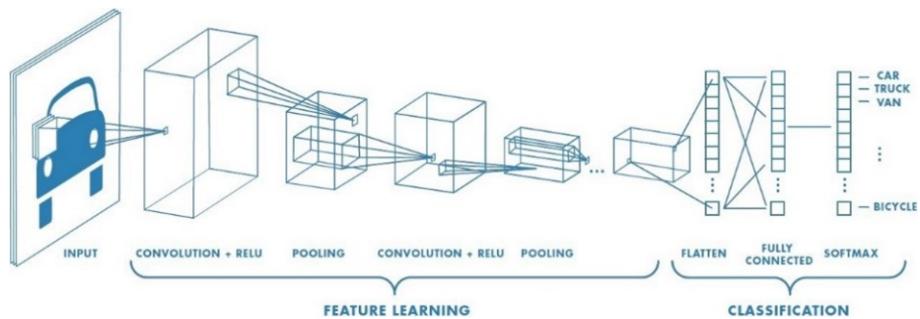
ในการทำ back propagation จะใช้อัลกอริทึม optimization เพื่อเพิ่มประสิทธิภาพตัวนึงที่ชื่อว่า Gradient descent algorithm [5] ที่จะทำการหา weight ที่ต่ำที่สุดที่จะทำให้โมเดลสามารถทำนายผลได้ดีที่สุด เกิด error น้อย

### 2.2.3.3 Cost function

Cost function หรือ loss function หรือค่า error ที่เกิดขึ้น ซึ่งจุดมุ่งหมายของโมเดลต่างๆ จะเหมือนกันคือ ต้องการให้เกิดค่า error ที่น้อยที่สุดที่เป็นไปได้ ซึ่งการคำนวณค่า error ที่เกิดขึ้นจะใช้การเปรียบเทียบระหว่างผลลัพธ์จริง ๆ ที่ได้มาจากการคำนวณกับผลลัพธ์ที่เกิดจากการทำนายโดยโมเดลมาเทียบกันโดยใช้วิธีต่างๆ ด้วยอย่างเช่น การคำนวณค่า Mean Squared Error Loss, Mean Absolute Error Loss, Binary Crossentropy Loss, Categorical Crossentropy Loss, Sparse Categorical Crossentropy Loss เป็นต้น ซึ่งการคำนวณแบบต่างๆ ก็จะมีความแตกต่างกันตามลักษณะแบบ ด้วยอย่างเช่น Binary Crossentropy Loss ที่หมายกับข้อมูลที่มีเพียง 2 คลาส เป็นต้น

### 2.2.4 Convolutional Neural Network (CNN)

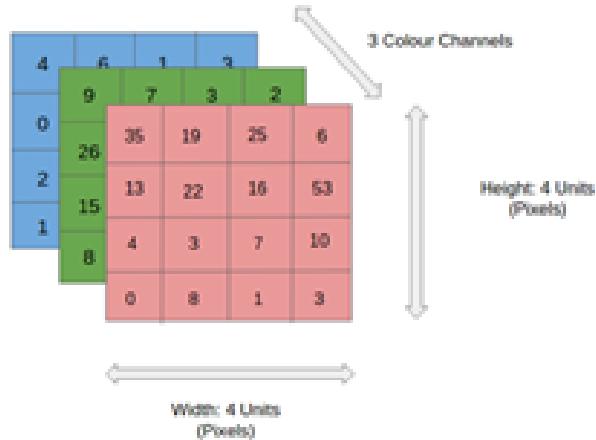
Convolutional Neural Network (CNN) หรือ ConvNet [6, 7] เป็นอัลกอริทึม Deep Learning ชนิดหนึ่งที่ใช้ประมวลผลรูปภาพโดยเรียนรู้คุณลักษณะต่างๆ ของรูปภาพที่ป้อนเข้าผ่านการใช้ filters ซึ่งเป็นการเรียนรู้ลักษณะของรูปภาพ เพื่อแยกแยะว่า รูปภาพที่ป้อนเข้ามาเป็นรูปอะไรดังรูปที่ 2.5



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 เข้ามาจะถูกแบ่งสีออกเป็น channels คือ Red(R), Green(G) และ Blue(B) และแบ่งขนาดความกว้างและความยาวของรูปภาพให้เป็นพิกเซล ทำให้อยู่ในรูป  $\text{heights} \times \text{weights} \times \text{จำนวนของ channels}$  ดังรูปที่ 2.6



รูปที่ 2.6 แสดงให้เห็นถึงลักษณะของ input data ที่มีการแบ่งชั้นสีเป็น 3 channels ซึ่งจากรูปนี้จะมีขนาดของรูปเป็น  $4 \times 4 \times 3$  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 ที่ใช้สำหรับบอกว่าต้องขยาย filter ไปกี่พิกเซล โดย filters จะเคลื่อนที่ไปเรื่อย ๆ จนผ่านครบทุกพิกเซลในข้อมูลที่เข้ามา และนอกจากนี้จะมีการใช้ non-linear functions เพื่อ scale ข้อมูลที่เข้ามาไม่ให้ติดลบซึ่งหมายความว่าใน CNNs โดยทั่วไปนิยมใช้คือ Rectified Linear Unit (ReLU) ที่มีฟังก์ชันคือ  $f(x) = \max(0, x)$

#### 2.2.4.3 Pooling layer

ในส่วนของ pooling layer จะทำงานต่อจาก convolution layer โดยจะทำการดึงข้อมูลออกมากหางจากที่ทำการทำ filter แต่ละครั้ง อกมาเป็นข้อมูลที่อกมา ณ ตำแหน่งนั้น ๆ ที่ทำการทำ filter ทำให้สามารถดึงลักษณะสำคัญของข้อมูลที่เข้ามาได้ ตัวอย่างเช่น ของสี รูปทรง เป็นต้น และยังทำให้ขนาดข้อมูลที่ได้ออกมา มีขนาดเล็กลงเมื่อเทียบกับข้อมูลที่เข้ามา โดย pooling มี 3 แบบ คือ min pooling, max pooling และ average pooling ซึ่ง convolution layer กับ pooling layer จะทำงานคู่กัน ซึ่งใน CNN จะมีกี่ layer นั้นขึ้นอยู่ กับความซับซ้อนของรูปภาพ

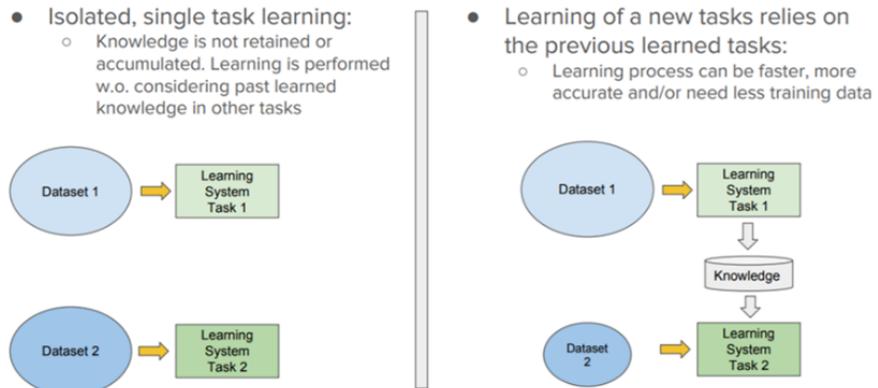
#### 2.2.4.4 Fully connected layer

เมื่อทำการ feature learning โดยใช้ convolution layer กับ pooling layer แล้ว output ที่ได้จะอยู่ในรูปแบบที่เป็น 3 ชั้นหรือมี 3 มิติอยู่ จะมีการนำ 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 ทำให้สามารถลดเวลาประมาณและเวลาในการสอนโมเดลได้ ดังรูปที่ 2.7

## Traditional ML vs Transfer Learning



รูปที่ 2.7 แสดงให้เห็นถึงการเปรียบเทียบการทำงานระหว่าง traditional ML กับ transfer learning ที่มีรูปแบบการทำงานที่ต่างกัน [3]

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

สำหรับการทำ transfer learning ที่เกี่ยวกับรูปภาพ[8][9] มีการสร้างโมเดลสำหรับ ImageNet ที่มีรูปภาพกว่า 1.4 ล้านรูปและมากกว่า 1,000 คลาสที่เป็นความท้าทายสำหรับการทำ image classification ทำให้มีองค์การวิจัยต่าง ๆ มีการพัฒนาโมเดลขึ้นมาสำหรับแก้ปัญหานี้ 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 และความคิดนี้ถูกพัฒนาโดย 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

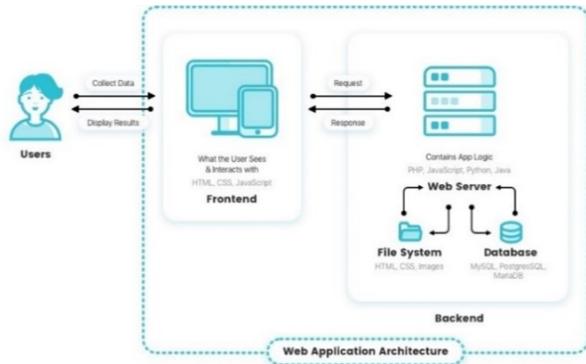
ConvNet configuration	
VGG16	VGG19
16 weight layers	19 weight layers
Input (224 x 224 RGB image)	
Conv3-64	Conv3-64
Conv3-64	Conv3-64
Max pool	
Conv3-128	Conv3-128
Conv3-128	Conv3-128
Max pool	
Conv3-256	Conv3-256
Conv3-256	Conv3-256
Conv3-256	Conv3-256
Max pool	
Conv3-512	Conv3-512
Conv3-512	Conv3-512
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 เป็นซอฟต์แวร์นึงที่สามารถเข้าถึงผ่าน browser โดยที่ผู้ใช้ไม่ต้องดาวน์โหลดก่อนใช้งานเพียงสามารถเข้าถึงได้ผ่าน network ซึ่ง Web Application นั้นออกแบบมาให้สามารถมีปฏิสัมพันธ์กับผู้ใช้ได้ไม่ใช่แค่การดูข้อมูลต่าง ๆ แต่สามารถจัดการหรือใช้งานได้ ซึ่งผู้ใช้หลายคนสามารถเข้าถึงพร้อมกัน

หลักการทำงานของ web Application คือ ผู้ใช้งานติดต่อสื่อสารกับ web application ผ่าน frontend หรือ client-side ซึ่ง frontend จะมีการเรียกใช้หรือดึงข้อมูลจาก 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 ซึ่งสามารถทำงานบนแพลตฟอร์ต่าง ๆ ได้หลากหลาย มีไวยากรณ์ที่เข้าใจง่ายสามารถเขียนได้ทั้งแบบเชิงวัตถุ 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 “<>” และมีชื่อยุ่งกลางเครื่องหมายนั้น ตัวอย่างเช่น <head> </head> เป็นต้น

### 2.2.7.3 Cascading Style Sheets (CSS)

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

### 2.2.7.4 JavaScript

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

### 2.2.7.5 Pytorch

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

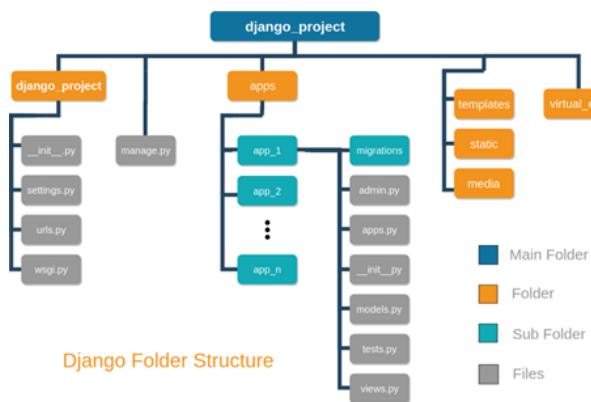
### 2.2.7.6 Django

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

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

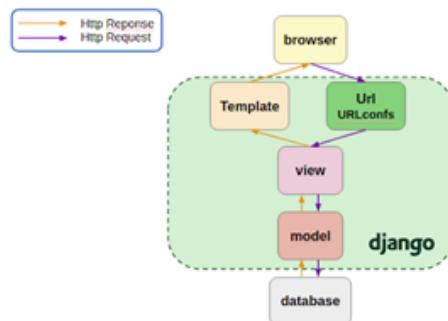
หลักการทำงานของ Django คือ 1 Project มี 1 app โดย 1 app คือ 1 module ในเว็บไซต์ ในการสร้าง Django Project [10] ผู้ใช้จะได้ไฟล์หลายไฟล์ดังรูปที่ 2.10 ซึ่งไฟล์หลัก ๆ ประกอบไปด้วย

1. manage.py ที่เป็นไฟล์ script สำหรับรันคำสั่งต่าง ๆ ที่เกี่ยวข้องกับ Django
2. \_\_init\_\_.py เป็นไฟล์ initial หรือไฟล์เปล่าที่ไม่สำหรับเก็บ package ของ python
3. settings.py เป็นไฟล์ที่ใช้ตั้งค่า project ตัวอย่างเช่น การตั้งค่า app, การตั้งเวลา, การตั้งค่า path เป็นต้น
4. urls.py เป็นไฟล์ที่มีไว้สำหรับเก็บ HTTP Request หรือ url pattern ที่เป็นการ routing ภายใน Django project
5. wsgi.py เป็นไฟล์ที่เก็บข้อมูลสำหรับการ deployment หรือการ production ให้ผู้ใช้งาน



รูปที่ 2.10 แสดงโครงสร้างไฟล์เดอร์ของ Django project

และ Django เป็น model-view-template (MVT) ซึ่งมีกระบวนการการทำงานหลัก ๆ 3 ส่วน ดังรูปที่ 2.11 คือ Model ที่เก็บข้อมูลของ application, View สำหรับประมวลผลคำสั่งหรือข้อมูลต่าง ๆ แล้วส่งไปที่ Template, Template คือหน้า application ที่ใช้แสดงผลข้อมูลที่ผ่านการประมวลผลจาก view ร่วงกับ html โดยมีการใช้ http request และ http response ในการติดต่อส่วนต่าง ๆ



รูปที่ 2.11 แสดงกระบวนการทำงานของ Django

#### 2.2.7.7 MongoDB

MongoDB เป็นฐานข้อมูลแบบ NoSQL ที่เป็น open source ชนิดหนึ่ง ซึ่งเก็บข้อมูลแบบไม่มีความสัมพันธ์กันซึ่งต่างจากการเก็บข้อมูลแบบ SQL ที่ร่วบทำให้ไม่จำเป็นต้องกำหนดโครงสร้างใด ๆ โดยจะมีโครงสร้างแบบ document database ที่เก็บข้อมูลในรูปแบบ JSON (JavaScript Object Notation) ซึ่งมี key และ value โดยจะเก็บข้อมูลใน collection ต่าง ๆ และเรียกแคล้วเป็น document และ colum เป็น field เนื่องจากไม่มีโครงสร้างใด ๆ ทำให้สามารถอ่านและเขียนได้อย่างรวดเร็ว

### 2.3 การทบทวนวรรณกรรม

งานวิจัยเรื่อง A Neural Algorithm of Artistic Style ของ Gatys et al.[1] ซึ่งทำการศึกษาการเอารูปภาพ (Content Image) ไปรวมกับรูปแบบ Artwork (Style Image) ของอีกภาพหนึ่ง โดยใช้ภาพ content Style จาก Neckarfront ในเมือง Tubingen ประเทศเยอรมัน และ style image จาก 5 ภาพของศิลปิน ดังนี้ ภาพ the shipwreck of the minotaur ของ J.M.W. Turner ภาพ The starry night ของ Vincent van Gogh ภาพ Der Schrei ของ Edvard Munch ภาพ Femme nue assise ของ Pablo Picasso และ ภาพ Composition ของ Wassily Kandinsky และใช้ VGG-network ที่มี 16 convolutional layers และ 5 pooling layers ของ VGG19 ในงานนี้ได้ใช้ full connected layer และเลือกใช้ average pooling แทนที่ max-pooling

งานวิจัยเรื่อง Stable and controllable neural texture synthesis and style transfer using histogram loss ของ Risser et al. ได้เพิ่มปัญหาจากวิธีของ Gatys et al. ที่มีข้อจำกัดในเรื่อง texture quality, stability, requisite parameter tunning และ lack of user control จึงเสนอให้ใช้ histogram loss แทนที่ gram loss ซึ่งพบว่าเมื่อใช้ histogram loss จะใช้ประมาณ 700 iteration ในการประมวลผลแต่ละภาพ (งานของ Gatys et al. ใช้ประมาณ 1000 iteration) แต่ภาพที่ได้จากการวิจัยนี้มีความ stable กว่า

งานวิจัยเรื่อง Two-stage color ink painting style transfer via convolution neural network ของ Zheng et al.[8] งานวิจัยนี้ ศึกษาวิธีการประมวลผลภาพวาดดอกไม้เป็นภาพวาดหมึกสี (color ink painting) ซึ่งเป็นภาพแบบดั้งเดิมของจีน (Traditional Chinese Painting) ซึ่งมีลักษณะการวาดที่แตกต่างกับภาพวาดทางตะวันตก งานวิจัยแบ่งวิธีการประมวลผลออกเป็น 2 งานย่อย คือ line drawing extraction และ image colorization โดยในส่วน line drawing extraction จะประมวลผลโดยใช้ CNN-based neural style transfer และในส่วนของ image colorization จะประมวลผลโดยใช้ GAN-based neural style transfer

## บทที่ 3 วิธีการดำเนินงาน

### 3.1 บทนำ

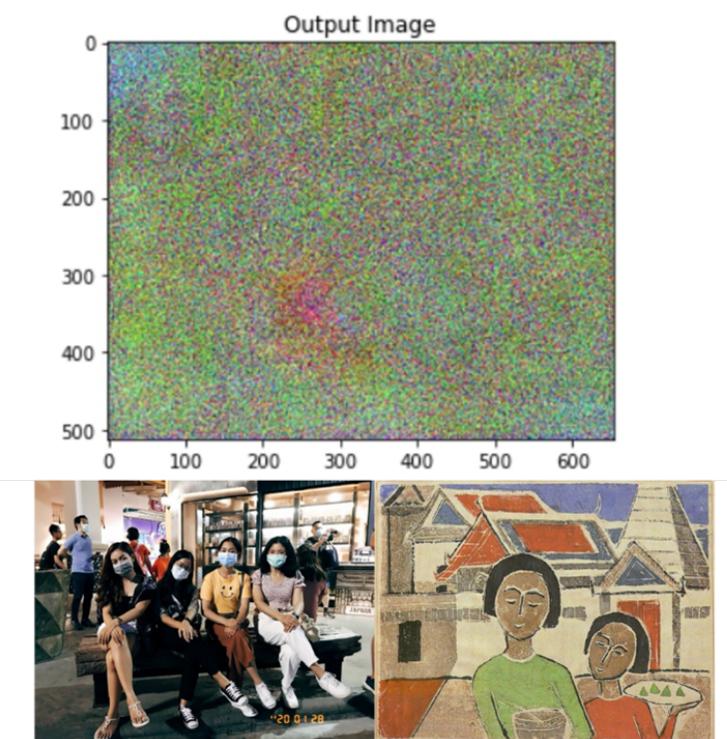
ในบทนี้ อธิบายเกี่ยวกับรายละเอียดของโครงงาน สถาปัตยกรรมที่เกี่ยวข้อง วิธีการจัดการข้อมูลว่า ข้อมูลที่นำมาใช้ในการทำโครงงานมีลักษณะอย่างไร มีการใช้อัลกอริทึมใดในการประมวลผลและแนวทางการประเมินผลการทำงาน

### 3.2 รายละเอียดของโครงงาน

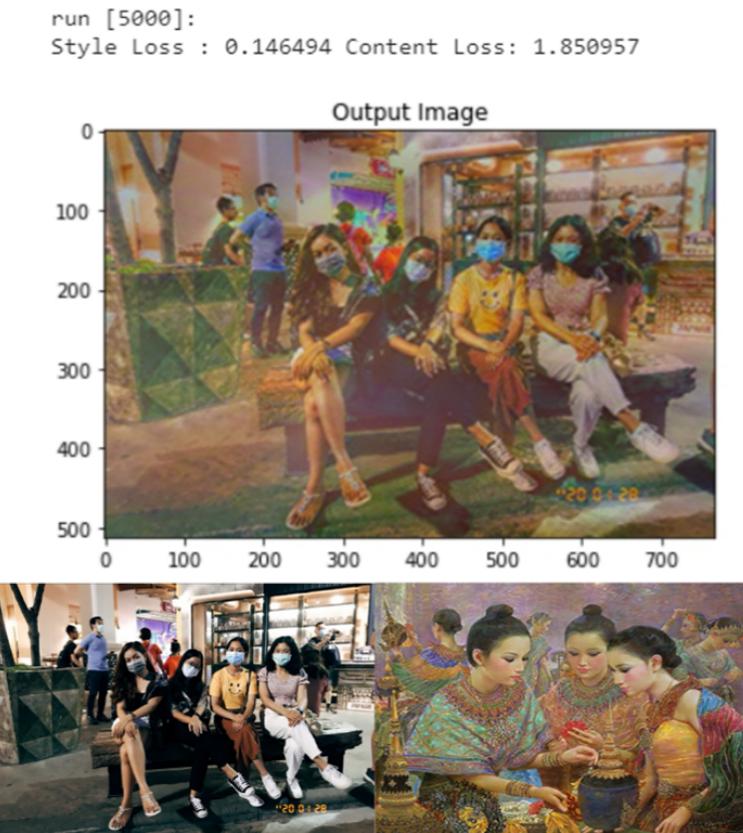
#### 3.2.1 การทบทวนวรรณกรรม

หลังจากการศึกษาจากการวิจัย คณบัญชัดทำได้คลอง download code ที่ implement จากมงานวิจัยของ Gatys et al. [7] มาศึกษาขั้นตอนวิธีการทำ neural style transfer และได้ทดลองนำภาพจากศิลปินไทย 2 ท่าน ท่านละ 1 style พับปัญหาจากวิธีการทำ neural style transfer แบบนี้คือ ขนาดของ style image และ content image ต้องมีขนาดที่เป็นความกว้างและความยาวเท่ากัน ระยะเวลาในการรันค่อยข้างนาน ใช้เวลา 45 นาที สำหรับ 5000 iteration บนคอมพิวเตอร์ intel(R) Core(TM) i5-8300H CPU @2.30GHz RAM 8 GB และ nvidia geforce gtx 1060 GPU 6 GB ผลลัพท์ในการประมวลแสดงดังภาพต่อไปนี้

```
run [5000]:  
Style Loss : 831907.562500 Content Loss: 90.193535
```



รูปที่ 3.1 (บน) แสดงผลการประมวลผล (ล่างซ้าย) content image (ล่างขวา) style image: จิตรกรรม ไปวัด ของชลุด นิ่มเสมอ



รูปที่ 3.2 (บน) แสดงผลการประมวลผล (ล่างซ้าย) content image (ล่างขวา) style image: ชื่อภาพ ผู้ก่อฯของจักรพันธุ์ โปษยกฤต

### 3.2.2 Modeling

สร้างโมเดลที่ใช้ input 2 ตัวคือ content image และ style image แล้วได้ output เป็นรูปภาพใหม่ที่รวมรูปแบบการวาดแบบ style image และมีเนื้อหาแบบ content image โดยได้พัฒนาโมเดลขึ้นมา ดังนี้

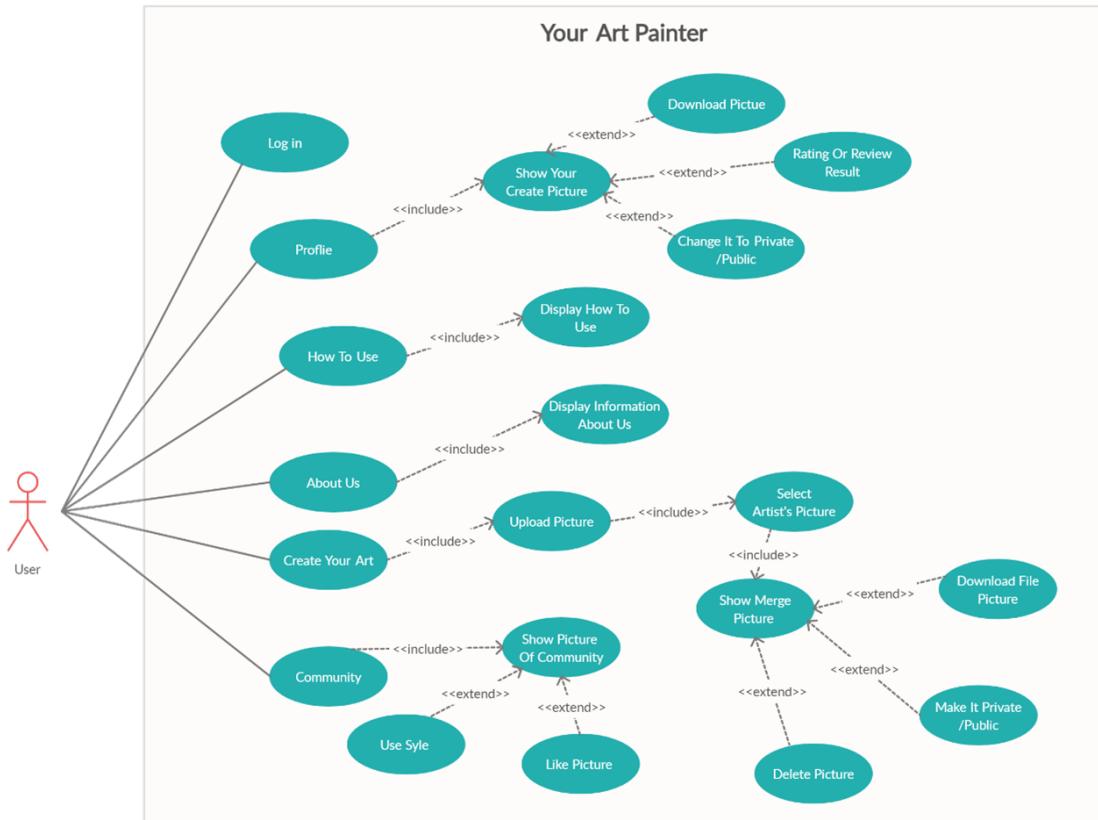
- ใช้โมเดล CNNs ในการแยกส่วนประกอบของรูปภาพ โดยทำ Transfer learning ด้วยการใช้ pre-trained โมเดล VGG19
- ปรับเปลี่ยนโมเดล VGG19 ที่เดิมใช้ pooling layer จาก max pooling เป็น average pooling เนื่องจากถ้าใช้ max pooling ผลลัพธ์ที่ออกมากอาจทำให้บางจุดในรูปภาพมีสีที่ได้เด่นเกินไปจึงใช้ average pooling แทน
- มีการตัด layer สุดท้ายที่ใช้ในการแบ่งคลาสของ VGG19 ออกเนื่องจากไม่ได้ใช้และเพิ่ม layer ที่เป็น Upsampling layer และ convolutional layer เพื่อทำให้รูปภาพผลลัพธ์ที่ได้ออกมามีขนาดใหญ่ขึ้น จากเดิมที่ได้ขนาด  $24 \times 24 \times 3$  เป็น  $224 \times 224 \times 3$  เมื่อขนาดรูปที่เป็น input
- มีการปรับ weight layer ที่ใช้โดยมีอ้างอิงกับโมเดลของ Gatys et al. [1] ที่ให้ weight ของ content layer ที่ layer เดียว (conv\_2\_2) และให้ weight ของ style layer 5 layers (conv 1\_1, conv 2\_1, conv 3\_1, conv 4\_1, conv 5\_1) จะทำการปรับเป็นให้ weight ของ content layer 2 layers (conv 2\_2, conv 4\_2) และให้ weight ของ style layer 5 layers เมื่อ้อนเดิม
- มีการปรับเปลี่ยนการใช้ gram matrices จากเดิมในโมเดลของ Gatys et al. [1] ที่ใช้เพียง 5 layers เป็นใช้ 16 layers ที่เป็น convolutional layer ทั้งหมดใน VGG19
- ทำการ train โมเดลโดยใช้ style image set ที่เป็นรูปภาพของ อ.จักรพันธุ์ โปษยกฤต และ อ.ชลุด นิ่มสมอและ content image set ที่เป็นรูปภาพคนหรือสัตว์ แต่เนื่องจากรูปภาพเหล่านี้มีขนาดที่แตกต่างกันทำให้ต้องทำ data preprocessing ปรับขนาดรูปภาพให้มีขนาดเดียวกันคือ  $224 \times 224 \times 3$  เพื่อที่จะสามารถนำไป train ในโมเดลได้

7. มีการคำนวณ loss function โดยคำนวณจาก loss function ของ image content loss และ image style loss และทำการ optimization ให้โมเดลมีประสิทธิภาพดีสุดโดยทำการ train หลาย ๆ รอบจนกว่าค่า loss ที่ได้จะมีค่าน้อยที่สุดที่เป็นไปได้
8. หลังจากทำสำเร็จแล้วรูปภาพใหม่แล้วจะทำ Color preservation โดยใช้ histogram matching เพื่อให้รูปภาพใหม่ที่ได้มีโทนสีเหมือนกับ content image

### 3.2.3 System Requirements สำหรับเว็บแอปพลิเคชัน

เว็บไซต์จะต้องสามารถสร้างรูปภาพใหม่ที่เกิดจากการผสมผสานระหว่างรูปภาพที่สร้างโดยศิลปินซึ่งดังกับรูปภาพที่ผู้ใช้ทำการอัปโหลดเข้าเว็บไซต์ โดยใช้หลักการ CNN ในการทำงาน ซึ่งจะมีแบบรูปภาพ (Style) ให้ผู้ใช้งานได้เลือกของศิลปิน 2 ท่าน ได้แก่ อ.จักรพันธุ์ โปษยกฤต และ อ.ชลุด นิ่มเสมอ โดยเว็บไซต์ ผู้ใช้งานสามารถแสดงผลลัพธ์จากการสร้างรูปภาพในเว็บไซต์ให้ผู้เข้าชมเว็บไซต์ท่านอื่น ๆ ให้ความคิดเห็นในรูปแบบการกดถูกใจได้

### 3.2.4 แผนผังแสดงความสัมพันธ์ (Use Case Diagram)



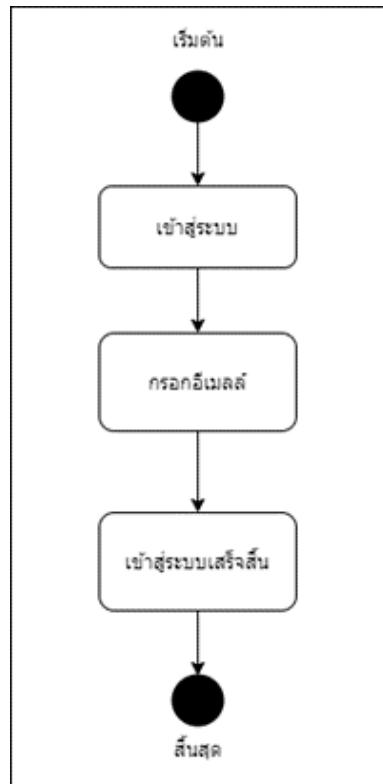
รูปที่ 3.3 แสดงภาพรวม Use Case Diagram

รูปที่ 3.3 แสดงความสัมพันธ์ระหว่างผู้ใช้งานกับตัวเว็บไซต์ โดยประกอบได้ด้วย 6 ระบบได้แก่

1. "Log In" - ผู้ใช้สามารถลงชื่อเข้าใช้เว็บไซต์ได้ โดยใช้ email ในการเข้าสู่ระบบ
2. "Profile" – แสดงรายละเอียดเกี่ยวกับผู้ใช้งาน เช่น email ที่ใช้ในการสมัคร, รูปภาพที่เคยทำการสร้างภายในเว็บไซต์ เป็นต้น
3. "How to use" – แสดงรายละเอียดเกี่ยวกับวิธีการใช้งานเว็บไซต์
4. "About us" – แสดงรายละเอียดเกี่ยวกับวัตถุประสงค์และผู้จัดทำเว็บไซต์
5. "Create your art" - ระบบสร้างรูปภาพใหม่
6. "Community" – ระบบชุมชนที่ช่วยให้ผู้ใช้งานท่านอื่นมาแสดงรูปภาพที่ตนเองสร้างขึ้นมาภายในเว็บไซต์

### 3.2.5 แผนภาพกิจกรรม (Activity Diagram)

#### 3.2.5.1 ระบบลงชื่อเข้าใช้งาน

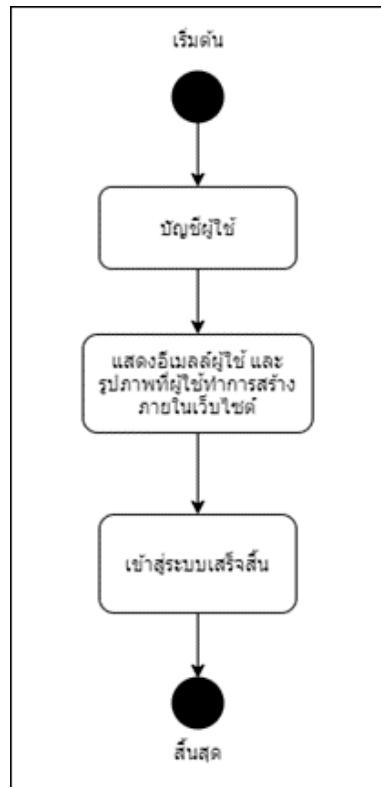


รูปที่ 3.4 แสดงระบบลงชื่อเข้าใช้งาน

ตารางที่ 3.1 ตารางอธิบายระบบลงชื่อเข้าใช้งาน

ตารางคำอธิบาย : ระบบการลงชื่อเข้าใช้งาน	
คำอธิบาย	การลงชื่อเข้าใช้งาน
จุดเริ่มต้น	ผู้ใช้งานเข้าถึง URL ใน การล็อกอิน ผ่านทางเว็บбраウเซอร์
เงื่อนไขตั้งต้น	-
เงื่อนไขหลังใช้	ผู้ใช้งานจะต้องเข้าสู่ระบบ จึงจะสามารถเข้าถึงฟังก์ชันการทำงานต่าง ๆ
ขั้นตอนหลัก	ผู้ใช้งานลงชื่อเข้าใช้โดยใช้ email ใน การเข้าสู่ระบบ ตัวระบบจะทำการดึงข้อมูลจากฐานข้อมูล ถ้าตัวระบบตรวจสอบว่า email ที่ผู้ใช้งานใช้เข้าสู่ระบบยังไม่มีในฐานข้อมูล ตัวระบบจะทำการสร้างบัญชีผู้ใช้ใหม่
ขั้นตอนพิเศษ	-

### 3.2.5.2 ระบบบัญชีผู้ใช้งาน

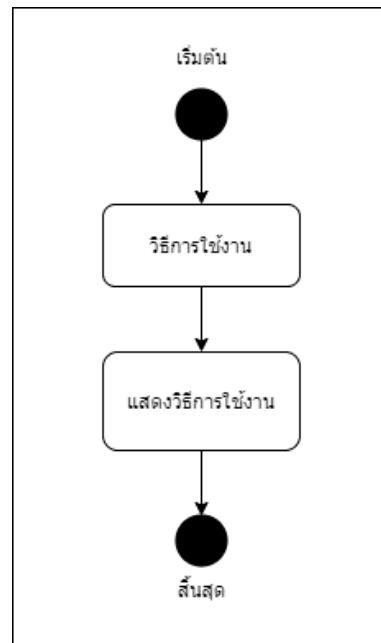


รูปที่ 3.5 แสดงระบบบัญชีผู้ใช้งาน

ตารางที่ 3.2 ตารางอธิบายระบบบัญชีผู้ใช้งาน

ตารางคำอธิบาย : ระบบบัญชีผู้ใช้งาน	
คำอธิบาย	ระบบบัญชีผู้ใช้งาน
จุดเริ่มต้น	ผู้ใช้กด บัญชีผู้ใช้งาน
เงื่อนไขตั้งต้น	ผู้ใช้งานต้องทำการลงชื่อเข้าใช้โดยใช้อีเมลในการเข้าใช้งาน
เงื่อนไขหลังใช้	ผู้ใช้งานสามารถเข้าสู่ระบบได้
ขั้นตอนหลัก	คลิก “Profile” เพื่อเข้าสู่หน้าบัญชีผู้ใช้งาน ตัวระบบจะทำการแสดงข้อมูลรายละเอียดต่าง เช่น อีเมลล์ที่ใช้งาน รูปภาพที่ผู้ใช้ทำการสร้างภายในเว็บไซต์ หรือให้รูปภาพที่สร้างมาแล้วแสดงในชุมชน หรือปิดการแสดงในชุมชน จำนวนยอดถูกใจแต่ละรูปภาพ ให้คะแนนความพึงพอใจกับผลลัพธ์รูปภาพหรือรายงานผลลัพธ์ให้กับผู้พัฒนา
ขั้นตอนพิเศษ	-

### 3.2.5.3 ระบบบริการใช้งาน

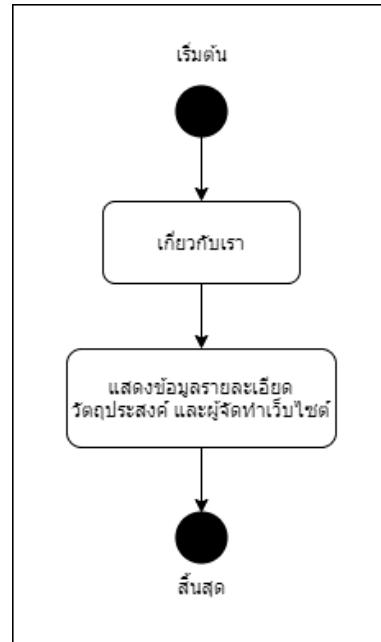


รูปที่ 3.6 แสดงระบบบริการใช้งาน

ตารางที่ 3.3 ตารางอธิบายระบบบริการใช้งาน

ตารางคำอธิบาย : ระบบบริการใช้งาน	
คำอธิบาย	ระบบบริการใช้งาน
จุดเริ่มต้น	ผู้ใช้กด วิธีการใช้งาน
เงื่อนไขตั้งต้น	-
เงื่อนไขหลังใช้	-
ขั้นตอนหลัก	กด “How to use ” เพื่อเข้าสู่หน้าแสดงวิธีการใช้งานเว็บไซต์ ตัวระบบจะแสดงขั้นตอนในการใช้งานเว็บไซต์
ขั้นตอนพิเศษ	-

### 3.2.5.4 ระบบเกี่ยวกับเรา

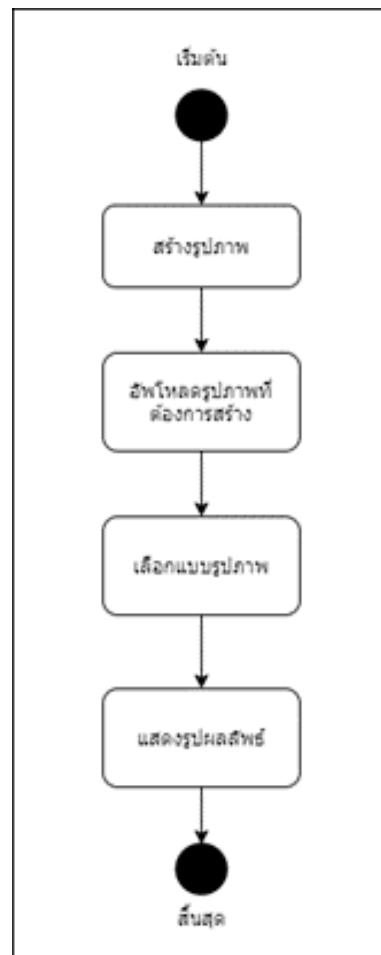


รูปที่ 3.7 แสดงระบบเกี่ยวกับเรา

ตารางที่ 3.4 ตารางอธิบายระบบเกี่ยวกับเรา

ตารางคำอธิบาย : ระบบเกี่ยวกับเรา	
คำอธิบาย	ระบบเกี่ยวกับเรา
จุดเริ่มต้น	ผู้ใช้กด เกี่ยวกับเรา
เงื่อนไขตั้งต้น	-
เงื่อนไขหลังใช้	-
ขั้นตอนหลัก	กด “About us” เพื่อเข้าสู่หน้าเกี่ยวกับเรา ตัวระบบจะแสดงข้อมูลรายละเอียดวัตถุประสงค์ และผู้จัดทำเว็บไซต์
ขั้นตอนพิเศษ	-

### 3.2.5.5 ระบบสร้างรูปภาพ

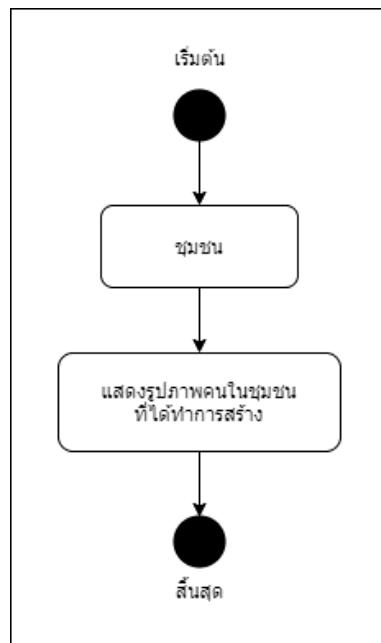


รูปที่ 3.8 แสดงระบบเกี่ยวกับเรา

ตารางที่ 3.5 ตารางอธิบายระบบสร้างรูปภาพ

ตารางคำอธิบาย : ระบบสร้างรูปภาพ	
คำอธิบาย	การสร้างรูปภาพจากภายในเว็บไซต์
จุดเริ่มต้น	ผู้ใช้กด สร้างรูปภาพ
เงื่อนไขตั้งต้น	-
เงื่อนไขหลังใช้	ผู้ใช้งานจะได้รับไฟล์ภาพ ที่ผู้ใช้ทำการสร้างรูปภาพขึ้นมา
ขั้นตอนหลัก	กด “Create your art” เพื่อเข้าสู่หน้าการสร้างรูปภาพ จากนั้นให้ผู้ใช้ทำการอัพโหลดรูปภาพที่ต้องการให้ เป็น content เพื่อทำการสร้างภาพใหม่ จากนั้นเลือก style image จากภาพของศิลปิน กด submit จากนั้นตัวระบบจะทำการสร้างรูปภาพใหม่ให้ผู้ใช้ โดยจะใช้เวลาประมาณ 10-15นาที ในการสร้างภาพ
ขั้นตอนพิเศษ	กรณีที่ผู้ใช้สร้างภาพใหม่เป็นภาพแรกในแต่ละครั้งของการเข้าใช้ ก่อนที่ผู้ใช้จะกด submit เพื่อสร้างภาพใหม่ได้นั้น ผู้ใช้จำเป็นต้องใส่อีเมล์ เพื่อเป็นการเข้าใช้งานเว็บไซต์

### 3.2.5.6 ระบบชุมชน



รูปที่ 3.9 แสดงระบบชุมชน

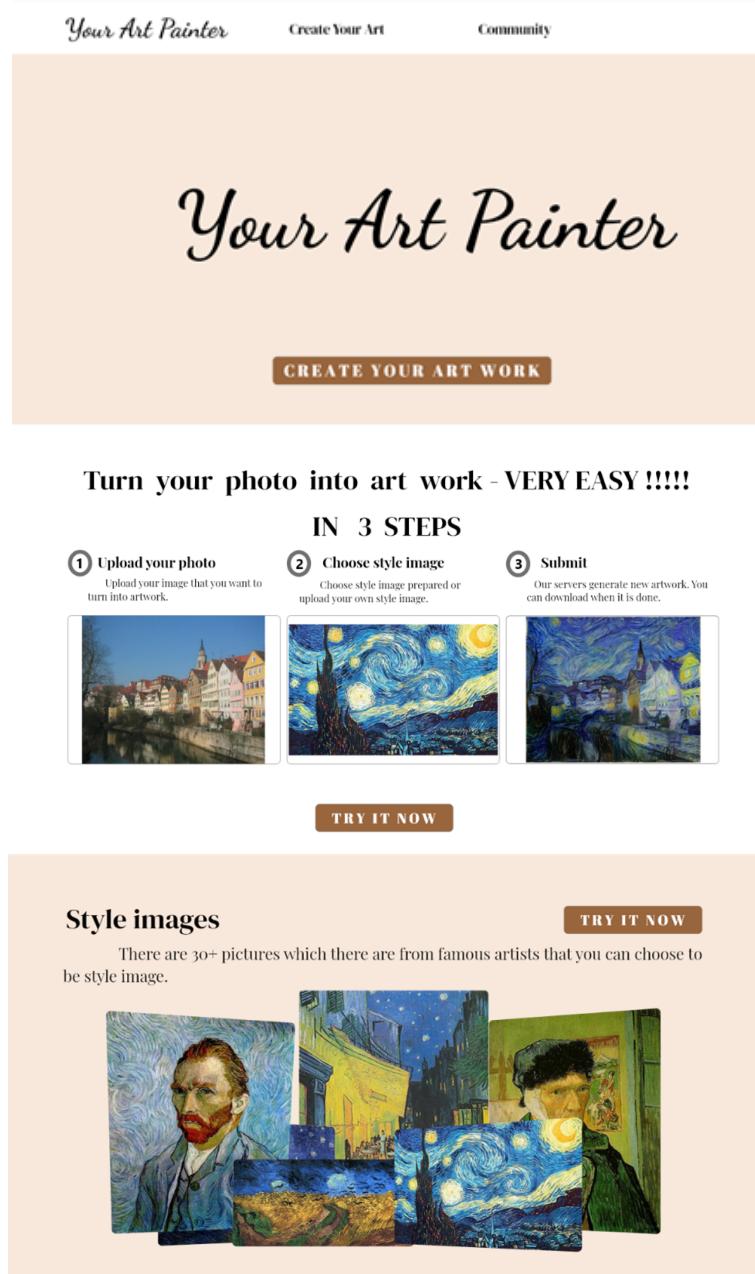
ตารางที่ 3.6 ตารางอธิบายระบบชุมชน

ตารางคำอธิบาย : ระบบชุมชน	
คำอธิบาย	ระบบชุมชนที่ให้ผู้ใช้งานเว็บไซต์แสดงผลงานให้เขียนชื่อเว็บไซต์
จุดเริ่มต้น	ผู้ใช้กด ชุมชน
เงื่อนไขตั้งต้น	-
เงื่อนไขหลังใช้	-
ขั้นตอนหลัก	กด “Community” เพื่อเข้าสู่หน้าชุมชน ตัวระบบจะทำการแสดงผลงานการสร้างรูปภาพภายในเว็บไซต์ ให้ผู้เขียนชื่อ ผู้ใช้สามารถกด ถูกใจ เพื่อแสดงความชื่นชอบแก่ผลงานของผู้ใช้ท่านอื่นๆ หรือเลือกใช้ style image แบบเดียวกับผู้ใช้งานท่านอื่นได้
ขั้นตอนพิเศษ	-

### 3.3 ไดอะแกรมสถาปัตยกรรมและคำอธิบาย

ใช้โมเดล VGGNet ที่มีอัลกอริทึม Convolutional neural network (CNN) อยู่เบื้องหลัง โดย VGG19 เป็น pre-train ซึ่งเป็นโมเดลที่มีคน train โมเดลด้วยรูปภาพมาศาลแล้วทำให้มันจำเป็นต้องมา train ใหม่เมื่อห้องหมวด ภายใน VGGNet มีโมเดลข้างใน 2 ตัวคือ VGG16 และ VGG19

#### 3.3.1 User interface design



## About us

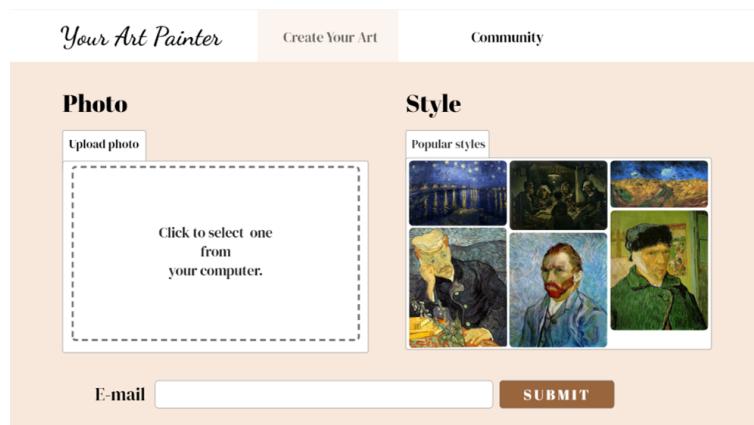
Our mission is to provide a intelligence painting tool that allows to create and share your own artwork with just a few steps.

We are one advisor and three members in 4th Computer engineering from King Mongkut's University of Technology Thonburi who develop this project.



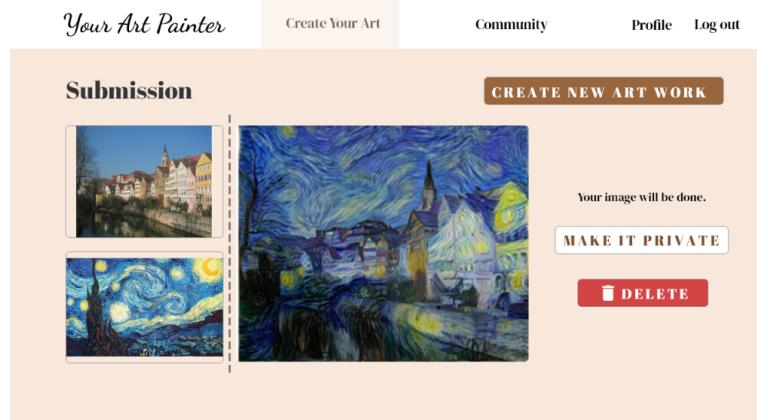
รูปที่ 3.10 แสดงหน้า Homepage

รูปที่ 3.10 แสดงหน้าแรกของ web application ซึ่งเป็นหน้าที่ผู้ใช้ทุกคนเห็นเมื่อเข้ามาในเว็บไซต์ ของเรา โดยกดแถบเมนูจะประกอบไปด้วยฟังก์ชัน create your art และ community ให้ผู้ใช้สามารถเข้าไปใช้งานได้ เนื้อหาที่แสดงในหน้า Homepage จะประกอบไปด้วยขั้นตอนการสร้างรูปภาพใหม่ style image และข้อมูลผู้พัฒนา web application นี้



รูปที่ 3.11 แสดงหน้า create your art ในขั้นตอนการเลือก content image และ style image

รูปที่ 3.11 เป็นหน้าที่แสดงเมื่อผู้ใช้คลิกมาใช้ฟังก์ชัน create your art ซึ่งผู้ใช้ต้องเลือกรูปภาพที่ต้องการเป็น content image ใน การสร้างภาพใหม่ (ด้านซ้าย) และเลือก style image จากภาพของศิลปิน (ด้านขวา) หากผู้ใช้สร้างภาพใหม่เป็นภาพแรกในแต่ละครั้งของ การเข้าใช้ ก่อนที่ผู้ใช้จะกด submit เพื่อสร้างภาพใหม่ได้นั้น ผู้ใช้จำเป็นต้องใส่ E-mail เพื่อเป็นการ log in เข้าใช้งาน web application ก่อน หากเป็นการสร้างรูปที่ 2 เป็นต้นไป ไม่จำเป็นต้องใส่ E-mail ใหม่อีกครั้ง



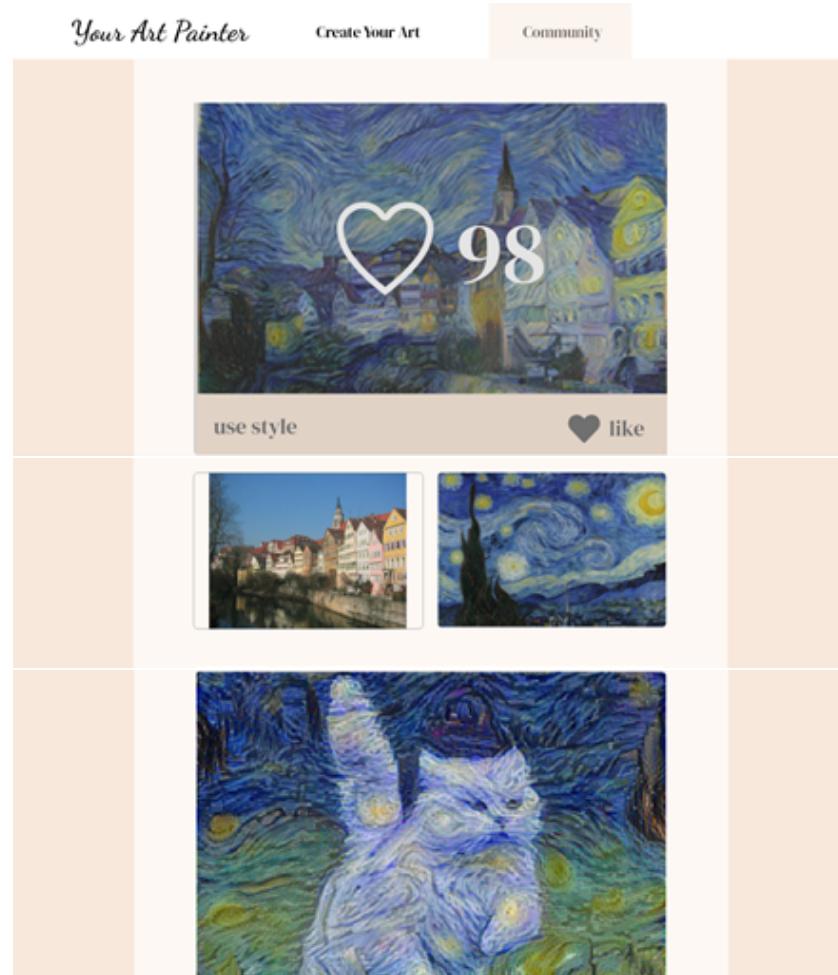
รูปที่ 3.12 แสดงหน้า create your art ในขั้นตอนการเลือก content image และ style image

รูปที่ 3.12 เป็นหน้าที่แสดงหลังจากที่ผู้ใช้กด submit เพื่อสร้างรูปใหม่แล้ว โดยแสดง content image ที่ผู้ใช้ upload มา style image ที่ผู้ใช้เลือก และภาพที่ได้หลังจากการประมวลผลเสร็จ หากผู้ใช้ต้องการลบภาพที่ผู้ใช้สร้างขึ้นมา ก็สามารถกดปุ่ม DELETE เพื่อลบได้ หลังจากการประมวลผล ภาพที่ได้จะถูกโพสต์ไปยัง Community อัตโนมัติหากผู้ใช้ไม่ต้องการโพสต์ลงใน Community ผู้ใช้สามารถกดปุ่ม MAKE IT PRIVATE ได้



รูปที่ 3.13 แสดงหน้า profile

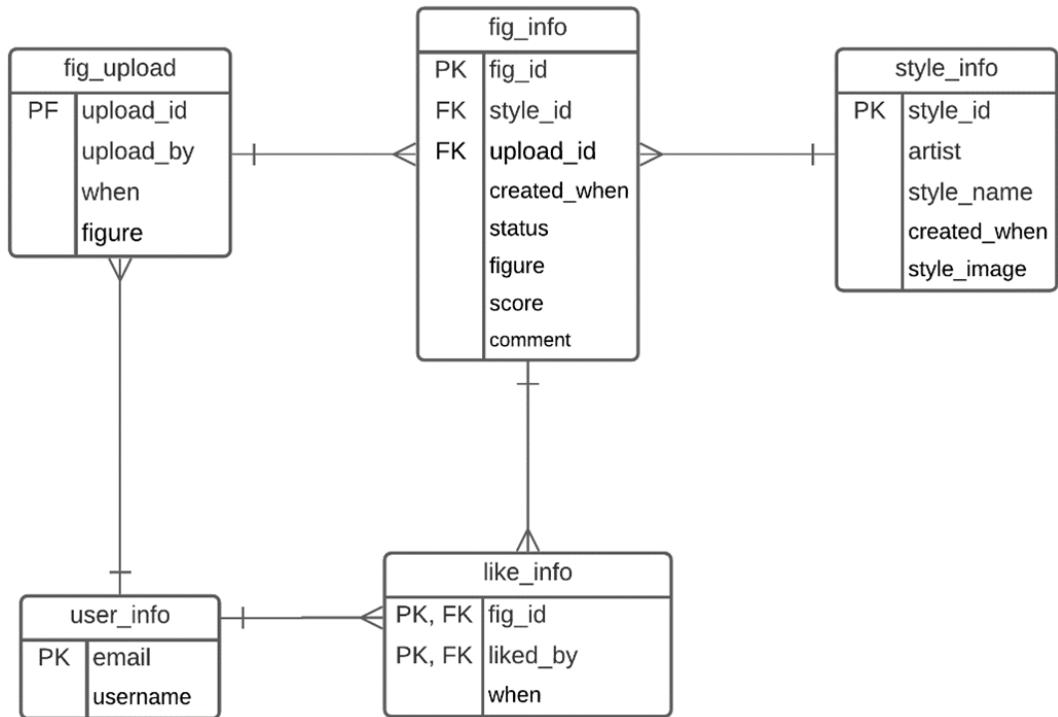
รูปที่ 3.13 แสดงหน้า profile ของผู้ใช้ โดยจะแสดง E-mail ของผู้ใช้ และรูปภาพที่ผู้ใช้ทำการสร้างขึ้น โดยประกอบไปด้วย content image, style image และรูปภาพที่ได้หลังจากการสร้างใหม่ อีกทั้งยังแสดงจำนวนคนที่กดถูกใจ ปุ่ม DOWNLOAD เพื่อให้ผู้ใช้ได้ดาวน์โหลดรูปภาพนั้น ปุ่มให้ผู้ใช้ตั้งค่าเป็นส่วนตัว (make it private หรือ make it public) สำหรับรูปภาพนี้ และปุ่ม DELETE เพื่อลบรูปภาพนั้น



รูปที่ 3.14 แสดงหน้า Community

รูปที่ 3.14 แสดงหน้าที่ผู้ใช้สามารถดูรูปที่มีการเผยแพร่ให้ผู้อื่นได้ดูภายใน web application ได้ รวมไปถึงสามารถกดถูกใจและสามารถเลือกใช้ style นั้นได้ถ้าหากผู้ใช้ชอบจากหน้านี้

### 3.3.2 Database



รูปที่ 3.15 แสดงหน้า Entity Relationship diagram ของ database

รูป 3.15 แสดงภาพรวมของ database ที่ใช้ในโปรเจคนี้ ซึ่งประกอบไปด้วย 5 ตาราง ซึ่งแต่ละตารางจะมีรายละเอียดดังตารางต่อไปนี้

ตารางที่ 3.7 แสดงรายละเอียดของตาราง user info

Entity Name	Attribute	Data Type	Constrain	Definition
user info	email	Varchar (50)	NOT NULL	อีเมลของผู้ใช้
	username	Varchar (50)	NULL	ชื่อที่แสดงใน web application

ตารางที่ 3.8 แสดงรายละเอียดของตาราง fig upload

Entity Name	Attribute	Data Type	Constrain	Definition
fig_upload	upload_id	Integer (10)	NOT NULL	Id ของ content image ที่ผู้ใช้เลือกใช้กับภาพนี้
	upload_by	Varchar (50)	NOT NULL	อีเมลของผู้ใช้ที่ upload รูปภาพนี้
	when	TIMESTAMP	NOT NULL	เวลาที่ upload รูปภาพ
	figure	GridFS	NOT NULL	รูปภาพ content ที่ผู้ใช้ upload

ตารางที่ 3.9 แสดงรายละเอียดของตาราง style\_info

Entity Name	Attribute	Data Type	Constrain	Definition
fig_info	fig_id	Integer (10)	NOT NULL	Id ของรูปภาพที่ผู้ใช้สร้างขึ้น
	style_id	Integer (10)	NOT NULL	Id ของ style image ที่ใช้เลือกมา ใช้กับภาพนี้
	upload_id	Integer (10)	NOT NULL	Id ของ content image ที่ผู้ใช้เลือก ใช้กับภาพนี้
	created_when	TIMESTAMP	NOT NULL	เวลาที่สร้างรูปภาพขึ้น
	status	Boolean	NOT NULL	รูปนั้นแสดงผลเป็น public หรือไม่
	figure	GridFS	NOT NULL	รูปภาพหลังจากสร้างขึ้นใหม่
	score	float	NONE	คะแนนความพึงพอใจของผู้ใช้ที่ให้ ต่อรูปภาพที่ประมวลผล
	comment	Varchar (100)	NONE	ความคิดเห็นต่อรูปภาพที่ประมวลผล ออกมานะ

ตารางที่ 3.10 แสดงรายละเอียดของตาราง style\_info

Entity Name	Attribute	Data Type	Constrain	Definition
style_info	style_id	Integer (10)	NOT NULL	Id ของ style image
	artist	Varchar (50)	NOT NULL	ชื่อศิลปินเจ้าของภาพ
	style_name	Varchar (50)	NOT NULL	ชื่อ style image
	created_when	TIMESTAMP	NOT NULL	เวลาที่มีการ upload style image นี้ขึ้นมา
	figure	GridFS	NOT NULL	รูปภาพ content ที่ผู้ใช้ upload

ตารางที่ 3.11 แสดงรายละเอียดของตาราง rating

Entity Name	Attribute	Data Type	Constrain	Definition
rating	rating_id	Integer (10)	NOT NULL	Id ของการให้คะแนนครั้งนั้น
	fig_id	Integer (10)	NOT NULL	Id ของรูปภาพที่สร้างขึ้นมา
	rated_by	Varchar (50)	NOT NULL	อีเมลของผู้ใช้ที่ให้คะแนน
	when	TIMESTAMP	NOT NULL	เวลาที่มีการให้คะแนนนี้ขึ้นมา
	score	Integer	NOT NULL	คะแนนที่ผู้ใช้ให้

ตารางที่ 3.12 แสดงรายละเอียดของตาราง like\_info

Entity Name	Attribute	Data Type	Constrain	Definition
Liked_info	fig_id	Integer (10)	NOT NULL	Id ของรูปภาพที่สร้างขึ้นมา
	liked_by	Varchar (50)	NOT NULL	อีเมลของผู้ใช้ที่ให้คะแนน
	when	TIMESTAMP	NOT NULL	เวลาที่มีการให้กดถูกใจนี้ขึ้นมา

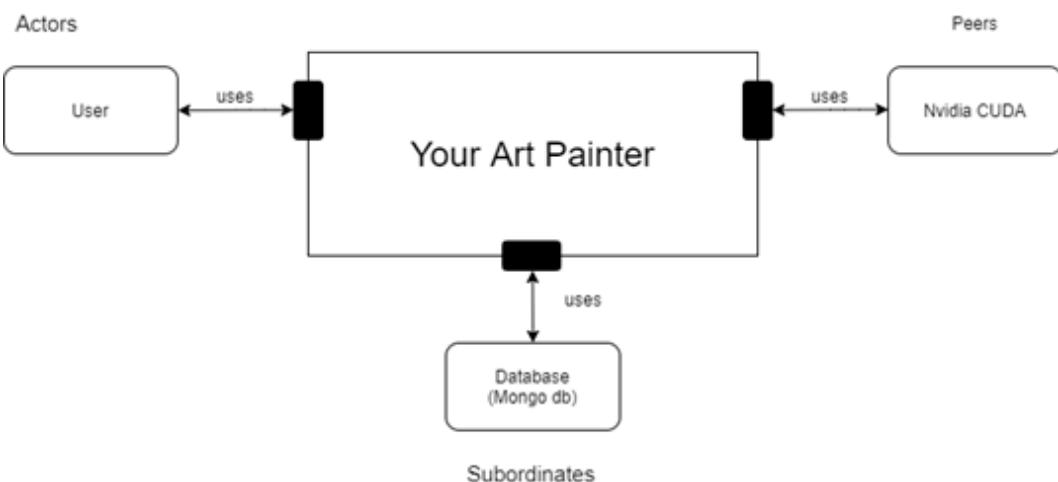
### 3.3.3 Data Management

ข้อมูลติดที่นำมาใช้มีลักษณะเป็นรูปภาพที่อยู่ในไฟล์ PNG เป็นหลักและมีสี 3 Channel (RGB) โดยจะแยกรูปภาพออกเป็น 2 แบบ คือ

1. Style image เป็นรูปภาพที่สร้างจากศิลปินที่มีชื่อเสียงโดยทางคณะผู้จัดทำเลือกร่วมรูปภาพที่มีลักษณะที่เด่นชัดสามารถบอกได้ว่า เป็นของศิลปินท่านไหนซึ่งเลือกศิลปินที่สนใจมา 2 ท่าน ได้แก่ อ.จักรพันธุ์ โปษยกฤต และ อ.ชลุด นิ่มสเมอ
2. Content image เป็นรูปภาพคนหรือสัตว์ทั่ว ๆ ไปที่ผู้ใช้ทำการอัพโหลด

### 3.3.4 Data structures/Class diagrams/ etc.

#### 3.3.4.1 Software Architectural Design



รูปที่ 3.16 แสดงโครงสร้างพื้นฐานของระบบซอฟต์แวร์ภายในเว็บไซต์ Your Art Painter

แสดงโครงสร้างพื้นฐานของระบบซอฟต์แวร์ภายในเว็บไซต์ Your Art Painter

#### Actor

1. User หรือผู้ใช้งานทั่วไป ที่เข้ามาใช้งานภายในเว็บไซต์

#### Peers

1. Nvidia CUDA คือแพลตฟอร์มสำหรับประมวลผลแบบคุ่นนานโดยใช้ Graphic Processing Unit (GPU) ในการประมวลผล เพื่อ-ใช้ในการประมวลผล VGG19

#### Subordinates

1. Mongo DB คือระบบฐานข้อมูลแบบ NoSQL ใช้สำหรับเก็บข้อมูลในระบบ

### 3.3.5 Evaluation plans

การประเมินผลในโครงการ แบ่งเป็น 2 ส่วนคือ ประเมินผลส่วนที่เป็นโมเดลโดยดูจากค่า loss function ที่เกิดขึ้น และเวลาให้การสร้างรูปภาพใหม่โดยเทียบกับโมเดลของ Leon A. Gatys ที่ผู้ที่ศึกษาเรื่อง Neural style transfer ใช้เป็นพื้นฐาน และอีกส่วนจะประเมินผลจากการให้บุคคลทั่วไปมาทดลองใช้งาน web application ของพวกเราเป็นจำนวน 20 คนโดยจะให้คะแนน และแสดงความคิดเห็น-เกี่ยวกับผลลัพธ์ของการสร้างรูปภาพว่าผู้ใช้รู้สึกอย่างไร

### 3.4 ภาคการเรียนที่ 2

ในภาคการเรียนที่ 2 ทางผู้จัดทำได้ดำเนินการตามแบบแผนที่ได้จัดวางไว้โดยแบ่งแยกงานได้เป็นดังนี้

#### 3.4.1 แผนดำเนินงาน

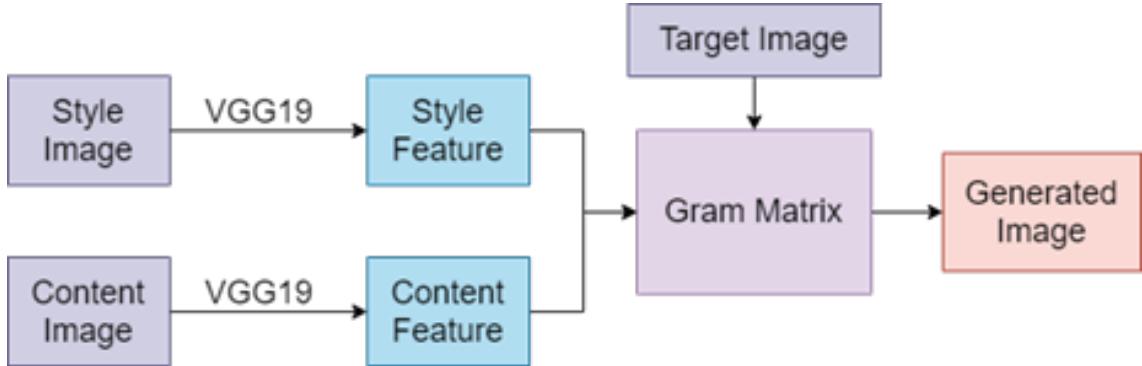
โดยทางผู้พัฒนาได้มีการปรับแก้ แผนการดำเนินงานในส่วนของแผนการทำงานภาคการศึกษาที่ 2 ดังนี้

Test/Week	Jan				Feb				Mar				Apr				ผู้รับผิดชอบ
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	
1. จัดการรูปภาพให้อยู่ในรูปแบบที่ไฟในโมเดล (Data preprocessing)	■	■															ก้าวพร นวฤทธิ์
2. ทดลองสร้างโมเดลและปรับแต่งให้มีประสิทธิภาพที่สุด																	
2.1 ทดลองสร้างโมเดลโดยใช้ parameter พื้นฐาน	■	■	■														ก้าวพร นวฤทธิ์
2.2 ปรับแต่งโมเดลด้วยการปรับค่าของ parameter ต่าง ๆ และวัดผลที่ได้ออกมา		■	■	■	■	■											ก้าวพร
2.3 เลือกโมเดลที่ได้ผลลัพธ์ที่ดีที่สุด						■	■										ก้าวพร
3. ทำให้โมเดลให้สามารถใช้งานได้ทั่วไปมากขึ้นโดยนำรูปภาพจากศึกปืนเข้ามา train โมเดลเพิ่มเติม								■	■	■	■						ก้าวพร นวฤทธิ์
4. ทดลองใช้งานโมเดลโดยนำรูปภาพคนหรือสัตว์ มา test โมเดลเพิ่มเติม									■	■	■	■					ก้าวพร นวฤทธิ์
5. ประเมินผลโมเดลที่ได้สร้างขึ้นมา										■	■	■					ทุกคน
6. พัฒนา Web Application ตามที่ได้ออกแบบไว้																	
6.1 Frontend	■	■	■	■	■												ก้าวพร นวฤทธิ์
6.2 Backend						■	■	■	■	■	■	■					สุกanya
7. นำเสนอโครงการ																	ทุกคน
8. จัดทำรายงานแสดงความคืบหน้า										■	■	■					ทุกคน
9. นำเสนอรายงานประจำการศึกษา													■	■	■		ทุกคน

ตารางที่ 3.13 ตารางแผนการทำงานภาคการศึกษาที่ 2 ใหม่

### 3.4.2 modeling

ในการสร้างรูปภาพใหม่ที่มีเนื้อหาของรูปภาพเหมือนกับรูปภาพ content image ที่เป็นภาพคนหรือสัตว์ที่ว่าไว้และมีลักษณะการวาด-เหมือนกับรูปภาพ style image ของ อ.จักรพันธุ์ โปษยกฤช และ อ.ชลุด นิ่มแสมอ ทางผู้พัฒนาได้นำวิธีการทำ Style Transfer มาใช้โดยมี-วิธีการทำงานดังรูปภาพที่ 3.17



รูปที่ 3.17 แสดงขั้นตอนการทำงานของ Style Transfer

เนื่องจากรูปภาพ content image และ style image ทางผู้พัฒนาได้นำมาจากการทดลองที่ทำให้ขนาดของรูปภาพมีความแตกต่างกัน จึงต้องทำการ transformation เพื่อแปลงรูปภาพให้มีขนาดที่ก่อนแล้วใช้โมเดล VGG19 ที่มีพื้นฐานมาจาก convolutional layer ซึ่งหมายความว่าการดึงคุณลักษณะของรูปภาพ ตัวอย่างเช่น สี ลายเส้น เป็นต้นโดยดึงคุณลักษณะของรูปภาพออกมายู่ในรูปแบบของค่าพิกเซล ของรูปภาพทั้ง style image และ content image จากนั้นทำการ train โดยกำหนดค่า weight ของ style feature และ content feature และใช้ gram matrix ในการจับคู่คุณลักษณะของรูปภาพที่ได้จาก style feature และ content feature เข้ากับ target image และทำการ train หลายรอบและใช้ optimizer Adam ที่มี learning rate เท่ากับ 0.03 เพื่อให้ได้ generated image ที่มีเนื้อหาของรูปภาพเหมือนกับรูปภาพ content image ที่เป็นภาพคนหรือสัตว์ที่ว่าไว้และมีลักษณะการวาดเหมือนกับรูปภาพ style image ตามที่ต้องการ โดยทางผู้พัฒนาได้ทำการทดลองหลายรูปแบบเพื่อหาวิธีที่เหมาะสม ดังนี้

1. ในขั้นตอนของการดึงคุณลักษณะของรูปภาพ โมเดล VGG19 นั้นมี pooling layer คือ max pooling หากผู้พัฒนาได้ตั้งสมมติฐานว่า ผลลัพธ์ที่ได้ออกมาจะมีบางจุดที่มีลักษณะเด่นเกินไปจึงทำการปรับเปลี่ยนจาก max pooling เป็น average pooling โดยกำหนดขนาดของ style image และ content image คือ 224\*224 และทำการ train 5,000 รอบ
2. จากการศึกษาจากการวิจัย พบว่า target image ที่นำมาใช้มี 2 รูปแบบ จึงได้ทำการทดลองทั้ง 2 แบบคือ target image เป็น noise image และ target image เป็นภาพที่มาจากการคัดลอก content image ก่อนนำไปรวมกับคุณลักษณะต่าง ๆ ที่ดึงออกมาใน gram matrix โดยกำหนดขนาดของ style image และ content image คือ 224\*224 ใช้โมเดล VGG19 ที่เป็น max pooling และทำการ train 5,000 รอบ
3. ปรับเปลี่ยนขนาดของ style image เนื่องจากทางผู้พัฒนาเห็นว่า ขนาดของ style image ส่งผลต่อการดึงคุณลักษณะของรูปเพื่อทำการสร้างรูปภาพใหม่ โดยทำการปรับขนาดของรูปภาพ style image เป็น 128\*128, 224\*224, 512\*512 และ 1024\*1024 โดยกำหนดขนาดของ content image คือ 224\*224 ใช้โมเดล VGG19 ที่เป็น max pooling และทำการ train 5,000 รอบ
4. ปรับเปลี่ยนจำนวนรอบที่ใช้ในการ train เพื่อสร้างรูปภาพใหม่ โดยกำหนดเป็น 100, 500, 1,000, 5,000, 10,000, 15,000, 20,000 และ 25,000 รอบ เนื่องจากจำนวนรอบมีผลต่อลักษณะของรูปภาพที่สร้างขึ้น โดยกำหนดขนาดของ style image และ content image คือ 224\*224
5. ปรับเปลี่ยน style weight และ content weight เนื่องจากค่า weight ที่ใช้ส่งผลต่อลักษณะของรูปภาพที่สร้างขึ้น จึงได้ทำการปรับเปลี่ยนเพื่อหาค่าที่เหมาะสม

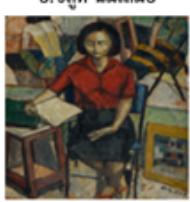
## บทที่ 4 ผลการดำเนินงาน

### 4.1 Modeling

จากที่ได้ทำการทดลองตามที่ได้ออกแบบไว้ในบทที่ 3 คือ นำรูปภาพ content image ที่เป็นภาพคนหรือสัตว์และ style image ที่เป็นรูปภาพของศิลปิน อ.จักรพันธุ์ ไปชัยกฤช และ อ.ชุด นิ่มเสมอ ผ่านโมเดล VGG19 เพื่อดึงคุณลักษณะของรูปภาพแล้วมาผ่านการทำ gram matrix เมื่อสร้างรูปภาพใหม่ที่สมาระห่าง style image และ content image โดยได้ผลการทดลองจากการปรับเปลี่ยนค่าต่าง ๆ ดังนี้

#### 4.1.1 ปรับเปลี่ยนโมเดล VGG19 จาก max pooling เป็น average pooling

โดยกำหนดขนาดของ style image และ content image คือ  $224 \times 224$  และทำการ train 5,000 รอบ พบรูปภาพที่ได้จากทั้ง max pooling และ average pooling นั้นสามารถสร้างรูปภาพใหม่ที่มีการสมาระห่าง style image และ content image และเมื่อพิจารณารูปภาพที่ได้จาก average pooling นั้นเห็นความซ้ำเจนของ style image ในรูปที่สร้างขึ้นมากกว่ารูปภาพที่ได้จาก max pooling ดังรูปภาพที่ 4.1

		Max Pooling	Average Pooling
Style Image ของ อ.ชุด นิ่มเสมอ			
Style Image ของ อ.จักรพันธุ์ ไปชัยกฤช			

รูปที่ 4.1 แสดงผลลัพธ์ที่ได้จากการเปลี่ยน pooling layer ใน VGG19

#### 4.1.2 ทำการทดลองโดยใช้ target image เป็น noise image และ target image เป็นภาพที่มาจากการคัดลอก content image

โดยกำหนดขนาดของ style image และ content image คือ  $224 \times 224$  ให้โมเดล VGG19 ที่เป็น max pooling และทำการ train 5,000 รอบ พบรูปภาพที่เป็น noise image นั้นไม่สามารถสร้างรูปภาพใหม่ที่มีการสมาระห่าง style image และ content image ได้ ดังรูปภาพที่ 4.2 อาจเนื่องจากการความผิดพลาดที่เกิดการวิธีการรวมคุณลักษณะของ style image และ content image ในขั้นตอนการเขียน code ก็เป็นได้

		target image เป็น noise image	target image เป็นภาพ จาก content image
Style Image ของ อ.ชูศร ปั้นเนมอ	Content Image		
Style Image ของ อ.จักรพันธุ์ โปษยกฤต			

รูปที่ 4.2 แสดงผลลัพธ์ที่เกิดจากการใช้ target image ที่แตกต่างกัน

#### 4.1.3 ปรับเปลี่ยนขนาดของ style image

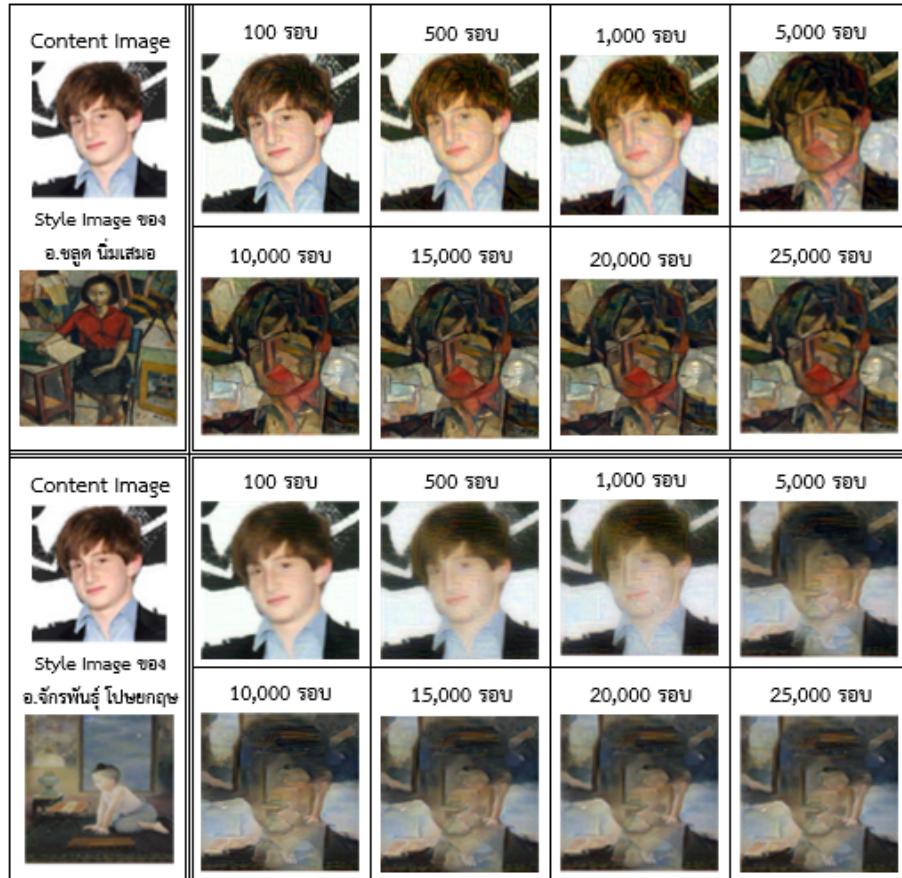
โดยได้ทำการปรับเปลี่ยน style image เป็น 128\*128, 224\*224, 512\*512 และ 1024\*1024 โดยกำหนดขนาดของ content image คือ 224\*224 ใช้โมเดล VGG19 ที่เป็น max pooling และทำการ train 5,000 รอบ พบว่าการใช้ style image ขนาด 224\*224 มีความหมายมากที่สุดเมื่อพิจารณาจากผลลัพธ์ที่ได้ดังรูปที่ 4.3

Content Image	style image เป็น 128*128	style image เป็น 224*224	style image เป็น 512*512	style image เป็น 1024*1024
Style Image ของ อ.ชูศร ปั้นเนมอ				
Style Image ของ อ.จักรพันธุ์ โปษยกฤต				

รูปที่ 4.3 แสดงผลลัพธ์ที่เกิดจากการปรับขนาดรูปภาพ style image

#### 4.1.4 ปรับเปลี่ยนจำนวนรอบที่ใช้ในการ train เพื่อสร้างรูปภาพใหม่

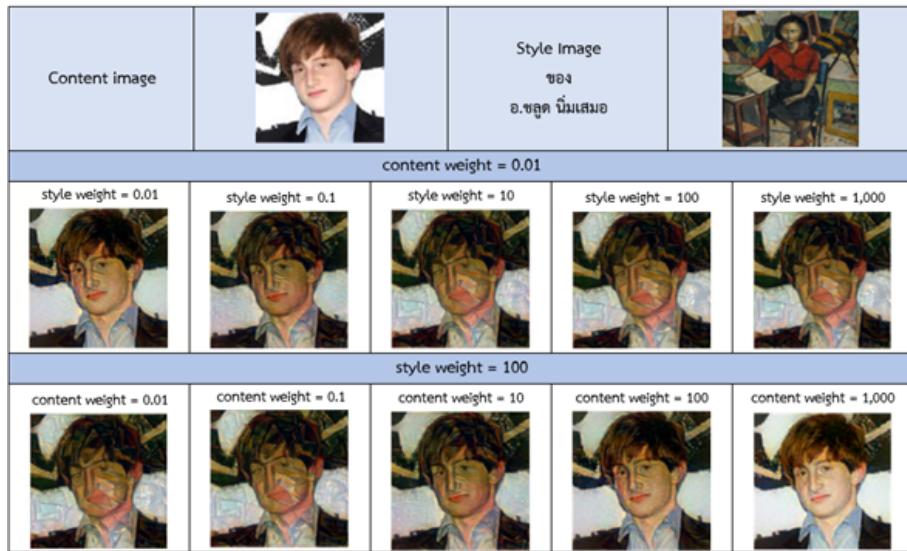
โดยได้ทำการปรับเปลี่ยนจำนวนรอบที่ใช้ในการ train โดยกำหนดเป็น 100, 500, 1,000, 5,000, 10,000, 15,000, 20,000 และ 25,000 รอบ โดยกำหนดขนาดของ style image และ content image คือ  $224 \times 224$  พบร้า เมื่อเพิ่มจำนวนรอบ ทำให้รูปภาพใหม่ที่ได้มีการผสมระหว่าง style image และ content image ที่มากขึ้นแต่มีเกิน 10,000 รอบขึ้นไป รูปภาพใหม่ที่ได้เริ่มไม่มีการเปลี่ยนแปลงตั้งรูปภาพที่ [4.4](#) ดังนั้นจำนวนรอบที่เหมาะสมคือ ประมาณ 5,000 รอบ



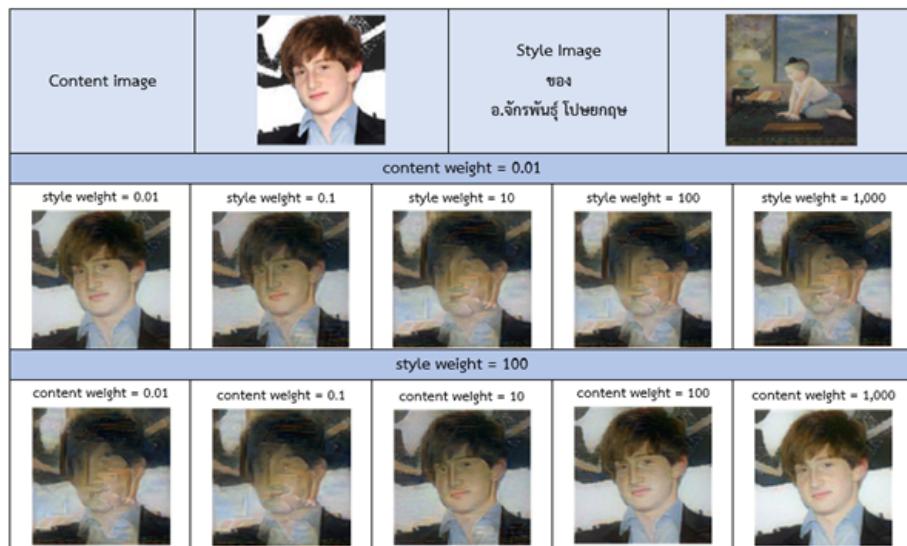
รูปที่ 4.4 แสดงผลลัพธ์ที่เกิดจากการเพิ่มจำนวนรอบในการ train เพื่อสร้างรูปภาพใหม่

#### 4.1.5 ปรับเปลี่ยน style weight และ content weight

เนื่องจากค่า weight ที่ใช้ส่งผลต่อถักขยันของรูปภาพที่สร้างขึ้นจึงได้ทำการปรับเปลี่ยนเพื่อหาค่าที่เหมาะสม โดยกำหนดขนาดของ style image และ content image คือ  $224 \times 224$  ใช้โมเดล VGG19 ที่เป็น max pooling และทำการ train 5,000 รอบ จากการทดลองพบว่า style image ที่เป็นรูปภาพของ อ.ชลุด นิ่มเสมอ ค่า style weight เท่ากับ 100 และ content weight เท่ากับ 0.01 สามารถให้รูปภาพใหม่ที่ได้มีการผสมระหว่าง style image และ content image เหมาะสมที่สุด เมื่อเปรียบเทียบกับ weight ค่าอื่น ๆ ดังรูปภาพที่ [4.5](#) และ style image ที่เป็นรูปภาพของ อ.จักรพันธุ์ โปษยกฤต ค่า style weight เท่ากับ 100 และ content weight เท่ากับ 10 สามารถให้รูปภาพใหม่ที่ได้มีการผสมระหว่าง style image และ content image เหมาะสมที่สุด ดังรูปภาพที่ [4.6](#)



รูปที่ 4.5 แสดงผลลัพธ์ที่เกิดจากการปรับ weight โดยใช้ style image เป็นรูปภาพของ อ.ชลุด นิมเสนอ

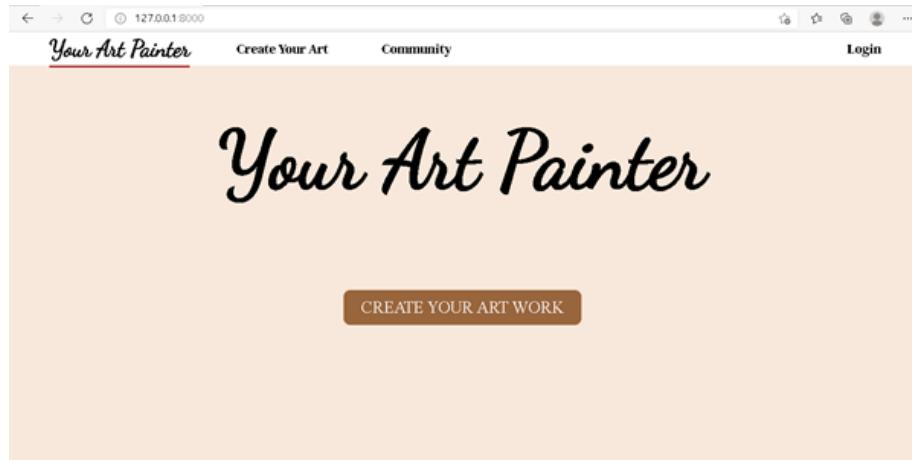


รูปที่ 4.6 แสดงผลลัพธ์ที่เกิดจากการปรับ weight โดยใช้ style image เป็นรูปภาพของ อ.จักรพันธุ์ โปษยกฤต

## 4.2 WEB APPLICATION

### 4.2.1 User Interface

- 4.2.1.1 Homepage
- 4.2.1.2 Create your art Page
- 4.2.1.3 Community Page
- 4.2.1.4 Login Page
- 4.2.1.5 Register Page



## Turn your photo into art work - VERY EASY!!!!

### IN 3 STEPS

#### Upload your photo

Upload your image that you want to turn into artwork.

Step 1

#### Choose style image

Choose style image prepared or upload your own style image. Our servers generate new artwork. You can download when it is done.

Step 2

#### Submit

Our servers generate new artwork. You can download when it is done.

Step 3

### Style image

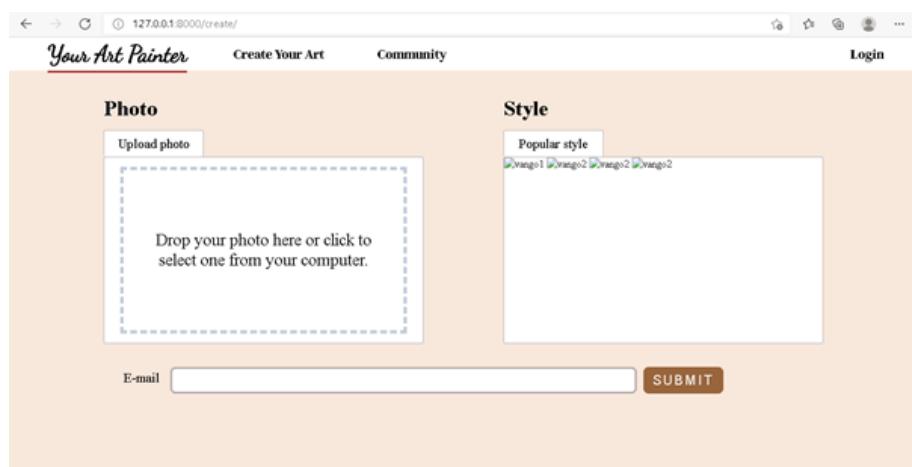
There are 30+ pictures which there are from famous artists that you can choose to be style image.

### About us

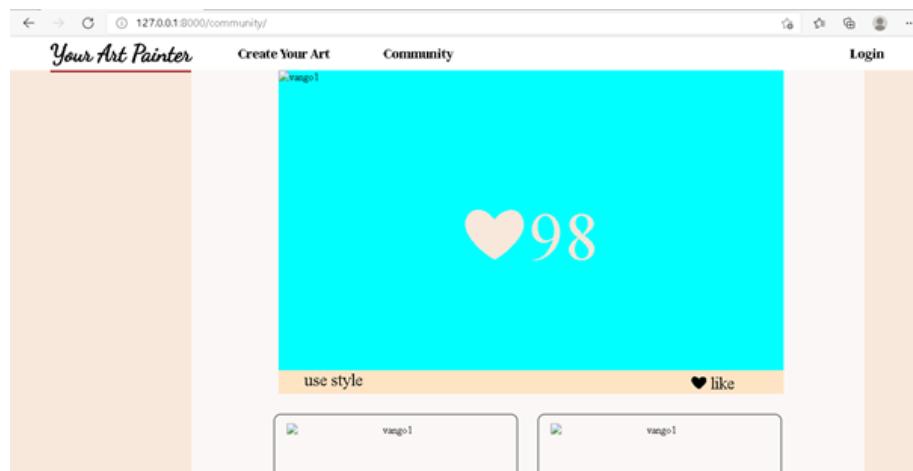
Our mission is to provide a intelligence painting tool that allows to create and share your own artwork with just a few steps.

We are one advisor and three members in 4th Computer engineering from King Mongkut's University of Technology Thonburi who develop this project.

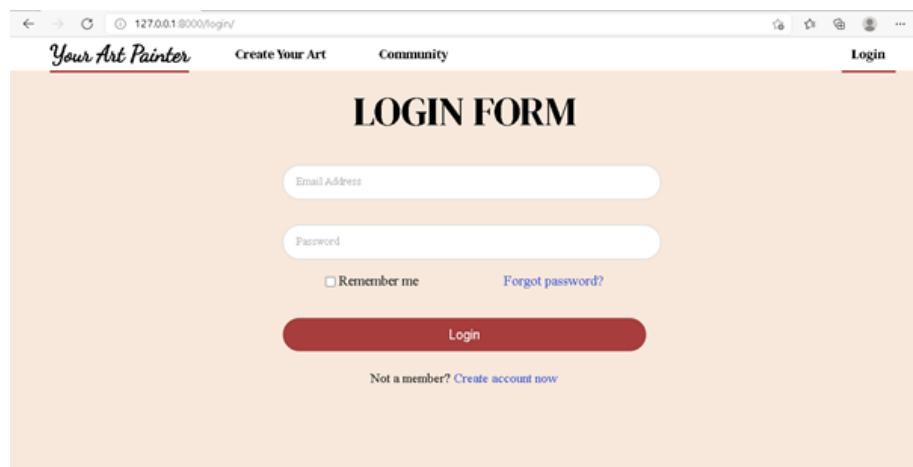
รูปที่ 4.7 แสดงหน้า Homepage



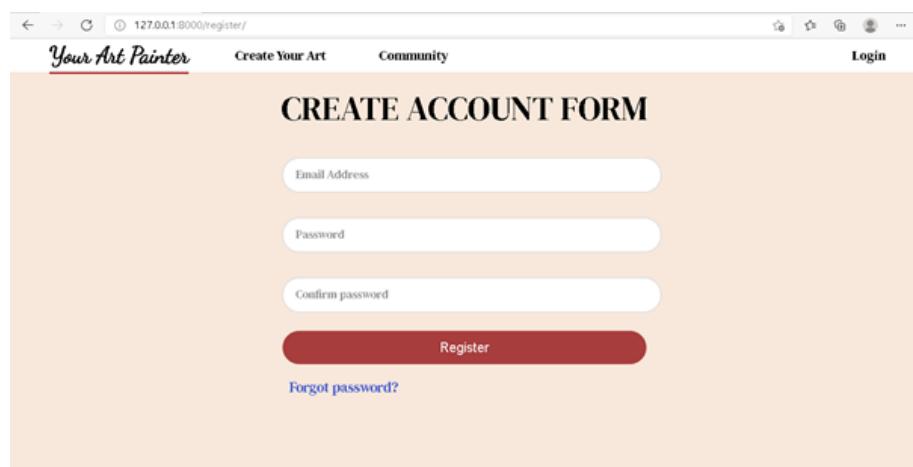
รูปที่ 4.8 แสดงหน้า Create your art



รูปที่ 4.9 แสดงหน้า Community

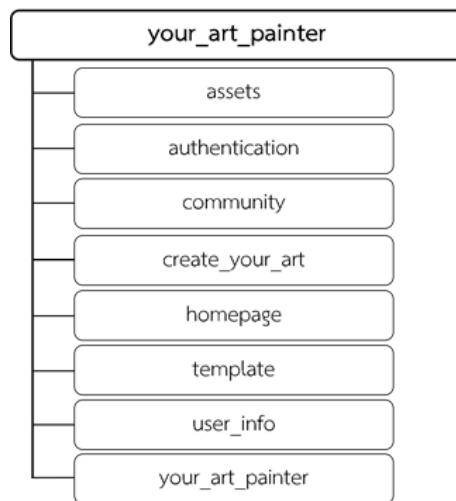


รูปที่ 4.10 แสดงหน้า Login



รูปที่ 4.11 แสดงหน้า Register

#### 4.2.2 Backend และ Database



รูปที่ 4.12 แสดง App (module) ในส่วนของ Backend

- 4.2.2.1 Module assets: เก็บข้อมูล static (image, css) ที่ถูกรวบรวมมาจาก directory ต่าง ๆ
- 4.2.2.2 Module authentication: เป็น module ที่สำหรับการ login, register
- 4.2.2.3 Module community: ใช้สำหรับแสดงผลภาพที่มีคนโพสต์
- 4.2.2.4 Module create\_your\_art: ใช้สำหรับเก็บประมวลผล Neural style transfer
- 4.2.2.5 Module Homepage: ใช้สำหรับแสดงผลหน้า homepage
- 4.2.2.6 Module template: ใช้สำหรับเก็บ fronten ที่เป็น .html
- 4.2.2.7 Module user\_info: ใช้สำหรับหน้า profile
- 4.2.2.8 Module your\_art\_painter: เป็น module หลักในการ setting ค่าต่าง ๆ

## บทที่ 5 บทสรุป

ผลสรุปของการทำโครงการ ปัญหาที่พบรหว่างการดำเนินงานและแนวทางการแก้ไข ข้อบกพร่อง ข้อจำกัดของเครื่องมือที่พัฒนาและแนวทางการพัฒนาโครงการ

### 5.1 ผลการดำเนินงานในภาคการศึกษาที่ 1 (Midterm)

ในเริ่มแรกทางผู้พัฒนาตั้งข้อกังวลของโครงงานนี้คือการสร้างรูปภาพใหม่โดยการเอา Source Code ที่มีการนำ CNN ในการประมวลผลมาใช้ หลังจากนั้นทางผู้พัฒนาจะปรับปรุงการ Pre-train ขึ้นมาใหม่ให้มีความเหมาะสมมากยิ่ง โดยจะมีแบบ 3 ศิลปิน ได้แก่ 1.โกลด์มอนแคน 2.ปาเบโล รุยซ์ ปิกasso และ 3.วินเซนต์ วน โถะ ศิลปินละ 5 style ซึ่งหลังจากนำเสนอ Proposal presentation ได้รับข้อเสนอแนะ ว่าโครงการของผู้พัฒนาเป็นงานที่น้อยเกินไปสำหรับสมาชิก 3 คน ทางผู้พัฒนาจึงได้ทำการเปลี่ยนขอบเขตการทำโครงงานนี้ใหม่ โดยเปลี่ยนมาเป็นทำเว็บไซต์ในการสร้างรูปภาพขึ้นมาใหม่โดยใช้ CNN ในการทำงานโดยมีแบบภาพวด 3 ศิลปิน เช่นเดิม

แต่เมื่อทางผู้พัฒนาทำการศึกษาโครงงานที่มีเนื้อหาคล้ายคลึงกับโครงการนี้ ก็มีความคิดที่จะเปลี่ยนการสร้างรูปภาพจากการใช้ CNN ในการสร้างรูปภาพ เปเลี่ยนมาเป็นใช้ Generative Adversarial Networks หรือ GANs เหตุผลที่ทำการเลือก GANs ใน การพัฒนา เพราะ GANs จะมีปัญญาประดิษฐ์ (AI) 2 ตัว ได้แก่ "Discriminator" โดยหน้าที่ของเจ้า discriminator จะทำการ classify งานที่เราต้องการให้มันทำ ยกตัวอย่าง เช่น เราอยากรู้ว่ามัน classify ว่ารูปภาพไหนเป็นรูปตัวเลขารabic ส่วน AI ตัวที่สองจะเรียกว่า "Generator" โดยมันจะทำหน้าที่สร้างข้อมูลใหม่ที่มีลักษณะใกล้เคียงกับตัวเลขารabic ซึ่งในแต่ละรอบของการทายทาง discriminator จะบอกคำตอบที่ตัวเองทายให้ generator ทราบในขณะที่ทาง generator จะให้ feedback กลับมาว่าทาง discriminator ทายถูกหรือไม่ โดยจะทำแบบนี้ต่อไปเรื่อยจน generator ไม่สามารถปั้บค่าในการสร้างรูปภาพใหม่ได้ และจนที่สุดท้ายจะได้วาไฟล์ที่เกิดจากการสร้างขึ้นมาใหม่ที่แนบเนียนสมจริงได้ตีกันว่า ผลลัพธ์ของ CNN ที่รูปภาพออกมาจะมีความเป็น Filter รูปภาพเสี่ยมากกว่าการสร้างรูปภาพ

แต่เมื่อได้ทำการศึกษาทดลองด้วย Source Code ที่ใช้หลักการ GANs ในการพัฒนา Neural Style Transfer ทางผู้พัฒนาพบปัญหาเรื่องของ library ที่ทาง Source Code แต่ละงานวิจัยได้ทำการพัฒนาล้วนใช้ tensorflow พัฒนาเป็นหลัก ซึ่งปัญหาที่ทางผู้พัฒนาพบคือเป็น Source Code ที่มีการพัฒนา 2-3 ปีมาแล้ว ซึ่ง library tensorflow ที่ใช้งานวิจัยเหล่านั้นเป็นเวอร์ชันก้า ทางผู้พัฒนาไม่สามารถโหลดเวอร์ชันที่สามารถรัน tensorflow ได้ ทั้งเวอร์ชัน 1.13 1.14 1.15 และเวอร์ชัน 2.0 ทางผู้พัฒนาจึงได้ประชุมกันภายในกลุ่มเพื่อหาข้อสรุปของเขตของโครงงานนี้ และจึงได้ข้อสรุปว่า จัดทำ web application ที่สามารถทำ Neural Style Transfer โดยจะใช้ CNN ในการพัฒนาและมีแบบสตูดี้รูป 2 แบบ คือ อ.จักรพันธุ์ ปะยิกฤษ และ อ.ชลุด นิ่มเสมอ โดยไม่เดลที่จะนำมายังทางผู้พัฒนาทำการพัฒนาขึ้นมาเอง

## 5.2 ผลการดำเนินงานในภาคการศึกษาที่ 2 (Final)

ในการดำเนินงานในภาคการศึกษาที่ 2 ทางผู้พัฒนาได้มีการปรับเปลี่ยน แผนการดำเนินงานในภาคเรียนที่ 2 ใหม่ ดังตารางที่ 3.13 เนื่องจาก ตอนการลงมือทำจริงในส่วนของ back-end ได้มีปัญหาและอุปสรรคที่ไม่คาดคิด ทั้งในเรื่องของ การตั้งรูปภาพจากฐานข้อมูลมา- ให้แล้ว แต่ทางเว็บแอปพลิเคชันไม่แสดงรูปภาพที่ตั้งมา เป็นต้น จึงได้ทำการปรับแก้ใหม่ ในส่วนของ front-end ของหัวเป้าไซต์สามารถ- พัฒนาได้ตามที่ออกแบบไว้ได้สำเร็จ

ในส่วนของโมเดล ที่ใช้ในการสร้างรูปภาพใหม่ ในเว็บไซต์ของทางผู้พัฒนา โดยเริ่มต้นทางผู้พัฒนาได้ลองใช้ source code ที่ได้ทำเรื่อง neural style transfer แต่เป็นโมเดลที่ออกแบบมาใช้กับภาพศิลปินต่างชาติ อาทิเช่น วินเซนต์ แวน กอสส์ หรือ ปานโอล รุยซ์ ปิกาโซ เอามาลองกับ สไตล์รูปภาพของคลิปปีไทร์ทั้ง อ.ชุด และ อ.จักรพันธุ์ โดยเป็นการเทรนโมเดลแบบหนึ่งต่อหนึ่ง ซึ่งผลลัพธ์ที่ได้ รูปภาพใหม่ไม่ได้มีการดึงจุดเด่นของรูปภาพที่เป็นสไตล์ต้นแบบเลย หรือทำให้เกิดรูปภาพใหม่ที่ผิดแปลกลไป ทางผู้พัฒนาจึงได้ทำการทดลอง เพื่อหาข้อสรุปที่จะนำมายังสไตล์รูปภาพของอาจารย์ทั้ง 2 ท่าน โดยจากการทดลองที่ทางผู้พัฒนาได้ทดลองทำ สรุปได้ว่า โมเดล VGG19 ที่ใช้ดึงคุณลักษณะของรูปภาพคร่าวๆ ใช้ max pooling ใน pooling layer ขนาดของรูปภาพ style image และ content image คือ 224\*224 และ target image ควรเป็นรูปภาพที่คัดลอกมาจาก content image โดยใช้การ train เพื่อสร้างรูปภาพใหม่ที่สมรรถห่วง style image และ content image จำนวน 5,000 รอบ โดย style image ของ อ.ชุด นิ่มเสมอใช้ style weight เท่ากับ 100 และ content weight เท่ากับ 0.01 ส่วน style image ของ อ.จักรพันธุ์ นโยบายก็คุณบุญใช้ style weight เท่ากับ 100 และ content weight เท่ากับ 10

## หนังสืออ้างอิง

1. Leon A Gatys, Alexander S Ecker, and Matthias Bethge, 2015, ``A neural algorithm of artistic style," [arXiv preprint arXiv:1508.06576](https://arxiv.org/abs/1508.06576), 2015.
2. ``Deep Learning แบบฉบับคนสามัญ EP 1 : Neural Network History | by Mr.P L | mmp-li | Medium," <https://medium.com/mmp-li/deep-learning-%E0%B9%81%E0%B8%9A%E0%B8%9A%E0%B8%89%E0%B8%9A%E0%B8%B1%E0%B8%9A%E0%B8%84%E0%B8%99%E0%B8%AA%E0%B8%B2%E0%B8%A1%E0%B8%B1%E0%B8%8D%E0%B8%8A%E0%B8%99-ep-1-neural-network-history-f7789236a9a3>, (Accessed on 03/20/2021).
3. Jason Brownlee, 2017, ``A gentle introduction to transfer learning for deep learning," **Machine Learning Mastery**, 2017.
4. Sumit Saha, 2018, ``A comprehensive guide to convolutional neural networks—the ELI5 way," **Towards Data Science**, vol. 15, 2018.
5. ``Gradient Descent คืออะไรดีที่ Data Science ต้องรู้จัก ! : Machine Learning 101 | by Mr.P L | mmp-li | Medium," <https://medium.com/mmp-li/gradient-descent-%E0%B8%84%E0%B8%B5%E0%B8%A2%E0%B9%8C%E0%B9%80%E0%B8%A7%E0%B8%B4%E0%B8%A3%E0%B9%8C%E0%B8%94%E0%B8%97%E0%B8%B5%E0%B9%88-data-science-%E0%B8%95%E0%B9%89%E0%B8%AD%E0%B8%87%E0%B8%A3%E0%B8%B9%E0%B9%89%E0%B8%88%E0%B8%B1%E0%B8%81-machine-learning-101-b5c78c1787a0>, (Accessed on 03/20/2021).
6. ``What is Web Application Architecture? How It Works, Trends, Best Practices and More – Stackify," <https://stackify.com/web-application-architecture/>, (Accessed on 03/20/2021).
7. ``Neural Transfer Using PyTorch — PyTorch Tutorials 1.8.0 documentation," [https://pytorch.org/tutorials/advanced/neural\\_style\\_tutorial.html](https://pytorch.org/tutorials/advanced/neural_style_tutorial.html), (Accessed on 03/20/2021).
8. 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.
9. 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-hands-on-guide-to-transfer-learning-with-real-world-applications-in-deep-learning-212bf3b2f27a>. Hakupäivä, vol. 20, pp. 2020, 2018.
10. ``ปฏิทินการพัฒนา Web Application ภาษา Python ด้วย Django Framework | by Kong Ruksiamza | Medium," <https://kongruksiamza.medium.com/%E0%B8%9B%E0%B8%B9%E0%B8%9E%E0%B8%B7%E0%B9%89%E0%B8%99%E0%B8%90%E0%B8%B2%E0%B8%99%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B8%9E%E0%B8%B1%E0%B8%92%E0%B8%99%E0%B8%B2-web-application-%E0%B8%81%E0%B8%B1%E0%B8%9A%E0%B8%A0%E0%B8%B2%E0%B8%A9%E0%B8%B2-python-%E0%B8%94%E0%B9%89%E0%B8%A7%E0%B8%A2-django-framework-9d3b7f48718a>, (Accessed on 03/20/2021).

ภาคผนวก A

ชื่อภาคผนวกที่ 1

## ໄສ່ທັງໝົດຕາມຄວາມເໜາະສນ

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,  $\lambda$  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  $\Delta$ . 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  $\lambda_n$  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

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

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  $\varepsilon$ , 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  $\lambda_n$ . The MVA technique basically approximates the input process  $\lambda_n$  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  $\lambda$  can be easily measured, it is not the case for  $\sigma^2$  and  $H$ . Consequently, the MVA technique suffers from the need of prior knowledge traffic parameters.

ภาคผนวก B

ชื่อภาคผนวกที่ 2

## ໄສ່ທັງໝົດຕາມຄວາມເໜາະສນ

Next, we show how  $\text{Var}\{X_n\}$  can be determined. Let  $C_\lambda(l)$  be the autocovariance function of  $\lambda_n$ . The MVA technique basically approximates the input process  $\lambda_n$  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  $\lambda$  can be easily measured, it is not the case for  $\sigma^2$  and  $H$ . Consequently, the MVA technique suffers from the need of prior knowledge traffic parameters.