



การพัฒนาระบบซอฟต์แวร์เทคโนโลยี AR สำหรับการจับต้องวัตถุในโลกเสมือนจริงผ่านแอปพลิเคชันบนอุปกรณ์สมาร์ตโฟน

Development of an AR Software System for Virtual Object Interaction via
Smartphone Applications (AR-VOI)

อิศวงศ์ ยูวพันธุ์กุล ภูรี เทพไพสร์ วีรภัทร กาญจนกันติ

ยุภาพร เปรมกมล¹

¹โรงเรียนวิทยาศาสตร์จุฬารามราชวิทยาลัย ชลบุรี, ตำบลหนองซาก, อำเภอบ้านบึง, จังหวัดชลบุรี 20170

Email: koo_took@hotmail.com

บทคัดย่อ

โครงการนี้เป็นการพัฒนาเทคโนโลยี AR สำหรับการจับวัตถุในโลกเสมือนจริงผ่านแอปพลิเคชันบนสมาร์ตโฟนที่รองรับ Google ARCore สามารถโต้ตอบกับผู้ใช้งานได้ คณะผู้จัดทำได้พัฒนาซอฟต์แวร์ที่มีความสามารถในการสร้างแอปพลิเคชันร่วมกับเทคโนโลยี Hand Tracking (MediaPipe) ในการตรวจจับพิคัดมือเพื่อควบคุมวัตถุ 3D ในรูปแบบการจับหมุน และย่อขยาย โดยมีกระบวนการทำงานเริ่มต้นจากการขออนุมัติเข้าถึงกล้องถ่ายภาพ เมื่อได้รับการอนุญาต ผู้ใช้จะสามารถสแกนพื้นที่ผิวโดยรอบเพื่อจัดวางวัตถุเสมือนจริงลงบนระนาบดังกล่าว โดยสามารถเคลื่อนย้ายตำแหน่งวัตถุได้ผ่านการสัมผัสหน้าจอสมาร์ตโฟน และสามารถปฏิสัมพันธ์กับวัตถุ ทั้งการจับ หมุน หรือย่อขยายวัตถุ ได้ด้วยการขยับมือของผู้ใช้งานในบริเวณพื้นที่หลังกล้อง ผลการทดสอบพบว่าระบบซอฟต์แวร์สามารถทำงานบนระบบปฏิบัติการ Android ได้อย่างมีประสิทธิภาพ ช่วยเพิ่มขีดความสามารถของเทคโนโลยี AR แบบดั้งเดิม และพร้อมที่จะนำไปประยุกต์ใช้เป็นตัวแบบนวัตกรรมสื่อการเรียนรู้ทางการศึกษา รวมถึงเป็นแนวทางให้ผู้พัฒนารุ่นหลังนำไปต่อยอดระบบในรูปแบบต่างๆ ต่อไป

คำสำคัญ: Google ARCore, MediaPipe, Hand Tracking

บทนำ

ในปัจจุบันเทคโนโลยีด้านดิจิทัลและอุปกรณ์สมาร์ตโฟนมีการพัฒนาอย่างรวดเร็ว ส่งผลให้การเรียนรู้ ไม่ได้จำกัดอยู่เพียงภายในห้องเรียนอีกต่อไป ผู้คนสามารถเข้าถึงข้อมูล ความรู้ และสื่อการเรียนรู้ต่างๆ ได้ทุกที่ทุกเวลา ผ่านอุปกรณ์พกพาและอินเทอร์เน็ต อีกทั้งเทคโนโลยีสมัยใหม่ยังช่วยเพิ่มความน่าสนใจและประสิทธิภาพในการเรียนรู้ให้มากยิ่งขึ้น โดยเฉพาะเทคโนโลยี Augmented Reality (AR) ที่สามารถผสมผสานวัตถุเสมือนจริงเข้ากับสภาพแวดล้อมจริง ทำให้ผู้ใช้งานเกิดการเรียนรู้ผ่านการมองเห็นและการโต้ตอบได้อย่างสมจริงจากความก้าวหน้าของเทคโนโลยี

AR ผู้พัฒนาจึงเล็งเห็นถึงความสำคัญของการนำเทคโนโลยีดังกล่าวมาประยุกต์ใช้ร่วมกับการศึกษาและการนำเสนอข้อมูล เพื่อช่วยเพิ่มความสะดวก ความเข้าใจ และสร้างประสบการณ์การเรียนรู้ที่น่าสนใจยิ่งขึ้น โดยใช้ Unity Technologies Unity เป็นเครื่องมือหลักในการพัฒนาแอปพลิเคชัน ร่วมกับ AR Foundation และ ARCore เพื่อให้สามารถแสดงผลวัตถุเสมือนจริงบนอุปกรณ์โทรศัพท์มือถือระบบ Android ได้อย่างมีประสิทธิภาพ รวมถึงใช้ Visual Studio Code (VS Code) ในการเขียนและแก้ไขโปรแกรมภาษา C# ภายในระบบนอกจากนี้ เทคโนโลยี AR ยังช่วยให้ผู้ใช้งานสามารถเรียนรู้และโต้ตอบเนื้อหาได้โดยตรงผ่านโทรศัพท์มือถือ ทำให้การเข้าถึง

ข้อมูลมีความสะดวกมากขึ้น และสามารถใช้งานได้ในชีวิตประจำวัน อีกทั้งยังช่วยส่งเสริมการเรียนรู้ด้วยตนเองและเพิ่มความน่าสนใจให้กับสื่อการเรียนรู้ในยุคดิจิทัล ดังนั้น โครงการนี้จึงถูกพัฒนาขึ้นเพื่อประยุกต์ใช้เทคโนโลยี Augmented Reality ร่วมกับเครื่องมือสมัยใหม่ เพื่อสร้างแอปพลิเคชันที่ช่วยเพิ่มประสิทธิภาพในการเรียนรู้ การนำเสนอข้อมูล และสร้างประสบการณ์การใช้งานที่ทันสมัย สะดวก และเหมาะสมกับการใช้งานในปัจจุบัน อีกทั้งยังสามารถใช้เป็นต้นแบบสำหรับนักวิจัยและนักพัฒนาในการศึกษาวิเคราะห์ และต่อยอดพัฒนาเทคโนโลยี AR ไปประยุกต์ใช้ในด้านอื่น ๆ ได้ในอนาคต เช่น ด้านการศึกษา การแพทย์ การท่องเที่ยว และงานด้านสื่อดิจิทัลต่างๆ

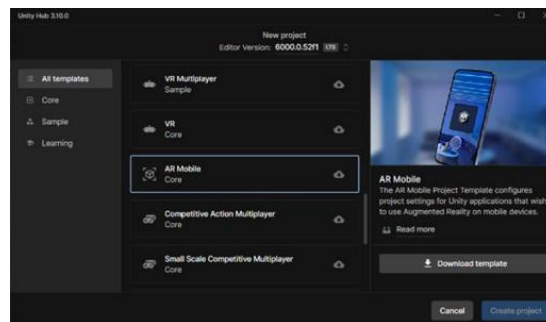
วัตถุประสงค์

1. เพื่อพัฒนาแอปพลิเคชัน AR ที่สามารถตรวจจับวัตถุได้อย่างแม่นยำ และโต้ตอบกับวัตถุภายในสภาพแวดล้อมจริงได้อย่างมีประสิทธิภาพ
2. เพื่อส่งเสริมการพัฒนาเทคโนโลยี AR ให้มีความทันสมัย, ความสามารถที่สูงขึ้น และรองรับการใช้งานในอนาคตได้ดียิ่งขึ้น
3. เพื่อเพิ่ม ความสะดวก, ประสิทธิภาพในการวิจัย, และสนับสนุนการพัฒนาของ นักวิจัยและนักพัฒนารุ่นต่อไป

วิธีการดำเนินงาน

1. สร้างสภาพแวดล้อมความจริง

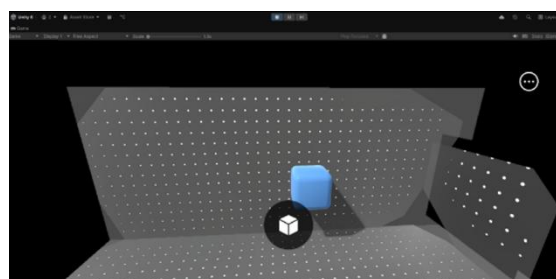
การสร้างสภาพแวดล้อมความจริงเพื่อเป็นพื้นหลังในโปรแกรม เพื่อให้โปรแกรมมีสภาพแวดล้อมความจริงใช้สำหรับการเป็นที่วางหรือที่จัดเตรียมวัตถุเสมือนจริงต่างๆ



ภาพที่ 1 Template AR Mobile

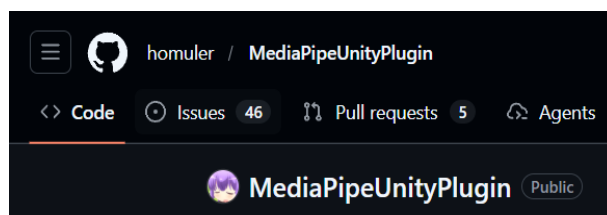
2. สร้าง Template AR Mobile

สร้าง Template AR Mobile ภายใน Unity เพื่อใช้เป็นสภาพแวดล้อมสำหรับการพัฒนาแอปพลิเคชัน Augmented Reality (AR) และช่วยเพิ่มความสะดวกในการพัฒนา โดย Template AR Mobile จะติดตั้ง AR Foundation และฟังก์ชันพื้นฐานสำหรับการตรวจจับพื้นผิว (Plane Detection) มาให้อัตโนมัติ ทำให้สามารถเริ่มพัฒนาแอปพลิเคชัน AR ได้รวดเร็วและมีประสิทธิภาพมากยิ่งขึ้น



ภาพที่ 2 ตัวอย่างการตรวจจับพื้นผิว

3. สร้างการตรวจจับมือ



ภาพที่ 3 github homuler/MediaPipeUnityPlugin

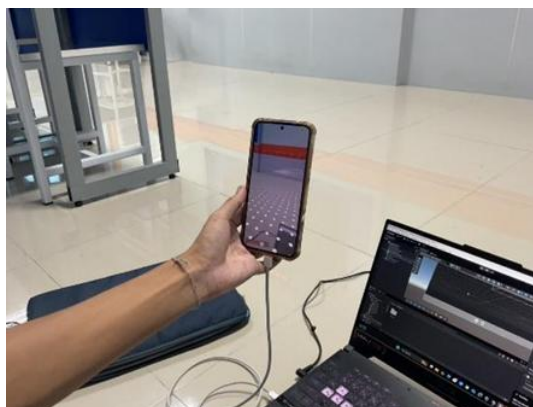
ทำการดาวน์โหลดไฟล์ UnityPackage และนำเข้าไปยัง Template ที่ได้สร้างไว้ เมื่อติดตั้งเรียบร้อยแล้วจึงเลือก Scene ชื่อ Hand Landmark Detection ซึ่งเป็นฉากตัวอย่างที่พัฒนาโดย Homuler เพื่อใช้ทดสอบการตรวจจับมือของระบบ รวมถึงปรับปรุงและเพิ่มความเสถียรในการตรวจจับมือให้มีความแม่นยำมากยิ่งขึ้น



ภาพที่ 4 การทดสอบการตรวจจับมือบนเครื่องคอมพิวเตอร์

4. อัปโหลดไฟล์ลงสมาร์ทโฟน

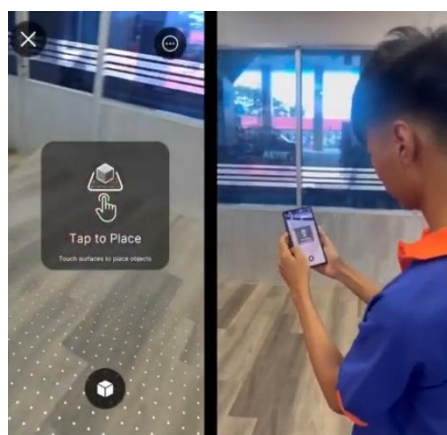
ทำการอัปโหลดแอปพลิเคชันลงบนสมาร์ทโฟนผ่านฟังก์ชัน Build and Run ภายใน Unity เพื่อใช้รวบรวมและแปลงโปรเจกต์ให้กลายเป็นแอปพลิเคชันที่สามารถใช้งานได้จริง โดยระบบจะทำการ Build ซึ่งเป็นขั้นตอนการคอมไพล์โค้ดและรวมไฟล์ทั้งหมดภายในโปรเจกต์ จากนั้นจะทำการ Run เพื่อติดตั้งและเปิดแอปพลิเคชันบนอุปกรณ์สำหรับทดสอบการทำงานโดยอัตโนมัติ



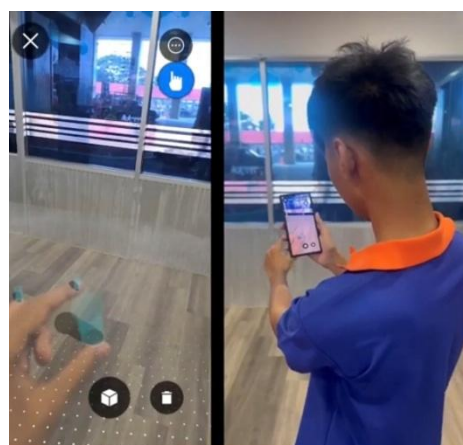
ภาพที่ 5 การอัปโหลดไฟล์ลงสมาร์ทโฟน

5. ทดสอบการตรวจจับ

ทำการทดสอบระบบตรวจจับพื้นผิว (Plane Detection) เพื่อตรวจสอบว่าสามารถใช้งานและแสดงผลในสภาพแวดล้อมจริงได้อย่างถูกต้อง จากนั้นจึงทำการทดสอบระบบตรวจจับมือ เพื่อประเมินว่าสามารถโต้ตอบและควบคุมวัตถุเสมือนจริงภายในแอปพลิเคชันได้ เช่น การหมุนวัตถุ และการย่อ-ขยายวัตถุผ่านการเคลื่อนไหวของมือได้



ภาพที่ 6 แสดงการทดสอบการตรวจจับพื้นผิว



ภาพที่ 7 แสดงการทดสอบระบบตรวจจับมือที่สามารถโต้ตอบกับวัตถุ

6. แก้ไขปัญหาที่เกิดขึ้น

หากเกิดข้อผิดพลาด (Bug) ต่างๆที่แสดงบน Development Console ระหว่างขั้นตอนการ Run ผู้พัฒนาจะทำการตรวจสอบ วิเคราะห์สาเหตุ และแก้ไข

ปัญหาที่เกิดขึ้นจนกว่าระบบจะสามารถทำงานได้อย่างถูกต้องโดยไม่พบข้อผิดพลาด จากนั้นจึงดำเนินการไปยังขั้นตอนถัดไปของการพัฒนาและทดสอบแอปพลิเคชันต่อไป



ภาพที่ 8 การแสดงข้อผิดพลาด ระหว่างการ Run บน Development Console

ข้อผิดพลาดประเภท OpenGL ที่เกิดขึ้นบน Development Console ซึ่งมีสาเหตุมาจากการทำงานของกล้องภายในระบบเกิดการซ้อนกัน ทำให้แอปพลิเคชันไม่สามารถแสดงผลภาพได้อย่างถูกต้อง และไม่มี การแสดงจุดตรวจจับบนมือ (Hand Landmark) ซึ่งเป็นข้อผิดพลาดสำคัญที่ส่งผลต่อการทำงานของระบบตรวจจับมือและการโต้ตอบกับวัตถุเสมือนจริง นอกจากนี้ระหว่างการพัฒนา ผู้พัฒนายังพบข้อผิดพลาดอื่น ๆ อีกหลายส่วน ทั้งด้านการแสดงผล การประมวลผล และความเสถียรของระบบ จึงต้องมีการตรวจสอบ แก้ไข และทดสอบการทำงานอย่างต่อเนื่อง

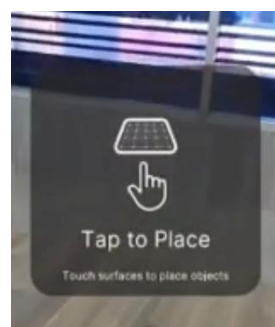
7. สร้างโมเดลสิ่งของ, สร้างระบบเมนูเลือกวัตถุ และ หน้าต่างคำแนะนำการใช้งาน



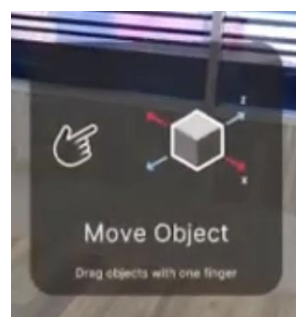
ภาพที่ 9 รูปภาพของโมเดลวัตถุ



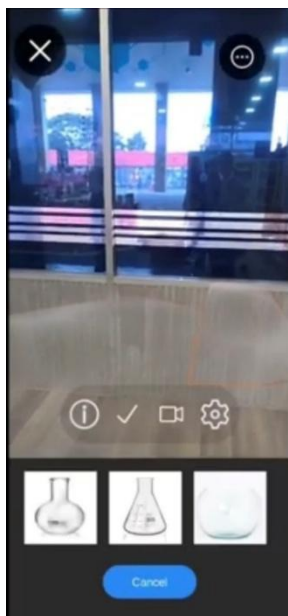
ภาพที่ 10 แสดงโมเดลวัตถุระหว่างการ Run



ภาพที่ 11 แสดงคำแนะนำในการวางวัตถุเสมือนจริงบนพื้นผิวภายในแอปพลิเคชัน

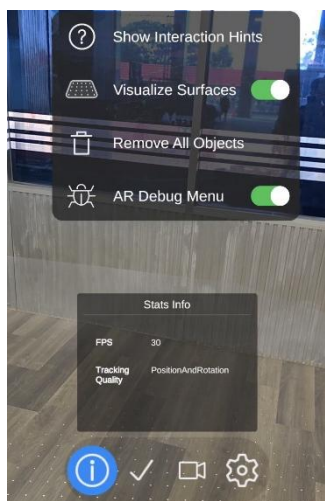


ภาพที่ 12 แสดงคำแนะนำในการเคลื่อนย้ายวัตถุเสมือนจริงบนพื้นผิวภายในแอปพลิเคชัน



ภาพที่ 13 แสดงหน้าต่างเมนูเลือกวัตถุเสมือนจริงบนแอปพลิเคชัน

8. ทดสอบระบบ



ภาพที่ 15 การทดสอบระบบ

ในหน้าต่างระหว่างการ Run จะมีฟังก์ชันชื่อ AR Debug Menu ซึ่งใช้สำหรับตรวจสอบและทดสอบการทำงานของระบบภายในแอปพลิเคชัน โดยสามารถ

แสดงค่า FPS (Frames Per Second) เพื่อตรวจสอบประสิทธิภาพการทำงานของแอปพลิเคชัน ตรวจสอบว่าระบบใดกำลังทำงานอยู่ รวมถึงสามารถเปิดแสดงทิศทางของแกน (Axis) เพื่อใช้ในการทดสอบตำแหน่งการเคลื่อนที่ และความเสถียรของระบบ AR ภายในแอปพลิเคชันได้

สรุปผลและอภิปรายผล

จากการดำเนินงานพัฒนาและทดสอบระบบซอฟต์แวร์ AR-VOI คณะผู้พัฒนาผลการวิจัยได้ดังนี้

ด้านการพัฒนาและคุณลักษณะของซอฟต์แวร์: คณะผู้พัฒนาประสบความสำเร็จในการพัฒนาซอฟต์แวร์ประยุกต์จำลอง 3 มิติเชิงโต้ตอบในรูปแบบแอปพลิเคชันบนระบบปฏิบัติการ Android (ขนาดไฟล์โดยประมาณ 500 เมกะไบต์) ซึ่งสามารถทำงานร่วมกับโครงสร้างพื้นฐาน Google ARCore ได้อย่างสมบูรณ์ด้านประสิทธิภาพของระบบสแกนพื้นที่และปฏิสัมพันธ์: ระบบสามารถประยุกต์ใช้ฟังก์ชันการตรวจจับพื้นผิวระนาบ (Plane Detection) ของเฟรมเวิร์ก Unity AR Foundation เพื่อกำหนดพิกัดในการสร้างและจัดวางวัตถุจำลอง 3 มิติ (Spawn Object) ได้อย่างถูกต้อง และเมื่อผสมผสานการทำงานร่วมกับ MediaPipe Unity Plugin ระบบจะสามารถตรวจจับพิกัดข้อต่อของมือ (Hand Landmarks) ได้แบบเรียลไทม์ผ่านกล้องสมาร์ทโฟน ช่วยให้ผู้ใช้สามารถควบคุมวัตถุเสมือน ทั้งการเคลื่อนย้ายตำแหน่ง (Move) การหมุน (Rotate) และการปรับขนาดวัตถุ (Resize) ด้วยมือเปล่าได้อย่างเป็นธรรมชาติโดยไม่ต้องสัมผัสหน้าจออุปกรณ์

อภิปรายผล

จากผลการวิจัยและทดสอบระบบ มีประเด็นสำคัญที่นำมาอภิปรายผลได้ดังนี้

การบูรณาการซอฟต์แวร์เพื่อทดแทนอุปกรณ์ฮาร์ดแวร์รากฐาน: ผลการทดสอบแสดงให้เห็นว่าการผสมผสานอัลกอริทึมปัญญาประดิษฐ์เพื่อตรวจจับพิกัดมือ (MediaPipe) ร่วมกับระบบความเป็นจริงเสริม (AR Foundation) สามารถสร้างอินเทอร์เฟซแบบธรรมชาติ

(Natural User Interface: NUI) ที่มีประสิทธิภาพสูงได้บนสมาร์ตโฟนส่วนบุคคลทั่วไป จุดนี้พิสูจน์ให้เห็นว่าระบบสามารถตอบโจทย์แนวคิด “ต้นทุนต่ำ ประสิทธิภาพสูง” ช่วยลดข้อจำกัดและงบประมาณในการเข้าถึงเทคโนโลยีระดับสูงสำหรับสถานศึกษาหรือนักวิจัย โดยไม่จำเป็นต้องพึ่งพาอุปกรณ์สวมใส่ราคาแพง เช่น แว่นตา AR/VR หรือถุงมือเซนเซอร์ข้อจำกัดในเวอร์ชันทดสอบและฮาร์ดแวร์ของอุปกรณ์ (System Limitations): เนื่องจากระบบปัจจุบันอยู่ในสถานะเวอร์ชันทดสอบ (Beta Version) ผลการทดลองจึงชี้ให้เห็นว่าความแม่นยำ (System Accuracy) และความเสถียรของการแสดงผลยังคงแปรผันตามสมรรถนะของสมาร์ตโฟนแต่ละรุ่น (Device Limitations) เช่น ความละเอียดของเซนเซอร์กล้องและขีดความสามารถในการประมวลผลกราฟิก ซึ่งส่งผลต่อความต่อเนื่องของเฟรมเรตในอุปกรณ์รุ่นเก่านอกเหนือจากนี้ ระบบยังถูกจำกัดให้รองรับเฉพาะระบบปฏิบัติการ Android เท่านั้นในปัจจุบัน

กิตติกรรมประกาศ

สิ่งประดิษฐ์ เรื่อง การพัฒนาเทคโนโลยี AR สำหรับการจับต้องวัตถุในโลกเสมือนจริงผ่านแอปพลิเคชันบนสมาร์ตโฟน ได้รับการสนับสนุนจากผู้อำนวยการโรงเรียน ตอกเตอร์ทวีศักดิ์ เจริญเตี้ย ที่ได้กรุณาเอื้อเฟื้อสถานที่ในการศึกษาและทดลองรองผู้อำนวยการกลุ่มบริหารงานวิชาการ นายจาตุรนต์ พันธวาส ที่ได้ให้คำแนะนำต่างๆ ตลอดจนได้รับความอนุเคราะห์จากคุณครูกลุ่มสาระการเรียนรู้วิทยาศาสตร์และเทคโนโลยี ครูยุภาพร เปรมกมล ครูที่ปรึกษาโครงการที่ได้ให้คำแนะนำในการทำโครงการตลอดจนเอกสาร และตำราต่างๆ ที่ให้ประโยชน์สำหรับการศึกษาค้นคว้า ขอขอบคุณ ผู้ปกครองที่ให้คำแนะนำ และให้การสนับสนุนในการทำโครงการตั้งแต่เริ่มจนโครงการเสร็จสิ้น ขอขอบคุณ คณะครูกลุ่มสาระการเรียนรู้วิทยาศาสตร์ที่ได้ให้คำแนะนำในการทดลอง และสนับสนุนด้านอุปกรณ์ และเครื่องมือในการทำโครงการจนสำเร็จ

เอกสารอ้างอิง

- Chalmers University of Technology.
 (2016). **AR Hand-Tracking Project**. Chalmers PublicationLibrary.https://publications.lib.chalmers.se/records/fulltext/233159/local_233159.pdf
- Homuler. (2023). **MediaPipe Unity Plugin (Version 0.10.x)** [Computer software]. GitHub.<https://github.com/homuler/MediaPipeUnityPlugin>
- Noobno01. (2023). **AR Hand-Tracking Project (AnatoAR)**: Integrating MediaPipe Hand Tracking with Unity AR Foundation [Computer software]. GitHub.
<https://github.com/Noobno01/ARAnatomy-MediaPipeHand>
- Unity Technologies. (2024). **Unity AR Foundation (Version 6.3)** [Manual documentation]. Unity Documentation.
<https://docs.unity3d.com/Packages/com.unity.xr.arfoundation@6.3/manual/index.html>
- Google LLC. (2024). **MediaPipe Tasks Hand Landmarker**. Google AI Edge.
https://ai.google.dev/edge/mediapipe/solutions/vision/hand_landmarker