1. CNN คือ Deep Learning ชนิดหนึ่ง ที่ใช้โครงข่ายประสาทเทียม ที่จะจำลองการมองเห็นเป็นพื้นที่ย่อยๆ และนำมาผสานกัน เพื่อดูว่าสิ่งนี้คืออะไร จะมีการแยกลักษณะของพื้นที่นั้นๆ ตั้งแต่ จุด เส้นแต่ละแนว พื้นผิว ลวดลาย ไปจนถึง วัตถุ ใช้เพื่อการจำแนกข้อมูลประเภทรูปภาพ

* ลักษณะการทำงาน เริ่มจากการนำ filter หรือ kernal หลายๆ filter ที่สำคัญ ใช้แยกแยะลักษณะหลายๆอย่างประกอบกัน เนื่องจาก filter 1 ตัว จะดึงลักษณะออกมาได้ 1อย่าง ซึ่งจะแตกต่างกันไป เช่น หาขอบรูป, หาความเบลอ, หาความคม โดยรูปภาพ จะประกอบด้วย pixel จากนั้น จะเริ่มนำ filter ไปทาบกับรูปภาพตามขนาดของ filter รูปภาพส่วนที่ทาบกันกับ filter จะนำค่าแต่ละตำแหน่งไป คูณกัน แบบ element-wise และ sum ออกมา และเลื่อนไปที่ตำแหน่งถัดไปจนถึงตำแหน่งสุดท้าย ค่าของตัวเลขที่ได้จะบอกถึงสิ่งที่เราต้องการ detect นั้น ตรงกันมากน้อยแค่ไหน ซึ่งจะเรียกว่า feature map ซึ่งทำให้ จำแนกลักษณะต่างๆในภาพ และยังสามารถลดขนาดของรูปภาพได้อีกด้วย
* CNN จึงเหมาะการงาน OCR เพราะสามารถจำแนกส่วนต่างๆของภาพได้ชัดเจน เช่น ขอบ, รูปร่าง และตัวอักษร และยังสามารถจัดการกับความแปรปรวนของรูปภาพได้ เช่น การหมุน ย่อ/ขยาย การเปลี่ยนแปลงแสง ทำให้สามารถจับคู่และจำตัวตัวอักษรได้แม่นยำ

1. กระบวนการ OCR
2. การป้อนข้อมูล อาจเป็นภาพหรือเอกสารที่ประกอบด้วยข้อความ
3. การปรับปรุงคุณภาพ เพื่อให้ง่ายต่อการประมวลผล
   1. แปลงเป็นภาพขาวดำ เพื่อลดความซับซ้อน โดยจะระบุส่วนที่สว่างเป็นพื้นหลัง และส่วนที่มืดเป็นข้อความ
   2. ปรับเอกสารให้ตรง หรือเอียงเล็กน้อย เพื่อแก้ไขปัญหาการจัดตำแหน่งระหว่างสแกน
   3. การลบนอยส์ หรือ ปรับขอบของรูปข้อความให้เรียบ
   4. ปรับความสว่างและความคมชัด
   5. ปรับขนาดภาพให้เหมาะสม
4. การแยกลักษณะ จะแยกย่อยรูปแบบอักษรออกเป็นส่วนต่างๆ
5. การรู้จำตัวอักษร หลังจากที่แยกลักษณะไว้แล้ว จะนำส่วนต่างๆนั้นมาค้นหาคู่ที่ใกล้เคียงที่สุดในรูปแบบตัวอักษรต่างๆที่จัดเก็บไว้
6. แก้ไขข้อผิดพลาด มีการปรับปรุงความถูกต้องของข้อความตามหลักไวยกรณ์หรือพจนานุกรม ในกรณีที่ข้อความผิดพลาด
7. แปลงข้อมูลเป็นข้อความ
8. การออกแบบ CNN
   * Convolutional Layers กำหนด filters ขนาด 3 X 3 จำนวน 32 filters เพื่อใช้ในการสร้าง feature map และใช้ **Rectified Linear Unit**
   * Pooling Layers ใช้ max pooling ขนาด 2 X 2 เพื่อลดขนาดของ feature map
   * Flatten Layer แปลงข้อมูลให้เป็น vector
   * Fully Connected Layers เชื่อมต่อทุก neuron ในเลเยอร์ก่อนหน้ากับทุก neuron ในเลเยอร์ปัจจุบัน
   * Output layer ใช้ function softmax แปลงผลละพธ์ของความน่าจะเป็น
9. เพื่อลดผลกระทบของสัญญาณรบกวนใน OCR เลือกใช้เทคนิค **Gaussian Blur ที่จะ**ใช้ Gaussian kernel เพื่อทำ convolution กับภาพ ซึ่งจะทำให้ลดสัญญาณรบกวนได้ดี
10. การฝึก CNN สำหรับ OCR
11. เตรียมข้อมูล
    1. รวบรวมภาพที่ประกอบด้วยข้อความ เช่นเอกสารสแกน, ภาพถ่าย หรือ MNIST
    2. ลบภาพที่ไม่มีคุณภาพ และไม่เกี่ยวข้อง ปรับทิศทางให้ตรง
    3. แบ่งชุดข้อมูลสำหรับ ฝึก, ตรวจสอบ, และทดสอบ
    4. เพิ่มความหลากหลายของชุดข้อมูล เช่น เพิ่มนอยส์, ย่อ/ขยาย, หมุน, เปลี่ยนความสว่าง
    5. กำหนด labels ให้แต่ละภาพ
12. ออกแบบโครงสร้างโมลเดล
    1. เลือกโครงสร้าง CNN
    2. กำหนด parameter convolutional layers และ filters, ขนาด kernel, จำนวน **neurons**
    3. เพิ่ม function กระตุ้น
    4. เพิ่ม dropout layer เพื่อลด overfitting  โดยการปิด neurons บางส่วน
13. Mode Compilation ฝึก model และ วัดประสิทธิภาพ
    1. เลือก function loss สำหรับงานจำแนกประเภท
    2. เลือก Optimizer เพื่อปรับปรุงน้ำหนักของ model
    3. เลือกเมตริกสำหรับวัดประสิทธิภาพ
14. การฝึกโมเดล
    1. ป้อนข้อมูล และ labels
    2. ใช้ backpropagation เพื่อปรับปรุงน้ำหนักของโมเดล
    3. ตรวจสอบประสิทธิภาพบน**ชุดตรวจสอบ**
    4. ปรับ  Hyperparameters เพื่อให้ model ทำงานดีขึ้น
    5. บันทึก model ที่ดีที่สุด
15. การประเมิน Model
    1. ใช้**ชุดทดสอบ** เพื่อวัดประสิทธิภาพ
    2. วิเคราะห์ผลลัพธ์ ความถูกต้อง ความแม่นยำ
    3. ปรับปรุง Model หากผลลัพธ์ยังไม่ดีเท่าที่ควร
16. การนำ Model ไปใช้งาน
    1. บันทึก model ที่สามารถนำไปใช้งานได้
    2. สร้าง api หรือ application เพื่อให้ user upload และรับข้อความที่แปลงแล้ว
    3. ทดสอบการใช้งาน
17. หากประสิทธิภาพของ OCR ต่ำหว่าที่คาดหวัง
    * ปรับปรุงชุดข้อมูลที่ใช้ฝึก อาจเพิ่มปริมาณ ความหลากหลาย ลบภาพที่มีคุณภาพต่ำ
    * ปรับปรุงการเตรียมข้อมูล ปรับความสว่าง ความชัด การแบ่งตัวอักษร คำ และบรรทัด
    * ปรับปรุงโครงสร้าง Model เพิ่มจำนวน layers ปรับจำนวน neurons
    * ปรับ Hyperparameters learning rate, batch size
    * เพิ่ม dropout layer เพื่อลด overfitting  โดยการปิด neurons บางส่วน
    * ทดลองกับ model อื่น เพื่อเปรียบเทียบประสิทธิภาพ