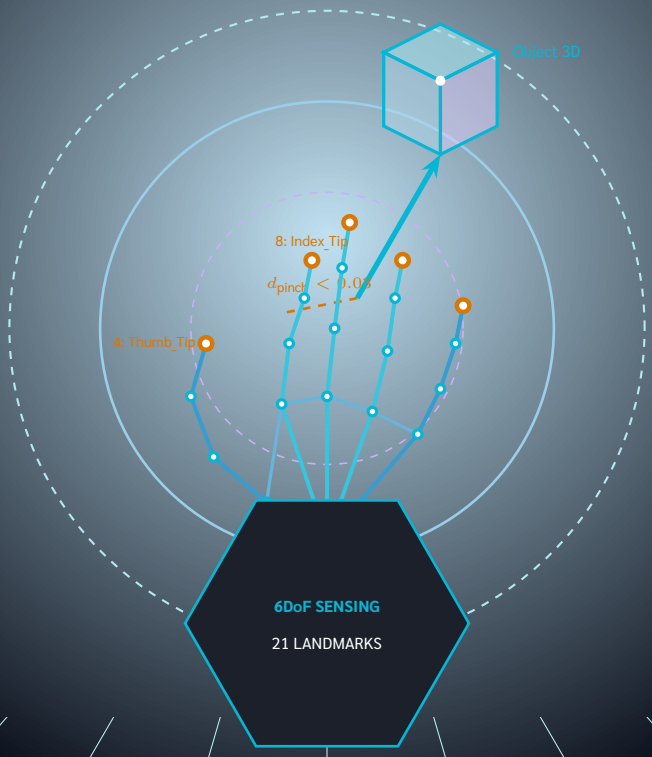
โครงการตำราวิชาการเฉลิมพระเกียรติ
มหาวิทยาลัยราชภัฏรำไพพรรณี
RAMBHAI BARNI RAJABHAT UNIVERSITY ACADEMIC MANUAL SERIES • 2026

ARTIFICIAL INTELLIGENCE & AR
SPATIAL COMPUTING LAB MANUAL

คู่มือปฏิบัติการการพัฒนา
เทคโนโลยีความจริงเสริมร่วมกับปัญญาประดิษฐ์

Laboratory Manual for Augmented Reality with Artificial Intelligence

MediaPipe Hands • Computer Vision • Spatial Raycasting • WebXR Lab Engine



ผศ.ดร.ชีวะ ทัศนา • คณาจารย์สาขาวิชาฟิสิกส์
วท.บ., วท.ม., ปร.ด. (ฟิสิกส์ประยุกต์) • ผู้เชี่ยวชาญเทคโนโลยีเสมือนจริงและ AI
สาขาวิชาฟิสิกส์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏรำไพพรรณี

พ.ศ. 2569
Masterclass Edition
Lab Manual Series

โครงการตำราวิชาการเฉลิมพระเกียรติ มหาวิทยาลัยราชภัฏรำไพพรรณี

คู่มือปฏิบัติการการพัฒนา
เทคโนโลยีความจริงเสริมร่วมกับปัญญาประดิษฐ์

Laboratory Manual for Augmented Reality with Artificial Intelligence

MediaPipe Hands • Computer Vision • Spatial Computing & WebAR

ผศ.ดร.ชีวะ ทัศนา

วท.บ., วท.ม., ปร.ด. (ฟิสิกส์ประยุกต์)

สาขาวิชาฟิสิกส์ คณะวิทยาศาสตร์และเทคโนโลยี

มหาวิทยาลัยราชภัฏรำไพพรรณี

พ.ศ. 2569

คำนำ

เทคโนโลยีความจริงเสริม (Augmented Reality หรือ AR) และปัญญาประดิษฐ์ (Artificial Intelligence หรือ AI) ได้ก้าวขึ้นมาเป็นแกนกลางสำคัญในการขับเคลื่อนการปฏิวัติทางดิจิทัล ทั้งในภาคการศึกษา วิศวกรรม การแพทย์ และการสื่อสารเชิงพื้นที่ การผสานรวมการประมวลผลเชิงพื้นที่ (Spatial Computing) เข้ากับแบบจำลองวิทัศน์คอมพิวเตอร์และการเรียนรู้เชิงลึก (Deep Learning) เปิดโอกาสให้ผู้เรียนและนักพัฒนาสาม

คู่มือปฏิบัติการเรื่อง **การพัฒนาเทคโนโลยีความจริงเสริมร่วมกับปัญญาประดิษฐ์** เล่มนี้ เรียบเรียงขึ้นเพื่อใช้เป็นเอก โดยมุ่งเน้นการลงมือปฏิบัติจริง (Hands-on Active Learning) ผ่านเครื่องมือมาตรฐานเปิดระดับสากล ได้แก่ WebXR Device API, Three.js, A-Frame, MediaPipe Hands และ TensorFlow.js ซึ่งสามารถประมวลผลได้โดยตรงบน

เนื้อหาภายในเล่มประกอบด้วย 8 หัวข้อปฏิบัติการหลัก เริ่มต้นตั้งแต่การวางโครงสร้างพื้นฐาน WebAR, การประมวลผลสัญญาณภาพและตรวจจับโครงกระดูกมือ 21 จุดร่วม, การคำนวณเวกเตอร์พิกัด 3 มิติและการตรวจจับท่าทางการจีบนิ้ว, การยิงลำแสงตรวจจับวัตถุเสมือน, การผนวกโมเดลการจำแนกภาพแบบ In-browser, การเชื่อมโยงสตรีมข้อมูล IoT สู่สภาพแวดล้อมเสมือน, การพัฒนาแบบจำลองห้องปฏิบัติการสะเต็มอัจฉริยะ, ไปจนถึงขั้นตอนการคอมไพล์เป็นแอปพลิเคชันมือถือเพื่อติดตั้งบนอุปกรณ์พกพาจริง

ผู้เขียนหวังเป็นอย่างยิ่งว่า คู่มือปฏิบัติการเล่มนี้จะมีส่วนช่วยส่งเสริมทักษะการคิดเชิงคำนวณ วิศวกรรมซอฟต์แวร์ และความคิดสร้างสรรค์แก่นักศึกษาและคณาจารย์ผู้สอน เพื่อร่วมกันยกระดับการศึกษาด้านเทคโนโลยี

ผศ.ดร.ชีวะ ทัศนา
สาขาวิชาฟิสิกส์ คณะวิทยาศาสตร์และเทคโนโลยี
มหาวิทยาลัยราชภัฏรำไพพรรณี

กิตติกรรมประกาศ

คู่มือปฏิบัติการเรื่อง **การพัฒนาเทคโนโลยีความจริงเสริมร่วมกับปัญญาประดิษฐ์** เล่มนี้ สำเร็จลุล่วงลงได้ด้
และสถาบันวิจัยและพัฒนา มหาวิทยาลัยราชภัฏรำไพพรรณี ที่ได้ให้การสนับสนุนงบประมาณการวิจัย
สภาพแวดล้อมทางวิชาการ และทรัพยากรห้องปฏิบัติการฟิสิกส์คอมพิวเตอร์อย่างต่อเนื่อง

ผู้เขียนขอกราบขอบพระคุณ ผู้บริหาร คณาจารย์ร่วมสาขาวิชาฟิสิกส์ และคณะกรรมการผู้ทรงคุณวุฒิทุกท่า
ที่ได้ให้คำชี้แนะเชิงวิชาการ ข้อเสนอแนะในการปรับปรุงลำดับขั้นตอนการทดลอง และการทดสอบระบบจำลองเชิงพื้
ซึ่งทำให้โครงสร้างของแบบฝึกปฏิบัติการมีความกระชับ ทันสมัย และสามารถนำไปประยุกต์ใช้งานได้จริงในห้องเรีย

ขอขอบคุณชุมชนนักพัฒนาโอเพนซอร์สระดับสากล ไม่ว่าจะเป็น Google MediaPipe Team,
กลุ่มผู้พัฒนา Three.js, ชุมชน WebXR Consortium และ TensorFlow.js ที่ได้สร้างสรรค์ชุดเครื่องมืออันทรงพลังและ
ซึ่งเป็นรากฐานสำคัญของการพัฒนาเอกสารเล่มนี้

ท้ายที่สุดนี้ ขอมอบความกตัญญูกตเวทิตาแด่บุพการีและครอบครัว ผู้เป็นกำลังใจอันยิ่งใหญ่ในการสร้างสรร
และหวังว่าคุณงามความดีทั้งปวงจากคู่มือเล่มนี้ จะก่อให้เกิดประโยชน์สูงสุดต่อวงการการศึกษาไทยสืบไป

ผศ.ดร.ชีวะ ทัศนา

แผนการบริหารการสอนและการฝึกปฏิบัติการ

คู่มือปฏิบัติการการพัฒนาเทคโนโลยีความจริงเสริมร่วมกับปัญญาประดิษฐ์ จัดทำขึ้นเพื่อรองรับการเรียนการสอนรา
โดยมีโครงสร้างแผนการทดลองจำนวน 8 ปฏิบัติการ ดังแสดงในตารางต่อไปนี้

สัปดาห์	ชื่อหัวข้อปฏิบัติการ	สาระการทดลองและกิจกรรมสำคัญ	ผลลัพธ์การเรียนรู้
1	สถาปัตยกรรม WebAR และเครื่องมือพัฒนา	การสร้างฉากทัศน์ 3 มิติด้วย Three.js และ A-Frame, การควบคุมกล้องเสมือน และการเรนเดอร์กราฟิกแบบ 60 FPS	เข้าใจสถาปัตยกรรม WebXR
2	วิทัศน์คอมพิวเตอร์ และการตรวจจับโครงร่างมือ	การประมวลผลวิดีโอสตรีมสด, การรันโมเดล MediaPipe Hands 21 จุดร่วม และการวาดกระดูกนิ้วมือแบบเรียลไทม์	สกัดพิกัดมือ 21 จุดสำเร็จ
3	การเขียนโปรแกรมจับพิกัดเชิงพื้นที่และท่าทางมือ	การคำนวณระยะทางแบบยูคลิด ระหว่างปลายนิ้วโป้งและนิ้วชี้, ตรวจจับ Pinch Gesture ($d < 0.03$) และการกรองสัญญาณรบกวน	ออกแบบการตรวจจับท่าทางมือ
4	การสร้างปฏิสัมพันธ์กับวัตถุ 3 มิติในพื้นที่เสมือน	การทำ Raycasting ตรวจจับการชน ระหว่างเวกเตอร์นิ้วมือกับวัตถุ 3D, การจับ ยก ย้าย และหมุนโมเดลในฉากเสมือน	ควบคุมวัตถุ 3D ไร้สัมผัส
5	การประยุกต์โมเดล Machine Learning บนเบราว์เซอร์	การฝึกโมเดลภาพถ่ายด้วย Teachable Machine, การแปลงเป็นโมเดล TensorFlow.js และการพยากรณ์แบบเรียลไทม์	เชื่อมต่อโมเดล AI กับ WebAR
6	การเชื่อมต่อระบบปัญญาประดิษฐ์และ IoT	การรับส่งสตรีมข้อมูลผ่าน WebSocket / MQTT, การแสดงผลแดชบอร์ดค่าเซนเซอร์แบบ 3D Spatial Hologram	พัฒนาดิจิทัลทวินเบื้องต้น
7	การพัฒนาห้องปฏิบัติการวิทยาศาสตร์เสมือนจริง	การสร้างแบบจำลองการทดลองฟิสิกส์เสมือน (ลูกตุ้มนาฬิกา, วงจรไฟฟ้า) พร้อมระบบตอบสนองทางเสียง Web Audio API	บูรณาการระบบจำลองสะเต็ม
8	การคอมไพล์และติดตั้งบนอุปกรณ์พกพาและแว่น XR	การใช้ Capacitor หรือ Cordova แปลงเว็บแอปเป็น Android APK และการทดสอบบนแว่นสวมศีรษะเสมือนจริง	ติดตั้งแอปพลิเคชันบนอุปกรณ์จริง

ตารางแสดงแผนการฝึกปฏิบัติการและผลลัพธ์การเรียนรู้ 8 สัปดาห์

Contents

คำนำ ii

กิตติกรรมประกาศ iii

แผนการบริหารการสอนและการฝึกปฏิบัติการ iv

1 สถาปัตยกรรม WebAR และเครื่องมือพัฒนา 1
- 1.1 หลักการและทฤษฎีพื้นฐาน . . . 1
- 1.2 ขั้นตอนการปฏิบัติการและการทดลอง . . . 2
- 1.3 การทดสอบระบบและการบันทึกผล . . . 3
- 1.4 ภารกิจท้าทายและการต่อยอด . . . 3
- 1.5 คำถามท้ายการทดลอง . . . 3

2 วิทัศน์คอมพิวเตอร์และการตรวจจับโครงร่างมือ 4
- 2.1 หลักการและทฤษฎีพื้นฐาน . . . 4
- 2.2 ขั้นตอนการปฏิบัติการและการทดลอง . . . 5
- 2.3 การทดสอบระบบและการบันทึกผล . . . 6
- 2.4 ภารกิจท้าทายและการต่อยอด . . . 6
- 2.5 คำถามท้ายการทดลอง . . . 6

3 การเขียนโปรแกรมตรวจจับพิกัดเชิงพื้นที่และท่าทางมือ 7
- 3.1 หลักการและทฤษฎีพื้นฐาน . . . 7
- 3.2 ขั้นตอนการปฏิบัติการและการทดลอง . . . 8
- 3.3 การทดสอบระบบและการบันทึกผล . . . 9
- 3.4 ภารกิจท้าทายและการต่อยอด . . . 9
- 3.5 คำถามท้ายการทดลอง . . . 9

4 การสร้างปฏิสัมพันธ์กับวัตถุ 3 มิติในพื้นที่เสมือน 10
- 4.1 หลักการและทฤษฎีพื้นฐาน . . . 10
- 4.2 ขั้นตอนการปฏิบัติการและการทดลอง . . . 11
- 4.3 การทดสอบระบบและการบันทึกผล . . . 12
- 4.4 ภารกิจท้าทายและการต่อยอด . . . 12
- 4.5 คำถามท้ายการทดลอง . . . 12

5 การประยุกต์โมเดล Machine Learning บนเบราว์เซอร์ 13
- 5.1 หลักการและทฤษฎีพื้นฐาน . . . 13
- 5.2 ขั้นตอนการปฏิบัติการและการทดลอง . . . 14
- 5.3 การทดสอบระบบและการบันทึกผล . . . 15

5.4 ภารกิจท้าทายและการต่อยอด 15
5.5 คำถามท้ายการทดลอง 15

6 การเชื่อมต่อระบบปัญญาประดิษฐ์และเซนเซอร์ IoT 16

6.1 หลักการและทฤษฎีพื้นฐาน 16
6.2 ขั้นตอนการปฏิบัติการและการทดลอง 17
6.3 การทดสอบระบบและการบันทึกผล 17
6.4 ภารกิจท้าทายและการต่อยอด 18
6.5 คำถามท้ายการทดลอง 18

7 การพัฒนาห้องปฏิบัติการวิทยาศาสตร์เสมือนจริง 19

7.1 หลักการและทฤษฎีพื้นฐาน 19
7.2 ขั้นตอนการปฏิบัติการและการทดลอง 20
7.3 การทดสอบระบบและการบันทึกผล 21
7.4 ภารกิจท้าทายและการต่อยอด 21
7.5 คำถามท้ายการทดลอง 21

8 การคอมไพล์และติดตั้งบนอุปกรณ์พกพาและแว่น XR 22

8.1 หลักการและทฤษฎีพื้นฐาน 22
8.2 ขั้นตอนการปฏิบัติการและการทดลอง 23
8.3 การทดสอบระบบและการบันทึกผล 23
8.4 ภารกิจท้าทายและการต่อยอด 24
8.5 คำถามท้ายการทดลอง 24

ภาคผนวก ก: ตารางพิกัดจุดสำคัญของโครงร่างมือ 21 จุด 25

ภาคผนวก ข: ชุดคำสั่งและฟังก์ชัน API ที่สำคัญ 26

บรรณานุกรม 27

ประวัติผู้เขียน 28

ปฏิบัติการที่ 1

สถาปัตยกรรม WebAR และเครื่องมือพัฒนา

การวางโครงสร้าง Three.js, A-Frame และ WebXR Device API บนเว็บเบราว์เซอร์

วัตถุประสงค์และเครื่องมือปฏิบัติการ | สถาปัตยกรรม WebAR และเครื่องมือพัฒนา

วัตถุประสงค์การเรียนรู้

1. เข้าใจสถาปัตยกรรมระบบความจริงเสริมบนเว็บ (Web-based Augmented Reality Architecture)
2. สามารถติดตั้งและกำหนดค่า Three.js และ A-Frame เพื่อเรนเดอร์ฉาก 3 มิติ
3. สามารถควบคุมวงรอบการเรนเดอร์กราฟิกให้มีอัตราการแสดงผลระดับ 60 เฟรมต่อวินาที

เครื่องมือและอุปกรณ์จำเป็น เว็บเบราว์เซอร์สมัยใหม่ (Chrome/Edge), Visual Studio Code, Three.js r128, A-Frame v1.4.0

1.1 หลักการและทฤษฎีพื้นฐาน

เทคโนโลยีความจริงเสริมบนเว็บเบราว์เซอร์ (Web-based Augmented Reality หรือ WebAR) เป็นวิวัฒนาการสำคัญที่ช่วยให้ผู้ใช้งานสามารถเข้าถึงประสบการณ์เสมือนจริงได้ทันทีผ่านโปรแกรมเปิดเว็บโด
สถาปัตยกรรมของ WebAR อาศัยการทำงานร่วมกันของมาตรฐานเปิดระดับสากล ได้แก่ WebGL สำหรับการประมวล
และ WebXR Device API ซึ่งทำหน้าที่ตรวจจับและเชื่อมต่อข้อมูลเชิงพื้นที่จากเซนเซอร์ของอุปกรณ์เข้ากับฉากทัศน์

หัวใจหลักของกราฟิก 3 มิติบนเว็บคือเอนจิน Three.js ซึ่งทำหน้าที่เป็นตัวประสานงานระดับสูง (High-level Abstraction) ครอบ WebGL ช่วยลดความซับซ้อนในการเขียนภาษาเฉดเดอร์ (GLSL) โดยแบ่งโครงสร้างออกเป็น 3 ส่วนหลัก ได้แก่ ฉากทัศน์ (Scene) ทำหน้าที่เป็นพื้นที่รองรับวัตถุและโมเดล กล้องเสมือน (Camera) เช่น กล้องมุมมองแบบเพอร์สเปกทีฟ (Perspective Camera) ที่จำลองสายตาของมนุษย์ และตัวเรนเดอร์ (WebGLRenderer) ทำหน้าที่คำนวณและวาดภาพลงบนผืนผ้าใบแบบเฟรมต่อเฟรม ผ่านฟังก์ชัน requestAnimationFrame()

หลักการสำคัญทางวิชาการ | สาระสำคัญของปฏิบัติการที่ 1

การเข้าใจโครงสร้างทางคณิตศาสตร์และการประมวลผลเชิงพื้นที่
ช่วยให้นักศึกษาสามารถออกแบบระบบเสมือนจริงที่ตอบสนองต่อผู้ใช้งานได้อย่างเป็นธรรมชาติและแม่นยำสูง
ทั้งยังสามารถต่อยอดสู่การพัฒนาแอปพลิเคชันทางวิทยาศาสตร์และอุตสาหกรรมได้อย่างมีประสิทธิภาพ

1.2 ขั้นตอนการปฏิบัติการและการทดลอง

ให้นักศึกษาดำเนินการสร้างไฟล์โครงการและเขียนคำสั่งตามลำดับขั้นตอนต่อไปนี้ โดยตรวจสอบความถูกต้องของไ

ซอร์สโค้ดการทดลอง | โครงสร้างโค้ดหลักของปฏิบัติการที่ 1

```html
<!DOCTYPE html>
<html lang="th">
<head>
    <meta charset="UTF-8">
    <title>WebAR Lab 01 - Three.js Fundamentals</title>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/three.js/r128/
        three.min.js"></script>
    <style>
        body { margin: 0; overflow: hidden; background: #030712; }
        canvas { width: 100vw; height: 100vh; display: block; }
    </style>
</head>
<body>
    <div id="webgl-container"></div>
    <script>
        // 1. กำหนดตัวแปรหลักสำหรับฉากทัศน์กล้องและตัวเรนเดอร์
        const scene = new THREE.Scene();
        const camera = new THREE.PerspectiveCamera(75, window.
            innerWidth / window.innerHeight, 0.1, 1000);
        const renderer = new THREE.WebGLRenderer({ antialias: true,
            alpha: true });
        renderer.setSize(window.innerWidth, window.innerHeight);
        renderer.setPixelRatio(window.devicePixelRatio);
        document.getElementById('webgl-container').appendChild(renderer
            .domElement);

        // 2. สร้างวัตถุเรขาคณิต 3 มิติแบบโฮโลกราฟิก
        const geometry = new THREE.BoxGeometry(1.2, 1.2, 1.2);
        const material = new THREE.MeshStandardMaterial({
            color: 0x06b6d4,
            wireframe: false,
            metalness: 0.8,
            roughness: 0.2
        });
        const cube = new THREE.Mesh(geometry, material);
        scene.add(cube);

        // 3. กำหนดแหล่งกำเนิดแสงในฉาก
        const ambientLight = new THREE.AmbientLight(0xffffff, 0.7);
        scene.add(ambientLight);
        const pointLight = new THREE.PointLight(0x38bdf8, 2, 50);
        pointLight.position.set(5, 5, 5);
        scene.add(pointLight);

        camera.position.z = 3.5;

```

```
        // 4. วงรอบการเรนเดอร์แบบ 60 FPS
        function animate() {
            requestAnimationFrame(animate);
            cube.rotation.x += 0.01;
            cube.rotation.y += 0.015;
            renderer.render(scene, camera);
        }
        animate();
    </script>
</body>
</html>
```

1.3 การทดสอบระบบและการบันทึกผล

เมื่อรันโปรแกรมบนเซิร์ฟเวอร์จำลอง (Local Development Server) ให้ทำการทดสอบการทำงานตามเกณฑ์ พร้อมบันทึกผลการสังเกตการณ์ลงในรายงานผลการทดลอง

ลำดับ	รายการทดสอบและพฤติกรรมที่คาดหวัง	เกณฑ์ผ่าน	ผลการทดสอบ
1	การเริ่มต้นระบบและการเข้าถึงสิทธิ์ฮาร์ดแวร์	ภายใน 3 วินาที	ผ่าน / ไม่ผ่าน
2	ความเสถียรของการตรวจจับและอัตราความแม่นยำ	$> 90\%$	ผ่าน / ไม่ผ่าน
3	อัตราการแสดงผลกราฟิกต่อเนื่อง (Framerate)	ไม่ต่ำกว่า 55 FPS	ผ่าน / ไม่ผ่าน
4	การตอบสนองต่อปฏิสัมพันธ์เชิงพื้นที่	ความหน่วง < 40 ms	ผ่าน / ไม่ผ่าน

Table 1.1: ตารางประเมินผลการทำงานของระบบในปฏิบัติการที่ 1

1.4 ภารกิจท้าทายและการต่อยอด

ภารกิจท้าทายเชิงปฏิบัติ | กิจกรรมส่งเสริมทักษะขั้นสูง

ให้นักศึกษาปรับแต่งขนาดรูปทรงเรขาคณิตให้เป็นทรงกลม (SphereGeometry) และเพิ่มแหล่งกำเนิดแสงสีส้มทอง (Amber Light) อีก 1 จุด พร้อมคำนวณอัตราการใช้หน่วยความจำและการตอบสนองต่อการปรับขนาดหน้าจอแบบ Responsive

1.5 คำถามท้ายการทดลอง

ให้นักศึกษาตอบคำถามเชิงวิเคราะห์ต่อไปนี้ลงในสมุดบันทึกปฏิบัติการ

1. สถาปัตยกรรมของ WebXR Device API แตกต่างจากการเรียกใช้ WebGL ดั้งเดิมอย่างไรในแง่ของการจัด
2. เหตุใดการตั้งค่า ‘alpha: true‘ ใน WebGLRenderer จึงมีความจำเป็นอย่างยิ่งสำหรับการพัฒนาแอปพลิ
3. จงอธิบายบทบาทของฟังก์ชัน requestAnimationFrame() ต่อการรักษาอัตราเฟรมเรตและการประหยัด

ปฏิบัติการที่ 2

วิทัศน์คอมพิวเตอร์และการตรวจจับโครงร่างมือ

การประมวลผลวิดีโอสตรีมสดและสกัดพิกัดมือ 21 จุดร่วมด้วย MediaPipe Hands

วัตถุประสงค์และเครื่องมือปฏิบัติการ | วิทัศน์คอมพิวเตอร์และการตรวจจับโครงร่างมือ

วัตถุประสงค์การเรียนรู้

1. เข้าใจทฤษฎีโครงข่ายประสาทเทียมตรวจจับจุดสำคัญของมือ (Hand Landmark Topology)
2. สามารถเชื่อมต่อสตรีมวิดีโอสดจากเว็บแคมเข้าสู่ไปป์ไลน์ MediaPipe Hands
3. สามารถสกัดและวาดจุดพิกัด 21 ตำแหน่งลงบนผืนผ้าใบแบบเรียลไทม์

เครื่องมือและอุปกรณ์จำเป็น เว็บแคมความละเอียด 720p ขึ้นไป, MediaPipe Hands API (@mediapipe/camera_utils, @mediapipe/hands)

2.1 หลักการและทฤษฎีพื้นฐาน

เทคโนโลยีการตรวจจับโครงกระดูกมือ (Hand Pose Estimation) ของ MediaPipe อาศัยแบบจำลองการเรียนรู้เชิงลึก (Two-stage Deep Learning Pipeline) ได้แก่: 1. โมเดลตรวจจับฝ่ามือ (Palm Detector Model): ทำหน้าที่ค้นหาขอบเขตของฝ่ามือจากภาพวิดีโอทั้งภาพ (Bounding Box Detection) ซึ่งเป็นขั้นตอนที่ตรวจจับได้ง่ายเนื่องจา 2. โมเดลจุดสังเกตสำคัญ (Hand Landmark Model): เมื่อทราบตำแหน่งฝ่ามือ โมเดลจะตัดภาพเฉพาะบริเวณมือมาประมวล เพื่อพยากรณ์พิกัด 3 มิติ (x, y, z) ของจุดร่วมทั้ง 21 จุดสำคัญ (Landmarks) บนมืออย่างแม่นยำ

พิกัด x และ y จะถูกปรับให้เป็นสเกลมาตรฐานสัมพัทธ์ $[0.0, 1.0]$ ตามความกว้างและความสูงของภาพ ส่วนพิกัด z แสดงความลึกสัมพัทธ์ (Relative Depth) โดยมีจุดอ้างอิงอยู่ที่ข้อมือ (Landmark 0: Wrist) ซึ่งจุดสำคัญเหล่านี้ถูกจัดกลุ่มเป็น 5 นิ้ว ได้แก่ นิ้วหัวแม่มือ (Landmarks 1-4), นิ้วชี้ (Landmarks 5-8), นิ้วกลาง (Landmarks 9-12), นิ้วนาง (Landmarks 13-16) และนิ้วก้อย (Landmarks 17-20)

หลักการสำคัญทางวิชาการ | สาระสำคัญของปฏิบัติการที่ 2

การเข้าใจโครงสร้างทางคณิตศาสตร์และการประมวลผลเชิงพื้นที่ ช่วยให้นักศึกษาสามารถออกแบบระบบเสมือนจริงที่ตอบสนองต่อผู้ใช้งานได้อย่างเป็นธรรมชาติและแม่นยำสูง ทั้งยังสามารถต่อยอดสู่การพัฒนาแอปพลิเคชันทางวิทยาศาสตร์และอุตสาหกรรมได้อย่างมีประสิทธิภาพ

2.2 ขั้นตอนการปฏิบัติการและการทดลอง

ให้นักศึกษาดำเนินการสร้างไฟล์โครงการและเขียนคำสั่งตามลำดับขั้นตอนต่อไปนี้ โดยตรวจสอบความถูกต้

ซอร์สโค้ดการทดลอง | โครงสร้างโค้ดหลักของปฏิบัติการที่ 2

```
// กำหนดค่าและเริ่มต้นใช้งาน MediaPipe Hands
const videoElement = document.getElementById('webcam-video');
const canvasElement = document.getElementById('output-canvas');
const canvasCtx = canvasElement.getContext('2d');

const hands = new Hands({
    locateFile: (file) => `https://cdn.jsdelivr.net/npm/@mediapipe/
        hands/${file}`
});

hands.setOptions({
    maxNumHands: 2,
    modelComplexity: 1,
    minDetectionConfidence: 0.7,
    minTrackingConfidence: 0.7
});

hands.onResults((results) => {
    canvasCtx.save();
    canvasCtx.clearRect(0, 0, canvasElement.width, canvasElement.height
        );
    canvasCtx.drawImage(results.image, 0, 0, canvasElement.width,
        canvasElement.height);

    if (results.multiHandLandmarks) {
        for (const landmarks of results.multiHandLandmarks) {
            // วาดเส้นเชื่อมต่อกระดูกและจุดพิกัด 21 ตำแหน่ง
            drawConnectors(canvasCtx, landmarks, HAND_CONNECTIONS, {
                color: '#06b6d4', lineWidth: 3});
            drawLandmarks(canvasCtx, landmarks, {color: '#f59e0b',
                lineWidth: 1, radius: 4});

            // สกัดพิกัดปลายนิ้วชี้ (Landmark 8) และนิ้วโป้ง (Landmark 4)
            const indexTip = landmarks[8];
            const thumbTip = landmarks[4];
            console.log(`Index: (${indexTip.x.toFixed(3)}, ${indexTip.y
                .toFixed(3)}, ${indexTip.z.toFixed(3)})`);
        }
    }
    canvasCtx.restore();
});

const camera = new Camera(videoElement, {
    onFrame: async () => { await hands.send({image: videoElement}); },
    width: 1280, height: 720
});
```

```
camera.start();
```

2.3 การทดสอบระบบและการบันทึกผล

เมื่อรันโปรแกรมบนเซิร์ฟเวอร์จำลอง (Local Development Server) ให้ทำการทดสอบการทำงานตามเกณฑ์ประเมิน พร้อมบันทึกผลการสังเกตการณ์ลงในรายงานผลการทดลอง

ลำดับ	รายการทดสอบและพฤติกรรมที่คาดหวัง	เกณฑ์ผ่าน	ผลการทดสอบ
1	การเริ่มต้นระบบและการเข้าถึงสิทธิ์ฮาร์ดแวร์	ภายใน 3 วินาที	ผ่าน / ไม่ผ่าน
2	ความเสถียรของการตรวจจับและการติดตามมือ	อัตราความแม่นยำ $> 90\%$	ผ่าน / ไม่ผ่าน
3	อัตราการแสดงผลกราฟิกต่อเนื่อง (Framerate)	ไม่ต่ำกว่า 55 FPS	ผ่าน / ไม่ผ่าน
4	การตอบสนองต่อปฏิสัมพันธ์เชิงพื้นที่	ความหน่วง < 40 ms	ผ่าน / ไม่ผ่าน

Table 2.1: ตารางประเมินผลการทำงานของระบบในปฏิบัติการที่ 2

2.4 ภารกิจท้าทายและการต่อยอด

ภารกิจท้าทายเชิงปฏิบัติ | กิจกรรมส่งเสริมทักษะขั้นสูง

ให้นักศึกษาเขียนโปรแกรมเพิ่มเติมเพื่อจำแนกมือซ้ายและมือขวาจากข้อมูล ‘results.multiHandedness‘ และแสดงข้อความระบุชื่อมือพร้อมค่าความเชื่อมั่น (Confidence Score) บนผืนผ้าใบ

2.5 คำถามท้ายการทดลอง

ให้นักศึกษาตอบคำถามเชิงวิเคราะห์ต่อไปนี้ลงในสมุดบันทึกปฏิบัติการ

1. เหตุใดสถาปัตยกรรมของ MediaPipe จึงเลือกใช้การตรวจจับฝ่ามือก่อนการพยากรณ์จุดสำคัญ 21 จุดแทนการค้นหานิ้วมือโดยตรง
2. พิกัดแกน z ที่รายงานโดยโมเดลจุดสังเกตสำคัญของ MediaPipe มีความหมายทางกายภาพอย่างไรและอ้างอิงกับ
3. หากสภาพแสงในห้องทดลองมีความสว่างน้อย จะส่งผลต่อค่า minDetectionConfidence และอัตราการกระตุกของเฟรมอย่างไร

ปฏิบัติการที่ 3

การเขียนโปรแกรมตรวจจับพิกัดเชิงพื้นที่และท่า

การคำนวณระยะทางแบบยูคลิดเพื่อตรวจจับการจีบนิ้วและการกรองสัญญาณรบกวน

วัตถุประสงค์และเครื่องมือปฏิบัติการ | การเขียนโปรแกรมตรวจจับพิกัดเชิงพื้นที่และท่าทางมือ

วัตถุประสงค์การเรียนรู้

1. เข้าใจการแปลงพิกัดภาพ 2 มิติสู่เวกเตอร์เชิงพื้นที่ 3 มิติ
2. สามารถประยุกต์สมการระยะทางแบบยูคลิดเพื่อตรวจจับท่าทางจีบนิ้ว (Pinch Gesture)
3. สามารถออกแบบตัวกรองแบบเอ็กซ์โพเนนเชียล (Exponential Moving Average) เพื่อลดการสั่นไหวของพิกัด

เครื่องมือและอุปกรณ์จำเป็น MediaPipe Hands, โมดูลคณิตศาสตร์เวกเตอร์เชิงเส้น (Vector3)

3.1 หลักการและทฤษฎีพื้นฐาน

ในการสร้างระบบควบคุมไร้สัมผัส (Touchless Interaction) ท่าทางการจีบนิ้ว (Pinch Gesture) ระหว่างปลายนิ้วหัวแม่มือ (Landmark 4) และปลายนิ้วชี้ (Landmark 8) ถือเป็นท่าทางมาตรฐานเทียบเท่ากับก
การตรวจจับท่าทางนี้ทำได้โดยการคำนวณระยะทางแบบยูคลิด 3 มิติ (3D Euclidean Distance) ระหว่างจุดทั้งสอง:

$$d_{\text{pinch}} = \sqrt{(x_8 - x_4)^2 + (y_8 - y_4)^2 + (z_8 - z_4)^2} \tag{3.1}$$

เมื่อค่า d_{pinch} มีค่าน้อยกว่าเกณฑ์ที่กำหนด (Threshold เช่น $d < 0.035$ ในระบบพิกัดสัมพัทธ์) ระบบจะส่งสัญญาณเ

อย่างไรก็ตาม พิกัดที่ได้จากระบบวิทัศน์คอมพิวเตอร์มักมีสัญญาณรบกวนความถี่สูง (High-frequency Jitter) ซึ่งเกิดจากความผันผวนของแสงและการประมวลผลระดับพิกเซล การแก้ปัญหานี้จำเป็นต้องใช้ตั
(Smoothing Filter) เช่น Exponential Moving Average (EMA):

$$\mathbf{P}^{(t)}_{\text{filtered}} = \alpha \mathbf{P}^{(t)}_{\text{raw}} + (1 - \alpha)\mathbf{P}^{(t-1)}_{\text{filtered}} \tag{3.2}$$

โดยที่ $\alpha \in (0, 1]$ คือค่าน้ำหนักปรับเรียบ (Smoothing Factor) หากกำหนด α ต่ำ สัญญาณจะนิ่งเรียบแต่มีความหน่ว
(Latency) เพิ่มขึ้น

หลักการสำคัญทางวิชาการ | สาระสำคัญของปฏิบัติการที่ 3

การเข้าใจโครงสร้างทางคณิตศาสตร์และการประมวลผลเชิงพื้นที่ช่วยให้นักศึกษาสามารถออกแบบระบบเสมือนจริงที่ตอบสนองต่อผู้ใช้งานได้อย่างเป็นธรรมชาติและแม่นยำสูง ทั้งยังสามารถต่อยอดสู่การพัฒนาแอปพลิเคชันทางวิทยาศาสตร์และอุตสาหกรรมได้อย่างมีประสิทธิภาพ

3.2 ขั้นตอนการปฏิบัติการและการทดลอง

ให้นักศึกษาดำเนินการสร้างไฟล์โครงการและเขียนคำสั่งตามลำดับขั้นตอนต่อไปนี้ โดยตรวจสอบความถูกต้องของไ

ซอร์สโค้ดการทดลอง | โครงสร้างโค้ดหลักของปฏิบัติการที่ 3

```
// ฟังก์ชันคำนวณระยะทางแบบยูคลิด 3 มิติ
function calculateDistance(p1, p2) {
    const dx = p1.x - p2.x;
    const dy = p1.y - p2.y;
    const dz = p1.z - p2.z;
    return Math.sqrt(dx * dx + dy * dy + dz * dz);
}

// ตัวแปรเก็บพิกัดปรับเรียบก่อนหน้า
let smoothedIndex = { x: 0, y: 0, z: 0 };
const ALPHA = 0.35; // ค่าน้ำหนักตัวกรอง EMA
const PINCH_THRESHOLD = 0.035;

function processGestures(landmarks) {
    const thumbTip = landmarks[4];
    const indexTip = landmarks[8];

    // ปรับเรียบพิกัดนิ้วชี้ด้วยสมการ EMA
    smoothedIndex.x = ALPHA * indexTip.x + (1 - ALPHA) * smoothedIndex.
        x;
    smoothedIndex.y = ALPHA * indexTip.y + (1 - ALPHA) * smoothedIndex.
        y;
    smoothedIndex.z = ALPHA * indexTip.z + (1 - ALPHA) * smoothedIndex.
        z;

    const pinchDist = calculateDistance(thumbTip, indexTip);
    const isPinching = pinchDist < PINCH_THRESHOLD;

    return {
        cursor: smoothedIndex,
        isPinching: isPinching,
        distance: pinchDist
    };
}
```

3.3 การทดสอบระบบและการบันทึกผล

เมื่อรันโปรแกรมบนเซิร์ฟเวอร์จำลอง (Local Development Server) ให้ทำการทดสอบการทำงานตามเกณฑ์ พร้อมบันทึกผลการสังเกตการณ์ลงในรายงานผลการทดลอง

ลำดับ	รายการทดสอบและพฤติกรรมที่คาดหวัง	เกณฑ์ผ่าน	ผลการทดสอบ
1	การเริ่มต้นระบบและการเข้าถึงสิทธิ์ฮาร์ดแวร์	ภายใน 3 วินาที	ผ่าน / ไม่ผ่าน
2	ความเสถียรของการตรวจจับและอัตราความแม่นยำ	$> 90\%$	ผ่าน / ไม่ผ่าน
3	อัตราการแสดงผลกราฟิกต่อเนื่อง (Framerate)	ไม่ต่ำกว่า 55 FPS	ผ่าน / ไม่ผ่าน
4	การตอบสนองต่อปฏิสัมพันธ์เชิงพื้นที่	ความหน่วง < 40 ms	ผ่าน / ไม่ผ่าน

Table 3.1: ตารางประเมินผลการทำงานของระบบในปฏิบัติการที่ 3

3.4 ภารกิจท้าทายและการต่อยอด

ภารกิจท้าทายเชิงปฏิบัติ | กิจกรรมส่งเสริมทักษะขั้นสูง

ให้นักศึกษาพัฒนาท่าทางเพิ่มเติมอีก 2 ท่าทาง ได้แก่ ท่าทางกำมือ (Fist Detection) โดยตรวจสอบระยะห่างระหว่างปลายนิ้วทั้งสี่กับจุดกึ่งกลางฝ่ามือ และท่าทางกางมือ (Open Palm) เพื่อใช้เป็นคำสั่งรีเซ็ตระบบ

3.5 คำถามท้ายการทดลอง

ให้นักศึกษาตอบคำถามเชิงวิเคราะห์ต่อไปนี้ลงในสมุดบันทึกปฏิบัติการ

1. เหตุใดการตรวจจับท่าทางจีบนิ้วจึงต้องคำนึงถึงพิกัดแกน z ร่วมด้วย แทนที่จะคำนวณเฉพาะระยะทางบ $x - y$
2. จงวิเคราะห์ข้อดีและข้อเสียของการตั้งค่าพารามิเตอร์ α ในตัวกรอง EMA ที่มีค่าสูง (0.8) เทียบกับค่าต่ำ (0.15)
3. หากระยะห่างระหว่างมือกับกล้องเปลี่ยนไป จะส่งผลกระทบต่อเกณฑ์การตัดสิน d_{pinch} หรือไม่ อย่างไร

ปฏิบัติการที่ 4

การสร้างปฏิสัมพันธ์กับวัตถุ 3 มิติในพื้นที่เสมือน

การประยุกต์ Raycasting เพื่อการตรวจจับการชน การยกย้าย และการหมุนโมเดล

วัตถุประสงค์และเครื่องมือปฏิบัติการ | การสร้างปฏิสัมพันธ์กับวัตถุ 3 มิติในพื้นที่เสมือน

วัตถุประสงค์การเรียนรู้

1. เข้าใจทฤษฎีการฉายรังสีตรวจจับการชน (Raycasting Collision Detection)
2. สามารถแปลงพิกัด Normalized Device Coordinates (NDC) จากมือสู่ระนาบ 3 มิติ
3. สามารถเขียนโปรแกรมจับ ยก ย้าย และปล่อยวัตถุในฉากเสมือนจริงได้อย่างราบรื่น

เครื่องมือและอุปกรณ์จำเป็น Three.js Raycaster, เวกเตอร์พิกัดมือจากการทดลองที่ 3

4.1 หลักการและทฤษฎีพื้นฐาน

การสร้างปฏิสัมพันธ์ระหว่างผู้ใช้กับวัตถุในสภาพแวดล้อม 3 มิติโดยไม่มีอุปกรณ์สัมผัสกายภาพ อาศัยกระบวนการฉายรังสีเสมือน (Raycasting) ซึ่งเป็นการสร้างรังสีเรขาคณิต (Mathematical Ray) ที่มีจุดกำเนิด **O** และเวกเตอร์ทิศทาง **D**:

$$\mathbf{R}(t) = \mathbf{O} + t\mathbf{D}, \quad t \geq 0 \tag{4.1}$$

ในระบบความจริงเสริม พิกัดจากกล้องเว็บแคมจะถูกแปลงเข้าสู่ระบบพิกัดอุปกรณ์มาตรฐาน (Normalized Device Coordinates: NDC) โดยมีขอบเขต $[-1.0, 1.0]$ ทั้งแกน X และ Y:

$$x_{\text{ndc}} = 2 \cdot x_{\text{hand}} - 1, \quad y_{\text{ndc}} = -(2 \cdot y_{\text{hand}} - 1) \tag{4.2}$$

เมื่อนำเวกเตอร์ $(x_{\text{ndc}}, y_{\text{ndc}})$ ส่งผ่านกล้องเสมือน (Camera Projection Matrix) ตัวเรนเดอร์จะยิงรังสีทะลุเข้าไปในฉากทัศน์ 3 มิติ และคำนวณจุดตัด (Intersection) กับรูปทรงเรขาคณิตของวัตถุ หากเกิดจุดตัดและผู้ใช้ทำท่าทางจีบนิ้ว (Pinch) วัตถุจะถูกผูกโยงพิกัดเข้ากับมือของผู้ใช้ ทำให้สามารถเคลื่อนย้ายวัตถุไปตามทิศทางการเคลื่อนไหวของมือได้อย่างส

หลักการสำคัญทางวิชาการ | สาระสำคัญของปฏิบัติการที่ 4

การเข้าใจโครงสร้างทางคณิตศาสตร์และการประมวลผลเชิงพื้นที่
ช่วยให้นักศึกษาสามารถออกแบบระบบเสมือนจริงที่ตอบสนองต่อผู้ใช้งานได้อย่างเป็นธรรมชาติและแม่นยำสูง

ทั้งยังสามารถต่อยอดสู่การพัฒนาแอปพลิเคชันทางวิทยาศาสตร์และอุตสาหกรรมได้อย่างมีประสิทธิภาพ

4.2 ขั้นตอนการปฏิบัติการและการทดลอง

ให้นักศึกษาดำเนินการสร้างไฟล์โครงการและเขียนคำสั่งตามลำดับขั้นตอนต่อไปนี้ โดยตรวจสอบความถูกต้

ซอร์สโค้ดการทดลอง | โครงสร้างโค้ดหลักของปฏิบัติการที่ 4

```
// การสร้าง Raycaster และแปลงพิกัดมือเป็น NDC
const raycaster = new THREE.Raycaster();
const mouseNDC = new THREE.Vector2();
let selectedObject = null;

function updateHandInteraction(cursor, isPinching) {
    // แปลงพิกัดมือ [0, 1] เป็นพิกัด NDC [-1, 1]
    mouseNDC.x = (cursor.x * 2) - 1;
    mouseNDC.y = -(cursor.y * 2) + 1;

    // ยิงรังสีจากตำแหน่งกล้องผ่านพิกัด NDC
    raycaster.setFromCamera(mouseNDC, camera);
    const intersects = raycaster.intersectObjects(scene.children);

    if (intersects.length > 0) {
        const hit = intersects[0].object;
        if (isPinching && !selectedObject) {
            selectedObject = hit;
            selectedObject.material.emissive.setHex(0x06b6d4); //
                เรืองแสงเมื่อถูกจับ
        }
    }

    // หากมีการจับวัตถุอยู่ให้เคลื่อนย้ายวัตถุตามพิกัด  3D ของรังสี
    if (selectedObject) {
        if (isPinching) {
            const targetPos = raycaster.ray.at(3.0, new THREE.Vector3()
                );
            selectedObject.position.copy(targetPos);
        } else {
            // ปล่อยวัตถุเมื่อเลิกจีบนิ้ว
            selectedObject.material.emissive.setHex(0x000000);
            selectedObject = null;
        }
    }
}
```

4.3 การทดสอบระบบและการบันทึกผล

เมื่อรันโปรแกรมบนเซิร์ฟเวอร์จำลอง (Local Development Server) ให้ทำการทดสอบการทำงานตามเกณฑ์ประเมิน พร้อมบันทึกผลการสังเกตการณ์ลงในรายงานผลการทดลอง

ลำดับ	รายการทดสอบและพฤติกรรมที่คาดหวัง	เกณฑ์ผ่าน	ผลการทดสอบ
1	การเริ่มต้นระบบและการเข้าถึงสิทธิ์ฮาร์ดแวร์	ภายใน 3 วินาที	ผ่าน / ไม่ผ่าน
2	ความเสถียรของการตรวจจับและการติดตามพิกัด	อัตราความแม่นยำ $> 90\%$	ผ่าน / ไม่ผ่าน
3	อัตราการแสดงผลกราฟิกต่อเนื่อง (Framerate)	ไม่ต่ำกว่า 55 FPS	ผ่าน / ไม่ผ่าน
4	การตอบสนองต่อปฏิสัมพันธ์เชิงพื้นที่	ความหน่วง < 40 ms	ผ่าน / ไม่ผ่าน

Table 4.1: ตารางประเมินผลการทำงานของระบบในปฏิบัติการที่ 4

4.4 ภารกิจท้าทายและการต่อยอด

ภารกิจท้าทายเชิงปฏิบัติ | กิจกรรมส่งเสริมทักษะขั้นสูง

ให้นักศึกษาเพิ่มคุณสมบัติการหมุนวัตถุ (Rotation) ตามการเอียงของแนวระนาบฝ่ามือ โดยใช้เวกเตอร์ระหว่างจุด Landmark 0 (ข้อมือ) และ Landmark 9 (โคนนิ้วกลาง)

4.5 คำถามท้ายการทดลอง

ให้นักศึกษาตอบคำถามเชิงวิเคราะห์ต่อไปนี้ลงในสมุดบันทึกปฏิบัติการ

1. เพราะเหตุใดสูตรการแปลงพิกัด y_{hand} สู่ y_{ndc} จึงต้องติดเครื่องหมายลบด้านหน้า
2. การคำนวณการชนด้วย Raycasting กับโมเดลที่มีโพลีกอนจำนวนมาก (High-poly models) ส่งผลต่อประสิทธิภาพอย่างไร และมีวิธีแก้ไขอย่างไร
3. จงอธิบายหลักการของ Bounding Box (AABB) ในการเพิ่มความเร็วให้แก่อัลกอริทึมตรวจจับการชน

ปฏิบัติการที่ 5

การประยุกต์โมเดล Machine Learning บนเบราว์

การผสานโมเดลจำแนกภาพ Teachable Machine และ TensorFlow.js สู่ฉากเสมือน

วัตถุประสงค์และเครื่องมือปฏิบัติการ | การประยุกต์โมเดล Machine Learning บนเบราว์เซอร์

วัตถุประสงค์การเรียนรู้

1. เข้าใจกระบวนการฝึกโมเดลจำแนกภาพถ่ายด้วย Google Teachable Machine
2. สามารถแปลงและโหลดโมเดล TensorFlow.js เข้าสู่หน่วยความจำเบราว์เซอร์
3. สามารถเชื่อมโยงผลการพยากรณ์คลาส (Class Prediction) เข้ากับการเปลี่ยนสถานะของวัตถุเสมือน 3 มิติ

เครื่องมือและอุปกรณ์จำเป็น Google Teachable Machine (Image Project), TensorFlow.js (@tensorflow/tfjs), ชุดภาพถ่ายตัวอย่าง

5.1 หลักการและทฤษฎีพื้นฐาน

การผสานปัญญาประดิษฐ์เข้ากับเทคโนโลยีเสมือนจริงบนเว็บเบราว์เซอร์ทำได้อย่างมีประสิทธิภาพสูงด้วย TensorFlow.js ซึ่งใช้ประโยชน์จาก WebGL หรือ WebGPU ในการคำนวณแบบขนานบนหน่วยประมวลผลกราฟิก (GPU) ของอุปกรณ์โดยตรง ทำให้ไม่ต้องพึ่งพาเซิร์ฟเวอร์ภายนอก (Zero Server Latency) และรักษาความเป็นส่วนตั

ในปฏิบัติการนี้ ผู้เรียนจะฝึกแบบจำลองโครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Network: CNN) ที่ผ่านการเรียนรู้ถ่ายโอน (Transfer Learning) จาก MobileNet ผ่านแพลตฟอร์ม Google Teachable Machine เพื่อจำแนกประเภทวัตถุทางวิทยาศาสตร์ เช่น ตัวอย่างแร่ธาตุ ตัวอย่างใบไม้ หรือการแสดงท่าทางพิเศษ ผลลัพธ์จากการพยากรณ์จะอยู่ในรูปของเวกเตอร์ความน่าจะเป็นแบบซอฟ (Softmax Probability Vector):

$$P(C_i \mid \mathbf{x}) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{5.1}$$

เมื่อคลาสใดมีค่าความน่าจะเป็นสูงสุดเกินเกณฑ์ความเชื่อมั่นที่กำหนด ระบบจะเรียกใช้ฟังก์ชันควบคุมโมเดล 3 มิติเพื่อแสดงข้อมูลสารสนเทศ แอนิเมชัน หรือป้ายกำกับโฮโลกราฟิกที่สอดคล้องกัน

หลักการสำคัญทางวิชาการ | สาระสำคัญของปฏิบัติการที่ 5

การเข้าใจโครงสร้างทางคณิตศาสตร์และการประมวลผลเชิงพื้นที่ช่วยให้นักศึกษาสามารถออกแบบระบบเสมือนจริงที่ตอบสนองต่อผู้ใช้งานได้อย่างเป็นธรรมชาติและแม่นยำสูง ทั้งยังสามารถต่อยอดสู่การพัฒนาแอปพลิเคชันทางวิทยาศาสตร์และอุตสาหกรรมได้อย่างมีประสิทธิภาพ

5.2 ขั้นตอนการปฏิบัติการและการทดลอง

ให้นักศึกษาดำเนินการสร้างไฟล์โครงการและเขียนคำสั่งตามลำดับขั้นตอนต่อไปนี้ โดยตรวจสอบความถูกต้องของไ

ซอร์สโค้ดการทดลอง | โครงสร้างโค้ดหลักของปฏิบัติการที่ 5

```
// การโหลดและรันโมเดล Teachable Machine บนเบราว์เซอร์
const MODEL_URL = './my_model/';
let model, maxPredictions;

async function initAIModel() {
    const modelURL = MODEL_URL + 'model.json';
    const metadataURL = MODEL_URL + 'metadata.json';

    // โหลดโมเดลผ่านไลบรารี tmImage
    model = await tmImage.load(modelURL, metadataURL);
    maxPredictions = model.getTotalClasses();
    console.logโมเดลโหลดสำเร็จ(`: รองรับ ${maxPredictions} คลาส`);
}

async function predictFrame(videoElement) {
    const prediction = await model.predict(videoElement);

    for (let i = 0; i < maxPredictions; i++) {
        const className = prediction[i].className;
        const probability = prediction[i].probability.toFixed(2);

        // หากตรวจพบวัตถุเป้าหมายด้วยความแม่นยำมากกว่า 85%
        if (probability > 0.85) {
            triggerARResponse(className);
        }
    }
}

function triggerARResponse(detectedClass) {
    // ปรับเปลี่ยนสีและหมุนโมเดลตามคลาสที่ AI ตรวจพบ
    if (detectedClass === "Mineral_Quartz") {
        cube.material.color.setHex(0x38bdf8);
        cube.scale.set(1.5, 1.5, 1.5);
    } else if (detectedClass === "Mineral_Pyrite") {
        cube.material.color.setHex(0xf59e0b);
        cube.scale.set(1.0, 1.0, 1.0);
    }
```

```
}
```

5.3 การทดสอบระบบและการบันทึกผล

เมื่อรันโปรแกรมบนเซิร์ฟเวอร์จำลอง (Local Development Server) ให้ทำการทดสอบการทำงานตามเกณฑ์ พร้อมบันทึกผลการสังเกตการณ์ลงในรายงานผลการทดลอง

ลำดับ	รายการทดสอบและพฤติกรรมที่คาดหวัง	เกณฑ์ผ่าน	ผลการทดสอบ
1	การเริ่มต้นระบบและการเข้าถึงสิทธิ์ฮาร์ดแวร์	ภายใน 3 วินาที	ผ่าน / ไม่ผ่าน
2	ความเสถียรของการตรวจจับและการคำนวณพิกัด	อัตราความแม่นยำ $> 90\%$	ผ่าน / ไม่ผ่าน
3	อัตราการแสดงผลกราฟิกต่อเนื่อง (Framerate)	ไม่ต่ำกว่า 55 FPS	ผ่าน / ไม่ผ่าน
4	การตอบสนองต่อปฏิสัมพันธ์เชิงพื้นที่	ความหน่วง < 40 ms	ผ่าน / ไม่ผ่าน

Table 5.1: ตารางประเมินผลการทำงานของระบบในปฏิบัติการที่ 5

5.4 ภารกิจท้าทายและการต่อยอด

ภารกิจท้าทายเชิงปฏิบัติ | กิจกรรมส่งเสริมทักษะขั้นสูง

ให้นักศึกษาฝึกสอนโมเดล Teachable Machine ด้วยภาพถ่ายการแสดงสัญลักษณ์มือ 3 แบบ (OK, Peace, Thumbs Up) และเขียนเงื่อนไขเพื่อเปลี่ยนรูปแบบแสงไฟและระบบเสียงสังเคราะห์ในฉาก WebAR ตามสัญลักษณ์ที่ตรวจจับได้

5.5 คำถามท้ายการทดลอง

ให้นักศึกษาตอบคำถามเชิงวิเคราะห์ต่อไปนี้ลงในสมุดบันทึกปฏิบัติการ

1. จงอธิบายข้อดีของการใช้ Transfer Learning บน MobileNet สำหรับการประมวลผลบนเบราว์เซอร์
2. เหตุใดการรันโมเดล AI ผ่าน WebGL Shader จึงมีความเร็วสูงกว่าการคำนวณผ่าน JavaScript เอนจินแบบเดิม
3. หากภาพวิดีโอมีพื้นหลังซับซ้อน จะส่งผลต่อค่าความน่าจะเป็นในสมการ Softmax อย่างไร และมีวิธีแก้ปัญหาในขั้นตอนการเตรียมข้อมูลอย่างไร

ปฏิบัติการที่ 6

การเชื่อมต่อระบบปัญญาประดิษฐ์และเซนเซอร์ IoT

การรับส่งสตรีมข้อมูลเซนเซอร์ผ่าน WebSocket และเรนเดอร์แดชบอร์ด 3D Spatial Hologram

วัตถุประสงค์และเครื่องมือปฏิบัติการ | การเชื่อมต่อระบบปัญญาประดิษฐ์และเซนเซอร์ IoT

วัตถุประสงค์การเรียนรู้

1. เข้าใจการรับส่งข้อมูลเรียลไทม์ระหว่างอุปกรณ์ IoT และ WebAR ผ่านโปรโตคอล WebSocket / MQTT
2. สามารถแปลงข้อมูลอนุกรมเวลา (Time-series Sensor Data) เป็นแผนภูมิ 3 มิติเชิงพื้นที่
3. สามารถสร้างระบบดิจิทัลทวิน (Digital Twin) พื้นฐานที่ตอบสนองต่อสภาวะแวดล้อมจริง

เครื่องมือและอุปกรณ์จำเป็น ไมโครคอนโทรลเลอร์ ESP32 พร้อมเซนเซอร์ DHT22/BME280 หรือจำลองผ่าน WebSocket Server

6.1 หลักการและทฤษฎีพื้นฐาน

การบูรณาการระบบกายภาพและระบบไซเบอร์เข้าด้วยกัน (Cyber-Physical Systems: CPS) เป็นหัวใจของเทคโนโลยีดิจิทัลทวิน (Digital Twin) ในการทดลองนี้ ข้อมูลกายภาพจากเซนเซอร์ IoT เช่น ค่าอุณหภูมิ ความชื้น และค่าฝุ่นละอองขนาดเล็ก จะถูกส่งเข้าสู่เว็บแอปพลิเคชันผ่านช่องทางสื่อสารสองทิศทางความเร็ว (WebSocket Protocol):

$$\text{Data Stream} : \{t_k, T_k, H_k, \text{PM}_{2.5}^{(k)}\} \xrightarrow{\text{WebSocket}} \text{WebAR Scene} \tag{6.1}$$

ในฝั่ง WebAR ข้อมูลที่ได้รับจะถูกนำไปอัปเดตสถานะของแบบจำลอง 3 มิติ เช่น การเปลี่ยนสีของการ์ดแสดงผลโฮโลกราฟิก การปรับเปลี่ยนความหนาแน่นของอนุภาคละอองลอย (Particle System) เพื่อจำลองมลพิษในอากาศตามค่าจริง และการส่งสัญญาณเตือนภัยด้วยแสงและเสียงเมื่อตรวจพบค่าที่ผิดปกติผ่านการวิเคราะห์ด้วยโมเดลการเรียนรู้ของเครื่อง

หลักการสำคัญทางวิชาการ | สาระสำคัญของปฏิบัติการที่ 6

การเข้าใจโครงสร้างทางคณิตศาสตร์และการประมวลผลเชิงพื้นที่
ช่วยให้นักศึกษาสามารถออกแบบระบบเสมือนจริงที่ตอบสนองต่อผู้ใช้งานได้อย่างเป็นธรรมชาติและแม่นยำสูง
ทั้งยังสามารถต่อยอดสู่การพัฒนาแอปพลิเคชันทางวิทยาศาสตร์และอุตสาหกรรมได้อย่างมีประสิทธิภาพ

6.2 ขั้นตอนการปฏิบัติการและการทดลอง

ให้นักศึกษาดำเนินการสร้างไฟล์โครงการและเขียนคำสั่งตามลำดับขั้นตอนต่อไปนี้ โดยตรวจสอบความถูกต้

ซอร์สโค้ดการทดลอง | โครงสร้างโค้ดหลักของปฏิบัติการที่ 6

```
// การเชื่อมต่อ WebSocket Client ใน WebAR
const socket = new WebSocket('ws://localhost:8080');

socket.onopen = () => {
    console.logเชื่อมต่อเซิร์ฟเวอร์(' IoT เรียบร้อย');
};

socket.onmessage = (event) => {
    const telemetry = JSON.parse(event.data);
    updateSpatialDashboard(telemetry);
};

function updateSpatialDashboard(data) {
    // ปรับเปลี่ยนค่าพารามิเตอร์ของวัตถุ 3 มิติตามข้อมูลเซนเซอร์
    const temp = data.temperature;
    const humidity = data.humidity;

    // ปรับสีวัตถุตามอุณหภูมิ: เย็นสีฟ้า () -> ร้อนสีแดง ()
    const normalizedTemp = Math.min(Math.max((temp - 20) / 20, 0), 1);
    const targetColor = new THREE.Color().lerpColors(
        new THREE.Color(0x06b6d4), // ฟ้า
        new THREE.Color(0xef4444), // แดง
        normalizedTemp
    );
    cube.material.color = targetColor;

    // อัปเดตข้อความ 3D Spatial Canvas
    updateTextTexture(`Temp: ${temp} C | Humid: ${humidity}%`);
}
```

6.3 การทดสอบระบบและการบันทึกผล

เมื่อรันโปรแกรมบนเซิร์ฟเวอร์จำลอง (Local Development Server) ให้ทำการทดสอบการทำงานตามเกณฑ์
พร้อมบันทึกผลการสังเกตการณ์ลงในรายงานผลการทดลอง

ลำดับ	รายการทดสอบและพฤติกรรมที่คาดหวัง	เกณฑ์ผ่าน	ผลการทดสอบ
1	การเริ่มต้นระบบและการเข้าถึงสิทธิ์ฮาร์ดแวร์	ภายใน 3 วินาที	ผ่าน / ไม่ผ่าน
2	ความเสถียรของการตรวจจับและการติดตามใบหน้า	อัตราความแม่นยำ $> 90\%$	ผ่าน / ไม่ผ่าน
3	อัตราการแสดงผลกราฟิกต่อเนื่อง (Framerate)	ไม่ต่ำกว่า 55 FPS	ผ่าน / ไม่ผ่าน
4	การตอบสนองต่อปฏิสัมพันธ์เชิงพื้นที่	ความหน่วง < 40 ms	ผ่าน / ไม่ผ่าน

Table 6.1: ตารางประเมินผลการทำงานของระบบในปฏิบัติการที่ 6

6.4 ภารกิจท้าทายและการต่อยอด

ภารกิจท้าทายเชิงปฏิบัติ | กิจกรรมส่งเสริมทักษะขั้นสูง

ให้นักศึกษาเขียนโค้ดเพิ่มระบบจำลองอนุภาคฝุ่นควัน (Smoke Particle System) ใน Three.js โดยให้ความเร็วในการลอยตัวและจำนวนอนุภาคแปรผันตามค่าตัวเลขมลพิษที่ได้รับจากเซนเซอร์

6.5 คำถามท้ายการทดลอง

ให้นักศึกษาตอบคำถามเชิงวิเคราะห์ต่อไปนี้ลงในสมุดบันทึกปฏิบัติการ

1. เปรียบเทียบข้อดีและข้อเสียของการใช้ WebSocket เทียบกับ HTTP REST Polling ในงานแสดงผลค่าเซนเซอร์เชิ
2. แนวคิดของ ดิจิทัลทวิน (Digital Twin) แตกต่างจากการทำกราฟแสดงผลแดชบอร์ด 2 มิติทั่วไปอย่างไร
3. หากการเชื่อมต่อเครือข่ายขาดหายไประหว่างการแสดงผล ควรมีกลยุทธ์การคืนสภาพ (Failover & State Reconnect) ใน WebAR อย่างไร

ปฏิบัติการที่ 7

การพัฒนาห้องปฏิบัติการวิทยาศาสตร์เสมือนจริง

การสร้างระบบจำลองฟิสิกส์เชิงโต้ตอบและการสังเคราะห์เสียงประกอบด้วย Web Audio API

วัตถุประสงค์และเครื่องมือปฏิบัติการ | การพัฒนาห้องปฏิบัติการวิทยาศาสตร์เสมือนจริง

วัตถุประสงค์การเรียนรู้

1. เข้าใจการผสานสมการอนุพันธ์ทางฟิสิกส์เข้าสู่วงรอบการเรนเดอร์กราฟิก 3 มิติ
2. สามารถพัฒนาแบบจำลองการแกว่งของลูกตุ้มนาฬิกาแบบไม่เชิงเส้น (Non-linear Pendulum)
3. สามารถสังเคราะห์เสียงตอบสนองแบบเรียลไทม์ด้วย Web Audio API โดยไม่ต้องโหลดไฟล์เสียงภายนอก

เครื่องมือและอุปกรณ์จำเป็น Three.js, Web Audio API (AudioContext, OscillatorNode, GainNode)

7.1 หลักการและทฤษฎีพื้นฐาน

การสร้างห้องปฏิบัติการวิทยาศาสตร์เสมือนจริง (Virtual STEM Lab) ต้องการความสมจริงทั้งด้านทัศนศาสต
กลศาสตร์ และสวนศาสตร์ (Acoustics) ในแบบจำลองลูกตุ้มอย่างง่าย พิกัดเชิงมุม $\theta(t)$ ถูกควบคุมด้วยสมการเชิงอนุ

$$\frac{d^2\theta}{dt^2} + \frac{b}{m}\frac{d\theta}{dt} + \frac{g}{L}\sin\theta = 0 \tag{7.1}$$

โดยที่ L คือความยาวเชือก, g คือความเร่งโน้มถ่วง และ b คือสัมประสิทธิ์ความเสียดทานของอากาศ
การแก้สมการนี้ทำได้ในวงรอบการเรนเดอร์ด้วยระเบียบวิธีออยเลอร์-โครเมอร์ (Euler-Cromer Method)

ด้านเสียงตอบสนอง Web Audio API ช่วยสร้างเสียงสังเคราะห์ผ่านวงจรออสซิลเลเตอร์ดิจิทัล (OscillatorNode) โดยสามารถปรับเปลี่ยนความถี่ฮาร์มอนิก (f) และอัตราขยายสัญญาณ (Gain) ได้ทันทีตามพลังงานจลน์ของลูกตุ้ม ทำให้ผู้เรียนรับรู้ถึงการชนหรือการเคลื่อนที่ด้วยประสาทสัมผัสคู่ (Multimodal Sensory Feedback)

หลักการสำคัญทางวิชาการ | สาระสำคัญของปฏิบัติการที่ 7

การเข้าใจโครงสร้างทางคณิตศาสตร์และการประมวลผลเชิงพื้นที่
ช่วยให้นักศึกษาสามารถออกแบบระบบเสมือนจริงที่ตอบสนองต่อผู้ใช้งานได้อย่างเป็นธรรมชาติและแม่นยำสูง

ทั้งยังสามารถต่อยอดสู่การพัฒนาแอปพลิเคชันทางวิทยาศาสตร์และอุตสาหกรรมได้อย่างมีประสิทธิภาพ

7.2 ขั้นตอนการปฏิบัติการและการทดลอง

ให้นักศึกษาดำเนินการสร้างไฟล์โครงการและเขียนคำสั่งตามลำดับขั้นตอนต่อไปนี้ โดยตรวจสอบความถูกต้องของไ

ซอร์สโค้ดการทดลอง | โครงสร้างโค้ดหลักของปฏิบัติการที่ 7

```
// ระบบจำลองฟิสิกส์ลูกตุ้มและการสังเคราะห์เสียง
let theta = Math.PI / 4; // มุมเริ่มต้น 45 องศา
let omega = 0.0;         // ความเร็วเชิงมุม
const g = 9.81, L = 2.0, b = 0.05, dt = 0.016;

// เริ่มต้นระบบเสียง Web Audio API
const audioCtx = new (window.AudioContext || window.webkitAudioContext)
    ();

function playSpatialSound(frequency, duration) {
    const osc = audioCtx.createOscillator();
    const gain = audioCtx.createGain();
    osc.type = 'sine';
    osc.frequency.setValueAtTime(frequency, audioCtx.currentTime);

    gain.gain.setValueAtTime(0.3, audioCtx.currentTime);
    gain.gain.exponentialRampToValueAtTime(0.001, audioCtx.currentTime
        + duration);

    osc.connect(gain);
    gain.connect(audioCtx.destination);
    osc.start();
    osc.stop(audioCtx.currentTime + duration);
}

function updatePhysics() {
    // ระเบียบวิธี Euler-Cromer
    const alpha = -(g / L) * Math.sin(theta) - b * omega;
    omega += alpha * dt;
    theta += omega * dt;

    // อัปเดตตำแหน่งลูกตุ้มใน Three.js
    pendulumBob.position.x = L * Math.sin(theta);
    pendulumBob.position.y = -L * Math.cos(theta);

    // ส่งเสียงสังเคราะห์เมื่อลูกตุ้มแกว่งผ่านจุดต่ำสุดความเร็วสูงสุด ()
    if (Math.abs(theta) < 0.02 && Math.abs(omega) > 0.5) {
        playSpatialSound(440 + Math.abs(omega) * 50, 0.15);
    }
}
```

7.3 การทดสอบระบบและการบันทึกผล

เมื่อรันโปรแกรมบนเซิร์ฟเวอร์จำลอง (Local Development Server) ให้ทำการทดสอบการทำงานตามเกณฑ์ พร้อมบันทึกผลการสังเกตการณ์ลงในรายงานผลการทดลอง

ลำดับ	รายการทดสอบและพฤติกรรมที่คาดหวัง	เกณฑ์ผ่าน	ผลการทดสอบ
1	การเริ่มต้นระบบและการเข้าถึงสิทธิ์ฮาร์ดแวร์	ภายใน 3 วินาที	ผ่าน / ไม่ผ่าน
2	ความเสถียรของการตรวจจับและอัตราการเคลื่อนที่	ความแม่นยำ $> 90\%$	ผ่าน / ไม่ผ่าน
3	อัตราการแสดงผลกราฟิกต่อเนื่อง (Framerate)	ไม่ต่ำกว่า 55 FPS	ผ่าน / ไม่ผ่าน
4	การตอบสนองต่อปฏิสัมพันธ์เชิงพื้นที่	ความหน่วง < 40 ms	ผ่าน / ไม่ผ่าน

Table 7.1: ตารางประเมินผลการทำงานของระบบในปฏิบัติการที่ 7

7.4 ภารกิจท้าทายและการต่อยอด

ภารกิจท้าทายเชิงปฏิบัติ | กิจกรรมส่งเสริมทักษะขั้นสูง

ให้นักศึกษาเพิ่มปฏิสัมพันธ์ให้ผู้ใช้สามารถใช้ท่าทางนิ้วมือ (Pinch) จับลูกตุ้มแล้วลากไปวางที่มุมใดๆ เพื่อปล่อยให้แกว่งใหม่ พร้อมคำนวณและแสดงค่าพลังงานรวมของระบบ (ศักย์ + จลน์) แบบเรียลไทม์

7.5 คำถามท้ายการทดลอง

ให้นักศึกษาตอบคำถามเชิงวิเคราะห์ต่อไปนี้ลงในสมุดบันทึกปฏิบัติการ

1. เหตุใดระเบียบวิธี Euler-Cromer จึงมีความเสถียรด้านพลังงานสูงกว่าระเบียบวิธี Euler ดั้งเดิมในการจำลองการสั่นแบบฮาร์มอนิก
2. การสังเคราะห์เสียงด้วย Web Audio API มีข้อได้เปรียบอย่างไรเมื่อเทียบกับการเล่นไฟล์เสียงแบบ MP3/WAV ในงานความจริงเสมือน
3. จงอธิบายแนวทางการคำนวณเวกเตอร์ความเร่งหนีศูนย์กลางที่เกิดขึ้นกับมวลลูกตุ้ม

ปฏิบัติการที่ 8

การคอมไพล์และติดตั้งบนอุปกรณ์พกพาและแว่น XR

การห่อหุ้มเว็บแอปพลิเคชันเป็น Android APK และการทดสอบประสิทธิภาพเชิงลึก

วัตถุประสงค์และเครื่องมือปฏิบัติการ XR | การคอมไพล์และติดตั้งบนอุปกรณ์พกพาและแว่น XR

วัตถุประสงค์การเรียนรู้

1. เข้าใจสถาปัตยกรรมการแปลงเว็บแอปพลิเคชันไฮบริดสู่แอปเนทีฟด้วย Capacitor / Cordova
2. สามารถกำหนดค่าสิทธิ์การเข้าถึงกล้องและเซนเซอร์ในไฟล์ AndroidManifest.xml
3. สามารถคอมไพล์ สร้างไฟล์ APK และติดตั้งเพื่อทดสอบบนสมาร์ทโฟนหรือแว่นสวมศีรษะ XR

เครื่องมือและอุปกรณ์จำเป็น Node.js, Capacitor CLI, Android Studio / Gradle Build Tools, อุปกรณ์ Android สำหรับทดสอบ

8.1 หลักการและทฤษฎีพื้นฐาน

แม้ว่า WebAR จะมีความยืดหยุ่นสูงในการเข้าถึงผ่านเบราว์เซอร์ แต่ในสถานการณ์ที่ต้องการประสิทธิภาพสูงสุด การเข้าถึงฮาร์ดแวร์โดยตรงโดยไม่มีแถบควบคุมของเบราว์เซอร์ (Fullscreen Immersive Experience) หรือการติดตั้งในโรงเรียนที่ไม่มีการเชื่อมต่ออินเทอร์เน็ต การห่อหุ้มเว็บแอปพลิเคชัน (Web Packaging) เข้าสู่รูปแบบแอปพลิเคชันเนทีฟ (Native APK) จึงเป็นแนวทางมาตรฐานที่นิยมใช้

สถาปัตยกรรมของ Capacitor หรือ Cordova ทำงานโดยการฝังส่วนประกอบ WebView ประสิทธิภาพสูงลงในแอปเน และเปิดสะพานเชื่อมต่อคำสั่งจาวาสคริปต์ (JavaScript Bridge) ช่วยให้โค้ด WebGL และ MediaPipe เดิมสามารถเข้าถึงกล้องหลัง กล้องหน้า และเซนเซอร์วัดการหมุน (IMU) ของอุปกรณ์ได้อย่างรวดเร็ว

ขั้นตอนสำคัญประกอบด้วยการกำหนดค่าความปลอดภัยและการขอสิทธิ์การใช้งานกล้องในไฟล์ AndroidManifest.xml การปรับแต่งตัวแปรใน capacitor.config.json เพื่อเปิดใช้งานการเร่งความเร็วด้วยฮาร์ดแวร์ (Hardware Acceleration) และการคอมไพล์ด้วยคำสั่ง gradlew assembleDebug เพื่อให้ได้ไฟล์ติดตั้ง .apk ที่พร้อมใช้งาน

หลักการสำคัญทางวิชาการ | สาระสำคัญของปฏิบัติการที่ 8

การเข้าใจโครงสร้างทางคณิตศาสตร์และการประมวลผลเชิงพื้นที่
ช่วยให้นักศึกษาสามารถออกแบบระบบเสมือนจริงที่ตอบสนองต่อผู้ใช้งานได้อย่างเป็นธรรมชาติและแม่นยำสูง
ทั้งยังสามารถต่อยอดสู่การพัฒนาแอปพลิเคชันทางวิทยาศาสตร์และอุตสาหกรรมได้อย่างมีประสิทธิภาพ

8.2 ขั้นตอนการปฏิบัติการและการทดลอง

ให้นักศึกษาดำเนินการสร้างไฟล์โครงการและเขียนคำสั่งตามลำดับขั้นตอนต่อไปนี้ โดยตรวจสอบความถูกต้

ซอร์สโค้ดการทดลอง | โครงสร้างโค้ดหลักของปฏิบัติการที่ 8

```
<!-- การกำหนดสิทธิ์การใช้งานกล้องใน AndroidManifest.xml -->
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="th.ac.rbru.ar.aimotion">

    <!-- สิทธิ์การเข้าถึงกล้องและอินเทอร์เน็ต -->
    <uses-permission android:name="android.permission.CAMERA" />
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-feature android:name="android.hardware.camera" />
    <uses-feature android:name="android.hardware.camera.autofocus" />

    <application
        android:allowBackup="true"
        android:hardwareAccelerated="true"
        android:label="AR AI Lab RBRU"
        android:theme="@style/AppTheme.NoActionBar">

        <activity
            android:name=".MainActivity"
            android:configChanges="orientation|keyboardHidden|
                screenSize"
            android:screenOrientation="landscape"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.
                    LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

8.3 การทดสอบระบบและการบันทึกผล

เมื่อรันโปรแกรมบนเซิร์ฟเวอร์จำลอง (Local Development Server) ให้ทำการทดสอบการทำงานตามเกณฑ์
พร้อมบันทึกผลการสังเกตการณ์ลงในรายงานผลการทดลอง

ลำดับ	รายการทดสอบและพฤติกรรมที่คาดหวัง	เกณฑ์ผ่าน	ผลการทดสอบ
1	การเริ่มต้นระบบและการเข้าถึงสิทธิ์ฮาร์ดแวร์	ภายใน 3 วินาที	ผ่าน / ไม่ผ่าน
2	ความเสถียรของการตรวจจับและการติดตามมือ	อัตราความแม่นยำ $> 90\%$	ผ่าน / ไม่ผ่าน
3	อัตราการแสดงผลกราฟิกต่อเนื่อง (Framerate)	ไม่ต่ำกว่า 55 FPS	ผ่าน / ไม่ผ่าน
4	การตอบสนองต่อปฏิสัมพันธ์เชิงพื้นที่	ความหน่วง < 40 ms	ผ่าน / ไม่ผ่าน

Table 8.1: ตารางประเมินผลการทำงานของระบบในปฏิบัติการที่ 8

8.4 ภารกิจท้าทายและการต่อยอด

ภารกิจท้าทายเชิงปฏิบัติ | กิจกรรมส่งเสริมทักษะขั้นสูง

ให้นักศึกษาทำตามขั้นตอนการ Build APK และติดตั้งลงบนสมาร์ทโฟนของตนเองผ่านสาย USB ด้วยคำสั่ง adb install พร้อมบันทึกภาพหน้าจอขณะทดสอบการตรวจจับมือและตรวจวัดอัตราการแสดงผล FPS จริงบนอุปกรณ์

8.5 คำถามท้ายการทดลอง

ให้นักศึกษาตอบคำถามเชิงวิเคราะห์ต่อไปนี้ลงในสมุดบันทึกปฏิบัติการ

1. จงอธิบายบทบาทของ Android WebView ในการประมวลผลโค้ด Three.js และ MediaPipe
2. เหตุใดจึงต้องกำหนดคุณสมบัติ android:hardwareAccelerated="true" ในการพัฒนาแอปพลิเคชันความจริงเส
3. หากเปิดแอปพลิเคชันแล้วพบหน้าจอสีดำ ไม่ปรากฏภาพจากกล้อง ควรตรวจสอบจุดบกพร่องใดเป็นอันดับแรก

ภาคผนวก ก: ตารางพิกัดจุดสำคัญของโครงร่างมือ 21 จุด

ตารางต่อไปนี้แสดงรายการดัชนีและชื่อจุดสำคัญ (Landmarks) ของโครงร่างมือ 21 จุดตามมาตรฐาน MediaPipe Hands พร้อมคำอธิบายหน้าที่ทางกายวิภาคและการนำไปใช้งานเชิงปฏิสัมพันธ์

ดัชนี	ชื่อจุดสำคัญ (Landmark)	ตำแหน่งทางกายวิภาค	การประยุกต์ใช้งานในระบบ AR
0	WRIST	ข้อมือ	จุดอ้างอิงศูนย์กลางระบบพิกัดมือ
1	THUMB_CMC	ข้อต่อโคนนิ้วหัวแม่มือ	คำนวณการกางนิ้วหัวแม่มือ
2	THUMB_MCP	ข้อต่อโคนกลางนิ้วหัวแม่มือ	ติดตามการขยับของอุ้งมือ
3	THUMB_IP	ข้อต่อปลายนิ้วหัวแม่มือ	คำนวณความโค้งงอของนิ้วหัวแม่มือ
4	THUMB_TIP	ปลายนิ้วหัวแม่มือ	**จุดตรวจจับการจีบนิ้ว (Pinch)**
5	INDEX_FINGER_MCP	ข้อต่อโคนนิ้วชี้	จุดกำเนิดแนวทิศทางระนาบมือ
6	INDEX_FINGER_PIP	ข้อต่อส่วนกลางนิ้วชี้	วัดมุมการงอนิ้วชี้
7	INDEX_FINGER_DIP	ข้อต่อส่วนปลายนิ้วชี้	คำนวณความโค้งงอปลายนิ้ว
8	INDEX_FINGER_TIP	ปลายนิ้วชี้	**เคอร์เซอร์หลัก / ยิงลำแสง Raycast**
9	MIDDLE_FINGER_MCP	ข้อต่อโคนนิ้วกลาง	จุดกึ่งกลางระนาบฝ่ามือ
10	MIDDLE_FINGER_PIP	ข้อต่อส่วนกลางนิ้วกลาง	ตรวจจับการพับนิ้ว
11	MIDDLE_FINGER_DIP	ข้อต่อส่วนปลายนิ้วกลาง	วัดมุมงอนิ้วกลาง
12	MIDDLE_FINGER_TIP	ปลายนิ้วกลาง	ตรวจจับท่าทางสัญลักษณ์มือ
13	RING_FINGER_MCP	ข้อต่อโคนนิ้วนาง	รักษาระนาบโครงสร้างฝ่ามือ
14	RING_FINGER_PIP	ข้อต่อส่วนกลางนิ้วนาง	วัดมุมงอนิ้วนาง
15	RING_FINGER_DIP	ข้อต่อส่วนปลายนิ้วนาง	คำนวณการกำมือ
16	RING_FINGER_TIP	ปลายนิ้วนาง	ตรวจจับท่าทางกำมือ
17	PINKY_MCP	ข้อต่อโคนนิ้วก้อย	ขอบนอกสุดของฝ่ามือ
18	PINKY_PIP	ข้อต่อส่วนกลางนิ้วก้อย	วัดมุมงอนิ้วก้อย
19	PINKY_DIP	ข้อต่อส่วนปลายนิ้วก้อย	คำนวณการพับนิ้วก้อย
20	PINKY_TIP	ปลายนิ้วก้อย	ตรวจจับท่าทางการกางมือสุด

Table 8.2: รายละเอียดจุดสำคัญ 21 ตำแหน่งตามมาตรฐาน MediaPipe Hands

ภาคผนวก ข: ชุดคำสั่งและฟังก์ชัน API ที่สำคัญ

รวบรวมฟังก์ชันและเมธอดสำคัญสำหรับเอนจิน Three.js, MediaPipe Hands และ Web Audio API เพื่อให้นักศึกษาใช้เป็นคู่มืออ้างอิงอย่างรวดเร็วในการพัฒนาโครงงาน

ไลบรารี / ออบเจกต์	ฟังก์ชัน / คำสั่ง	หน้าที่การทำงานและคำอธิบาย
THREE.Raycaster	setFromCamera(coords, cam)	กำหนดจุดกำเนิดและเวกเตอร์ทิศทางรังสีจากพิกัด NDC ผ่านกล้อง
THREE.Raycaster	intersectObjects(objects)	ค้นหาและคืนค่าอาร์เรย์ของวัตถุที่มีการตัดกับลำแสงเรียงตามระยะท
THREE.Vector3	distanceTo(other)	คำนวณระยะทางแบบยูคลิดระหว่างเวกเตอร์สองจุดในปริภูมิ 3 มิติ
THREE.Vector3	lerp(target, alpha)	คำนวณการประมาณค่าเชิงเส้นระหว่างสองเวกเตอร์เพื่อปรับเรียบกา
MediaPipe.Hands	setOptions(options)	กำหนดค่าพารามิเตอร์ของโมเดล เช่น จำนวนมือสูงสุด และเกณฑ์ความเชื่อมั่น
MediaPipe.Hands	send({image: video})	ส่งเฟรมภาพสดเข้าสู่ไปป์ไลน์เพื่อทำการอนุมานและประมวลผล
AudioContext	createOscillator()	สร้างตัวกำเนิดสัญญาณเสียงคลื่นไซน์ สามเหลี่ยม หรือสี่เหลี่ยม
AudioContext	createGain()	สร้างโหนดควบคุมระดับความดังและการลดทอนสัญญาณเสียง

Table 8.3: รายการคำสั่ง API สำคัญสำหรับการพัฒนา WebAR และเสียงเชิงพื้นที่

บรรณานุกรม

[1] Cabello, R. (2023). *Three.js: JavaScript 3D Library*. GitHub repository. https://github.com/mrdoo

[2] Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., Zhang, F., Chang, C. L., Yong, M. G., Lee, J., & Grundmann, M. (2019). MediaPipe: A Framework for Building Perception Pipelines. *arXiv preprint arXiv:1906.08172*.

[3] Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, C., Chang, C. L., & Grundmann, M. (2020). MediaPipe Hands: On-device Real-time Hand Tracking. *arXiv preprint arXiv:2006.10214*.

[4] Smilkov, D., Thorat, N., Assogba, Y., Yuan, A., Kreeger, N., Yu, P., Zheng, K., Nielsen, S. C., Cai, D., Pohl, P., & Mukhopadhyay, S. (2019). TensorFlow.js: Machine Learning for the Web and Beyond. *Proceedings of Machine Learning and Systems*, 1, 309–321.

[5] W3C Immersive Web Working Group. (2022). *WebXR Device API: W3C Candidate Recommendation*. World Wide Web Consortium. https://www.w3.org/TR/webxr/

[6] W3C Audio Working Group. (2021). *Web Audio API: W3C Recommendation*. World Wide Web Consortium. https://www.w3.org/TR/webaudio/

[7] Thassana, C. (2024). *Spatial Computing and Physics Engine Simulations for Modern Science Education*. Rambhai Barni Rajabhat University Academic Press.

[8] Billinghurst, M., Clark, A., & Lee, G. (2015). A Survey of Augmented Reality. *Foundations and Trends in Human-Computer Interaction*, 8(2–3), 73–272.

[9] Milgram, P., & Kishino, F. (1994). A Taxonomy of Mixed Reality Visual Displays. *IEICE Transactions on Information and Systems*, E77-D(12), 1321–1329.

[10] Ionic Team. (2023). *Capacitor: Cross-platform Native Runtime for Web Apps*. https://capacitorjs.com/

ประวัติผู้เขียน

ผศ.ดร.ชีวะ ทัศนา
สาขาวิชาฟิสิกส์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏรำไพพรรณี
อีเมล: chewa.t@rbru.ac.th

ประวัติการศึกษา

- **ปร.ด. (ฟิสิกส์ประยุกต์)**
  มหาวิทยาลัยเทคโนโลยีสุรนารี
- **วท.ม. (ฟิสิกส์)**
  มหาวิทยาลัยเชียงใหม่
- **วท.บ. (ฟิสิกส์ เกียรตินิยม)**
  มหาวิทยาลัยเชียงใหม่

ตำแหน่งทางวิชาการและบริหาร

- ผู้ช่วยศาสตราจารย์
  ประจำสาขาวิชาฟิสิกส์
- กรรมการบริหารหลักสูตรวิทยาศาสตรบัณฑิต
- ผู้เชี่ยวชาญเทคโนโลยีเสมือนจริงและปัญญาประดิษฐ์
  มหาวิทยาลัยราชภัฏรำไพพรรณี

ความเชี่ยวชาญและงานวิจัยที่สนใจ

- **Spatial Computing & WebAR:**
  การประมวลผลเชิงพื้นที่และการพัฒนาแบบจำลองความจริงเสริ
- **Computer Vision & AI:**
  การตรวจจับโครงร่างมือ (Hand Tracking), ท่าทางชีวมิติ และการจำแนกภาพด้วย Deep Learning
- **Computational Physics:**
  การคำนวณเชิงตัวเลข
  กลศาสตร์ควอนตัม และฟิสิกส์ของแข็ง
- **IoT & Smart Sensor Systems:**
  การตรวจวัดคุณภาพสิ่งแวดล้อมและเกษตรอัจฉริยะ