DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

การสร้างข้อมูลภาพจาก GANS เพื่อนำไปต่อยอดในด้าน vector art

ชาลี ชีพพิมลชัย

สารนิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมข้อมูลขนาดใหญ่
วิทยาลัยนวัตกรรมด้านเทคโนโลยีและวิศวกรรมศาสตร์
มหาวิทยาลัยธุรกิจบัณฑิตย์
ปีการศึกษา 2564

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

GENERATING PICTURES FROM GANS TO VECTOR ART

CHALEE CHEAPPIMOLCHAI

An Independent Study Submitted in Partial Fulfillment of the

Requirements for the Degree of Master of Big Data Engineering,

College of Innovative Technology and Engineering,

Dhurakij Pundit University

Academic Year 2021

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

ใบรับรองงานสารนิพนธ์

วิทยาลัยนวัตกรรมด้านเทคโนโลยีและวิศวกรรมศาสตร์ มหาวิทยาลัยธุรกิจบัณฑิตย์

ปริญญา วิศวกรรมศาสตรมหาบัณฑิต

หัวข้อสารนิพนธ์ การสร้างข้อมูลภาพจาก GANS เพื่อนำไปต่อยอดในด้าน vector art

เสนอโดย ชาลี ชีพพิมลชัย

สาขาวิชา วิศวกรรมข้อมูลขนาดใหญ่

อาจารย์ที่ปรึกษาสารนิพนธ์ ดร.ธนภัทร ฆังคะจิตร

ได้พิจารณาเห็นชอบโดยคณะกรรมการสอบสารนิพนธ์แล้ว

.....ประธานกรรมการ

(ดร.สรรพฤทธิ์ มฤคทัต)

..... กรรมการและอาจารย์ที่ปรึกษา

(ดร.ธนภัทร ฆังคะจิตร)

..... กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร.ดวงใจ จิตคงชื่น)

วิทยาลัยนวัตกรรมด้านเทคโนโลยีและวิศวกรรมศาสตร์รับรองแล้ว

.....

(ดร.ชัยพร เขมะภาตะพันธ์)

คณบดีวิทยาลัยนวัตกรรมด้านเทคโนโลยีและวิศวกรรมศาสตร์

วันที่ 6 เดือน ธันวาคม พ.ศ. 2564

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

หัวข้อสารนิพนธ์	การสร้างข้อมูลภาพจาก GANS เพื่อนำไปต่อยอดในด้าน art vector
ชื่อผู้เขียน	ชาลี ชีพพิมลชัย
อาจารย์ที่ปรึกษา	ดร.ธนภัทร ฆังคะจิตร
สาขาวิชา	วิศวกรรมข้อมูลขนาดใหญ่
ปีการศึกษา	2564

บทคัดย่อ

ธุรกิจด้านการออกแบบ graphic หรือ อาชีพ graphic design เป็นงานที่สร้างมูลค่าให้กับหลายธุรกิจเช่น ธุรกิจสื่อ , ธุรกิจ Event รวมทั้งยังสร้างอาชีพที่เกิดใหม่เช่น Vector specialist , การออกแบบ UX/UI ดังนั้นเพื่อปรับปรุงงานออกแบบให้มีคุณภาพมากขึ้น ควรอาศัยความก้าวหน้าทางด้านเทคโนโลยีของคอมพิวเตอร์วิชั่นซึ่งเทคนิคการเรียนรู้เชิงลึกนี้จะใช้ Python เพื่อสร้างผลลัพธ์ graphic ที่มีคุณภาพออกมา

ในปัจจุบันเทคโนโลยีการเรียนรู้เชิงลึกทางด้านคอมพิวเตอร์วิชั่นนี้ถูกนำมาใช้ประโยชน์ในหลาย ๆด้าน ไม่ว่าจะเป็นการสร้างรูปภาพเหมือนหน้าคน , การสร้าง Animation ในต้นทุนที่ไม่สูง , การออกแบบรถโดยหลักการออกแบบที่ดีที่สุด , การสร้างข้อมูลสำหรับdetection object ในหลากหลายมิติ ผู้วิจัยจึงได้ มองเห็นศักยภาพของระบบเทคโนโลยีในปัจจุบันที่สามารถนำมาประยุกต์ใช้ในงานออกแบบ และจาก ผลงานวิจัยนี้จึงทำให้สามารถสร้างวัตถุที่ต้องการได้อย่างถูกต้อง รวดเร็วและแม่นยำขึ้น

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

Independent Study Title	GENERATING PICTURES FROM GANS TO VECTOR ART
Author	Chalee Cheappimolchai
Independent Study Advisor	Dr. Thanapat Kangkachit
Department	Big Data Engineering
Academic Year	2021

ABSTRACT

A graphic design business or graphic design career is a field that creates value for many companies in various types of business such as media and event. It also creates new careers namely vector specialist and UX / UI designer. To improve design to be of high quality, advancement of computer vision technology, i.e., deep learning, is required. This deep learning technique uses Python to produce quality graphic results.

Today, deep learning in computer vision technology is being utilized in many tasks, for instance, creating human face images, creating low-cost animations, designing cars under the best design principles. Furthermore, it generates data to detect objects in multiple dimensions. As the researcher sees the potential of such today's technology that can be applied in design work, this research therefore assesses possibility in creating desired objects with deep learning more accurately, faster, and of higher quality.

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

กิตติกรรมประกาศ

ผู้วิจัยขอกราบขอบพระคุณในความกรุณาของอาจารย์ที่ปรึกษาสารนิพนธ์ ดร.ธนภัทร ฆังคะจิตร เป็นอย่างสูงที่เสียสละเวลาอันมีค่าเพื่อให้คำปรึกษาและคำแนะนำในการทำวิจัยตลอดระยะเวลาที่ผ่านมา อาจารย์ได้ให้ข้อเสนอแนะ ความคิดเห็นและทรัพยากรที่มีประโยชน์ ต่องานวิจัยชิ้นนี้ รวมถึงเอาใจใส่ผู้วิจัยเป็นอย่างดี ขอขอบพระคุณอาจารย์ที่สละเวลาเพื่อมาเป็น กรรมการในการสอบวิทยานิพนธ์ และได้ให้คำแนะนำแนวทางเป็นประโยชน์ต่องานวิจัย ขอขอบคุณเจ้าหน้าที่หลักสูตรวิศวกรรมศาสตร์มหาบัณฑิต สาขา วิชา วิศวกรรมข้อมูลขนาดใหญ่ทุกท่านสำหรับการช่วยเหลือและประสานงานเพื่อให้การดาเนินการทำวิจัยเป็นไปอย่างราบรื่น ขอขอบคุณมิตรสหายทุกท่านที่ได้ให้คำปรึกษาและให้กำลังใจในเสมอมา ตลอดจนมหาวิทยาลัยที่ ให้โอกาสในการศึกษาเรียนรู้ตามความสนใจของผู้วิจัย และสุดท้ายผู้วิจัยขอขอบคุณครอบครัวที่ให้ การสนับสนุน และความเข้าใจ ซึ่งทั้งหมดได้ส่งผลให้งานวิจัยสำเร็จลุล่วงไปได้ด้วยดีหากมีสิ่งได้ที่

ผู้วิจัยได้ทำผิดพลาดหรือบกพร่องประการใด ผู้วิจัยต้องกราบขออภัยเป็นอย่างสูงมา ณ โอกาสนี้ ผู้วิจัยหวังเป็นอย่างยิ่งว่าสารนิพนธ์ฉบับนี้ จะเป็นพื้นฐานในการต่อยอดองค์ความรู้ของผู้ที่สนใจ ศึกษาในงานด้านนี้ต่อไป

ชาลี ชีพพิมลชัย

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

สารบัญ

	หน้า
บทคัดย่อภาษาไทย……	ฆ
บทคัดย่อภาษาอังกฤษ ……..	ง
กิตติกรรมประกาศ ……	จ
สารบัญตาราง ……..…	ซ
สารบัญภาพ ……..	ฌ
บทที่	
1. บทนำ……	1
1.1 ที่มาและความสำคัญของปัญหา…….....	1
1.2 วัตถุประสงค์ของการวิจัย……	2
1.3 ประโยชน์ที่คาดว่าจะได้รับ ……	2
2. แนวคิด ทฤษฎี และผลงานวิจัยที่เกี่ยวข้อง……..	7
2.1 บทนำ……...	7
2.2 ความรู้เกี่ยวกับการเรียนรู้เชิงลึก……...	7
2.3 ความรู้เกี่ยวกับการเรียนรู้เชิงลึก สำหรับ Vision System…….	14
2.4 ความรู้เกี่ยวกับGANs……...	28
2.5 งานวิจัยที่เกี่ยวข้อง……..	34
3. ระเบียบวิธีวิจัย ……...	44
3.1 เครื่องมือที่ใช้ในงานวิจัย……..	44
3.2 ข้อมูลที่ใช้ในงานวิจัย……	45
3.3 การติดตั้ง Packages และ Environments สำหรับเขียนโค้ดใน Python…….	47
3.4 ขั้นตอนการวิจัยในกระบวนการเรียนรู้ GANS……..	51
4. ผลการทดลอง…….	63
4.1 ผลการเตรียมข้อมูลภาพก่อนการเทรนโมเดล……...	63
4.2 ผลการดำเนินงานในส่วนของโมเดล GANS…….	63
4.3 ผลลัพธ์จากการสร้างภาพจาก pretrain…….	65

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

สารบัญ (ต่อ)

หน้า

5. สรุปผลการทดลอง..... 67
5.1 สรุปผลการวิจัย..... 67
5.2 อภิปรายผล..... 67
5.3 ข้อเสนอแนะ..... 68
บรรณานุกรม..... 70
ภาคผนวก..... 72
ประวัติผู้เขียน..... 75

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

สารบัญตาราง

ตารางที่ **หน้า**

4.1 ตารางการวัดประสิทธิภาพของ model โดยใช้ Frèchet inception distance (FID)... **65**

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

สารบัญภาพ

ภาพที่ **หน้า**

1.1 กรอบแนวคิดงานวิจัย..... 5

2.1 การเรียนของ Machine Learning..... 8

2.2 การทำงานของสมองมนุษย์..... 10

2.3 feature map..... 12

2.4 Stride และ Padding 13

2.5 Pooling layer..... 14

2.6 Computer vision pipeline..... 15

2.7 Image Input..... 16

2.8 การมองภาพของคอมพิวเตอร์..... 16

2.9 การมองภาพสี..... 17

2.10 Feature extraction..... 18

2.11 ภายในFeature extraction..... 18

2.12 Deep learning flow..... 18

2.13 กระบวนการ ANN..... 19

2.14 กระบวนการ Single perceptron..... 19

2.15 Weight Vector..... 20

2.16 Multilayer perceptron..... 20

2.17 Activation function..... 21

2.18 Softmax function..... 22

2.19 Leaky ReLU..... 22

2.20 Feedforward Process..... 22

2.21 Mean square error..... 23

2.22 Cross-entropy..... 23

2.23 Backpropagation..... 24

2.24 CNN architecture overview..... 25

2.25 CNN architecture process..... 25

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

สารบัญภาพ (ต่อ)

ภาพที่ **หน้า**

2.26 CNN architecture show classification..... 26

2.27 Convolutional layers..... 27

2.28 Pooling layers..... 27

2.29 Fully connected layers..... 28

2.30 What GANs..... 28

2.31 GANs overview..... 29

2.32 Generative model..... 30

2.33 Up sample layer..... 30

2.34 Transpose Convolutional Layer..... 31

2.35 Discriminator model..... 31

2.36 Discriminator training..... 32

2.37 Generator training..... 33

2.38 Minimax function..... 33

2.39 ConvNets..... 34

2.40 สมการ Batch normalization..... 35

2.41 กระบวนการใน DCGAN..... 36

2.42 กระบวนการใน Style Gan..... 37

2.43 กระบวนการใน Adaptive instance normalization..... 38

2.44 ตัวอย่างผลลัพธ์ที่ได้จาก Pix2Pix GAN..... 39

2.45 ตัวอย่างผลลัพธ์ Style Transfer..... 40

2.46 ตัวอย่างผลลัพธ์ Object Transfiguration..... 40

2.47 ตัวอย่างผลลัพธ์ Season Transfer..... 41

2.48 ตัวอย่างผลลัพธ์ Photograph Generation..... 41

2.49 ตัวอย่างผลลัพธ์ Photograph Enhancement..... 41

2.50 Updrawing overview..... 42

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

สารบัญภาพ (ต่อ)

ภาพที่ **หน้า**

2.51 U-Net structure..... 43

2.52 ผลลัพธ์ของ APDrawingGAN ที่มีผลลัพธ์ที่ดีกว่า..... 43

3.1 TensorFlow..... 44

3.2 PyTorch..... 44

3.3 NumPy..... 44

3.4 Colab..... 45

3.5 ภาพตัวอย่างข้อมูล..... 45

3.6 ภาพตัวอย่างการลบพื้นหลัง..... 46

3.7 ภาพโปรแกรมที่เอาไว้ปรับแก้ขนาดของภาพ..... 47

3.8 ภาพโปรแกรมGoogle colab..... 48

3.9 ภาพโปรแกรมGoogle colab set GPU..... 48

3.10 ภาพโปรแกรมGoogle colab install TensorFlow..... 49

3.11 ภาพโปรแกรมGoogle colab install stylegan..... 49

3.12 ภาพชุด Codeที่สร้างมาเพื่อเปลี่ยนสกุลไฟล์..... 50

3.13 ภาพชุดข้อมูลที่ถูก zip file..... 50

3.14 ภาพชุดcode unzip file..... 50

3.15 รูปแบบการเก็บข้อมูลแบบ TFRecord..... 51

3.16 ภาพชุดข้อมูลที่ถูกเก็บแบบ TFRecord..... 51

3.17 ภาพโปรแกรมGoogle colab new file..... 52

3.18 ภาพชุด Code mount drive..... 52

3.19 ภาพโปรแกรมGoogle colab install stylegan..... 52

3.20 ภาพชุด Code install TensorFlow and ml4a..... 53

3.21 ภาพชุด Code Train model..... 53

3.22 ข้อมูลที่ Train เสร็จแล้ว..... 54

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

สารบัญภาพ (ต่อ)

ภาพที่ **หน้า**

3.23 ภาพโปรแกรมGoogle colab new file..... 54
3.24 ภาพโปรแกรมGoogle colab install stylegan pytorch..... 55
3.25 ภาพชุด Code Check GPU..... 55
3.26 ภาพชุด Code เตรียมข้อมูลก่อน train..... 56
3.27 ภาพชุด Code train ข้อมูล..... 56
3.28 ผลลัพธ์ของโมเดล GAN ที่สร้างจากจุดเริ่มต้น..... 57
3.29 ผลลัพธ์ของโมเดล GAN ที่เทรนด้วยภาพ 40 ภาพ..... 57
3.30 ผลลัพธ์ของโมเดล GAN ที่เทรนด้วยภาพ 80 ภาพ..... 57
3.31 ผลลัพธ์ของโมเดล GAN ที่เทรนด้วยภาพ 120 ภาพ..... 58
3.32 ผลลัพธ์ของโมเดล GAN ที่เทรนด้วยภาพ 160 ภาพ..... 58
3.33 ผลลัพธ์ของโมเดล GAN ที่เทรนด้วยภาพ 200 ภาพ..... 58
3.34 ผลลัพธ์ของโมเดล GAN ที่เทรนด้วยภาพ 240 ภาพ..... 59
3.35 ผลลัพธ์ของโมเดล GAN ที่เทรนด้วยภาพ 280 ภาพ..... 59
3.36 ผลลัพธ์ของโมเดล GAN ที่เทรนด้วยภาพ 320 ภาพ..... 59
3.37 ผลลัพธ์ของภาพที่ใช้เวลาในการเทรน1 อาทิตย์..... 60
3.38 ภาพชุด Code install TensorFlow and ml4a..... 60
3.39 ภาพชุด Code install Ninja..... 61
3.40 ภาพชุด Code export weight..... 61
3.41 ภาพชุด Code การปรับ code generator..... 61
3.42 ภาพชุด Code การปรับ export data..... 62
3.43 ภาพชุด Code Generate data..... 62
4.1 ผลการเตรียมข้อมูล..... 63
4.2 ผลลัพธ์ของข้อมูลที่ไม่ได้ถูกการตัด Background..... 64
4.3 ผลลัพธ์ของข้อมูลที่ไม่ได้ถูกการตัด Background..... 64
4.4 ตัวอย่างของภาพที่สร้าง..... 66

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มาของปัญหา

การพัฒนาให้คอมพิวเตอร์มีความสามารถเหมือนมนุษย์ เช่น การมองเห็น การพูด การตัดสินใจ ความรู้สึก เป็นต้น ในงานประชุม "International Conference on Neural Information" ปี 2014 ได้ให้กำเนิดงานที่เป็นรากฐานในการ Generate รูปภาพ สำหรับ computer vision โดย model นั้น มีชื่อ ว่า GANS จุดประสงค์คือ เพื่อสร้าง Framework รูปแบบใหม่ ในการเทรน 2 model คือ 1) Generative model เอาไว้ จับ data distribution 2) Discriminative model เอาไว้ประมาณ ค่าความน่าจะเป็น ของตัวอย่างในการ training

Machine Learning เป็นการใช้อัลกอริทึมในการแยกแยะและวิเคราะห์ข้อมูล เรียนรู้จากข้อมูลนั้น เพื่อสร้างโมเดลในการตัดสินใจหรือคาดการณ์เกี่ยวกับบางสิ่งบางอย่าง แทนที่จะเขียนโค้ด เป็นชุดคำสั่งให้ปฏิบัติตามขั้นตอนที่กำหนดไว้ แต่จะใช้การ "เทรนด์" โดยใช้ชุดข้อมูลขนาดใหญ่ใน การเรียนรู้ที่จะปฏิบัติงานนั้น ๆ การที่จะสอนให้เครื่องคอมพิวเตอร์สามารถเรียนรู้ได้นั้นก็มีเทคนิคต่าง ๆ มากมาย เช่น Supervised Learning, Unsupervised Learning, Reinforcement Learning หรือ Artificial Neuron Networks เป็นต้น และหนึ่งในเทคนิคที่ถูกพูดถึงกันเป็นอย่างมากก็คือ "Deep Learning"

Artificial Neural Networks (ANN) เป็นเทคนิคที่เลียนแบบการทำงานของสมองมนุษย์ ซึ่ง ประกอบด้วยเซลล์ประสาท (Neuron) และแต่ละเซลล์จะถูกเชื่องโยงกันเป็นโครงข่าย ซึ่งในซอฟต์แวร์ Neuron จะเรียกว่า "โหนด (Node)" และแต่ละโหนดจะถูกแบ่งออกเป็นชั้น (Layer) โดยหลักการ ของ Deep Learning ก็จะเป็น ANN ที่มีโหนดหลายๆชั้น ทำให้สามารถประมวลผลได้ครั้งละจำนวน มาก ช่วยให้การเรียนรู้ของเครื่องสามารถให้ผลลัพธ์ในการตัดสินใจและคาดการณ์ได้ดีมากยิ่งขึ้น

ปัจจุบันมีกลุ่มนักพัฒนางานด้าน Creating image เกิดขึ้นและแพร่ขยายไปทั่วโลก ทั้งในระดับ ทวีป หรือประเทศ โดยจะจัด Conference ทุกปี ใช้ชื่อว่า International computer vision (ICCV) ซึ่งเป็นองค์กรอิสระไม่หวัง ผลทางธุรกิจ เกิดจากการรวมกลุ่มของบริษัทต่างๆ ใน

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

ภาคอุตสาหกรรมที่เกี่ยวข้องกับ Computer Vision และ ภาคการศึกษาที่มีการทำ Research ในด้าน Computer Vision

1.2 วัตถุประสงค์ของงานวิจัย

เพื่อสร้างผลลัพธ์ของงานรูปภาพและวิเคราะห์ให้ Generative Adversarial Networks model(GANs) มีความเสถียรมากขึ้น โดยจะนำ StyleGAN มาประยุกต์ใช้ในกระบวนการงานวิจัย และนำรูปภาพไปใช้ในธุรกิจ Graphic Design

1.3 ความสำคัญของงานวิจัย

การศึกษาเทคโนโลยีการเรียนรู้เชิงลึก (Deep Learning) ในการเตรียมข้อมูลและสร้างรูปภาพ เพื่อนำไปสร้างผลลัพธ์ ให้ รวดเร็ว และแม่นยำ มากยิ่งขึ้น

1.4 ขอบเขตของการวิจัย

1.4.1 ขอบเขตด้านข้อมูล

งานวิจัยนี้ ผู้จัดทำได้กำหนดรูปแบบของชุดข้อมูลภาพที่ต้องการจะสร้างขึ้นมาใหม่ในการศึกษาอย่างชัดเจน โดยแนวคิดของงานวิจัยนี้ เป็นการนำเทคโนโลยีการเรียนรู้เชิงลึก (Deep Learning) มาช่วยในการสร้าง model ที่สามารถนำไปทำงานร่วมกับภาพที่เป็น ศิลปะได้ เพื่อนำไปใช้ ในงาน Graphic Design ดังนั้น หากจะนำข้อมูล หรือกระบวนการต่างๆไปใช้นอกเหนือจากงาน Graphic Design อาจต้องมีการศึกษา และ ปรับปรุงข้อมูลเพิ่มเติม เพื่อที่จะสามารถนำไปประยุกต์ใช้ให้ตรงกับงานในด้านนั้นๆ

1.5 นิยามศัพท์เฉพาะ

AI คือ สิ่งที่มนุษย์พยายามสร้างให้เครื่องจักร (Machine) มีความคิด มีปัญญา (Intelligence) สามารถคิดแก้ปัญหาที่ซับซ้อนมากๆได้ ในระดับที่ใกล้เคียงมนุษย์ หรือมากกว่ามนุษย์ในเฉพาะด้านด้วยซ้ำ ยกตัวอย่าง AI ในชีวิตประจำวัน เช่น Siri ที่สามารถโต้ตอบคำถามของมนุษย์ได้อย่างชาญฉลาด รวมถึง AI ที่มองไม่เห็น เช่นระบบจัดการอีเมลสแปม ของGmail ที่คอยแยกแยะเมลจริงกับเมลสแปม

เทคโนโลยี computer vision นั้น เป็นแขนงหนึ่งของวิทยาการปัญญาประดิษฐ์หรือ AI ซึ่งทำการฝึกฝนคอมพิวเตอร์ และระบบให้สามารถเข้าใจและตอบสนองต่อข้อมูลภาพได้อย่าง ชาญ

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

ฉลาด ด้วยภาพดิจิทัลจากกล้องถ่ายภาพและวิดีโอต่าง ๆ และแบบจำลอง deep learning นั้น อุปกรณ์ต่าง ๆ จะสามารถเรียนรู้ที่จะระบุและทราบถึงวัตถุต่าง ๆ จากนั้นจะสามารถทำการตอบสนองต่อสิ่งที่ มองเห็น ได้ต่อไป รูปแบบของเทคโนโลยี computer vision หรือคอมพิวเตอร์วิทัศน์หลายรูปแบบ และมีการใช้งานในหลายสถานการณ์ตามไปด้วย ดังนี้

1. Image classification คือการแยกส่วนของภาพออกเป็นหลาย ๆ ส่วนหรือชิ้นองค์ประกอบย่อย ๆ เพื่อพิจารณาแยกส่วนกัน

2. Object detection - หรือการตรวจหาวัตถุแบบเฉพาะเจาะจงในภาพแต่ละภาพ ซึ่งมีการทำงานในระดับสูงที่สามารถระบุวัตถุหลายชิ้นในภาพเดียวกันได้ เช่น ในภาพของการแข่งขันฟุตบอลนั้น อาจระบุวัตถุต่าง ๆ ได้แก่ สนามฟุตบอล ผู้เล่นฝั่งทีมรุก ผู้เล่นฝั่งทีมรับ ฯลฯ โดยการทำงานของแบบจำลองการวิเคราะห์นี้อาศัยการทำจุดพิกัดตามแกน X และ Y เพื่อสร้างกล่องสำหรับการพิจารณา และระบุวัตถุทุกชิ้นที่อยู่ในในพื้นที่กล่องแต่ละกล่องที่กำหนดขึ้น

3. Facial recognition - หรือการจดจำใบหน้า เป็นรูปแบบการระบุวัตถุขั้นสูงที่มิได้ทำแค่การระบุว่ามีใบหน้าของมนุษย์อยู่ในภาพเท่านั้น แต่ยังสามารถแยกแยะบุคคลแต่ละบุคคลออกจากกันและระบุบุคคลที่เจาะจงได้อีกด้วย

4. Edge detection - เป็นเทคนิคการระบุหาขอบหรือมุมของวัตถุ หรือภาพทิวทัศน์เพื่อให้ทราบได้ง่ายขึ้นว่าองค์ประกอบในภาพมีสิ่งใดบ้าง

5. Pattern detection - คือการระบุวัตถุจากรูปทรง หรือสี หรือสิ่งบ่งชี้ต่าง ๆ ที่พบในภาพ ที่เป็นรูปแบบเดียวกันซ้ำ ๆ สำหรับวัตถุประเภทนั้น ๆ

6. Feature matching - เป็นรูปแบบหนึ่งของการตรวจหารูปแบบหรือ pattern detection ที่ระบุจุดที่เหมือนหรือคล้ายคลึงกันในภาพต่าง ๆ เพื่อจัดหมวดหมู่แก่วัตถุและภาพเหล่านั้น

นิยามศัพท์เฉพาะการเรียนรู้เชิงลึก (Deep Learning) เป็นหนึ่งในฟังก์ชันของปัญญาประดิษฐ์ (AI) ที่เรียนแบบ การทำงานของสมองมนุษย์ในกระบวนการประมวลผลข้อมูลและเป็นการสร้างรูปแบบ สำหรับใช้ในการตัดสินใจ นอกจากนี้ Deep Learning ยังเป็นเซตย่อยของ Machine Learning ใน Artificial Intelligence (AI) เป็นเครือข่ายที่มีประสิทธิภาพของการเรียนรู้ที่ไม่มีผู้สอน หรือ Unsupervised Learning จากข้อมูลแบบไม่มีโครงสร้างและแบบที่ไม่กำกับข้อมูล ทั้งนี้เป็นที่รู้จักกันในชื่อ Deep Neural Learning และ Deep Neural Network

Deep Learning เป็นเซตย่อยของ Machine Learning หรือการเรียนรู้ด้วยเครื่อง ซึ่งเป็นลำดับชั้นของเครือข่ายประสาทเทียม (Artificial Neural Network) โดยดำเนินการด้วย Machine Learning เนื่องจากเครือข่ายประสาทเทียมถูกสร้างมาเหมือนสมองของมนุษย์มี Nodes เชื่อมต่อกัน

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

เหมือนเว็บไซต์ ระบบ Deep Learning ช่วยให้เครื่องประมวลผลข้อมูลด้วยวิธีการไม่ใช่เชิงเส้น ซึ่งเป็นเทคนิคการ เรียนรู้เชิงลึก เพื่อกำจัดสิ่งแปลกปลอมในการทำธุรกรรมที่เสี่ยงต่อการหลอกลวงจะประกอบด้วยเวลา, ที่ตั้งทางภูมิศาสตร์, IP address, ประเภทของธุรกิจค้าปลีก และคุณลักษณะอื่น ๆ ที่ก่อให้เกิดวิธีการ หลอกลวงจากบุคคลที่ 3

โครงข่ายประสาทแบบคอนโวลูชัน(CNN) เป็นโครงข่ายประสาทเทียมหนึ่งในกลุ่ม bio-inspired โดยที่ จะจำลองการมองเห็นของมนุษย์ที่มองพื้นที่เป็นที่ย่อยๆ และนำกลุ่มของพื้นที่ย่อยๆมาผสานกัน การมองพื้นที่ย่อยของมนุษย์จะมีการแยกคุณลักษณะ (feature) ของพื้นที่ย่อยนั้น เช่น ลายเส้น และการตัดกันของสี ซึ่งการที่มนุษย์รู้ว่าพื้นที่ตรงนี้เป็นเส้นตรงหรือสีตัดกัน เพราะมนุษย์ดูทั้งจุดที่สนใจและบริเวณรอบ ๆ ประกอบกัน มีส่วน

1. Convolution state คือ ส่วนที่ทำการแยกองค์ประกอบของรูปออกมา เช่น ขอบ สี รูปทรง เป็นต้น โดยที่ CNN จะมีตัวกรอง หรือ Filter ในการตรวจสอบเพื่อแยกองค์ประกอบของรูป

2. Detector stage คือ ส่วนที่รับoutput มาจาก convolution state และ แปลงให้อยู่ในรูปของ nonlinear และเป็นส่วนที่ต้องระบุ อัลกอริทึ่ม (Activation) ที่จะใช้ในการจำแนกคุณลักษณะของข้อมูล

3. Pooling stage คือ ตัวกรองแบบหนึ่งที่หาค่าสูงสุดในบริเวณที่ตัวกรองทาบอยู่บนข้อมูล แล้วเลือกค่าที่สูงที่สุดบนตัวกรองนั้นมาเป็นผลลัพธ์ใหม่ และจะเลื่อนตัวกรองไปตาม Stride ที่กำหนดไว้

การจำแนกประเภทข้อมูลภาพ (Image classification) การจำแนกประเภทข้อมูลภาพเป็นการประมวลผลในทางสถิติ เพื่อแยกข้อมูลจุดภาพทั้งหมดที่ประกอบเป็นพื้นที่ศึกษาออกเป็นกลุ่มย่อย โดยใช้ลักษณะทางสถิติเป็นตัวกำหนดความแตกต่างระหว่างกลุ่มจุดภาพ โดยจุดภาพที่ถูกจัดให้อยู่กลุ่ม เดียวกันจะมีลักษณะทางสถิติเฉพาะกลุ่มเป็นไปในทิศทางเดียวกัน แต่ละกลุ่มจุดภาพที่จำแนกได้นั้นจะ แสดงถึงสิ่งปกคลุมประเภทใดประเภทหนึ่งแตกต่างกันไป

การจำแนกประเภทข้อมูลภาพด้วยระบบคอมพิวเตอร์แบ่งออกได้เป็น 2 วิธี คือ การจำแนก ประเภทข้อมูลภาพแบบควบคุม (Supervised classification) และการจำแนกประเภทข้อมูลภาพ แบบไม่ควบคุม(Unsupervised classification) การจำแนกประเภทข้อมูลภาพทั้งสองวิธีให้ได้ผลลัพธ์ ที่มีประสิทธิภาพ ก่อนนำไปใช้ใน model ควรศึกษาค่าสถิติ ที่มีผล ต่อประสิทธิภาพ

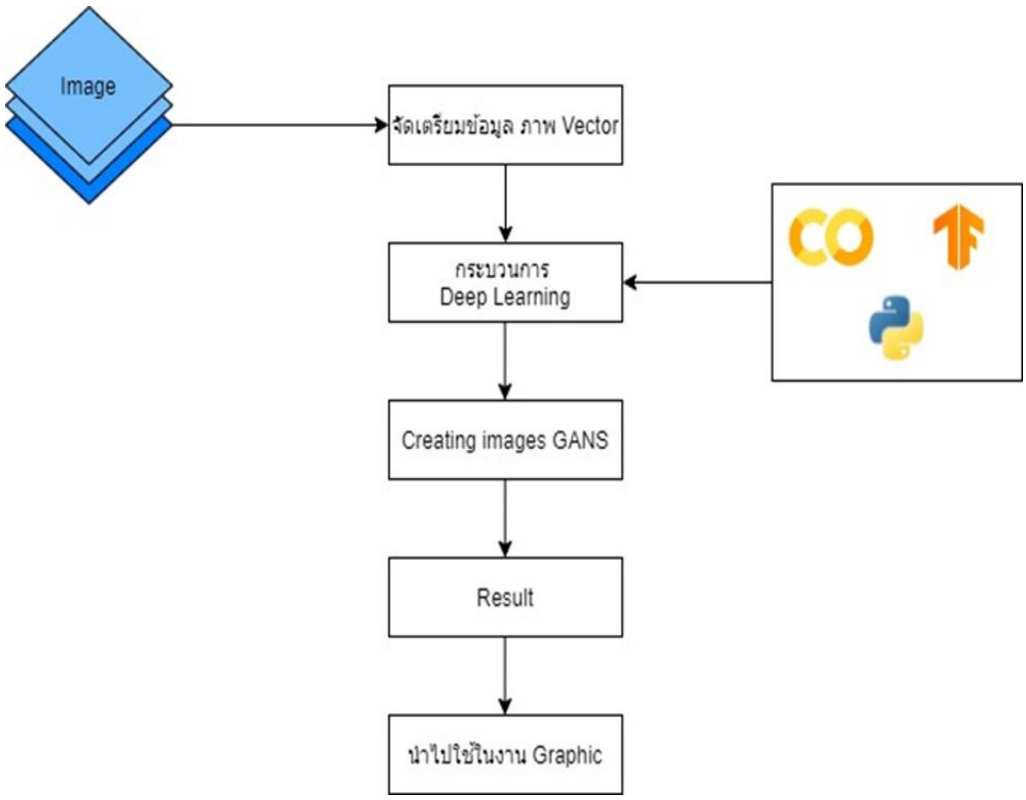
1. การจำแนกประเภทข้อมูลแบบควบคุม เป็นศาสตร์แขนงหนึ่งใน AI หรือปัญญาประดิษฐ์ภายใต้หัวข้อ Machine Learning โดยทำให้คอมพิวเตอร์สามารถหาคำตอบของปัญหาได้ด้วยตัวเอง หลังจากเรียนรู้จากชุดข้อมูลตัวอย่างไปแล้วระยะหนึ่งโดยผู้ใช้งานเป็นผู้กำหนดลักษณะ ของประเภทข้อมูลเอง โดยเป็นผู้เลือกตัวอย่างประเภทข้อมูลให้แก่เครื่อง จึงเรียกการจำแนกข้อมูล

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

ประเภทนี้ว่าเป็นวิธีแบบควบคุมโดยผู้วิเคราะห์ต้องควบคุมอย่างใกล้ชิด การนำ Supervised Learning มาใช้ ในการสร้าง model เรียกว่า การ Train เป็นสิ่งจำเป็นสำหรับ การจำแนกข้อมูลแบบควบคุม

2. การจำแนกประเภทข้อมูลแบบไม่ควบคุม เป็นกลุ่ม algorithm ที่ไม่มี label หรือการสอนอย่างชัดเจนว่าถ้าทำงานแล้วได้ผลลัพธ์แบบนี้หมายถึงถูกหรือผิด โดยคอมพิวเตอร์จะจำแนกมีข้อมูลคุณลักษณะ (Feature) ด้วยตัวเอง คำตอบที่ได้จากการเรียนรู้แบบนี้คือลักษณะโครงสร้างความสัมพันธ์ของข้อมูล ที่แยกแยะระหว่างข้อมูลมากกว่า 1 กลุ่มขึ้นไป

1.6 กรอบแนวคิดงานวิจัย



ภาพที่ 1.1 กรอบแนวคิดงานวิจัย

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

1.7 ประโยชน์ที่คาดว่าจะได้รับ

1. เพื่อลดระยะเวลาในการสร้างงาน Art โดยใช้ Model ที่ถูกพัฒนาเพื่อให้เหมือนกับงานต้นแบบมากที่สุด

2. สามารถสร้างภาพ vector เป็นภาพจริงได้ละเอียด และสมจริงขึ้น

3. ทำให้คนที่อยากทำ Sticker หรือ อยากได้ Graphic สามารถ ทำได้โดยไม่จำเป็นต้องมีพื้นฐาน

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

บทที่ 2
เอกสารและงานวิจัยที่เกี่ยวข้อง

2.1 ความรู้เกี่ยวกับการเรียนรู้เชิงลึก

2.1.1 ปัญญาประดิษฐ์ (Artificial Intelligence) หรือเอไอ (AI) เครื่องจักร (machine) ที่มีฟังก์ชันทีมีความสามารถในการทำความเข้าใจ เรียนรู้องค์ความรู้ต่างๆ อาทิเช่น การรับรู้ การเรียนรู้ การให้เหตุผล และการแก้ปัญหาต่างๆ โดย AI ถูกจำแนกเป็น 3 ระดับตามความสามารถหรือความฉลาดดังนี้

1) ปัญญาประดิษฐ์เชิงแคบ (Narrow AI) หรือ ปัญญาประดิษฐ์แบบอ่อน (Weak AI) คือ AI ที่มีความสามารถเฉพาะทางได้ดีกว่ามนุษย์(เป็นที่มาของคำว่า Narrow(แคบ) ก็คือ AI ที่เก่งในเรื่องแคบๆหรือเรื่องเฉพาะทางนั่นเอง) อาทิ เช่น AI ที่ช่วยในการผ่าตัด (AI-assisted robotic surgery)

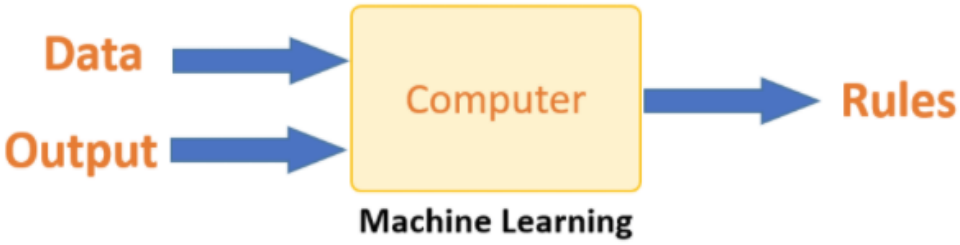
2) ปัญญาประดิษฐ์ทั่วไป (General AI) คือ AI ที่มีความสามารถระดับเดียวกับมนุษย์ สามารถทำทุกๆอย่างที่มนุษย์ทำได้และได้ประสิทธิภาพที่ใกล้เคียงกับมนุษย์

3) ปัญญาประดิษฐ์แบบเข้ม (Strong AI) คือ AI ที่มีความสามารถเหนือมนุษย์ในหลายๆ ด้าน

2.1.2 การเรียนรู้ของเครื่อง (machine learning) เป็นสาขาหนึ่งของปัญญาประดิษฐ์ที่พัฒนามาจากการศึกษาการรู้จำแบบ เกี่ยวข้องกับการศึกษาและการสร้างอัลกอริทึมที่สามารถเรียนรู้ ข้อมูลและทำนายข้อมูลได้ อัลกอริทึมนั้นจะทำงานโดยอาศัยโมเดลที่สร้างมาจากชุดข้อมูลตัวอย่างขาเข้าเพื่อการทำนายหรือตัดสินใจในภายหลัง แทนที่จะทำงานตามลำดับของคำสั่งโปรแกรมคอมพิวเตอร์ Machine Learning ประกอบได้ด้วยข้อมูลและเครื่องมือทางสถิติเพื่อทำนายผลลัพธ์ออกมาผลลัพธ์ในที่นี้ถูกใช้เพื่อทำประโยชน์กับภายในองค์กรเชิงลึกต่อไป Machine Learning เกี่ยวข้องอย่างมากกับ การทำเหมืองข้อมูล (Data mining) และ โมเดลการทำนายของ Bayes (Bayesian predictive models) เครื่องคอมพิวเตอร์จะรับข้อมูลเข้ามาและใช้อัลกอริทึมเพื่อหาคำตอบ วัตถุประสงค์หลักของ Machine Learning คือ การเรียนรู้ (learning) และการอนุมาน (inference) สิ่งแรกคือ machine จะเรียนรู้ผ่านการค้นพบรูปแบบหรือแบบแผนซ้ำ ๆ เดิม ๆ การค้นพบต้องขอบคุณ

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

ถึงการมีอยู่ข้อมูล (data) ส่วนหนึ่งที่สำคัญของนักวิทยาศาสตร์ข้อมูล (data scientist) จะเป็นการเลือกอย่างระมัดระวังว่าข้อมูลไหนสนับสนุนกับเครื่อง คุณสมบัติต่าง ๆ ที่ถูกใช้เพื่อแก้ปัญหาถูกเรียกว่า feature vector คุณสามารถนึกถึง feature vector ว่าเป็นซับเซ็ตย่อยของข้อมูลทั้งหมดที่ถูกใช้ในการแก้ปัญหา



ภาพที่ 2.1 การเรียนของ Machine Learning

การเรียนรู้ของเครื่อง สามารถแบ่งโดยกว้างๆได้เป็น 3 ประเภท ตามประเภทของข้อมูลฝึกหรือ ข้อมูลขาเข้า ได้ดังนี้

Supervised learning: การเรียนรู้แบบมีผู้สอนจะมีการกำหนดโมเดลตัวอย่างไว้ล่วงหน้าเพื่อให้แน่ใจว่ามีการจัดสรรข้อมูลอย่างเพียงพอกับกลุ่มโมเดลที่เกี่ยวข้องของอัลกอริทึมเหล่านี้ ซึ่งจากนั้นต้องมีการจำเพาะเจาะจง กล่าวอีกนัยหนึ่งระบบจะเรียนรู้บนพื้นฐานของ input และ output ที่กำหนดไว้ ทั้งนี้ ตามหลักของเรียนรู้ที่ได้รับการตรวจสอบ โปรแกรมเมอร์ซึ่งทำหน้าที่เป็นเหมือนครูผู้สอนเป็นผู้จัดสรรค่าที่เหมาะสมสำหรับ particular input เป้าหมายก็คือการ train ระบบในบริบทของการคำนวณต่อเนื่องกับ inputและ output ที่แตกต่างกันและเพื่อสร้างการเชื่อมต่อ

Unsupervised learning การเรียนรู้แบบไม่มีผู้สอน ปัญญาประดิษฐ์จะเรียนรู้โดยไม่มีค่าเป้าหมายที่กำหนดไว้ล่วงหน้าและไม่มีผลตอบแทน ส่วนใหญ่ใช้เพื่อการแบ่งกลุ่มการเรียนรู้ (clustering) เครื่องจักรพยายามที่จะจัดโครงสร้างและจัดเรียงข้อมูลที่ถูกป้อนเข้ามาตามลักษณะบางอย่าง ตัวอย่างเช่นเครื่องสามารถ เรียนรู้ได้ว่าเหรียญที่มีสีแตกต่างกันสามารถจัดเรียงตามลักษณะ สี เพื่อจัดโครงสร้างให้มันได้

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

Reinforcement learning เป็น การเรียนรู้ ที่มีหลักการทำงานเสมือนกับการที่มนุษย์เรียนรู้บางสิ่งบางอย่างด้วยการลองผิดลองถูก โดย คอมพิวเตอร์มีปฏิสัมพันธ์กับ สิ่งแวดล้อมที่เปลี่ยนไปตลอดเวลาโดยคอมพิวเตอร์จะต้องทำงานบางอย่าง

นอกจากนี้ การเรียนรู้ของเครื่องยังสามารถแบ่งประเภทของงาน ได้ตาม ข้อมูลขาออก จากระบบที่ เครื่องจักรได้เรียนรู้แล้ว เป็นหลายประเภทดังนี้

1. การแบ่งประเภทข้อมูล (classification) คือ model ต้องมี Target หรือ ตัวแปรที่ใช้วัดเป้าหมาย เป็นตัวตั้งต้นให้เรียนรู้ โดย Target ของ Classification จะเป็นแบ่งออกเป็นกลุ่ม หรือมีลักษณะเป็น Discrete เช่น yes/no, A/B/C เป็นต้น ดังนั้น ในการประเมินผลลัพธ์ที่ได้จาก Classification Model จะสามารถวัดค่าความแม่นยำ หรือ Accuracy ได้ เช่น ใช้ Confusion matrix

2. การวิเคราะห์การถดถอย (regression) ใช้หลักการเดียวกับการแบ่งประเภทข้อมูล โดย ข้อมูลที่มีความต่อเนื่อง ไม่ได้แบ่งเป็นกลุ่มใช้เพื่อวิเคราะห์ข้อมูลที่มี Range ข้อมูลค่อนข้างกว้าง เช่น Weather Forecast หรือ การพยากรณ์สภาพอากาศ

3. การแบ่งกลุ่มข้อมูล (clustering) เป้าหมายคือการแบ่งข้อมูลขาเข้าเป็นกลุ่มๆ โดยอัลกอริทึมจะไม่ทราบกลุ่มดังกล่าวล่วงหน้า โดย Model จะจัดกลุ่มให้ โดยใช้ Attribute หรือคุณลักษณะ เป็นตัวแจกแจง แนวทางในการจัดกลุ่ม มีทั้งแบบ Connectivity models, Centroid models และ Distribution models

2.1.3 การเรียนรู้เชิงลึก (Deep learning) เป็นสาขาของการเรียนรู้ของเครื่อง พื้นฐานของการ เรียนรู้เชิงลึกคือ อัลกอริทึมที่พยายามจะสร้างแบบจำลองเพื่อแทนความหมายของข้อมูลในระดับสูง โดยการสร้างสถาปัตยกรรมข้อมูลขึ้นมาที่ประกอบไปด้วยโครงสร้างย่อย ๆหลายอัน และแต่ละอันนั้น ได้มาจากการแปลงที่ไม่เป็นเชิงเส้นคำว่า deep learning มีที่มาจากการที่ใช้ layer ของ neural network หลายอันๆมาต่อกัน(มี Hidden layer มากกว่า 2 layer ก็ถือเป็น deep learning แล้ว) เนื่องจาก layer เหล่านี้เป็นโครงสร้างที่ถูกจัดเก็บแบบเป็นกองซ้อน (stack) จึงเปรียบได้ว่า layer ที่จะนวนเยอะๆ ก็จะทำให้มีโครงสร้างที่ลึก(deep) ยิ่งขึ้น โดย Hidden layer ของแต่ละ layer จะเปรียบเสมือนว่าประกอบด้วย เซลล์ประสาท (neural) จำนวนมาก ซึ่งมีหน้าที่ในการประมวลผลรับข้อมูลจาก layer ที่อยู่เหนือกว่า และส่งข้อมูลที่ประมวลผลเสร็จแล้วไปยัง layer ที่อยู่ล่างกว่า , ข้อดีของการส่งข้อมูลแบบนี้ก็คือ layer แต่ละ layer สามารถที่จะมี ค่าถ่วงน้ำหนัก (weight), ค่าความเอนเอียงของข้อมูล (bias) และ วิธีการประมวลผลทางคณิตศาสตร์ (activation function) ที่

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

เป็นอิสระต่อกันได้ ยิ่งป้อนข้อมูลให้กับ model มากเท่าไหร่ , layer แต่ละ layer ก็จะสามารถสกัด feature ที่มีความซับซ้อนมากยิ่งขึ้น โดย Deep Learning จะเกิดการเรียนรู้ขึ้นจาก 2 เฟส

เฟสแรกคือ การประยุกต์ใช้ การแปลงแบบไม่เป็นเชิงเส้น (nonlinear transformation) กับข้อมูลที่ได้รับ (input) ได้ผลลัพธ์ (output) ออกมาอยู่ในรูปของโมเดลทางสถิติ (statistical model)

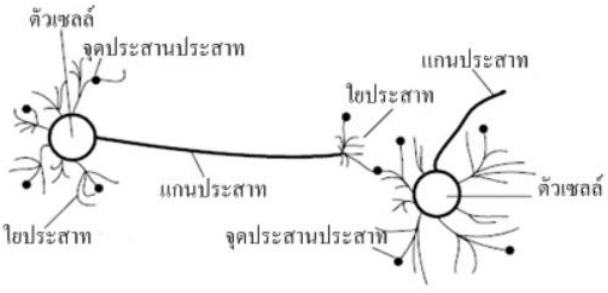
เฟสที่สอง คือ การนำโมเดลมาผ่านวิธีการทางคณิตศาสตร์ อย่าง derivative หรือการดิฟ

โดยทั้ง 2 เฟสนี้จะถูกทำซ้ำแล้วซ้ำเล่าจนกว่าจะได้มาซึ่งโมเดลที่ความแม่นยำ(accuracy)ในระดับที่น่าพึงพอใจ ซึ่ง การทำซ้ำๆของ 2 เฟสนี้ มีชื่อว่า iteration

ชนิดของโครงข่าย deep learning (Types of Deep Learning Networks)

- โครงข่ายประสาทเทียม (Artificial Neural Network: ANN)
- โครงข่ายแบบวนซ้ำ (Recurrent neural networks : RNN)
- โครงข่ายประสาทแบบคอนโวลูชัน (Convolutional neural networks :CNN)

โครงข่ายประสาทเทียม (Artificial Neural Network: ANN) มักจะเรียกสั้นๆ ว่า ระบบโครงข่ายประสาท (Neural Network หรือ Neural Net) เป็นศาสตร์แขนงหนึ่งของทางด้านปัญญาประดิษฐ์ (AI) มีรูปแบบโครงสร้างและการทำงานของการประมวลผลเหมือนกับสมองของสิ่งมีชีวิตซึ่งมีปรับเปลี่ยนตัวเองต่อการตอบสนองของอินพุตตามกฎของการเรียนรู้ (learning rule) โครงข่ายประสาทเทียมได้ถูกพัฒนาขึ้นโดยอาศัยหลักการทำงานของสมองมนุษย์ ซึ่งสมองประกอบด้วยหน่วยประมวลผลพื้นฐานที่เรียกว่า นิวรอล (เซลล์ประสาท หรือ neuron) ภายในสมองประกอบด้วยนิวรอลจำนวนมหาศาล (ประมาณ 1011) และมีจุดต่อจำนวนโครงข่ายประสาทประกอบขึ้นด้วยส่วนสำคัญ 3 ส่วน คือ ใยประสาท (dendrite) ตัวเซลล์ (soma) และแกนประสาท (axon) ในแต่ละโครงข่ายประสาทจะเชื่อมต่อกันโดยจุดประสานประสาท (synapse)



ภาพที่ 2.2 การทำงานของสมองมนุษย์

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

การประมวลผลต่าง ๆ เกิดขึ้นในหน่วยประมวลผลย่อย เรียกว่า โหนด (node) ซึ่งโหนดเป็นการจำลองลักษณะการทำงานมาจากเซลล์การส่งสัญญาณ ระหว่างโหนดที่เชื่อมต่อกัน จำลองมาจากการเชื่อมต่อของใยประสาท และแกนประสาทในระบบประสาทของสมองมนุษย์ ภายในโหนด จะมีฟังก์ชันกำหนดสัญญาณส่งออกที่เรียกว่า ฟังก์ชันการแปลง (transfer function) ซึ่งทำหน้าที่เปรียบเสมือนกระบวนการทำงานในเซลล์ แสดงในรูปที่ 2 โครงข่ายประสาทเทียมประกอบด้วย 5 องค์ประกอบ ดังนี้

1. ข้อมูลอินพุต (input) เป็นข้อมูลที่เป็นตัวเลข หากเป็นข้อมูลเชิงคุณภาพ ต้องแปลงให้อยู่ในรูปเชิงปริมาณที่โครงข่ายประสาทเทียมยอมรับได้

2. ข้อมูลเอาต์พุต (output) คือ ผลลัพธ์ที่เกิดขึ้นจริง จากกระบวนการเรียนรู้ของโครงข่ายประสาทเทียม

3. ค่าน้ำหนัก (weights) คือ สิ่งที่ได้จากการเรียนรู้ของโครงข่ายประสาทเทียม หรือเรียกอีกอย่างหนึ่งว่า ค่าความรู้ (knowledge) ค่านี้จะถูกเก็บเป็นทักษะเพื่อใช้ในการจดจำข้อมูลอื่น ๆ ที่อยู่ในรูปแบบเดียวกัน

4. ฟังก์ชันผลรวม (Summation function: S) เป็นผลรวมของข้อมูลป้อนเข้า (pi) และค่าน้ำหนัก (wi)

5. ฟังก์ชันการแปลง (transfer function) ฟังก์ชันการแปลง เป็นส่วนที่ทำหน้าที่รวมค่าเชิงตัวเลขจากเอาต์พุตของนิวรอล แล้วทำการตัดสินใจว่าจะส่งสัญญาณเอาต์พุตออกไปในรูปใด ฟังก์ชันการแปลงสามารถเป็นได้ทั้งแบบเชิงเส้นหรือไม่เป็นเชิงเส้น การเลือกใช้ฟังก์ชันการแปลงจะขึ้นอยู่กับลักษณะของระบบ ที่นำเอาโครงข่ายประสาทเทียมไปประยุกต์ใช้

โครงข่ายแบบวนซ้ำ (Recurrent neural networks : RNN) Recurrent neural networks คือ neural networks หลายเลเยอร์ที่สามารถเก็บ(store)ข้อมูล(information)ไว้ที่ node จึงทำให้สามารถรับข้อมูลเป็นแบบลำดับ (data sequences) และให้ผลลัพธ์ออกเป็นลำดับของข้อมูลได้ โดยมีประโยชน์อย่างมากในข้อมูลที่มีความต่อเนื่อง เช่น time series ข้อมูลเสียง ข้อความ หรือแม้แต่รูปภาพเองก็ตาม ในแต่ละ node ของ RNNs จะมีข้อมูลขาเข้าสองอย่างอันได้แก่ input ณ node นั้น ๆ และผลลัพธ์ที่ได้จากการคำนวณใน node ก่อนหน้า ซึ่งทั้งสองข้อมูลจะถูกนำมารวมเข้าด้วยกันและออกผลลัพธ์มาเป็นสองทางคือ ผลลัพธ์ที่ออก ณ node นั้น ๆ และออกเพื่อไปเข้าเป็นข้อมูลขาเข้าใน node ถัดไป

ข้อดี ของ RNNs คือ มันมีการใช้ข้อมูลก่อนหน้าในการทำนายสิ่งที่อาจจะเกิดขึ้นในอนาคต นั่นหมายถึง อะไรที่เคยเกิดขึ้นในอดีตย่อมส่งผลต่อเหตุการณ์ที่จะเกิดขึ้นในอนาคตด้วย

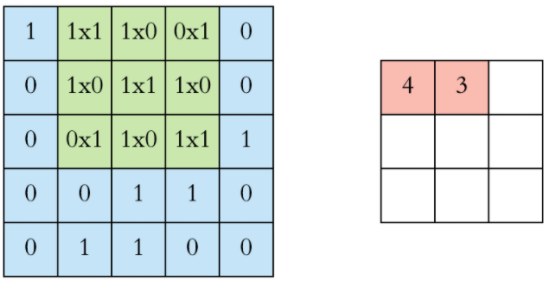
DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

ข้อเสียของ RNNs ที่ต้องพบเจอคือ มันสามารถดูย้อนกลับได้แค่เพียงในช่วงระยะเวลาสั้น ๆ เท่านั้น ซึ่งปัญหาหลัก ๆ ของ RNNs นั้นเกิดมาจากค่า gradient ที่เริ่มน้อยลงในข้อมูลที่มีความยาวมากขึ้น จนแทบจะไม่สามารถเห็นความเปลี่ยนแปลงของ gradient ได้เลย ซึ่งปัญหานี้ถูกเรียกว่า Vanishing Gradient Problem

โครงข่ายประสาทแบบคอนโวลูชัน (Convolutional neural networks :CNN)

CNN คือ neural network หลายเลเยอร์ที่มีโครงสร้างเฉพาะตัว โดยถูกออกแบบมาเพื่อการเพิ่มความสามารถในการสกัดเอา feature ที่มีความซับซ้อนมากยิ่งขึ้นจากข้อมูลโดย CNN นั้นตอบโจทย์ปัญหาประเภทการรับรู้ (perceptual tasks) CNN มักจะถูกใช้เพื่อการสกัด feature จากข้อมูลประเภทที่ไม่ค่อยเป็นระเบียบหรือไม่ได้มีโครงสร้างเป็นรูปแบบเฉพาะตัว (unstructured data) อย่างเช่น รูปภาพ (image)

Feature Extraction จะเริ่มจากการกำหนดค่าใน ตัวกรอง (filter) หรือ เคอร์เนล (kernel) ที่ช่วยดึงคุณลักษณะที่ใช้ในการรู้จำวัตถุออก โดยตัวกรอง/เคอร์เนลอันหนึ่งจะดึงคุณลักษณะที่สนใจออกมาได้หนึ่งอย่าง ดังนั้นจึงจำเป็นต้องมีตัวกรองหลายตัวกรองด้วย เพื่อหาคุณลักษณะทางพื้นที่หลายอย่างประกอบกัน สำหรับ Filter ของภาพดิจิทัลนั้น โดยปกติแล้วจะเป็นตารางสองมิติที่มีขนาดตามมี พื้นที่ย่อยๆที่เราอยากพิจารณา ตำแหน่งตรงกลางที่มีกรอบสีฟ้าคือ Anchor ที่เอาไว้ทาบบนพิกเซล ของภาพข้อมูลเข้า ตัวกรองจะถูกทาบลงในพิกเซลแรกที่เอาไว้ทาบบนพิกเซลของภาพข้อมูลเข้า ตัวกรองจะถูกทาบลงในพิกเซลแรกของภาพข้อมูลเข้า จากนั้นจะถูกเลื่อนไปทาบ บนพิกเซลอื่นในภาพทีละพิกเซลจนครบทุกพิกเซลในภาพ เราอาจจะไม่ทาบตัวกรองบนพิกเซลที่อยู่ใกล้กรอบภาพ เพราะตัวกรองจะล้นออกไปนอกภาพ เมื่อเราเลื่อนตัวกรองไปเรื่อยๆจนครบทุกพิกเซลที่สามารถเลื่อนได้ในภาพ สิ่งที่เราได้นั้นจะเป็นสิ่งที่เรียกว่า ผังคุณลักษณะ (feature map)

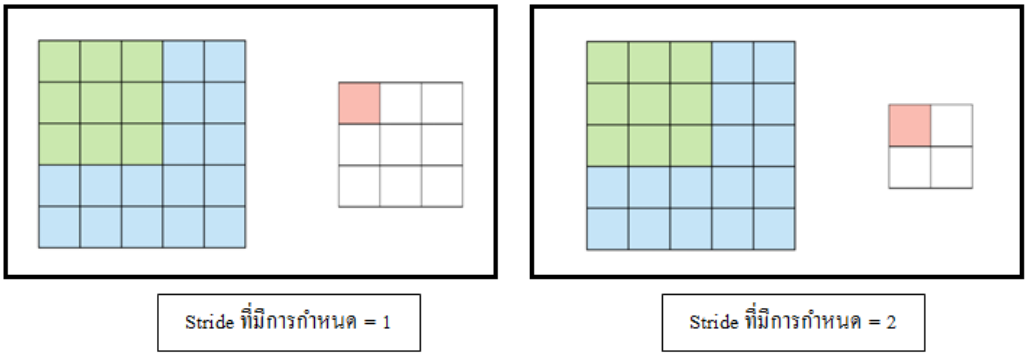


ภาพที่ 2.3 feature map

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

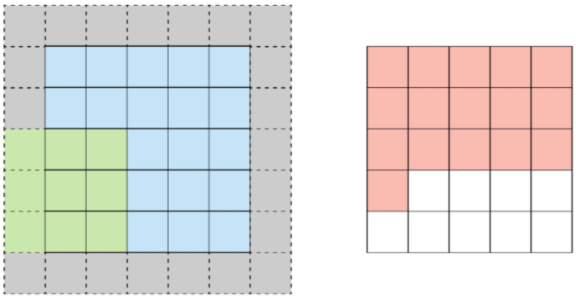
Stride และ Padding

Stride เป็นตัวกำหนดว่าเราจะเลื่อนตัวกรอง (filter) ไปด้วย Step เท่าไร สามารถกำหนดค่าของStride ให้มากขึ้นก็ได้ ถ้าเราต้องการให้การคำนวนหาคุณลักษณะมีพื้นที่ทับซ้อนกันน้อยขึ้นแต่การกำหนดค่าของ Stride ที่มากขึ้นจะทำให้เราได้ผังคุณลักษณะ (feature map) ที่มีขนาดเล็กลง



ภาพที่ 2.4 Stride และ Padding

Padding จากรูปด้านล่างเราจะพื้นที่สีเทารอบๆ Input พื้นที่เหล่านี้เป็นพื้นที่ที่เรามักเติมเข้าไป โดยอาจจะเป็นเติม 0 หรือค่าต่างๆเข้าไป เพื่อให้เวลาในการทำ CNN นั้น Feature Map ที่ได้ยังคงมีขนาดเท่ากับ Input

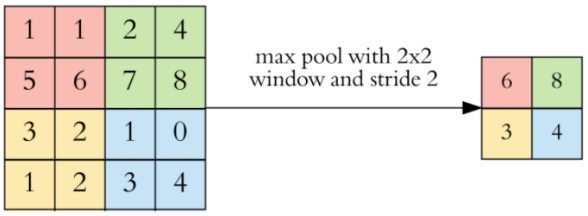


ภาพที่ 2.4 (ต่อ) Stride และ Padding

ปัญหา Input ที่อยู่ตามขอบภาพจะส่งผลต่อการตัดสินใจบางอย่างดังนั้นจึงจำเป็นต้องเก็บคุณลักษณะตามขอบของรูปภาพไว้

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

Pooling layer หน้าที่ของ Pooling layer คือการสกัดเอาส่วนที่สำคัญที่สุดของข้อมูล และเพิ่มประสิทธิภาพการประมวลผลให้รวดเร็วยิ่งขึ้น กลไกของ Pooling layer นั้นเรียบง่ายมาก คือการสกัดเอาเฉพาะค่าสูงสุดของ Grid เก็บไว้ใน Output โดย Pooling มีสองประเภทหลักที่นิยมกันคือ max pooling และ mean pooling



ภาพที่ 2.5 Pooling layer

Max Pooling เป็นตัวกรองแบบหนึ่งที่หาค่าสูงสุดในบริเวณที่ตัวกรองทาบอยู่มาเป็นผลลัพธ์ โดยเราจะเตรียมตัวกรองในลักษณะเดียวกับการทำ Feature Extraction ของ CNN มาทาบบนข้อมูลแล้วเลือกค่าที่สูงที่สุดบนตัวกรองนั้นมาเป็นผลลัพธ์ใหม่ และจะเลื่อนตัวกรองไปตาม Stride ที่กำหนดไว้ โดยขนาดตัวกรองของการทำ max pooling จะนิยมเรียกกันว่า pool size

2.3 ความรู้เกี่ยวกับการเรียนรู้เชิงลึก สำหรับ Vision System

2.3.1 Computer vision คือการทำให้computer สามารถเลียนแบบการมองเห็นของมนุษย์ เพื่อจำแนกวัตถุหรือ สร้างขึ้นมาใหม่ โดยทำการรับ Input จาก อุปกรณ์ เช่น กล้อง หรือ เซนเซอร์ต่างๆ เพื่อจำแนกวัตถุ หรือ ลักษณะต่างๆ โดยสามารถจำแนกประเภทวัตถุหลักๆ เป็น 2 ประเภท

1) Low-Level จำแนก ขอบ, เส้น, มุม ในรูปภาพ

2) High-Level จำแนกชิ้นส่วนต่างๆของรูปภาพ

ประเภทของ Computer vision ที่ใช้ในปัจจุบันมี 5 รูปแบบ

1.Image Classification การให้computer เรียนรู้และจำแนกภาพจาก Label (Supervise Learning) โดยงานที่นำมาประยุกต์ใช้ เช่น การจำแนกภาพปอดที่เป็นมะเร็ง

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

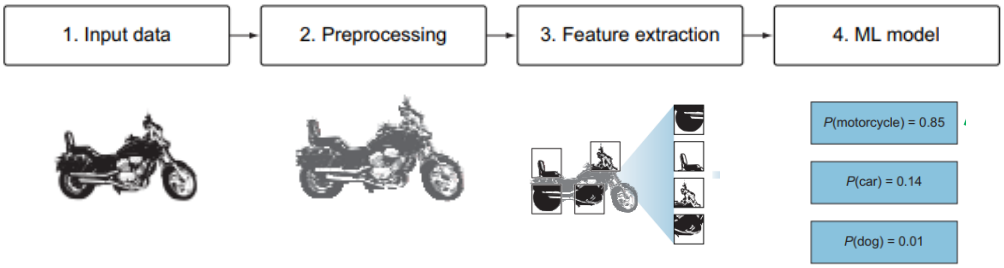
2. Object detection การที่คอมพิวเตอร์มีความสามารถเลียนแบบมนุษย์ในการการ จำแนกภาพที่มองเห็นและแยก แยะ ได้ละเอียดมากขึ้น โดย Systems ที่ปัจจุบันใช้ เช่นYolo (you only look once) SSD(Single shot detector) , Faster R-CNN

3. Generating Art คอมพิวเตอร์สามารถcopy style จาก ภาพ A ไป B

4. Creating Images คอมพิวเตอร์สามารถสร้างภาพขึ้นมาจากภาพต้นแบบโดยใช้ machine learning

5. Face Recognition คอมพิวเตอร์สามารถระบุหรือติดตามใบหน้าคนโดยจะแบ่งเป็น Face Identification และ Face Verification

Computer vision pipeline



ภาพที่ 2.6 Computer vision pipeline

1. ขั้นตอนแรก คอมพิวเตอร์ จะรับ input เข้ามา จาก อุปกรณ์เช่น กล้อง ที่จับภาพหรือ วิดีโอ

2. ขั้นตอนที่สอง แต่ละภาพต้องทำการเปลี่ยนรูปแบบ(preprocessing) จุดประสงค์เพื่อให้ภาพ มีพื้นฐานเหมือนกัน โดยส่วนใหญ่ จะทำ การปรับขนาดภาพ(resizing) , การทำให้ภาพชัดขึ้น(blurring) , การหมุนภาพ (rotating) , การเปลี่ยนรูปร่าง หรือ การเปลี่ยนจากภาพสีให้เป็นขาวดำ

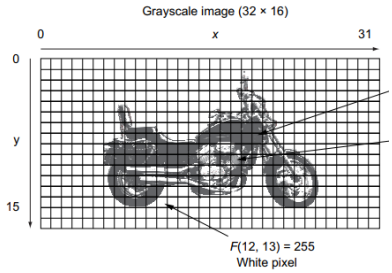
3. ขั้นตอนที่สาม การจับคุณลักษณะ(extract feature) กระบวนการนี้จะทำการจับรูปแบบคุณลักษณะของภาพ เช่น รูปทรง , สี โดยจะตรวจสอบและ list คุณลักษณะที่มีความเฉพาะ ออกมา

4. ขั้นตอนที่สี่ Feature จะถูกนำเข้าไปใน การจัดกลุ่ม(classification model) เพื่อทำกระบวนการแยกแยะว่าภาพนี้คือ อะไร โดยจะใช้ feature ในการตัดสินใจภาพนั้นๆ

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

Image Input

การนำรูปภาพเข้ามาจะมี 2 ตัวแปร คือ X และ Y โดยในทุกๆภาพจะมี Pixel ที่มีความหนาแน่นของเม็ดสีอยู่ โดยจะมีความสว่างของแต่ละจุดไม่เท่ากัน



ตัวอย่างของภาพเป็นภาพ ขนาด กว้าง 32 pixel ยาว 16 pixel ภาพทั้งหมด 512 (32 × 16) pixels. เป็น ภาพขาวดำ (greyscale) มีค่าอยู่ระหว่าง 0-255 pixel 0 จะแทนจุดที่มืดที่สุด ส่วน 255จะแทนจุดที่สว่างที่สุด

ภาพที่ 2.7 Image Input

ใน ภาพสี ค่า pixel จะแทนค่าด้วย RGB system, มี 3 ค่า red , green ,blue s

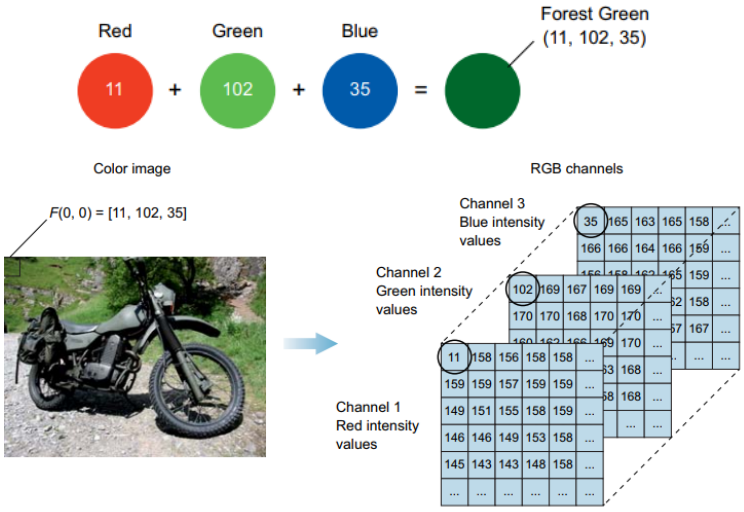
Color image in RGB => F(x, y) = [red (x, y), green (x, y), blue (x, y)]

08	02	22	97	38	15	00	40	00	75	04	05	07	78	52	12	50	77	91	08
49	49	99	40	17	81	18	57	60	87	17	40	98	43	69	46	04	56	62	00
81	49	31	73	55	79	14	29	93	71	40	67	53	99	30	03	49	13	36	65
52	90	95	23	04	60	11	42	69	24	48	56	01	32	54	71	37	02	34	91
22	31	14	71	51	67	43	59	41	92	34	54	22	40	40	28	44	33	13	80
24	47	32	60	99	03	45	02	44	75	33	53	78	36	64	20	35	09	12	80
32	98	81	28	64	23	67	10	26	38	40	67	59	54	70	66	18	38	64	70
47	24	20	68	02	62	12	20	95	63	94	39	63	04	49	91	44	49	94	21
24	55	58	05	66	73	99	26	97	17	78	78	94	83	14	88	34	89	63	72
21	36	23	09	75	00	74	44	20	45	35	14	00	41	33	97	34	31	33	95
78	17	53	28	22	75	31	67	15	94	03	80	04	42	16	14	09	53	56	92
16	39	05	42	96	35	31	47	55	58	88	24	00	17	54	24	34	29	85	57
84	56	00	48	35	71	89	07	05	44	44	37	44	40	21	58	51	54	17	58
19	80	61	68	05	94	47	49	28	73	92	13	86	52	17	77	04	09	55	40
04	52	08	83	97	35	99	14	07	97	57	32	16	26	26	79	33	27	98	44
04	36	68	81	57	62	20	72	03	16	33	67	46	55	12	32	43	93	53	69
04	42	14	73	38	25	39	11	24	94	72	18	06	46	29	32	40	62	74	36
20	49	34	41	72	30	23	88	34	62	99	69	82	47	59	85	74	04	34	24
20	23	35	29	78	31	90	01	74	31	49	71	48	86	81	14	23	57	05	54
01	70	54	71	83	51	54	49	16	92	33	48	61	43	51	01	89	19	67	48

ภาพที่ 2.8 การมองภาพของคอมพิวเตอร์

คอมพิวเตอร์จะมองรูปภาพในรูปแบบ 2D matrix ของ pixel โดยจะคำนวณในรูปแบบ spectrum ถ้าเป็นภาพขาวดำจะมีความหนาแน่นของเม็ดสี แค่สีเดียว แต่ถ้าเป็นภาพสี จะมีความหนาแน่นของเม็ดสี 3สี (red ,green , blue) เวลา คอมพิวเตอร์มองภาพสี จะมองเป็น 3D matrix เช่น dimension 700 × 700 จะเป็น (700, 700, 3) matrix แรก จะมอง red channel matrix ถัดๆไปจะเป็น green และ blue โดยในแต่ละpixel ของสี จะอยู่ระหว่าง 0 -255 เช่น shade สีเขียวดั่งรูปภาพข้างล่าง

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์



ภาพที่ 2.9 การมองภาพสี

การเปลี่ยนรูปแบบของภาพ(Image preprocessing) การทำ Image preprocessing ลดความซับซ้อน และ เพิ่ม accuracy ให้กับ algorithm โดยมีหลายขั้นตอนดังนี้

1. การแปลงภาพสี ให้เป็นภาพขาวดำ(grey scale)เพื่อลดความซับซ้อนในการประมวลผล ถ้าหากจุดประสงค์ของงานไม่จำเป็นต้องใช้ภาพสี เพื่อจำแนกคุณลักษณะบางอย่าง การทำ grey scaleจะช่วยลดปริมาณในการ ประมวลผลให้ทำงานได้เร็วขึ้น

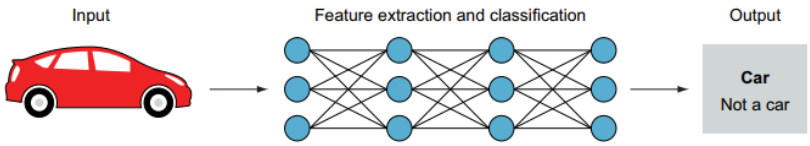
2. การทำภาพให้มีมาตรฐานเหมือนกัน (standardizing Image) วิธีนี้จะช่วยทำให้ model เรียนรู้ได้เร็วขึ้น และ ลดความผิดพลาด ของ out put ที่จะออกมา

3. การทำให้ภาพเพิ่มขึ้น (data augmentation) วิธีนี้ คือการนำข้อมูลที่มีอยู่แต่อาจไม่เพียงพอให้ model เรียนรู้ feature มาปรับภาพ เช่น การหมุนของภาพ , การขยายภาพ เพื่อทำให้ model สามารถเรียนรู้ได้อย่างมีประสิทธิภาพ

Feature ใน computer vision

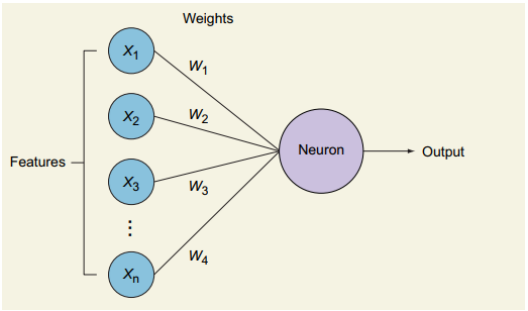
สิ่งสำคัญคือต้องทำ หา feature ที่ดี เพื่อนำไปสร้าง model ต่อได้ โดยปัจจุบันจะใช้ Automatic extract เพื่อให้เกิดความแม่นยำ และลดการ bias ในการเลือก feature โดยจะใช้ Deep learning ในการ extract feature และ สกัดแต่ feature ที่สำคัญเท่านั้น

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์



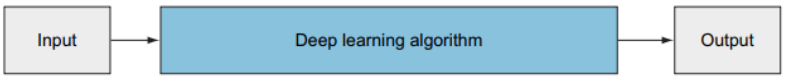
ภาพที่ 2.10 Feature extraction

โดยในภายใน การสกัดเพื่อให้ได้แต่ feature ที่สำคัญ จะใช้การ scope ทั้งหมด และ สุ่มค่าน้ำหนักของ feature ในระหว่างการ train process neural network จะทำการปรับค่า weight เพื่อให้ได้แต่ feature ที่สำคัญและเลือก feature ที่มีผลต่อ output



ภาพที่ 2.11 ภายในFeature extraction

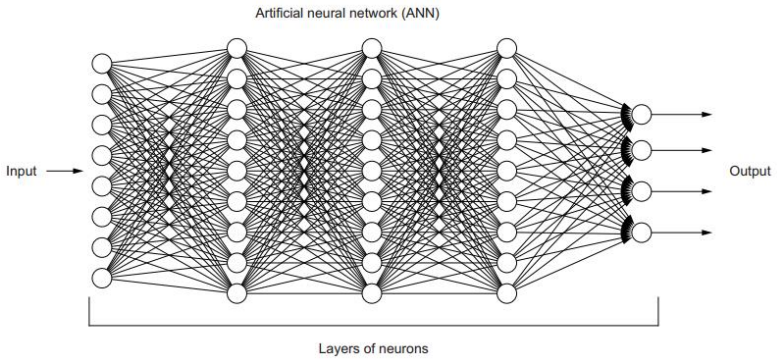
2.2.2 Deep Learning



ภาพที่ 2.12 Deep learning flow

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

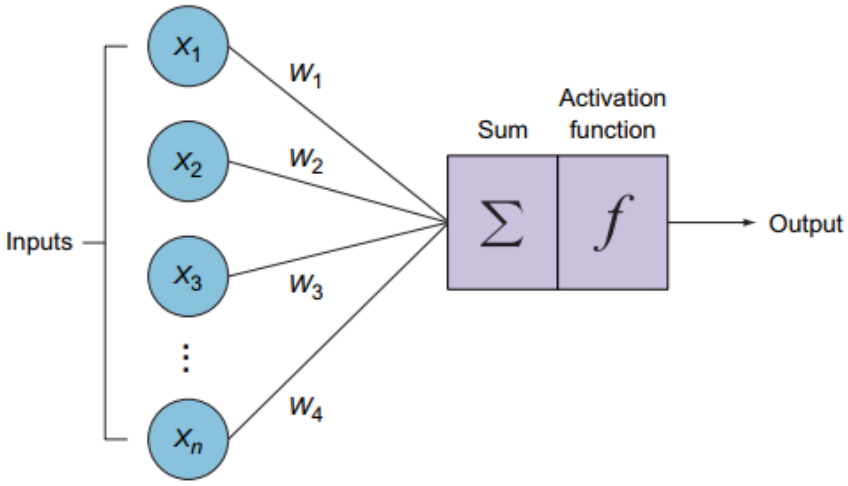
Deep Learning: Artificial neural network (ANN)



ภาพที่ 2.13 กระบวนการ ANN

โครงสร้าง ANN จะมี neuron ที่หลากหลาย โดยในแต่ละ Layer ก็จะทำ การคำนวณ และ ทำนาย ผลลัพธ์ออกมา โดน จะเรียกว่า Multilayer perceptron ซึ่งมีการประกอบหลายๆ layer เข้าด้วยกัน ซึ่ง perceptron เปรียบเหมือน โครงสร้างประสาทที่ใช้ในการเชื่อมต่อกัน

กระบวนการใน single perceptron มีดังนี้



ภาพที่ 2.14 กระบวนการ Single perceptron

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

1. Input Vector: Feature Vector จะถูกนำเข้าไปใน Neuron

2. Weight Vector: เมื่อ x ผ่านเข้ามาในจุด weight ค่า weight จะถูกคำนวญโดยหาความสำคัญระหว่างจุดของ ข้อมูล โดยจะรวม input ทุกตัวมาคูณกับ ค่าน้ำหนัก และ เพิ่ม ค่า Bias เข้าไป

$$z = \sum x_i \cdot w_i + b \text{ (bias)}$$

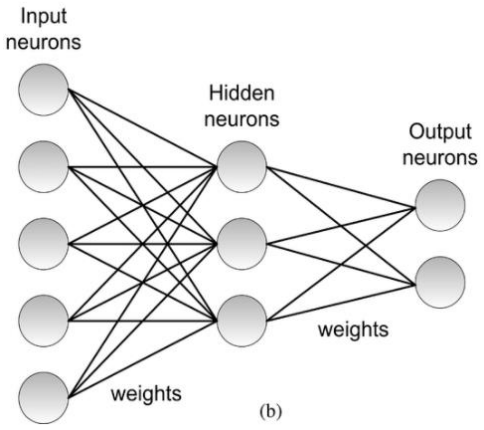
$$z = x_1 \cdot w_1 + x_2 \cdot w_2 + x_3 \cdot w_3 + \cdots + x_n \cdot w_n + b$$

ภาพที่ 2.15 Weight Vector

3. Neuron Functions: คำนวณค่า ภายใน neuron ให้เหมาะสมกับผลลัพธ์

4. Output: ควบคุม ชนิดของ activation function โดยเลือกสำหรับ network activation function เปรียบเหมือน กระบวนการตัดสินใจ โดยภายในกระบวนการ จะเลือกค่า weight ที่เหมือนกันมารวมกัน จาก Input ก่อนหน้า และ neuron จะตัดสินใจเลือก ถ้าค่าน้ำหนักที่ได้ สูงกว่าเกณฑ์ที่กำหนดไว้

กระบวนการใน Multilayer perceptron



ภาพที่ 2.16 Multilayer perceptron

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

สิ่งที่เพิ่มเข้ามาคือ Hidden layer โดย hidden layer เปรียบเหมือน กระบวนการในการเรียนรู้ feature ของ model โดนภายในจะเรียนรู้จากในแต่ละจุดที่ซับซ้อน จนหาว่า layer ตัวไหนที่เหมาะสมกับข้อมูล ในการทำ multilayer perceptron สิ่งสำคัญคือ การกำหนด จำนวน neuron และจำนวน layer และ หา hyper parameter มาปรับ เพื่อให้สามารถทำงานได้ดีกับ neural network

ข้อควรระวัง การ Set neural network hyperparameter

1. จำนวนของ Hidden Layer ยิ่งมีจำนวน neuron เยอะเท่าไหร่ ยิ่งส่งผลดีต่อการให้ network เรียนรู้ ใน การ training data แต่ต้องระวัง ข้อมูล เกิดการ overfitting

2. Activation function ในงานปัจจุบัน จะใช้ ReLU ใน hidden layer และ Softmax ใน output layer

3. Error function เอาไว้ใช้วัด ค่าที่ network ทำนายออกมา โดย Mean square error ใช้สำหรับปัญหาด้าน regression ส่วน Cross entropy ใช้สำหรับ Classification

4. Optimizer เอาไว้ใช้หาค่าน้ำหนักที่เหมาะสมที่ทำให้เกิดค่า Error น้อยสุด

5. Batch size มีหน้าที่ สุ่มตัวอย่างใน network หลังจาก parameter update เสร็จแล้ว โดยค่าที่ตั้งส่วนใหญ่ตั้งที่ 32

6. Number of epochs จำนวนครั้งในการ training ข้อมูล

7. Learning rate เอาไว้ปรับ Input parameter ในการ tune

Activation function

Rectified linear unit (ReLU) คือ ฟังก์ชันเส้นตรงที่ถูกปรับแก้ Rectified เอาไว้ หาค่า ถ้า input มีค่ามากกว่า 0 จะมีค่าเท่ากับ 1 ในกรณีที่ต่ำกว่า 0 input จะแสดงค่า 0 เท่านั้น input มากกกว่า 0 แสดงว่า มีความสัมพันธ์ กับ output และที่นิยมวิธีนี้ เพราะ Gradient ไม่หาย (ไม่เกิด Vanishing Gradient) ส่งผลให้เราเทรนโมเดลได้เร็วขึ้น โดยมี สมการ function ดังนี้

$$f(x) = \max(0, x) = \begin{cases} 0 & \text{for } x \leq 0 \\ x & \text{for } x > 0 \end{cases}$$

ภาพที่ 2.17 Activation function

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

Softmax function คือ ฟังก์ชันส่วนหนึ่งของ sigmoid function โดยมีหน้าที่ในการจำแนก และ แบ่งกลุ่ม หาความน่าจะเป็น เมื่อมีข้อมูล มากกว่า 2 class หรืออีกความหมาย คือ การรับตัวเลขเข้าไป แล้วแปลงออกมาเป็น Probability ส่วนใหญ่มักวางที่ตำแหน่ง Layer สุดท้าย ของ Neural Network เพื่อให้ Output ออกมาเป็น Probability โดยมี สมการ function ดังนี้

ภาพ $$\sigma(\mathbf{z})i = \frac{e^{z_i}}{\sum j=1^K e^{z_j}} \text{ for } i = 1, \ldots, K \text{ and } \mathbf{z} = (z_1, \ldots, z_K) \in R^K$$

2.18 Softmax function

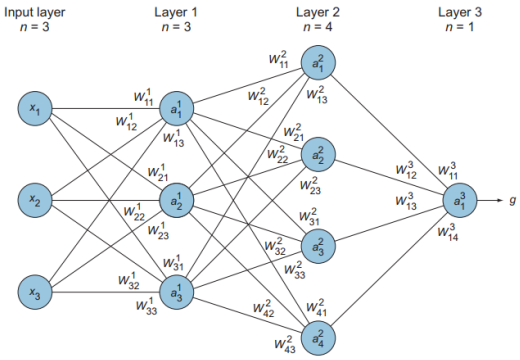
Leaky ReLU คือ ฟังก์ชัน ที่เอามาใช้แก้ปัญหาถ้า input น้อยกว่า 0 จะแสดง เป็น ค่า range ไม่ใช่ 0 โดยจะอยู่ ประมาณ 0.01 เมื่อค่าติดลบ หรืออีกความหมาย Leaky ReLU จะช่วยเพิ่ม range ของ ReLU ออกไปกลายเป็น (-infinity ถึง infinity) โดยมี สมการ function ดังนี้

$$f(x) = \max(0.01x, x)$$

ภาพที่ 2.19 Leaky ReLU

Training Process

1. Feedforward Process คือ กระบวนการ ในการคำนวณระหว่าง การรวมกัน ของ linear และ การนำ activation เข้ามาในกระบวนการ



ภาพที่ 2.20 Feedforward Process

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

ในตัวอย่างมี 3 layer ใน neural network

1. Layer: ใน network ประกอบไปด้วย 3 input feature และ 3 hidden layer กับ 3,4,1 neurons แต่ละ layer

2. Weights และ Bias: ระหว่าง node จะทำการสุ่มค่าน้ำหนัก เพื่อระบุ layer number และ ระบุค่าน้ำหนัก ใน แต่ละจุดที่ได้มีการเชื่อมต่อกัน ส่วน Bias จะถูกทำกระบวนเดียวกันกับ Weight เพื่อให้ค่าแต่ละตัวเกิดการเรียนรู้

3. Activation functions $\sigma(x)$ ในตัวอย่างใช้ Sigmoid Function

4. Node values: เกิดจากการคำนวณค่าน้ำหนัก และ นำ Activation Function มาปรับใช้ ในแต่ละ node

2. Error functions คือ กระบวนการ เอาไว้วัดค่าความผิดพลาด ในกระบวนการการ neural network โดยค่าที่จะอ้างถึง คือ Cost function และ Loss function โดยวิธีที่นิยมจะมี mean squared error และ cross-entropy

Mean square error (MSE) จะใช้ในการหา loss function ของ regression problem โดยมีสมการ คือ

$$E(W, b) = \frac{1}{N}\sum_{i=1}^{N} (\hat{y}_i - y_i)^2$$

ภาพที่ 2.21 Mean square error

MSE จะมีปัญหากับ outlier เป็น อย่างมาก โดยมีความผิดพลาดจากยกกำลังสอง ทุกครั้งก่อน predict ต้อง แน่ใจว่าจะไม่มี outlier ลงเหลืออยู่

Cross-entropy จะใช้ ในการ จัดกลุ่มสำหรับ การหาความน่าจะเป็นของ 2 กลุ่ม มีสูตรดังนี้

$$E(W, b) = -\sum_{i=1}^{m} \hat{y}_i \log(p_i)$$

ภาพที่ 2.22 Cross-entropy

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

1. Error optimization เป็นวิธีการที่ทำให้ หา parameter ที่เหมาะสมที่สุด สำหรับค่า error ที่ต่ำที่สุด หรือ อีกความหมายคือ การกำหนดกรอบปัญหาเพื่อเพิ่มหรือลดคุณค่าบางอย่าง มี 2 วิธี 1) Gradient descent 2) Backpropagation

1. Gradient descent มีหลายเทคนิค Batch gradient descent (BGD), Stochastic gradient descent (SGD) และ Mini-batch GD (MB-GD)

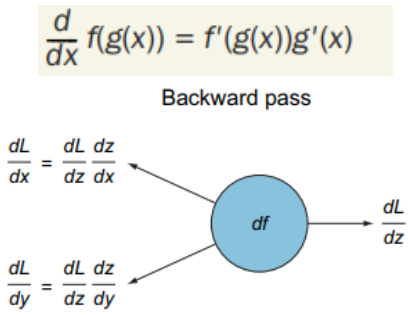
Batch gradient descent (BGD) คือการนำ ค่าเฉลี่ยของการไล่ระดับสีของตัวอย่าง training data ทั้งหมดแล้วใช้ค่าเฉลี่ยการไล่ระดับสีเพื่ออัปเดตพารามิเตอร์ โดยค่า Error จะ อัปเดตโมเดล หลังจากประเมินตัวอย่างการฝึกอบรมทั้งหมดแล้วเท่านั้น

Stochastic gradient descent (SGD) คือ อัลกอริทึมที่อัพเดทค่าพารามิเตอร์ในทุกๆชุดข้อมูลฝึกฝน เป็นอัลกอริทึมที่ค่อนข้างไว อัพเดทแค่ครั้งเดียวต่อการเทรน 1 รอบ โดยในทุกๆครั้งที่มีการอัพเดท ค่าพารามิเตอร์ที่อัพเดทจะมีค่าความแปรปรวนสูงและส่งผลกับค่า Loss function

Mini Batch Gradient Descent คือ การนำข้อดีของ Batch Gradient Descent และ Stochastic Gradient Descent มารวมกันและสำหรับอัลกอริทึมนี้เราจะอัพเดทค่าเป็น ชุด โดยภายในชุดจะประกอบด้วยข้อมูลจำนวน n ข้อมูล

2. Backpropagation

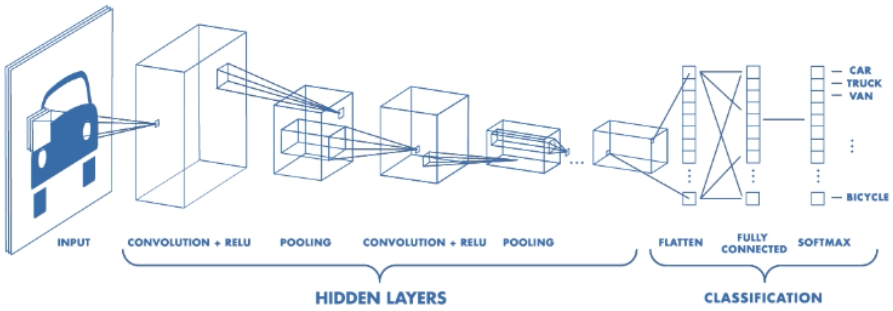
เป็นการค้นหาค่าต่ำสุดของฟังก์ชันข้อผิดพลาดในพื้นที่น้ำหนักโดยใช้เทคนิคที่เรียกว่ากฎเดลต้าหรือการไล่ระดับสี น้ำหนักที่ลดฟังก์ชันข้อผิดพลาดจะถือว่าเป็นวิธีแก้ปัญหาการเรียนรู้ โดยจะใช้ chain rule ในการหา error เมื่อคิดย้อนกลับ



ภาพที่ 2.23 Backpropagation

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

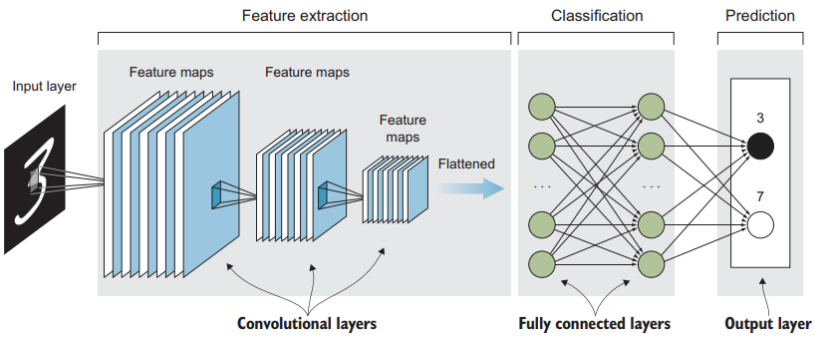
Deep Learning: Convolutional neural networks (CNN)



ภาพที่ 2.24 CNN architecture over view

CNN architecture

หลักการทำงานใน CNN คือ layer แรก จะเรียนรู้ feature พื้นฐาน เช่นเส้นและขอบ และใน layer ถัดไป จะเรียนรู้ feature ที่ซับซ้อนขึ้น เช่นรูปทรงต่างๆ โดย cnn architecture มีรูปแบบแบบเดียวกับ neural network โดยค่าน้ำหนักจะถูกสุ่ม และ เรียนรู้ระหว่าง training network และนำ activation function เข้ามาใช้ หลังจากนั้นทำการคำนวณ ค่า error และสุดท้าย ทำ back propagate ค่า error เพื่อ update ค่า น้ำหนักใหม่อีกรอบ



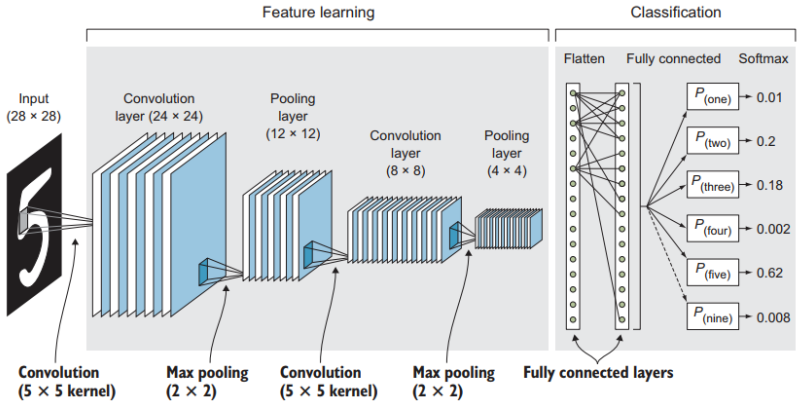
ภาพที่ 2.25 CNN architecture process

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

การเรียนรู้ของ neuron ใน feature จาก input ใน CNN เรียกว่า feature map เป็นการเลือกfilter ก่อนหน้ามาใช้งาน โดยจะทำการ mapping feature ต่างๆที่เจอในรูปภาพ

องค์ประกอบ ใน CNN

1. Convolutional layer (CONV)
2. Pooling layer (POOL)
3. Fully connected layer (FC)



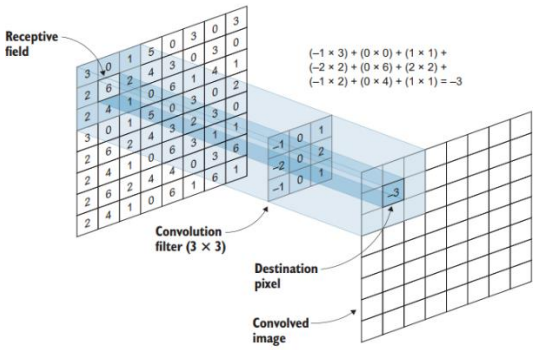
ภาพที่ 2.26 CNN architecture show classification

จากตัวอย่างเป็นการนำ Text เข้ามาใน CNN โดยมีโครงสร้างดังนี้

CNN architecture: INPUT ⇒ CONV ⇒ RELU ⇒ POOL ⇒ CONV ⇒ RELU ⇒ POOL ⇒ FC ⇒ SOFTMAX

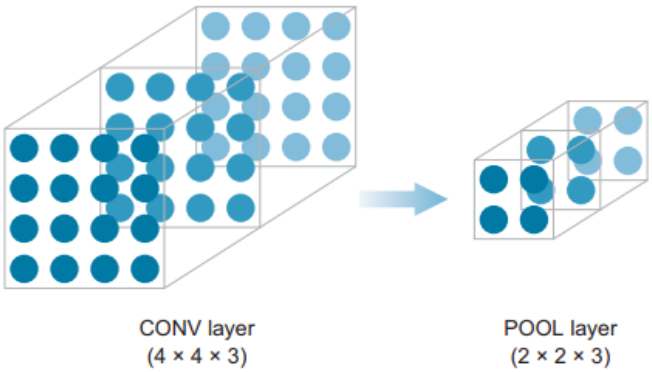
Convolutional layers (CONV) คือ โครงสร้างที่ทำหน้าที่คล้าย feature finder โดยจะทำการเลื่อนช่อง pixel to pixel โดยภายในจะมี filter หรือ เรียกว่า kernel มีหน้าที่ในการสุ่มและเรียนรู้ค่าต่างๆใน network

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์



ภาพที่ 2.27 Convolutional layers

Pooling layers or subsampling (POOL) มีหน้าที่ในการลด parameter ลง เพื่อทำให้ความซับซ้อนลดลง โดยที่ใช้คือ max pooling

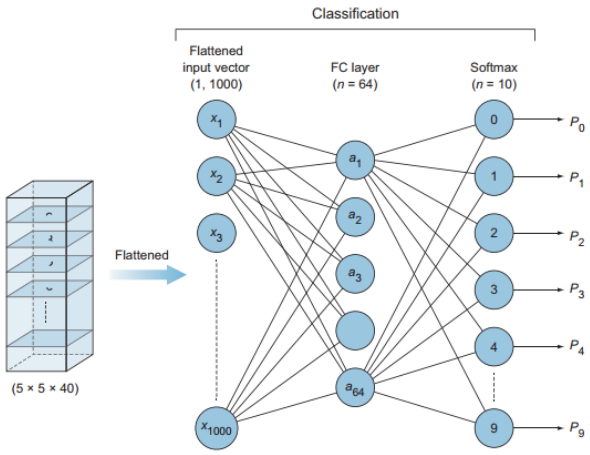


ภาพที่ 2.28 Pooling layers

ประโยชน์ของการใช้ pooling layers คือ เก็บ แต่ feature ที่สำคัญหลักๆไว้ หรืออีกความหมายคือ การ บีบอัดรูปภาพให้เล็กลง

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

Fully connected layers (FC) การนำ multilayer perceptron (MLPs) มาใช้งาน โดยจุดประสงค์ให้ทุก node ตั้งแต่ layer แรก เชื่อมกันทั้งหมด



ภาพที่ 2.29 Fully connected layers

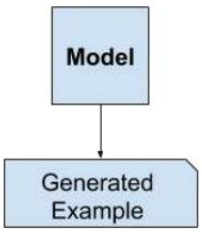
2.4 ความรู้เกี่ยวกับGANs

2.4.1. ภาพรวมของ model

Generative Adversarial Networks (GANs) คือ generative model ที่ใช้ deep learning ในการเรียนรู้ โดย GANs จัดเป็น unsupervised learning โดยจะหา รูปแบบในการเรียนรู้ด้วยตัวเอง โดยหน้าที่ของ model นี้คือ สร้างผลลัพธ์จาก input และเกิดเป็น ผลลัพธ์ใหม่ขึ้นมา

2.4.1.1 Generative model

ภายใน model จะมีการสรุปการกระจายตัว ของ input ทำให้สามารถสร้าง ตัวอย่างขึ้นมาใหม่ ภายใน input distribution ได้

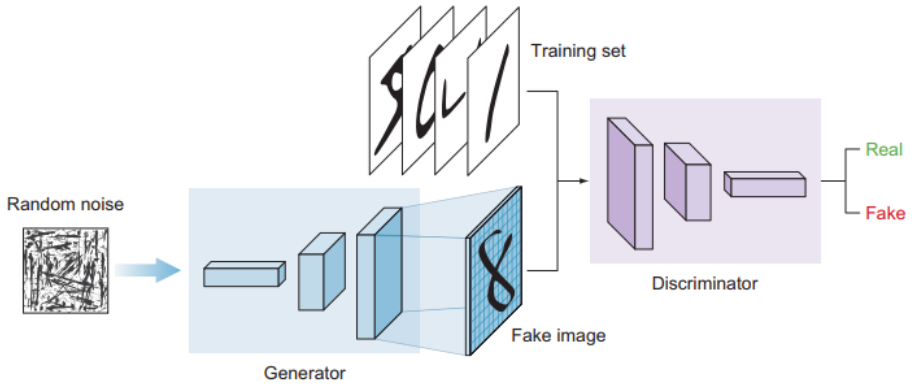


ภาพที่ 2.30 What GANs

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

2.4.1.2 Generative Adversarial Networks (GANs) **ใน GANs มีพื้นฐานมาจาก Generative model โดนรูปแบบโครงสร้างใช้เหมือนกับ Generative model โดยจะมีองค์ประกอบอยู่ 2 ส่วน**

- Generator คือ โมเดลที่ใช้ในการสร้างตัวอย่างขึ้นมาใหม่ โดนการสุ่มความแปรปรวน จาก ข้อมูล ตั้งต้น
- Discriminator คือ โมเดลที่ใช้ในการจำแนกตัวอย่าง จริง หรือ ตัวอย่างปลอม



ภาพที่ 2.31 GANs overview

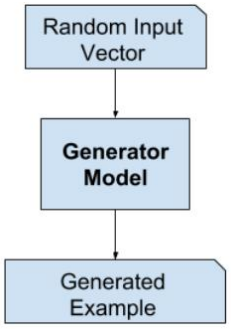
2.4.1.3 GANs architecture process

1. Generator จะสุ่มค่าความแปรปรวน และ แปลงออกมาเป็นภาพ
2. รูปภาพที่ถูกสร้างขึ้นจะถูกนำเข้าไปใน Discriminator และจะจำแนกคู่กับ ชุดข้อมูลจริง
3. Discriminator จะรับทั้งข้อมูล จริง และ ข้อมูลที่ถูกสร้างขึ้นมา (FAKE) และจะแปลงเป็นค่าความน่าจะเป็น โดยมีค่าระหว่าง 0 และ 1 โดยถ้าค่า ออก 1 เป็น ภาพ จริง ถ้าค่าออกเป็น 0 เป็น ภาพที่ถูกสร้างขึ้นมา (FAKE)

2.4.2 Generator model and Discriminator model

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

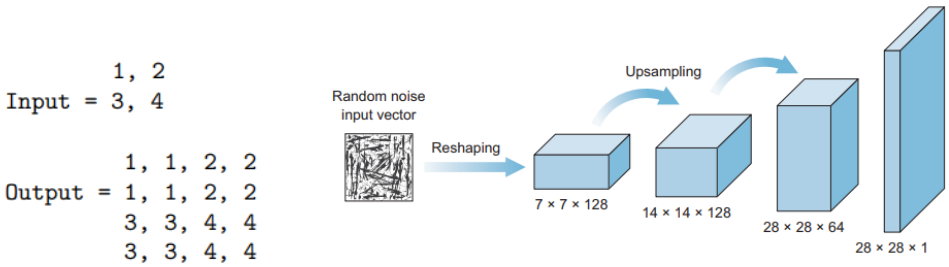
Generative model จะทำการ fixed -length ในการสุ่ม vector ของ input และสร้างตัวอย่างขึ้นมาใหม่ โดย vector จะทำการสุ่ม จาก Gaussian distribution และจะใช้ความแปรปรวนใน กระบวนการ generative ในการกระจายตัวของข้อมูล จะอ้างถึง การบีบอัดของข้อมูล (Latent space) โดยใน generator model จะเลือก latent space ในการหาจุดใหม่เพื่อเตรียมให้ input สามารถสร้าง ตัวอย่างที่แตกต่างได้



ภาพที่ 2.32 Generative model

The generator จะพยายาม ทำให้ตัวอย่างที่ สร้างขึ้นมาใกล้เคียงกับของจริง โดยจะวิธีที่ใช้ในการ extract feature มี 2 วิธี คือ 1) Up sample layer และ 2) Transpose Convolutional layer

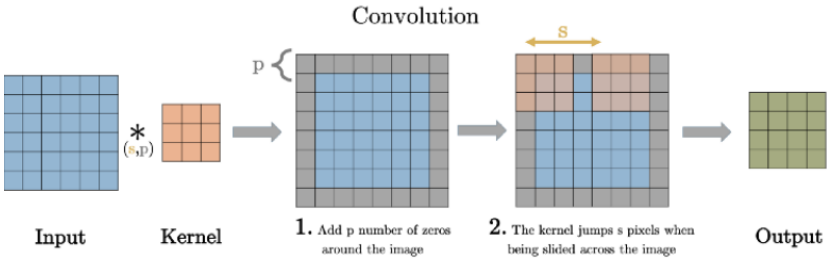
Up sample layer คือ เพิ่ม layer โดยทำซ้ำๆ ทั้ง คอลัมน์ และ แถว โดยจำเป็นต้องทำเนื่องจาก ต้องการให้ layer มี dense ที่มากขึ้น และ ต้องมีรายละเอียดของ output ตัวอย่างเช่น การ up sample จาก 2 × 2 เป็น 4 × 4.



ภาพที่ 2.33 Up sample layer

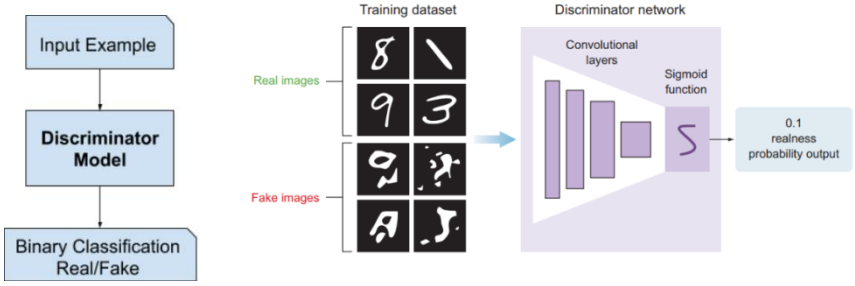
DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

Transpose Convolutional Layer คือการนำ Up sample operation และ การประเมินรายละเอียดในขณะที่ Up sampling รูปแบบของ transpose Convolutional Layer เปรียบเหมือน normal convolutional layer โดยสิ่งที่ต้องกำหนดคือ จำนวน filters และ kernel size ในแต่ละ filter โดนเรียกกระบวนการนี้ว่า stride คือการที่ kernel จะเปลี่ยน เมื่อเกิดการเลื่อนระหว่าง Input image



ภาพที่ 2.34 Transpose Convolutional Layer

Discriminator model มีหน้าที่ในการจำแนกข้อมูลจริง และ ข้อมูลที่ถูกสร้างขึ้นมา โดยข้อมูลจริงจะอยู่ใน training dataset ส่วน ผลลัพธ์จากข้อมูลที่ถูกสร้าง จะมาจาก generator model หลังจากที่ทำการ Training process discriminator จะถูกกำจัดเนื่องจากสิ่งที่ต้องสนใจคือ generator สามารถ หลอกdiscriminator ได้แค่ไหน โดยจุดประสงค์ในการ training GANs คือ การ นำ generator และ discriminator กลับมาใช้ใหม่ ในการทำ feature extraction



ภาพที่ 2.35 Discriminator model

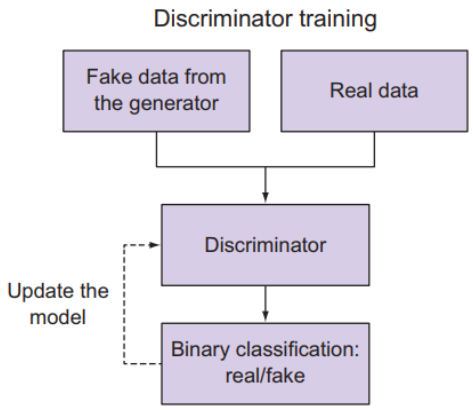
DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

จากตัวอย่าง จะใช้ sigmoid activation function ในการจัดกลุ่ม แบบ binary โดยผลลัพธ์ที่ได้ออกมาจะมีค่าอยู่ระหว่าง 0 และ 1

2.4.3 Training GANs

GANs จะมีการ training 2 ขั้นตอน

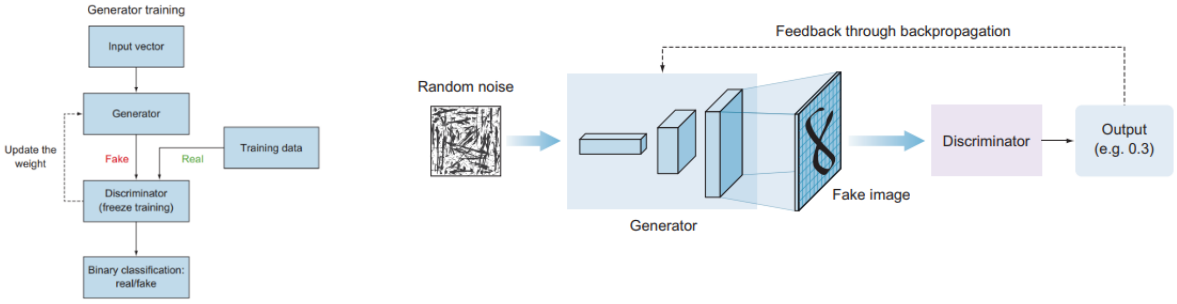
1. Train discriminator ใน network จะมี label ที่มาจาก generator(fake) และ training data (real) โดยจะเรียนรู้และจัดกลุ่มระหว่าง fake และ real ผ่าน sigmoid prediction ขั้นตอนการ train discriminator ต้อง maximizing log ค่าความน่าจะเป็น ของ real และ log invert ของ fake โดยจะเฉลี่ยออกมาใน minibatch โดยจะได้ minibatch 2 กลุ่ม กลุ่มแรกมาจาก label ที่เป็น 1 จากตัวอย่างทั้งหมด และ กลุ่มสอง label ที่เป็น 0 จาก generator โดยถ้า loss function หาค่าความน่าจะเป็น ใกล้ 1 แสดงว่า เป็น real ถ้าความน่าจะเป็น ใกล้ 0 แสดงว่าเป็น fake



ภาพที่ 2.36 Discriminator training

2. Train generator จะซับซ้อนกว่า discriminator โดย generator จะไม่สามารถ training ตัวเดียวได้ จำเป็นที่ต้องให้ discriminator เป็นตัวบอกว่า fake ที่ดีควรทำยังไง โดยจะต้องสร้าง network ร่วมกัน ระหว่าง generator กับ discriminator เมื่อ ต้องการที่จะ train generator ต้อง freeze ค่าน้ำหนัก ของ discriminator เพราะค่า loss function จะมีความต่างกัน

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์



ภาพที่ 2.37 Generator training

2.4.4 Minimax function

ในการ Training GANs จะมีปัญหา zero-sum game โดยถ้าหากจะเพิ่มคะแนนตัวใดตัวหนึ่งอีกตัวคะแนนก็จะลดลงไปดังนั้นจึงจำเป็น ที่จะต้องใช้ Minimax function จุดประสงค์เพื่อ จุดสมดุลระหว่างตัวแปรทั้งคู่โดยแบ่งเป็น maximizer หา ค่า maximum และ minimizer หาค่า minimum โดยใน GANs จะมีสมการดังนี้

$$\min_G \max_D V(D,G) = E_{x\sim p_{data}}[\log D(x)] + E_{z\sim P_{z(z)}}[\log(1 - D(G(z)))]$$

Discriminator output for real data x — **Discriminator output for generated fake data G(z)**

ภาพที่ 2.38 Minimax function

เป้าหมายของ Discriminator คือการหาค่าความน่าจะเป็นที่สูงที่สุด สำหรับการเลือก label ให้ถูก ส่วน Generator คือการหาค่าความน่าจะเป็นที่ต่ำที่สุด

2.3.5 Evaluating GAN model

การวัดประสิทธิภาพของ GANs ไม่ได้วัดแค่ model ตัวเดียว แต่ต้องวัดไปพร้อมกัน คล้ายกับ train generator โดยวิธีที่ เหมาะสม สำหรับการวัดข้อมูล ประเภทคุณภาพ(Image quality) และ ปริมาณ(diversity) ใน GANs มี 2 วิธี 1.) Inception Score 2.) Fréchet inception distance (FID)

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

2.5 งานวิจัยที่เกี่ยวข้อง

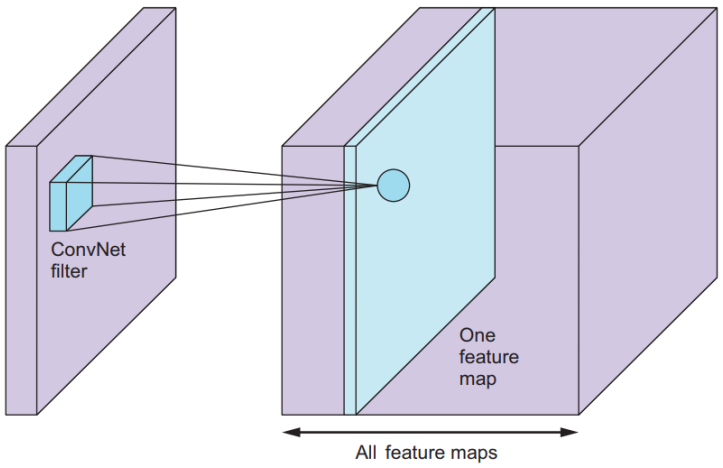
2.5.1 Deep Convolutional GAN

DCGAN concept มาจาก ConvNets โดยจะมีองค์ประกอบ คือ Convolutional filters, Parameter sharing, ConvNets visualized

Convolutional filters จะไม่เหมือนกับ feed-forward neural network ที่จะถูกจัดแบบ fully connect layers โดยใน ConvNets layer จะถูกแบ่งเป็น 3มิติ (กว้าง x ยาว x สูง) โดยใน convolution จะทำการเลื่อน filter บน input layer ทุกๆช่องที่เลื่อนไปใน input ของ แต่ละ filter ผลลัพธ์ที่ได้ออกมาจะเป็น single activation value โดยกระบวนการทั้งหมดผลลัพธ์ ที่ได้จะเป็นการ map 2 dimension ระหว่าง input และ filter

Parameter sharing คือการให้ Parameter ให้ input ทั้งหมด ไปยังfilter ที่กำหนด parameter จะเป็นตัวที่กำหนด การเรียนรู้ feature และ รูปร่าง โดย parameter sharing จะมีประโยชน์ คือลดการ overfitting และ สามารถนำเทคนิค scale up high resolution มาใช้ได้

ConvNets visualized input ที่เข้ามาจะเป็น 3มิติ และเวลา filter เป็น ลำดับขั้นการกำหนดจำนวน filter ขึ้นอยู่กับความลึกของผลลัพธ์ที่ต้องการ



ภาพที่ 2.39 ConvNets

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

DCGAN เป็นพื้นฐานในการใช้เทคนิคเพื่อให้ การ training มีความสเถียรมากขึ้นโดยใช้ Batch normalization และใน DCGAN จะไม่ใช้ fully connected layer แบบเดิม แต่จะใช้ Transposed convolution technique: Up-sampling 2d

Batch normalization ถูกเปิดตัวครั้งแรกโดย Sergey Ioffe และ Christian Szegedy ในปี 2015 โดยจุดประสงค์ ของการ normalize เพื่อปรับ input เวลาผ่านในแต่ละ layer ขณะ training minibatch ในการผ่าน Network การ normalize เพื่อ ปรับค่าเฉลี่ยเป็น 0 และ ปรับค่าความแปรปรวน ข้อดีของการ normalize มีหลายอย่าง เช่น ต้องเปรียบเทียบระหว่าง feature ซึ่งการ normalize จะทำให้ การ train มีประสิทธิภาพที่ดี โดย batch normalization คือการ ทำ normalize input เดี่ยวๆ เมื่อ input ไหลเข้ามาใน network จาก layer หนึ่ง ไปอีก layer หนึ่ง จะมีการขยาย parameter ในแต่ละ layer โดย parameter จะถูก ปรับค่าใน กระบวนการ backpropagation ทำให้การกระจายตัวของข้อมูลมีการเปลี่ยนแปลงตลอดเวลา และ เรียกปัญหานี้ว่า covariate shift ซึ่ง Batch normalization สามารถเข้ามาช่วยแก้ปัญหา โดยการเพิ่มขนาด ของ value ในแต่ละ mini-batch โดยมีสมการ ดังนี้

$$\hat{x} = \frac{x - \mu_B}{\sqrt{\sigma^2 + \varepsilon}}$$

ภาพที่ 2.40 สมการ Batch normalization

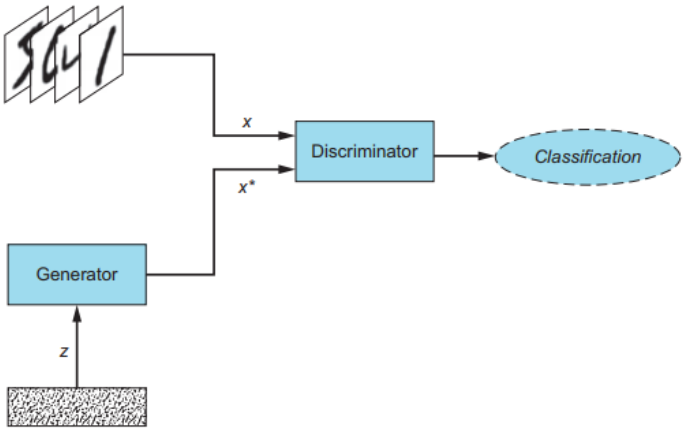
μ_B คือค่าเฉลี่ย ของ mini-batch

σ_{B^2} คือ ค่าความแปรปรวน ของ mini-batch

ε (epsilon) คือ เลขที่เพิ่มเพื่อหลีกเลี่ยงการหารด้วย 0

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

กระบวนการใน DCGAN



ภาพที่ 2.41 กระบวนการใน DCGAN

กระบวนการใน DCGAN จะมีความแตกต่างกับ GANs คือ ในgenerator ของ GANs จะใช้ fully connected network ส่วน generator ใน DCGAN จะใช้ เทคนิค transposed convolution ในการทำ up-sampling ของ 2D image size

2.5.2 Style GAN

กระบวนการ GAN ชนิดนี้ เป็น กระบวนการส่วนที่ขยาย มาจาก progressive growing GAN วิธีการคือ generator model สังเคราะห์ ภาพขนาดใหญ่ คุณภาพสูง โดยเริ่มต้นจากทั้ง model discriminator และ generator โดยขนาดภาพนั้น จะเริ่มจากเล็กไปใหญ่ ในกระบวนการ training

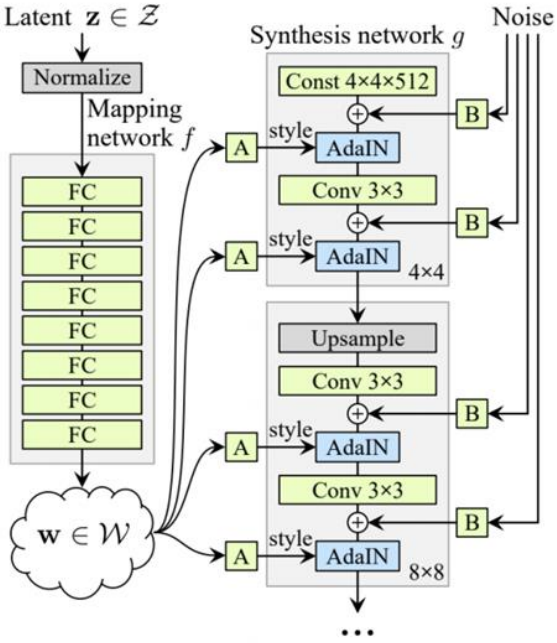
ในส่วน ของ generator จะไม่ได้ใช้จุดใดจุดหนึ่งใน latent space แต่จะใช้การสุ่มเพื่อสร้างภาพ แบบ mapping network แบบเดียว และ noise layers ผลลัพธ์ที่เกิดจากการทำ mapping network คือ vector ที่มีการกำหนดรูปแบบที่รวมไว้ในแต่ละจุดของ generator โดย layer แบบใหม่ เรียกว่า adaptive instance normalization

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

โครงสร้างใน Style Gan

สิ่งที่เปลี่ยนแปลงใน Generator ของ Style Gan

- พื้นฐาน มาจาก Progressive GAN
- เพิ่มการปรับ และ bilinear up sampling
- ปรับกระบวนการ mapping network และ AdaIN
- ไม่ใช้ latent vector input เพื่อ generator
- เพิ่ม noise ในแต่ละ process
- ใช้การ Mixing regularization
-



ภาพที่ 2.42 กระบวนการใน **Style Gan**

Bilinear Sampling

ในกรณีที่เป็น progressive growing GAN จะใช้ neighbor layers ในการทำ up sampling แทนที่จะใช้ transpose convolutional layers แบบ Generator ดั้งเดิม จุดเปลี่ยนใน style gan คือ จะใช้ layer แบบสุ่ม(bilinear up sampling layers) แทนการใช้ neighbor layers

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

Mapping Network and AdaIN

การ standalone mapping network เป็น การใช้จุดสุ่มตัวอย่างจาก latent space as input และ สร้างในรูปแบบ vector ดังนั้น Mapping Network คือ ประกอบ fully connected layers 8 ชั้น หลังจากนั้น vector จะ ถูกแปลง และ รวมเข้ากับแต่ละ blog ของ model โดยทำหลังจากกระบวนการที่เรียกว่า adaptive instance normalization(AdaIN)

AdaIN layers คือ การสร้างมาตรฐาน output ของ feature map เพื่อทำ standard Gaussian และเพิ่ม vector ในรูปแบบ ของ bias

$$\mathrm{AdaIN}(x_i, y) = y_{s,i}\frac{x_i - \mu(x_i)}{\sigma(x_i)} + y_{b,i}$$

ภาพที่ 2.43 กระบวนการใน Adaptive instance normalization

การเพิ่ม รูปแบบใหม่ของ mapping network ใน โครงสร้าง ส่งผลต่อผลลัพธ์ใน Generator model ในการ synthesis network

Remove latent point ตัว model ได้ปรับ Generator model เพื่อไม่ให้ใช้จุดจาก latent space ในขณะเดียวกัน ให้ input มีค่าคงที่ $4 \times 4 \times 512$ ในกระบวนการสร้างภาพ

Addition of noise ผลลัพธ์ในแต่ละ layer ที่สร้างภาพ คือ block ของ activation map โดย gaussian noise จะถูกเพิ่ม เข้าไปในแต่ละ activation maps ก่อนที่จะเริ่มกระบวนการ AdaIN ในการเพิ่มปริมาณของ noise จะแตกต่างขึ้นอยู่กับ ขนาดของ layer

Mixing regularization รูปแบบการสร้าง vector 2 รูปแบบ จากการ mapping network โดยจะทำการแบ่ง จุดในการสร้างภาพ และกระบวนการ AdaIN ทั้งหมด โดยจะแยกจุดแรกของ vector และ AdaIN ทั้งหมดหลังจาก แยก vector รูปแบบที่สอง กระบวนการนี้จะทำให้ layer และ block ที่กำหนดสามารถกำหนดรูปแบบโดยเฉพาะของ model และ ทำให้ภาพที่สร้างขึ้นมีรายละเอียดที่สอดคล้อง

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

2.5.3 Pix2Pix

Image-to-image translation คือการควบคุมของรูปภาพต้นฉบับที่กำหนดเป็นรูปภาพเป้าหมาย เช่นการเปลี่ยนภาพขาวดำเป็นภาพสี การทำ image translation เป็นเรื่องที่ยากเนื่องจากต้องกำหนด model โดยเฉพาะ และ ค่า loss function ที่จะทำให้ผลลัพธ์ออกมาสมบูรณ์

Pix2Pix GAN เป็น generator model มีหน้าที่ในการจับคู่รูปภาพระหว่างภาพที่เป็นเป้าหมายกับภาพจริง เช่นการใช้ Pix2Pix GAN ในการสร้างภาพจากภาพตอนกลางวันให้เป็นตอนกลางคืน หรือ การสร้างภาพ art จาก ลายเส้นไปจับกับภาพสี ทำให้ได้ภาพ ที่มีสีออกมา

Pix2Pix GAN เป็นประเภทของ conditional gan โดย discriminator จะเตรียม รูปภาพจริง และ รูปภาพเป้าหมาย โดยเป้าหมายใน model คือการพิจารณาความเป็นไปได้ในการเปลี่ยนผลลัพธ์ ส่วน generator มีหน้าที่ในการสร้างภาพเพื่อให้สามารถทำให้ใกล้เคียง กับ target domain ได้

ภาพที่ 2.44 ตัวอย่างผลลัพธ์ที่ได้จาก Pix2Pix GAN

2.5.4 CycleGAN

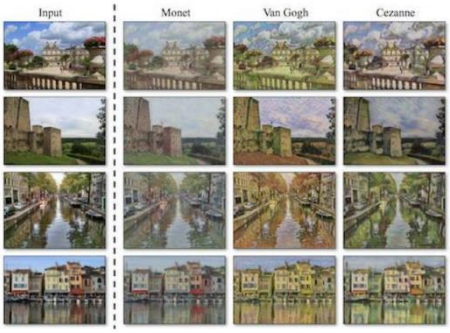
กระบวนการ GAN ชนิดนี้ เป็นเทคนิค การ training ของ image translation โดยไม่ต้องมีภาพตัวอย่าง CycleGAN จะถูก train ในลักษณะ unsupervised โดยจะใช้ข้อมูลตั้งต้น และ target domain ที่ไม่ได้มีความเกี่ยวข้องกัน ตัวอย่างที่นำเทคนิคนี้ไปใช้คือ การแปลงภาพม้าเป็นม้าลายและม้าลายเป็นม้า

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

CycleGAN จะมีส่วนขยายเพิ่ม จาก GANs ปกติคือ cycle consistency แนวคิดคือการนำเอาผลลัพธ์อันแรกที่ได้จากการ Train generator ครั้งที่หนึ่ง มาใช้ใน generator ครั้งที่สอง และเอาผลลัพธ์ครั้งที่สอง มาดูความสัมพันธ์กับภาพจริง

งานที่นำ CycleGAN มาใช้มี 5ประเภท

1. Style Transfer คือการเรียนรู้จาก Target domain เช่นการนำภาพถ่ายมาแปลงเป็นภาพศิลปะ



ภาพที่ 2.45 ตัวอย่างผลลัพธ์ Style Transfer

2. Object Transfiguration คือการ เปลี่ยนแปลงจากวัตถุรูปแบบแรก ไปเป็นรูปแบบสอง เช่น การเปลี่ยนม้า เป็นม้าลาย

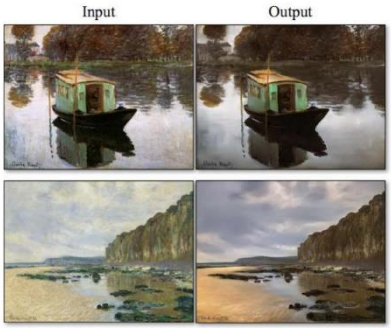
ภาพที่ 2.46 ตัวอย่างผลลัพธ์ Object Transfiguration

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

3. Season Transfer คือการแปลงภาพถ่ายที่ถ่ายในฤดูหนึ่งเป็นอีกฤดูหนึ่ง

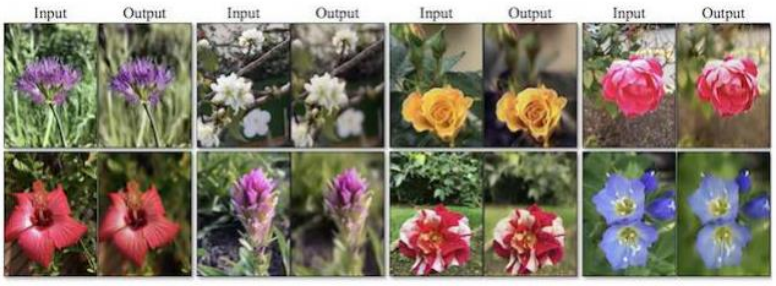
ภาพที่ 2.47 ตัวอย่างผลลัพธ์ Season Transfer

4. Photograph Generation from Paintings คือ การสร้างภาพถ่ายจากภาพวาด โดยสังเคราะห์ภาพที่เหมือนจริงขึ้นมา



ภาพที่ 2.48 ตัวอย่างผลลัพธ์ Photograph Generation

5. Photograph Enhancement คือการนำภาพต้นฉบับมาเพิ่มคุณภาพ โดยเพิ่มความลึกของภาพ และ สีให้มีความคมชัด



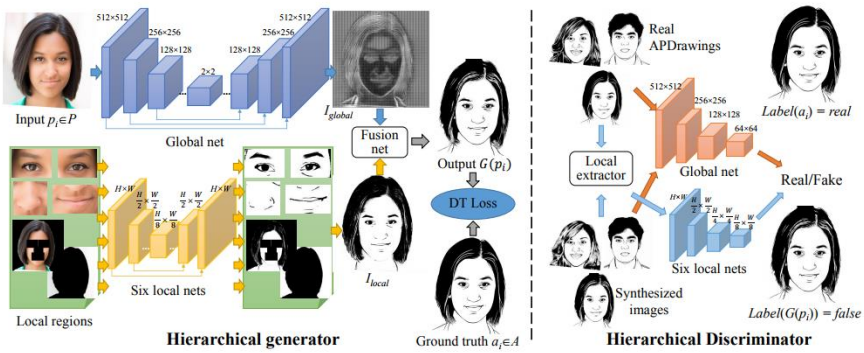
ภาพที่ 2.49 ตัวอย่างผลลัพธ์ Photograph Enhancement

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

2.5.5 APDrawingGAN

กระบวนการ GANs ชนิดนี้ เป็นโครงสร้าง GANs รูปแบบใหม่ โดยถูกจัดในรูปแบบ hierarchical generators และ discriminator รวมเข้าไปใน global network จุดประสงค์ของ Model นี้ คือ การสร้างสถาปัตยกรรม GANs แบบลำดับชั้นสำหรับการสังเคราะห์การวาดภาพบุคคลทางศิลปะจากภาพถ่ายใบหน้า โดยจะสามารถสร้างผลงานศิลปะคุณภาพสูง โดยวิธีที่ใช้สามารถ เรียนรู้ทรงผมที่ซับซ้อนได้

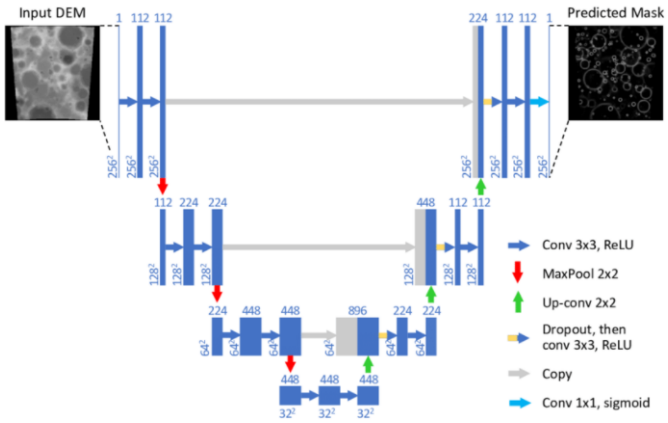
สิ่งสำคัญใน Model นี้คือการแยกผลลัพธ์ที่แสดงผลของ GANs ออกเป็นหลายๆชั้น โดยแต่ละชั้นจะถูกคุมจาก loss function ที่ต่างกัน



ภาพที่ 2.50 APDrawingGAN overview

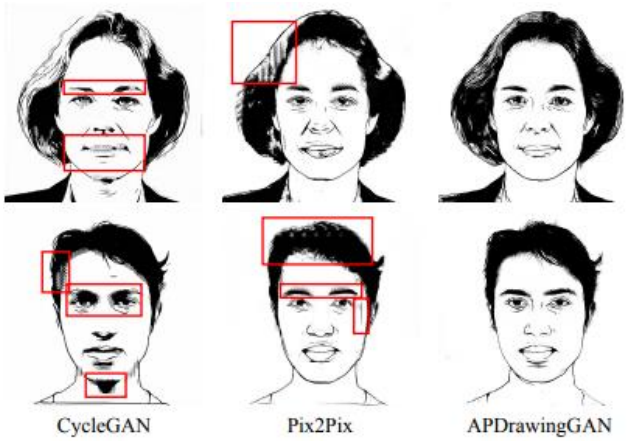
Hierarchical generator G คือ generator จะทำหน้าที่ในการแปลงข้อมูลภาพให้เข้าอยู่ในรูปแบบ APDrawingGAN ซึ่งอยู่ภายในรูปแบบ U-Net structure

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์



ภาพที่ 2.51 U-Net structure

Hierarchical discriminator D คือ discriminator ที่มีหน้าที่ แยกแยะภาพวาดของศิลปินว่าเป็นของจริงหรือไม่โดยใช้ต้นแบบ discriminator จาก Pix2Pix โดยความแตกต่างจะอยู่ตรง Input ที่มีการกำหนด ขนาดของภาพที่แตกต่างกันในแต่ละลำดับชั้น



ภาพที่ 2.52 ผลลัพธ์ของ APDrawingGAN ที่มีผลลัพธ์ที่ดีกว่า model CycleGAN และ Pix2P

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

บทที่ 3

วิธีดำเนินการวิจัย

3.1 เครื่องมือที่ใช้ในงานวิจัย

3.1.1 โปรแกรมคอมพิวเตอร์

TensorFlow เป็นไลบรารีที่ใช้ในการพัฒนา Machine Learning พัฒนาโดย Google

ภาพที่ 3.1 TensorFlow

PyTorch เป็น Deep learning library ของภาษา Python พัฒนาโดย Facebook

ภาพที่ 3.2 PyTorch

NumPy เป็นโมดูลเสริมใน Python มีฟังก์ชันเกี่ยวกับคณิตศาสตร์ และ การคำนวณ

ภาพที่ 3.3 NumPy

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

Colab เป็นบริการ Software as a Service (Saas) โฮสต์โปรแกรม Jupyter Notebook บน Cloud จาก Google ใช้ เขียนโปรแกรมภาษา Python และ มี GPU และ TPU มาให้

ภาพที่ 3.4 Colab

3.2 ข้อมูลที่ใช้ในงานวิจัย

ข้อมูลที่ใช้ในงานวิจัยเป็นข้อมูลภาพถ่ายรถ GTR R33,R32,R34,R35 ซึ่งใช้ภาพทั้งหมด 300 ภาพ มีขนาด 512 x 512 pixel โดยภาพที่ใช้ในงานวิจัยนี้จะต้องเป็นภาพที่มีการclean noise และ มิติที่ถูกต้องแล้วเท่านั้น โดยกระบวนการเตรียม ข้อมูลภาพมีขั้นตอนทั้งหมด 3 ขั้นตอน ดังนี้

3.2.1 ดาวน์โหลดภาพที่ต้องการ

ในงานวิจัยนี้ใช้ภาพถ่ายรถ GTR เฉพาะด้านข้างรถ จาก jdm-expo และ Nissan Japan official เป็นภาพที่มีรายละเอียดสูงมุมภาพถูกต้อง แต่ก็มีข้อเสียคือด้านข้างรถนั้นไม่ค่อยมีความหลากหลายและมีปริมาณที่น้อย ดังนั้นจึงมีการใช้ Filter ให้ภาพ และ เปลี่ยน object บางอย่าง เพื่อให้ภาพมีความหลากหลาย เพื่อให้มีความหลากหลายมากขึ้น

ภาพแรก เป็นภาพ GTR R34 กับล้อ NISMO

ภาพสอง เป็นภาพ GTR R34 กับล้อ Enkei

ภาพที่ 3.5

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

3.2.2 การลบภาพพื้นหลังของภาพ

เมื่อทำการดาวน์โหลดภาพมาเรียบร้อยภาพจะมี Background ที่ไม่เกี่ยวข้องมาด้วยทำให้ modelเรียนรู้ได้ยาก สิ่งที่ต้องทำก่อนปรับขนาดรูปภาพคือใช้ tool ของ remove.bg โดย tool จะทำหน้าที่เอาแต่ object ที่เราต้องการ

ภาพ 3.6 ภาพตัวอย่างการลบพื้นหลัง

ภาพที่ 3.6

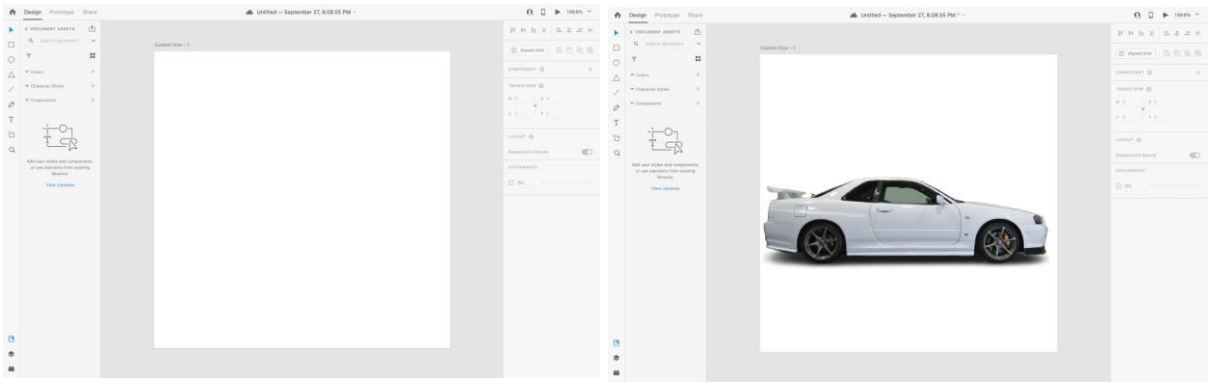
3.2.3 การปรับแก้ขนาดของภาพ

หลังจากตัดภาพพื้นหลังของภาพออกขั้นตอนต่อไปคือการ ปรับแก้ขนาดภาพให้มีขนาดเท่ากับ 512 x 512 pixel เนื่องจากเป็นข้อมูลที่นำไปใช้ในการเทรนโมเดล (Training Data) ซึ่งบทที่ 2 ได้เอกสารและงานวิจัยที่เกี่ยวข้องทำให้ทราบว่า การเทรน โมเดล GANS จะมีการจำลองพื้นที่ภาพให้เป็นพื้นที่ย่อยๆ และนำกลุ่มของพื้นที่ย่อย ๆมาผสานกัน เพื่อดูว่าสิ่งที่เห็น อยู่เป็นอะไรแล้วสร้างขึ้นมาใหม่ ดังนั้นจึงต้องมีการกำหนดขนาดของพื้นที่ย่อยๆให้เท่ากันโดย ขนาดของภาพทั้งตอน train data และ output ออกมาขนาดจะเท่ากัน เทคนิคที่งานวิจัยนี้เลือกใช้คือการจำลองภาพพื้นหลังให้เหมือนกันทั้งหมดโดยใช้พื้นหลังสีขาว ขั้นตอนมีดังนี้

1) เปิดโปรแกรม AdobeXd แล้วสร้าง canvas size 512x512px

2) นำภาพที่ตัดback ground แล้วในขั้นตอนตัดback ground มาวางกึ่งกลางของ canvas แล้วปรับให้เหมาะสม

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์



ภาพที่ 3.7 ภาพโปรแกรมที่เอาไว้ปรับแก้ขนาดของภาพ

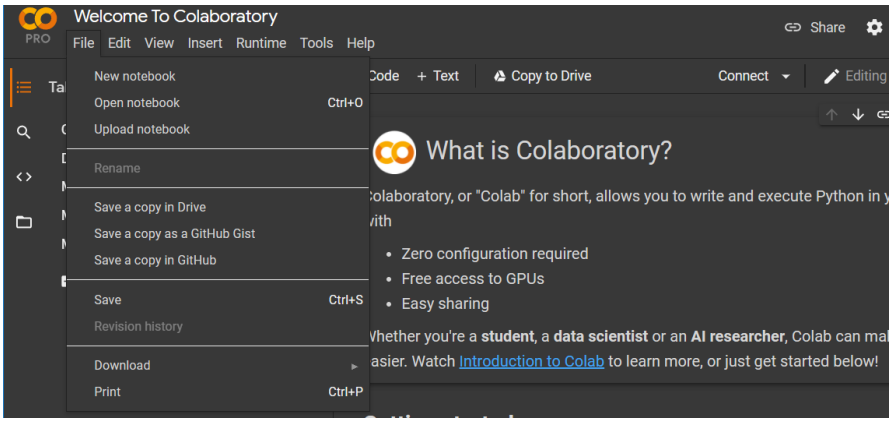
3.3 การติดตั้ง Packages และ Environments สำหรับเขียนโค้ดใน Python

ในงานวิจัย GAN ฉบับนี้ได้เลือกใช้ Environments จาก google colab เพื่อให้ผู้ที่สนใจสามารถนำไปประยุกต์หรือศึกษาต่อได้โดย google colab ที่ใช้จะเป็น google colab ชนิด pro ซึ่งมีค่าใช้จ่ายในการสมัครซึ่งได้รับความนิยมเป็นอย่างมากในหมู่ นักพัฒนาโปรแกรม รวมถึงผู้สนใจด้าน Deep learning เพราะติดตั้งได้ง่าย รวมถึง การติดตั้ง package ต่าง ๆ ที่สำคัญ โดยเฉพาะ TensorFlow , ml4a(Machine Learning for Artists) , และอื่น ๆ อีกมากมาย

นอกจากนี้ google colab ยังมีการ์ดจอมาให้ใช้แบบไม่คิดค่าบริการสำหรับงานtrain ที่ไม่ได้มีความละเอียดสูงและ Dashboard TensorFlow ช่วยบริหารจัดการ ram และ การ์ดจออัตโนมัติ

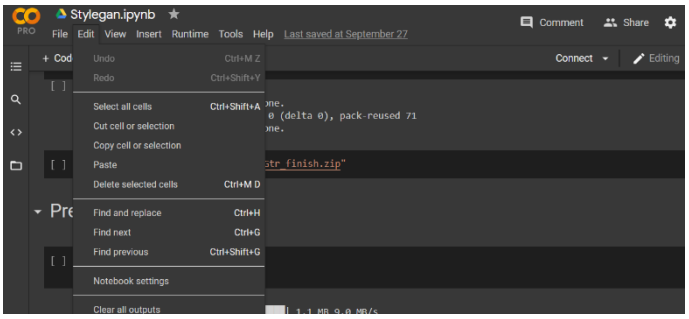
1. เปิด Colab เข้าไปที่เว็บไซต์ https://colab.research.google.com File > New notebook

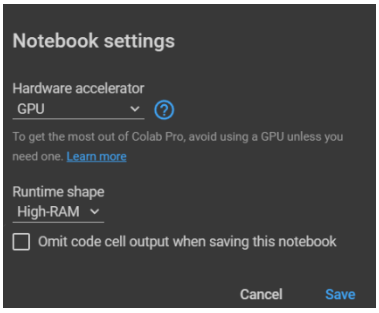
DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์



ภาพที่ 3.8 ภาพโปรแกรมGoogle colab

2. ให้กดที่ Edit เปลี่ยน Notebook setting เป็น GPU และ High-RAM



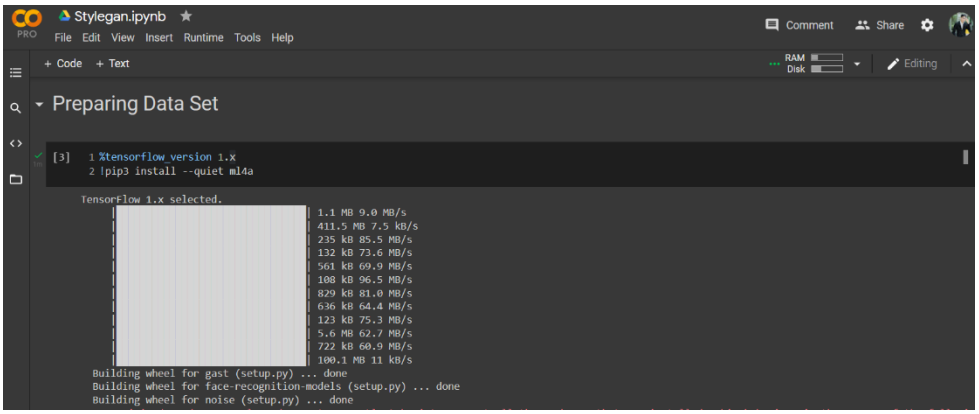


ภาพที่ 3.9 ภาพโปรแกรมGoogle colab set GPU

การติดตั้ง Python Library ที่จำเป็นต้องใช้งานในโปรเจค ไพธอนมีไลบรารี่สำหรับการใช้งานในวัตถุประสงค์การสร้างงานที่หลากหลาย ดังนั้นเวลาที่จะพัฒนางานในลักษณะใดๆก็จำเป็นจะต้องติดตั้ง โมดูล ของงานนั้น หรือ บางครั้งการทดลองติดตั้งโมดูล ต่างๆอาจจะผิดเวอร์ชันก็จำเป็นจะต้องลบโมดูลนั้นทิ้ง โดยจะใช้คำสั่งหลักคือ pip (python install package) ที่จะทำหน้าที่ดาวน์โหลดและติดตั้งเข้า

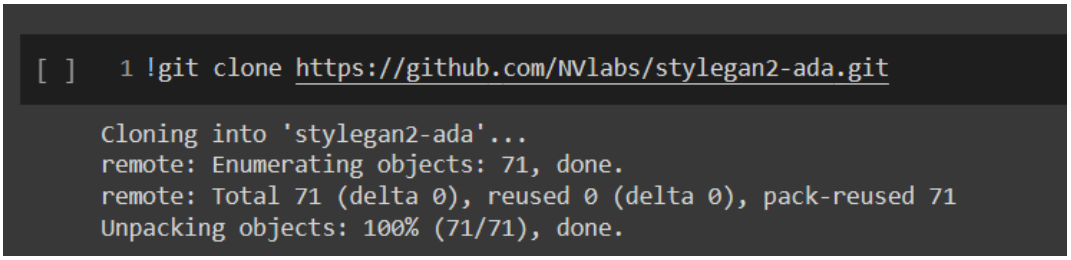
DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

2.1 ติดตั้ง TensorFlow version แรกเท่านั้น ห้ามติดตั้ง version สอง เพราะ style gan ไม่รองรับ พร้อมกับติดตั้ง ml4a โดยจะทำหน้าที่เชื่อมระหว่าง texture กับ style ของ ภาพที่นำเข้ามาใช้



ภาพที่ 3.10 ภาพโปรแกรมGoogle colab install TensorFlow

2.2 ติดตั้งชุด Code จาก stylegan2-ada โดยใช้คำสั่ง git clone



ภาพที่ 3.11 ภาพโปรแกรมGoogle colab install stylegan

3. ขั้นตอนการ Preprocess ก่อนเข้า model training

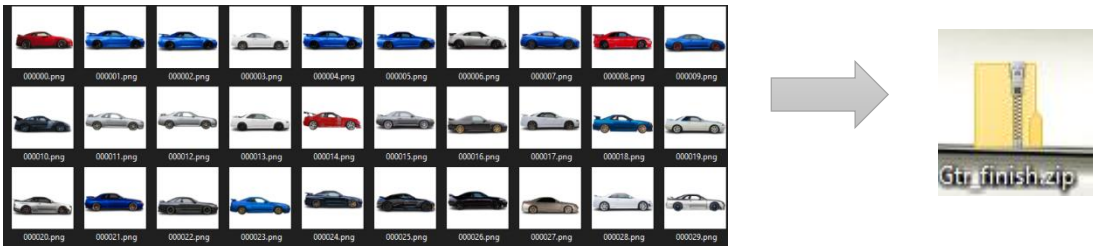
3.1) ให้นำภาพทั้งหมดที่ทำเป็นขนาด 512x512 PX มาเปลี่ยนชนิดของfile โดยขั้นตอนนี้ผู้วิจัยพบว่าในกรณีที่ถ้าไม่เปลี่ยนชนิดของไฟล์ให้เหมาะสม เช่น ชื่อไฟล์ไม่ได้เรียงตามเลข หรือชนิดไฟล์เป็นไฟล์ PNG จะไม่สามารถแปลงไฟล์ได้

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์


```
import os
import glob

os.chdir(r"c://Users//chalee//Desktop//art_gtr//gtr//images")
for index, oldfile in enumerate(glob.glob("*.png"), start = 0 ):
    newfile = '00000{}.png'.format(index)
    os.rename (oldfile,newfile)
```

ภาพที่ 3.12 ภาพชุด Codeที่สร้างมาเพื่อเปลี่ยนสกุลไฟล์

3.2) นำภาพทั้งหมดที่เปลี่ยนชื่อ และ ชนิดไฟล์ ไป zip พร้อม upload ขึ้น google drive



ภาพที่ 3.13 ภาพชุดข้อมูลที่ถูก zip file

3.3) Unzip ภาพใน colab

```
[ ]   1 !unzip "/content/drive/MyDrive/Gtr_finish.zip"
```

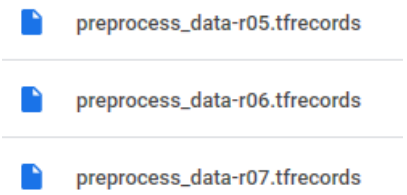
ภาพที่ 3.14 ภาพชุดcode unzip file

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

3.4) เลือก path ที่ dataset_tool.py และ ไฟล์ที่จะนำมาแปลงข้อมูลให้เหมาะสม ถัดไปจะ set path ที่จะ save data สำหรับการ train model โดยไฟล์ที่จะได้ออกมาชนิดไฟล์คือ tfrecords หรือ large numpy arrays โดยทำหน้าที่เก็บหลายๆ sample

```
[ ]  1 ! python /content/stylegan2-ada/dataset_tool.py create_from_images  /content/drive/MyDrive/preprocess_data     /content/Gtr_finish
```

ภาพที่ 3.15 รูปแบบการเก็บข้อมูลแบบ TFRecord



ภาพที่ 3.16 ภาพชุดข้อมูลที่ถูกเก็บแบบ TFRecord

3.4 ขั้นตอนการวิจัยในกระบวนการเรียนรู้ GANS

ในขั้นตอนการวิจัยในส่วนนี้ เป็นการทำงานในกระบวนของ GANS ซึ่งการทำงานของ GANS จะมีขั้นตอนหลักๆอยู่ 2 ขั้นตอน คือ การทำงานในส่วน ของโมเดล และ การนำโมเดลที่ได้ไปใช้งานโดยการทำงานในส่วนโมเดลนั้นจะมีขั้นตอนการทำงาน หลักๆอยู่ 5 ขั้นตอน ดังนี้

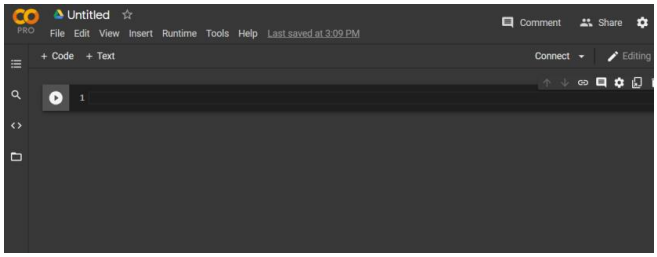
1) Load library 2) Load Data 3) Load code 4) Save target model 5) Generate

เมื่อได้โมเดลแล้วจะนำโมเดลที่ได้ไปใช้งาน โดยนำโมเดลที่ได้ไปทำการสร้างภาพขึ้นมาใหม่ซึ่งจะได้ผลลัพธ์ออกมาเป็นข้อมูลภาพ และขนาดที่ต้องการ สามารถนำผลลัพธ์ไปประยุกต์ใช้ได้

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

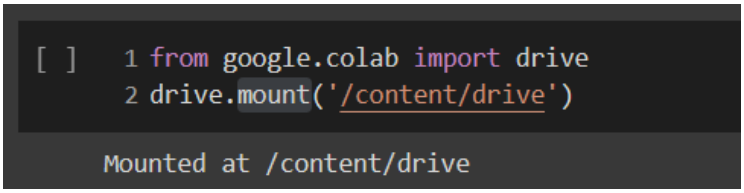
ขั้นตอนการดำเนินงานของโมเดล GANS

1) เปิดโปรแกรม Colab ขึ้นมา และสร้างไฟล์ขึ้นมาใหม่ กดที่ Edit เปลี่ยน Notebook setting เป็น GPU และ High-RAM



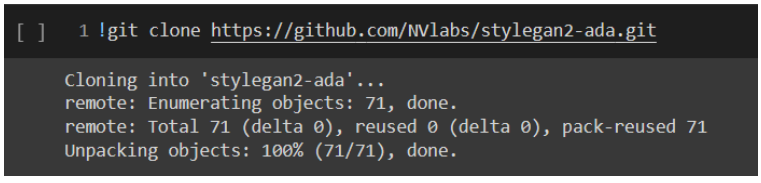
ภาพที่ 3.17 ภาพโปรแกรมGoogle colab new file

2) mount drive ของตัวเองเหมือนตัวอย่างในรูป



ภาพที่ 3.18 ภาพชุด Code mount drive

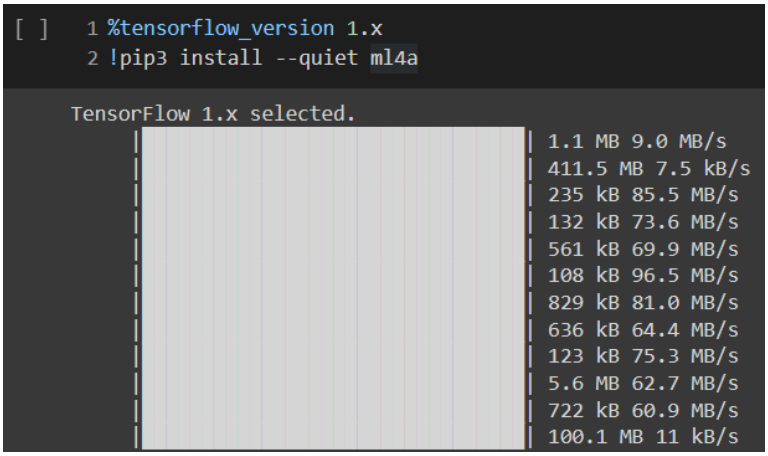
3) ใช้คำสั่ง Git clone เอา code base stylegan2-ada มาใช้



ภาพที่ 3.19 ภาพโปรแกรมGoogle colab install stylegan

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

4) ติดตั้ง TensorFlow version แรกเท่านั้น ห้ามติดตั้ง version สอง เพราะ style gan ไม่รองรับ พร้อมกับติดตั้ง ml4a โดยจำทำหน้าที่เชื่อมระหว่าง texture กับ style ของ ภาพที่นำเข้ามาใช้



ภาพที่ 3.20 ภาพชุด Code install TensorFlow and ml4a

5) ให้ copy path ที่จะTrain model (train.py) และ set path ที่จะ save data สำหรับการ train model โดยให้ตั้งค่าดังนี้

```
!python  /content/stylegan2-ada/train.py --outdir  /content/drive/MyDrive/gtr_finish_pkl/7 \
 --snap=10 --data=/content/drive/MyDrive/preprocess_data --augpipe=bgcfnc \
 --res=512  --resume=/content/drive/MyDrive/network-snapshot-000080.pkl
```

ภาพที่ 3.21 ภาพชุด Code Train model

5.1 snap ให้ set เท่ากับ 10 โดย model จะทำการ save เป็นไฟล์ PKL ทุกๆ 10 รอบ model จะ save 1 ครั้ง

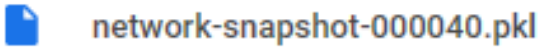
5.2 res ให้ set เท่ากับ 512 เนื่องจาก ถ้ากรณีที่ภาพมีความละเอียดที่เยอะ ทำให้ต้องใช้เวลามากในการ train ดังนั้นผู้วิจัยเห็นว่า ควรใช้ res = 512 เพื่อประหยัดเวลาและไม่ทำให้ การ์ดจอทำงานหนักจนเกินไป

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

5.3 augpipe ให้ set เท่ากับ bgcfnc โดย function นี้จะทำให้ data set ชุดนี้ ผ่าน augmentations ทุกตัว ดังนี้ blit, geom, color, filter, noise, cutout

5.4 resume กรณีที่ถ้ามี pretrain ที่ได้มาจากการ train model แล้ว ให้นำมาใส่ เพื่อ train ต่อได้เลย

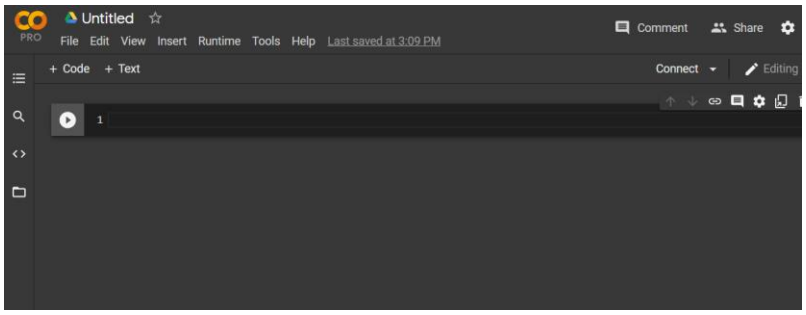
5.5 กรณีที่ Train บน colab จะสามารถ Train ได้แค่ 24 ชั่วโมง ดังนั้น ให้นำไฟล์ Pkl ไป resume ต่อ โดยระยะเวลาที่จะได้ pretrain ที่มีคุณภาพ นำไปใช้ในงานอื่นต่อได้ ขึ้นอยู่กับขนาดของภาพ และ จำนวนการ์ดจอ



ภาพที่ 3.22 ข้อมูลที่ Train เสร็จแล้ว

กรณีผู้ที่สนใจที่จะทำ stylegan2-ada base on PyTorch จะมีวิธีที่แตกต่างดังนี้

1) เปิดโปรแกรม Colab ขึ้นมา และสร้างไฟล์ขึ้นมาใหม่ กดที่ Edit เปลี่ยน Notebook setting เป็น GPU และ High-RAM



ภาพที่ 3.23 ภาพโปรแกรมGoogle colab new file

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

2) ติดตั้งชุด Code จาก stylegan2-ada-pytorch โดยใช้คำสั่ง git clone

```
[ ]  1 !git clone https://github.com/dvschultz/stylegan2-ada-pytorch

     Cloning into 'stylegan2-ada-pytorch'...
     remote: Enumerating objects: 524, done.
     remote: Total 524 (delta 0), reused 0 (delta 0), pack-reused 524
     Receiving objects: 100% (524/524), 8.40 MiB | 13.10 MiB/s, done.
     Resolving deltas: 100% (298/298), done.
```

ภาพที่ 3.24 ภาพโปรแกรมGoogle colab install stylegan pytorch

3) Check GPU

```
[ ]  1 !nvidia-smi -L

     GPU 0: Tesla T4 (UUID: GPU-50f602a7-2180-eda2-06f5-f9f5893cf77a)
```

ภาพที่ 3.25 ภาพชุด Code Check GPU

4) ให้ set ค่าตามรูปนี้ โดย dataset ไม่ต้องแปลงไฟล์ zip แล้วใช้ได้เลย ในกรณีที่มี pretrain data สามารถนำมา ใส่ path ตรง resume ได้เลย ค่า mirror_x คือการทำ augmentation ให้ภาพพลิกซ้าย ขวา ข้างล่างที่เขียนว่า option ในส่วนนี้ เป็นส่วนที่set เพื่อให้ค่าออกมาดีสุด กรณีอยากเปลี่ยน snapshot ผู้วิจัย แนะนำ 10 เพราะไม่กินทรัพยากรของการ์ดจอมากเกินไป

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์


```
#required: definitely edit these!
dataset_path = /content/drive/MyDrive/preprocess_data
resume_from = /content/drive/MyDrive/network-snapshot-000080.pkl
aug_strength = 0.0
train_count = 0
mirror_x = True
#mirror_y = False

#optional: you might not need to edit these
gamma_value = 50.0
augs = 'bg'
config = '11gb-gpu'
snapshot_count = 4
```

ภาพที่ 3.26 ภาพชุด Code เตรียมข้อมูลก่อน train

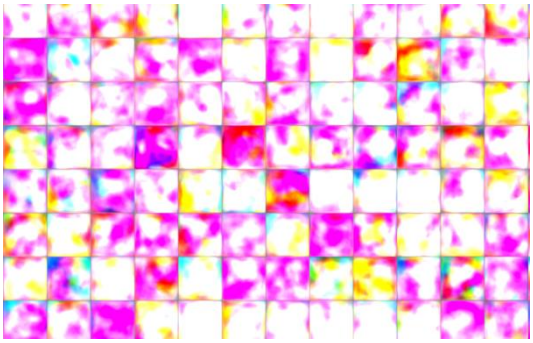
5) ให้ copy path ที่จะTrain model (train.py) แล้วกด Run

```
!python /content/stylegan2-ada-pytorch/train.py
--gpus=1 \
--cfg=$config \
 --metrics=None \
 --outdir=/content/drive/MyDrive/gtr_stylegan_ada_pytorch/3 \
 --data=/content/drive/MyDrive/Gtr_finish.zip \
 --snap=10 \
 --resume=/content/drive/MyDrive/network-snapshot-000160.pkl \
 --augpipe=bgcfnc --initstrength=$aug_strength \
 --gamma=$gamma_value --mirror=False  \
 --mirrory=False \
 --nkimg=$train_count
```

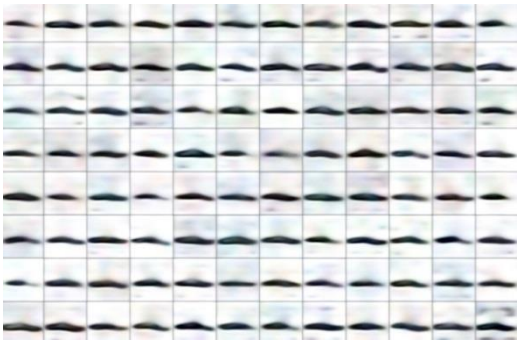
ภาพที่ 3.27 ภาพชุด Code train ข้อมูล

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

ผลลัพธ์ของโมเดล Style GAN



ภาพที่ 3.28 ผลลัพธ์ของโมเดล **GAN** ที่สร้างจากจุดเริ่มต้น



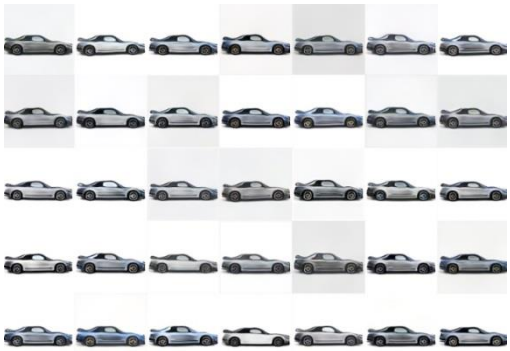
ภาพที่ 3.29 ผลลัพธ์ของโมเดล GAN ที่เทรนด้วยภาพ 40 ภาพ

ภาพที่ 3.30 ผลลัพธ์ของโมเดล GAN ที่เทรนด้วยภาพ 80 ภาพ

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

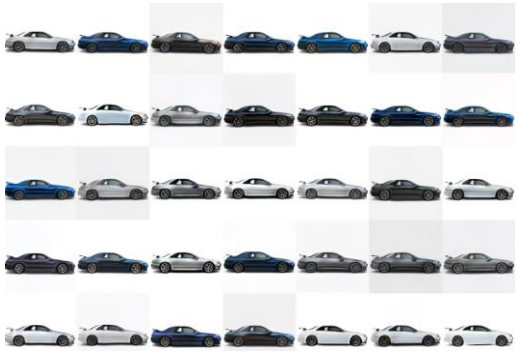
ภาพที่ 3.31 ผลลัพธ์ของโมเดล **GAN** ที่เทรนด้วยภาพ 120 ภาพ

ภาพที่ 3.32 ผลลัพธ์ของโมเดล **GAN** ที่เทรนด้วยภาพ 160 ภาพ

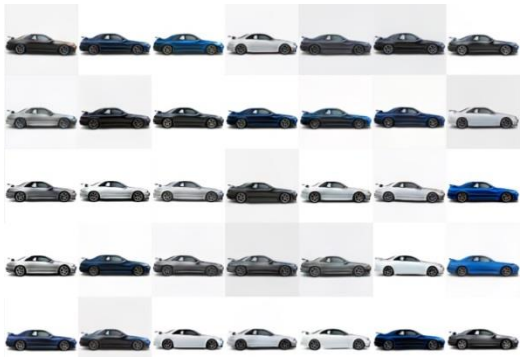


ภาพที่ 3.33 ผลลัพธ์ของโมเดล **GAN** ที่เทรนด้วยภาพ 200 ภาพ

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์



ภาพที่ 3.34 ผลลัพธ์ของโมเดล **GAN** ที่เทรนด้วยภาพ 240 ภาพ



ภาพที่ 3.35 ผลลัพธ์ของโมเดล **GAN** ที่เทรนด้วยภาพ 280 ภาพ

ภาพที่ 3.36 ผลลัพธ์ของโมเดล **GAN** ที่เทรนด้วยภาพ 320 ภาพ

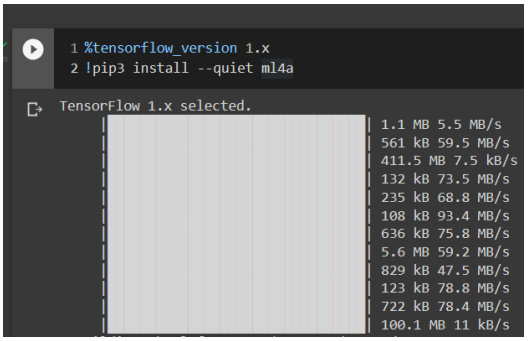
DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

ในงานวิจัยชุดนี้จะใช้เวลา Train model ประมาณ 1 อาทิตย์ เพื่อนำ pretrain ไปทำในกรณีที่ผู้สนใจอยากได้ภาพที่มีความละเอียดมากขึ้น รูปร่างรูปทรงชัดเจน เวลาขั้นต่ำที่ใช้ในการ train ให้มีประสิทธิภาพคือ 3วัน หรือ ประมาณ 600 รูป(ผลลัพธ์ที่ได้ มาจาก NVIDIA Tesla P100 PCIe 16 GB)

ภาพที่ 3.37 ผลลัพธ์ของภาพที่ใช้เวลาในการเทรน1 อาทิตย์

ขั้นตอนการconvert weight ให้เป็น pytorch เพื่อนำไป generate

1) ติดตั้ง TensorFlow version แรกเท่านั้น ห้ามติดตั้ง version สอง เพราะ style gan ไม่รองรับ พร้อมกับติดตั้ง ml4a โดยจำทำหน้าที่เชื่อมระหว่าง texture กับ style ของ ภาพที่นำเข้ามาใช้



ภาพที่ 3.38 ภาพชุด Code install TensorFlow and ml4a

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

2) ติดตั้ง Ninja ทำหน้าที่เป็นภาษา C++

```
[ ]   1 !pip install ninja
```

ภาพที่ 3.39 ภาพชุด Code install Ninja

3) copy path จาก ada pytorch เลือก export weight แล้ว set ไฟล์pretrain ที่เตรียมไว้ และ set path ที่เอาไว้ใช้ในการ export pt

```
1 !python /content/stylegan2-ada-pytorch/export_weights.py  \
2 /content/drive/MyDrive/network-snapshot-000040.pkl \
3 /content/drive/MyDrive/pt/gtr.pt
```

ภาพที่ 3.40 ภาพชุด Code export weight

4. ปรับไฟล์ generate.py ของ stylegan2 pytorch โดย ปรับ 2จุด

```
utils.save_image(
    sample,
    f"sample/{str(i).zfill(6)}.png",
    nrow=1,
    normalize=True,
    range=(-1, 1),
)
```

ภาพที่ 3.41 ภาพชุด Code การปรับ code generator

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

ลบ sample ออก

```
parser.add_argument(
    "--size", type=int, default=1024, help="output image size of the generator"
)
```

ภาพที่ 3.42 ภาพชุด Code การปรับ export data

เปลี่ยนค่า default =512 (ถ้าในกรณีข้อมูล เทรนที่ 1024 ไม่ต้องเปลี่ยน)

5. กด Run เพื่อ Generate รูปออกมา โดยกำหนดจำนวนรูปตรง pics และใช้ pretrain pt file ที่ได้มา ตรง ckpt

```
!python /content/drive/MyDrive/lee_stylegan2-pytorch-master/generate.py \
--size 512 \
--pics 20 \
--ckpt /content/drive/MyDrive/pt/gtr.pt
```

ภาพที่ 3.43 ภาพชุด Code Generate data

เปลี่ยนค่า default =512 (ถ้าในกรณีข้อมูล เทรนที่ 1024 ไม่ต้องเปลี่ยน

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

บทที่ 4
ผลการดำเนินงานวิจัย

จากการวิเคราะห์ศึกษาและประยุกต์ใช้เทคโนโลยี GANSตามกระบวนการ เทคนิค และขั้นตอนต่างๆที่ใช้โปรแกรมที่กล่าวมาในงานวิจัยบทที่ 3 วิธีดำเนินการวิจัย ได้นำเทคนิคและกระบวนการต่างๆมาปรับใช้ในงานวิจัยในหัวข้อการสร้างข้อมูลภาพจาก GANS เพื่อนำไปต่อยอดในด้าน art vector มีการนำเสนอผลดังต่อไปนี้

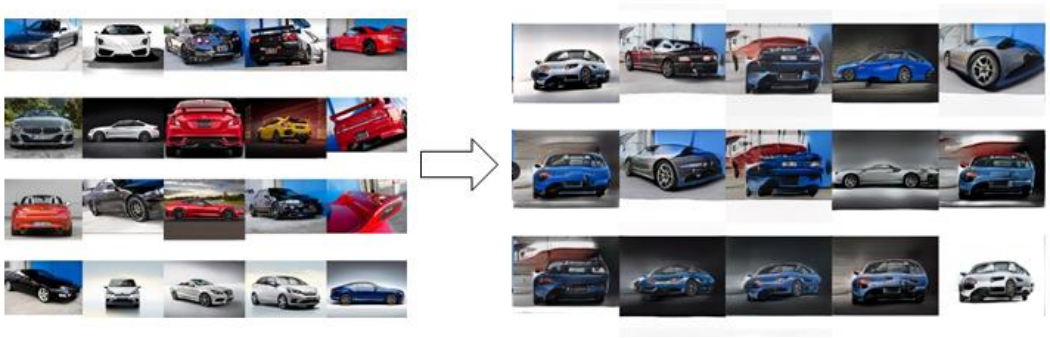
4.1 ผลการเตรียมข้อมูลภาพก่อนการเทรนโมเดล

ภาพที่ 4.1 ผลการเตรียมข้อมูล

4.2 ผลการดำเนินงานในส่วนของโมเดล GANS

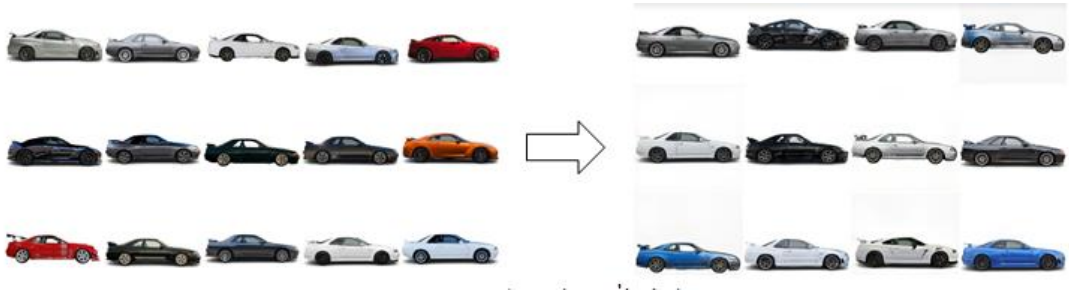
ผลลัพธ์ของโมเดล GANS จะแบ่ง เป็นข้อมูล2 ชุด 1)ภาพที่ไม่ได้ถูกการตัด Background ออก มีหลายมิติ และ ใส่ canvas เทียม 2)ภาพโครงสร้างรถคล้ายกัน ปรับภาพในทิศทางเดียวกัน และ ใส่ canvas เทียม โดยจะเห็นได้ว่าผลลัพธ์ของภาพที่ผ่านโมเดลมีความแตกต่างอย่างชัดเจน

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์



ภาพที่ 4.2 ผลลัพธ์ของข้อมูลที่ไม่ได้ถูกการตัด Background

ผลลัพธ์ของข้อมูลที่ไม่ได้ถูกการตัด Background คือ ภาพไม่สามารถจับตัวเป็นรูปทรง และ สร้างออกมาไม่ได้ แม้ใช้เทคนิคการสร้าง canvas เทียมขึ้นมา โดยภาพข้างบนได้ผ่านการ เทรน model มา 2 อาทิตย์ หรือ 2,800 รูป



ภาพที่ 4.3 ผลลัพธ์ของข้อมูลที่ไม่ได้ถูกการตัด Background

ผลลัพธ์ของข้อมูลที่ถูกการตัด Background และไปในทิศทางเดียวกัน คือ ภาพจับตัวเป็นรูปทรง และ สร้างออกมาได้ชัด เจนตามความละเอียดที่กำหนด

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

4.3 ผลลัพธ์จากการสร้างภาพจาก pretrain

ผลลัพธ์ของข้อมูลจาก pretrain แสดงให้เห็นถึงลายเส้นของตัวรถที่ชัดเจน ซึ่งตรงกับวัตถุประสงค์งานวิจัยในการเก็บรายระเอียดที่ยังคงเป็นเอกลักษณ์ของวัตถุ โดย model style gan สามารถสร้างความละเอียดที่หลากหลาย ทางผู้วิจัยเลือกใช้ความระเอียด 512x512 px เนื่องจากข้อมูล pretrain เป็นขนาด 512px และจะมีการวัดผลลัพธ์ที่ได้จากการวัดค่า Frèchet inception distance (FID)

ตารางที่ 4.1 ตารางการวัดประสิทธิภาพของ model โดยใช้ Frèchet inception distance (FID) ยิ่งค่า Frèchet inception distance (FID) มีค่าที่เข้าใกล้ 0 แสดงว่าภาพนั้นสามารถสร้างได้เทียบเท่ากับของจริง

Frèchet inception distance (FID)	
Method	GTR_DATASET
A Baseline Progressive GAN	8.05
B +Tuning (incl. bilinear up/down)	5.5
C +Add mapping and styles	4.85
D +Remove traditional input	4.9
E +Add noise inputs	4.45
F +Mixing regularization	4.2

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

ตัวอย่างของภาพที่สร้างขึ้นมา

ภาพที่ 4.4 ตัวอย่างของภาพที่สร้าง

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

บทที่ 5
สรุปผล อภิปรายผลและข้อเสนอแนะ

จากผลการศึกษางานวิจัยบทที่ 4 ผลการดำเนินงานวิจัย ในหัวข้อการสร้างข้อมูลภาพจาก GANS เพื่อนำไปต่อยอดในด้าน art vector สามารถสรุปผล อภิปรายผลและข้อเสนอแนะการศึกษาได้ดังนี้

5.1 สรุปผลการวิจัย

5.1.1 การเตรียมข้อมูลภาพ

ในการเตรียมข้อมูลรูปภาพได้มีการนำ canvas เทียม มาประยุกต์ใช้งาน เนื่องจากภาพถ่ายที่มีหลายมุม เป็นภาพที่เมื่อเปลี่ยนองศาของภาพจะทำให้ภาพเปลี่ยนรูปร่างไปด้วยโดยจะเอาเฉพาะมุมที่ต้องการมาประยุกต์ใช้เท่านั้น รวมถึงการตัดพื้นหลังที่ไม่เกี่ยวทำให้ผลลัพธ์ที่ได้จากการวิเคราะห์ ข้อมูลภาพมีความละเอียดถูกต้อง และมีความสะดวก รวดเร็วในการเตรียมข้อมูลภาพมากยิ่งขึ้น

5.1.2 การประยุกต์ใช้เทคโนโลยี GANS

ในส่วนของการประยุกต์ใช้เทคโนโลยี GANS กับภาพถ่าย รถยนต์ ได้เลือกใช้ ภาษาไพธอนในการดำเนินงานเนื่องจากเป็นภาษาคอมพิวเตอร์ระดับสูงที่สามารถนำไปประยุกต์ใช้ ได้กับงานในหลายๆด้าน และสามารถนำไปประยุกต์ใช้กับระบบปฏิบัติการอื่นๆได้ ส่วนไลบรารี่ที่ใช้ในการพัฒนาจะใช้ไลบรารี่ TensorFlow กับ PyTorch เป็นหลัก เนื่องจากเป็น deep learning library ที่ใช้งานง่ายและมีประสิทธิภาพสูงในการรันโมเดล ได้รับความนิยมในปัจจุบันในการใช้พัฒนา Machine Learning โดยแนวคิดของเทคโนโลยีการเรียนรู้เชิงลึกนั้นเป็นการนำข้อมูลไม่ว่าจะเป็นข้อมูลภาพตัวอักษร หรืออื่นๆ มาสอนหรือ train ให้โมเดลเข้าใจว่าสิ่งๆนั้นคืออะไร โดยในการ train โมเดลนั้นใน บางครั้งอาจจะต้องใช้ข้อมูลจำนวนมากเพื่อให้โมเดลสามารถเรียนรู้และเข้าใจได้ เมื่อทำการ train โมเดลแล้วจะนำโมเดลที่ได้ไปทดสอบกับภาพใหม่เพื่อให้ได้ผลลัพธ์ออกมา ซึ่งเทคโนโลยีการเรียนรู้เชิงลึกที่นำมาใช้คือ GANS โดยโมเดลสามารถนำไปสร้างข้อมูลภาพขึ้นมา

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

ใหม่ได้จากจุดเริ่มต้นโดยจะมีคุณภาพมากกว่าการทำ data augmentation ทั่วไปคือ การสามารถ style transfer ในสิ่งที่ต้องการได้และสามารถสร้างภาพซ้ำได้ในปริมาณมาก

5.2 อภิปรายผล

งานวิจัยนี้เป็นการประยุกต์ใช้เทคโนโลยีการเรียนรู้เชิงลึก GANS ในการสร้างข้อมูลภาพโดยมุ่งเน้นไปที่การจัดเตรียมข้อมูล และเทคนิคต่างๆเพื่อปรับปรุงการเทรนข้อมูลให้มีคุณภาพและผู้ที่สนใจสามารถนำไปทำการศึกษาต่อได้ เทคโนโลยี GANS กำลังเป็นที่นิยมในปัจจุบัน เมื่อนำมาประยุกต์กับงานด้านการออกแบบ , ด้านสื่อต่างๆ เมื่อเทียบกับการออกแบบแบบรุ่นเก่า ที่ต้องทำบนกระดาษเท่านั้น ในอนาคตหากเทคโนโลยีการ์ดจอมีความสามารถมากขึ้นและมีข้อมูล pretrain ที่หลากหลายขึ้น เทคโนโลยี GANS ช่วยเพิ่มข้อมูลภาพได้อย่างสะดวกและรวดเร็วขึ้น ซึ่งงานวิจัยนี้สอดคล้องกับ Czinger โดยบริษัทนี้ได้มีการนำ เทคโนโลยีการเรียนรู้เชิงลึกมาช่วยในงานออกแบบโดยสร้างเป็น3d-printingmodel ก่อนจะทำเป็น Production จริง

5.3 ข้อเสนอแนะ

5.3.1 สามารถนำไปพัฒนาต่อยอดให้กับธุรกิจต่างๆได้เช่น ธุรกิจการออกแบบ, ธุรกิจสื่อ เป็นต้น

5.3.2 สามารถใช้ภาพถ่ายประเภทอื่นได้เช่น ยานพาหนะ และ อื่นๆ มาประยุกต์กับโมเดล ได้แต่ต้องมีการวิเคราะห์และตรวจสอบรายละเอียดของภาพให้ชัดเจน เนื่องจากถ้าเป็นภาพที่คนละประเภทกันและมุมที่ต่างกันการนำ pretrain ที่ไม่ได้เกี่ยวข้องมา train ภาพจะทำให้ภาพไม่สามารถสร้างภาพขึ้นมาได้

5.3.3 สามารถนำไปประยุกต์ใช้ในงานด้านอื่น ๆ ได้ แต่ต้องมีการปรับโมเดลให้เข้ากับงานนั้นด้วย เนื่องจากโมเดล Deep Learning เป็นโมเดลที่ทำงานเฉพาะด้าน

5.3.4 ในการเตรียมข้อมูลภาพที่ใช้เทรนด์โมเดลควรเก็บข้อมูลให้ถูกต้องทุกครั้งและตัดสิ่งที่ไม่ต้องการจะ train ออก เพื่อให้ผลลัพธ์ที่สร้างออกมามีความถูกต้องและลดระยะเวลาการทำงาน

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

บรรณานุกรม

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

บรรณานุกรม

[1] Jakub Langr and Vladimir Bok (GANs in Action) September 2019

[2] Jason Brownlee (Generative Adversarial Networks with Python) July 2019

[3] Lyle Regenwetter, Brent Curry, Faez Ahmed “BIKED: A Dataset for Computational Bicycle Design with Machine Learning Benchmarks” arXiv:2103.05844v3 [cs.LG] 17 Aug 2021

[4] Rinon Gal , Dana Cohen , Amit Bermano “SWAGAN: A Style-based WAvelet-driven Generative Model”
arXiv:2102.06108v1 [cs.CV] 11 Feb 2021

[5] Ran Yi, Yong-Jin Liu “APDrawingGAN: Generating Artistic Portrait Drawings from Face Photos with Hierarchical GANs” IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019

[6] Tero Karras , Miika Aittala ,Samuli Laine “Training Generative Adversarial Networks with Limited Data”
arXiv:2006.06676v2 [cs.CV] 7 Oct 2020

[7] Tero Karras , Miika Aittala ,Samuli Laine ”Analyzing and Improving the Image Quality of StyleGAN” arXiv:1912.04958v2 [cs.CV] 23 Mar 2020

[8 Yanan Miao , Huan Ma , Jia Cui1 and Xiaoming Tao “Joint Monocular 3D Car Shape Estimation and Landmark Localization via Cascaded Regression” International Conference on Pattern Recognition Applications and Methods (ICPRAM 2018)

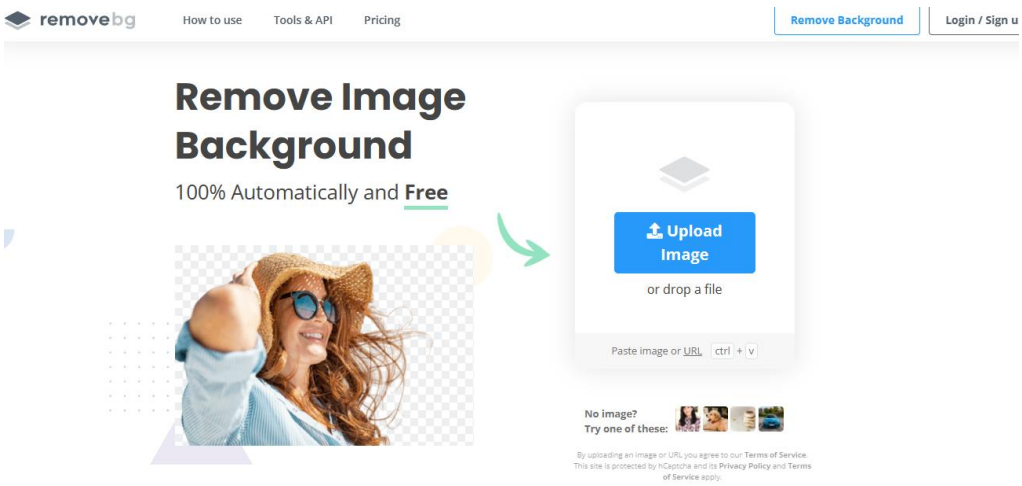
DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

ภาคผนวก

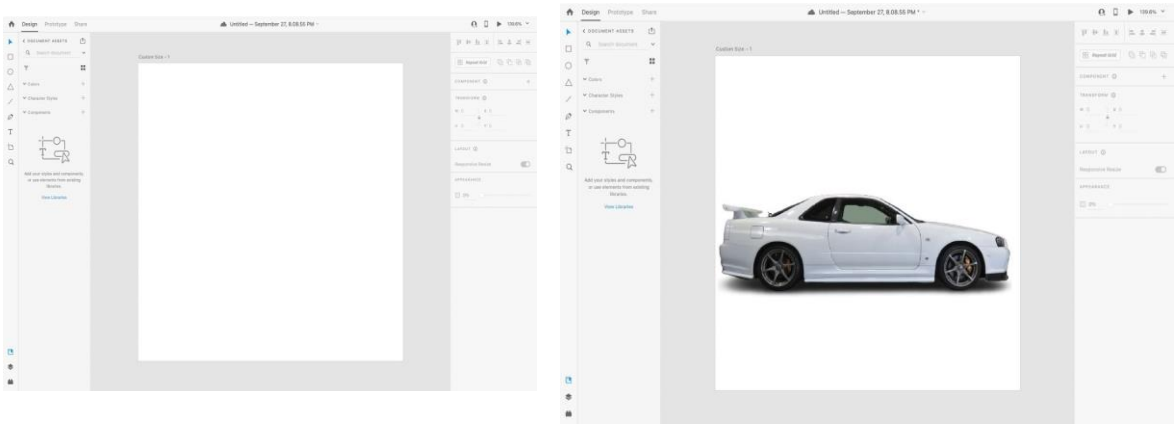
DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

ชุดคำสั่งในกระบวนการ Deep Learning

```
from google.colab import drive  drive.mount('/content/drive')

%tensorflow_version 1.x

!pip3 install --quiet ml4a

from ml4a import image

!pip install ninja
```



ตัวอย่างโปรแกรม ที่ใช้ในการ Remove ภาพพื้นหลัง



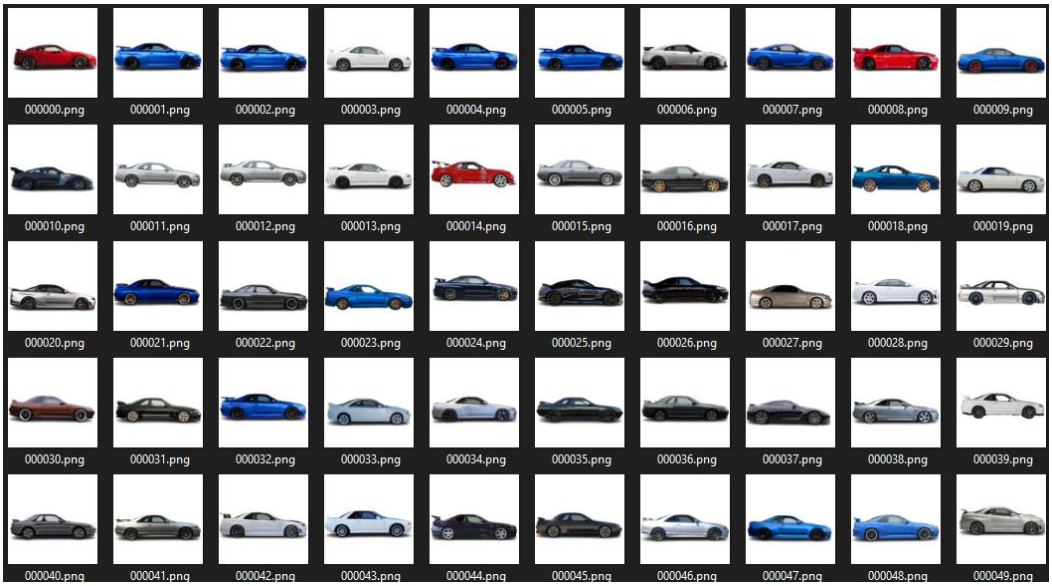
ตัวอย่างโปรแกรม adobe xd set canvas 512px

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

ตัวอย่างการวางรูปที่ตัด Background บนโปรแกรม adobe xd

```
import os
import glob

os.chdir(r"c://Users//chalee//Desktop//art_gtr//gtr//images")
for index, oldfile in enumerate(glob.glob("*.png"), start = 0 ):
    newfile = '00000{}.png'.format(index)
    os.rename (oldfile,newfile)
```

ตัวอย่างการเปลี่ยนชื่อสกุลไฟล์



ตัวอย่างภาพที่พร้อมนำเข้า train model

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์


```
            utils.save_image(
                sample,
                f"{str(i).zfill(6)}.png",
                nrow=1,
                normalize=True,
                range=(-1, 1),
            )


if __name__ == "__main__":
    device = "cuda"

    parser = argparse.ArgumentParser(description="Generate samples from the generator")

    parser.add_argument(
        "--size", type=int, default=512, help="output image size of the generator"
```

ตัวอย่างการปรับขนาดความละเอียดของภาพและสกุลไฟล์ stylegan-pytorch

DPU DHURAKIJ PUNDIT UNIVERSITY มหาวิทยาลัยธุรกิจบัณฑิตย์

ประวัติผู้เขียน

ชื่อ นามสกุล	ชาลี ชีพพิมลชัย
ประวัติการศึกษา	ปริญญาตรี
	BBA Logistics, Materials, and Supply Chain Management
	Bangkok University
	พ.ศ. 2561
ตำแหน่งและสถานที่ทางานปัจจุบัน	Senior Data Analytic
	บริษัท วีจีไอ จำกัด (มหาชน)