AI Project

OmniMetric_Model

จัดทำโดย

นายพงษ์พัฒน์ บางข่า 6604062630358

เสนอ

ผู้ช่วยศาสตราจารย์ ดร.สุวัจชัย กมลสันติโรจน์

รายงานนี้เป็นส่วนหนึ่งของรายวิชา **Artificial Intelligence Software Development**

ภาควิชาวิทยาการคอมพิวเตอร์และสารสนเทศ คณะวิทยาศาสตร์ประยุกต์

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

ภาคเรียนที่ 2 ปีการศึกษา 2568

คำนำ

รายงานฉบับนี้ จัดทำขึ้นเพื่อเป็นส่วนหนึ่งของรายวิชา รายวิชา 040613703 การพัฒนาซอฟต์แวร์ทางปัญญาประดิษฐ์ (Artificial Intelligence Software Development) สำหรับการเป็นเอกสารเพื่อบอกถึงการทำงานของโมเดล ว่ามีแนวคิดอย่างไรบ้าง โดยจะมีการอธิบายเกี่ยวกับการเตรียมข้อมูลและการแสดงคุณลักษณะด้วย

สุดท้ายนี้ผู้เขียนหวังเป็นอย่างยิ่งว่า เอกสารรายงานนี้จะเป็นประโยชน์ แก่ผู้ที่ได้อ่าน เพื่อเข้าใจถึงโมเดลของระบบนี้ ไม่มากก็น้อย

นายพงษ์พัฒน์ บางข่า

ผู้เรียบเรียง

สารบัญ

หัวข้อที่

หน้า

- คำนำ ก

1 ML Model 1

1.1 Model explanation 1

1.2 Data preparation 1-2

1.3 Feature representation 2

1.4 Example 3

ML Model

1.1 Model explanation

ในระบบนี้ เราใช้สถาปัตยกรรมแบบ Multi-Stage Neural Networks โดยผสานการทำงานของโมเดล Deep Learning สองส่วนหลัก เพื่อให้ได้ทั้ง "การระบุตำแหน่ง" และ "การวัดระยะทาง" ที่แม่นยำ ดังนี้

1. Object Detection
   - ใช้ **YOLO** ในการการตรวจจับวัตถุ
   - ทำหน้าที่ระบุ Label ของวัตถุและตำแหน่ง Pixel บนภาพ โดยจะใช้ Bounding Box ที่ได้จาก YOLO เพื่อหาจุดกึ่งกลางของวัตถุนั้นๆมาเป็นตัวแทนของระยะห่างวัตถุ
2. Depth Estimation
   - ใช้ **Depth Pro** หรือ **Depth Anything V2** ในการประมาณความลึกของภาพ
   - ทำหน้าที่แปลงข้อมูลภาพ 2 มิติให้กลายเป็นข้อมูลความลึก (Depth Map) โดยสามารถระบุระยะห่างเป็นหน่วยเมตร (Metric Depth) โดยจะนำจุดกึ่งกลางของ Bounding Box จาก YOLO มา Map เข้ากับ Depth Map ที่สร้างขึ้น เพื่อดึงค่าระยะทางของวัตถุนั้นๆ ออกมา
3. Spatial Transformation
   - ใช้ **Py360Convert** ใช้จัดการกับข้อมูลภาพแบบ 360° (panorama)
   - ทำหน้าที่แปลงภาพที่บิดเบี้ยวของภาพ 360° ให้เป็นภาพแบบ Perspective ก่อนส่งเข้าโมเดล Depth Estimation

1.2 Data preparation

ไม่ว่าจะเป็นภาพชนิดใดข้อมูลดิบจะถูกประมวลผลผ่านขั้นตอน Standardization ก่อนจะแยกกันในภายหลังในแต่ละประเภทของข้อมูล

1. Preprocessing พื้นฐาน
   - ต้องแปลงค่าที่ OpenCV อ่านภาพเป็น BGR เป็น RGB เพื่อให้โมเดล Depth Pro/YOLO ทำงานได้ถูกต้อง
   - Normalization : ภาพจะถูกส่งผ่านฟังก์ชันที่จะแปลง ค่า Pixel (0-255) ให้เป็น Tensor (0.0-1.0) และทำการ Normalize

2. Specific เฉพาะทาง
    - ภาพถ่ายปกติ
        - เราจะประมวลผลรอบเดียว แต่จะมีการ Resize Input ให้มีขนาดที่เหมาะสมแต่ Aspect Ratio ยังคงเหมือนเดิม
    - ภาพ 360°
        - Phase 1 (Visuals) : จะทำการนำภาพให้โมเดล Depth ทำ Depth Map ออกมาเพื่อแสดงผลสวยๆ
        - Phase 2 (Measurement) : จะทำการใช้งาน Py360Convert ในการแปลงภาพโค้งๆ ให้เป็นภาพ Perspective แบบปกติก่อนนำเข้า Model Depth

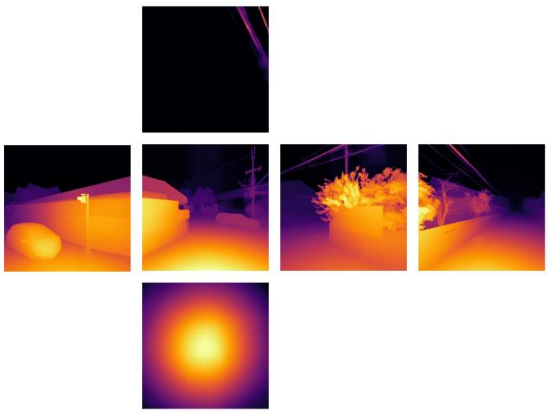
1.3 Feature representation

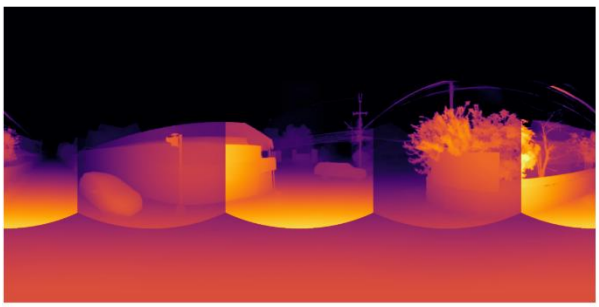
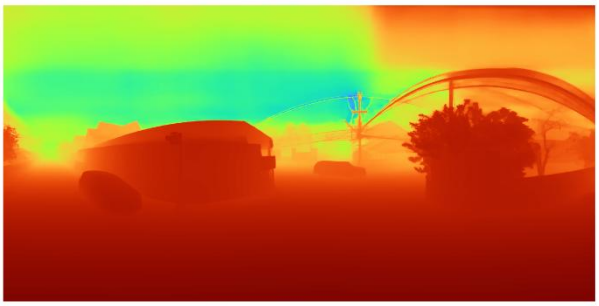
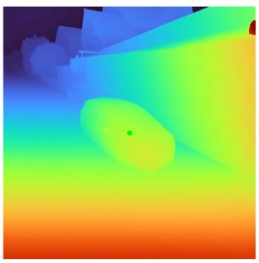
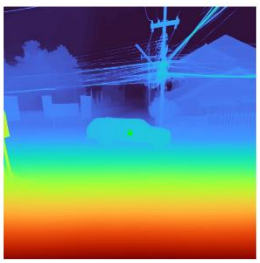
ข้อมูลที่ถูกประมวลผลแล้วจะถูกจัดเก็บและใช้งานในรูปแบบที่ต่างกันตามขั้นตอนดังนี้

1. ข้อมูลรูปภาพ
    - รูปภาพที่ผ่านการเตรียมแล้ว จะถูกแปลงเป็น Tensor 3 มิติ เพื่อใช้เป็น Input หลักของโมเดล
2. ข้อมูลตำแหน่งวัตถุ
    - เมื่อ YOLO เจอวัตถุได้จะได้ข้อมูลเป็น Vector ของตัวเลขชุดหนึ่งที่จะบอกตำแหน่งของ box ที่ YOLO เจอวัตถุ
    - เพื่อใช้ในการระบุตำแหน่งที่จะเป็นตัวแทนของวัตถุนั้นๆ เพื่อใช้หาระยะ
3. ข้อมูลระยะความลึก
    - ผลลัพธ์ที่ได้จะเป็นค่า Array 2 มิติ ที่จะมีขนาดเท่ากับรูปภาพโดยค่าในนั้นจะหมายถึงระยะความลึกในจุดนั้นๆ

1.4 Example

ทางแรก (Cube Map)





ทางสอง (ตามจำนวนวัตถุ)