

การทำแบบจำลองความลึกท้องน้ำตื้นจากภาพถ่ายดาวเทียมและค่าระดับจากไลเดาร์บนดาวเทียม
กรณีศึกษาพื้นที่ชายฝั่งทะเลประเทศไทย

นายเทพชัย ศรีน้อย

โครงการทางวิศวกรรมนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมสำรวจ ภาควิชาวิศวกรรมสำรวจ
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2564

โครงการทางวิศวกรรม

การทำแบบจำลองความลึกท้องน้ำตื้นจากภาพถ่ายดาวเทียม
และค่าระดับจากไลเดาร์บนดาวเทียม กรณีศึกษาพื้นที่ชายฝั่ง
ทะเล ประเทศไทย

โดย

นายเทพชัย ศรีน้อย

สาขาวิชา

วิศวกรรมสำรวจ

อาจารย์ที่ปรึกษาโครงการทางวิศวกรรม

รองศาสตราจารย์ ดร. ไพบูล สันติธรรมนนท์

ลายเซ็นอาจารย์ที่ปรึกษา

..... อาจารย์ที่ปรึกษาโครงการทางวิศวกรรม
(รองศาสตราจารย์ ดร. ไพบูล สันติธรรมนนท์)

เทพชัย ศรีน้อย : การทำแบบจำลองความลึกท้องน้ำตื้นจากภาพถ่ายดาวเทียมและค่าระดับจากໄลเดอร์บัน
ดาวเทียม กรณีศึกษาพื้นที่ชายฝั่งทะเลประเทศไทย

อาจารย์ที่ปรึกษา : รองศาสตราจารย์ ดร. ไพบูล สันติธรรมนนท์

การศึกษารังนี้เป็นการศึกษาการใช้ประโยชน์จากการถ่ายภาพถ่ายดาวเทียม Sentinel-2 และค่าระดับจากໄලเดอร์บันดาวเทียม ICESat-2 ซึ่งเปิดให้ใช้ได้ฟรีผ่านทางระบบอินเทอร์เน็ต กับการทำแบบจำลองความลึกท้องน้ำตื้น (Bathymetry Model) ภายในพื้นที่ศึกษาชายฝั่งทะเลประเทศไทย 3 พื้นที่อันได้แก่ ชายฝั่งทะเลอ่าวไทย บริเวณหาดเจ้าสำราญ จังหวัดเพชรบุรี ชายฝั่งทะเลอันดามัน บริเวณหาดท้ายเหมือง จังหวัดพังงา และเกาะหมาก จังหวัดตราด โดยศึกษาแนวทางการทำแบบจำลองด้วยการเปรียบเทียบอัลกอริทึมการทำแบบจำลองจากสมการเชิงประจักษ์ (Empirical Formula) ของ Stumpf และ Lyzenga โดยคำนึงถึงการตรวจแก้ภาพอันเนื่องมาจากการอิทธิพลของแสงอาทิตย์ (Sun Glint Correction) และการตรวจแก้อันเนื่องมาจากการอิทธิพลของชั้นบรรยากาศที่มีต่อวัตถุมีดมิด (Dark Object Atmospheric Correction) ด้วยการเขียนโปรแกรม Python ด้วย Google Colab เชื่อมต่อระบบ Google Earth Engine นำเข้าภาพถ่ายดาวเทียมและทำการจัดเตรียมค่าระดับชุดผลิตแบบจำลอง (Seed Depth) เพื่อสร้างแบบจำลองความลึกท้องน้ำตื้น เพื่อทำการประเมินแนวทางความเหมาะสมของการวัดความถูกต้องของแบบจำลองออกมาเป็น Root Mean Square Error (RMSE) เทียบกับค่าระดับชุดทดสอบ (Test Depth) ผลลัพธ์ที่ได้นั้นแบบจำลองจากอัลกอริทึมของ Stumpf แบบที่มีการตรวจแก้ภาพอันเนื่องมาจากการอิทธิพลของชั้นบรรยากาศที่มีต่อวัตถุมีดมิดมีความเหมาะสมสมกับพื้นที่ชายฝั่งอันดามัน จังหวัดพังงา ($RMSE = 1.474$ เมตร) และแบบไม่ตรวจแก้ภาพ สำหรับพื้นที่เกาะหมาก จังหวัดตราด ($RMSE = 2.706$ เมตร) ส่วนอัลกอริทึมของ Lyzenga แบบไม่ผ่านการตรวจแก้ภาพ ($RMSE = 1.186$ เมตร) มีความเหมาะสมสำหรับชายฝั่งทะเลอ่าวไทย จังหวัดเพชรบุรี ซึ่งเป็นบริเวณที่มีความตื้นของชายฝั่งมากที่สุดในบรรดาพื้นที่ศึกษาทั้งสาม

สารบัญ

หน้า

บทคัดย่อ	๑
สารบัญ	๑
สารบัญตรางา	๑
สารบัญรูป	๒
บทที่ 1 บทนำ	๑
1.1 ความเป็นมาและความสำคัญของปัญหา	๑
1.2 วัตถุประสงค์	๓
1.3 ขอบเขตการศึกษา	๓
1.4 พื้นที่ศึกษา	๓
1.5 ข้อมูลที่ใช้ในการศึกษาวิจัย	๓
บทที่ 2 แนวคิดและทฤษฎีที่เกี่ยวข้อง	๔
2.1 ทฤษฎีที่เกี่ยวข้อง	๔
2.2 ทบทวนงานวิจัยที่เกี่ยวข้อง	๑๗
บทที่ 3 วิธีการดำเนินการศึกษา	๑๘
3.1 วิธีการดำเนินงานวิจัย	๑๘
3.2 ประโยชน์ที่คาดว่าจะได้รับ	๓๔
3.3 แผนการดำเนินงาน	๓๔
บทที่ 4 ผลการศึกษา	๓๕
4.1 พื้นที่ศึกษาที่ ๑ ชายฝั่งทะเลอ่าวไทย จังหวัดเพชรบุรี	๓๕
4.2 พื้นที่ศึกษาที่ ๒ ชายฝั่งทะเลอันดามัน จังหวัดพังงา	๔๒
4.3 พื้นที่ศึกษาที่ ๓ เกาะมาก จังหวัดตราราด	๔๙
บทที่ ๕ อภิปราย สรุปผลและข้อเสนอแนะ	๕๔
5.1 อภิปรายผลการศึกษา	๕๔
5.2 สรุปผลการศึกษา	๕๗
5.3 ข้อเสนอแนะ	๕๘
บรรณานุกรม	๕๙

สารบัญตาราง

หน้า

ตารางที่ 1 รูปแบบของคลื่นแม่เหล็กไฟฟ้าและตัวอย่างการใช้ประโยชน์	5
ตารางที่ 2 ตารางแสดงแผนการดำเนินงาน	34
ตารางที่ 3 ข้อมูลพื้นฐานของค่าระดับจากไอลดาร์บริเวณพื้นที่ศึกษาชายฝั่งทะเลอ่าวไทย	35
ตารางที่ 4 ข้อมูลพื้นฐานของภาพถ่ายดาวเทียมสำหรับพื้นที่ศึกษาชายฝั่งทะเลอ่าวไทย	37
ตารางที่ 5 ผลการทดสอบความถูกต้องของแบบจำลอง สำหรับพื้นที่ศึกษาชายฝั่งทะเลอ่าวไทย	37
ตารางที่ 6 ข้อมูลพื้นฐานของค่าระดับจากไอลดาร์บริเวณพื้นที่ศึกษาชายฝั่งทะเลอันดามัน	42
ตารางที่ 7 ข้อมูลพื้นฐานของภาพถ่ายดาวเทียม สำหรับพื้นที่ศึกษาชายฝั่งทะเลอันดามัน	44
ตารางที่ 8 ผลการทดสอบความถูกต้องของแบบจำลอง สำหรับพื้นที่ศึกษาชายฝั่งทะเลอันดามัน	44
ตารางที่ 9 ข้อมูลพื้นฐานของค่าระดับจากไอลดาร์บริเวณพื้นที่ศึกษาเกาะหมาก	49
ตารางที่ 10 ข้อมูลพื้นฐานของภาพถ่ายดาวเทียม สำหรับพื้นที่ศึกษาเกาะหมาก	51
ตารางที่ 11 ผลการทดสอบความถูกต้องของแบบจำลอง สำหรับพื้นที่ศึกษาเกาะหมาก.....	51
ตารางที่ 12 ตารางสรุปการประเมินความถูกต้องของแบบจำลอง	54

สารบัญรูปภาพ

หน้า

ภาพที่ 1 พื้นที่ศึกษาทั้งสามสำหรับการทำวิจัยทำแบบจำลอง SDB	3
ภาพที่ 2 กระบวนการของการรับรู้ระยะใกล้	4
ภาพที่ 3 Spectral Reflectance of Earth Surface	7
ภาพที่ 4 กระบวนการทำแบบจำลอง SDB	9
ภาพที่ 5 The Graph of Sun Glint Correction	11
ภาพที่ 6 ลักษณะแนวทางของเลเซอร์จาก ATLAS	12
ภาพที่ 7 ตัวอย่าง Geolocated Photon ATL03 บริเวณหาดแม่รำพึง ยะອง	13
ภาพที่ 8 Refraction Correction	13
ภาพที่ 9 ตัวอย่างการสืบค้นภาพและแสดงผลภาพถ่ายดาวเทียมผ่าน Google Earth Engine (JavaScript)	16
ภาพที่ 10 ตัวอย่างการสืบค้นภาพและแสดงผลภาพถ่ายดาวเทียมผ่าน Google Colab (Python)	16
ภาพที่ 11 พื้นที่ศึกษาร้อมทั้งแนวของค่าระดับ LiDAR ทั้งสามแนวจากดาวเทียม ICESat-2	18
ภาพที่ 12 ตัวอย่างภาพตัดตามยาวของค่าระดับจาก ICESat-2 ภายในพื้นที่ศึกษาที่สนใจ	20
ภาพที่ 13 ตัวอย่างภาพตัดตามยาวของค่าระดับจาก ICESat-2 เพื่อเลือกค่าระดับผิวน้ำ	21
ภาพที่ 14 ตัวอย่างภาพตัดตามยาวของค่าระดับจาก ICESat-2 เพื่อเลือกค่าระดับห้องน้ำ	22
ภาพที่ 15 ตัวอย่างภาพตัดตามยาวของค่าระดับจาก ICESat-2 ค่าระดับของห้องน้ำเทียบกับค่าระดับทั้งหมด ..	22
ภาพที่ 16 ตัวอย่างค่าระดับ ก่อนการตรวจแก้ (สีน้ำเงิน) และหลังการตรวจแก้ (สีแดง) เทียบกับค่าทั้งหมด (สีดำ)	24
ภาพที่ 17 การแสดงผลภาพถ่ายดาวเทียมจาก Google Earth Engine ผ่าน Folium Map	26
ภาพที่ 18 ตัวอย่างการแสดงผลแบบจำลอง SDB	32

สารบัญรูปภาพ (ต่อ)

หน้า

ภาพที่ 19 แผนผังแสดงขั้นตอนการวิจัย	34
ภาพที่ 20 ค่าระดับของไลเดาร์บริเวณพื้นที่ศึกษาชายฝั่งทะเลอ่าวไทย	35
ภาพที่ 21 แนวของค่าระดับจากไลเดาร์ที่ใช้ทำ Seed Depth (ซ้าย แนวสีส้ม) และทำ Test Depth (ขวา แนวสีเหลือง) ภายในพื้นที่ศึกษาชายฝั่งทะเลอ่าวไทย พื้นหลังเป็นภาพถ่ายดาวเทียม จาก Sentinel2 L2A	36
ภาพที่ 22 กราฟการกระจายของค่าความลึกจาก SDB (bathymetry depth) เทียบกับ ค่าความลึกจากค่าระดับบนดาวเทียม (test depth) เส้นตรงสีแดงแสดงแนวที่ค่าความลึก เท่ากัน ในพื้นที่ศึกษาชายฝั่งทะเลอ่าวไทย	37
ภาพที่ 23 ผลการทำแบบจำลองความลึกท้องน้ำตื้น กรณีไม่ผ่านการตรวจแก้ภาพ บริเวณพื้นที่ศึกษาชายฝั่งทะเลอ่าวไทย เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)	38
ภาพที่ 24 ผลการทำแบบจำลองความลึกท้องน้ำตื้น กรณีตรวจแก้ Sun Glint อย่างเดียว บริเวณพื้นที่ศึกษาชายฝั่งทะเลอ่าวไทย เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)	39
ภาพที่ 25 ผลการทำแบบจำลองความลึกท้องน้ำตื้น กรณีตรวจแก้ DOS อย่างเดียว บริเวณพื้นที่ศึกษาชายฝั่งทะเลอ่าวไทย เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)	40
ภาพที่ 26 ผลการทำแบบจำลองความลึกท้องน้ำตื้น กรณีตรวจแก้ทั้งสองแบบ บริเวณพื้นที่ศึกษาชายฝั่งทะเลอ่าวไทย เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)	41
ภาพที่ 27 ค่าระดับของไลเดาร์บริเวณพื้นที่ศึกษาชายฝั่งทะเลล้อมดามัน	42
ภาพที่ 28 แนวของค่าระดับจากไลเดาร์ที่ใช้ทำ Seed Depth (บน แนวสีส้ม) และทำ Test Depth (ล่าง แนวสีเหลือง) ภายในพื้นที่ศึกษาชายฝั่งทะเลล้อมดามัน พื้นหลังเป็นภาพถ่ายดาวเทียมจาก Sentinel2 L2A	43

สารบัญรูปภาพ (ต่อ)

หน้า

ภาพที่ 29 กราฟการกระจายของค่าความลึกจาก SDB (bathymetry depth) เทียบกับค่าความลึกจากค่าระดับน้ำดาวเทียม (test depth) เส้นตรงสีแดง แสดงแนวที่ค่าความลึกเท่ากัน ในพื้นที่ศึกษาชายฝั่งทะเลอันดามัน	44
ภาพที่ 30 ผลการทำแบบจำลองความลึกท้องน้ำตื้น กรณีไม่ผ่านการตรวจแก้ภาพ บริเวณพื้นที่ศึกษาชายฝั่งทะเลอันดามัน เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)	45
ภาพที่ 31 ผลการทำแบบจำลองความลึกท้องน้ำตื้น กรณีตรวจแก้ Sun Glint อย่างเดียว บริเวณพื้นที่ศึกษาชายฝั่งทะเลอันดามัน เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)	46
ภาพที่ 32 ผลการทำแบบจำลองความลึกท้องน้ำตื้น กรณีตรวจแก้ DOS อย่างเดียว บริเวณพื้นที่ศึกษาชายฝั่งทะเลอันดามัน เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)	47
ภาพที่ 33 ผลการทำแบบจำลองความลึกท้องน้ำตื้น กรณีตรวจแก้ทั้งสองแบบ บริเวณพื้นที่ศึกษาชายฝั่งทะเลอันดามัน เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)	48
ภาพที่ 34 ค่าระดับของไอลดาร์บริเวณพื้นที่ศึกษาเกาะมาก	49
ภาพที่ 35 แนวของค่าระดับจากไอลดาร์ที่ใช้ทำ Seed Depth (บน แนวสีส้ม) และทำ Test Depth (ล่าง แนวสีเหลือง) ภายในพื้นที่ศึกษาเกาะมาก พื้นหลังเป็นภาพถ่ายดาวเทียม Sentinel-2 L2A	50
ภาพที่ 36 กราฟการกระจายของค่าความลึกจาก SDB (bathymetry depth) เทียบกับค่าความลึกจากค่าระดับน้ำดาวเทียม (test depth) เส้นตรงสีแดง แสดงแนวที่ค่าความลึกเท่ากัน ในพื้นที่ศึกษาเกาะมาก	51
ภาพที่ 37 ผลการทำแบบจำลองความลึกท้องน้ำตื้น กรณีไม่ผ่านการตรวจแก้ภาพ บริเวณพื้นที่ศึกษาเกาะมาก เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)	52

สารบัญรูปภาพ (ต่อ)

หน้า

ภาพที่ 38 ผลการทำแบบจำลองความลึกห้องน้ำตื้น กรณีตรวจแก้ Sun Glint อย่างเดียว บริเวณพื้นที่ศึกษาเกาะมาก เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)	52
ภาพที่ 39 ผลการทำแบบจำลองความลึกห้องน้ำตื้น กรณีตรวจแก้ DOS อย่างเดียว บริเวณพื้นที่ศึกษาเกาะมาก เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)	53
ภาพที่ 40 ผลการทำแบบจำลองความลึกห้องน้ำตื้น กรณีตรวจแก้ทั้งสองแบบ บริเวณพื้นที่ศึกษาเกาะมาก เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)	53
ภาพที่ 41 กราฟแท่งเปรียบความถูกต้องของแบบจำลองสองแบบเทียบกับ แนวทางการตรวจแก้ภาพ พื้นที่เพชรบุรี	55
ภาพที่ 42 กราฟแท่งเปรียบความถูกต้องของแบบจำลองสองแบบเทียบกับ แนวทางการตรวจแก้ภาพ พื้นที่พังงา	56
ภาพที่ 43 กราฟแท่งเปรียบความถูกต้องของแบบจำลองสองแบบเทียบกับ แนวทางการตรวจแก้ภาพ พื้นที่ตราด	57
ภาพที่ 44 Automated Shallow Water Bathymetry Mapping Using Google Earth Engine	59

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัจจุบัน

การติดตามการเปลี่ยนแปลงของชายฝั่งทะเล เป็นประเด็นที่มีการศึกษาอย่างแพร่หลายในปัจจุบัน การเปลี่ยนแปลงดังกล่าวมีสาเหตุทั้งจากการเปลี่ยนแปลงภูมิอากาศ การกัดเซาะชายฝั่ง กิจกรรมของมนุษย์ที่เข้ามาเปลี่ยนแปลงการใช้ที่ดินบริเวณชายฝั่ง วิธีการหนึ่งที่ถูกนำมาใช้ในการติดตามการเปลี่ยนแปลงของพื้นที่บริเวณชายฝั่งทะเลคือการทำแบบจำลองความลึกของท้องน้ำตื้น (Bathymetry Model) ซึ่งสามารถทำได้จากการรังวัดในภาคสนาม บริเวณที่เป็นชายฝั่ง จนถึงบริเวณน้ำตื้นมาก สามารถใช้การรังวัดงานระดับด้วยกล้องระดับและไม้อ่านค่าระดับ (Staff) ได้ เมื่อความลึกของท้องน้ำตื้นเพิ่มขึ้น การรังวัดค่าระดับจะเปลี่ยนเป็นการวัดด้วยคลื่นโซนาร์ (SONAR) ที่ติดตั้งใต้ท้องเรือ นำข้อมูลที่ได้ไปจัดทำแบบจำลองความลึกต่อไป วิธีดังกล่าวมีข้อจำกัด คือพื้นที่ชายฝั่งนั้นมีความกว้างขวางเกินกว่าที่วิธีที่ก่อสร้างจะบันทึกได้ครบถ้วนทุกพื้นที่ ส่วนใหญ่แล้วจะเก็บค่าระดับในบางพื้นที่แล้วทำการประมาณค่าระดับภายในพื้นที่สนใจ (Spatial Interpolation) จึงได้แบบจำลองออกมานา แต่ทว่าบริเวณชายฝั่งทะเลนั้นมีความหลากหลายของสิ่งปักคลุมท้องน้ำมากมายเป็นอย่างยิ่ง ทั้งทรัพย์ พืชน้ำ ประการัง ค่าระดับในพื้นที่เล็กๆ มีความหลากหลายเช่นกัน การรังวัดค่าระดับเพื่อทำแบบจำลองที่ละเอียดมากนั้นต้องใช้เวลานานมาก ค่าใช้จ่ายสูง และใช้แรงงานหนัก วิธีดังกล่าวจึงเหมาะสมกับพื้นที่ชายฝั่งที่ไม่กว้างมากนัก

เทคโนโลยีการสำรวจด้วยการรับรู้ระยะไกล (Remote Sensing) ทำให้สามารถบันทึกข้อมูลบนพื้นโลก บริเวณกว้างขวางได้ ด้วยการติดตัวรับรู้ (Sensor) บนดาวเทียม อาศัยหลักของการบันทึกค่าการสะท้อนเชิงคลื่น ของวัตถุ (Spectral Reflectance) แปลงเป็นข้อมูลภาพดิจิทัล (Digital Image) การสำรวจด้วยเทคโนโลยีดังกล่าว ในช่วงคลื่นที่ตามองเห็นและอินฟราเรดใกล้ (Visible and Near Infrared : VNIR) พบว่าข้อมูลที่บันทึกในช่วงคลื่นน้ำเงินและเขียว่าจะสามารถนำมาคำนวณความลึกของท้องน้ำตื้นได้ เกิดเป็นการศึกษาแบบจำลองความลึกของพื้นน้ำตื้นโดยใช้ข้อมูลภาพถ่ายดาวเทียม (Satellite Derived Bathymetry : SDB) ขึ้น โดยแบบจำลองความลึกได้มาจากการคิดค้นและปรับปรุงสมการเชิงประจักษ์ (Empirical Equation) ของการแปลงจากค่าการสะท้อนเป็นค่าความลึก ร่วมกับการรังวัดค่าระดับในภาคสนามจำนวนหนึ่ง (Seed Depth) นำมาสร้างแบบจำลองระดับผ่านสมการดังกล่าวออกมานา ทำให้สามารถได้แบบจำลองในพื้นที่กว้างขวางได้ในเวลาที่รวดเร็วกว่าวิธีก่อน

ทั้งนี้ข้อมูลภาพถ่ายทางดาวเทียมดังกล่าวมีขนาดไฟล์ที่ใหญ่ มีความหลากหลายของคุณภาพข้อมูลที่ได้จากการบันทึกภาพแต่ละครั้งอันเนื่องมาจากสภาพบรรยากาศซึ่งเป็นเรื่องปกติของข้อมูลภาพถ่ายดาวเทียมในช่วงคลื่นตามองเห็น (Optical Remote Sensing) ที่จะมีเมฆ หมอก บดบังพื้นที่ที่สนใจ จึงต้องสืบค้นภาพหลายช่วง

เวลา manipul กันเพื่อให้ได้ภาพที่เหมาะสมที่สุด เหตุผลนี้ทำให้ต้องเสียค่าใช้จ่ายในการจัดเตรียมพื้นที่เก็บข้อมูล ข้อมูลภาพถ่ายทางดาวเทียมในสมัยก่อนมีค่าใช้จ่ายที่แพง ต่อมาระบบที่สามารถเปิดให้สามารถดาวน์โหลดข้อมูลได้โดยไม่เสียค่าใช้จ่าย (Open Remote Sensing) เช่นข้อมูลภาพถ่ายดาวเทียม Landsat-8 ขององค์การ NASA หรือ Sentinel-2 ขององค์การ EU ทำให้การศึกษาทางด้าน Remote Sensing เพิ่มขึ้นเป็นอย่างยิ่ง เนื่องจากการเข้าถึงข้อมูลภาพถ่ายที่ง่ายกว่าเดิม และมีโปรแกรมประมวลผลภาพอุปกรณ์เชิงพาณิชย์อย่าง ENVI หรือเป็นแบบเปิดฟรีไม่เสียค่าใช้จ่ายอย่าง ESA SNAP ประเด็นที่ตามมาคือระบบคอมพิวเตอร์ต้องมีความพร้อมต่อการประมวลผลภาพถ่ายทางดาวเทียมที่มีความละเอียด การทำวิจัยด้านนี้จึงต้องเสียค่าใช้จ่ายมากพอสมควรสำหรับเรื่องนี้ด้วย

ในปัจจุบันนี้ได้มีการเปิดบริการเครื่องมือออนไลน์ Google Earth Engine สำหรับการทำงานศึกษาทางด้าน Remote Sensing ทำได้สะดวกขึ้น เนื่องจากบริการดังกล่าวสามารถสืบค้นภาพถ่ายทางดาวเทียมภายใต้เงื่อนไขคุณสมบัติที่ต้องการ ทำการคำนวณประมวลผลตามความต้องการและแสดงผลลัพธ์ภาพถ่ายได้ครบในเครื่องมือเดียว ด้วยการเขียนโปรแกรมคอมพิวเตอร์ JavaScript หรือ Python ผ่าน Google Colab ช่วยลดประเด็นปัญหาดังที่กล่าวมาก่อนหน้าได้ เนื่องจากการทำงานทุกอย่างที่เกี่ยวข้องทำผ่านระบบอินเทอร์เน็ต

อีกประเด็นหนึ่งคือการรังวัดค่าระดับท้องน้ำตื้น ซึ่งมีความผันแปรตลอดอันเนื่องมาจากการเปลี่ยนแปลงสิ่งปักคลุมดิน อิทธิพลของกระแสน้ำทะเล ปัจจุบันสามารถค้นข้อมูลค่าระดับท้องน้ำตื้นได้จากข้อมูล LiDAR Point Cloud จากตัวรับสัญญาณ ATLAS บนดาวเทียม ICESat-2 ซึ่งเป็นดาวเทียมที่ก่อตั้งหน้าใช้สำราญเกี่ยวกับการสำรวจน้ำแข็งของโลก ต่อมามีการนำมาใช้รังวัดในพื้นที่ชายฝั่งทะเลท้องน้ำตื้นจากความแรงของสัญญาณ LiDAR ที่สามารถทะลุน้ำตื้นลงไปถึงท้องน้ำแล้วสะท้อนกลับมาอย่างตัวรับสัญญาณได้ จากข้อมูลการสะท้อนกลับสามารถคัดเลือกเฉพาะท้องน้ำที่เป็นข้อมูลความลึกท้องน้ำตื้นในบริเวณที่สนใจได้

จากการพัฒนาของเทคโนโลยีทางด้านวิศวกรรมสำรวจทั้ง Google Earth Engine ซึ่งใช้ทำงานกับข้อมูล Open Optical Remote Sensing และ ATLAS LiDAR บนดาวเทียม ICESat-2 ซึ่งเป็นข้อมูลค่าระดับท้องน้ำตื้นทำให้เล็งเห็นถึงหนทางการนำมาใช้ทำแบบจำลอง Satellite Derived Bathymetry ได้ โดยประเทศไทยมีพื้นที่ชายฝั่งทะเลเป็นจำนวนมาก เหมาะสมอย่างยิ่งสำหรับการนำเทคโนโลยีสมัยใหม่มาศึกษาพื้นที่ชายฝั่งภายในประเทศไทย เพื่อนำมาใช้ในการติดตามการเปลี่ยนแปลงของชายฝั่งทะเล การจัดการชายฝั่งต่อไป

โครงการทางวิศวกรรมนี้ จะทำการศึกษาการเขียนโปรแกรมทาง Remote Sensing ผ่าน Google Earth Engine ประกอบกับการคัดเลือกข้อมูลค่าระดับจาก ATLAS LiDAR บนดาวเทียม ICESat-2 เพื่อสร้างแบบจำลอง Satellite Derived Bathymetry สำหรับพื้นที่ชายฝั่งประเทศไทย โดยทดสอบหา Empirical Formula ที่เหมาะสมสำหรับการทำแบบจำลอง จากการประเมินความถูกต้องของแบบจำลองที่ได้

1.2 วัตถุประสงค์การวิจัย

- 1.2.1 ศึกษาการทำแบบจำลองความลึกท้องน้ำด้วยภาพถ่ายทางดาวเทียม (Satellite Derived Bathymetry) สำหรับพื้นที่ศึกษาภายในประเทศไทย จาก Google Earth Engine
- 1.2.2 ศึกษาการใช้ประโยชน์จากข้อมูลค่าระดับ LiDAR จาก ATLAS บนดาวเทียม ICESat-2

1.3 ขอบเขตการศึกษา

ทำการเปรียบเทียบแนวทางการประมาณผลและอัลกอริทึมที่เหมาะสมกับการทำแบบจำลองความลึกท้องน้ำด้วย (Satellite Derived Bathymetry) ด้วย Google Earth Engine ประกอบกับค่าระดับ LiDAR จากดาวเทียม ICESat-2 สำหรับพื้นที่ศึกษาชายฝั่งทะเลภายในประเทศไทย

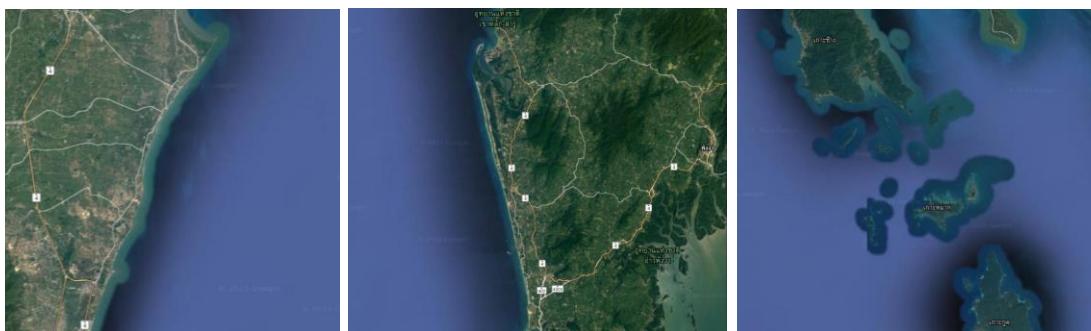
1.4 พื้นที่ศึกษา

พื้นที่ศึกษาทำการวิจัยเป็นพื้นที่ชายฝั่งทะเลภายในประเทศไทย ทั้งทะเลอ่าวไทยและทะเลอันดามัน โดยมีทั้งบริเวณชายหาดและบริเวณหมู่เกาะ จำนวน 3 พื้นที่ศึกษาดังนี้

พื้นที่ศึกษาที่ 1 ชายฝั่งทะเลอ่าวไทย บริเวณหาดเจ้าสำราญ ตอนเหนือของอำเภอชะอำ จังหวัดเพชรบุรี

พื้นที่ศึกษาที่ 2 ชายฝั่งทะเลอันดามัน บริเวณหาดท้ายเหมือง จังหวัดพังงา

พื้นที่ศึกษาที่ 3 เกาะหมาก (ระหว่างเกาะช้างกับเกาะกูด) จังหวัดตราราด



ภาพที่ 1 พื้นที่ศึกษาทั้งสามสำหรับการทำวิจัยทำแบบจำลอง SDB

จาก Google Earth

1.5 ข้อมูลที่ใช้ในการศึกษาวิจัย

- 1.5.1 ข้อมูลภาพถ่ายดาวเทียม Sentinel 2 แบบ Bottom of Atmosphere (BOA) ซึ่งบันทึกภาพในช่วงปี พ.ศ. 2563 – 2565 (เดือนมกราคม 2565) ประกอบกับคันจากระบบ Google Earth Engine
- 1.5.2 ข้อมูลค่าระดับจาก ATLAS LiDAR บนดาวเทียม ICESAT-2 ซึ่งผ่านพื้นที่ศึกษา ภายในช่วงปี พ.ศ. 2563 – 2564 ดาวน์โหลดข้อมูลได้จาก openaltimetry.org ในรูปแบบ hdf5 file

บทที่ 2

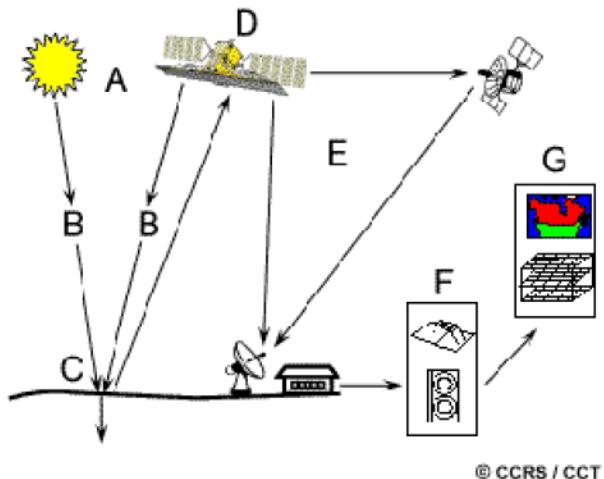
แนวคิดและทฤษฎีที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้องกับงานวิจัย

2.1.1 ข้อมูลภาพถ่ายดาวเทียม (Satellite Image)

ข้อมูลภาพถ่ายดาวเทียม เป็นผลผลิตที่ได้จากเทคโนโลยีการสำรวจที่เรียกว่า การรับรู้ระยะไกล หรือ โทรสัมผัส (Remote Sensing) เป็นระบบสำรวจเก็บข้อมูลเกี่ยวกับ พื้นที่ วัตถุ และปรากฏการณ์ต่างๆบนพื้นผิวโลก โดยที่ไม่มีการสัมผัสถูกกับเป้าหมายโดยตรง แต่จะอาศัยเครื่องรับรู้ (Sensor) ซึ่งติดไปกับยานอวกาศ ดาวเทียมหรือ เครื่องบิน ในการบันทึกพลังงานคลื่นแม่เหล็กไฟฟ้าที่สะท้อนหรือเปล่งออกมายจากแหล่งพลังงาน (เช่น ดวงอาทิตย์ และวัตถุบนพื้นผิวโลก) ซึ่งจะได้ข้อมูลที่มีคุณลักษณะ 3 ประการคือ ลักษณะแรก เป็นลักษณะการสะท้อนช่วงคลื่น แม่เหล็กไฟฟ้า (Spectral Characteristics) ลักษณะที่สอง เป็นลักษณะเชิงพื้นที่ของวัตถุบนพื้นผิวโลก (Spatial Characteristics) และลักษณะที่สาม เป็นลักษณะการเปลี่ยนแปลงของวัตถุตามช่วงเวลา (Temporal Characteristics) หลังจากนั้นมีการแปลงเป็นข้อมูลเชิงเลขซึ่งนำไปใช้แสดงเป็นภาพและทำแผนที่

การรับรู้ระยะไกลมี 3 องค์ประกอบหลัก คือ การได้มาซึ่งข้อมูล (Data Acquisition) การวิเคราะห์ข้อมูล (Data Analysis) และ การนำข้อมูลไปประยุกต์ (Data Applications) กระบวนการแสดงดังภาพที่ 2



ภาพที่ 2 กระบวนการของการรับรู้ระยะไกล

จาก Fundamentals of Remote Sensing (A Canada Centre for Remote Sensing)

จากภาพที่ 2 สามารถอธิบายกระบวนการของการรับรู้ระยะไกลได้ดังนี้

A. แหล่งพลังงานหรือการส่องสว่าง (Energy Source or Illumination) การรับรู้ระยะไกลใช้คลื่นแม่เหล็กไฟฟ้า ซึ่งมีทั้งเกิดจากแหล่งกำเนิดพลังงานตามธรรมชาติอย่างดวงอาทิตย์ หรือมนุษย์สร้างขึ้น การรับรู้ระยะไกลสามารถจำแนกตามแหล่งกำเนิดคลื่นแม่เหล็กไฟฟ้าได้ดังนี้

1. การรับรู้จากการรับรู้ระยะไกลแบบพาสซีฟ (Passive Remote Sensing)

เป็นการรับรู้ระยะไกลที่ใช้ดวงอาทิตย์เป็นแหล่งพลังงานที่มีการสะท้อนและการปล่อยรังสีผ่านชั้นบรรยากาศปัจจุบันคลื่นผิวโลกและสะท้อนกลับไปยังเครื่องรับรู้ สามารถใช้ได้ในช่วงเวลากลางวันเท่านั้น ไม่สามารถใช้ได้ในเวลากลางคืนหรือช่วงเวลาที่สภาพอากาศไม่เหมาะสม เช่นมีเมฆ หมอก ฝน ปักคุณ

2. การรับรู้ระยะไกลแบบแอคทีฟ (Active Remote Sensing)

เป็นการรับรู้ระยะไกลที่เครื่องรับรู้มีอุปกรณ์สร้างแหล่งกำเนิดพลังงานขึ้นเอง เป็นคลื่นวิทยุหรือคลื่นแสงเลเซอร์ที่ส่งพลังงานไปยังวัตถุเป้าหมาย และมีการกระจายคลื่นมา.yang เครื่องรับรู้ สามารถใช้ได้ทุกสภาพอากาศ รับส่งสัญญาณได้ทั้งเวลากลางวันและกลางคืน ตลอดจนสามารถลงสู่พื้นดินได้ลึกประมาณ 3 – 5 เมตร เช่น RADAR บันทึกข้อมูลในช่วงคลื่นไมโครเวฟ หรือ LiDAR ใช้แสงเลเซอร์ระดับปานกลางในช่วงคลื่นสั้นกว่าในช่วงตามองเห็นและอินฟราเรดใกล้

B. การแผ่รังสีผ่านชั้นบรรยากาศ (Radiation and the Atmosphere) พลังงานในรูปคลื่นแม่เหล็กไฟฟ้าเคลื่อนผ่านชั้นบรรยากาศระหว่างพื้นโลกแล้วสะท้อนกลับไปยังเครื่องรับรู้บนดาวเทียมหรือเครื่องบิน จากการศึกษาพบว่ามีคลื่นแม่เหล็กไฟฟ้าบางช่วงคลื่นที่ถูกดูดกลืนหมดที่ชั้นบรรยากาศ บางส่วนเคลื่อนลงมาถึงพื้นโลกได้ และแต่ละช่วงคลื่นมีการนำไปใช้ประโยชน์ได้หลากหลาย ดังตารางที่ 1

ช่วงคลื่น	ความยาวคลื่น	รายละเอียด/ ตัวอย่างการใช้ประโยชน์
รังสีแกมมา (Gamma Ray)	น้อยกว่า 0.03 ไมโครเมตร	ถูกดูดกลืนหมดที่ชั้นบรรยากาศ
รังสีเอกซ์ (X Ray)	0.03 – 3.0 ไมโครเมตร	ถูกดูดกลืนหมดที่ชั้นบรรยากาศ
รังสีหนึ่อม่วง (Ultraviolet)	0.03 – 0.4 ไมโครเมตร	ช่วงคลื่นสั้นกว่า 0.3 ไมโครเมตร ถูกดูดกลืนหมดโดยโอโซนในชั้นบรรยากาศ (สตราโตสเฟียร์)
แบนด์หนึ่อม่วงภาพถ่าย (Photographic Ultraviolet Band)	0.3 – 0.4 ไมโครเมตร	ช่วงคลื่นที่ผ่านชั้นบรรยากาศ สามารถถ่ายภาพด้วยฟิล์มถ่ายรูป แต่กระจายในชั้นบรรยากาศ
ช่วงคลื่นตามองเห็น (Visible Range)	0.4 – 0.7 ไมโครเมตร	บันทึกด้วยฟิล์มและอุปกรณ์บันทึกภาพได้ดี ช่วงคลื่นแคบที่มีผลตอบสนองต่อสายตามนุษย์มี 3 ช่วง คือ ช่วงสีน้ำเงิน ช่วงสีเขียว และช่วงสีแดง

ช่วงคลื่น	ความยาวคลื่น	รายละเอียด/ ตัวอย่างการใช้ประโยชน์
คลื่นใกล้赤外线 (Near Infrared)	0.7 – 1.3 ไมโครเมตร	ใช้ศึกษาพืชพรรณ, แยกดินและน้ำ
คลื่นใกล้赤外线สั้น (Short wave infrared)	1.3 – 3.0 ไมโครเมตร	ใช้ศึกษาการใช้ที่ดินและแร่
คลื่นใกล้赤外线กลาง (Middle Infrared)	3.0 – 8.0 ไมโครเมตร	ใช้แยกแร่และวัสดุสหัสห้องแสง
คลื่นใกล้赤外线ความร้อน (Thermal Infrared)	8.0 – 14.0 ไมโครเมตร	ใช้ศึกษาโรคพืช
ไมโครเวฟ (Microwave)	0.1 – 30.0 เซนติเมตร	หลุมมอง เมฆ ฝนได้ บันทึกภาพได้ทั่วระบบ พาสซีฟและแอคทีฟ
วิทยุ (Radio wave)	มากกว่า 30 เซนติเมตร	บางครั้งเรารู้อยู่ในช่วงนี้ด้วย

ตารางที่ 1 รูปแบบของคลื่นแม่เหล็กไฟฟ้าและตัวอย่างการใช้ประโยชน์

C. ปฏิสัมพันธ์กับเป้าหมาย (Interaction with the target) คลื่นแม่เหล็กไฟฟ้าจะมีปฏิสัมพันธ์กับชั้นบรรยากาศ โลกและพื้นผิวโลก อนุภาคของอากาศดูดกลืนคลื่นแม่เหล็กไฟฟ้าบางช่วง ทำให้คลื่นไม่สามารถส่งผ่านมาถึงพื้นผิวได้ เมื่อถึงพื้นผิวโลกแล้ว คลื่นบางส่วนจะถูกดูดกลืนโดยพื้นผิวโลกซึ่งมีความแตกต่างกัน การสะท้อนกลับจากพื้นผิวต่างกันจึงได้ข้อมูลที่แตกต่างกัน เป็นประโยชน์ต่อการแปลสภาพและการจำแนกประเภทของวัตถุ

พลังงานคลื่นแม่เหล็กไฟฟ้า ณ ช่วงคลื่นหนึ่งที่ตกกระทบวัตถุใดๆ จะมีค่าเท่ากับผลรวมพลังงานที่สะท้อน (Reflection) พลังงานที่ถูกดูดกลืน (Absorption) และพลังงานที่ส่งผ่าน (Transmission) พลังงานสะท้อนกลับมีความสำคัญอย่างมาก อัตราส่วนพลังงานที่สะท้อนจากพื้นผิววัตถุต่อผลกระทบจากพื้นผิววัตถุเรียกว่า ค่าการสะท้อนเชิงสเปกตรัม (Spectral Reflectance) เป็นสมบัติที่สำคัญในการตีความ แยกและวัดวัตถุต่างๆ ออกจากกัน ตัวอย่างรายละเอียดเกี่ยวกับการสะท้อนของวัตถุบนโลก

ก. พืชพรรณ (Vegetation)

ใบพืชจะมีคลอรอฟิลล์ (Chlorophyll) สำหรับช่วงคลื่นที่ตามองเห็น คลอรอฟิลล์ดูดกลืนคลื่น ในช่วงสีน้ำเงินและสีแดง แต่จะสะท้อนคลื่นในช่วงสีเขียวออกมานอกจากนี้ ใบพืชเป็นสีเขียวหากใบพืชมีอาการผิดปกติ เช่น ใบแห้ง ใบเหลือง ทำให้การสะท้อนคลื่นสีแดงสูงขึ้นในช่วงอินฟราเรด สะท้อน ปัจจัยที่มีผลต่อลักษณะการสะท้อน ได้แก่ ลักษณะใบ ลักษณะของเรือนยอด ปริมาณน้ำในใบพืช อายุของพืช สีของใบและโรคภัย

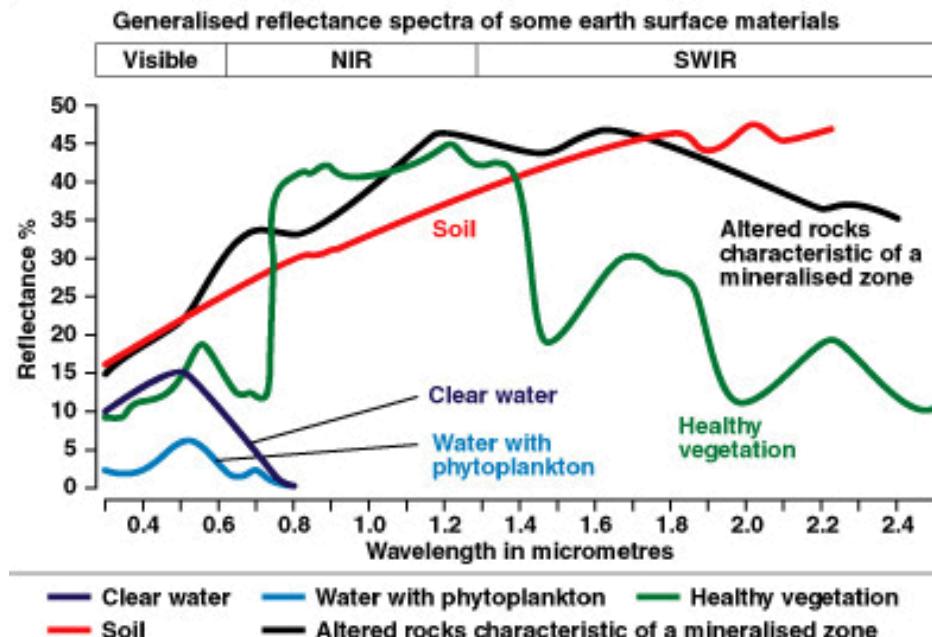
ข. ดิน (Soil)

องค์ประกอบของดินได้แก่ ออกริเจน ชิลิกอน อะลูมิเนียม เหล็ก แมกนีเซียม แคลเซียม โซเดียม โพแทสเซียม และธาตุอื่นๆ ปฏิสัมพันธ์ของดินจะมีค่าการสะท้อนสูงกว่าน้ำและพืช ลักษณะบ่งชี้ว่า สเปกตรัมจะมีลักษณะขึ้นๆลงๆ แต่มีความแตกต่างไม่มากเมื่อเทียบกับพืช ปัจจัยที่มีผลต่อลักษณะ การสะท้อนของดินได้แก่ ความชื้นในดิน องค์ประกอบและความชุ�性ของเนื้อดิน ปริมาณสนิมเหล็กและสารอินทรีย์ สีของดิน และปริมาณพืชปกคลุมดิน

ค. น้ำ (Water)

น้ำมีค่าการสะท้อนต่ำกว่าในทุกช่วงคลื่น เพราะน้ำเป็นของเหลวและสามารถดูดกลืนพลังงาน เอ้าไว้ได้สูง น้ำมีค่าการสะท้อนค่าสูงสุดในกลุ่มคลื่นที่ 1.4 ไมครอนที่มีขนาดน้อยกว่า 0.6 ไมครอน และถูกดูดกลืนมากขึ้นเรื่อยๆ ในช่วงคลื่นอินฟราเรดใกล้ สั้น และกลาง และช่วงไมโครเรฟ ปัจจัยที่มีผลต่อการสะท้อนของน้ำคือ ความลึกและสิ่งแขวนลอยกรณีของน้ำ

ลักษณะการสะท้อนเชิงคลื่นของวัตถุ แสดงได้ดังภาพที่ 3



ภาพที่ 3 Spectral Reflectance of Earth Surface

จาก <https://www.usna.edu/>

D. การบันทึกพลังงานโดยเครื่องรับรู้ (Recording of Energy by the sensor)

E. การส่งผ่าน การรับรู้และกระบวนการ (Transmission, Reception and Processing) ข้อมูลที่ได้จากเครื่องรับรู้จะถูกส่งผ่านในรูปแบบอิเล็กทรอนิกส์ ไปยังสถานีรับภาคพื้นดินและแปลงสัญญาณนั้นเป็นข้อมูลเชิงเลข

(Digital Data) หรือข้อมูลภาพ (Image Data) โดยมีการตรวจแก้เบื้องต้นในเชิงเรขาคณิต (Geometric Correction) เชิงรังสี (Radiometric Correction) และความบกพร่องของสัญญาณที่ได้รับ

(1) กระบวนการก่อนการประมวลผลภาพ (Pre-processing) เป็นการตรวจแก้ความคลาดเคลื่อนของข้อมูล สัญญาณรบกวน และความบิดเบี้ยงเชิงเรขาคณิตที่เกิดขึ้นระหว่างกระบวนการถ่ายภาพ การบันทึกข้อมูลสัญญาณ การสะท้อนคลื่นแม่เหล็กไฟฟ้า การส่งสัญญาณ และการโคจรของดาวเทียม มีกระบวนการที่เกี่ยวข้อง 2 กระบวนการ

กระบวนการที่ 1 : การตรวจแก้คลื่นรังสี (Radiometric Correction) ทั้งการขาดเฉยค่าการสะท้อนที่บิดเบือนของสภาพอากาศ (Haze Compensation) การเปลี่ยนค่าความสว่างเป็นค่าการแปรรังสีสมบูรณ์ (Conversion of digital numbers to absolute radiance value) และการลบสัญญาณรบกวน (Noise Removal)

กระบวนการที่ 2 : การตรวจแก้เชิงเรขาคณิต (Geometric Correction) เป็นผลมาจากการผิดพลาดทางตำแหน่ง ก่อให้เกิดการบิดเบี้ยวยางเรขาคณิต มีทั้งภายใน ซึ่งมาจากความบกพร่องของอุปกรณ์ในเครื่องรับสัญญาณ และภายนอกมาจากการทรงตัวของเครื่องวัด ความโถ้งของโลก การเคลื่อนที่ของโลก การหมุนของโลก บรรยากาศนานที่พาอุปกรณ์ไป วงโคจรของยาน ความสูงพื้นผิว รูปร่างวัตถุ เป็นต้น การตรวจแก้เป็นการสร้างความสมมัติของระบบพิกัดระหว่างข้อมูลที่ตรวจแก้กับระบบพิกัดภูมิศาสตร์ของข้อมูลอ้างอิง เพื่อที่พิกัดของข้อมูลที่ต้องการตรวจแก้ ถูกเปลี่ยนให้เป็นระบบพิกัดใหม่ตามระบบพิกัดข้อมูลอ้างอิง โดยต้องมีจุดควบคุมภาคพื้นดิน ซึ่งความมีมากพอและกระจายอย่างกว้างขวางสำหรับที่ศึกษา

(2) การเน้นข้อมูลภาพ (Image Enhancement) เป็นกระบวนการปรับเปลี่ยนค่าของจุดภาพหรือค่าระดับสีเทา เพื่อเพิ่มรายละเอียด เพิ่มความแตกต่างระหว่างวัตถุ เน้นความคมชัดเฉพาะส่วนที่ต้องการศึกษาเพื่อความง่ายในการตีความ เช่น การทำให้เห็นความต่าง (Contrast Adjustment) อย่าง Greyscale Thresholding, Level Slicing และ Contrast Stretching

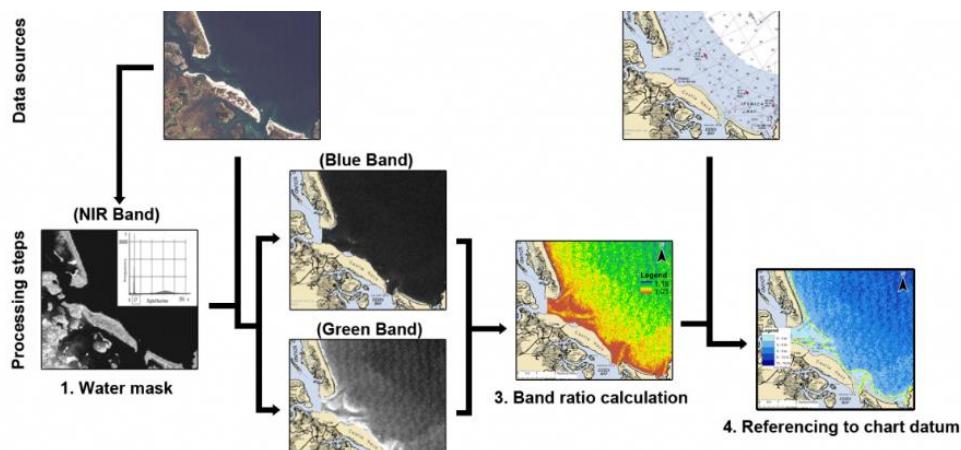
(3) การจำแนกประเภทข้อมูลภาพ (Image Classification) เป็นการประมวลผลทางสถิติเพื่อแยกข้อมูลจุดภาพทั้งหมดที่ประกอบเป็นพื้นที่ศึกษาออกเป็นกลุ่มย่อย โดยใช้ลักษณะทางสถิติเป็นตัวกำหนดความแตกต่างระหว่างกลุ่มจุดภาพ เป็นการแบ่งจุดภาพที่มีคุณสมบัติการสะท้อนแสง หรือ ลายเส้นเชิงคลื่น (Spectral Reflectance) คล้ายกันออกเป็นกลุ่มหรือระดับ

F. การแปลผลและการวิเคราะห์ (Interpretation and Analysis) เช่น การนำภาพถ่ายทางดาวเทียมไปใช้ศึกษา จำแนกการใช้ประโยชน์ที่ดินและสิ่งปลูกสร้าง (Land use/Landcover Classification) การติดตามการเปลี่ยนแปลงของวัตถุที่อยู่บนพื้นโลก การสร้างแบบจำลองอธิบายปรากฏการณ์บนพื้นโลก

G. การประยุกต์ (Application) ทั้งทางด้านปัจมี การเกษตร ผังเมือง ธรณีวิทยา อุทกวิทยาและการจัดการทรัพยากรน้ำ สมุทรศาสตร์และการจัดการชายฝั่ง ภัยธรรมชาติ สิ่งแวดล้อม

2.1.2 แบบจำลองความลึกท้องน้ำตื้นจากภาพถ่ายดาวเทียม (Satellite Derived Bathymetry : SDB)

แบบจำลองความลึกท้องน้ำตื้นจากภาพถ่ายดาวเทียม (Satellite Derived Bathymetry : SDB) มีพื้นฐานมาจากความเข้าใจโดยทั่วไปว่า สำหรับบริเวณชายฝั่งทะเลท้องน้ำตื้นที่น้ำใส่มาก ท้องน้ำเรียบร้าบแล้ว (Horikawa, 1988) ความเข้มของสีน้ำทะเลบ่งบอกความลึกของน้ำทะเล หมายความว่าสีน้ำทะเลเข้มมาก แสดงว่าบริเวณนั้นน้ำทะเลมีความลึกมาก สามารถเห็นได้จากการวิเคราะห์จากภาพถ่ายในช่วงคลื่นสีน้ำเงินและสีเขียว (Sabins JR, 1978) ตัวอย่างกระบวนการทำแบบจำลอง SDB แสดงดังภาพที่ 4



ภาพที่ 4 กระบวนการทำแบบจำลอง SDB

จาก Hydro International By Shachak Pe'eri, Chukwuma Azuike, Christopher Parrish

ความสัมพันธ์ระหว่างค่าการสะท้อนของภาพถ่ายดาวเทียมกับความลึกของท้องน้ำ ในงานวิจัยส่วนใหญ่ยังคงเป็นแบบสมการเชิงประจักษ์ (Empirical Equation) ทั้งนี้รูปแบบสมการแบบอิงหลักการทางฟิสิกส์ (Physics-based model equation) ได้มีการศึกษาเรื่อยๆ ตลอดมา งานวิจัยนี้สนใจสมการเชิงประจักษ์ 2 สมการเพื่อนำมาทำแบบจำลอง SDB (Thomas et al., 2021) และประเมินความเหมาะสมต่อพื้นที่ศึกษาในประเทศไทย

กำหนดให้ ค่าการสะท้อนซึ่งบอกเป็นเลขดิจิตอลในแต่ละช่องพิกเซลของภาพถ่ายทางดาวเทียมภาพหนึ่ง มีค่าการสะท้อนในช่วงคลื่นสีน้ำเงินเป็น L_B และมีค่าในช่วงคลื่นสีเขียวเป็น L_G บริเวณนั้นมีค่าความลึกเป็น H สมการแสดงความสัมพันธ์ของค่าการสะท้อนกับค่าความลึกเชิงประจักษ์เป็นดังนี้ (Geyman and Maloof, 2019)

A. Stumpf Algorithm (Stumpf et al., 2003)

$$H = m_{1S} \frac{\ln(1000L_B)}{\ln(1000L_G)} + m_{0S}$$

เมื่อ m_{1S}, m_{0S} เป็นสัมประสิทธิ์ของสมการ

B. Lyzenga Algorithm (Lyzenga, 1978) สำหรับงานวิจัยนี้ได้ใช้สมการในรูปแบบอย่างง่าย ตามงานของ Geyman and Maloof, 2019

$$H = m_{2L} \ln(L_B) + m_{1L} \ln(L_G) + m_{0L}$$

เมื่อ m_{2L}, m_{1L}, m_{0L} เป็นสัมประสิทธิ์ของสมการ

สัมประสิทธิ์ของสมการมาจากการคำนวณจากการตรวจแก้กำลังสองน้อยสุด (Least Square Adjustment) สมการค่ารังวัด (Observation Equations) ของระบบสมการเชิงเส้น จากข้อมูลความลึกเริ่มต้นเพื่อทำแบบจำลอง (Seed depth ในงานวิจัยนี้บางครั้งเรียกว่า Training Data/Depth) กับ ค่าการสะท้อนในแบบดีไซน์และน้ำเงิน ผ่านแบบจำลองอัลกอริทึมดังกล่าว ในงานวิจัยนี้ได้เลือกทำการคำนวณตรวจแก้ด้วยการเขียนโปรแกรม python จากบริการ sklearn regression

ภาพถ่ายทางดาวเทียมที่นำมาทำแบบจำลอง อย่างเช่นภาพถ่าย Multispectral Passive Remote Sensing จากดาวเทียม Sentinel 2 ทั้งรูปแบบ Top of Atmosphere (L1C) หรือ Bottom of Atmosphere (L2A) (Babbel et al., 2021) ควรมีกระบวนการตรวจสอบแก้ภาพถ่ายดาวเทียม ดังนี้ (Serco Italia SPA, 2021)

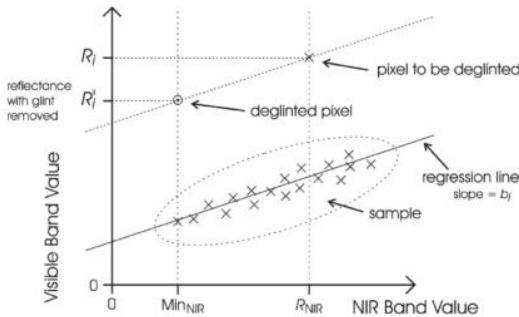
กระบวนการที่ 1 การแยกทำเฉพาะส่วนของน้ำด้วยการทำ Water Detection หรือ Land mask วิธีการทำเบื้องต้นคือใช้ข้อมูลจากแบบดีที่ 8 เพื่อแยกเฉพาะส่วนที่คาดว่าเป็นน้ำ จากความรู้ที่ว่าแบบดีที่ 8 ซึ่งอยู่ในช่วงอินฟราเรดใกล้ น้ำมีการสะท้อนต่ำมาก ใช้เป็นขีดจำกัด (Threshold) และส่วนที่เป็นน้ำกับไม่ใช่น้ำออกมานะ

- ค่าในแบบดีของ Land mask เป็น 1 เมื่อค่าในแบบดีเดิมน้อยกว่าขีดจำกัดที่กำหนด
- ไม่มีค่าในแบบดีของ Land mask เมื่อค่าในแบบดีเดิมมากกว่าขีดจำกัดที่กำหนด

กระบวนการที่ 2 การตรวจแก้ผลกระทบจากการสะท้อนของดวงอาทิตย์ผ่านพิวน้ำของพื้นผิวทะเล (Sun Glint Correction) ด้วยวิธี deglint methodology ของ Hedley (Hedley et al, 2005) ซึ่งบรรยายสมการเชิงเส้นแสดงความสัมพันธ์ของค่าในแบบดีอินฟราเรดใกล้กับแบบดีช่วงตามองเห็น ในพิกเซลตัวอย่าง การตรวจแก้ทำได้จากการนำพิกเซลตัวอย่างจำนวนหนึ่งในพื้นที่งานมาค้นค่าการสะท้อนในแบบดีช่วงคลื่นตามองเห็นกับอินฟราเรดใกล้มาเขียนกราฟการกระจาย และทำ Linear Regression หากความชัน นำภาพเก่ามาปรับด้วยสมการ

$$R'_i = R_i - b_i (R_{NIR} - Min_{NIR})$$

เมื่อ R' , เป็นค่าการสะท้อนที่ตรวจแก้แล้ว (deglint pixel) ในแบบที่ i , R_i , เป็นค่าการสะท้อนในแบบที่ i , b_i เป็นความชันจากการ regression, R_{NIR} เป็นค่าการสะท้อนในแบบที่อินฟราเรดใกล้ และ Min_{NIR} เป็นค่าการสะท้อนในแบบที่อินฟราเรดใกล้ที่ต่ำสุดในพิกเซลตัวอย่าง



ภาพที่ 5 The Graph of Sun Glint Correction

จาก RUS Lectures Serco Italia SPA, 2021

กระบวนการที่ 3 การตรวจแก้ก้อนเนื้องมาจากการผลกระทบทางชั้นบรรยากาศสำหรับวัตถุมีดมิด (Dark-Object Atmospheric Correction) ยึดหลักที่ว่าการสะท้อนของน้ำที่ลึกมากๆ ในช่วงคลื่นที่ตามองเห็น คาดหวังว่าจะเป็น 0 วิธีการเบื้องต้นในการตรวจแก้ (Chavez, 1998) คือการค้นค่าการสะท้อนในช่วงคลื่นตามมองเห็นจากพิกเซลตัวอย่างแล้วหาค่าการสะท้อนต่ำสุด $Min(R)_{sample}$ นำค่านั้นลบค่าการสะท้อน R ในภาพ

$$R_{DOS} = R - Min(R)_{sample}$$

2.1.3 ข้อมูลค่าระดับ LiDAR จากดาวเทียม ICESat-2

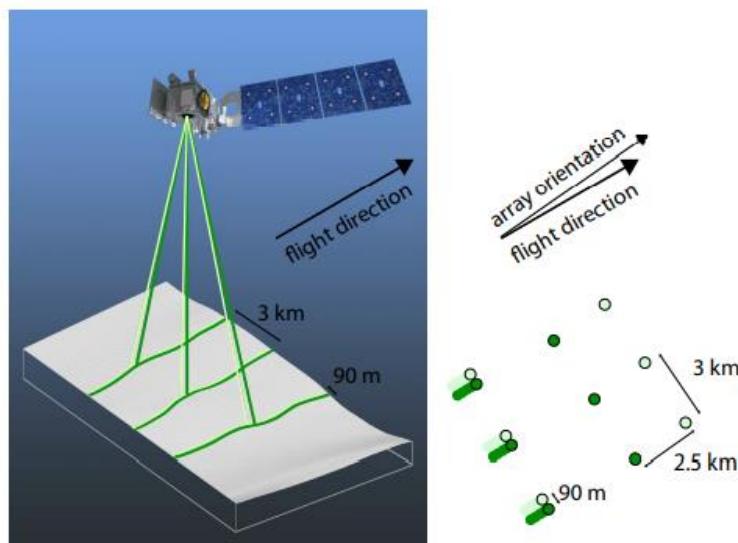
Light Detection and Ranging (LiDAR) เป็น Active Remote Sensing ประเภทหนึ่ง ซึ่งสามารถให้ค่าพิกัดของ Point Cloud จำนวนมากได้ จากการติดตั้งเครื่องรับสัญญาณพาหะ ระบบจะสร้างพลังงานคลื่นแม่เหล็กไฟฟ้ามาในรูปแบบแสงเลเซอร์ ออกไปกระทบเป้าหมายแล้วสะท้อนกลับเข้าเครื่องรับสัญญาณ เป้าเป็นเวลาที่ใช้เคลื่อนที่เปล่งเป็นระยะทางจากตัวรับสัญญาณ ทราบค่าพิกัดของเครื่องรับสัญญาณจาก Global Navigation Satellite System (GNSS) ทราบการวางตัวของเครื่องรับสัญญาณจาก Inertial Measurement Unit (IMU) และ เวกเตอร์สามมิติจากตัวรับสัญญาณ 陂จังเป้าหมายทำให้ทราบค่าพิกัดของเป้าหมายได้

ดาวเทียม ICESat-2 ย่อมาจาก Ice, Cloud and land Elevation Satellite เป็นผลงานการพัฒนาของทีมงานองค์กรอวกาศ NASA ขึ้นสู่空域เมื่อวันที่ 15 กันยายน พ.ศ. 2561 ดาวเทียมนี้มีการติดตั้งเครื่องรับสัญญาณ

ATLAS ย่อมาจาก Advance Topographic Laser Altimeter System เป็นระบบ photon-counting LiDAR ประกอบกับระบบเสริมอื่นๆ (GNSS และ star camera) เพื่อให้ได้การรังวัดสามมิติ

1. เวลาการเคลื่อนที่ของโฟตอนจาก ATLAS ถึงพื้นโลกแล้วสะท้อนกลับเข้าเครื่องรับรู้ดังกล่าว
2. เวลาตอบรับของตำแหน่งที่โฟตอนปล่อยออกมายก ATLAS
3. ตำแหน่งของดาวเทียม ณ เวลาที่โฟตอนถูกบันทึกโดย ATLAS

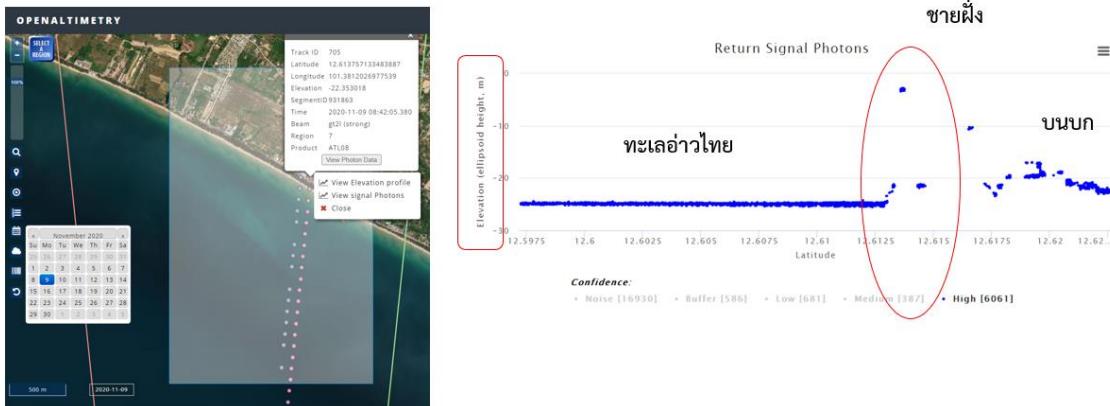
ระบบเลเซอร์ที่ปล่อยออกมายก ATLAS เป็นแสงสีเขียว 532 nm ที่ปล่อยด้วยความถี่ 10 kHz โดยระบบนี้ปล่อยลำแสงเลเซอร์ออกมาราว 6 ลำแสง โดยแบ่งเป็น 3 คู่ แยกไปตามแนวทาง 3 แนว แต่ละแนวห่างกัน 3.3 กิโลเมตร ในแนววางการเคลื่อนที่ของดาวเทียม แต่ละคู่จะมีความแตกต่างของพลังงานที่ปล่อยออกมายก โดยมี weak กับ strong อัตราส่วนพลังงาน 1 : 4 โดยคู่ห่างกัน 90 เมตร ในแนววางการเคลื่อนที่ของดาวเทียม และ 2.5 กิโลเมตร ในแนวเดียวกับการเคลื่อนที่ของดาวเทียม โดยแนวทางซ้ายเรียกว่า GT1L (weak) GT1R (strong) กลาง GT2L (weak) GT2R (strong) ขวา GT3L (weak) GT3R (strong) ภาพประกอบเป็นไปดังภาพที่ 6



ภาพที่ 6 ลักษณะแนวทางของเลเซอร์จาก ATLAS

จาก Algorithm Theoretical Basis Document for Global Geolocated Photons ATL03

ผลผลิตที่ได้จากระบบดังกล่าวคือ Global Geolocated Photons (ATL03) ซึ่งบ่งบอกตำแหน่งและเวลาของ photon ที่บันทึกเข้าระบบ และพารามิเตอร์อื่นๆ ตัวอย่างผลลัพธ์บริเวณหาดแม่รำพึง ระยะ



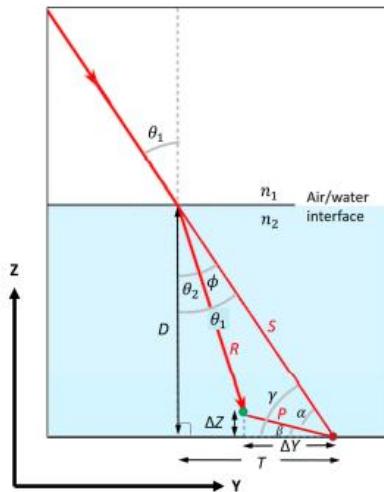
IceSAT2 Lidar – Photon Data

ภาพที่ 7 ตัวอย่าง Geolocated Photon ATL03 บริเวณหาดแม่รำพึง ระยอง

จาก openaltimetry.org

ข้อมูลค่าระดับสามารถดาวน์โหลดได้จาก open altimetry โดยเลือกพื้นที่ศึกษาและเวลาที่บันทึกข้อมูล
สามารถดาวน์โหลดข้อมูลในรูปแบบเต็ม hdf5 file ได้หรือแบบลดรูปเป็น csv file

ข้อมูลค่าระดับจำเป็นต้องทำการตรวจสอบแก้ผลกระทบของการหักเห (Refraction Correction) จากการวิจัย (Parrish et. al, 2019) ได้เสนอแนวคิด โดยมีการตรวจแก้ผลอันเนื่องมาจากการเคลื่อนผ่านรอยต่อระหว่างตัวกลางอากาศและน้ำ และการเปลี่ยนแปลงความเร็วในคลื่มน้ำของน้ำ จากจุดเดิมเป็นสีเขียว ดังภาพที่ 8



ภาพที่ 8 Refraction Correction

จาก Validation of ICESat-2 ATLAS Bathymetry and Analysis of ATLAS's Bathymetric Mapping Performance (Parrish et. al, 2019)

ข้อมูลที่ต้องทราบประกอบด้วยค่าระดับของห้องน้ำ (Geolocated seafloor photon returns) พร้อมค่าพิกัดทางระบบที่ต้องทราบ (E, N, H) และผิวน้ำ (Water surface photon returns) $H_{surface}$ ค่าประมาณของดัชนีหักเหของอากาศ n_1 และน้ำ n_2 มุมตักษะของโฟตอน (θ_1) คำนวณจากค่า $90^\circ - \text{ref_elev}$ และ มุมทิศของ unit pointing vector (A_Z) ของโฟตอนจาก ref_azimuth ซึ่งสามารถค้นได้จาก hdf5 file ของ geolocated photon จากภาพที่ 8 วิเคราะห์การคำนวณตรวจแก้ได้ดังนี้

เริ่มต้นที่รอยต่อระหว่างอากาศและน้ำทะเล แสงที่วิ่งผ่านส่องตัวกลางจะมีการหักเหเป็นไปตาม Snell's Law

$$n_1 \sin \theta_1 = n_2 \sin \theta_2 \rightarrow \theta_2 = \sin^{-1} \left[\frac{n_1 \sin \theta_1}{n_2} \right]$$

อัตราเร็วของเลเซอร์เมื่อวิ่งจากอากาศเข้าน้ำทะเล จะลดลงโดยเป็นไปตามค่าดัชนีหักเหของแสงในตัวกลาง ส่งผลให้ระยะทางในน้ำ (ระยะ R) ที่คำนวณตอนแรกผิดไป (เป็นระยะ S) ในเวลา t เดียวกัน สามารถปรับได้ดังนี้

$$n = \frac{c}{v} = \frac{c}{d/t} \rightarrow ct = nd = n_1 S = n_2 R \rightarrow R = \left[\frac{n_1}{n_2} S \right]$$

พิจารณาสามเหลี่ยมที่มีด้าน DTS

$$D = H_{surface} - H \quad S = D \sec \theta_1 \quad \gamma = \frac{\pi}{2} - \theta_1$$

พิจารณาสามเหลี่ยมที่มีด้าน DTS ใช้ Law of Sine

$$\frac{\sin \alpha}{R} = \frac{\sin \phi}{P} \rightarrow \alpha = \sin^{-1} \left(\frac{R \sin \phi}{P} \right)$$

โดยที่ $\phi = \theta_1 - \theta_2$

พิจารณาสามเหลี่ยมที่มีด้าน RTS ใช้ Law of Cosine

$$P = \sqrt{R^2 + S^2 - 2RS \cos(\phi)}$$

พิจารณาสามเหลี่ยมที่มีด้าน YZP

$$\beta = \gamma - \alpha = (\frac{\pi}{2} - \theta_1) - \sin^{-1} \left(\frac{R \sin \phi}{P} \right)$$

จะได้ ค่าเพื่อเตรียมตรวจแก้ค่าพิกัดทางราบ และค่าตรวจแก้พิกัดทางดิ่ง

$$\Delta Y = P \cos \beta \quad \Delta Z = P \sin \beta$$

ท้ายสุดจะได้ค่าตรวจแก้พิกัดทางราบ

$$\Delta E = \Delta Y \sin Az \quad \Delta N = \Delta Y \cos Az$$

คำนวณตรวจแก้ค่าระดับห้องน้ำ

$$E' = E + \Delta E \quad N' = N + \Delta N \quad H' = H + \Delta H$$

2.1.4 Google Earth Engine

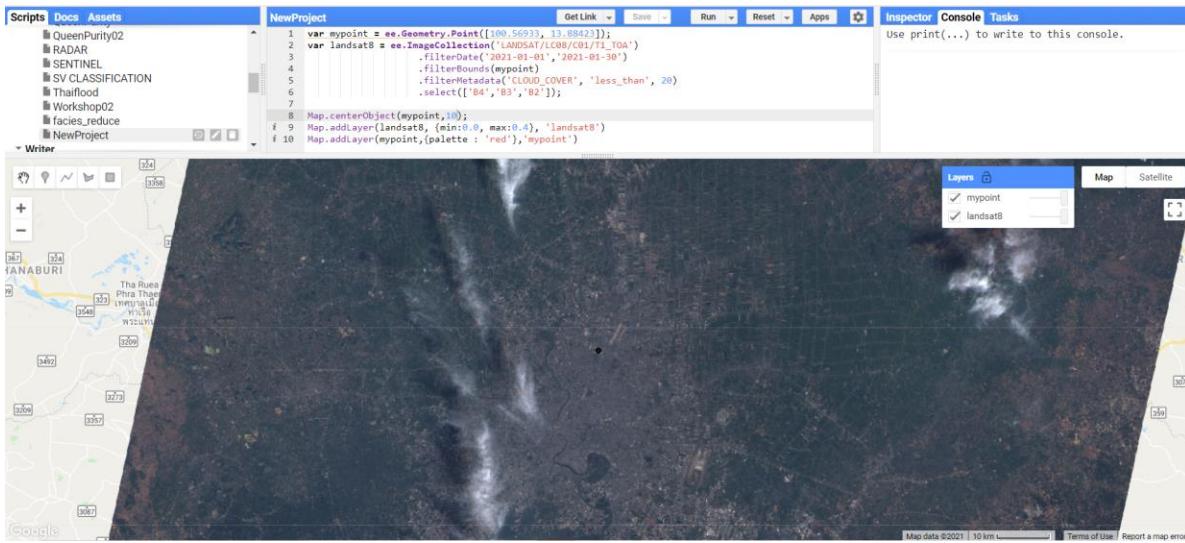
เครื่องมือ Google Earth Engine (GEE) เป็นระบบการประมวลผลและแสดงผลข้อมูลทางภูมิสารสนเทศ ตามเงื่อนไขที่เรากำหนด ทั้งเรื่องของประเภทดาวเทียม วันที่ทำการบันทึกข้อมูล ขอบเขตพื้นที่ที่ต้องการ แบบดีของดาวเทียม และเงื่อนไขอื่นๆ ซึ่งเกี่ยวข้องกับลักษณะทางเรขาคณิตหรือเชิงรังสีของภาพถ่าย เช่น เรื่องของการปกคลุมของเมฆ ความถูกต้องทางตำแหน่ง ระยะทางจากดวงอาทิตย์ เป็นต้น นำข้อมูลที่ค้นมาแสดงผลเป็นชั้นข้อมูลแผนที่ และสามารถนำมาประมวลผลเพื่อสร้างเป็นสารสนเทศทางภูมิศาสตร์ เพื่อเข้าใจในปัญหา ช่วยในการตัดสินใจแก้ไขปัญหาต่อไป ด้วยการเขียนโปรแกรมผ่านระบบอินเทอร์เน็ต

วิธีการนำเข้าข้อมูลทางด้านภูมิสารสนเทศ สามารถทำได้ทั้งการนำเข้าจากคอมพิวเตอร์ของตนเอง การนำเข้าข้อมูล Raster สามารถนำเข้าในรูปแบบ Geo TIFF ส่วนข้อมูล Vector สามารถนำเข้าในรูปแบบ shapefile และการเรียกใช้ผ่าน Google Earth Engine ด้วยการเขียนโปรแกรม โดยระบบได้เก็บข้อมูลทางภูมิศาสตร์ไว้แล้วโดยเฉพาะภาพถ่ายทางดาวเทียมไว้จำนวนมหาศาล ทั้งดาวเทียม Landsat 1-8, Sentinel 1-5 และ MODIS นอกจากนี้ยังมีแบบจำลองระดับ DEM SRTM ข้อมูลเขตการปกครอง (LSIB) และข้อมูลอื่นๆ

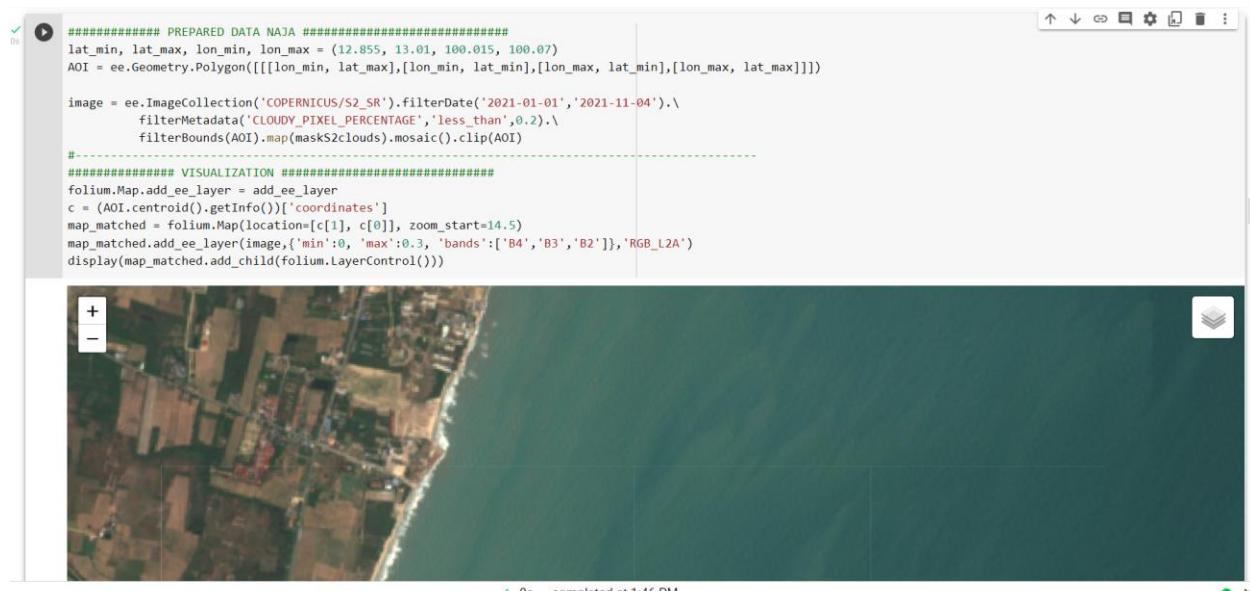
ข้อมูลภาพถ่ายทางอากาศคันได้จากเครื่องมือ ImageCollection แล้วสืบคันภาพตามวันที่ จาก filterDate สืบคันตามพื้นที่จาก filterBounds กรองเพิ่มเติมจาก filterMetadata และเลือกแบบดัชนีจาก select

การแสดงผลชั้นข้อมูลลงในแผนที่ด้านล่าง จะต้องทำการกำหนดศูนย์กลางพื้นที่จาก setCenter หรือ centerObject และแสดงผลแผนที่ จากนั้นทำการแสดงชั้นข้อมูล จาก addLayer

วิธีการใช้งานสามารถทำได้จากการเขียน javascript ผ่าน code.earthengine.google.com ดังตัวอย่าง ภาพที่ 9 หรือการเขียน python ผ่าน google colab โดยใช้งานผ่านการ import ee ดังภาพที่ 10



ภาพที่ 9 ตัวอย่างการสืบค้นภาพและแสดงผลภาพถ่ายดาวเทียมผ่าน Google Earth Engine (JavaScript)
จาก google earth engine



ภาพที่ 10 ตัวอย่างการสืบค้นภาพและแสดงผลภาพถ่ายดาวเทียมผ่าน Google Colab (Python)
จาก google colab

งานวิจัยนี้เลือกใช้ python programming เนื่องจากว่าต้องการเขียนโดยข้อมูลภาพถ่ายดาวเทียมเข้ากับข้อมูลค่าระดับจำนวนมากซึ่งการจัดการโดยใช้ pandas dataframe ตลอดจนการทำแบบจำลองด้วยการทำ Regression ผ่าน sklearn ซึ่งทั้งสองเป็น library ของ python ซึ่งมีประสิทธิภาพเป็นอย่างมาก

2.2 เอกสารและงานวิจัยที่เกี่ยวข้อง

การทำแบบจำลองระดับห้องน้ำตื้น (SDB) ต้องใช้ภาพถ่ายทางดาวเทียม ค่าระดับตัวอย่างสำหรับการเป็นค่าความลึกเริ่มต้นเพื่อนำเข้าทำแบบจำลอง (seed depth) และค่าความลึกทดสอบคุณภาพแบบจำลอง (test depth) และอัลกอริทึมในการทำแบบจำลองระดับห้องน้ำตื้น (SDB Algorithm)

ประเด็นเรื่องการเลือกใช้ภาพถ่ายทางดาวเทียม พบว่าการทำแบบจำลอง SDB จะเลือกใช้ภาพถ่ายดาวเทียมเชิงทัศน์ (Optical Satellite Imagery) ซึ่งเป็น Passive Remote Sensing อย่างภาพถ่ายจากดาวเทียม Landsat 8 (Spatial Resolution 30 m) หรือ Sentinel 2 (Spatial Resolution 10 m) โดยมีงานวิจัย (Babbel et al., 2021) ได้ทำการเปรียบเทียบความถูกต้องของแบบจำลอง SDB จากภาพถ่ายดาวเทียมสองแหล่งข้างต้นโดยใช้ค่าระดับจาก ICESat-2 และอัลกอริทึม SDB แบบ Stumpf algorithm ทดสอบในพื้นที่เกาะ St.Croix ใน U.S. Virgin Islands พบว่าการใช้ Sentinel-2 ให้ค่า Root Mean Square Error (RMSE) น้อยกว่าการใช้ Landsat-8 โดยค่าอยู่ที่ 0.96 m และ 1.54 m ตามลำดับ

ประเด็นเรื่องของอัลกอริทึมในการทำแบบจำลอง ได้มีงานวิจัย (Thomas, N. et al., 2021) ศึกษาเปรียบเทียบอัลกอริทึมสามแบบได้แก่ Cluster-based regression with Lyzenga (CBL) algorithm, Stumpf (CBS) algorithm และ Support Vector Algorithms (SVR) โดยใช้ค่าระดับจาก ICESat-2 กับภาพ Sentinel-2 ภายในพื้นที่ Bermuda, Biscayne Bay และ Crete พบว่าทั้งสามพื้นที่วิธี Lyzenga (CBL) ให้ RMSE ต่ำสุดรองลงมาเป็นวิธี Stumpf (CBS) และ SVR การวัดความถูกต้องเทียบกับค่าระดับจาก ICESat-2 อีกชุดที่เป็นอิสระต่อกัน นอกจากนี้งานวิจัยเปรียบเทียบอัลกอริทึมระหว่าง Lyzenga และ Stumpf สามารถพบได้หลากหลายงานวิจัย อย่างงานวิจัยในประเทศไทยในโคนีเซีย (L Meliala et al., 2019) ทำที่ Semak Daun reef platform ที่ทะเลสาบใช้ภาพ Sentinel-2 ประกอบค่าระดับจาก Single-beam echo sounder พบว่าทั้งสองวิธีให้ความถูกต้องอยู่ที่ 2 เมตร สำหรับความลึกไม่เกิน 10 m โดยวิธี Lyzenga ให้ความถูกต้องดีกว่า Stumpf เล็กน้อย อีกงานวิจัยเป็นที่ประเทศไทยในโคนีเซีย (Najhan Md Said, 2017) ทำที่ชายฝั่งทะเล Tawau area กับภาพถ่ายดาวเทียม Pleiades ค่าระดับที่ใช้มาจาก Single-beam echo sounder อีกเช่นกัน พบว่าวิธี Stumpf ให้ RMSE น้อยกว่า Lyzenga เล็กน้อย (1.432 m กับ 1.728 m ตามลำดับ) สองอัลกอริทึมดังกล่าวมีความน่าสนใจที่นำมาศึกษาในพื้นที่ประเทศไทยเป็นอย่างยิ่ง โดยการนำมาศึกษาทั้งในพื้นที่ชายฝั่งทะเล พื้นที่เกาะ และพื้นที่อื่น เพื่อศึกษาความเหมาะสมของอัลกอริทึมการทำแบบจำลอง SDB ต่อไป

ประเด็นเรื่องค่าระดับจาก ATLAS บนดาวเทียม ICESat-2 ควรทำการตรวจสอบแก้การหักเห (Refraction Correction) ก่อนนำไปใช้งาน (Parrish et al., 2019) เพราะช่วยลด RMSE ลงไปได้ ภายในพื้นที่ศึกษา St.Thomas Airport พบว่าลดจาก 5.14 m (Mean Error 3.41 m) ลงมา 0.43 m (Mean Error 0.24 m)

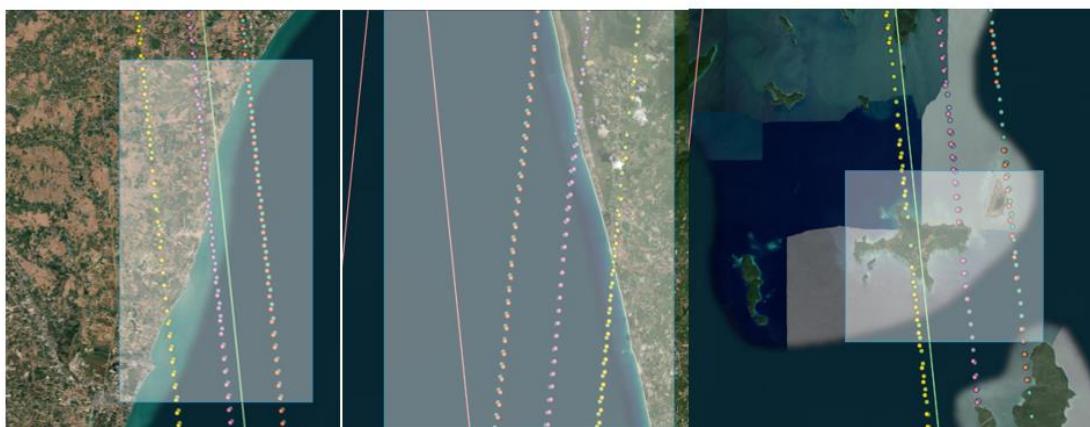
บทที่ 3

วิธีดำเนินการศึกษา

3.1 วิธีการดำเนินงานวิจัย

3.1.1 ศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้อง

3.1.2 เลือกพื้นที่ศึกษา และเลือกข้อมูลที่ผ่านบริเวณพื้นที่ศึกษา ส่องแนวเส้นทาง (ประเมินจาก Profile ที่ใกล้เคียงความจริง) เก็บไว้สำหรับ seed depth และ test depth พร้อมดาวน์โหลดข้อมูลค่าระดับ Geolocated Photon ATL03 จากระบบ ATLAS LiDAR บนดาวเทียม ICESat-2 ผ่าน openaltimetry.org แบบ Download subsetted HDF5 (via NSIDC)



ภาพที่ 11 พื้นที่ศึกษาพร้อมทั้งแนวของค่าระดับ LiDAR ทั้งสามแนวจากดาวเทียม ICESat-2 จาก openaltimetry.org

นำข้อมูลค่าระดับเข้าเก็บใน Google Drive งานวิจัยจะใช้ Google Colab ในการนำเข้าข้อมูลนี้มายังระบบประมวลผลงานวิจัย โดยการ Mount Google Drive เพื่อดึง path name ของไฟล์เราเข้าระบบ ตัวอย่าง

```
FILE_NAME = '/content/drive/MyDrive/SeniorProject/myinput/processed_ATL03_20210204162053_06521001_004_01.h5' # TRAT
```

3.1.3 คัดเลือกข้อมูลเป็น seed depth และ test depth จากแนว ICESat-2 sensor เลือกสองจากสามแนว ดังภาพที่ 9 และประเมินจากขอบเขตพื้นที่ศึกษา

3.1.4 ข้อมูลที่ได้เป็นรูปแบบ hdf5 file ให้นำข้อมูลค่าพิกัดทางราบและทางดิ่ง (latitude, longitude, photon ellipsoidal height) จัดเก็บใน folder heights ของ sensor ใน hdf5 file นำเข้า pandas dataframe

```

import h5py
import pandas as pd

with h5py.File(FILE_NAME, mode='r') as f :
    # GOT THE RAW DATA
    latfield = f['/gt2r/heights/lat_ph']
    lat = latfield[:]

    lonfield = f['/gt2r/heights/lon_ph']
    lon = lonfield[:]

    heightfield = f['/gt2r/heights/h_ph']
    h = heightfield[:]

# Data
df_train = pd.DataFrame(list(zip(lat,lon,h)),
                         columns = ['lat', 'lon', 'h'])

```

และข้อมูลค่าพิกัด reference photon พร้อมค่ามุกการเอียงตัวของเวกเตอร์เส้นทางไฟต่อน (ref_elev) และแอซิมัทเส้นทางของแนวทาง (ref_azimuth) ซึ่งจัดเก็บใน folder geolocation ของ sensor ใน hdf5 file นำเข้า dataframe อีกชุด ใช้ในการตรวจแก้ refraction correction ต่อไป

```

with h5py.File(FILE_NAME, mode='r') as f :
    # PREPARED FOR REFRACTION CORRECTION
    azimuthfield = f['/gt2r/geolocation/ref_azimuth'] ## azimuth
    az = azimuthfield[:]

    elevfield = f['/gt2r/geolocation/ref_elev'] ## looking angle
    elev = elevfield[:]

    reference_photon_lat_field = f['/gt2r/geolocation/
                                    reference_photon_lat'] ## lat
    r_lat = reference_photon_lat_field[:]

    reference_photon_lon_field = f['/gt2r/geolocation/
                                    reference_photon_lon'] ## lon
    r_lon = reference_photon_lon_field[:]

# Correction Data
df_train_cor = pd.DataFrame(list(zip(r_lat,r_lon, az,elev)),
                            columns = ['ref_lat', 'ref_lon','az', 'elev'])

```

3.1.5 ทำการเลือกส่วนที่เป็นผิวน้ำ และ ห้องน้ำ (Seasurface and seafloor segmentation) จากการ Plot ค่าระดับ ผ่าน matplotlib แบ่งเป็นส่วนย่อย และเลือกเฉพาะส่วนที่ต้องการ นำมารวมกันเป็นข้อมูลแยก ตัวอย่างการเลือกข้อมูลในพื้นที่ที่สนใจ จากกำหนดเงื่อนไขจากละเอียดและค่าระดับ

```
min_lat, max_lat = [11.829, 11.900]

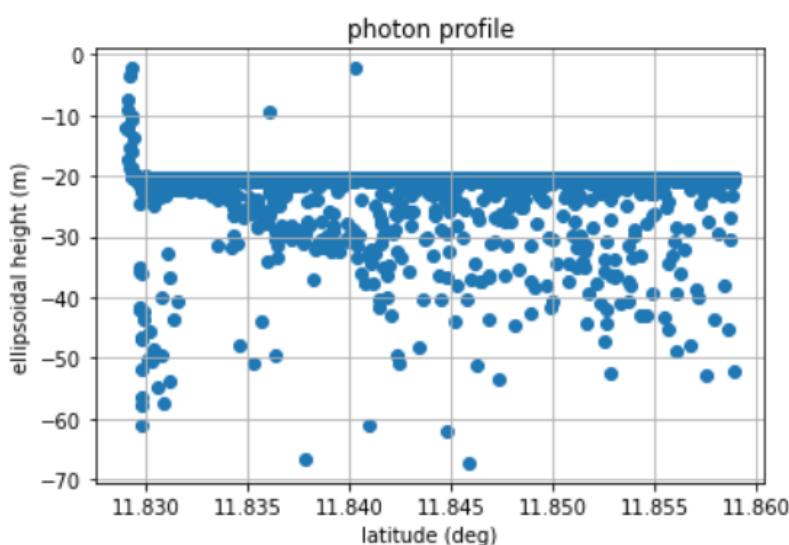
df_train_select = df_train.loc[(df_train.lat > min_lat) &
                               (df_train.lat < max_lat) & (df_train.h > -70) & (df_train.h < 0)]

df_train_cor_select = df_train_cor.loc[(df_train_cor.ref_lat > min_lat) &
                                         (df_train_cor.ref_lat < max_lat)]
train_azimuth = df_train_cor_select.az.mean()
train_elevangle = df_train_cor_select.elev.mean()
```

ตัวอย่างการแสดงผล profile ของพื้นที่จากค่าระดับที่คัดเลือกมาเป็นต้น

```
import matplotlib as mpl
import matplotlib.pyplot as plt
def photonplot(X,Y) :
    plt.title('photon profile')
    plt.ylabel('ellipsoidal height (m)')
    plt.xlabel('latitude (deg)')
    plt.scatter(X,Y)
    plt.grid()
    plt.show()

photonplot(df_train_select.lat,df_train_select.h)
```

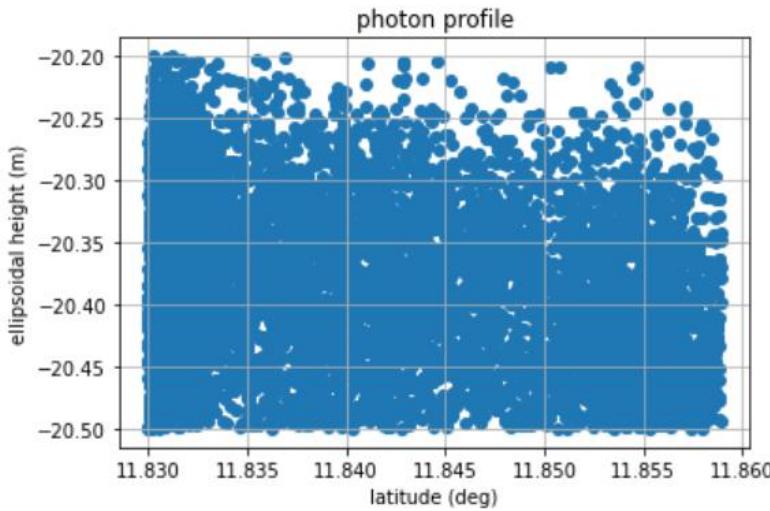


ภาพที่ 12 ตัวอย่างภาพตัดตามยาวของค่าระดับจาก ICESat-2 ภายในพื้นที่ศึกษาที่สนใจ

การเลือกส่วนของผิวน้ำ (Seafloor Segmentation) สามารถทำได้จากการกำหนดเงื่อนไขละเอียดและค่าระดับ แล้วพิจารณา profile ที่ได้ เพื่อเลือกค่าระดับผิวน้ำด้วยตนเอง

```
df_train_select_surface = (df_train_select.loc
    [(df_train_select.lat > 11.830) & (df_train_select.lat < 11.860) &
     (df_train_select.h < -20.2) & (df_train_select.h > -20.5)])
train_surface = -20.25

photonplot(df_train_select_surface.lat, df_train_select_surface.h)
```



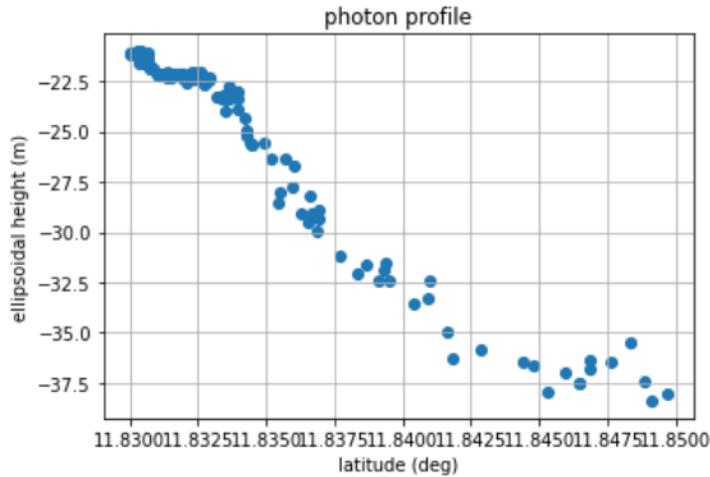
ภาพที่ 13 ตัวอย่างภาพตัดตามยาวของค่าระดับจาก ICESat-2 เพื่อเลือกค่าระดับผิวน้ำ

การเลือกส่วนของท้องน้ำ (Seasurface Segmentation) สามารถทำได้จากการกำหนดเงื่อนไขละเอียด และค่าระดับ แล้วพิจารณา profile ที่ได้ คัดเลือกทีละส่วน นำมารวมกันด้วย pd.concat

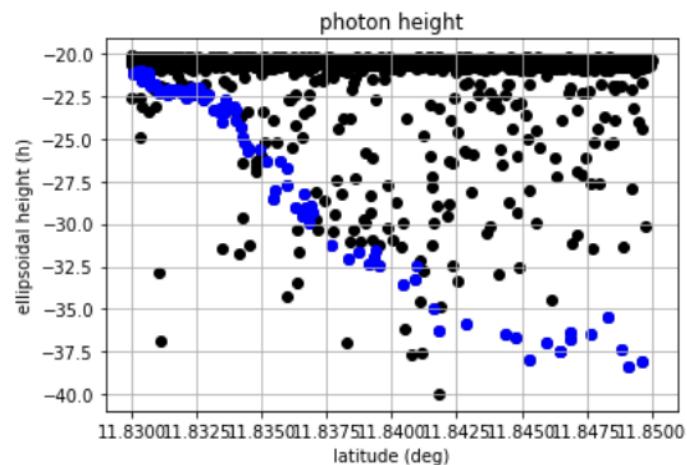
```
ps = df_train_select.copy()      # เลือกชื่อตัวแปรล้วนๆ ps = photon sample
S1 = (ps.loc[(ps.lat > 11.830) & (ps.lat < 11.831) & (ps.h < -21) & (ps.h > -22)])
S2 = (ps.loc[(ps.lat > 11.831) & (ps.lat < 11.833) & (ps.h < -22) & (ps.h > -23)])
S3 = (ps.loc[(ps.lat > 11.833) & (ps.lat < 11.834) & (ps.h < -22) & (ps.h > -24)])
S4 = (ps.loc[(ps.lat > 11.834) & (ps.lat < 11.835) & (ps.h < -24) & (ps.h > -26)])
S5 = (ps.loc[(ps.lat > 11.835) & (ps.lat < 11.837) & (ps.h < -26) & (ps.h > -30)])
S6 = (ps.loc[(ps.lat > 11.837) & (ps.lat < 11.838) & (ps.h < -31) & (ps.h > -35)])
S7 = (ps.loc[(ps.lat > 11.838) & (ps.lat < 11.841) & (ps.h < -31.5) & (ps.h > -35)])
S8 = (ps.loc[(ps.lat > 11.841) & (ps.lat < 11.842) & (ps.h < -35) & (ps.h > -37.5)])
S9 = (ps.loc[(ps.lat > 11.842) & (ps.lat < 11.850) & (ps.h < -35) & (ps.h > -40)])

trainphoton = pd.concat([S1, S2, S3, S4, S5, S6, S7, S8, S9])

photonplot(trainphoton.lat, trainphoton.h)
```



ภาพที่ 14 ตัวอย่างภาพตัดตามยาวของค่าระดับจาก ICESat-2 เพื่อเลือกค่าระดับห้องน้ำ



ภาพที่ 15 ตัวอย่างภาพตัดตามยาวของค่าระดับจาก ICESat-2 ค่าระดับของห้องน้ำเทียบกับค่าระดับทั้งหมด

3.1.6 นำข้อมูลห้องน้ำทำการตรวจสอบค่าระดับห้องน้ำเนื่องจาก refraction correction โดยใช้ค่า ref_azimuth และ ref_elev เฉลี่ยในพื้นที่ศึกษา และค่าระดับผิวน้ำ ตรวจสอบแก้แล้วนำมาหาความลึกของน้ำได้ การคำนวณความลึกของน้ำคำนวณได้จากค่าระดับของผิวน้ำ นำมาลบกับค่าระดับของห้องน้ำ

```
def getdepth(df, surface) :
    def caldepth(h_seafloor) :
        return surface - h_seafloor
    df['Depth'] = df[['new_h']].apply(caldepth, axis=1)
    return df
```

เนื่องจากสมการการตรวจแก้ใช้ค่าพิกัดในระบบพิกัดแผนที่ ยู ที เอ็ม จึงต้องมีการแปลงพิกัดโดยใช้ library ของ pyproj เขียนฟังก์ชัน การแปลงระหว่างระบบพิกัดจีออดีติกับระบบพิกัดแผนที่ ยู ที เอ็ม

```
def llatoutm(df) :
    transformer = Transformer.from_crs(4326,32647)
    UTMx,UTMy = transformer.transform(df["lat"].values, df["lon"].values)
    df['UTMx'] = UTMx
    df['UTMy'] = UTMy
    return df

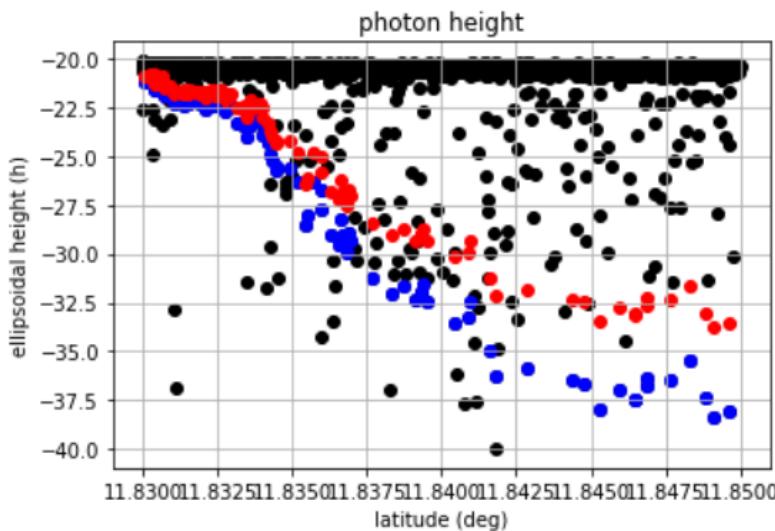
def utmtolla(df) :
    transformer = Transformer.from_crs(32647,4326)
    new_lat,new_lon = transformer.transform(df["new_E"].values,
                                              df["new_N"].values)
    df['new_lat'] = new_lat
    df['new_lon'] = new_lon
    return df
```

เมื่อเตรียมข้อมูลเรียบร้อย สามารถตรวจแก้ค่าระดับท้องน้ำเนื่องจาก Refraction Correction จาก

```
def refractioncorrection(df, azimuth, elevangle, surface) :
    def refcor_calculate (vec) :
        E, N, h = vec
        n1,n2 = [1.00029, 1.34116]
        thetal = m.pi/2 - elevangle
        theta2 = m.asin((n1*m.sin(thetal)))/n2
        phi = thetal - theta2
        D = surface - h          #depth : h(surface) - h(seafloor)
        S = D/(m.cos(thetal))
        gamma = elevangle
        R = S*(n1/n2)
        P = m.sqrt(R**2 + S**2 - 2*R*S*m.cos(thetal - theta2))
        alpha = m.asin((R*m.sin(phi))/P)
        beta = m.pi/2 - thetal - alpha
        dY = P*m.cos(beta)
        dE = dY*m.sin(azimuth)
        dN = dY*m.cos(azimuth)
        dZ = P*m.sin(beta)
        new_E = E + dE
        new_N = N + dN
        new_h = h + dZ
        return new_E, new_N, new_h
    df[['new_E', 'new_N', 'new_h']] = df[['UTMx', 'UTMy','h']]
        .apply(refcor_calculate, axis=1,result_type ='expand' )
    return df
```

ลำดับการนำไปใช้คำนวณจริงทำได้ดังนี้

```
trainphoton = llatoutm(trainphoton)
trainphoton = refractioncorrection(trainphoton, train_azimuth,
                                    train_elevangle, train_surface)
trainphoton = utmtolla(trainphoton)
trainphoton = getdepth(trainphoton, train_surface)
```



ภาพที่ 16 ตัวอย่างค่าระดับ ก่อนการตรวจแก้ (สีน้ำเงิน) และหลังการตรวจแก้ (สีแดง) เทียบกับค่าทั้งหมด (สีดำ)

3.1.7 ส่วนของภาพถ่ายดาวเทียม เขียนโปรแกรม Google Earth Engine ผ่าน Google Colab (Python)

เริ่มต้นใช้งานจะต้องเข้มต่อระหว่าง Earth Engine กับ Colab ผ่าน gmail ก่อน

```
Import ee
# Authenticate to the Earth Engine servers
ee.Authenticate()
# Initialize the API
ee.Initialize()
```

ดาวน์โหลด Sentinel 2 ในรูปแบบ L2A (Bottom of the Atmosphere) จากนั้นกรองภาพด้วยเงื่อนไข
เกี่ยวกับ เวลาถ่ายภาพ (filterDate) พื้นที่ครอบคลุมภาพ (Area of Interest : AOI, filterBounds) การปักคุณ
เมฆ (filterMetadata : Cloudy pixel percentage) และเลือกช่วงคุณที่ใช้วิจัยได้แก่ 2 (น้ำเงิน), 3 (เขียว), 4
(แดง) และ 8 (อินฟราเรด) โดยทาง Google Earth Engine ได้ช่วยกรองเมฆให้ผ่านฟังก์ชัน maskS2clouds

```

def maskS2clouds(image) :
    qa = image.select('QA60')

    # Bits 10 and 11 are clouds and cirrus, respectively.
    cloudBitMask = 1 << 10
    cirrusBitMask = 1 << 11

    # Both flags should be set to zero, indicating clear conditions.
    mask = (qa.bitwiseAnd(cloudBitMask).eq(0)) and
           (qa.bitwiseAnd(cirrusBitMask).eq(0))

    return image.updateMask(mask).divide(10000)

```

การเขียนโปรแกรมค้นภาพผ่าน Google Earth Engine

```

lat_min, lat_max, lon_min, lon_max = (11.770, 11.880, 102.440, 102.540)

AOI = ee.Geometry.Polygon([[[lon_min, lat_max], [lon_min, lat_min],
                           [lon_max, lat_min], [lon_max, lat_max]]])

Area = (lat_max - lat_min)*(lon_max - lon_min)*(111**2)      #square km

fromdate = '2020-01-01'
todate = '2022-1-31'

image = ee.ImageCollection('COPERNICUS/S2_SR').filterDate(fromdate,todate).\
         filterMetadata('CLOUDY_PIXEL_PERCENTAGE','less_than',0.05).\
         filterBounds(AOI).map(maskS2clouds).mosaic().clip(AOI)

```

สามารถศึกษาลักษณะภาพถ่ายดาวเทียมภายในพื้นที่ผ่านฟังก์ชัน

```

Import folium
def add_ee_layer(self, ee_image_object, vis_params, name):
    map_id_dict = ee.Image(ee_image_object).getMapId(vis_params)
    folium.raster_layers.TileLayer(
        tiles=map_id_dict['tile_fetcher'].url_format,
        attr='Map Data © <a href="https://earthengine.google.com/">
              Google Earth Engine</a>',
        name=name,
        overlay=True,
        control=True
    ).add_to(self)

```

การแสดงผลต้องกำหนดพื้นที่แสดงผลจาก Area of Interest, ค่าการขยายเริ่มต้น (zoom_start) ภาพถ่ายดาวเทียมจะบันทึกลงตัวแปร image สามารถแสดงเป็นชั้นข้อมูลหนึ่งของแผนที่ กำหนดพารามิเตอร์การแสดงผล ได้แก่ ค่าแบบต่ำสุด (min) ค่าแบบสูงสุด (max) และแบบตี่ที่นำมาแสดงผล ดังตัวอย่างเป็นการสร้างภาพสมสีจริง (True Color Composite) แสดงผลเป็นชั้นที่ชื่อว่า RGB_L2A ลงในแผนที่ Folium

```
folium.Map.add_ee_layer = add_ee_layer
c = (AOI.centroid()).getInfo()['coordinates']
map_matched = folium.Map(location=[c[1], c[0]], zoom_start=14)

map_matched.add_ee_layer(image, {'min':0, 'max':0.3,
                                'bands':['B4','B3','B2']}, 'RGB_L2A')

display(map_matched.add_child(folium.LayerControl()))
```



ภาพที่ 17 การแสดงผลภาพถ่ายดาวเทียมจาก Google Earth Engine ผ่าน Folium Map

3.1.8 ทำการปรับภาพถ่ายดาวเทียม ด้วยการทำ Land mask เลือกส่วนที่คาดว่าเป็นน้ำ จากเงื่อนไขการสหท้อนผ่านแบบตี่ 8 ผ่านคำสั่ง selfMask() เลือกค่าการสหท้อนที่ต่ำกว่าค่าที่กำหนด จะได้ raster ที่มีค่าในแบบตี่เป็น 1 เมื่อตรงเงื่อนไข และไม่มีค่า เมื่อไม่ตรงเงื่อนไข นำไปปรับกับแบบตี่ 2 3 4 และ 8

```

B8_threshold = 0.07
B8_c = (image.select('B8').lt(B8_threshold)).selfMask()

B2_lm = image.select('B2').multiply(B8_c)
B3_lm = image.select('B3').multiply(B8_c)
B4_lm = image.select('B4').multiply(B8_c)
B8_lm = image.select('B8').multiply(B8_c)

image = image.addBands(B2_lm.rename('B2_lm'))
image = image.addBands(B3_lm.rename('B3_lm'))
image = image.addBands(B4_lm.rename('B4_lm'))
image = image.addBands(B8_lm.rename('B8_lm'))

```

จาก Sun Glint Correction, Dark Object Subtraction Atmospheric Correction เป้าหมายคือการได้แบบต์สองและสามเตรียมพร้อมทำแบบจำลองในรูปแบบดังนี้

- แบบต์สองและสามที่ไม่ผ่านการตรวจแก้ ใช้ผลจากการ Land mask มาเลย ['B2_lm','B3_lm']
- แบบต์สองและสามที่ผ่านการตรวจแก้เพียง Sun Glint Correction ['B2_deg','B3_deg']
- แบบต์สองและสามที่ผ่านการตรวจแก้เพียง Dark Object Subtraction Atmospheric Correction ['B2_dos_fromlm','B3_dos_fromlm']
- แบบต์สองและสามที่ผ่านการตรวจแก้ทั้ง Sun Glint Correction และ Dark Object Subtraction Atmospheric Correction ['B2_dos_fromdeg','B3_dos_fromdeg']

เริ่มต้นเขียนฟังก์ชันตรวจสอบแก้ Sun Glint Correction (Deglint) ก่อน เลือกแบบต์ที่ผ่านการ landmask แล้วมาทำการประมวลผล มีการสร้างจุดสุ่มขึ้นมา 1000 จุด บันทึกค่าในแบบต์ นำค่าที่บันทึกในจุดแปลงเข้า dataframe เพื่อทำการคำนวณการ Deglint parameter ในแบบต์ 2, 3 โดยเทียบกับแบบต์ 8 จากการทำ linear regression กับ deglint model บันทึกแบบต์ที่ได้เป็น ['B2_deg', 'B3_deg', 'B4_deg']

```

from sklearn.linear_model import LinearRegression
if 1 : #Open Sun Glint Correction ....

    sampoint_landmask = image.select('B2_lm','B3_lm','B8_lm')
    POI_landmask = sampoint_landmask.sample(**{'scale': 10,'numPixels':1000,
                                                'seed':0, 'geometries': True})

    # Keep data from FeatureCollection to list, dict, pandas dataframe
    mydata_landmask = POI_landmask.getInfo()
    mylist_landmask = mydata_landmask['features']
    mydict_landmask = dict()
    mydict_landmask['lng'] = list()

```

```

mydict_landmask['lat'] = list()
mydict_landmask['B2LM'] = list()
mydict_landmask['B3LM'] = list()
mydict_landmask['B8LM'] = list()
for i in range(len(mylist_landmask)) :
    mydict_landmask['lng'].append(((mylist_landmask[i])['geometry'])['coordinates'][0])
    mydict_landmask['lat'].append(((mylist_landmask[i])['geometry'])['coordinates'][1])
    mydict_landmask['B2LM'].append(((mylist_landmask[i])['properties'])['B2_lm'])
    mydict_landmask['B3LM'].append(((mylist_landmask[i])['properties'])['B3_lm'])
    mydict_landmask['B8LM'].append(((mylist_landmask[i])['properties'])['B8_lm'])
df_landmask = pd.DataFrame(data = mydict_landmask)

# B2 Vs B8
b8 = (df_landmask.B8LM.values).reshape((-1,1))
b2 = (df_landmask.B2LM.values)
model = LinearRegression().fit(b8,b2)

# B3 Vs B8
b3 = (df_landmask.B3LM.values)
model2 = LinearRegression().fit(b8,b3)

# Deglint
x = image.select('B8_lm').subtract(df_landmask.B8LM.values.min())

B2_deg = image.select('B2_lm').subtract((x).multiply(float(model.coef_)))
B3_deg = image.select('B3_lm').subtract((x).multiply(float(model2.coef_)))

image = image.addBands(B2_deg.rename('B2_deg'))
image = image.addBands(B3_deg.rename('B3_deg'))

```

ต่อมาเป็นการเขียนฟังก์ชัน Dark Object Subtraction Atmospheric Correction (DOS) เลือกแบบนี้ที่ผ่านการ landmask และ deglint แล้วมาทำการประมวลผล มีการสร้างจุดสุ่มขึ้นมา 1000 จุด บันทึกค่าในแบบนี้ นำค่าที่บันทึกในจุดแปลงเข้า dataframe นำค่าแบบนี้ตัดสุดจากจุดสุ่ม มาลบค่าในแบบนี้นิดเดียวกัน ผลของการนำแบบนี้จาก landmask มาตรวจแก้จะได้ ['B2_dos_fromlm', 'B3_dos_fromlm'] ผลของการนำแบบนี้จาก deglint มาตรวจแก้จะได้ ['B2_dos_fromdeg', 'B3_dos_fromdeg']

```

mydata_forDOS = POI_forDOS.getInfo()
mylist_forDOS = mydata_forDOS['features']

mydict_forDOS = dict()
mydict_forDOS['lng'] = list()
mydict_forDOS['lat'] = list()
mydict_forDOS['B2_lm'] = list()
mydict_forDOS['B3_lm'] = list()
mydict_forDOS['B2_deg'] = list()
mydict_forDOS['B3_deg'] = list()

for i in range(len(mylist_forDOS)) :
    mydict_forDOS['lng'].append(((mylist_forDOS[i])['geometry'])['coordinates'][0])
    mydict_forDOS['lat'].append(((mylist_forDOS[i])['geometry'])['coordinates'][1])
    mydict_forDOS['B2_lm'].append(((mylist_forDOS[i])['properties'])['B2_lm'])
    mydict_forDOS['B3_lm'].append(((mylist_forDOS[i])['properties'])['B3_lm'])
    mydict_forDOS['B2_deg'].append(((mylist_forDOS[i])['properties'])['B2_deg'])
    mydict_forDOS['B3_deg'].append(((mylist_forDOS[i])['properties'])['B3_deg'])

df_forDOS = pd.DataFrame(data = mydict_forDOS)

## DOS FROM LANDMASK SAMPLING DATA
B2_dos_fromlm = image.select('B2_lm').subtract(df_forDOS['B2_lm'].min())
B3_dos_fromlm = image.select('B3_lm').subtract(df_forDOS['B3_lm'].min())

image = image.addBands(B2_dos_fromlm.rename('B2_dos_fromlm'))
image = image.addBands(B3_dos_fromlm.rename('B3_dos_fromlm'))

## DOS FROM DEGLINT SAMPLING DATA
B2_dos_fromdeg = image.select('B2_deg').subtract(df_forDOS['B2_deg'].min())
B3_dos_fromdeg = image.select('B3_deg').subtract(df_forDOS['B3_deg'].min())

image = image.addBands(B2_dos_fromdeg.rename('B2_dos_fromdeg'))
image = image.addBands(B3_dos_fromdeg.rename('B3_dos_fromdeg'))

```

3.1.9 นำภาพถ่ายดาวเทียม (image) และค่าระดับ Seed depth (trainphoton) มาทำแบบจำลองความลึกห้องน้ำตื้น (SDB) โดยเขียนฟังก์ชันสำหรับการประมวลผล เพื่อรับรับ 4 กรณีการตรวจแก้คือ

image = SDB(image,trainphoton,'B2_lm','B3_lm','No_Function') ไม่ตรวจแก้เลย

image = SDB(image,trainphoton,'B2_dos_fromdeg','B3_dos_fromdeg','Full_Function') ปรับหมวด

image = SDB(image,trainphoton,'B2_deg','B3_deg','Deglint_Only') ปรับเพียง Sun Glint

image = SDB(image,trainphoton,'B2_dos_fromlm','B3_dos_fromlm','DOS_Only') ปรับเพียง DOS

การทำแบบจำลอง SDB ต้องการหาค่าพารามิเตอร์ของโมเดล ด้วยการนำค่าพิกัดของค่าระดับ seed depth ไปหาค่าแบบน์ที่จัดเตรียมไว้ทั้ง 8 แบบ นำมาบันทึกใน dataframe เพื่อดึงค่าเข้ามาทำ linear regression สำหรับ SDB Model parameter ต่อไป นำผลที่ได้ไปคำนวณกับภาพถ่ายดาวเทียมเพื่อสร้างแบบจำลอง SDB

```
def getvalueforsdb(train_df,image,nametype) :
    def getvalue(vec) :
        lat,lng = vec
        u_poi = ee.Geometry.Point([lng,      lat ])
        sdb_band = image.select('lnB2perlnB3'+'_'+nametype,
                               'lnB2'+'_'+nametype,'lnB3'+'_'+nametype)
        mydata_sdb = sdb_band.sample(u_poi, 10).getInfo()
        mylist_sdb = (mydata_sdb)['features']
        if mylist_sdb != [] :
            v1 = ((mylist_sdb[0])['properties'])['lnB2perlnB3'+'_'+nametype]
            v2 =((mylist_sdb[0])['properties'])['lnB2'+'_'+nametype]
            v3 = ((mylist_sdb[0])['properties'])['lnB3'+'_'+nametype]

        else :
            v1 = None
            v2 = None
            v3 = None

    return [v1,v2,v3]

train_df[['lnB2perlnB3'+'_'+nametype,'lnB2'+'_'+nametype,
          'lnB3'+'_'+nametype]] = train_df[['new_lat', 'new_lon']]
          .apply(getvalue, axis=1, result_type ='expand')
return train_df
```

ลักษณะของฟังก์ชันทำแบบจำลอง SDB จะรับ (input) ภาพ, ข้อมูลค่าระดับ seed depth, แบบดีสอง และสามในรูปแบบการตรวจแก้ต่างๆ และซึ่อเพื่อบอกรูปแบบการตรวจแก้ ผลลัพธ์คือแบบจำลอง SDB จากสอง อัลกอริทึม (Stumpf, Lyzenga) โดยซึ่อแบบดีจะระบุรูปแบบตรวจแก้ให้ด้วย

```
def SDB(image,sampledimage,nameB2,nameB3,nametype) :
    nB2 = (image.select(nameB2)).multiply(1000)
    nB3 = (image.select(nameB3)).multiply(1000)
    lnB2perlnB3 = (nB2.log()).divide(nB3.log())
    image = image.addBands(lnB2perlnB3.rename('lnB2perlnB3'+'_'+nametype))

    lnB2 = ((image.select(nameB2)).log()).divide(0.4342944819)
    lnB3 = ((image.select(nameB3)).log()).divide(0.4342944819)
    image = image.addBands(lnB2.rename('lnB2'+'_'+nametype))
```

```

image = image.addBands(lnB3.rename('lnB3'+'_'+nametype))

SD = sampledata.copy()
getvalueforsdb(SD,image,nametype)

#####
## STUMPH ALGORITHM
SD = SD.loc[~(SD[['lnB2perlnB3'+'_'+nametype].isna())]
x_ST = (SD[['lnB2perlnB3'+'_'+nametype]].values).reshape((-1,1))
y = (SD[['Depth']].values)
model_sdb_st = LinearRegression().fit(x_ST,y)
SDB_ST = ((image.select('lnB2perlnB3'+'_'+nametype))
            .multiply(float(model_sdb_st.coef_)))
            .add(float(model_sdb_st.intercept_))
#SDB_ST = m0 + m1((ln1000B2)/(ln1000B3))
image = image.addBands(SDB_ST.rename('SDB_ST'+'_'+nametype))

#####
## LYZENGA ALGORITHM
x_LZ = SD[['lnB2'+'_'+nametype,'lnB3'+'_'+nametype]]
model_sdb_ly = LinearRegression().fit(x_LZ,y)
a0 = float(model_sdb_ly.intercept_)
a1 = (image.select('lnB2'+'_'+nametype))
            .multiply(float((model_sdb_ly.coef_[0][0])))
a2 = (image.select('lnB3'+'_'+nametype))
            .multiply(float((model_sdb_ly.coef_[0][1])))
SDB_LZ = (a1.add(a2)).add(a0)
#SDB_LZ = a0 + a1*ln(B2) + a2*ln(B3)
image = image.addBands(SDB_LZ.rename('SDB_LZ'+'_'+nametype))
return image

```

ผลลัพธ์ที่ได้สามารถนำมาแสดงผลได้ โดยสามารถปรับ visualization parameter ให้แสดงเนื้อสีได้ ดังตัวอย่าง

```

folium.Map.add_ee_layer = add_ee_layer
c = (AOI.centroid()).getInfo()['coordinates']
map_matched = folium.Map(location=[c[1], c[0]], zoom_start=13)

depth_palette = ['#caf0f8', '#90e0ef', '#00b4d8', '#0077b6', '#03045e']

map_matched.add_ee_layer(image.select('SDB_LZ_No_Function'),
                         {'min': 0, 'max': 20, 'palette' : depth_palette},
                         'SDB Stumpf No Function')
display(map_matched.add_child(folium.LayerControl()))

```



ภาพที่ 18 ตัวอย่างการแสดงผลแบบจำลอง SDB

3.1.10 นำข้อมูล test depth ไปหา Model Depth (Extract value to the points) เพื่อประเมินความถูกต้องโดยวัดเป็น Root mean square error และเขียนกราฟช่วยวิเคราะห์ผลลัพธ์ ลองปรับการประมวลผลแบบปรับ Dark Object อย่างเดียว, Sun Glint Correction อย่างเดียว และไม่มีตรวจสอบแก้เลย

การเขียนฟังก์ชันค้นค่าความลึกจากแบบจำลอง SDB จากค่าพิกัด test depth (Testphoton) มาบันทึกใน dataframe เพื่อคำนวน RMSE และวิเคราะห์แบบจำลองและอัลกอริทึมที่เหมาะสม

```
# Get Depth (raster value)
def getsdb(vec) :
    lat,lng = vec
    u_poi = ee.Geometry.Point([lng,      lat ])

    sdb_band = image.select('SDB_ST_No_Function','SDB_LZ_No_Function',
                           'SDB_ST_Full_Function','SDB_LZ_Full_Function',
                           'SDB_ST_Deglint_Only','SDB_LZ_Deglint_Only',
                           'SDB_ST_DOS_Only','SDB_LZ_DOS_Only')

    mydata_sdb = sdb_band.sample(u_poi, 10).getInfo()
    mylist_sdb = (mydata_sdb)['features']
    if mylist_sdb != [] :
        v1 = ((mylist_sdb[0])['properties'])['SDB_ST_No_Function']
        v2 = ((mylist_sdb[0])['properties'])['SDB_LZ_No_Function']
        v3 = ((mylist_sdb[0])['properties'])['SDB_ST_Full_Function']
        v4 = ((mylist_sdb[0])['properties'])['SDB_LZ_Full_Function']
```

```

v5 = ((mylist_sdb[0])['properties'])['SDB_ST_Deglint_Only']
v6 = ((mylist_sdb[0])['properties'])['SDB_LZ_Deglint_Only']
v7 = ((mylist_sdb[0])['properties'])['SDB_ST_DOS_Only']
v8 = ((mylist_sdb[0])['properties'])['SDB_LZ_DOS_Only']

else :
    v1 = None
    v2 = None
    v3 = None
    v4 = None
    v5 = None
    v6 = None
    v7 = None
    v8 = None

return [v1,v2,v3,v4,v5,v6,v7,v8]

```

นำไปใช้กับ Test Depth พอดีค่ามาแล้ว สามารถคำนวณ Root Mean Square Error ได้

```

import math as m
mytestdepth = ['ST_No_Function', 'LZ_No_Function',
               'ST_Full_Function', 'LZ_Full_Function',
               'ST_Deglint_Only', 'LZ_Deglint_Only',
               'ST_DOS_Only', 'LZ_DOS_Only']

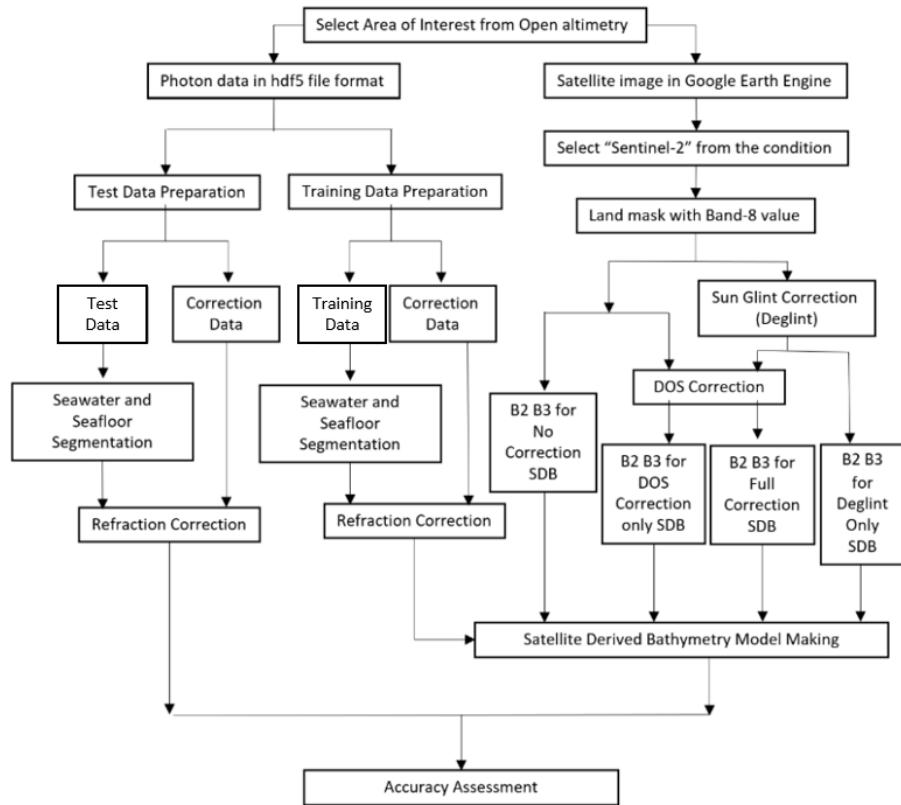
testphoton[mytestdepth] = testphoton[['new_lat', 'new_lon']]
                           .apply(getsdb, axis=1, result_type ='expand')

testphoton = testphoton.loc[~(testphoton['ST_No_Function'].isna())]
testphoton_count = testphoton['ST_No_Function'].count()
RMSE = dict()
for name in mytestdepth :
    testphoton['Diff'+name] = (testphoton[name] - testphoton['Depth'])
    testphoton['sq_Diff'+name] = (testphoton['Diff'+name])**2
    RMSE[name] = m.sqrt((testphoton['sq_Diff'+name].sum()/testphoton_count))

```

3.1.11 สรุปผลและวิเคราะห์ผลลัพธ์ เพื่อหาแนวทางการประมวลผลและอัลกอริทึมที่เหมาะสม

3.1.12 เรียบเรียงและจัดทำรายงานฉบับสมบูรณ์ของโครงการทางวิศวกรรม



ภาพที่ 19 แผนผังแสดงขั้นตอนการวิจัย

3.2 ประโยชน์ที่คาดว่าจะได้รับ

- 3.2.1 ทราบแนวทางการทำแบบจำลองความลึกห้องน้ำตื้น (SDB) ผ่าน Google Earth Engine
- 3.2.2 เข้าใจวิธีการจัดการข้อมูลและใช้ประโยชน์จากค่าระดับ LiDAR บน ICESat-2
- 3.2.3 เห็นถึงความแตกต่างของ SDB Empirical formula เพื่อเลือกใช้ในพื้นที่งานได้เหมาะสม

3.3 แผนการดำเนินงาน

ระยะเวลาดำเนินงาน 9 เดือน (สิงหาคม - เมษายน)

	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.
เลือกหัวข้อวิจัย									
ศึกษางานวิจัยที่เกี่ยวข้อง									
ดำเนินงานวิจัย									
สรุปผลและนำเสนอผลงาน									

ตารางที่ 2 ตารางแสดงแผนการดำเนินงาน

บทที่ 4

ผลการศึกษา

4.1 พื้นที่ศึกษาที่ 1 ชายฝั่งทะเลอ่าวไทย จังหวัดเพชรบุรี

ข้อมูลพื้นฐานเกี่ยวกับค่าระดับจากลิเดาร์บนดาวเทียม ICESat-2

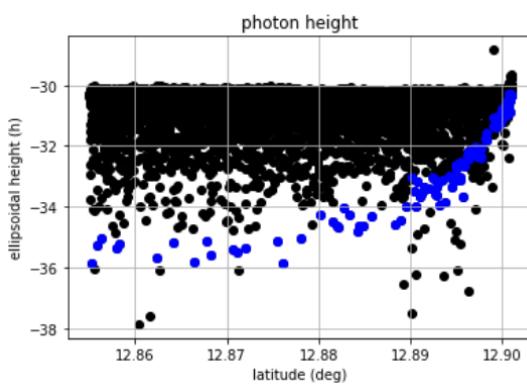
Detail	Seed Depth Data	Test Depth Data
Type of Sensor	GT2L	GT3R
Number of Observation Data	181	101
Start Latitude	12.903 degree	13.000 degree
End Latitude	12.855 degree	12.855 degree
Local Sea Level	-30.200 m	-30.250 m
Azimuth	1.902525782 rad	-0.208860323 rad
Elevation Angle	1.562974839 rad	1.568929195 rad

ตารางที่ 3 ข้อมูลพื้นฐานของค่าระดับจากลิเดาร์บริเวณพื้นที่ศึกษาชายฝั่งทะเลอ่าวไทย

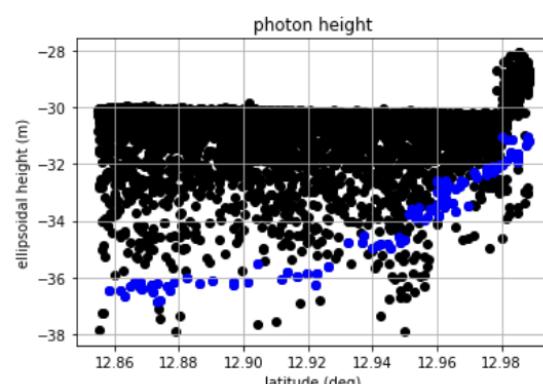
ข้อมูลภาพตัดตามยาว (Profile) ค่าระดับจากลิเดาร์บนดาวเทียม ICESat-2

ค่าระดับที่เลือกมาทำแบบจำลอง SDB ใช้สีน้ำเงิน ส่วนค่าระดับที่ได้จากการตั้งฉบับทั้งหมดใช้สีดำ

Seed Depth Data



Test Depth Data



ภาพที่ 20 ค่าระดับของลิเดาร์บริเวณพื้นที่ศึกษาชายฝั่งทะเลอ่าวไทย



ภาพที่ 21 แนวของค่าระดับจากไอลาร์ที่ใช้ทำ Seed Depth (ชั้ย แนวสีส้ม) และทำ Test Depth (ขาว แนวสีเหลือง) ภายในพื้นที่ศึกษาชายฝั่งทะเลอ่าวไทย พื้นหลังเป็นภาพถ่ายดาวเทียมจาก Sentinel2 L2A

ข้อมูลพื้นฐานเกี่ยวกับภาพถ่ายดาวเทียม

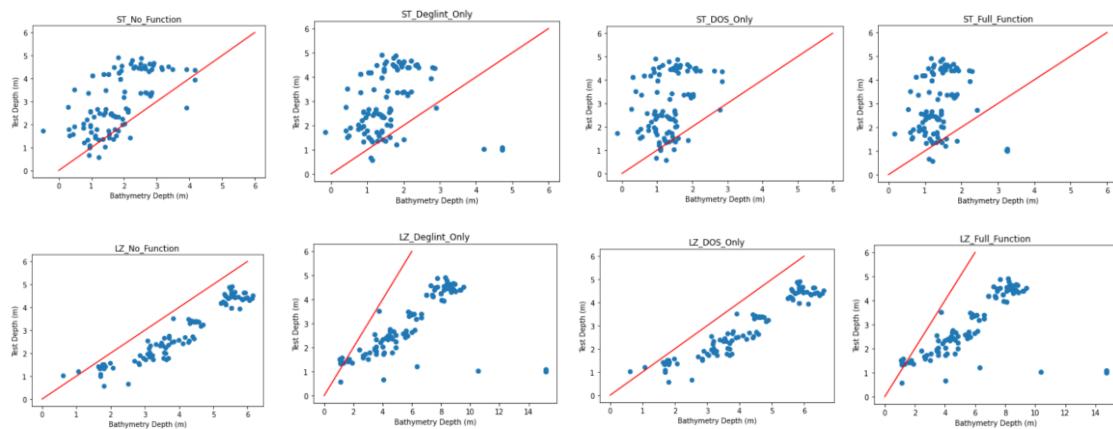
Detail	Data Description
Satellite image	Sentinel-2 L2A
Area	105.036 sq km
From date - To date	2021-01-01 to 2022-01-31
From – To Latitude	12.855 to 13.010 degree
From – To Longitude	100.015 to 100.070 degree
Cloud percentage parameter	0.2
Band 8 Threshold	0.17

ตารางที่ 4 ข้อมูลพื้นฐานของภาพถ่ายดาวเทียมสำหรับพื้นที่ศึกษาชายฝั่งทะเลอ่าวไทย

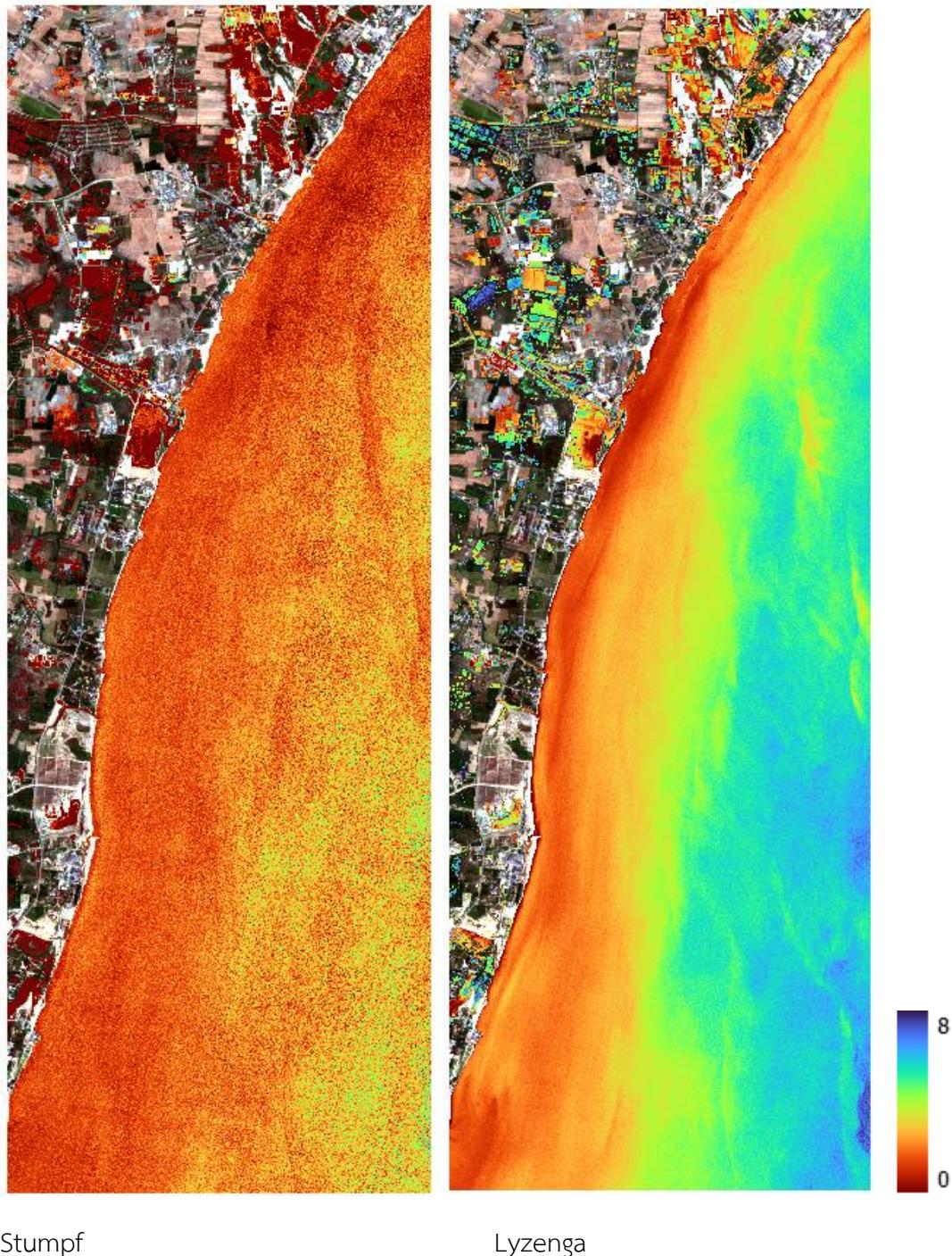
ผลลัพธ์ของการทดสอบความถูกต้อง (Accuracy Assessment)

รูปแบบการตรวจแก้ภาพ	ค่า RMSE ตามอัลกอริทึมการทำแบบจำลอง SDB (m)	
	Stumpf Algorithm	Lyzenga Algorithm
กรณีไม่ผ่านการตรวจแก้ภาพ	1.538	1.186
กรณีตรวจแก้ Sun Glint อย่างเดียว	2.008	3.865
กรณีตรวจแก้ DOS อย่างเดียว	2.071	1.359
กรณีตรวจแก้ทั้งสองแบบ	2.080	3.768

ตารางที่ 5 ผลการทดสอบความถูกต้องของแบบจำลอง สำหรับพื้นที่ชายฝั่งทะเลอ่าวไทย



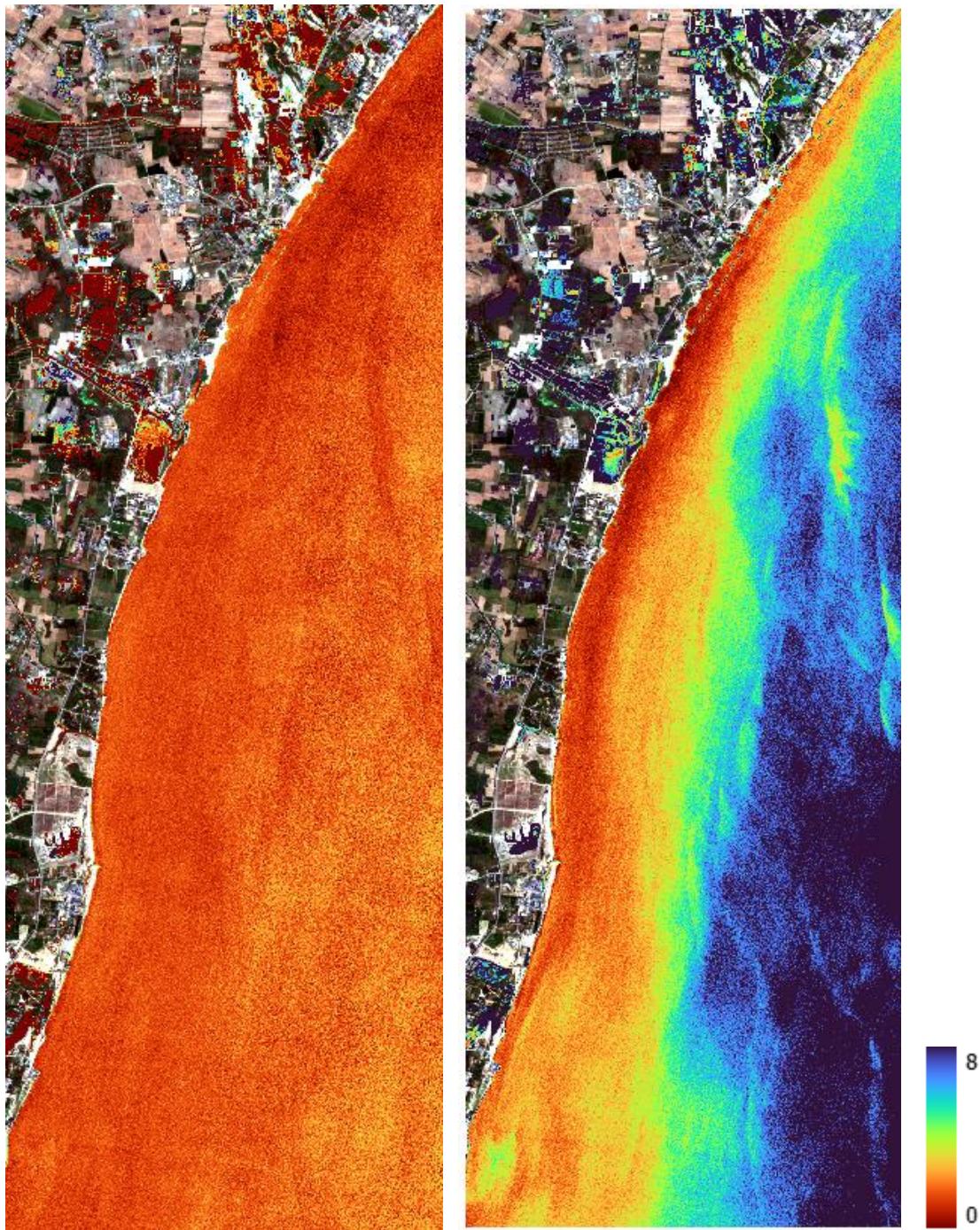
ภาพที่ 22 กราฟการกระจายของค่าความลึกจาก SDB (bathymetry depth) เทียบกับค่าความลึกจากค่าระดับบนดาวเทียม (test depth) เส้นตรงสีแดงแสดงแนวที่ค่าความลึกเท่ากัน ในพื้นที่ศึกษาชายฝั่งทะเลอ่าวไทย



Stumpf

Lyzenga

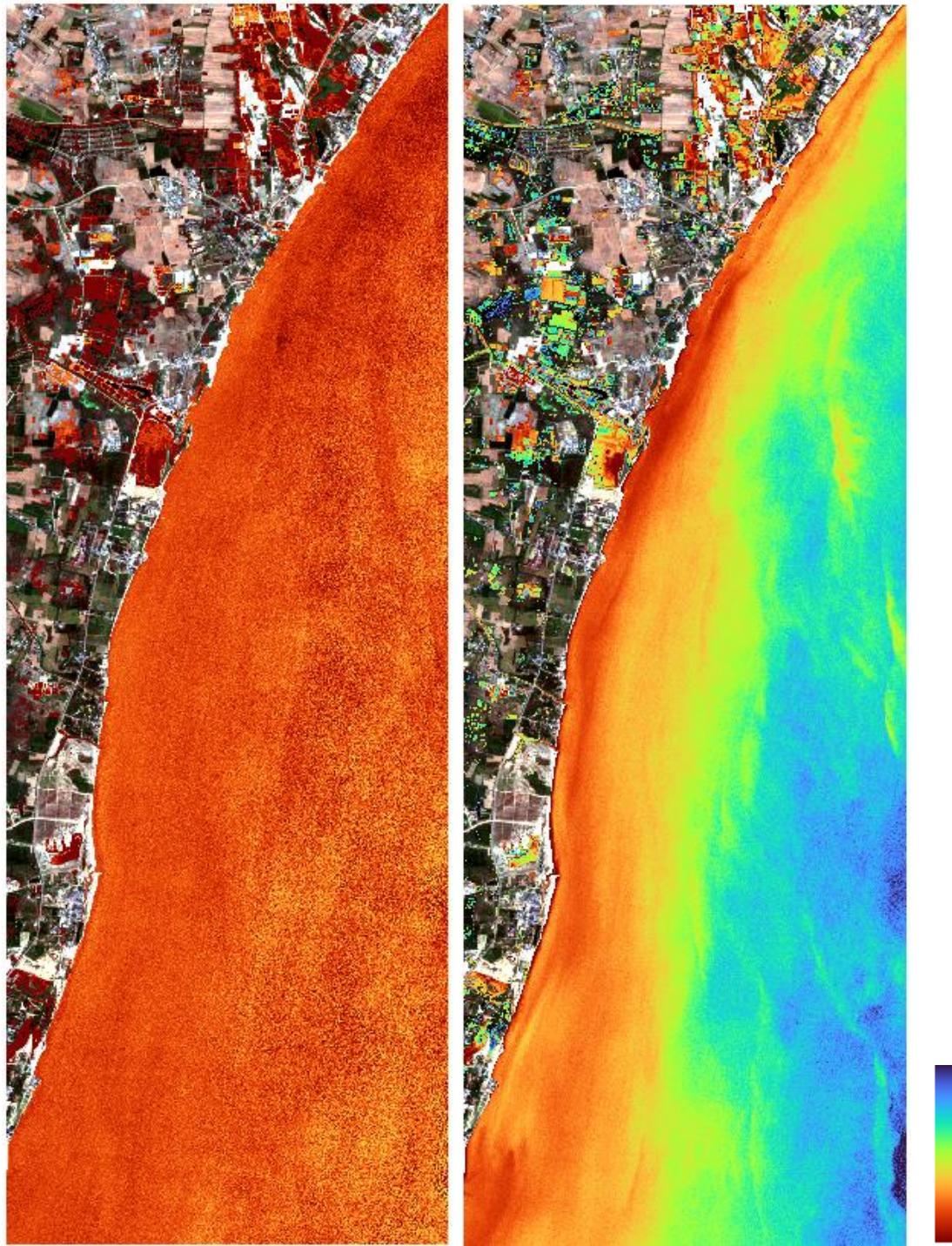
ภาพที่ 23 ผลการทำแบบจำลองความลึกห้องน้ำตื้น กรณีไม่ผ่านการตรวจแก้ภาพ บริเวณพื้นที่ศึกษาอย่างละเอียดไทย เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)



Stumpf

Lyzenga

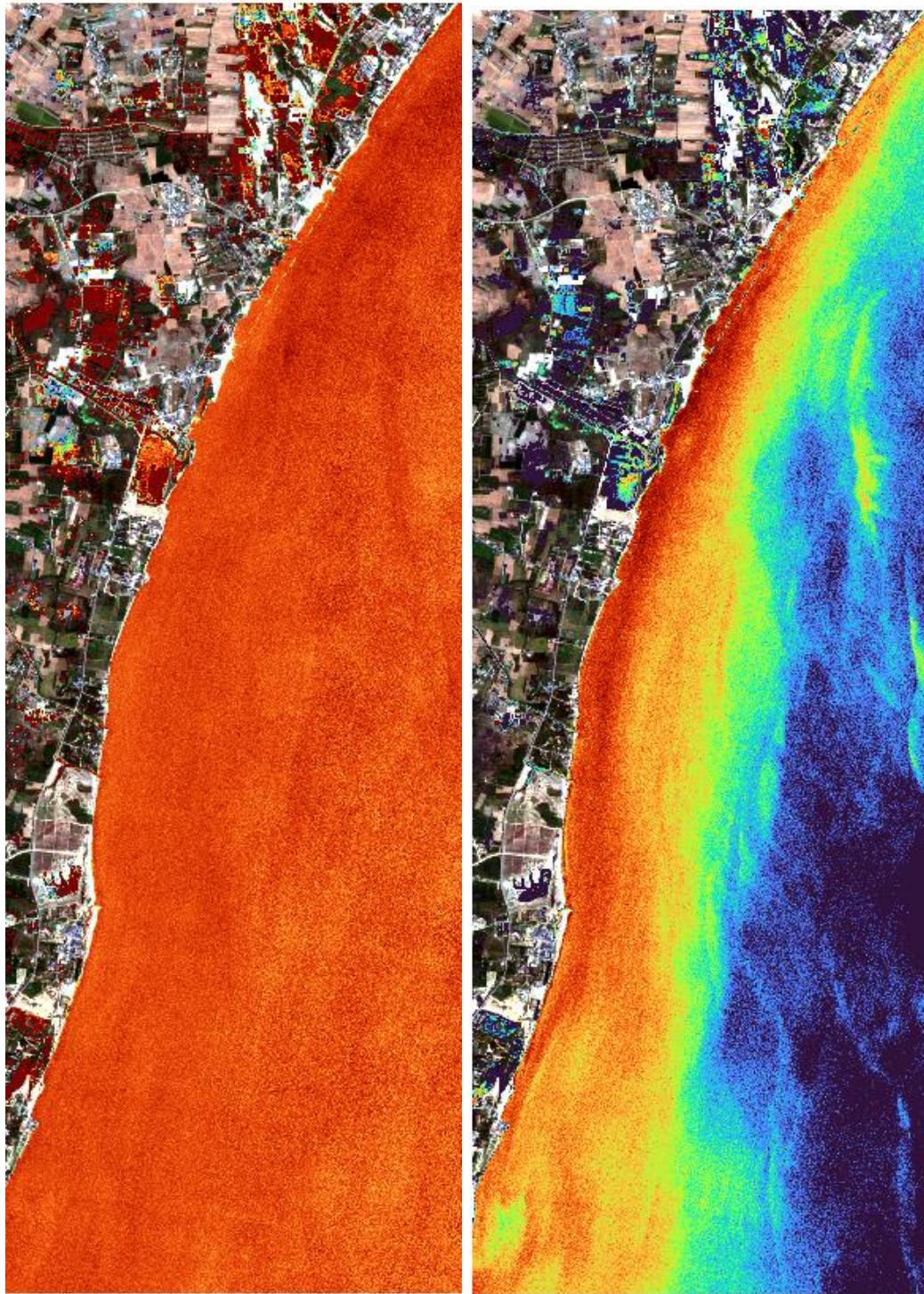
ภาพที่ 24 ผลการทำแบบจำลองความลึกห้องน้ำตื้น กรณีตรวจแก้ Sun Glint อย่างเดียว บริเวณพื้นที่ศึกษา ชายฝั่งทะเลอ่าวไทย เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)



Stumpf

Lyzenga

ภาพที่ 25 ผลการทำแบบจำลองความลึกห้องน้ำตื้น กรณีตรวจแก้ DOS อย่างเดียว บริเวณพื้นที่ศึกษาชายฝั่งทะเลอ่าวไทย เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)



Stumpf

Lyzenga

ภาพที่ 26 ผลการทำแบบจำลองความลึกห้องน้ำตื้น กรณีตรวจแก้ทั้งสองแบบ บริเวณพื้นที่ศึกษาชัยป่างทะเลอ่าวไทย เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)

พื้นที่ศึกษาที่ 2 ชายฝั่งทะเลอันดามัน บริเวณหาดท้ายเหมือง จังหวัดพังงา

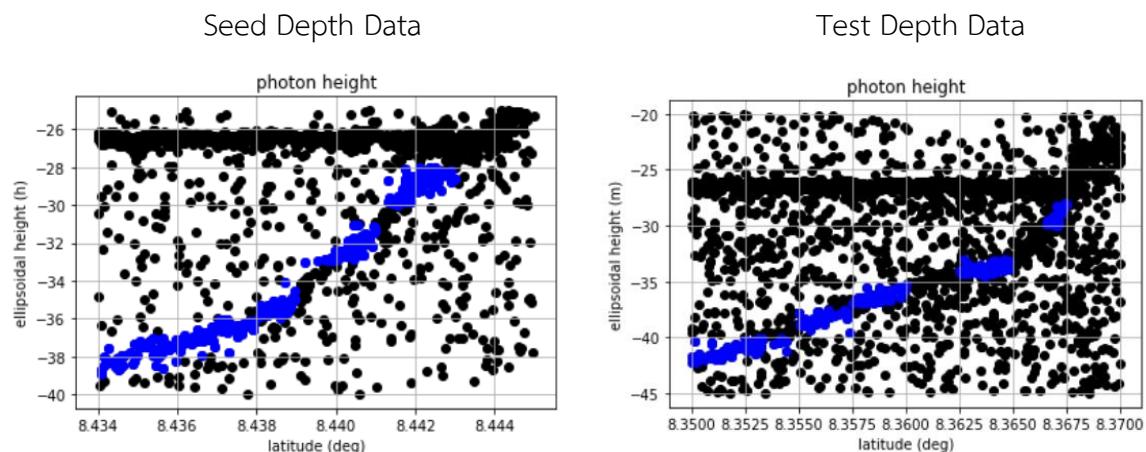
ข้อมูลพื้นฐานเกี่ยวกับค่าระดับจากกล้องดาวเทียม ICESat-2

Detail	Seed Depth Data	Test Depth Data
Type of Sensor	GT2r	GT1r
Number of Observation Data	309	331
Start Latitude	8.400	8.350
End Latitude	8.445	8.357
Local Sea Level	-26.40	-26.35
Azimuth	1.732078433 rad	1.776629567 rad
Elevation Angle	1.548252821 rad	1.555323123 rad

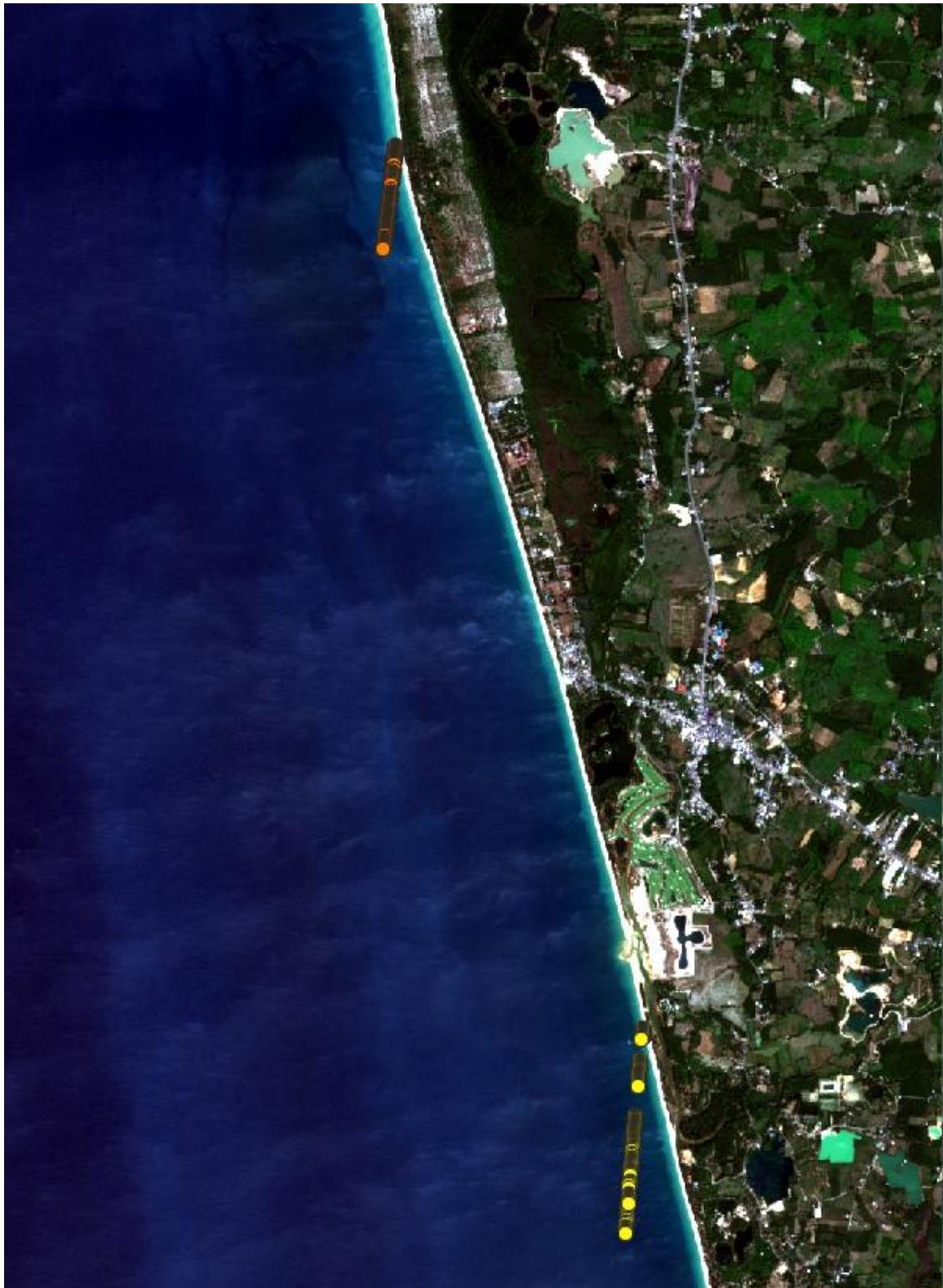
ตารางที่ 6 ข้อมูลพื้นฐานของค่าระดับจากกล้องดาวเทียม ICESat-2 ที่ศึกษาชายฝั่งทะเลอันดามัน

ข้อมูลภาพตัดตามยาว (Profile) ค่าระดับจากกล้องดาวเทียม ICESat-2

ค่าระดับที่ได้ออกมาทำแบบจำลอง SDB ใช้สีน้ำเงิน ส่วนค่าระดับที่ได้จากไฟล์ต้นฉบับทั้งหมดใช้สีดำ



ภาพที่ 27 ค่าระดับของกล้องบริเวณพื้นที่ศึกษาชายฝั่งทะเลอันดามัน



ภาพที่ 28 แนวของค่าระดับจากไลดาร์ที่ใช้ทำ Seed Depth (บน แนวสีส้ม) และทำ Test Depth (ล่าง แนวสีเหลือง) ภายในพื้นที่ศึกษาชายฝั่งทะเลอันดามัน พื้นหลังเป็นภาพถ่ายดาวเทียมจาก Sentinel2 L2A

ข้อมูลพื้นฐานเกี่ยวกับภาพถ่ายดาวเทียม

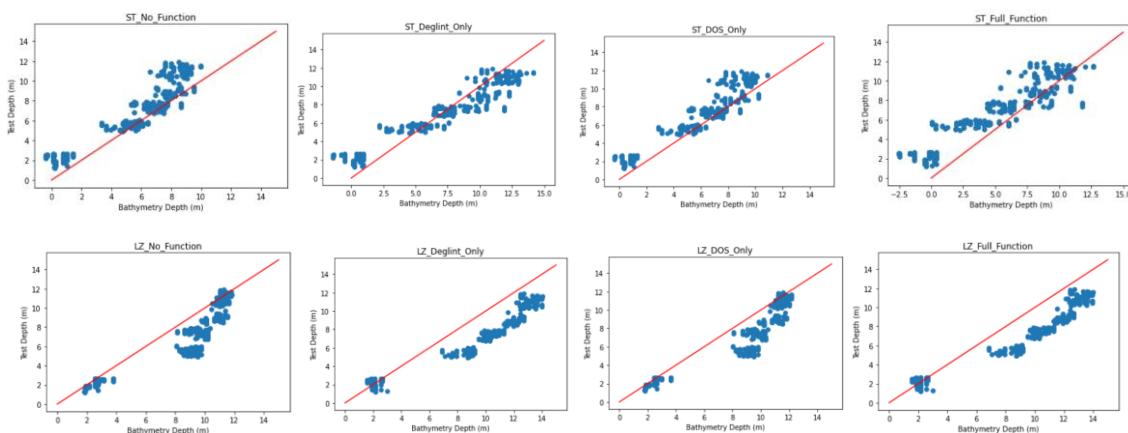
Detail	Data Description
Satellite image	Sentinel-2 L2A
Area	108.425 sq km
From date - To date	2021-01-01 to 2022-01-31
From – To Latitude	8.345 to 8.455 degree
From – To Longitude	98.200 to 98.280 degree
Cloud percentage parameter	0.06
Band 8 Threshold	0.1

ตารางที่ 7 ข้อมูลพื้นฐานของภาพถ่ายดาวเทียม สำหรับพื้นที่ศึกษาอย่างทั่วไปเลียนดามัน

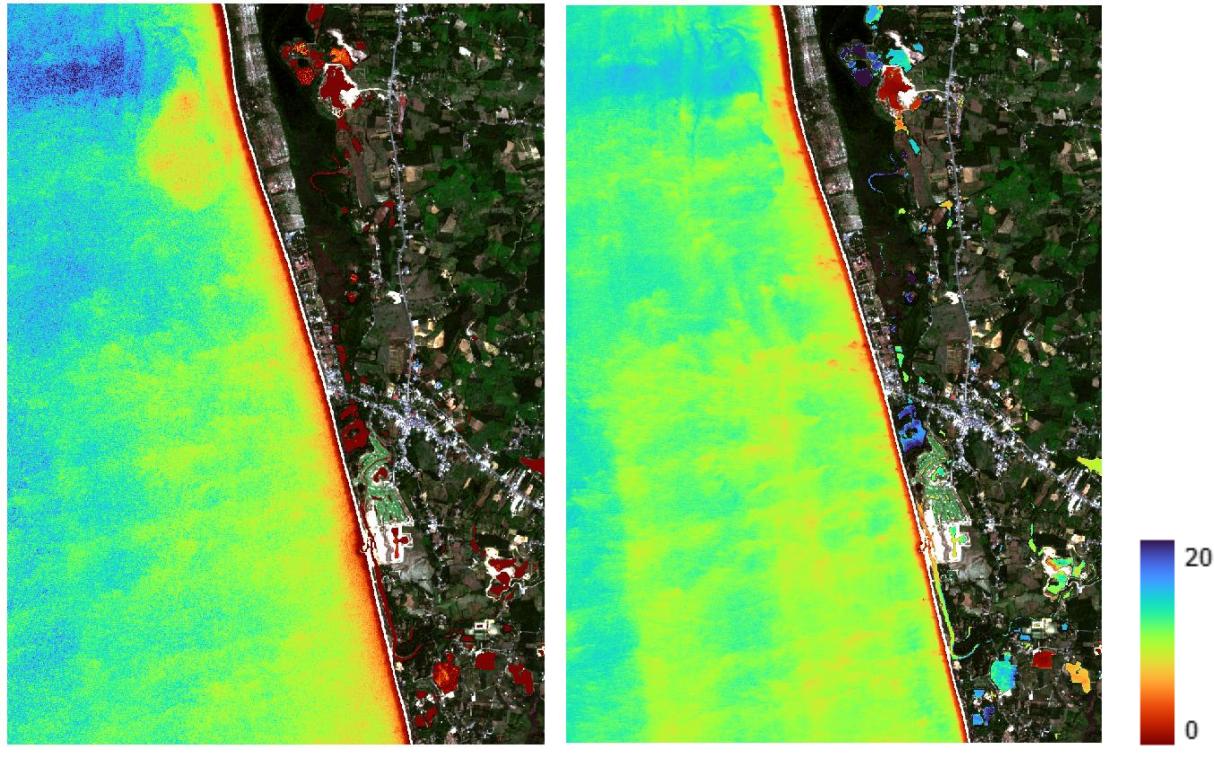
ผลลัพธ์ของการทดสอบความถูกต้อง (Accuracy Assessment)

รูปแบบการตรวจแก้ภาพ	ค่า RMSE ตามอัลกอริทึมการทำแบบจำลอง SDB (m)	
	Stumpf Algorithm	Lyzenga Algorithm
กรณีไม่ผ่านการตรวจแก้ภาพ	1.586	2.152
กรณีตรวจแก้ Sun Glint อย่างเดียว	1.661	2.566
กรณีตรวจแก้ DOS อย่างเดียว	1.474	2.194
กรณีตรวจแก้ทั้งสองแบบ	2.260	2.544

ตารางที่ 8 ผลการทดสอบความถูกต้องของแบบจำลอง สำหรับพื้นที่ช้าอย่างทั่วไปเลียนดามัน



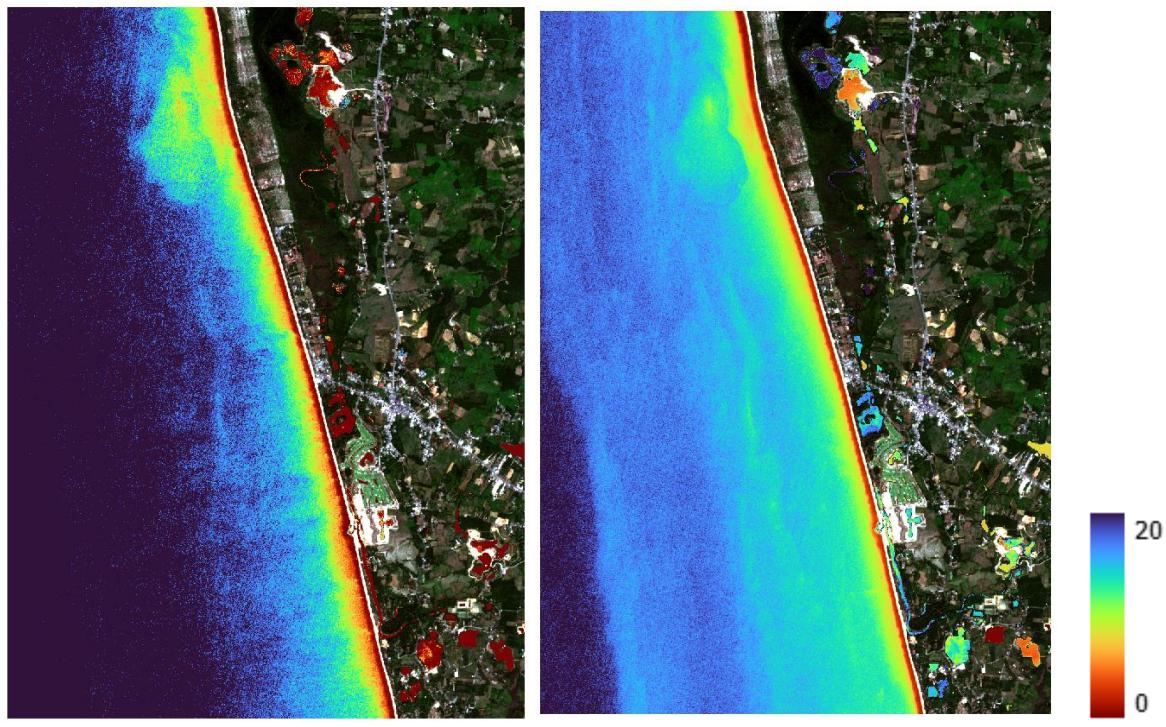
ภาพที่ 29 กราฟการกระจายของค่าความลึกจาก SDB (bathymetry depth) เทียบกับค่าความลึกจากค่าระดับบนดาวเทียม (test depth) เส้นตรงสีแดงแสดงแนวที่ค่าความลึกเท่ากัน ในพื้นที่ศึกษาอย่างทั่วไปเลียนดามัน



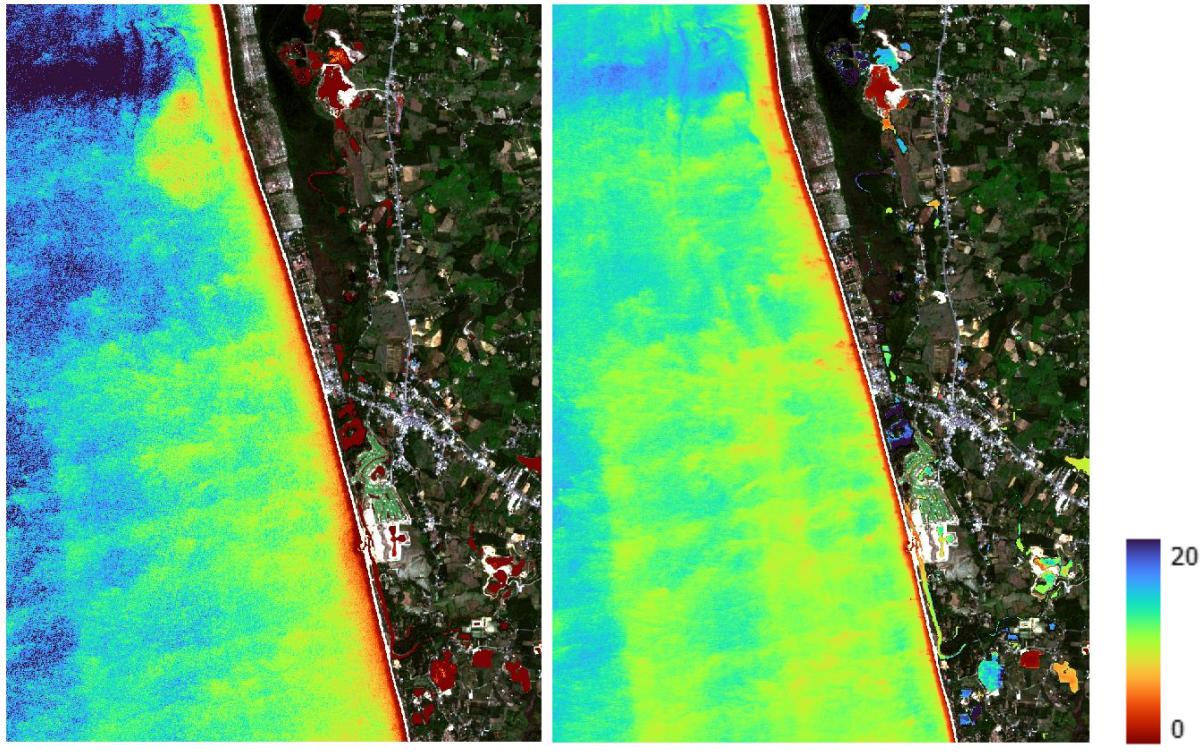
Stumpf

Lyzenga

ภาพที่ 30 ผลการทำแบบจำลองความลึกห้องน้ำตื้น กรณีไม่ผ่านการตรวจแก้ภาพ บริเวณพื้นที่ศึกษาชายฝั่งทะเล อันดามัน เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)



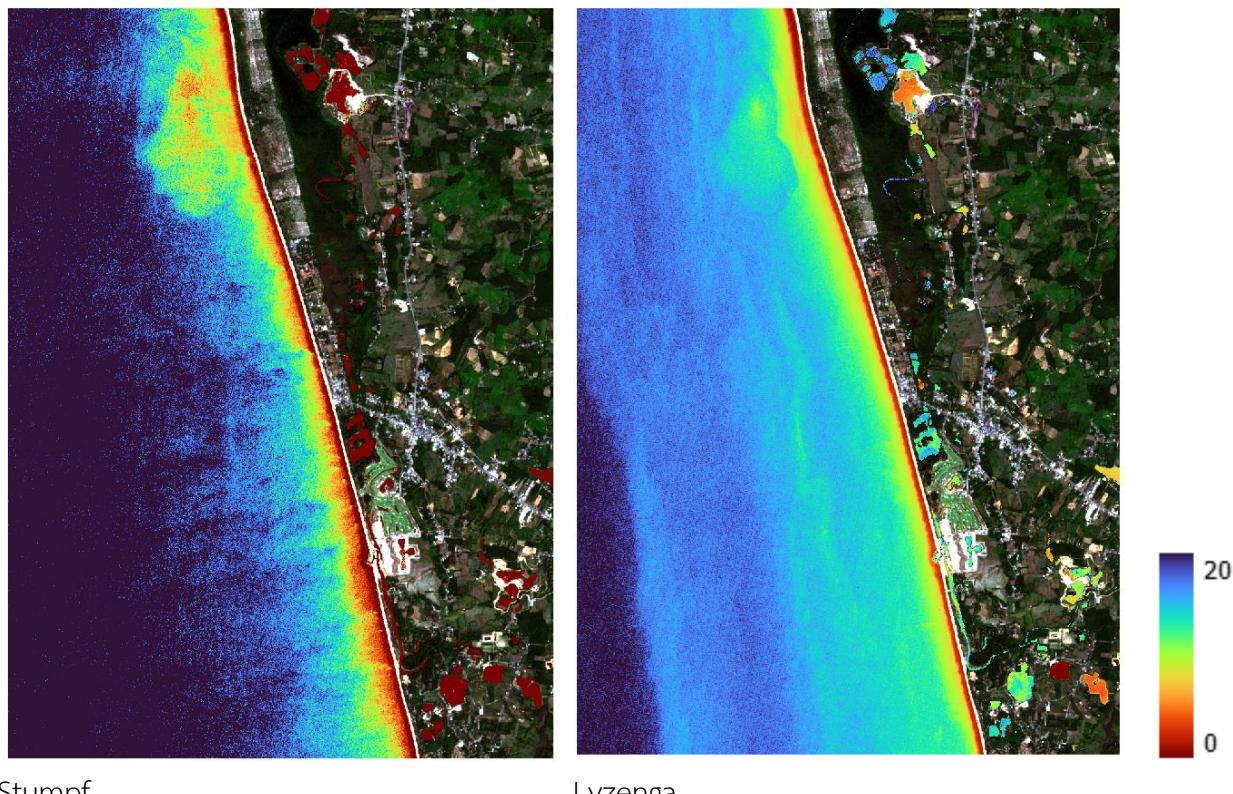
ภาพที่ 31 ผลการทำแบบจำลองความลึกห้องน้ำตื้น กรณีตรวจแก้ Sun Glint อย่างเดียว บริเวณพื้นที่ศึกษา ชายฝั่งทะเลอันดามัน เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)



Stumpf

Lyzenga

ภาพที่ 32 ผลการทำแบบจำลองความลึกห้องน้ำตื้น กรณีตรวจแก้ DOS อย่างเดียว บริเวณพื้นที่ศึกษาชายฝั่งทะเล อันดามัน เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)



ภาพที่ 33 ผลการทำแบบจำลองความลึกท้องน้ำด้านกรานีตตรวจสอบแก้ไขส่องแบบ บริเวณพื้นที่ศึกษาชายฝั่งทะเลอันดามัน เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)

พื้นที่ศึกษาที่ 3 เกาะหมาก จังหวัดตราด

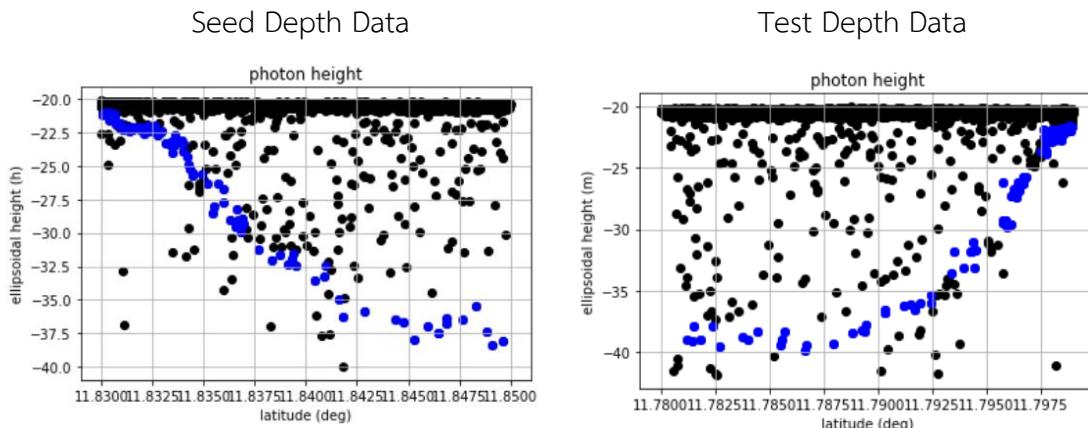
ข้อมูลพื้นฐานเกี่ยวกับค่าระดับจากกล้องดาวเทียม ICESat-2

Detail	Seed Depth Data	Test Depth Data
Type of Sensor	GT2r	GT1r
Number of Observation Data	155	99
Start Latitude	11.770	11.75
End Latitude	11.880	11.80
Local Sea Level	-20.25	-20.20
Azimuth	-0.367803722 rad	1.011665940 rad
Elevation Angle	1.567811489 rad	1.563741683 rad

ตารางที่ 9 ข้อมูลพื้นฐานของค่าระดับจากกล้องบริเวณพื้นที่ศึกษาเกาะหมาก

ข้อมูลภาพตัดตามยาว (Profile) ค่าระดับจากกล้องบริเวณพื้นที่ศึกษาเกาะหมาก

ค่าระดับที่ได้ออกมาทำแบบจำลอง SDB ใช้สีน้ำเงิน ส่วนค่าระดับที่ได้จากการไฟล์ต้นฉบับทั้งหมดใช้สีดำ



ภาพที่ 34 ค่าระดับของกล้องบริเวณพื้นที่ศึกษาเกาะหมาก



ภาพที่ 35 แนวของค่าระดับจากไลดาร์ที่ใช้ทำ Seed Depth (บน แนวสีส้ม) และทำ Test Depth (ล่าง แนวสีเหลือง) ภายในพื้นที่ศึกษาเกษตรมาก พื้นหลังเป็นภาพถ่ายดาวเทียม Sentinel-2 L2A

ข้อมูลพื้นฐานเกี่ยวกับภาพถ่ายดาวเทียม

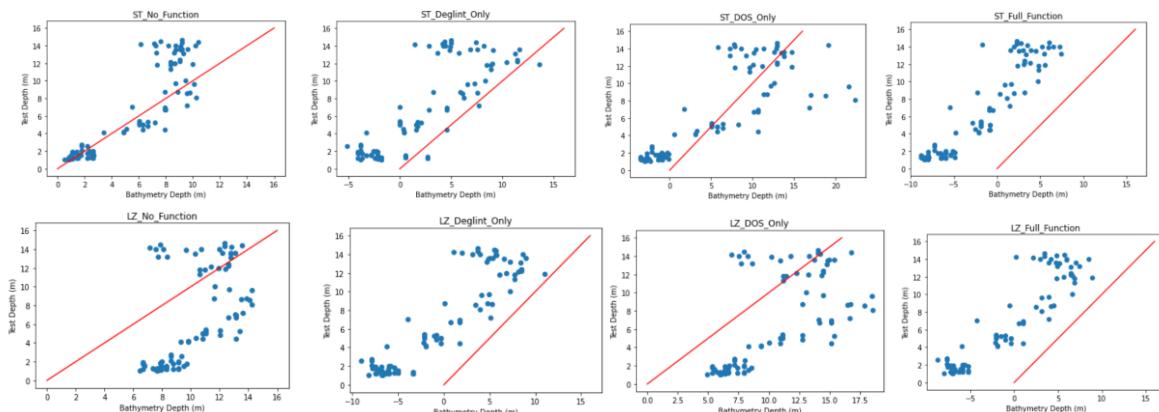
Detail	Data Description
Satellite image	Sentinel-2 L2A
Area	135.531 sq km
From date - To date	2021-01-01 to 2022-01-31
From – To Latitude	11.770 to 11.880 degree
From – To Longitude	102.440 to 102.540 degree
Cloud percentage parameter	0.05
Band 8 Threshold	0.07

ตารางที่ 10 ข้อมูลพื้นฐานของภาพถ่ายดาวเทียม สำหรับพื้นที่ศึกษาเก่ามาก

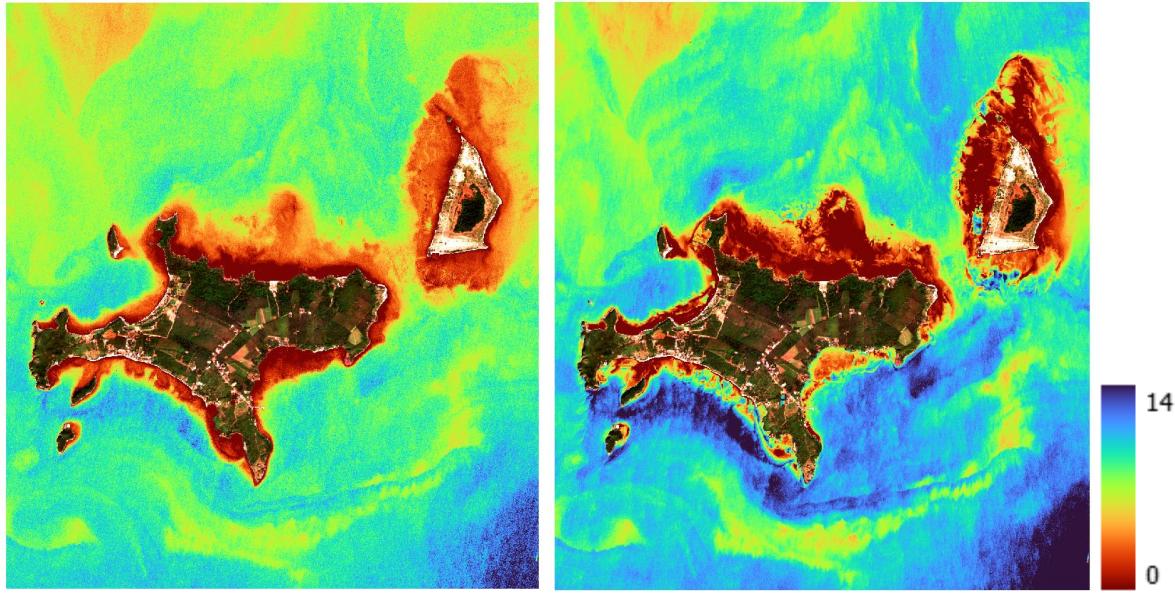
ผลลัพธ์ของการทดสอบความถูกต้อง (Accuracy Assessment)

รูปแบบการตรวจแก้ภาพ	ค่า RMSE ตามอัลกอริทึมการทำแบบจำลอง SDB (m)	
	Stumpf Algorithm	Lyzenga Algorithm
กรณีไม่ผ่านการตรวจแก้ภาพ	2.707	5.614
กรณีตรวจแก้ Sun Glint อย่างเดียว	5.058	7.461
กรณีตรวจแก้ DOS อย่างเดียว	4.244	5.408
กรณีตรวจแก้ทั้งสองแบบ	8.738	7.755

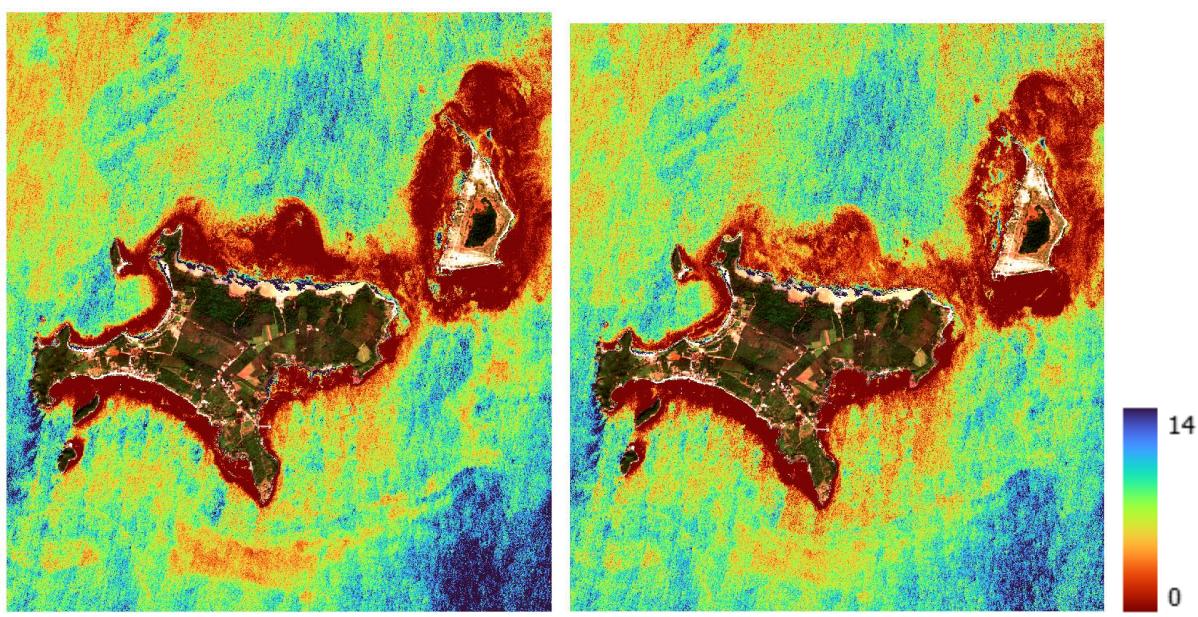
ตารางที่ 11 ผลการทดสอบความถูกต้องของแบบจำลอง สำหรับพื้นที่ศึกษาเก่ามาก



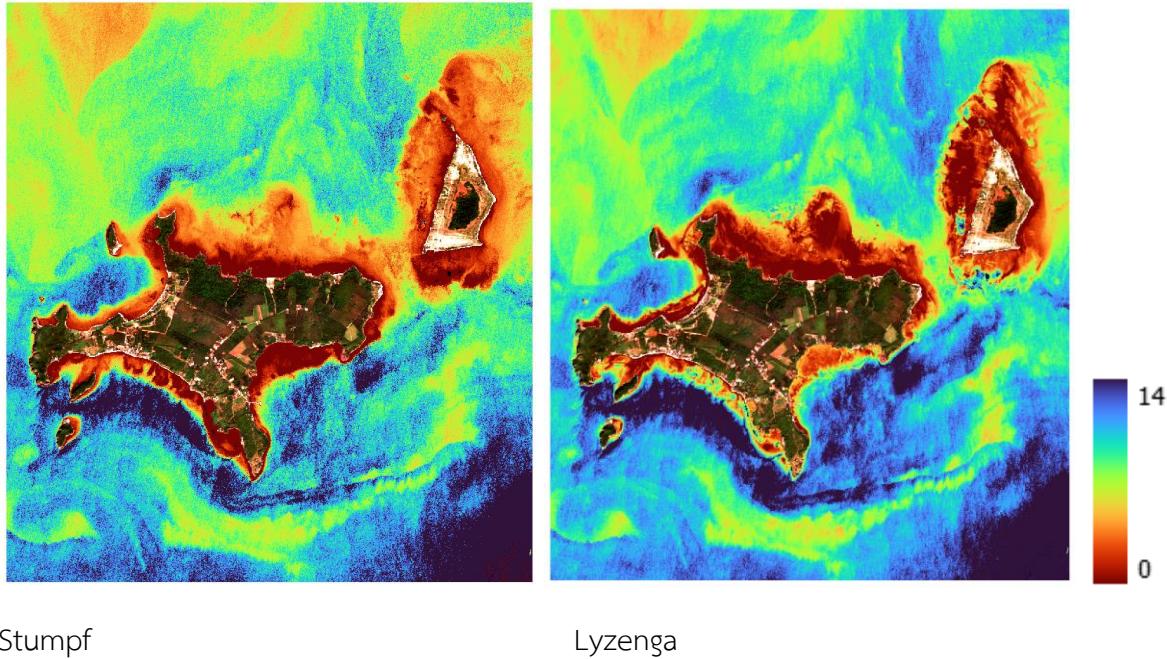
ภาพที่ 36 กราฟการกระจายของค่าความลึกจาก SDB (bathymetry depth) เทียบกับค่าความลึกจากค่าระดับบนดาวเทียม (test depth) เส้นตรงสีแดงแสดงแนวที่ค่าความลึกเท่ากัน ในพื้นที่ศึกษาเก่ามาก



ภาพที่ 37 ผลการทำแบบจำลองความลึกท้องน้ำตื้น กรณีไม่ผ่านการตรวจแก้ภาพ บริเวณพื้นที่ศึกษาเก่ามาก
เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)



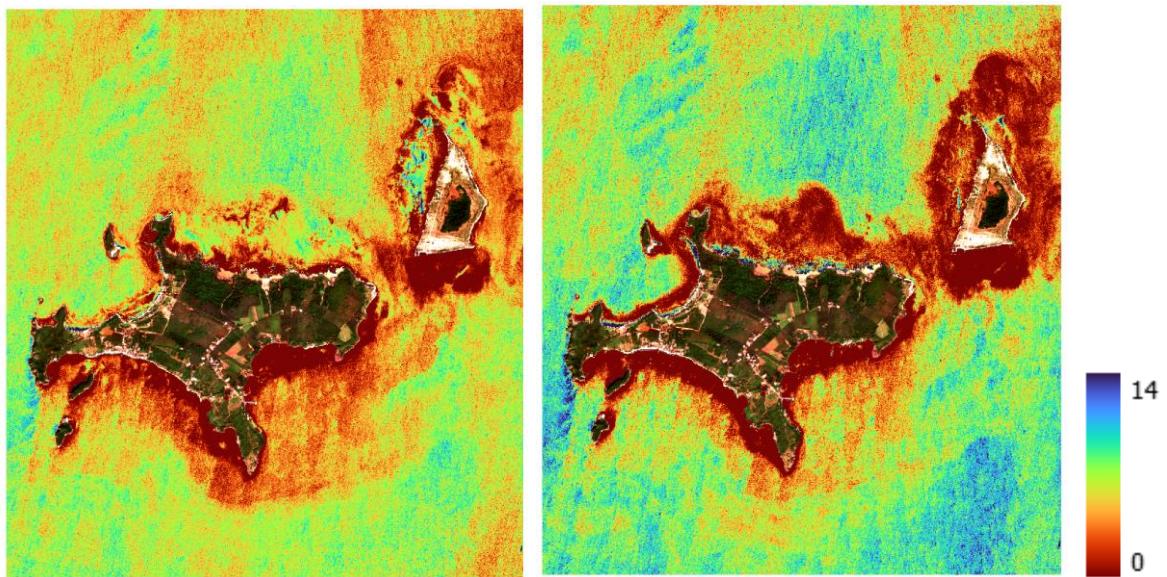
ภาพที่ 38 ผลการทำแบบจำลองความลึกท้องน้ำตื้น กรณีตรวจแก้ Sun Glint อย่างเดียว บริเวณพื้นที่ศึกษาเก่า
มาก เปรียบเทียบอัลกอริทึมของ Stumpf (ซ้าย) และ Lyzenga (ขวา)



Stumpf

Lyzenga

ภาพที่ 39 ผลการทำแบบจำลองความลึกท้องน้ำตื้น กรณีตรวจแก้ DOS อย่างเดียว บริเวณพื้นที่ศึกษาເກະໝາກ
ເປີຍບ່ອຍບ້ອລກອຣີທຶນຂອງ Stumpf (ຊ້າຍ) ແລະ Lyzenga (ຂວາ)



Stumpf

Lyzenga

ภาพที่ 40 ผลการทำแบบจำลองความลึกท้องน้ำตื้น กรณีตรวจแก้ທັງສອງແບບ บริเวณพื้นທີ່ศັກຫາເກະໝາກ
ເປີຍບ່ອຍບ້ອລກອຣີທຶນຂອງ Stumpf (ຊ້າຍ) ແລະ Lyzenga (ຂວາ)

บทที่ 5

อภิราย สรุปผล และข้อเสนอแนะ

5.1 อภิรายผลการศึกษา

งานวิจัยนี้เป็นการศึกษาการทำแบบจำลองความลึกห้องน้ำตื้น ด้วยการเขียนโปรแกรม python ผ่าน Google Colab เชื่อมโยงข้อมูลภาพถ่ายดาวเทียมจาก Google Earth Engine และค่าระดับจากไลดาร์ Geolocated Photon ATL03 บนดาวเทียม ICESat-2 พบว่าการเลือกใช้ python programming สำหรับงานวิจัยสามารถเชื่อมโยงข้อมูลภาพถ่ายดาวเทียมกับค่าระดับจากดาวเทียมได้อย่างสะดวก ประกอบกับมี library ช่วยจัดการค่าระดับ pandas dataframe และเครื่องมือประมวลผล sklearn linear regression

ผลการประเมินความถูกต้องของแบบจำลองความลึกห้องน้ำตื้นด้วยค่า Root Mean Square Error (RMSE) ของค่าความลึกจากแบบจำลองเทียบกับค่าความลึกจากค่าระดับชุดทดสอบอีกชุด สำหรับพื้นที่ศึกษาสามที่ได้แก่ ชายฝั่งทะเลอ่าวไทย จังหวัดเพชรบุรี ชายฝั่งทะเลอันดามัน จังหวัดพังงา และหมู่เกาะหมาก จังหวัดตราด ได้ผลสรุปอุปกรณ์ดังตารางที่ 12

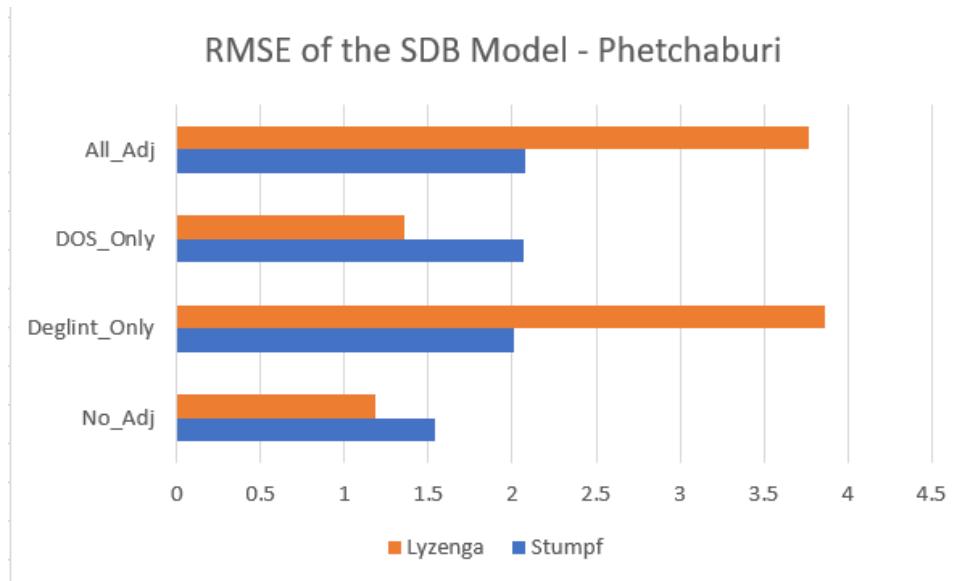
รูปแบบตรวจแก้ภาพ	RMSE (m) พื้นที่ศึกษาที่ 1 เพชรบุรี		RMSE (m) พื้นที่ศึกษาที่ 2 พังงา		RMSE (m) พื้นที่ศึกษาที่ 3 ตราด	
	Stumpf	Lyzenga	Stumpf	Lyzenga	Stumpf	Lyzenga
แบบไม่ตรวจแก้ภาพ	1.538	1.186	1.586	2.152	2.707	5.614
Sun Glint อายุ่งเดียว	2.008	3.865	1.661	2.566	5.058	7.461
DOS อายุ่งเดียว	2.071	1.359	1.474	2.194	4.244	5.408
ตรวจแก้ทั้งสองแบบ	2.080	3.768	2.260	2.544	8.738	7.755

ตารางที่ 12 ตารางสรุปการประเมินความถูกต้องของแบบจำลอง

พิจารณา RMSE ที่ต่ำสุดของแต่ละพื้นที่ศึกษา และจะพบว่าสำหรับพื้นที่ชายฝั่งทะเลอ่าวไทย จังหวัดเพชรบุรีนั้น แบบจำลองจากการไม่ตรวจแก้ภาพโดยผ่าน Lyzenga Algorithm ให้ผลต่ำสุดคือ 1.186 เมตร ส่วนพื้นที่ชายฝั่งทะเลอันดามัน จังหวัดพังงานั้น แบบจำลองจาก Dark Object Subtraction Correction (DOS) อายุ่งเดียว โดยผ่าน Stumpf Algorithm ให้ผลต่ำสุดคือ 1.474 เมตร ส่วนพื้นที่หมู่เกาะหมาก จังหวัดตราด แบบจำลองแบบไม่ตรวจแก้ภาพโดยผ่าน Stumpf Algorithm ให้ผลต่ำสุดคือ 2.706 เมตร การตัดสินว่าแนวทางการตรวจแก้ภาพและยัลกาวิทึมใดให้ความถูกต้องดีสุดตัดสินได้ยาก แต่พอเห็นได้ว่าการไม่ตรวจแก้ภาพก็

สามารถให้ความถูกต้องดีกว่าได้ โดยเฉพาะพื้นที่ศึกษาเพชรบุรี กับตราด ส่วนพังงานนั้นความถูกต้องแบบปั้ม่ตรวจแก้ภาพต่างจากแบบตรวจแก้ DOS เพียงเล็กน้อย อีกประเด็นหนึ่งคือพื้นที่หมู่เกาะมาก ค่าความถูกต้องได้ต่ำมาก มีตั้งแต่เกือบ 3 เมตร จนถึงเกือบ 9 เมตร แสดงให้เห็นถึงความซับซ้อนของพื้นที่ซึ่งแนวทางที่ทำยังไม่เหมาะสมเพียงพอ ต่อไปจะเป็นการวิเคราะห์ที่ละเอียดพื้นที่ศึกษาเพื่อหาประเด็นที่น่าสนใจดังต่อไปนี้

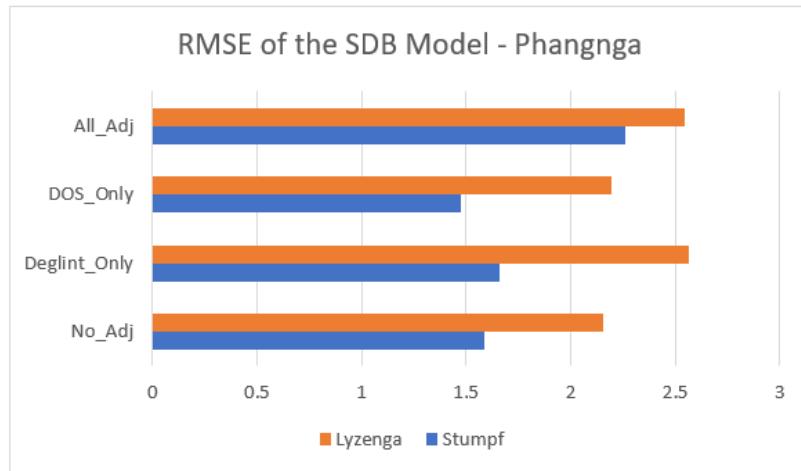
พื้นที่ศึกษาที่ 1 ชายฝั่งทะเลอ่าวไทย จังหวัดเพชรบุรี บริเวณนี้เป็นชายฝั่งยกตัว ความลึกของห้องน้ำบริเวณนี้ไม่นานนัก อยู่ในระดับหลักหน่วย กราฟแท่งเปรียบเทียบค่าความถูกต้องในพื้นที่เป็นดังนี้



ภาพที่ 41 กราฟแท่งเปรียบความถูกต้องของแบบจำลองสองแบบเทียบกับแนวทางการตรวจแก้ภาพ พื้นที่เพชรบุรี พื้นที่ศึกษานี้เป็นพื้นที่เดียวที่แบบจำลองที่ถูกต้องสูงสุดมาจาก Lyzenga Algorithm โดยทำแบบจำลอง

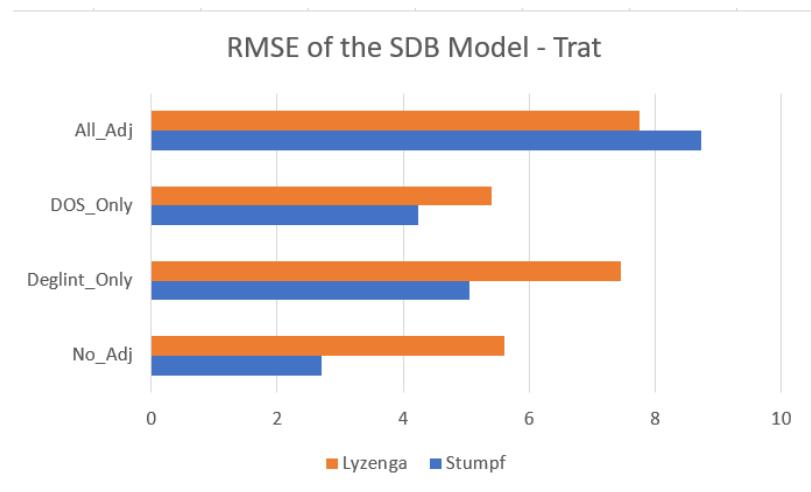
จากการที่ไม่ผ่านการตรวจแก้เลย แต่การเลือกตรวจแก้แนวทางใดแนวทางหนึ่งทำให้ความถูกต้องต่ำกว่าเดิมมาก เมื่อพิจารณาภาพที่ 22 เป็นกราฟการกระจายของความลึกจากแบบจำลองเทียบกับชุดทดสอบจะพบว่า ค่าของแบบจำลองใกล้เคียงกับค่าของชุดทดสอบที่ความลึกน้อยๆ บริเวณที่ความลึกไม่เกิน 2 เมตร โดย Stumpf Algorithm ให้ค่าความลึกต่ำกว่าค่าของชุดทดสอบ (Underestimated) จากการพบว่าจุดทั้งหลายมีค่าความลึกจากแบบจำลองน้อยแต่ค่าของชุดทดสอบมีค่ามาก กราฟจะกระจุกอยู่ทางซ้าย ส่วน Lyzenga Algorithm ให้ค่าความลึกที่มากกว่าชุดทดสอบ (Overestimated) โดยเฉพาะผลจากการตรวจแก้เพียง Sun Glint และตรวจแก้ทั้งสองแบบ จะมีบริเวณที่ความลึกมากกว่าค่าของชุดทดสอบมากๆ (จุดซึ่งออกนอกกลุ่มเป็นแนวเรียงกันแล้วล่าง) โดย Lyzenga แบบไม่ตรวจแก้และแบบที่ตรวจแก้ DOS อย่างเดียว มีแนวโน้มเกาะกับแนวเส้นตรงสีแดง ซึ่งเป็นแนวที่ค่าความลึกแบบจำลองใกล้เคียงค่าของชุดทดสอบ

พื้นที่ศึกษาที่ 2 ชายฝั่งทะเลอันดามัน บริเวณหาดท้ายเหมือง จังหวัดพังงา เป็นบริเวณชายฝั่งจะมีความสามารถพบรความลึกในระดับสิบเมตร ซึ่งกราฟแท่งเปรียบเทียบค่าความถูกต้องในพื้นที่เป็นดังนี้



ภาพที่ 42 กราฟแท่งเปรียบความถูกต้องของแบบจำลองสองแบบเทียบกับแนวทางการตรวจแก้ภาพ พื้นที่พังงา

พื้นที่ศึกษานี้ความถูกต้องของแบบจำลองสูงสุดมาจากการแบบจำลองของ Stumpf Algorithm สำหรับภาพที่ตรวจแก้ Dark Object Subtraction Correction อย่างเดียวที่ 1.474 เมตร ส่วนแนวทางการตรวจแก้ใน Stumpf Algorithm ยังคงให้ความถูกต้องที่ดีกว่า Lyzenga Algorithm เมื่อพิจารณาภาพที่ 29 จะพบว่า Stumpf Algorithm นั้นบรรดาจุดที่กระจายในภาพอยู่ในแนวเดียวกับเส้นตรงสีแดง บ่งบอกว่าความลึกจากแบบจำลองให้ใกล้เคียงกับค่าจากชุดทดสอบ ส่วน Lyzenga Algorithm จุดกระจายในภาพส่วนใหญ่อยู่ใต้เส้นตรงสีแดง แสดงว่าข้อมูลนั้นมีค่าความลึกเกินกว่าค่าของชุดทดสอบ แบบจำลองนั้นประมาณได้ลึกเกินไป



ภาพที่ 43 กราฟแท่งเปรียบความถูกต้องของแบบจำลองสองแบบเทียบกับแนวทางการตรวจแก้ภาพ พื้นที่ตราด

พื้นที่ศึกษาที่ 3 หมู่เกาะมาก จังหวัดตราด เป็นบริเวณที่แนวของค่าระดับ ICESat-2 ตัดผ่านเกาะพอดี จากราฟที่ 35 ได้เลือกค่าระดับ Seed depth เป็น profile ที่ออกจากเกาะ และ Test Depth เป็น profile ที่เข้าหาเกาะ ค่าความลึกลงไปถึงหลักสิบเมตร ซึ่งกราฟแท่งเปรียบเทียบค่าความถูกต้องในพื้นที่เป็นดังภาพที่ 43

แบบจำลองความลึกแบบไม่ตรวจแก้ภาพโดยใช้ Stumpf Algorithm ให้ความถูกต้องได้ดีสุด ค่า RMSE น้อยสุดเมื่อเทียบกับกรณีอื่นอย่างเห็นได้ เมื่อพิจารณาภาพที่ 36 กราฟการกระจายของค่าความลึกจากแบบจำลอง เทียบกับค่าของชุดทดสอบ พบร่วมแบบจำลองที่ให้ค่าความถูกต้องดีสุด จุดจะกระจายในแนวเส้นตรง แต่พอถึงความลึกแบบจำลองมากขึ้น แบบจำลองจะให้ความลึกที่น้อยกว่าค่าของชุดทดสอบ บ่งว่าความลึกแบบจำลองตื้นกว่าค่าของชุดทดสอบ เมื่อความลึกมากขึ้น ส่วนแบบจำลองไม่ตรวจแก้ภาพโดยใช้ Lyzenga นั้น ส่วนใหญ่แล้ว ความลึกจากแบบจำลองมากกว่าค่าของชุดทดสอบ แต่พบตรวจแก้ภาพแบบ Sun Glint อย่างเดียวและปรับทั้งสองแบบนั้นผลลัพธ์ที่ได้คล้ายกัน คือจะได้แบบจำลองที่มีความลึกจากแบบจำลองน้อยกว่าค่าของชุดทดสอบ จากการเห็นบรรดาจุดกระจายเหนือเส้นตรงสีแดง ผลนี้ยังเกิดกับ Stumpf Algorithm แบบปรับสองแบบ

พิจารณาเปรียบเทียบผลการศึกษา กับงานวิจัยที่ใกล้เคียงกัน (Thomas, N. et al., 2021) ซึ่งใช้ค่าระดับ ICESat-2 ทั้งในขั้นการทำแบบจำลองและการตรวจสอบความถูกต้องแบบจำลอง พบร่วมสำหรับพื้นที่ศึกษาซึ่งขยายผ่านห้องน้ำตื้นมากอย่าง Biscayne Bay (ความลึกมากสุดในพื้นที่ศึกษาคือ 5 เมตร) Lyzenga Algorithm ให้ RMSE ต่ำกว่า Stumpf Algorithm เช่นเดียวกับกรณีพื้นที่ศึกษาขยายผ่านทะเลอ่าวไทยของงานวิจัยนี้ ส่วนพื้นที่ศึกษาอีกสองพื้นที่คือ Bermuda กับ Crete (ความลึกมากสุดในพื้นที่ศึกษามากกว่า 20 เมตร) Lyzenga ยังคงให้ผลที่ดีกว่า Stumpf แต่งานวิจัยกรณีขยายผ่านทะเลประเทศไทยอีกสองที่คือพังงา กับเกาะหมาก Stumpf Algorithm ให้ผลที่ดีกว่า เป็นความแตกต่างของสองงานวิจัย ซึ่งอาจเป็นผลจากความแตกต่างของพื้นที่คุณลักษณะของภาพถ่ายดาวเทียมที่ใช้ทำแบบจำลอง รูปแบบอัลกอริทึมของ Lyzenga ที่ใช้ทำแบบจำลอง โดยอัลกอริทึมนี้มีทั้งรูปแบบสมการเต็มซึ่งใช้ในงานวิจัยที่นำมาเปรียบเทียบ กับสมการปรับปรุงซึ่งใช้ในงานวิจัยนี้

5.2 สรุปผลการศึกษา

สำหรับการศึกษาการทำแบบจำลองความลึกห้องน้ำตื้นด้วยภาพถ่ายดาวเทียมผ่าน Google Earth Engine และค่าระดับจากดาวเทียม ICESat-2 นั้นแบบจำลองความลึกห้องน้ำตื้นจาก Stumpf Algorithm แบบที่มีการตรวจแก้ภาพ Dark Object Subtraction Correction มีความเหมาะสมกับพื้นที่ขยายผ่านอันดามัน จังหวัดพังงา (RMSE = 1.474 เมตร) และแบบไม่ตรวจแก้ภาพสำหรับ เกาะหมาก จังหวัดตราด (RMSE = 2.706 เมตร) ส่วน Lyzenga Algorithm แบบไม่ผ่านการตรวจแก้ภาพ (RMSE = 1.186 เมตร) มีความเหมาะสมสำหรับชายฝั่งทะเลอ่าวไทย จังหวัดเพชรบุรี ซึ่งเป็นบริเวณที่มีความตื้นของชายฝั่งมากที่สุดในบรรดาพื้นที่ศึกษาทั้งสาม

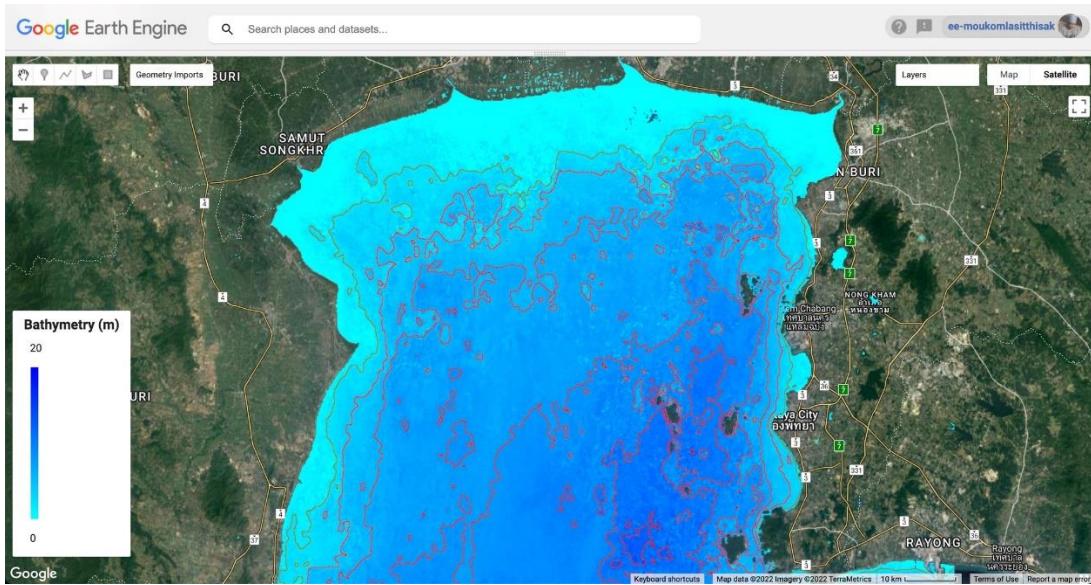
5.3 ข้อเสนอแนะ

การทำแบบจำลองความลึกท้องน้ำด้วย Empirical Formula จะมีความเหมาะสมกับพื้นที่ริมชายฝั่งทะเลที่มีอิทธิพลจากสภาพแวดล้อมน้อยสุด การกรองภาพผ่าน Google Earth Engine นอกจากทำตามวิธีค้นรูปที่พื้นที่ Metadata อื่นๆแล้ว จำเป็นต้องพิจารณาด้วยว่าภาพถ่ายในวันนั้นมีลักษณะพื้นผิวน้ำเหมาะสมหรือไม่ (เรื่องคลื่น ขยะ เรือ ตะกอน หรืออื่นๆ) ต้องให้เวลา กับการพิจารณาภาพถ่ายที่เหมาะสมด้วย จากการศึกษาการрафการกระจายของค่าความลึกจากแบบจำลองเทียบกับชุดทดสอบพบว่าลักษณะการกระจายของจุดในภาพของกรณีไม่ตรวจแก้ภาพและปรับเพียง Dark Object Subtraction Correction มีลักษณะคล้ายกัน เพราะ Dark Object Subtraction Correction เป็นการสุ่มจุดในพื้นที่ศึกษาแล้วนำค่าในแบบดั้งเดิมมาลบค่าในแบบดั้งเดิมภาพใหญ่ เป็นวิธีที่มีความไม่ซับซ้อน เมื่อเทียบกับการทำ Sun Glint และตรวจแก้ทั้งสองแบบ

แบบจำลองความลึกท้องน้ำตื้นและแนวทางการตรวจแก้ภาพมีการพัฒนาวิจัยอย่างต่อเนื่องในปัจจุบัน การเลือกใช้แนวทางการตรวจแก้ที่ดีกว่า สามารถเพิ่มคุณภาพความถูกต้องของแบบจำลองได้ โดยเฉพาะ Sun Glint Correction ซึ่งหากหาแบบจำลองตรวจแก้ภาพที่ดีกว่า คาดว่าสามารถเพิ่มคุณภาพได้ เพียงแต่ว่าในงานวิจัยนี้ Sun Glint Correction ไม่ได้ให้ผลลัพธ์ที่ดีเท่าที่ควร ตลอดจนการตรวจแก้ Dark Object Atmospheric Correction ยังมีแบบจำลองที่ซับซ้อน คาดว่าดีกว่าให้ได้ทำวิจัยในอนาคต และแบบจำลองความลึกท้องน้ำตื้นไม่ได้มีเพียง Empirical Formula ซึ่งใช้ได้ภายใต้สภาวะเงื่อนไขจำกัดอย่างหนึ่ง

ปัจจุบันนี้มีการพัฒนาการทำแบบจำลองความลึกท้องน้ำตื้นแบบบัตโนเมติกผ่าน Google Earth Engine (Jiwei Li et al., 2021) โดยใช้ภาพถ่ายดาวเทียมตลอดทั้งปีในพื้นที่ทำการโดยรวมประกอบกัน ทำการคำนวณค่าจาก mosaic surface reflectance เป็น remote sensing reflectance จากนั้นทำการลดผลกระทบอันเนื่องมาจากผิวน้ำและอากาศเป็น below-surface remote sensing นำเข้าแบบจำลองความลึกด้วย Stumpf Algorithm โดยสัมประสิทธิ์ทั้งสองนั้นทำการคำนวณจาก Natural Exponential ของค่าความเข้มข้น Chlorophyll-a บริเวณน้ำทะเลและริมชายฝั่ง วิธีดังกล่าวไม่ต้องใช้ค่าระดับชุดทำแบบจำลองเลย สามารถใช้ประมวลผลแบบจำลองได้ทั่วโลก ในพื้นที่ขยายฝั่งทะเลประเทศไทยได้มีการทดลองประมวลผลโดย facebook page ชื่อ Geography lounge ในบทความ Shallow Water Bathymetry Mapping Using Google Earth Engine เมื่อวันที่ 6 พฤษภาคม พ.ศ. 2565 ดังภาพที่ 44 ซึ่งสามารถต่อยอดเป็นงานวิจัยการประเมินความถูกต้องของแบบจำลองนี้ภายในพื้นที่ประเทศไทยได้ โดยต้องมีการรังวัดค่าความลึกในภาคสนามภายหลัง

การจัดการกับค่าระดับไอลาร์บันดาวเทียม ICESat-2 ด้วยวิธีแบบ manual selection ทำให้งานวิจัยนี้มีความลำเอียง โดยเฉพาะการเลือกค่าระดับทั้งท้องน้ำและผิวน้ำ การเลือกค่าระดับของผิวน้ำควรเลือกใช้วิธีที่อัตโนมัติกว่า เช่นการใช้เทคนิค DBSCAN (Density-based spatial clustering of applications with noise)



ภาพที่ 44 Automated Shallow Water Bathymetry Mapping Using Google Earth Engine

จาก Geography Lounge – facebook page

เพื่อเลือกส่วนของผิวน้ำ ประกอบกับข้อมูลน้ำขึ้นน้ำลงหรือค่าตรวจสอบแก่อื่นๆ เพื่อให้ได้ค่าระดับผิวน้ำทะเล ท้องถิ่นที่ถูกต้องมากขึ้น (Xie C. et al., 2021) ส่วนท้องน้ำนั้นก็ควรทำเช่นเดียวกับกรณีของการเลือกผิวน้ำ ทั้งนี้ ลักษณะของท้องน้ำมีความซับซ้อนมากทั่วทั้งพื้นผิวท้องน้ำ บางครั้งอาจพบสิ่งมีชีวิตใต้ทะเล ร่องที่ผิวท้องน้ำ โขด หินประการัง ที่เป็นเหตุให้การเลือกด้วยตนเองมีความผิดพลาดได้ หากจะนำงานวิจัยนี้ไปใช้ต่อนั้น มีประเด็นที่ควรทำการศึกษาคือการตรวจสอบคุณภาพของแบบจำลองโดยใช้ค่าความลึกที่ได้ไว้จริงในพื้นที่ภาคสนาม เพื่อสร้างความน่าเชื่อถือต่อแบบจำลองที่ได้ ตลอดจนการตรวจสอบความถูกต้องของค่าระดับจาก ICESat-2 ประกอบกับความถูกต้องจากการตรวจสอบแก่การหักเหลาบริเวณที่ประเทศไทย

เทคโนโลยีการทำแบบจำลองความลึกท้องน้ำต้นจากภาพถ่ายและค่าระดับจากดาวเทียม เป็นผลมาจากการก้าวหน้าทางวิทยาศาสตร์และ ที่ทำให้เราสามารถสร้างแบบจำลองทางวิศวกรรมสำรวจนี้โดยไม่ต้องลงไปรังวัดภาคสนาม เพียงทำการเขียนโปรแกรม ดาวน์โหลดข้อมูลดิบทั้งภาพถ่ายและค่าระดับซึ่งเปิดให้ใช้บริการโดยไม่เสียค่าใช้จ่าย ผ่านโปรแกรมในอินเทอร์เน็ต ประมวลผลเพื่อสร้างเป็นข้อมูลภูมิสารสนเทศซึ่งเป็นประโยชน์แก่การศึกษาทางด้านสมุทรศาสตร์สำหรับพื้นที่ชายฝั่งทะเลทั่วประเทศไทยได้

បរណាឌកម្ម

A Canada Centre for Remote Sensing. (n.d.) **Fundamental to Remote Sensings.** Canada.

Floyd F. Sabins (1978) **Remote Sensing : Principles and Interpretation**, Freeman & Company, W. H., San Francisco, ISBN 10: 0716700239 ISBN 13: 9780716700234

Geography Lounge. (2022, 6 May). **Shallow Water Bathymetry Mapping Using Google Earth Engine** [status update]. Facebook. <https://www.facebook.com/Geoglounge/>

Geyman, E. C., & Maloof, A. C. (2019). **A simple method for extracting water depth from multi spectral satellite imagery in regions of variable bottom type.** Earth and Space Science, 6, 527–537. <https://doi.org/10.1029/2018EA000539>

Kiyoshi Horikawa et. al. (1988) **Nearshore Dynamics and Coastal Processes: Theory, Measurement, and Predictive Models**, University of Tokyo Press, Tokyo, ISBN4-13-068138-9, ISBN0-86008-418-3

L Meliala, W A Wibowo, and Jessica Amalia, (2019), **Satellite Derived Bathymetry on Shallow Reef Platform: A Preliminary Result from Semak Daun, Seribu Islands, Java Sea, Indonesia** in The 1st International Conference on Geodesy, Geomatics, and Land Administration 2019, KnE Engineering, Li, J.; Knapp, D.E.;

Li, J.; Knapp, D.E.; Lyons, M.; Roelfsema, C.; Phinn, S.; Schill, S.R.; Asner, G.P. **Automated Global Shallow Water Bathymetry Mapping Using Google Earth Engine.** Remote Sens. 2021, 13, 1469. <https://doi.org/10.3390/rs13081469>

Lyzenga, D. R. (1978). **Passive remote sensing techniques for mapping water depth and bottom features.** Applied Optics, 17(3), 379–383

Najhan Md Said, Mohd Razali Mahmud and Rozaimi Che Hasan (2017). **Satellite-Derived Bathymetry: Accuracy Assessment on depths derivation algorithm for shallow water area.** The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLII-4/W5, 2017 GGT 2017, 4 October 2017, Kuala Lumpur, Malaysia

NASA. (2021) ICESat-2 Algorithm Theoretical Basis Document for Global Geolocated Photons (ATL03) Release 004. Retrieved from https://icesat-2.gsfc.nasa.gov/sites/default/files/page_files/ICESat2_ATL03_ATBD_r004.pdf

Parrish, C. E., Magruder, L. A., Neuenschwander, A. L., Forfinski-Sarkozi, N., Alonso, M., & Jasinski, M. (2019). **Validation of ICESat-2 ATLAS bathymetry and analysis of ATLAS's bathymetric mapping performance.** *Remote Sensing*, 11(14), 1634.

Serco Italia SPA (2021). **Nearshore Bathymetry Derivation with Sentinel-2.** (version 1.1). Retrieved from RUS Lectures at <https://rus-copernicus.eu/portal/the-rus-library/train-with-rus/>

Shachak Pe'eri, Chukwuma Azuike, Christopher Parrish (2013). **Satellite-derived Bathymetry A Reconnaissance Tool for Hydrography.** Retrieved from <https://www.hydro-international.com/content/article/satellite-derived-bathymetry>

Stumpf, R. P., Holderied, K., & Sinclair, M. (2003). **Determination of water depth with high-resolution satellite imagery over variable bottom types.** *Limnology and Oceanography*, 48(1), 547–556.

Tien Thanh Nguyen, Quang Minh Nguyen, Xiu Guo Liu, Yao Yevenyo Ziggah (2012) **3D OBJECT MODEL RECONSTRUCTION BASED ON LASER SCANNING POINT CLOUD DATA.** International Symposium on Geoinformatics for Spatial Infrastructure Development in Earth and Allied Sciences 2012

Thomas, N., Pertiwi, A. P., Traganos, D., Lagomasino, D., Poursanidis, D., Moreno, S., & Fatooyinbo, L. (2021). **Space-borne cloud-native Satellite Derived Bathymetry (SDB) models using ICESat-2 and Sentinel-2.** *Geophysical Research Letters* , 48, e2020GL092170.

<https://doi.org/10.1029/2020GL092170>

United States Naval Academy (2020). **Reflectance Spectra.** Retrieved from https://www.usna.edu/Users/oceano/pguth/md_help/html/ref_spectra.html

Xie, C.; Chen, P.; Pan, D.; Zhong, C.; Zhang, Z. Improved Filtering of ICESat-2 LiDAR Data for Nearshore Bathymetry Estimation Using Sentinel-2 Imagery. *Remote Sens.* 2021, 13, 4303.
<https://doi.org/10.3390/rs13214303>

มูลนิธิ สوان. 2559. *ภูมิศาสตร์เทคนิค Geographic Techniques*. พิมพ์ครั้งที่ 2. กรุงเทพฯ:
บริษัท ด่านสุทธาราการพิมพ์ จำกัด

สำนักงานพัฒนาเทคโนโลยีอวกาศและภูมิสารสนเทศ (องค์การมหาชน). 2552. *ตำราเทคโนโลยีอวกาศและ
ภูมิสารสนเทศศาสตร์*. กรุงเทพฯ : ออมรินทร์พรินติ้งแอนด์พับลิชิ่ง.