**สมการและวิธีการคำนวณ**

Position and velocity update

ในการจำลองการเคลื่อนที่ของหุ่นยนต์ด้วยโปรแกรม MATLAB เราใช้ คำสั่ง

* figure เพื่อสร้างหน้าจอ figure
* axes เพื่อกำหนดขนาดของ axis
* line เพื่อวาดหุ่นยนต์เป็นรูปวงกลม

ในการจะทำให้หุ่นยนต์ของเราขยับได้นั้น จะต้องทำการ update ตำแหน่งของหุ่นยนต์ตลอดเวลาที่เรากดปุ่มลูกศรบนคีย์บอร์ด เพราะฉะนั้นเราจึงกำหนด ตำแหน่งเริ่มต้นของหุ่นยนต์เป็น position (0,0) ความเร็วเริ่มต้นของหุ่นยนต์เป็น velocity=0 และมุมเริ่มต้น angle=0 จากนั้นเราจะสร้าง Loop และกำหนดตำแหน่งใหม่จะเกิดจาก

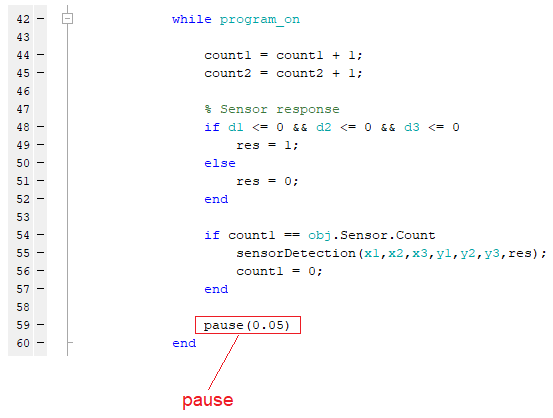
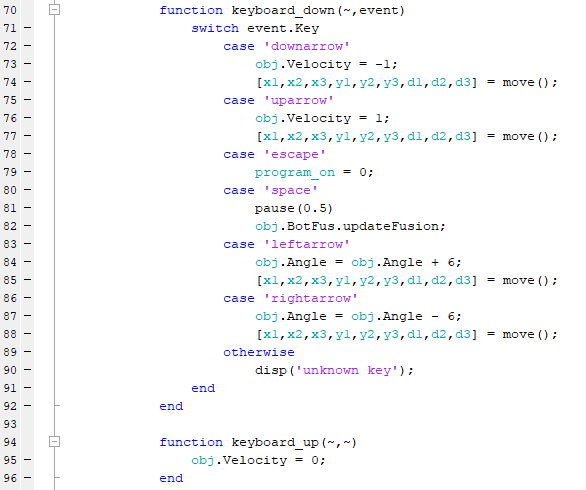
New\_position = [Positionx+Positiony]+[Vx,Vy]

โดยที่ Vx = velocity+cos(angle) และ Vy = velocity+sin(angle)

แล้วกำหนดลูกศรทั้ง 4 ดังนี้

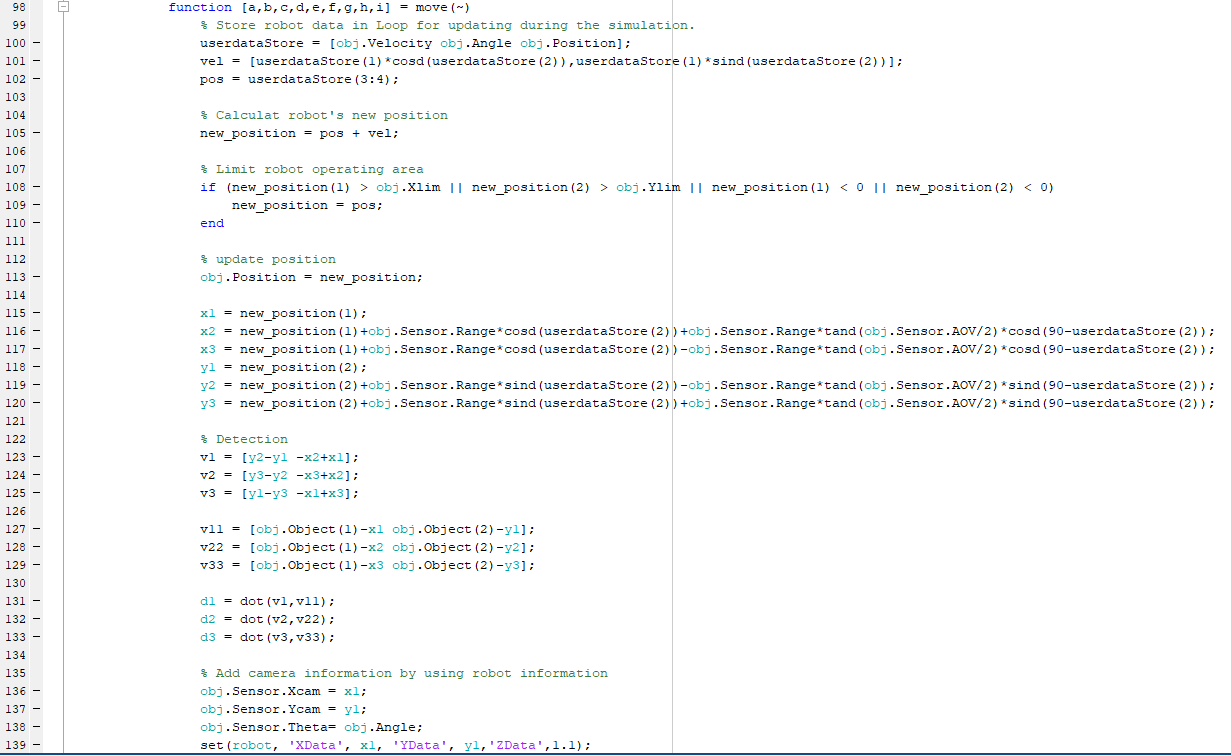
* Up arrow จะทำให้ velocity=1 m/pause\*
* Down arrow จะทำให้ velocity=-1 m/pause\*
* Left arrow จะทำให้ angle ใหม่ = angle+6 deg
* Right arrow จะทำให้ angle ใหม่ = angle-6 deg

pause\* ในส่วนนี้เนื่องจากในการ Loop เราจะตั้งค่าเวลาที่ใช้หยุดเพื่ออัพเดทการเคลื่อนที่ของหุ่นยนต์ (โดยการ plot ใหม่ทุกๆ ครั้งหลัง pause เช่น ถ้า pause=0.05 s เราจะได้ความเร็ว velocity=1 m/0.05 s หรือ velocity=20 m/s )

Keyboard function

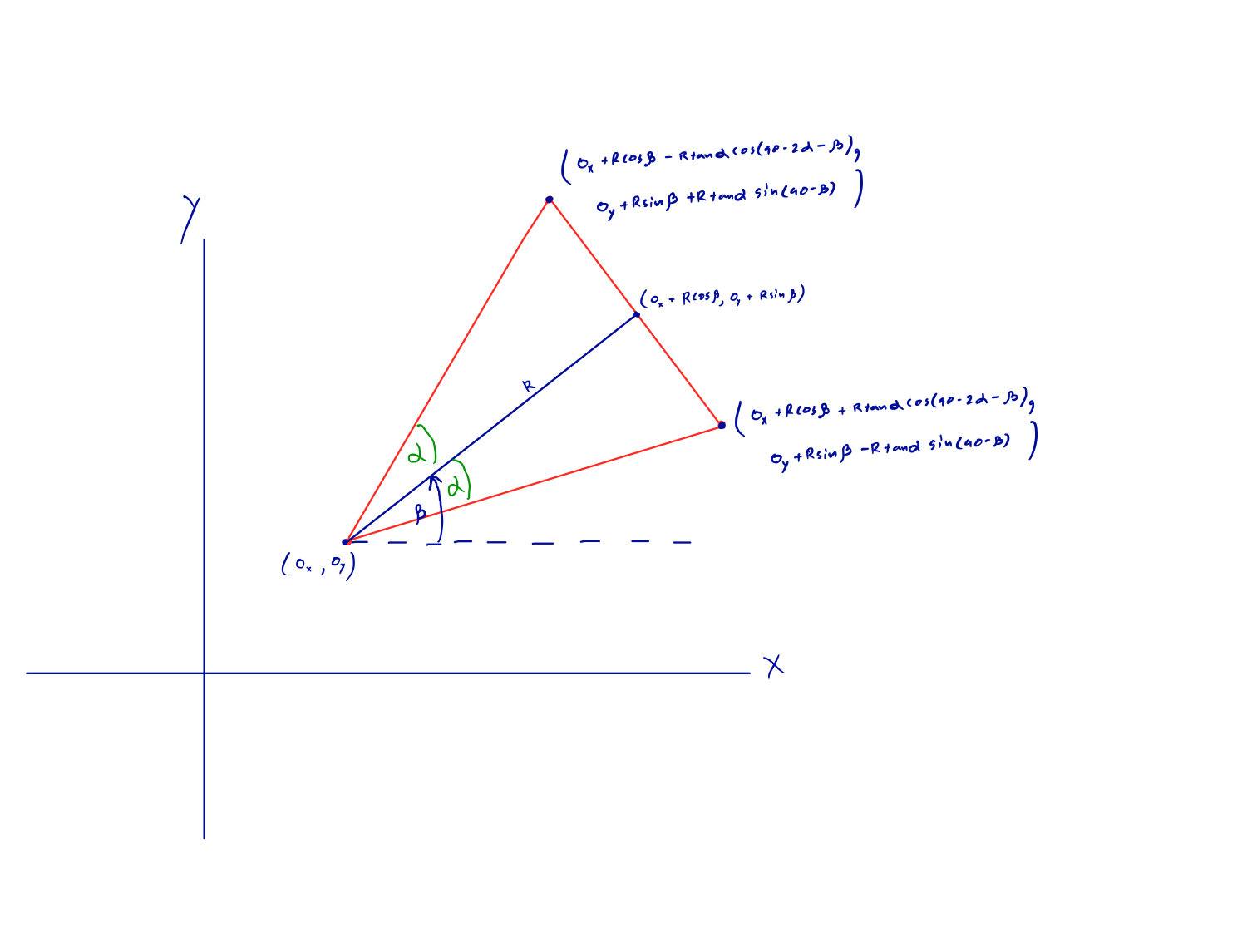
Pause ที่อยู่ใน Loop



Function ในการ plot ตำแหน่ง Robot ใหม่

Triangle Plot

ในการสร้างสามเหลี่ยมเราจะกำหนดจุดยอดของสามเหลี่ยมทั้งสามมุม โดยสามารถคำนวณจุดยอดได้ตามรูปด้านล่าง



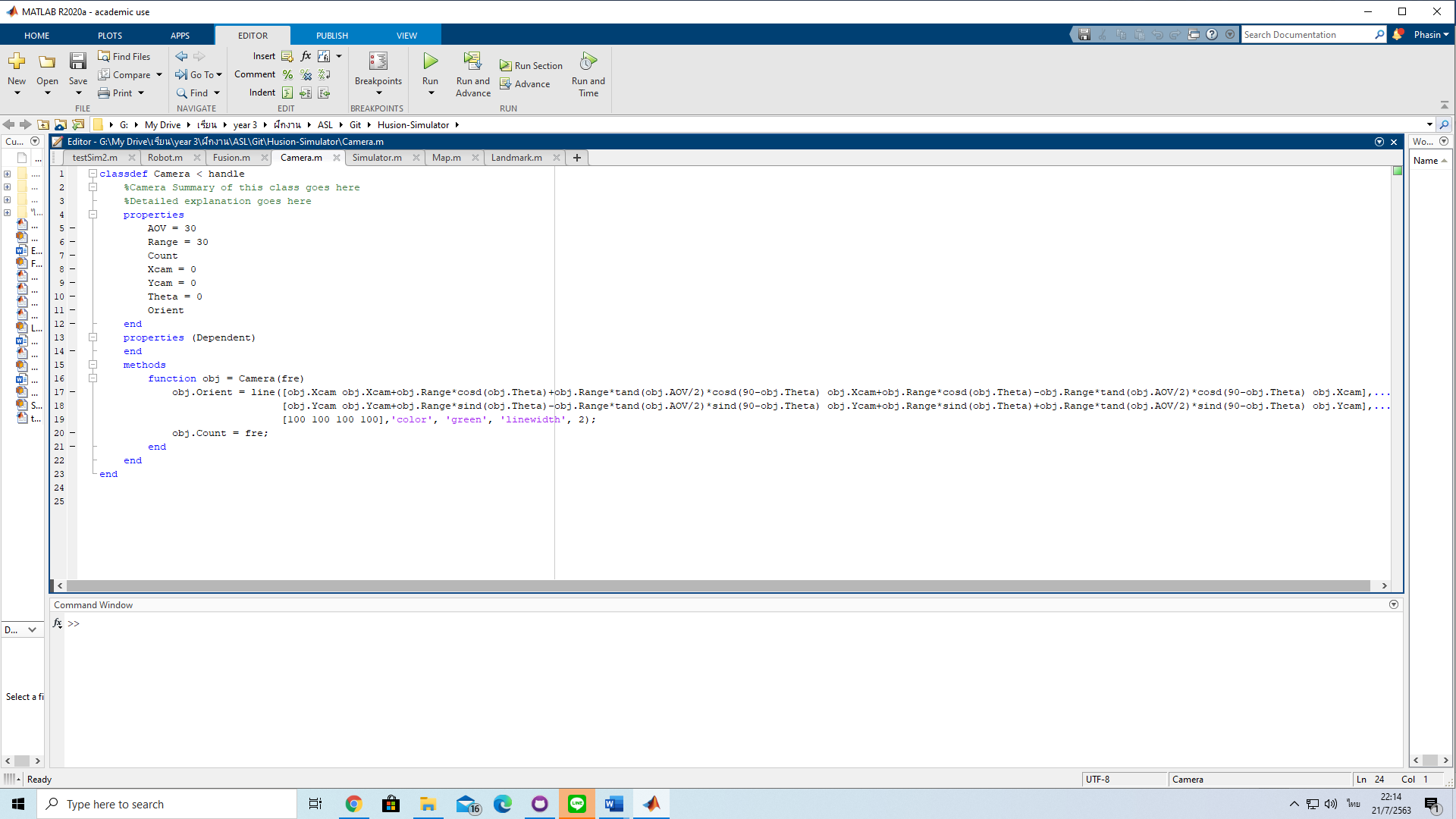
โดยกำหนดให้

* (ox, oy) คือ ตำแหน่งของหุ่นยนต์
* R คือ ระยะที่สามารถส่องถึงของกล้อง (Range) (meter)
* α คือ ครึ่งหนึ่งของ Angle of view (degree)
* β คือ มุมระหว่างแกนกลางของกล้องและแกน x (degree)

โดยที่จุดที่ 1 สามารถหาได้ดังนี้

และจุดที่ 2 สามารถหาได้ดังนี้

นำสมการข้างต้นมาใช้ใน Class Camera เพื่อสร้าง Area ที่กล้องมองเห็นให้กับหุ่นยนต์ ดังนี้



Bayes Theorem

P(B|A) = P(A|B) \* P(B) / P(A)

แต่สำหรับโปรเจคนี้เราจะไม่คำนวณ P(A) เนื่องจากมีความยากในการคำนวณ และไม่ได้มีผลต่อการคาดเดาความน่าจะเป็นของ B มากนัก ดังนั้นเราจะคำนวณค่า P(A) ในภายหลัง โดยเราจะลดรูป Bayes Theorem เป็น

P(B|A) α P(A|B) \* P(B)

prior

likelihood

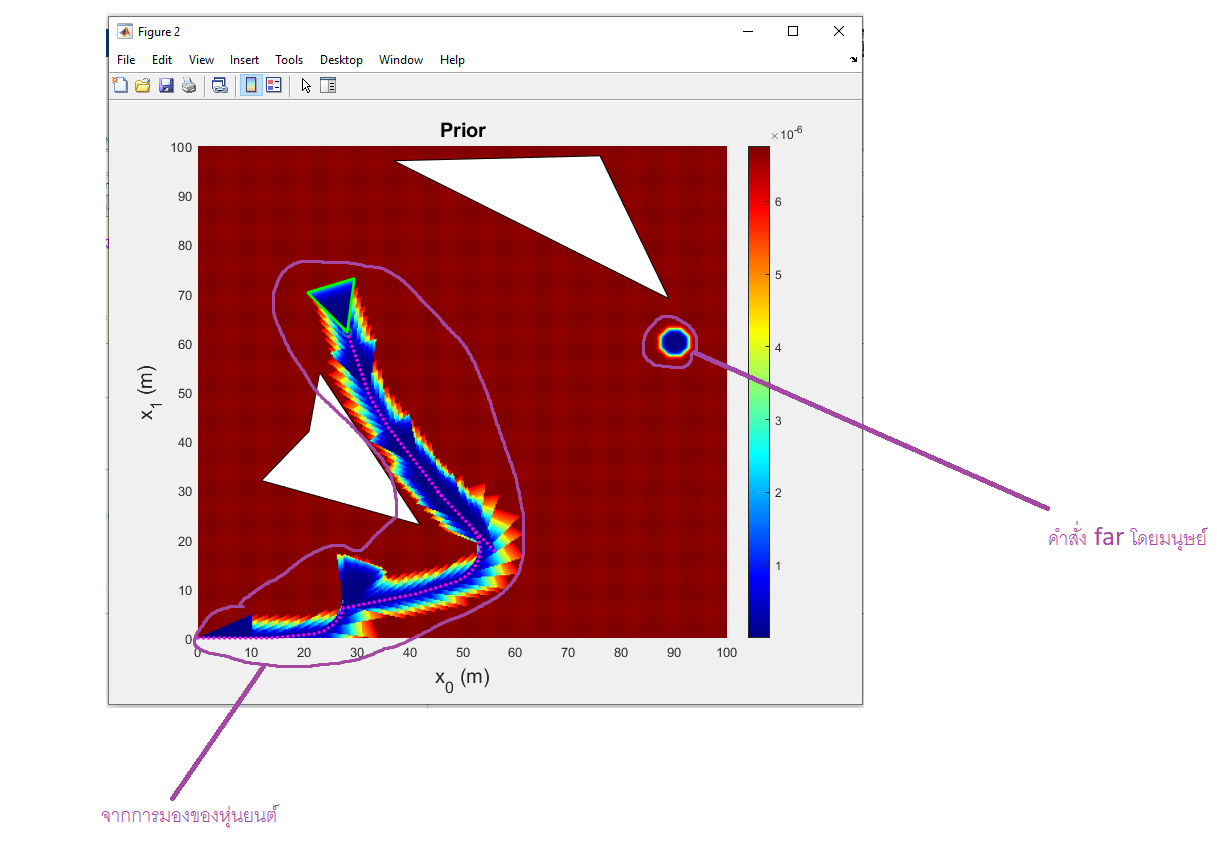
posterior

เราจะนำสมการนี้มาช่วยคำนวณหาความน่าจะเป็นที่จะพบหรือไม่พบวัตถุในบริเวณที่หุ่นยนต์กำลังค้นหา จากนั้นเราจะนำค่า P(A) ซึ่งเป็นค่าความน่าจะเป็นมากที่สุดหลังการคำนวณมาทำการ Normalization ในภายหลัง

โดยในการ update ข้อมูล likelihood เราสามารถทำได้ 2 วิธี คือ

1. ป้อนข้อมูลจากมนุษย์ด้วย 3 เงื่อนไข คือ near, far และ at
2. ข้อมูลที่อ่านได้จากกล้องของหุ่นยนต์

อ่านเรื่อง Bayes Theorem เพิ่มเติมได้ที่ <https://datarockie.com/2019/01/21/ep1-inverse-probability/#:~:text=Bayes%20Theorem,B%7CA)%20%E0%B9%84%E0%B8%94%E0%B9%89%E0%B9%80%E0%B8%A5%E0%B8%A2%E0%B8%87%E0%B9%88%E0%B8%B2%E0%B8%A2%E0%B9%86>

****