

การพัฒนา ออกแบบ และวิเคราะห์กล้องหลักดาวเทียม THEOS-Z โดยใช้สมการทางคณิตศาสตร์และ
โปรแกรมภาษาไพทอนเป็นเครื่องมือช่วยเหลือ

DEVELOPMENT, DESIGN AND ANALYSIS OF THEOS-Z SATELLITE IMAGING PAYLOAD BY
USING MATHEMATICAL EQUATIONS AND PYTHON PROGRAMMING AS A TOOL.

ศศิศ ศรีรัตน์

SASIT SRIRAT

Payload and OBDH Sub-System

กล้องถ่ายภาพหลักของดาวเทียม(Imaging Payload) ถือเป็นหัวใจหลักของดาวเทียมสำรวจโลกหรือ Earth observation satellite เป็นอย่างมาก เนื่องจากเป็นดาวเทียมที่ออกแบบมาเพื่อใช้สำหรับสำรวจโลก จากวงโคจร ซึ่งรวมถึงดาวเทียมสอดแนมและอื่น ๆ ที่ใช้สำหรับกิจการพลเรือนต่าง ๆ เช่น การตรวจสอบสิ่งแวดล้อม อุตุณิยมวิทยา การทำแผนที่ การสำรวจหาทรัพยากร และอื่น ๆ อีกมากมาย รวมทั้งดาวเทียมสำรวจโลก ถือเป็นดาวเทียมที่มีความสำคัญต่อประเทศชาติเป็นอย่างมาก เนื่องจากเป็นดาวเทียมที่ให้ความสำคัญต่อมนุษยชาติเป็นอันดับแรก รวมทั้งยังเป็นการพัฒนาบุคลากรในด้านต่างๆที่เกี่ยวข้อง ซึ่งนอกจากนี้ยังถือเป็นการสานสัมพันธ์ไมตรีกับนานาประเทศที่ให้ความร่วมมือ ซึ่งก่อให้เกิดการลงทุนในด้านต่าง ๆ จากทั้งในประเทศและนอกประเทศอีกด้วย

แนวทางการเลือกใช้งาน

เริ่มจากการได้รับ Heritage Baseline ที่ได้จากทางที่ปรึกษา ซึ่งต่อมาสมาชิกในกลุ่มได้ตัดสินใจเลือกดาวเทียม Carbonite-2 มาเป็นต้นแบบในการพัฒนา เนื่องจากมีคุณสมบัติหลายข้อที่สามารถตอบโจทย์ความต้องการของลูกค้า รวมทั้งเมื่อพิจารณาหลาย ๆ องค์ประกอบ ทำให้ทราบว่าดาวเทียม Carbonite-2 เหมาะที่จะเป็นดาวเทียมต้นแบบสำหรับโครงการ THEOS-Z และไม่ยากเกินไปนักสำหรับการพัฒนาดาวเทียมเพื่อให้สามารถใช้งานได้อย่างเต็มประสิทธิภาพตามความต้องการของลูกค้า

ปัญหาที่พบ

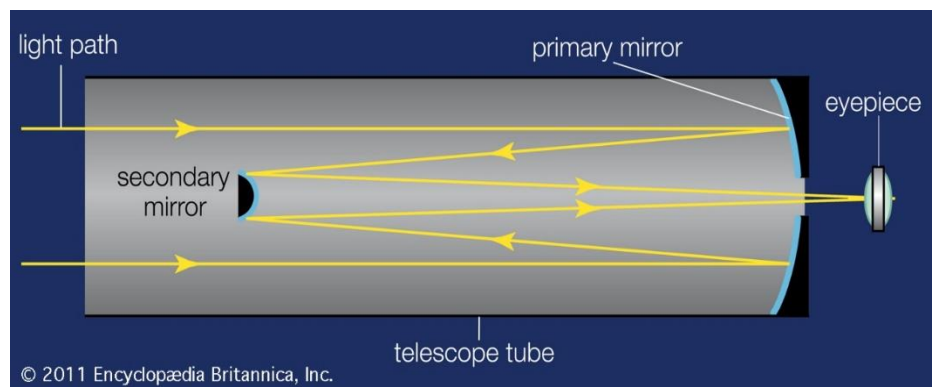
เนื่องจากการที่เลือก Heritage Baseline ของดาวเทียม Carbonite-2 ทำให้เกิดปัญหาตามมาหลายอย่าง ซึ่งเมื่อตรวจสอบและวิเคราะห์แล้ว พบว่าปัญหาใหญ่ในนั้น คือปัญหาที่อยู่ในระบบ Imaging Payload โดยที่ค่า Ground Sample Distance (GSD) และค่า Coverage Area นั้นไม่สามารถทำงานได้ตามความต้องการของลูกค้า ซึ่งในความต้องการของลูกค้าถูกระบุไว้ว่า

1. The coverage area shall be greater than 6.5 km x 6.5 km
2. The native Ground Sample Distance (GSD) of individual pixels of the imager shall be better than 1 m

แนวทางการพัฒนาและแก้ไข

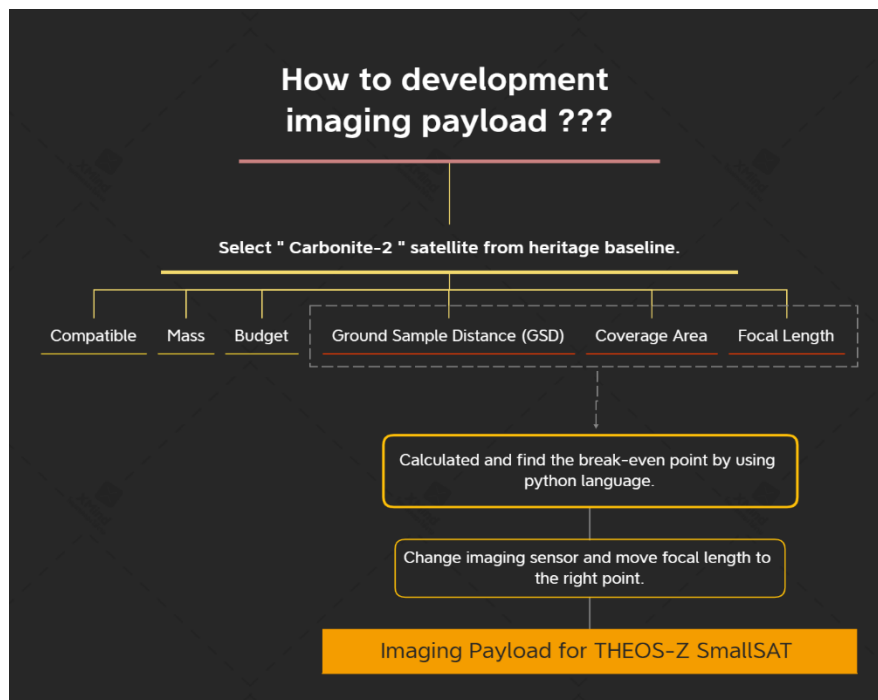
จากการที่ได้เข้าปรึกษาหาแนวทางแก้ไขและพัฒนากล้องดาวเทียมกับทางทีมที่ปรึกษาด้าน Payload System ทำให้ได้วิธีการพัฒนากล้องและแก้ไขที่ตรงจุด ซึ่งที่ปรึกษากล่าวว่าสามารถขยับเลนส์ Secondary Mirror ออกจาก Primary Mirror ได้ เพื่อให้เกิดระยะ Focal Length ที่สามารถทำงานได้ในระดับความสูงต่าง ๆ รวมทั้งสามารถแก้ไขและเปลี่ยนแปลง Pixel Sensor (อยู่ตำแหน่งเดียวกันกับ Eyepiece) เพื่อให้สามารถทำงานครอบคลุมระยะ Coverage Area ที่เป็นความต้องการของลูกค้าได้

หลังจากการปรึกษากับที่ปรึกษา ทำให้เห็นว่าต้นแบบดาวเทียม Carbonite-2 ที่เลือกมานี้ยังสามารถใช้งานได้สำหรับโครงการนี้ โดยเพียงแต่ต้องพัฒนาเพิ่มขึ้น



เพื่อให้สามารถทำงานตามความต้องการของลูกค้า โดยหลักการพัฒนาและแก้ไขกล้องของดาวเทียมดวงนี้้อย่างแรก นั่นคือการคำนวณหาค่าจากสมการทางฟิสิกส์และคณิตศาสตร์ ซึ่งค่าคำตอบของเลขที่ออกมา นั้นจะทำให้ทราบและสามารถแก้ไขกล้องของดาวเทียมได้อย่างถูกต้อง

ขั้นตอนการแก้ไขปัญหา



ขั้นตอนการพัฒนาและแก้ไขปัญหาล้องของดาวเทียม

จากสมการคำนวณต่าง ๆ ที่สามารถอ้างอิงได้จากบทเรียน “THEOS-2 Online Basic Course” ซึ่งสามารถหาคำตอบสำหรับการแก้ไขได้จากสมการ $GSD = \frac{Altitude \times Pixel Size}{Focal Length}$ และสมการ $C.A = GSD \times N$ โดยให้ทั้งหมดอยู่ในหน่วย เมตร (m) เพื่อให้ง่ายต่อการคำนวณ โดยที่ GSD คือ Ground Sample Distance ส่วน C.A คือ Coverage Area และ N คือ Number of Pixel หรือ Pixel Size ในแนวนอน X และแกน Y

โดยสิ่งต่อมาที่ใช้ในการช่วยเหลือสำหรับการคำนวณหาค่าต่าง ๆ จากสมการ โดยการเลือกใช้โปรแกรม Pycharm ซึ่งเป็นโปรแกรมจำพวก Integrated Development Environment (IDE) ที่ใช้ภาษา Python เป็นภาษาหลักของโปรแกรม เหมาะกับการพัฒนาและหาค่าตัวเลขต่าง ๆ ได้อย่างแม่นยำ รวมถึงสามารถหาค่าความน่าจะเป็นต่าง ๆ และสามารถวิเคราะห์ข้อมูลออกมาเป็นรูปแบบกราฟได้ ทำให้ง่ายต่อการวิเคราะห์และพิจารณา โดยการคำนวณหาค่าคำตอบนี้ ใช้คู่กับ Library อย่าง Matplotlib และ Numpy ซึ่งเป็น Library ที่สามารถนำมาใช้ได้อย่างถูกกฎหมาย อีกทั้งยังสามารถวิเคราะห์ค่าต่าง ๆ ได้อย่างละเอียดและแม่นยำ

ขั้นตอนการดำเนินงาน

1. เริ่มจากการสร้างสมการในโปรแกรม ซึ่งหลังจากการตรวจสอบความถูกต้องของไวยากรณ์แล้ว กำหนดค่าเริ่มต้นของโปรแกรม โดยให้ Pixel size เป็น 5.3×10^{-6} um หรือ 0.0000053 m และใช้ค่า Number of Pixels ในแนวนอน $X = 7200$ px เพื่อให้ง่ายต่อการคำนวณ (อ้างอิงค่าต่าง ๆ จากการใช้ Image Sensor ของกล้อง Pentax 645Z)

```
import matplotlib.pyplot as plt
from numpy import array as np
from pandas import DataFrame
from random import randint

numbera = [] #focal
numberb = [] #alt
numberc = [] #gsd
numberd = [] #altd
numbere = []
numberf = []

#Pixel size and Sensor
pxp = 0.0000053 #Refer from Canon EOS 50SR
size = 7200.00

#Focal Length
gsd = float(input("Maximum gsd = ? : "))
alt = float(input("Minimum Altitude = ? : "))
print("F.Length \t\tAltitude")
for i in range(31):
    n = 10000
    f = ((alt+(n+1)) * pxp / gsd)
    altd = (alt + (n + 1))
    print("%.3f m \t\t%.3f m" % (f,altd))
    #ค่า f
    a = float((f))
    numbera.append(a)
    b = float((altd))
    numberb.append(b)

#GSD and Coverage Area
fol = float(input("focal length fix = ? : "))
alt = float(input("altitude = ? : "))
print("GSD \t\tAltitude \t\tCoverage Area")
for i in range(31):
    n = 10000
    altd = (alt - (n + 1))
    gsd = ((alt-(n+1)) * pxp / fol)
    carea = gsd*size
    print("%.3f m \t\t%.3f m \t\t%.3f km" % (gsd,altd,carea,carea))
    #ค่า gsd
    c = float((gsd))
    numberc.append(c)
    d = float((altd))
    numberd.append(d)
    e = float((carea))
    numbere.append(e)

#print(numberc) #gsd
#print(numberd) #altd
#print(numberc[0:5]) #carea
#print('Min number = %d' % min(numberc=6500))
print("")

plt.xlabel("Focal Length (m)")
plt.ylabel("Altitude (m)")
plt.scatter(numbera, numberb, c='n', label='Data')
plt.plot(numbera, numberb)
plt.legend()
plt.grid()
plt.show()
```

โครงสร้างของโปรแกรมคำนวณที่ถูกเขียนบนภาษา Python

2. Run โปรแกรม โดยให้ค่าเริ่มต้นของ GSD เป็น 1.00m (อ้างอิงจาก Mission Requirement) และกำหนดค่า Altitude เริ่มต้นไว้ที่ 500 km หรือ 500000 m เพื่อให้สามารถหาค่า Focal Length ที่ดีที่สุดของระยะได้

Maximum gsd(m) = ? : 1.00	Minimum Altitude(m) = ? : 500000		
F.Length	Altitude		
0.) 2.658 m	500000.000 m	20.) 3.710 m	700000.000 m
1.) 2.783 m	510000.000 m	21.) 3.763 m	710000.000 m
2.) 2.756 m	520000.000 m	22.) 3.816 m	720000.000 m
3.) 2.809 m	530000.000 m	23.) 3.869 m	730000.000 m
4.) 2.862 m	540000.000 m	24.) 3.922 m	740000.000 m
5.) 2.915 m	550000.000 m	25.) 3.975 m	750000.000 m
6.) 2.968 m	560000.000 m	26.) 4.028 m	760000.000 m
7.) 3.021 m	570000.000 m	27.) 4.081 m	770000.000 m
8.) 3.074 m	580000.000 m	28.) 4.134 m	780000.000 m
9.) 3.127 m	590000.000 m	29.) 4.187 m	790000.000 m
10.) 3.180 m	600000.000 m	30.) 4.240 m	800000.000 m
11.) 3.233 m	610000.000 m		
12.) 3.286 m	620000.000 m		
13.) 3.339 m	630000.000 m		
14.) 3.392 m	640000.000 m		
15.) 3.445 m	650000.000 m		
16.) 3.498 m	660000.000 m		
17.) 3.551 m	670000.000 m		
18.) 3.604 m	680000.000 m		
19.) 3.657 m	690000.000 m		
20.) 3.710 m	700000.000 m		
21.) 3.763 m	710000.000 m		
22.) 3.816 m	720000.000 m		
23.) 3.869 m	730000.000 m		
24.) 3.922 m	740000.000 m		

ผลลัพธ์ที่ได้จากการคำนวณเมื่อโปรแกรมทำงาน

3. หลังจากหาค่าของ Focal Length ได้แล้ว ใส่ Altitude เริ่มต้นและใส่ค่าของ Focal Length ลงไป เพื่อหา GSD ที่จะคลาดเคลื่อนเมื่อ Altitude ลดระดับลงมา รวมทั้งได้ค่า Coverage Area ที่จะลดลงเช่นกัน เมื่อระดับ Altitude ลดลง

Set focal length (m) = ? : 2.725			5.) 0.909 m	500000.000 m	6545.455 km x 6545.455 km
Start Altitude(m) = ? : 550000			6.) 0.891 m	490000.000 m	6414.545 km x 6414.545 km
			7.) 0.873 m	480000.000 m	6283.636 km x 6283.636 km
			8.) 0.855 m	470000.000 m	6152.727 km x 6152.727 km
			9.) 0.836 m	460000.000 m	6021.818 km x 6021.818 km
			10.) 0.818 m	450000.000 m	5890.909 km x 5890.909 km
			11.) 0.800 m	440000.000 m	5760.000 km x 5760.000 km
			12.) 0.782 m	430000.000 m	5629.091 km x 5629.091 km
			13.) 0.764 m	420000.000 m	5498.182 km x 5498.182 km
			14.) 0.745 m	410000.000 m	5367.273 km x 5367.273 km
			15.) 0.727 m	400000.000 m	5236.364 km x 5236.364 km
			16.) 0.709 m	390000.000 m	5105.455 km x 5105.455 km
			17.) 0.691 m	380000.000 m	4974.545 km x 4974.545 km
			18.) 0.673 m	370000.000 m	4843.636 km x 4843.636 km
			19.) 0.655 m	360000.000 m	4712.727 km x 4712.727 km
			20.) 0.636 m	350000.000 m	4581.818 km x 4581.818 km
			21.) 0.618 m	340000.000 m	4450.909 km x 4450.909 km
			22.) 0.600 m	330000.000 m	4320.000 km x 4320.000 km
			23.) 0.582 m	320000.000 m	4189.091 km x 4189.091 km
			24.) 0.564 m	310000.000 m	4058.182 km x 4058.182 km
			25.) 0.545 m	300000.000 m	3927.273 km x 3927.273 km
			26.) 0.527 m	290000.000 m	3796.364 km x 3796.364 km
			27.) 0.509 m	280000.000 m	3665.455 km x 3665.455 km
			28.) 0.491 m	270000.000 m	3534.545 km x 3534.545 km
			29.) 0.473 m	260000.000 m	3403.636 km x 3403.636 km
			30.) 0.455 m	250000.000 m	3272.727 km x 3272.727 km
			=====Develop for Gistda Project 2021=====		

ผลลัพธ์ของค่า GSD, Altitude และ Coverage Area เมื่อได้จากการรันโปรแกรมเสร็จสิ้น

ผลลัพธ์เชิงตัวเลข

จากการคำนวณของโปรแกรม ทำให้สามารถหาค่า Focal Length ได้ที่ระยะ **2.915m** โดยที่ค่าของ GSD นั้นตั้งไว้อยู่ที่ระยะ **1.00m** และค่าของ Altitude ตั้งไว้ที่ระยะ **550000m** ตามที่ทางกลุ่มได้วิเคราะห์ และเลือกไว้

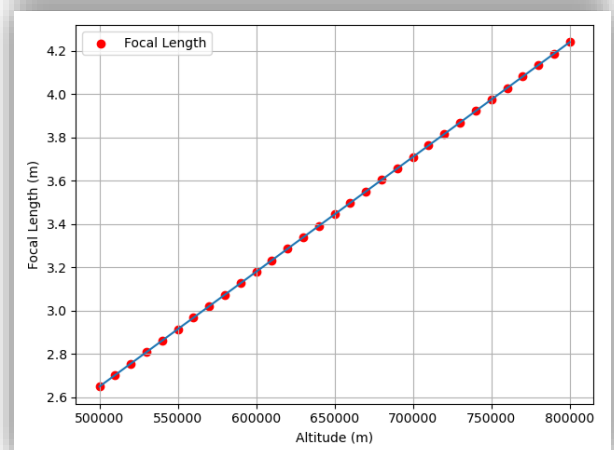
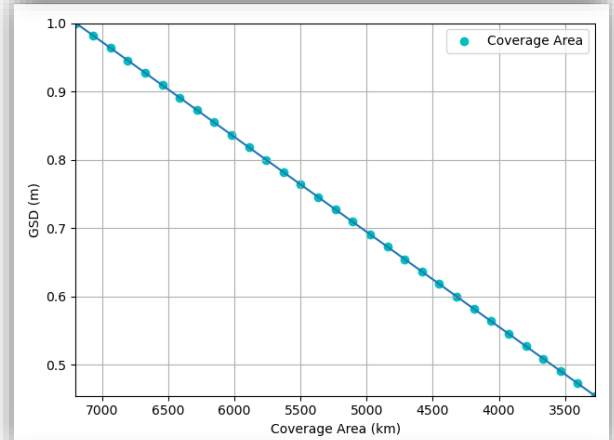
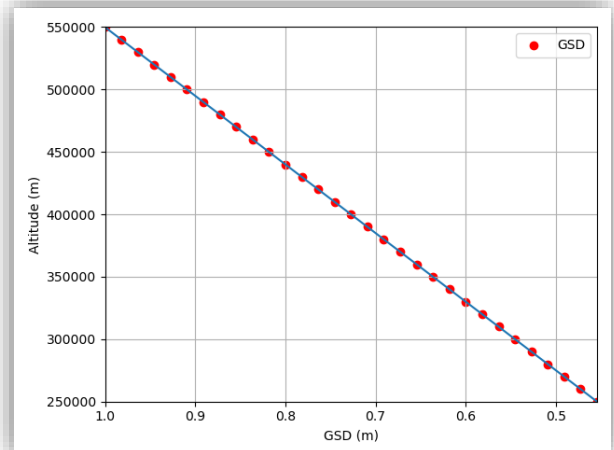
หลังจากที่ได้ค่า Focal Length ทำให้สามารถหาค่า GSD ที่ระยะเริ่มต้นของ Altitude ได้ โดยสามารถคาดการณ์ได้ว่าค่า GSD จะลดลงตาม Altitude ที่เท่าใด รวมทั้งสามารถหาค่า Coverage Area ที่ยังสามารถทำงานครอบคลุมได้ โดยจากตารางด้านล่าง เมื่อผ่านระดับ Altitude ที่ประมาณ **500000m** ลงไป ค่า Coverage Area จะไม่สามารถใช้งานครอบคลุมที่ระยะ **6.5km x 6.5km** ได้ แต่กระนั้นก็ถือว่าได้ผ่านระยะความต้องการของลูกค้าแล้ว ซึ่งถูกกำหนดไว้ที่ระยะ Altitude ระหว่าง 500km ถึง 800km

GSD	Altitude	Coverage Area
1.) 1.000 m	550000.000 m	7200.000 m x 7200.000 m
2.) 0.982 m	540000.000 m	7069.091 m x 7069.091 m
2.) 0.964 m	530000.000 m	6938.182 m x 6938.182 m
3.) 0.945 m	520000.000 m	6807.273 m x 6807.273 m
4.) 0.927 m	510000.000 m	6676.364 m x 6676.364 m
5.) 0.909 m	500000.000 m	6545.455 m x 6545.455 m
6.) 0.891 m	490000.000 m	6414.545 m x 6414.545 m
7.) 0.873 m	480000.000 m	6283.636 m x 6283.636 m
8.) 0.855 m	470000.000 m	6152.727 m x 6152.727 m
9.) 0.836 m	460000.000 m	6021.818 m x 6021.818 m
10.) 0.818 m	450000.000 m	5890.909 m x 5890.909 m

ตารางแสดงความสัมพันธ์ระหว่าง GSD , Altitude และ Coverage Area

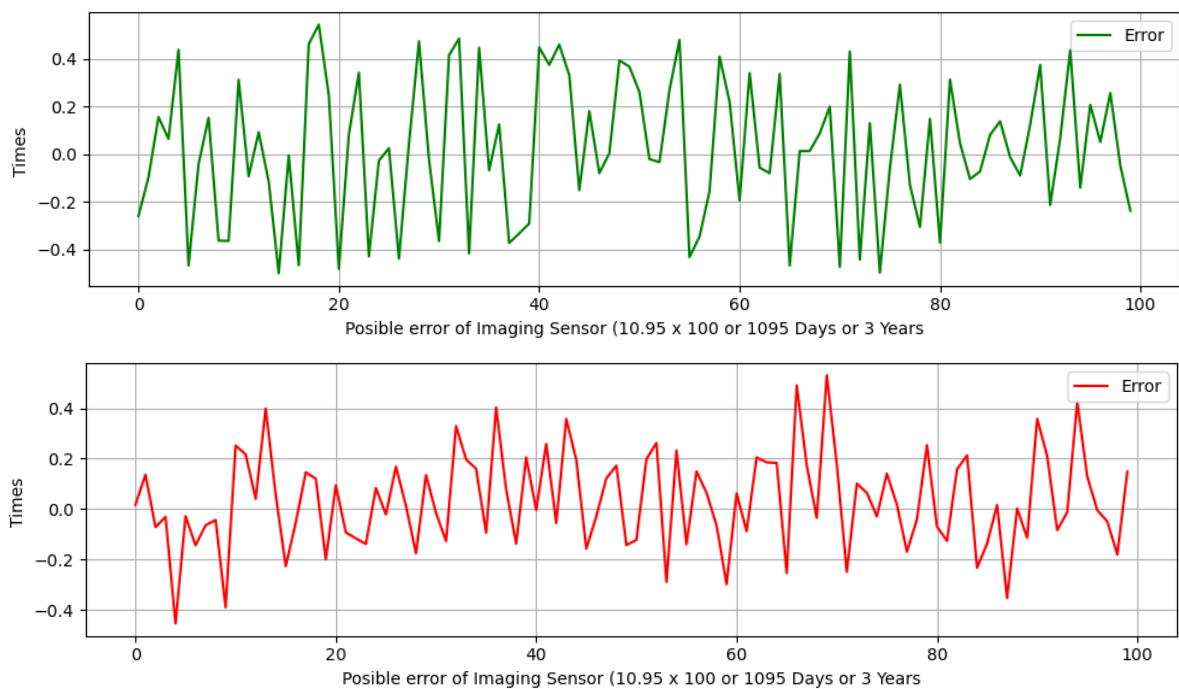
ผลลัพธ์เชิงกราฟ

1. จากการวิเคราะห์ภาพกราฟ GSD ที่ได้จากโปรแกรม ทำให้เห็นความสัมพันธ์ระหว่างค่า Altitude (แกน Y) และ GSD (แกน X) ว่า “ยิ่ง Altitude อยู่ในระดับที่สูง GSD ก็จะมียิ่งสูง” ซึ่งกราฟในข้อนี้ ได้ให้คำตอบเชิงประมาณว่า ระยะความสูงของดาวเทียมส่งผลต่อค่า GSD ที่ทำให้ 1 จุดพิกเซลต่อพื้นที่ตารางเมตร มีค่ามากขึ้น เมื่อดาวเทียมลดระยะลงมาใกล้โลกมากขึ้น
2. จากการวิเคราะห์ภาพกราฟ Coverage Area ที่ได้จากโปรแกรม ทำให้เห็นความสัมพันธ์ระหว่างค่า GSD (แกน Y) และ Coverage Area (แกน X) ว่า “ยิ่งค่า GSD สูง ค่า Coverage Area ก็จะมีสูงตาม” ซึ่งแปรผันตรงกัน โดยหมายความว่า เมื่อดาวเทียมลดระยะลงมา จะส่งผลให้ภาพที่ถ่ายได้นั้น มีค่า Coverage Area ที่ไม่มากพอ ซึ่งหมายความว่าเมื่ออยู่ใกล้พื้นโลกมากขึ้น ขนาดที่ครอบคลุมก็จะเล็กลงเรื่อย ๆ เมื่อดาวเทียมลดระยะลงมา
3. จากการวิเคราะห์ภาพกราฟ Focal Length ที่ได้จากโปรแกรม ทำให้เห็นความสัมพันธ์ระหว่างค่า Focal Length (แกน Y) และ Altitude (แกน X) ว่า “ยิ่งค่า Altitude อยู่ต่ำ ค่า Focal Length ก็จะมีต่ำด้วยเช่นกัน” ซึ่งหมายความว่า เมื่อดาวเทียมอยู่ในระยะ Altitude ที่ต่ำ จะส่งผลให้ระยะโฟกัสทำงานได้น้อยลง



ปัจจัยความเสี่ยงและความน่าจะเป็น

เนื่องจากการเลือกใช้ Sensor ที่เลือกเข้ามาใหม่ ดังนั้นจึงต้องทำการวิเคราะห์ความเสี่ยงและความน่าจะเป็นที่อาจจะเกิดข้อผิดพลาดระหว่างการทำงานบนอวกาศ และในกรณีเช่นนี้จะหาได้ด้วยการทำการสุ่มค่าที่หลาย ๆ ครั้งเพื่อหาคำตอบ โดยใช้การสุ่มแบบ Uniform และการสุ่มแบบ Triangular ซึ่งทั้งสองชนิดเป็นการสุ่มหาค่ากลาง โดยการกำหนดให้ค่าความน่าจะเป็นอยู่ที่ -0.5 ถึง 0.5 ซึ่งหมายถึงว่าเหตุการณ์ที่อาจจะเกิด เท่ากับ X เขียนเป็นสมการได้ $-0.5 < X < 0.5$ ซึ่งถ้าหากค่าน้อยกว่าหรือเท่ากับ -0.5 หมายถึงโอกาสมีน้อยที่จะเกิดข้อผิดพลาด และถ้าหากค่ามากกว่าหรือเท่ากับ 0.5 หมายความว่ามีโอกาสเกิดขึ้นได้



จากกราฟด้านบนจะเห็นได้ว่า กราฟเขียว ที่เกิดจากการสุ่มแบบ Uniform มีค่าเฉลี่ยเท่ากับ **0.01995049957987912** ซึ่งถือได้ว่ามีโอกาสเกิดขึ้นน้อยมากสำหรับเหตุการณ์ที่จะเกิดข้อผิดพลาดบน Imaging Sensor และจากการสุ่มแบบที่สอง กราฟด้านล่างเส้นสีแดง ที่เกิดจากการสุ่มแบบ Triangular ซึ่งมีค่าเฉลี่ยที่มีโอกาสเกิดข้อผิดพลาดเพียงแค่ **0.03411791747702883** เท่านั้น ซึ่งค่าคำตอบทั้งสองถือว่าการเลือกใช้ Sensor นั้น อาจจะไม่เกิด Error ระหว่างการใช้งาน 3 ปีเลยก็เป็นไปได้

ตัวอย่างค่าตัวเลขที่เกิดจากการสุ่มค่าหาความน่าจะเป็น 100 ครั้ง

[-0.25907688 -0.09756457 0.15605504 0.06464816 0.43821825 -0.46694751
-0.04148662 0.15285237 -0.36252349 -0.3637092 0.3123761 -0.09247944
0.09261565 -0.11422496 -0.49945816 -0.00486795 -0.46616467 0.46252794
0.54442611 0.24762397 -0.48059035 0.08185796 0.34233946 -0.42822946
-0.02647041 0.0253068 -0.43856713 0.04477258 0.47426074 -0.02284545
-0.36410138 0.41589498 0.48542586 -0.41594562 0.44681992 -0.06783303
0.12515862 -0.37242392 -0.33254687 -0.29082274 0.4482436 0.37519639
0.46021461 0.33167758 -0.1510122 0.18042303 -0.07926091 0.00399711
0.39390167 0.36747996 0.26155663 -0.02029644 -0.03320154 0.26885561
0.47979979 -0.43201187 -0.34568305 -0.15471588 0.41010726 0.21981978
-0.1934742 0.34099502 -0.05591444 -0.07987862 0.33717129 -0.46686375
0.01297894 0.01352779 0.08801504 0.20009537 -0.47284313 0.43157619
-0.44213989 0.13014348 -0.49725781 -0.05835705 0.29248958 -0.12892186
-0.30466802 0.14850624 -0.37102053 0.31286857 0.04576304 -0.10422512
-0.0720542 0.08149456 0.13836342 -0.01204514 -0.08909371 0.12360701
0.37482219 -0.21274001 0.06676776 0.43694593 -0.14002202 0.20727843
0.05134101 0.25708743 -0.04754871 -0.23711194]

โดยที่ค่าเฉลี่ย หรือ Mean = 0.01995049957987912

เอกสารอ้างอิง

Tananiti Promwongsa, *Optics and Imager Electronics Introduction Course*

. <http://www.saintlouis.or.th>. 2021. แหล่งที่มา : <http://training.gistda.or.th/theos-2/online/courses>. ค้นเมื่อ 10 สิงหาคม, 2564.

Digital Camera Database, *Pentax 645z* . <http://www.saintlouis.or.th>. 2021.

แหล่งที่มา : https://www.digicamdb.com/specs/pentax_645z/. ค้นเมื่อ 15 สิงหาคม, 2564.

The Editors of Encyclopaedia Britannica, *Cassegrain Telescope* .

<https://www.britannica.com>. 2021.

แหล่งที่มา : <https://www.britannica.com/science/Cassegrain-reflector>. ค้นเมื่อ 14 สิงหาคม, 2564.

Surrey, *Carbonite-2 : Launcher* . <https://www.sstl.co.uk/>. 2021.

แหล่งที่มา : <https://www.sstl.co.uk/space-portfolio/launched-missions/2010-2019/carbonite-2-launched-2018>. ค้นเมื่อ 19 สิงหาคม, 2564.

Herbert J. Kramer, *Carbonite* . <https://directory.eoportal.org>. 2021.

แหล่งที่มา : <https://directory.eoportal.org/web/eoportal/satellite-missions/c-missions/carbonite>. ค้นเมื่อ 01 สิงหาคม, 2564.

Gunter Dirk Krebs, *Carbonite 2 (CBTN2/VividX2)* . <https://space.skyrocket.de/>. 2021.

แหล่งที่มา : https://space.skyrocket.de/doc_sdat/carbonite-2.htm.

ค้นเมื่อ 05 สิงหาคม, 2564.