สำนักหอสมุดกลาง สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ระบบนำทางรถยนต์และส่วนบุคคล

CAR AND PERSONAL NAVIGATOR

ประสิทธิ์ พุฒพิริยะ

ไพสิฐ มั่นคงปรีชากุล

เลขหมู่ 83125
วัน,เดือน,ปี - 5 ส.ค. 2551

b 11959332
i

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2550

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2550

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบนำทางรถยนต์และส่วนบุคคล

Car and Personal Navigator

ผู้จัดทำ

1. ประสิทธิ์ พุฒพิริยะ รหัสนักศึกษา 48015347

2. ไพสิฐ มั่นคงปรีชากุล รหัสนักศึกษา 48015356

_____อาจารย์ที่ปรึกษา

(ผศ.ดร. วิศิษฏ์ หิรัญกิตติ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบนำทางรถยนต์และส่วนบุคคล

นาย ประสิทธิ์	พุฒพิริยะ	48015347
นาย ไพสิฐ	มั่นคงปรีชากุล	48015356
ผศ. ดร. วิศิษฏ์	หิรัญกิตติ	อาจารย์ที่ปรึกษา

ปีการศึกษา 2550

บทคัดย่อ

ในอนาคตอันใกล้ระบบนำทางจะเป็นอุปกรณ์สามัญประจำรถและประจำตัวบุคคลเพื่อทำให้ในการเดินทางให้มีประสิทธิภาพ ช่วยบอกเส้นทางวิธีการเดินทางไปยังสถานที่ต่างๆ ทั้งนี้รู้จักและไม่เคยไปมาก่อน โดยระบบนี้จะนำเอาวิธีการค้นหาเส้นทางอย่างชาญฉลาดมาประยุกต์ใช้เพื่อให้การเดินทางคุ้มค่าทางเศรษฐกิจมากที่สุดคือการแนะนำเส้นทางใกล้ที่สุด

ในโครงงานนี้ได้ทำการสร้างต้นแบบระบบนำทาง ประกอบด้วยส่วนแผนที่คอมพิวเตอร์ ส่วนการค้นหาเส้นทางที่ใกล้ที่สุด ส่วนทำหน้าที่แสดงตำแหน่งโดยมี GPS ซึ่งทำหน้าที่ระบุจุดพิกัด ณ. ตำแหน่งปัจจุบันของรถแล้วส่งไปยังส่วนวินิจฉัยเพื่อระบุตำแหน่งบนแผนที่ และส่วนการนำเส้นทางให้แก่รถบนแผนที่ในขณะรถวิ่งโดยใช้เสียงบอกนำเส้นทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Car and Personal Navigator

Mr. Prasit	Putpiriya	48015347
Ms. Paisit	Monkongprechagos	48015356
Asst. Prof. Dr. Visit	Hirankitti	Advisor

Academic Year 2007

ABSTRACT

In the near future a navigator system will be used in every car and by every person on the street to navigate efficiently their journey. It can suggest a route to reach a place never been visited before. This system adopts intelligent routing for economical traveling, i.e. guiding through a shortest route.

In this project a prototype of a navigation system is developed. It consists of digital maps, shortest-path routing part, a GPS position display part (which displays a car's current position based on the GPS data and sends this position to inference part) and the navigation part which navigates the car through a route via voices.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จได้อย่างดี ด้วยคำแนะนำและคำปรึกษาจาก ผศ. ดร. วิศิษฏ์ หิรัญกิตติ ซึ่งเป็นอาจารย์ที่ปรึกษาปริญญานิพนธ์ กลุ่มข้าพเจ้ารู้สึกซาบซึ้งในความอนุเคราะห์จากท่านอาจารย์และขอขอบพระคุณเป็นอย่างสูงมา ณ. โอกาสนี้

ขอกราบพระคุณคณาจารย์ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้กลุ่มข้าพเจ้า

ขอขอบพระคุณ คุณเกียรติศักดิ์ คิวขุนทด และ คุณสราญพงศ์ หนูยิ้มซ้าย จากห้องวิจัยการสื่อสารและคมนาคมชาญฉลาดภาควิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง สำหรับคำแนะนำและเทคนิคต่างๆ อันมีค่ายิ่งในปริญญานิพนธ์

ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ทุกคนในห้องวิจัยการสื่อสารและคมนาคมชาญฉลาดภาควิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ให้ความช่วยเหลือ และคำแนะต่างๆ รวมทั้งให้กำลังใจเสมอมา

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจ และให้การสนับสนุนในทุกๆเรื่อง ทำให้ข้าพเจ้า สามารถทำโครงงานนี้สำเร็จลุล่วงด้วยดี

คุณค่าและประโยชน์อันพึงมีจากโครงงานนี้ ข้าพเจ้าขอมอบแด่ผู้มีพระคุณทุกๆ ท่าน

ประสิทธิ์ พุฒพิริยะ

ไพสิฐ มั่นคงปรีชากุล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญตาราง VIII

สารบัญรูป IX

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญของปัญหา 1

1.2 ความมุ่งหมายและวัตถุประสงค์ ของการศึกษา 1

1.3 สมมุติฐานของการศึกษา 2

1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย 2

1.5 ขอบเขตการวิจัย 2

1.6 ขั้นตอนของการศึกษา 3

บทที่ 2 ทฤษฎี GIS (Geographic Information System) 4

2.1 บทนำ 4

2.2 ระบบสารสนเทศภูมิศาสตร์ 4

2.3 องค์ประกอบของ GIS (Components of GIS) 5

2.4 ระบบของ GIS 7

2.5 Spherical Coordinate System 7

2.5.1 ละติจูด (Latitude) 8

2.5.2 ลองติจูด (Longitude) 9

2.6 ระบบพิกัดกริดแบบ UTM 9

2.7 คุณลักษณะ Spherical Coordinate System และ Planar Coordinate System 11

2.7.1 Spherical Coordinate System 11

2.7.2 Planar Coordinate System (UTM) 11

2.8 ระบบแผนที่ 11

2.8.1 แผนที่ดิจิตอล 12

2.8.2 การคำนวณระยะทางบนแผนที่ 13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.8.3 ระยะทางระหว่างจุดสองจุดบนแผนที่ 13
2.8.4 ระยะทางระหว่างจุดถึงเส้นตรง 14
2.9 การระบุพิกัดตำแหน่งภาคพื้นดินด้วยดาวเทียมจีพีเอส 14
2.9.1 ระบบระบุตำแหน่งบนโลกด้วยดาวเทียมระบบจีพีเอส 14
2.9.2 หลักการทำงานของเครื่องรับสัญญาณจีพีเอส 19
2.9.3 ความคลาดเคลื่อนในการใช้งานจีพีเอส 21
2.9.4 ความคลาดเคลื่อนวงโคจรดาวเทียม 22
2.9.5 ความคลาดเคลื่อนนาฬิกาดาวเทียมและนาฬิกาเครื่องรับ 22
2.9.6 ความคลาดเคลื่อนเนื่องจากการหักเหในชั้นบรรยากาศ 22
2.9.7 คลื่นสะท้อน 22
2.9.8 ความคลาดเคลื่อนของเครื่องรับ 23
2.10 มาตรฐานในการสื่อสารข้อมูลจีพีเอส 23
บทที่ 3 กราฟฟิกไลบรารี OpenGL 25
3.1 บทนำ 25
3.2 OpenGL 25
3.3 ข้อดีของ OpenGL 26
3.4 รูปทรงเรขาคณิตพื้นฐาน (Geometric Primitives) 26
3.5 การแปลงทางเรขาคณิต 27
3.5.1 การเลื่อนตำแหน่ง (Translation) 27
3.5.2 การย่อ/ขยายภาพ (Scaling) 28
3.5.3 การหมุนภาพ (Rotation) 28
3.6 เมทริกซ์กับการแปลงทางเรขาคณิต 29
3.7 ภาพรวมแนวคิดการมองภาพ 3 มิติ 30
3.8 โปรเจ็กต์ชั่น (Projection) 30
3.9 โคออร์ดิเนตการมองภาพ 3 มิติ 32
3.10 การทำ Frustum Culling 32
3.11 ค่าที่ใช้คำนวณ Frustum Culling 33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

บทที่ 4 การออกแบบโครงสร้างของระบบ 37

4.1 การออกแบบโครงสร้างของระบบนำทางรถยนต์ 37

4.2 ส่วน GPS 37

4.3 ส่วนวินิจฉัย 38

4.3.1 การกำจัดสัญญาณรบกวน 39

4.3.2 รหัสลูกโซ่ (Chan Code) 40

4.3.3 วิธีการ Snap 44

4.4 ส่วนข้อมูล GIS 44

4.5 ส่วนพิกัดแผนที่ 46

4.6 ส่วนติดต่อผู้ใช้ 46

4.6.1 การแสดงผลแผนที่ 46

4.6.2 การค้นหาเส้นทาง 49

4.6.3 การนำทางผู้ใช้งาน 51

บทที่ 5 การทดลองและผลการทดลอง 52

5.1 การทดลอง GPS 52

5.2 การทดลอง Snap เส้นทาง 52

5.3 การทดลองแสดงภาพแผนที่โดยใช้ข้อมูลแผนที่ 53

5.4 การทดลองใช้อัลกอริทึม A* ในการหาเส้นทางที่สั้นที่สุด 54

5.5 การทดลองการนำเส้นทาง 54

5.6 การทดลองระบบในโทรศัพท์มือถือ N95 57

บทที่ 6 บทวิจารณ์และสรุป 58

6.1 บทสรุป 58

6.2 วิจารณ์สิ่งที่ได้จากโครงงาน 58

6.3 ปัญหาอุปสรรคและแนวทางแก้ไข 58

6.4 แนวทางการพัฒนาต่อ 58

บรรณานุกรม 59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

ภาคผนวก ก ข้อมูลอ้างอิงการโปรแกรมในโทรศัพท์ 60
ก.1 วิธีการลง Python SDK ใน S60 60
ก.1.1 ติดตั้งโปรแกรมภาษา Python บนคอมพิวเตอร์ผ่านตัว Emulator 60
ก.1.2 ติดตั้งโปรแกรมภาษา Python ลงบนโทรศัพท์มือถือ 61
ก.2 OpenGL ES API 62

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ **หน้า**

4.1 แสดงตาราง เปรียบเทียบค่ามุม ของรหัสลูกโซ่ 41
4.2 เปรียบเทียบองศากับทิศทางการเคลื่อนที่ 51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

รูปที่	หน้า
2.1 แสดงชั้นข้อมูลของระบบสารสนเทศภูมิศาสตร์ GIS	5
2.2 ลักษณะข้อมูลที่แสดงทิศทาง	6
2.3 แสดงองค์ประกอบทั้ง 5 ของ GIS	7
2.4 Spherical Coordinate System	8
2.5 การแบ่งกริดโซนระบบพิกัดกริด UTM	10
2.6 ตัวอย่างแผนที่ดิจิตอลจากโปรแกรม Map point Asia	12
2.7 ข้อมูลของแผนที่แบบราสเตอร์ และแบบเวคเตอร์	13
2.8 ตัวอย่างของแผนที่แบบราสเตอร์ และแบบเวกเตอร์	13
2.9 การคำนวณระยะห่างระหว่างจุดและเส้นตรงกับจุด	14
2.10 องค์ประกอบหลักของระบบ จีพีเอส	15
2.11 แสดงรหัสและคลื่นวิทยุที่ส่งจากดาวเทียมจีพีเอส	16
2.12 ดาวเทียม NAVSTAR	17
2.13 สถานีควบคุมภาคพื้นดิน	18
2.14 การใช้ประโยชน์ของดาวเทียมระบบจีพีเอส	19
2.15 การรับสัญญาณจากดาวเทียม 4 ดวง	20
2.16 ความคลาดเคลื่อนในการใช้งานจีพีเอส	21
3.1 สถาปัตยกรรมของ OpenGL	26
3.2 การอ้างอิงโคออร์ดิเนตสำหรับการมองของภาพ 3 มิติ	31
3.3 มุมมองของการมองภาพแบบเพอร์สเปกทีพ	31
3.4 ปริมาณ frustum กับค่าความสัมพันธ์ของมุมกล้อง	32
3.5 ตัวอย่างการแสดงวัตถุที่ตำแหน่งต่างๆ ของfrustum	33
3.6 ค่าพารามิเตอร์ต่างๆ ที่ใช้คำนวณ frustum(1)	34
3.7 ค่าพารามิเตอร์ต่างๆ ที่ใช้คำนวณ frustum(2)	35
3.8 ค่าพารามิเตอร์ต่างๆ ที่ใช้คำนวณ frustum(3)	35
4.1 ภาพแสดงโครงสร้างของระบบ	37
4.2 แสดงตำแหน่งของรถบนถนน	38
4.3 แสดงการเคลื่อนที่ของรถจาก P_1 ไป P_2	40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

รูปที่ **หน้า**

4.4 แสดงค่ามุมสำหรับรหัสลูกโซ่ 16 ทิศ 41

4.5 แสดงการเบี่ยงเบนของรหัสลูกโซ่ 43

4.6 การวิเคราะห์ค่าเบี่ยงเบน 43

4.7 แสดงตัวอย่างวิธีการ Snap 44

4.8 ตัวอย่างการแสดงผลข้อมูลแผนที่โดยใช้ OpenGL แบบ 2 มิติ 48

4.9 ตัวอย่างการแสดงผลข้อมูลแผนที่โดยใช้ OpenGL แบบ 3 มิติ 48

4.10 ส่วนประกอบ heuristic function f(n) 49

4.11 แสดงลักษณะของ function h(n) และ f(n) 50

5.1 การทดสอบการรับค่าของ GPS และ แปลงเป็นพิกัด (X, Y) 52

5.2 ผลเปรียบเทียบของการลดความผิดพลาดการรับค่าจาก GPS 53

5.3 แอปพลิเคชันที่ใช้แสดงแผนที่จากแบบเว็กเตอร์ 53

5.4 ตัวอย่างตัวอย่างการค้นหาเส้นทางสั้นที่สุดจาก พระจอมเกล้าลาดกระบัง ไป เดอะมอลล์บางกะปิ 54

5.5 การแสดงผลเมื่อทำการ Focus ไปยังตำแหน่งสัญญาณ GPS 55

5.6 การเลือกเส้นทางจากตำแหน่ง GPS ไปยังจุดบนแผนที่ 55

5.7 ภาพตัวอย่างแสดงการใช้งานระบบนำทาง 56

5.8 ภาพตัวอย่างการใช้งานบนโทรศัพท์มือถือ N95 57

ก.1 ภาพแสดงตัวเลือก Network card ที่ต้องการใช้ 60

ก.2 ภาพแสดงการเลือก Path ที่ต้องการติดตั้ง 61

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในอนาคตระบบนำทางควรเป็นอุปกรณ์สามัญประจำรถและประจำตัวบุคคล ซึ่งระบบนำทางนี้จะสามรถอำนวยความสะดวกให้กับผู้ขับรถยนต์ และผู้เดินทางได้ทราบถึงข้อมูลมี่สำคัญและเป็นประโยชน์ในการเดินทางซึ่งจะช่วยลดปัญหาต่างๆ ที่เกิดขึ้นในการเดินทางได้ ปัญหาที่พบในระหว่างการขับขี่รถยนต์และการเดินทางก็คือ การขับรถหลงเส้นทางโดยไม่ทราบว่าขณะนั้นขับรถอยู่ตรงส่วนใดของถนน ตำแหน่งปัจจุบันอยู่ห่างจากจุดหมายเท่าไหร่ ถนนเส้นใดที่ควรเดินทางในช่วงเวลาใดบ้าง ตรงไหนเป็นที่จอดรถ หรือมีปั๊มน้ำมันอยู่ตรงไหนบ้าง เป็นต้น ปัญหาดังกล่าวในปัจจุบันมักนิยมใช้แผนที่แบบแผ่นกระดาษชนิดแผ่นพับใช้ประกอบในขณะขับขี่ วิธีนี้อาจจะทำให้ไม่สะดวกในการใช้งานเพราะแผนที่มีขนาดใหญ่ และอาจจะเป็นสาเหตุในการเกิดอุบัติเหตุเกิดขึ้นได้ จากปัญหานี้จึงได้มีการจัดทำแผนที่แบบแผ่นกระดาษให้อยู่ในรูปของแผนที่คอมพิวเตอร์ โดยใช้คอมพิวเตอร์ ประมวลในรูปของการจัดเก็บข้อมูลแผนที่ เมื่อแผนที่อยู่ในรูปแบบของข้อมูลคอมพิวเตอร์แล้ว ผู้ใช้สามารถค้นหาข้อมูลต่างๆ ที่เกี่ยวกับแผนที่ด้วยการป้อนสิ่งที่ต้องการค้นหา เช่น ถนน ตรอก ซอย เส้นแบ่งเขต เป็นต้น รวมทั้งยังสามารถย่อหรือขยายภาพแผนที่ได้

จากการที่นำเอาแผนที่มาใส่ไว้ในคอมพิวเตอร์นั้น ในการใช้งานแผนที่อย่างเดียวคงไม่สะดวกมากนัก ตัวอย่างเช่น ไม่ทราบตำแหน่งปัจจุบันว่าอยู่ตรงส่วนไหนของแผนที่ การค้นหาเส้นทางจะต้องมีการป้อนทั้งตำแหน่งเริ่มต้นและตำแหน่งจุดหมาย เป็นต้น ดังนั้นถ้าเรานำเครื่องบอกตำแน่งมาใช้กับแผนที่จะทำให้เราทราบตำแหน่งปัจจุบันได้ ในส่วนของการค้นหาก็เพียงแค่ระบุตำแหน่งจุดหมายเพียงอย่างเดียวก็สามารถเดินทางได้ เครื่องบอกตำแหน่งที่นำมาใช้ในโครงงานนี้คือเครื่องรับ GPS (Global Positioning System) เมื่อนำระบบแผนที่ใน Pocket PC มาเชื่อมต่อกับเครื่องรับ GPS จะทำให้เกิดเป็นระบบที่เรียกว่า "ระบบนำทางรถยนต์และส่วนบุคคล (Car and Personal Navigator)"

1.2 ความมุ่งหมายและวัตถุประสงค์ ของการศึกษา

1. สร้างต้นแบบระบบนำทางที่ใช้ในรถยนต์และพร้อมเป็นระบบนำทางส่วนบุคคล
2. ออกแบบระบบนำทางที่อำนวยความสะดวกกับผู้ขับขี่รถยนต์และผู้เดินทาง โดยใช้วิธีการค้นหาเส้นทางที่ดีที่สุด ของปัญญาประดิษฐ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เพื่อศึกษาการวางแผนเส้นทางอย่างชาญฉลาด โดยเลือกการวางแผนเส้นทางแบบเส้นทางที่สั้นที่สุด

1.3 สมมุติฐานของการศึกษา

ระบบนำทางรถยนต์และส่วนบุคคลจะประกอบไปด้วย 3 ส่วนที่สำคัญคือ

1. การแสดงแผนที่จากข้อมูล GIS ซึ่งเป็นส่วนแสดงผลหลักของในการแสดงแผนที่พื้นที่ให้ผู้ใช้งานได้ทราบโดยข้อมูลของแผนที่จะมีรูปแบบต่างๆ ทั้งแบบ raster และ vector

2. การอ่านค่าและวิเคราะห์ค่าจาก GPS เป็นการรับสัญญาณจาก GPS ซึ่งเป็นสัญญาณจากดาวเทียมโดยมีลักษณะข้อมูล หรือ data ที่ส่งมาเป็นข้อความที่มีความหมาย โดยจะเลือกใช้ข้อความที่ใช้ระบุตำแหน่งปัจจุบันที่รับได้จาก GPS และทำการลดความผิดพลาดลงโดยใช้ algorithm ต่างๆ จากนั้นจึงจะสามารถนำใช้บอกตำแหน่งของผู้ใช้งานได้

3. ส่วนของการนำทางให้แก่ผู้ใช้งานเป็นส่วนที่ใช้ algorithm ในการหาเส้นทางจุดหมายปลายทางที่ผู้ใช้ต้องการเดินทางไป โดยวิเคราะห์เลือกใช้ระยะทางที่มีค่า Cost ต่ำที่สุด เพื่อเป็นการประหยัดทรัพยากรทั้งเงิน และ เวลา

1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย

1. นำหลักการวิธีดูทิศทางการเคลื่อนที่โดยใช้การคำนวณทาง Vector เพื่อหาทิศทางที่รถเคลื่อนไปและนำไปเปรียบเทียบกับทิศทางของเส้นทางที่ค้นหาได้และนำไปนำเส้นทางแก่ผู้ใช้งาน

2. นำหลักการของการมองภาพในสามมิติมาใช้งานเพื่อประยุกต์ใช้การควบคุมการมองเห็นของมุมกล้องและการแสดงผลของภาพ

3. นำวิธีการ Snap ซึ่งเป็นวิธีที่ใช้ในการหาจุดตั้งฉากบน path ของถนนที่ใกล้ที่สุดแล้วทำการ Projection ลงบน path นั้นทำให้จุดที่ Projection ลงบนถนนนั้นได้จุดใกล้ที่ใกล้เคียงความเป็นจริงมากที่สุดเพราะใช้ path จากข้อมูลของถนนจริง

1.5 ขอบเขตการวิจัย

1. วิเคราะห์และใช้งาน ข้อมูลแผนที่แบบดิจิตอล

2. ใช้เครื่องรับสัญญาณดาวเทียม GPS ในการทดสอบโดยมีค่าความผิดพลาดเชิงตำแหน่ง 5 - 15 เมตร และทำการใช้ algorithm ในการลดค่าความผิดพลาดของสัญญาณที่รับได้

3. ใช้หลักการของ ค้นหาเส้นทางที่ดีที่สุดกับการนำทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.6 ขั้นตอนของการศึกษา

1. ศึกษา ทฤษฎี "GIS ระบบสารสนเทศภูมิศาสตร์ (Geographic Information System)" ซึ่งเป็นพื้นฐานเกี่ยวกับแผนที่

2. ศึกษา การแปลความหมายจาก GPS Data

3. ศึกษา การนำ GPS Data ไปใช้งานกับระบบพิกัดแผนที่

4. ศึกษา อัลกอริธึม เอสตาร์ (A*) เพื่อใช้ในการหาเส้นทางที่ดีที่สุดจากโครงข่ายของเส้นถนน

5. ศึกษา OpenGL Canvas และการแสดงผลภาพสามมิติ

6. ศึกษาการกำจัดสัญญาณรบกวน (Filter Noise) ที่เกิดจากความคลาดเคลื่อนของตำแหน่ง GPS

7. ศึกษาวิธี การ Snap ตำแหน่งของรถบนเส้นถนน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎี GIS (Geographic Information System)

2.1 บทนำ

ระบบสารสนเทศภูมิศาสตร์ (Geographic Information System : GIS) หมายถึง เครื่องมือที่ใช้ระบบคอมพิวเตอร์ เพื่อใช้ในการนำเข้า จัดเก็บ จัดเตรียม ดัดแปลง แก้ไข จัดการ และวิเคราะห์ พร้อมทั้งแสดงผลข้อมูลเชิงพื้นที่ ตามวัตถุประสงค์ต่างๆ ที่ได้กำหนดไว้ ดังนั้น GIS จึงเป็นเครื่องมือที่มีประโยชน์เพื่อใช้ในการจัดการ และบริหารการใช้ทรัพยากรธรรมชาติและสิ่งแวดล้อม และสามารถติดตามการเปลี่ยนแปลงข้อมูลด้านพื้นที่ ให้เป็นไปอย่างมีประสิทธิภาพ เนื่องจากเป็นระบบที่เกี่ยวข้องกับระบบการไหลเวียนของข้อมูล และการผสานข้อมูลจากแหล่งข้อมูลต่างๆ เช่น ข้อมูล ปฐมภูมิ (primary data) หรือข้อมูลทุติยภูมิ (secondary data) เพื่อให้เป็นข่าวสารที่มีคุณค่า

2.2 ระบบสารสนเทศภูมิศาสตร์

ระบบสารสนเทศภูมิศาสตร์ (Geographic Information System : GIS) คือกระบวนการทำงานเกี่ยวกับข้อมูลในเชิงพื้นที่ด้วยระบบคอมพิวเตอร์ ที่ใช้กำหนดข้อมูลและสารสนเทศ ที่มีความสัมพันธ์กับตำแหน่งในเชิงพื้นที่ เช่น ที่อยู่ บ้านเลขที่ สัมพันธ์กับตำแหน่งในแผนที่ ตำแหน่งเส้นรุ้ง เส้นแวง ข้อมูลและแผนที่ใน GIS เป็นระบบข้อมูลสารสนเทศที่อยู่ในรูปของตารางข้อมูลและฐานข้อมูลที่มีส่วนสัมพันธ์กับข้อมูลเชิงพื้นที่ (Spatial Data) ซึ่งรูปแบบและความสัมพันธ์ของข้อมูลเชิงพื้นที่ทั้งหลาย จะสามารถนำมาวิเคราะห์ด้วย GIS และทำให้สื่อความหมายในเรื่องการเปลี่ยนแปลงที่สัมพันธ์กับเวลาได้ เช่น การแพร่ขยายของโรคระบาด การเคลื่อนย้าย ถิ่นฐาน การบุกรุกทำลาย การเปลี่ยนแปลงของการใช้พื้นที่ ฯลฯ ข้อมูลเหล่านี้ เมื่อปรากฏบนแผนที่ทำให้สามารถแปลและสื่อความหมาย ใช้งานได้ง่าย GIS เป็นระบบข้อมูลข่าวสารที่เก็บไว้ในคอมพิวเตอร์ แต่สามารถแปลความหมายเชื่อมโยงกับสภาพภูมิศาสตร์อื่นๆ สภาพท้องที่ สภาพการทำงานของระบบสัมพันธ์กับสัดส่วนระยะทางและพื้นที่จริงบนแผนที่ ข้อแตกต่างระหว่าง GIS กับ MIS นั้นสามารถพิจารณาได้จากลักษณะของข้อมูล คือ ข้อมูลที่จัดเก็บใน GIS มีลักษณะเป็นข้อมูลเชิงพื้นที่ (Spatial Data) ที่แสดงในรูปของภาพ (graphic) แผนที่ (map) ที่เชื่อมโยงกับข้อมูลเชิงบรรยาย (Attribute Data) หรือฐานข้อมูล (Database)การเชื่อมโยงข้อมูลทั้งสองประเภทเข้าด้วยกัน จะทำให้ผู้ใช้สามารถที่จะแสดงข้อมูลทั้งสองประเภทได้พร้อมๆกัน เช่นสามารถจะค้นหาตำแหน่งของจุดตรวจวัดควันดำ - ควันขาวได้โดยการระบุชื่อจุดตรวจ หรือในทางตรงกันข้าม สามารถที่จะสอบถามรายละเอียดของ จุดตรวจจากตำแหน่งที่เลือกขึ้นมา ซึ่งจะต่างจาก MIS ที่แสดง ภาพเพียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อย่างเดียว โดยจะขาดการเชื่อมโยงกับฐานข้อมูลที่เชื่อมโยงกับรูปภาพนั้น เช่นใน CAD (Computer Aid Design) จะเป็นภาพเพียงอย่างเดียว แต่แผนที่ใน GIS จะมีความสัมพันธ์กับตำแหน่งในเชิงพื้นที่ทางภูมิศาสตร์ คือค่าพิกัดที่แน่นอน ข้อมูลใน GIS ทั้งข้อมูลเชิงพื้นที่และข้อมูลเชิงบรรยายสามารถอ้างอิงถึงตำแหน่งที่มีอยู่จริงบนพื้นโลกได้โดยอาศัยระบบพิกัดทางภูมิศาสตร์ (Geo code) ซึ่งจะสามารถอ้างอิงได้ทั้งทางตรงและทางอ้อม ข้อมูลใน GIS ที่อ้างอิงกับพื้นผิวโลกโดยตรง หมายถึง ข้อมูลที่มีค่าพิกัดหรือมีตำแหน่งจริงบนพื้นโลกหรือในแผนที่ เช่น ตำแหน่งอาคาร ถนน ฯลฯ สำหรับข้อมูล GIS ที่จะอ้างอิงกับข้อมูลบนพื้นโลกได้โดยทางอ้อมได้แก่ ข้อมูลของบ้าน(รวมถึงบ้านเลขที่ ซอย เขต แขวง จังหวัด และรหัสไปรษณีย์) โดยจากข้อมูลที่อยู่ เราสามารถทราบได้ว่าบ้านหลังนี้มีตำแหน่งอยู่ ณ ที่ใดบนพื้นโลก เนื่องจากบ้านทุกหลังจะมีที่อยู่ไม่ซ้ำกัน

รูปที่ 2.1 แสดงชั้นข้อมูลของระบบสารสนเทศภูมิศาสตร์ GIS

2.3 องค์ประกอบของ GIS (Components of GIS)

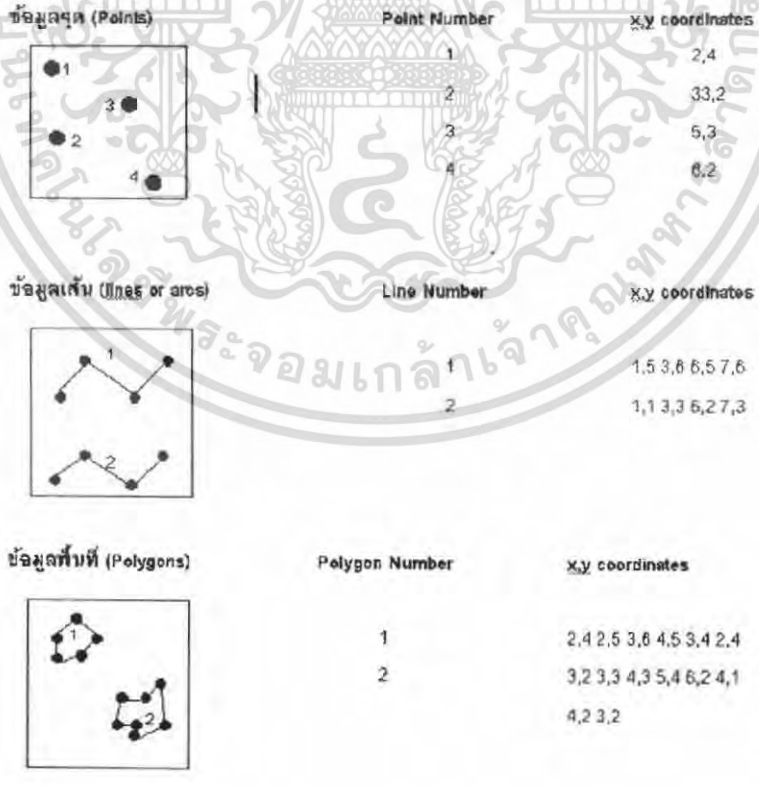
องค์ประกอบหลักของระบบ GIS จัดแบ่งออกเป็น 5 ส่วนใหญ่ ๆ คือ อุปกรณ์คอมพิวเตอร์ (Hardware) โปรแกรม (Software) ขั้นตอนการทำงาน (Methods) ข้อมูล (Data) และบุคลากร (People) โดยมีรายละเอียดของแต่ละองค์ประกอบดังต่อไปนี้

1. อุปกรณ์คอมพิวเตอร์ คือ เครื่องคอมพิวเตอร์รวมไปถึงอุปกรณ์ต่อพ่วงต่าง ๆ เช่น Digitizer, Scanner, Plotter, Printer หรืออื่น ๆ เพื่อใช้ในการนำเข้าข้อมูล ประมวลผล แสดงผล และผลิตผลลัพธ์ของการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. โปรแกรม คือ ชุดของคำสั่งสำเร็จรูป เช่น โปรแกรม Arc/Info, MapInfo ฯลฯ ซึ่งประกอบด้วยฟังก์ชั่น การทำงานและเครื่องมือที่จำเป็นต่าง ๆ สำหรับนำเข้าและปรับแต่งข้อมูล, จัดการระบบฐานข้อมูล, เรียกค้น, วิเคราะห์และจำลองภาพ

3. ข้อมูล คือ ข้อมูลต่าง ๆ ที่จะใช้ในระบบ GIS และถูกจัดเก็บในรูปแบบของฐานข้อมูลโดยได้รับการดูแล จากระบบจัดการฐานข้อมูลหรือ DBMS ข้อมูลที่จะนำเข้าระบบสารสนเทศทางภูมิศาสตร์ ควรเป็นข้อมูลเฉพาะเรื่อง (theme) และเป็นข้อมูลที่สามารถนำมาใช้ในการตอบคำถามต่างๆ ได้ตรงตามวัตถุประสงค์ เป็นข้อมูลที่มีความถูกต้องและเชื่อถือได้และเป็นปัจจุบันมากที่สุด อนึ่ง ข้อมูลหรือสารสนเทศสามารถแบ่งออกได้เป็น 2 ประเภท คือ ข้อมูลที่มีลักษณะเชิงพื้นที่ (spatial data) และข้อมูลอธิบายพื้นที่ (non-spatial data or attribute data) ข้อมูลเชิงพื้นที่ เป็นข้อมูลที่แสดงตำแหน่งที่ตั้งทางภูมิศาสตร์ (geo-referenced data) ของรูปลักษณ์ของพื้นที่ (graphic feature) ซึ่งมี 2 แบบ คือ ข้อมูลที่แสดงทิศทาง (vector data) และ ข้อมูลที่แสดงเป็นตารางกริด (raster data) ข้อมูลที่มีทิศทางประกอบด้วยลักษณะ 3 อย่าง คือ ใ ข้อมูลจุด (point) เช่น ที่ตั้งหมู่บ้าน โรงเรียน เป็นต้น ข้อมูลเส้น (arc or line) เช่น ถนน แม่น้ำ ท่อประปา เป็นต้น ข้อมูลพื้นที่ หรือเส้นรอบรูป (polygon) เช่น พื้นที่ป่าไม้ ตัวเมือง เป็นต้น ข้อมูลนี้จะเป็นองค์ประกอบที่สำคัญรองลงมาจากบุคลากร

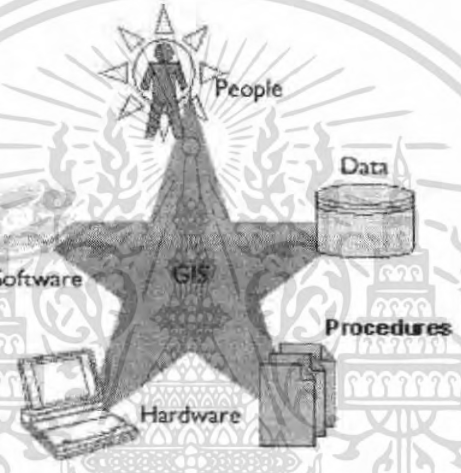


รูปที่ 2.2 ลักษณะข้อมูลที่แสดงทิศทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. บุคลากร คือ ผู้ปฏิบัติงานซึ่งเกี่ยวข้องกับระบบสารสนเทศภูมิศาสตร์ เช่น ผู้นำเข้าข้อมูล ช่างเทคนิค ผู้ดูแลระบบฐานข้อมูล ผู้เชี่ยวชาญสำหรับวิเคราะห์ข้อมูล ผู้บริหารซึ่งต้องใช้ข้อมูลในการตัดสินใจ บุคลากรจะเป็นองค์ประกอบที่สำคัญที่สุดในระบบ GIS เนื่องจากถ้าขาดบุคลากร ข้อมูลที่มีอยู่มากมายมหาศาลนั้น ก็จะไม่มีคุณค่าใดเลยเพราะไม่ได้ถูกนำไปใช้งาน อาจจะกล่าวได้ว่าถ้าขาดบุคลากรก็จะไม่มีระบบ GIS

5. วิธีการหรือขั้นตอนการทำงาน คือ วิธีการที่องค์กรนั้น ๆ นำเอาระบบ GIS ไปใช้งาน โดยแต่ละ ระบบแต่ละองค์กรย่อมีความแตกต่างกันออกไป ฉะนั้นผู้ปฏิบัติงานต้องเลือกวิธีการในการจัดการกับปัญหาที่เหมาะสมที่สุดสำหรับของหน่วยงานนั้น ๆ เอง



รูปที่ 2.3 แสดงองค์ประกอบทั้ง 5 ของ GIS

2.4 ระบบของ GIS

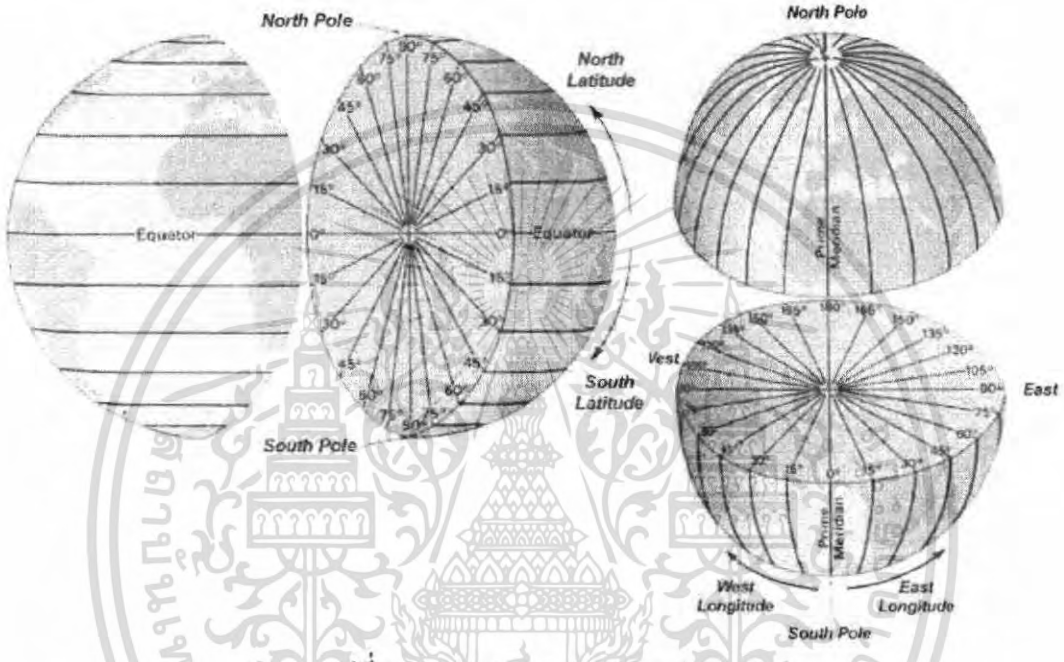
เนื่องจาก GIS เป็นระบบสารสนเทศที่รวบรวมข้อมูลเชิงพื้นที่ จากแหล่งข้อมูลและมาตราส่วนต่างๆ ดังนั้น ระบบค่าพิกัดทางภูมิศาสตร์ จึงเป็นสิ่งที่จำเป็น ทั้งนี้เพราะหากข้อมูลดังกล่าวใช้ระบบอ้างอิงที่แตกต่างกันจะทำให้ไม่สามารถซ้อนทับ หรือซ้อนทับได้แต่มีตำแหน่งที่ผิดพลาด หรือการเชื่อมต่อข้อมูลแผนที่ที่อยู่ใกล้เคียงกัน ไม่สามารถดำเนินการได้ ระบบค่าพิกัดภูมิศาสตร์สามารถแบ่งออกเป็น 2 ระบบใหญ่ๆ คือ Spherical Coordinate และ Map Projection

2.5 Spherical Coordinate System

Spherical Coordinate System เป็นระบบค่าพิกัดที่อ้างอิงเส้นรุ้ง (Latitude) และเส้นแวง (Longitude)โดยตั้งอยู่บนพื้นฐานว่าโลกมีลักษณะกลม ซึ่งเป็นภาพ 3 มิติ เส้นแวง (longitude or meridians) จะลากจากขั้วโลกเหนือมายังขั้วโลกใต้ เส้นแวง 0 องศา (Prime Meridian) จะลากผ่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมือง Greenwich ประเทศอังกฤษ ทางทิศตะวันออก จะมีค่า 0-180 องศา ส่วนทางทิศตะวันตกของเส้น Prime Meridian จะมีค่า 0-(-180) องศา เส้นรุ้ง บางครั้งเรียกว่า Parallels เนื่องจากจะมีระยะห่างที่เท่ากันตลอด เส้นรุ้งที่ลากผ่านเส้นศูนย์สูตร (equator) จะมีค่า 0 องศา เส้นรุ้งที่อยู่ทางทิศเหนือของเส้นศูนย์สูตร จะมีค่าจาก 0-90 องศา ถึงขั้วโลกเหนือ และเส้นรุ้งที่อยู่ทางทิศใต้ของเส้นศูนย์สูตร จะมีค่าจาก 0-(-90) องศา ถึงขั้วโลกใต้ ดังนั้น ค่าพิกัดหนึ่งๆ ของระบบเส้นรุ้ง เส้นแวงจะมีเพียงตำแหน่งเดียวบนพื้นโลก ดังรูปที่ 2.4



รูปที่ 2.4 Spherical Coordinate System

2.5.1 ละติจูด (Latitude)

1. เส้นรอยตัดบนพื้นผิวพิภพที่เกิดจากการสมมติใช้พื้นราบตัดพิภพโดยให้พื้นราบนั้นตั้งได้ฉากกับแกนหมุนของพิภพเสมอ เส้นรอยตัดดังกล่าวนั้นคือเส้นละติจูด นิยมเรียกสั้นๆ ว่า "เส้นขนาน"

2. ละติจูดศูนย์องศา คือ เส้นรอยตัดบนพื้นผิวพิภพ ที่เกิดจากพื้นราบที่ตั้งได้ฉากกับแกนหมุนตัดผ่านจุดศูนย์กลางของพิภพ เส้นรอยตัดเส้นนี้มีชื่อเรียกอีกอย่างหนึ่งว่า "เส้นศูนย์สูตร" (Equator) ซึ่งเป็นวงขนานละติจูดวงใหญ่ที่สุด

3. ค่าละติจูดของวงละติจูดใด คือ ค่ามุมที่จุดศูนย์กลางของพิภพนับไปตามพื้นราบที่บรรจุแกนหมุนของพิภพ เริ่มจากพื้นศูนย์สูตรถึงแนวเส้นตรงที่ลากจากจุดศูนย์กลางพิภพไปยังวงละติจูดนั้น

4. ที่จุดขั้วเหนือของพิภพมีค่าละติจูดเท่ากับ 90 องศาเหนือ และที่จุดขั้วใต้ของพิภพมีค่าละติจูดเท่ากับ 90 องศาใต้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เนื่องจากพื้นของวงละติจูดศูนย์องศา หรือพื้นศูนย์สูตร เป็นพื้นที่ตัดผ่านจุดศูนย์กลางของพิภพ วงศูนย์สูตรจึงถูกเรียกว่า "วงกลมใหญ่" ส่วนละติจูดอื่นๆ เป็นวงกลมเล็ก วงละติจูดจะมีขนาดเล็กลงๆ เมื่อห่างวงศูนย์สูตรออกไปจนกระทั่งกลายเป็นจุดที่ขั้วโลกเหนือ และขั้วโลกใต้

6. ระยะห่างระหว่างเส้นละติจูด 1 องศา คิดเป็นระยะทางบนผิวพิภพประมาณ 111 กิโลเมตร (69 ไมล์) และ 1 พิลิปดา มีระยะห่างประมาณ 30.48 เมตร (100 ฟุต)

2.5.2 ลองติจูด (Longitude)

1. เส้นรอยตัดบนพื้นผิวพิภพที่เกิดจากการสมมติใช้พื้นราบตัดพิภพ โดยให้พื้นราบผ่านแนวแกนหมุนของพิภพ เส้นรอยตัดบนพื้นผิวพิภพดังกล่าวเรียกว่าเส้นลองติจูด หรือเส้นเมริเดียน (Meridian)

2. ลองติจูดศูนย์องศา คือเส้นลองติจูดที่ผ่านหอส่งดาว ณ เมืองกรีนิส (Greenwich) ในประเทศอังกฤษ มีชื่อเรียกอีกชื่อหนึ่งว่า เมริเดียนหลัก (Prime Meridian)

3. การกำหนดค่าลองติจูด คือค่าง่ามมุมที่จุดศูนย์กลางพิภพบนพื้นศูนย์สูตร โดยใช้แนวเส้นตรงที่ลากจากจุดศูนย์กลางพิภพมายังเมริเดียนหลัก เป็นแนวเริ่มนับค่าง่ามมุมไปทางตะวันออก 180 องศา เรียกว่า ลองติจูดตะวันออก และนับค่าง่ามมุมไปทางตะวันตก 180 องศา เรียกว่า ลองติจูดตะวันตก เส้นลองติจูดที่ 180 องศาตะวันออกและตะวันตกเป็นเส้นเดียวกัน

4. ลองติจูดทุกเส้นเป็นส่วนโค้งของวงกลมใหญ่(Great Circle)

5. ระยะห่างระหว่างเส้นลองติจูด 1 องศา ตามเส้นศูนย์สูตร คิดเป็นระยะทางประมาณ 111 กิโลเมตร (69 ไมล์) และ 1 พิลิปดา มีระยะห่างประมาณ 30.48 เมตร (100 ฟุต) แต่เนื่องจากเส้นลองติจูดทุกเส้นจะไปบรรจบกันที่จุดขั้วเหนือและขั้วใต้ของพิภพ ดังนั้น ระยะห่างระหว่างเส้นลองติจูดจึงน้อยลงๆ เมื่อยิ่งห่างจากเส้นศูนย์สูตรออกไป

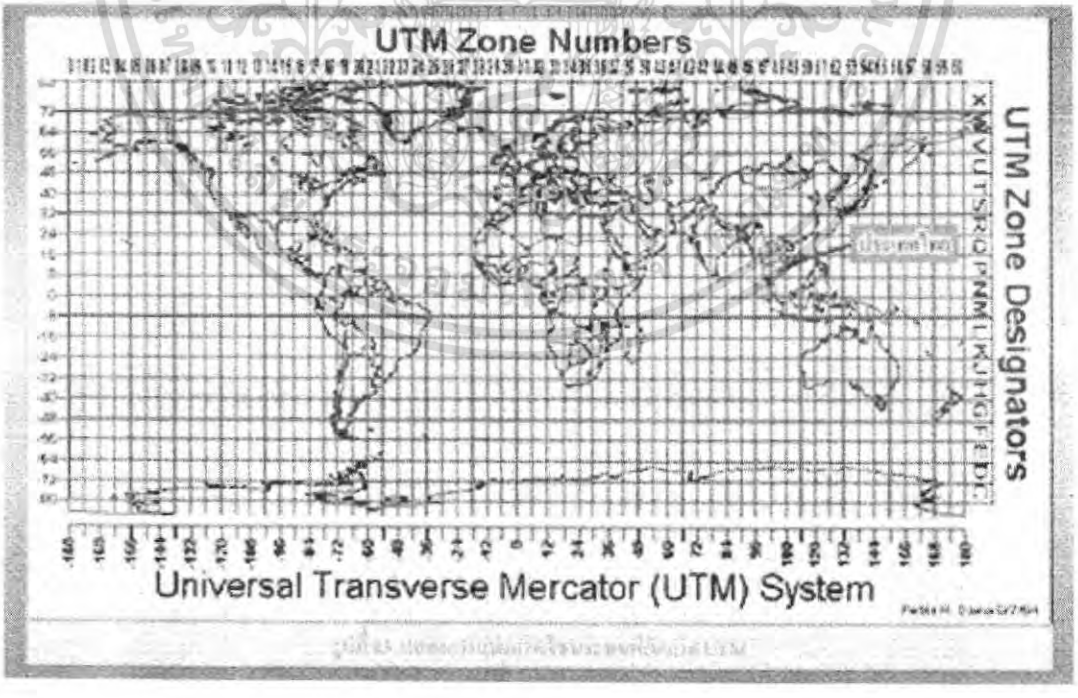
2.6 ระบบพิกัดกริดแบบ UTM

ระบบพิกัดกริดแบบ UTM (Universal Transverse Mercator co-ordinate System) เป็นระบบตารางกริดที่ใช้ช่วยในการกำหนดตำแหน่งและใช้อ้างอิงในการบอกตำแหน่ง ที่นิยมใช้กับแผนที่ในกิจการทหารของประเทศต่างๆ เกือบทั่วโลกในปัจจุบัน เพราะเป็นระบบตารางกริดที่มีขนาดรูปร่างเท่ากันทุกตารางและมีวิธีการกำหนดบอกค่าพิกัดที่ง่าย และถูกต้องเป็นระบบกริดที่นำเอาเส้นโครงแผนที่แบบ Universal Transverse Mercator Projection ของ Gauss-Krueger มาใช้ดัดแปลงการถ่ายทอดรายละเอียดของพื้นผิวโลกให้รูปทรงกระบอก Mercator Projection อยู่ในตำแหน่ง Mercator Projection (แกนของรูปทรงกระบอกจะทับกับแนวเส้นศูนย์สูตร และตั้งฉากกับแนวแกนของขั้วโลก) ประเทศไทยเราได้นำเอาเส้นโครงแผนที่แบบ UTM นี้มาใช้กับการทำแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นชุด L-7017 ที่ใช้ในปัจจุบันแผนที่ระบบพิกัดกริด ที่ใช้เส้นโครงแผนที่แบบ UTM เป็นระบบเส้นโครงชนิดหนึ่งที่ใช้ผิวรูปทรงกระบอกเป็นผิวแสดงเส้นเมริเดียน(หรือเส้นลองติจูด) และเส้นละติจูดของโลก โดยใช้ทรงกระบอกตัดโลกระหว่างละติจูด 84 องศาเหนือ และ 80 องศาใต้ ในลักษณะแกนรูปทรงกระบอกแล้วทำมุมกับแกนโลก 90 องศารอบโลก แบ่งออกเป็น 60 โซนๆ ละ 6 องศา

โซนที่ 1 อยู่ระหว่าง 180 องศา กับ 174 องศาตะวันตก และมีลองติจูด 177 องศาตะวันตก เป็นเมริเดียนย่านกลาง (Central Meridian) มีเลขกำกับแต่ละโซนจาก 1 ถึง 60 โดย นับจากซ้ายไปทางขวาระหว่างละติจูด 84 องศาเหนือ 80 องศาใต้ แบ่งออกเป็น 2 ช่อง ช่องละ 8 องศา ยกเว้นช่องสุดท้าย เป็น 12 องศา โดยเริ่มนับตั้งแต่ละติจูด 80 องศาใต้ ขึ้นไปทางเหนือ ให้ช่องแรกเป็นอักษร C และช่องสุดท้ายเป็นอักษร X (ยกเว้น I และ O) จากการแบ่งตามที่กล่าวแล้วจะเห็นพื้นที่ในเขตลองติจูด 180 องศาตะวันตก ถึง 180 องศาตะวันออก และละติจูด 80 องศาใต้ ถึง 84 องศาเหนือ จะถูกแบ่งออกเป็นรูปสี่เหลี่ยมผืนผ้า 1,200 รูป แต่ละรูปมีขนาดกว้างยาว 6 องศา x 8 องศา จำนวน 1,140 รูป และกว้างยาว 6 องศา x 12 องศา จำนวน 60 รูป รูปสี่เหลี่ยมนี้เรียกว่า Grid Zone Designation (GZD) การเรียกชื่อ Grid Zone Designation ประเทศไทยมีพื้นที่อยู่ ระหว่างละติจูด 5 องศา 30 ลิปดา เหนือ ถึง 20 องศา 30 ลิปดา เหนือ และลองติจูดประมาณ 97 องศา 30 ลิปดา ตะวันออก ถึง 105 องศา 30 ลิปดาตะวันออก ดังนั้น ประเทศไทยจึงอยู่ใน GZD 47N 47P 47Q 48N 48P และ 48 Q



รูปที่ 2.5 การแบ่งกริดโซนระบบพิกัดกริด UTM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 คุณลักษณะ Spherical Coordinate System และ Planar Coordinate System

2.7.1 Spherical Coordinate System

1. เป็นค่าพิกัดจริงโดยอ้างอิงจากผิวโลกพบเพียงตำแหน่งเดียว
2. เส้นแนวตั้ง (เส้นแวง) ไม่ขนานกัน แต่จะบรรจบกันที่ขั้วโลก เส้นแนวนอน (เส้นรุ้ง) จะขนาน กับบนพื้นโลก
3. ทิศเหนือของแผนที่ (cartographic north) แตกต่างจากทิศเหนือของขั้วโลก (pole)
4. หน่วยเป็นองศา ลิปดาและ ฟิลิปดา
5. ค่าพิกัดระบุเป็น เส้นรุ้ง (latitude) เส้นแวง (longitude)

2.7.2 Planar Coordinate System (UTM)

1. เป็นค่าพิกัดจริง อ้างอิงจากจุดสมมุติ พบหลายตำแหน่งบนพื้นโลก
2. พื้นที่แต่ละตารางกริดมีขนาดเท่ากัน เพราะพิจารณาในแนวราบ โลกถูกแบ่งออกเป็น 60 ส่วน ขนาดเท่ากัน ส่วนละ 6° latitude - แต่ละส่วนจะมีเส้นแนวตั้งสมมุติ ค่าที่ได้จะมีความถูกต้องมากในบริเวณ เส้นแวง +/- 60° เหนือและใต้
3. หน่วยเป็นเมตร แต่ค่าความถูกต้องอาจเป็นเซนติเมตร สามารถคำนวณหาพื้นที่ได้
4. ค่าพิกัดระบุเป็น X และ Y

2.8 ระบบแผนที่

แผนที่ที่ใช้อยู่ในปัจจุบันแบ่งได้เป็น 2 ชนิดใหญ่ ๆ คือแผนที่เฉพาะเรื่อง (Thematic map) และแผนที่ภูมิประเทศ (Topographic map) โดยแผนที่เฉพาะเรื่องนี้เป็นแผนที่ที่มีองค์ประกอบอื่น ๆ เข้ามามาก ส่วนแผนที่ภูมิประเทศจะเป็นแผนที่ที่เน้นแสดงสภาพทางภูมิศาสตร์โดยเฉพาะ

แผนที่เฉพาะเรื่อง คือแผนที่ที่แสดงรายละเอียดของข้อมูลเชิงพื้นที่ที่ต้องการนำเสนอ โดยทำการแปลงข้อมูลเหล่านั้นให้เป็นเครื่องหมายแผนที่เสียก่อน แล้วนำไปพิมพ์ซ้อนทับลงบนแผนที่ฐาน ตามตำแหน่งที่ตั้งของข้อมูลนั้นๆ ซึ่งประกอบไปด้วย ข้อมูลเชิงพื้นที่ (Spatial Data) และแผนที่ฐาน (Base map)ในการทำแผนที่นี้เมื่อเตรียมการเสร็จแล้วจะทำการพิมพ์ลงบนกระดาษ (Paper map) สำหรับปัญหาของแผนที่แบบกระดาษคือ ถ้ามีการเพิ่มเติมหรือแก้ไขข้อมูลจะไม่สามารถแก้ไขข้อมูลในเวลาสั้นๆได้ จะต้องทำการพิมพ์แผนที่ออกมาใหม่ทั้งหมด ทำให้เสียเวลาและค่าใช้จ่ายมาก ปัจจุบันนี้ได้ทำการดัดแปลงแผนที่เฉพาะเรื่อง มาจัดเก็บไว้ในคอมพิวเตอร์โดยนำข้อมูลไปเก็บไว้ในฐานข้อมูลคอมพิวเตอร์ มีการแสดงผลโดยการวางซ้อนทับกับฐานข้อมูล การนำคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เข้ามาใช้ในการจัดเก็บข้อมูลนี้จะทำได้ดีกว่าแผนที่กระดาษ เพราะว่าสามารถเลือกดูชั้นข้อมูลที่จำเป็นเท่านั้น ทำให้เข้าใจง่ายกว่าแผนที่กระดาษ

2.8.1 แผนที่ดิจิตอล

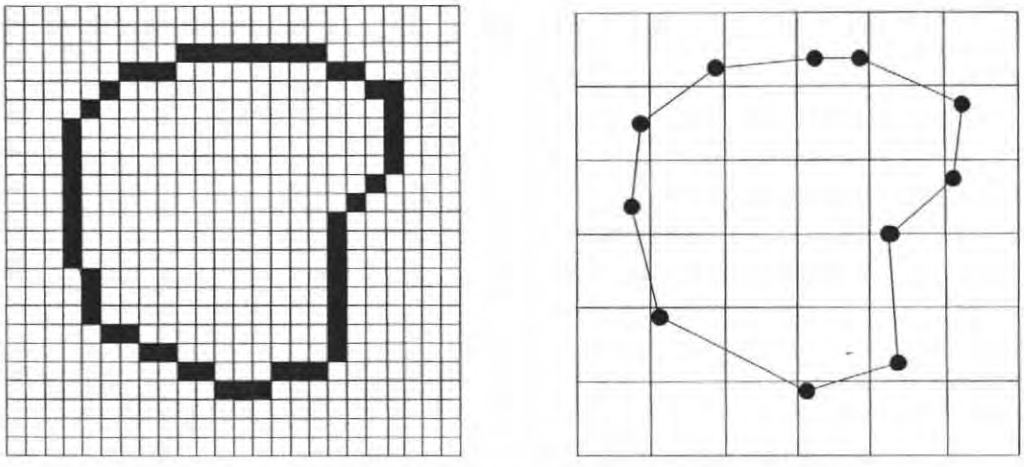
แผนที่ดิจิตอล (Digital map) หรือแผนที่เชิงตัวเลข เป็นแผนที่ที่ใช้คอมพิวเตอร์ในการประมวลผล และมีการจัดเก็บข้อมูลของแผนที่ให้อยู่ในรูปของข้อมูลคอมพิวเตอร์ ซึ่งข้อมูลคอมพิวเตอร์จะทำการจัดเก็บในรูปแบบของฐานข้อมูลคอมพิวเตอร์ ตัวอย่างของแผนที่ดิจิตอลแสดงในรูปที่ 2.6 แผนที่ดิจิตอลแบ่งตามการจัดเก็บออกเป็น 2 แบบคือ แบบราสเตอร์และแบบเวกเตอร์

รูปที่ 2.6 ตัวอย่างแผนที่ดิจิตอลจากโปรแกรม Map point Asia

แผนที่แบบราสเตอร์ หมายถึงแผนที่ที่มีการจัดเก็บและแสดงผลในรูปของจุดภาพ การสร้างแผนที่แบบนี้ทำได้โดยรับภาพแผนที่จากแผนที่กระดาษผ่านทางเครื่องสแกนภาพ ซึ่งวิธีการสแกนภาพเป็นการนำรูปภาพทั้งรูปเข้าไปเก็บไว้ในลักษณะของจุดภาพ ซึ่งการแก้ไขจะทำได้ยากรวมทั้งใช้เนื้อที่ในการจัดเก็บมาก และไม่สามารถคำนวณทางคณิตศาสตร์ได้

แผนที่แบบเวกเตอร์ หมายถึงแผนที่ที่มีการจัดเก็บและแสดงผลในรูปของลายเส้น มีตำแหน่งและทิศทาง การสร้างแผนที่แบบนี้ทำได้โดยใช้วิธีการลอกแบบจากเครื่องดิจิตไตเซอร์ ซึ่งจะเก็บเฉพาะข้อมูลในส่วนที่ต้องการลอกแบบ ดังนั้นข้อมูลแบบนี้จึงใช้เนื้อที่ในการจัดเก็บน้อยกว่า สามารถแก้ไขได้ในภายหลังโดยที่มาตราส่วนไม่ผิดไปจากเดิม อีกทั้งยังสามารถนำข้อมูลมาทำการคำนวณทางคณิตศาสตร์ได้อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก) (ข)

รูปที่ 2.7 ข้อมูลของแผนที่แบบราสเตอร์ และแบบเวกเตอร์

(ก) (ข)

รูปที่ 2.8 ตัวอย่างของแผนที่แบบราสเตอร์ และแบบเวกเตอร์

2.8.2 การคำนวณระยะทางบนแผนที่

จากที่ได้อธิบายเกี่ยวกับพิกัดตำแหน่งไปแล้ว จะพบว่าทุกพื้นที่บนแผนที่จะประกอบได้พิกัดตำแหน่งมากมาย ดังนั้นการคำนวณระยะทางจึงหมายถึงระยะห่างระหว่างตำแหน่ง 2 ตำแหน่งบนแผนที่ ซึ่งเมื่อเราทราบระบบพิกัดตำแหน่งบนแผนที่แล้ว เราสามารถหาระยะทางบนแผนที่ได้

2.8.3 ระยะทางระหว่างจุดสองจุดบนแผนที่

ระยะทางระหว่างจุดสองจุดบนแผนที่คำนวณได้ตาม (2.1) โดยที่ d หมายถึงระยะทางระหว่างตำแหน่งทั้ง 2 และ (x, y) คือ พิกัดตำแหน่งใด ๆ

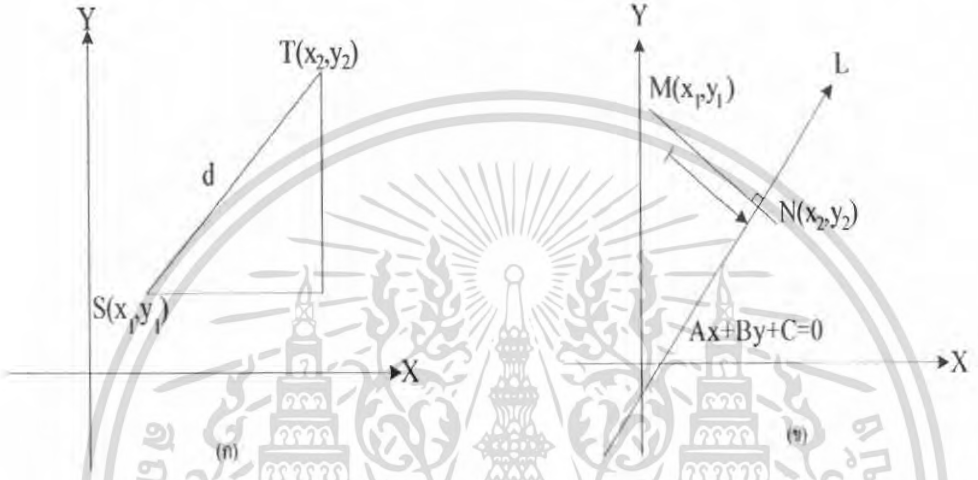
$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \qquad (2.1)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.4 ระยะทางระหว่างจุดถึงเส้นตรง

ระยะทางระหว่างจุดถึงเส้นตรง คำนวณได้ตาม (2.2) โดยที่ d หมายถึงระยะทางระหว่างตำแหน่งทั้ง 2 และ (x, y) คือพิกัดตำแหน่งใด ๆ

$$d = \frac{|Ax1 + By1 + C|}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}} \qquad (2.2)$$



รูปที่ 2.9 การคำนวณระยะห่างระหว่างจุดและเส้นตรงกับจุด

ตามรูปที่ 2.9(ก) เป็นตัวอย่างการคำนวณระยะทางจากตำแหน่ง S(x1, y1) ถึงตำแหน่ง T(x2, y2) ค่าของระยะทาง d ละรูปที่ 2.9(ข) เป็นตัวอย่างการคำนวณระยะทางจากตำแหน่ง M(x1, y1) ไปยังตำแหน่ง N(x2, y2) ที่อยู่บนเส้นตรง Ax + By + C = 0

จากหลักการนี้เราจะนำไปใช้ในการคำนวณหาพื้นที่บนแผนที่ เช่น พื้นที่สี่เหลี่ยมผืนผ้าคำนวณได้จาก d1*d2 เมื่อ d1 คือความกว้างและ d2 คือความยาว

2.9 การระบุพิกัดตำแหน่งภาคพื้นดินด้วยดาวเทียมจีพีเอส

ในส่วนนี้จะมาดูว่าวิธีการที่จะได้พิกัดตำแหน่งภาคพื้นดินนั้นจะต้องทำอย่างไรบ้างและมีอุปกรณ์อะไรที่จำเป็น โดยมีรายละเอียดที่ต้องศึกษาดังต่อไปนี้

2.9.1 ระบบระบุตำแหน่งบนโลกด้วยดาวเทียมระบบจีพีเอส

ระบบกำหนดตำแหน่งบนโลกด้วยดาวเทียมจีพีเอส เป็นระบบสำรวจหาตำแหน่งพื้นโลกด้วยดาวเทียมที่ออกแบบและจัดสร้างโดยกองทัพสหรัฐอเมริกา เพื่อใช้ในการนำร่อง (Navigation)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พัฒนาขึ้น เมื่อต้นปี 1978 มีชื่ออย่างเป็นทางการว่า Navigation Signal Timing and Ranging Global Positioning System (NAVSTAR GPS) มีวัตถุประสงค์ในการออกแบบคือ

1. เพื่อให้มีผู้ใช้ประโยชน์ทั้งฝ่ายทหารและพลเรือนได้เป็นจำนวนมาก
2. เพื่อใช้ได้สะดวกไม่มีข้อจำกัด นั่นคือ ใช้ได้ตลอด 24 ชั่วโมง โดยไม่ขึ้นกับสภาพภูมิอากาศและสถานที่
3. ให้ความถูกต้องทางตำแหน่งตามเงื่อนไขที่ฝ่ายทหารกำหนด

จีพีเอสมีหลักการทำงานโดยอาศัยคลื่นวิทยุ และรหัสที่ส่งมาจากดาวเทียม NAVSTAR จำนวน 24 ดวง ที่โคจรอยู่รอบโลกวันละ 2 รอบและมีตำแหน่งอยู่เหนือพื้นโลกที่ความสูง 20,200 กิโลเมตร สามารถใช้ในการหาตำแหน่งบนพื้นโลกได้ตลอด 24 ชั่วโมง ที่ทุกๆ จุดบนผิวโลกใช้นำร่องจากที่หนึ่งไปยังอีกที่หนึ่งตามต้องการ ใช้ติดตามการเคลื่อนที่ของคน สัตว์และสิ่งของต่างๆ การทำแผนที่ การทำงานรังวัด (Surveying) ตลอดจนใช้อ้างอิงการวัดเวลาที่เที่ยงตรงที่สุดในโลก

ระบบกำหนดตำแหน่งบนพื้นโลกด้วยดาวเทียมจีพีเอส ประกอบด้วย 3 ส่วนหลัก คือ ส่วนอวกาศ (Space Segment) ส่วนสถานีควบคุม (Control Segment) และส่วนผู้ใช้ (User Segment) ดังรูปที่ 2.10

รูปที่ 2.10 องค์ประกอบหลักของระบบ จีพีเอส

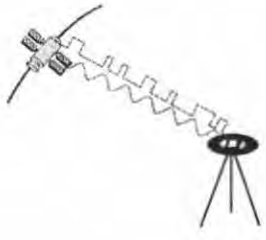
ส่วนอวกาศ เป็นส่วนที่อยู่บนอวกาศจะประกอบด้วย ดาวเทียม 24 ดวง โดยมีดาวเทียม 21 ดวงทำหน้าที่ส่งสัญญาณคลื่นวิทยุจากอวกาศ (Space Vehicles: SVs) ส่วนอีก 3 ดวง เป็นดาวเทียมปฏิบัติการเสริม วงโคจรของดาวเทียมแต่ละดวงจะใช้เวลาโคจร 12 ชั่วโมง ต่อ 1 รอบ โดยจะมีทั้งหมด 6 วงโคจร แต่ละวงโคจรมีดาวเทียม 4 ดวง วงโคจรมีมุมเอียง 55° กับระนาบศูนย์สูตรและห่างกัน 60° วงโคจรในลักษณะดังกล่าวจะทำให้มีดาวเทียมอย่างน้อย 4 ดวงอยู่บนท้องฟ้าทุกๆ จุดบนพื้นผิวโลก ตลอดเวลา 24 ชั่วโมง ดาวเทียมชุดแรก เรียกจีพีเอส Block I มีทั้งหมด 10 ดวง ดาวเทียมแต่ละดวงจะมีนาฬิกาที่มีความแม่นยำสูง ซึ่งเป็นชุดของนาฬิกาอะตอมมิค แบ่งออกเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แหล่งกำเนิดความถี่รูบิเดียม 2 เรือน และ ซีเซียม 2 เรือน ทำให้เวลามาตรฐานของดาวเทียมมีความถูกต้องสูงมาก นาฬิกาดังกล่าวช่วยในการคำนวณระยะทางระหว่างดาวเทียมกับเครื่องรับสัญญาณเพื่อที่จะคำนวณค่าพิกัดตำแหน่งได้

ภาคกำเนิดสัญญาณเวลาความแม่นยำสูง เป็นหัวใจสำคัญ ที่เป็นตัวกำหนด ความแม่นยำถูกต้อง ในการคำนวณตำแหน่งพิกัด ของเครื่องรับจีพีเอส ที่รับสัญญาณบนโลก หากภาคกำเนิดสัญญาณเวลาบนดาวเทียม ดวงใดเสื่อมสภาพ หรือไม่มีความแม่นยำเพียงพอ ดาวเทียมดวงนั้นจะถูกปลดออกจากการใช้งาน ตามแผนงานจะมีดาวเทียมโคจรทั้งหมด 24 ดวง และสำรอง 3 ดวง โดยมีการส่งดาวเทียมใหม่เข้าสู่วงโคจรตามระยะเวลาที่เหมาะสม เพื่อชดเชยดาวเทียมที่เสื่อมสภาพ แต่เนื่องจากเทคโนโลยีของดาวเทียมและภาคกำเนิดสัญญาณเวลาความแม่นยำสูง มีการพัฒนาตลอดเวลาทำให้อายุการใช้งานของดาวเทียมยาวกว่าที่คำนวณไว้ ดาวเทียมจำนวนมากยังอยู่ในภาวะใช้งานได้ปกติถึงแม้จะอยู่ในวงโคจร มานานกว่า 8 ปี (อายุขัยเฉลี่ยของดาวเทียม) ทำให้ปัจจุบันมีดาวเทียมอยู่ในวงโคจรที่ใช้งานได้มากกว่า 24 ดวง

คลื่นสัญญาณวิทยุที่ส่งออกมาจากดาวเทียมจะมีการเข้ารหัสสัญญาณ 2 รหัสด้วยกัน คือ รหัส C/A (Coarse/Acquisition) และรหัส P (Precision) คือ รหัส C/A มีความถี่เป็น 1/10 ของความถี่พื้นฐานคือ 1.023 MHz ความยาวคลื่นเป็น 300 เมตร มีคาบเป็น 1 ใน 1,000 วินาที นั่นคือในช่วงเวลา 1 วินาที จะสร้างรหัส C/A ที่มีรูปแบบเหมือนกันซ้ำถึง 1,000 ครั้ง การตรวจสอบรูปแบบของรหัส C/A จึงทำได้ง่ายและรวดเร็วมาก รหัส C/A เปิดให้ทุกคนใช้ได้อย่างอิสระ ส่วนรหัส P มีความถี่เท่ากับความถี่พื้นฐานคือ 10.23 MHz ความยาวคลื่นเป็น 30 เมตร และมีคาบเป็น 267 วัน นั่นคือ ในช่วง 267 วัน รหัส P ที่ส่งออกมาจะมีรูปแบบที่ไม่ซ้ำกัน จึงเป็นการยากที่จะตรวจสอบว่ารหัส P ที่ดาวเทียมใช้ในแต่ละวันเป็นส่วนไหนของรหัส ผู้ที่ไม่มีข้อมูลเกี่ยวกับการสร้างรหัส P ของดาวเทียม จึงไม่อาจใช้ประโยชน์จากรหัส P เพื่อหาตำแหน่งได้ รหัส P จะถูกสงวนไว้ใช้เฉพาะวงการทหารและหน่วยงานของรัฐบาลสหรัฐอเมริกา รวมทั้งพันธมิตรทางทหารของสหรัฐอเมริกาเท่านั้น คลื่นส่ง L1 ถูกรวมสัญญาณ (Modulate) ด้วยรหัสทั้งสองชนิด ส่วนคลื่นส่ง L2 มีเพียงรหัส P และรหัส P จะถูกเปลี่ยนเป็นรหัส Y ในกรณีที่ต้องการป้องกันการใช้ประโยชน์จากรหัส P ดังรูปที่ 2.11 แสดงรหัส และคลื่นวิทยุที่ส่งจากดาวเทียมจีพีเอส



รูปที่ 2.11 แสดงรหัสและคลื่นวิทยุที่ส่งจากดาวเทียมจีพีเอส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

นอกจากคลื่นวิทยุแล้วยังมีข้อมูลดาวเทียมอื่นๆ ที่ใช้สำหรับการคำนวณตำแหน่งถูกรวมสัญญาณมาพร้อมคลื่นส่งด้วย ได้แก่ อีเฟเมอริสดาวเทียม (Satellite Ephemeris) ซึ่งเป็นข้อมูลจำเพาะ ประกอบไปด้วยข้อมูลวงโคจร สถานภาพของดวงเทียม เวลามาตรฐานดาวเทียม พฤติกรรมของนาฬิกาดาวเทียม อัตราเร็วที่ใช้ในการส่งข้อมูลดาวเทียมคือ 50 bps (bits per second) หรือ 50 ตัวอักษรต่อวินาที กลุ่มดาวเทียม NAVSTAR และรายละเอียดของดาวเทียมแสดงในรูปที่ 2.12

เจ้าของ	: กระทรวงกลาโหม ประเทศสหรัฐอเมริกา
น้ำหนัก	: 900 กิโลกรัม กว้าง : 5 เมตร
วงโคจร	: รอบโลกในเวลา 11 ชั่วโมง 58 นาที ที่ ความสูง 20,200 กิโลเมตร โคจรเอียง 55 องศากับระนาบศูนย์สูตร และทำมุมระหว่างกัน 60 องศา แบ่งเป็น 6 วงโคจรๆ ละ 4 ดวง รวมดาวเทียมใน วงโคจร 24 ดวง
รหัส	: รหัส C/A ที่ความถี่ 1.023 MHz และรหัส P ที่ ความถี่ 10.23 MHz
ส่งข้อมูลดาวเทียม	: อีเฟเมอริสดาวเทียม เวลามาตรฐาน ดาวเทียม พฤติกรรมของนาฬิกา ดาวเทียม สถานะภาพของระบบดาวเทียม

รูปที่ 2.12 ดาวเทียม NAVSTAR

ส่วนสถานีควบคุม ประกอบไปด้วยสถานีภาคพื้นดินที่ควบคุมระบบ (Operational Control System : OCS) ที่กระจายอยู่ตามส่วนต่างๆ บนโลกมีหน้าที่ปรับปรุงให้ข้อมูลดาวเทียมมีความถูกต้องทันสมัยอยู่ตลอดเวลา โดยแบ่งออกเป็น

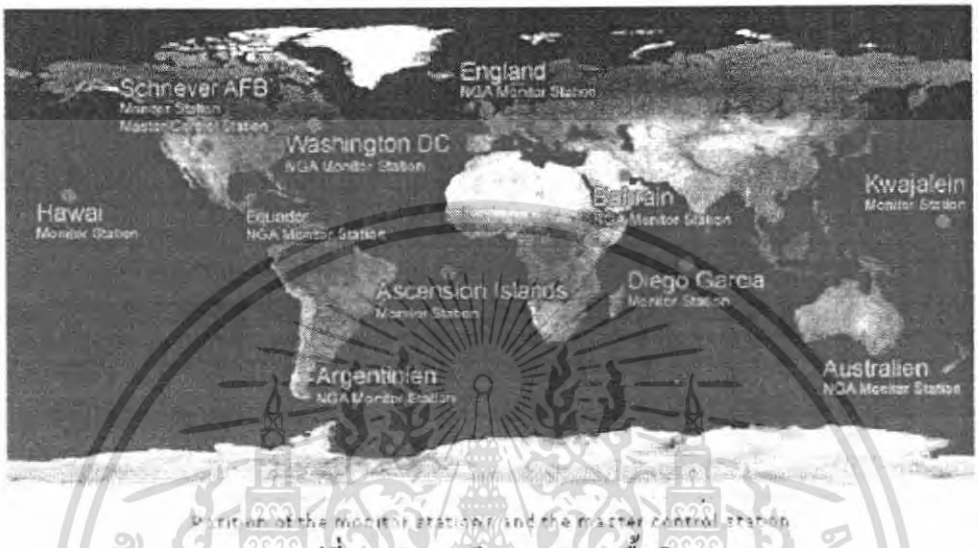
สถานีควบคุมหลัก ตั้งอยู่ที่ฐานทัพอากาศในเมืองโคโลราโดสปริงส์ มลรัฐโคโรลาโดของประเทศสหรัฐอเมริกา สถานีติดตามดาวเทียม 5 แห่ง ทำการรังวัดติดตามดาวเทียม ตลอดเวลา โดยตั้งอยู่ที่หมู่เกาะฮาวาย (Hawaii) ในมหาสมุทรแปซิฟิก หมู่เกาะแอสเซนซัน (Ascension) มหาสมุทรแอตแลนติกหมู่เกาะดิเอโกการ์เซีย (Diego Garcia) มหาสมุทรอินเดีย หมู่เกาะควาจาเลียน (Kwajalein) ประเทศฟิลิปปินส์ และเมืองโคโลราโดสปริงส์ สถานีรับส่งสัญญาณ 3 แห่ง ได้แก่ หมู่เกาะควาจาเลียน หมู่เกาะดิเอโกการ์เซีย และหมู่เกาะแอสเซนซัน

ในช่วงเดือนสิงหาคมและกันยายน ปี 2548 National Geospatial-Intelligence Agency (NGIA) ได้เพิ่มสถานีติดตามดาวเทียมอีก 6 แห่ง ดังนั้นในปัจจุบันมีสถานีควบคุมหลักและสถานีติดตามดาวเทียมรวมทั้งสิ้น 11 แห่ง ทำให้สถานีติดตามดาวเทียมอย่างน้อย 2 สถานีมองเห็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

83125

ดาวเทียมทั้งหมด ดังแสดงในรูปที่ 2.13 เพื่อทำการประมวลผลหาวงโคจรของดาวเทียมดวงต่างๆ เมื่อได้พยากรณ์ตำแหน่งของดาวเทียมล่วงหน้าแล้ว ก็จะจัดส่งข้อมูลที่ได้ปรับปรุงแล้ว พร้อมกับข้อมูลเวลาและข้อมูลอุตุนิยมวิทยาไปยังสถานีรับส่งสัญญาณ 3 แห่ง เพื่อส่งไปเก็บบันทึกไว้ในดาวเทียมต่อไป



รูปที่ 2.13 สถานีควบคุมภาคพื้นดิน

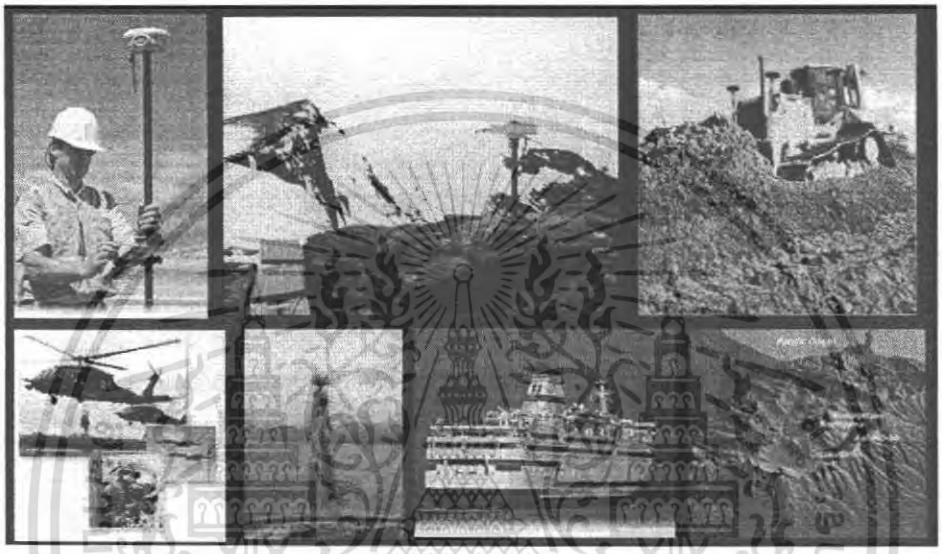
สถานีภาคพื้นดินที่ควบคุมระบบ จะเฝ้าระวังติดตามดาวเทียม และข้อมูลที่ได้จากการเฝ้าระวังติดตามดาวเทียม สามารถที่จะบอกวงโคจรล่วงหน้าได้อย่างถูกต้องแม่นยำ และจะส่งสัญญาณข้อมูลวงโคจรจากสถานีสู่ดาวเทียมวันละ 3 ครั้ง และกระจายข้อมูล วงโคจรเหล่านั้นจากดาวเทียมสู่เครื่องรับสัญญาณจีพีเอสได้ การรับสัญญาณนี้เรียกว่า "Broadcast Ephemeris" ซึ่งจะทำให้เครื่องรับสัญญาณจีพีเอสคำนวณตำแหน่งในเวลาจริงได้ สำหรับการสำรวจในทาง Geodetic อาจจะไม่ละเอียดเพียงพอ อย่างไรก็ตามก็ไม่มีความจำเป็นที่จะต้องคำนวณตำแหน่งในเวลาจริง แต่อาจสามารถทำได้ด้วยวิธีที่เรียกว่า Precise Ephemeris ซึ่งจะให้ข้อมูลเกี่ยวกับวงโคจรภายหลังที่ละเอียดกว่าการติดตามเฝ้าระวัง ไม่เพียงแต่ใช้สถานี OCS ทั้ง 11 สถานี แต่ใช้สถานีเครือข่ายอื่นๆ "Cooperative International GPS Network" (CIGNET) ซึ่งมีอยู่ทั้งสิ้น 29 สถานีกระจายทั่วโลก

ส่วนผู้ใช้ ประกอบด้วยเครื่องรับสัญญาณหรือตัวจีพีเอสที่เราใช้อยู่มีหลายขนาด สามารถพกพาได้ หรือติดไว้ในรถ เรือ หรือเครื่องบิน เครื่องจีพีเอสจะทำหน้าที่ในการเปลี่ยนสัญญาณจากดาวเทียม เป็นตำแหน่ง ความเร็ว และเวลาโดยประมาณ ถ้าหากต้องการทราบค่า X, Y, Z และเวลาต้องใช้ดาวเทียมอย่างน้อย 4 ดวง ความถูกต้องของตำแหน่งขึ้นอยู่กับนาฬิกาและตัวจีพีเอสซึ่งอาจจะหาตำแหน่งที่มีความผิดพลาดได้น้อยกว่า 3 ฟุต นาฬิกาที่ใช้จะมีความถูกต้องสามารถวัดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในเวลา 0.000000003 วินาที ซึ่งเวลาที่ใช้ในการอ้างอิงสำหรับระบบดาวเทียมจีพีเอส เรียกว่าเวลาจีพีเอส

ส่วนผู้ใช้ประกอบด้วย 2 ส่วนใหญ่ ๆ คือ ส่วนที่เกี่ยวข้องกับพลเรือน และส่วนที่เกี่ยวกับทางทหาร ในส่วนของผู้ใช้จะมีหน้าที่พัฒนาเครื่องรับสัญญาณ (Receiver) ให้ทันสมัยและสะดวกแก่การใช้งาน สามารถที่จะใช้ได้ทุกแห่งในโลก และให้ค่าที่มีความถูกต้องสูง ดังตัวอย่างการประยุกต์ใช้ดาวเทียมระบบจีพีเอสในรูปที่ 2.14



รูปที่ 2.14 การใช้ประโยชน์ของดาวเทียมระบบจีพีเอส

2.9.2 หลักการทำงานของเครื่องรับสัญญาณจีพีเอส

หลักการทำงานของเครื่องจีพีเอส คือการคำนวณหาระยะทางระหว่างดาวเทียมกับเครื่องรับจีพีเอส ซึ่งจะต้องใช้ระยะทางจากดาวเทียมอย่างน้อย 4 ดวง เพื่อให้ได้ตำแหน่งที่แน่นอน ซึ่งเมื่อเครื่องจีพีเอส สามารถรับสัญญาณจากดาวเทียมได้ 4 ดวงเป็นอย่างต่ำแล้ว จะมีการคำนวณระยะทางระหว่างดาวเทียม ถึงเครื่องจีพีเอส จากสูตรคำนวณทางฟิสิกส์ คือ

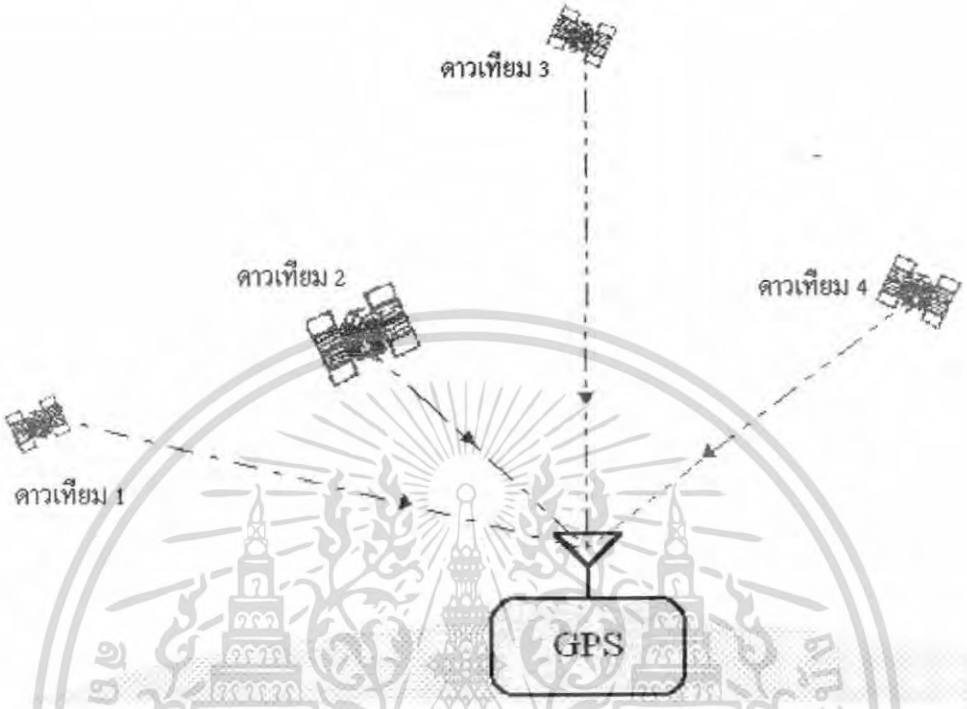
$$\text{ความเร็ว} \times \text{เวลา} = \text{ระยะทาง} \qquad (2.3)$$

ดาวเทียมทั้ง 4 ดวงจะส่งสัญญาณคลื่นวิทยุมายังเครื่องจีพีเอส ด้วยความเร็วแสง (186,000 ไมล์ต่อวินาทีหรือประมาณ 300,000 กิโลเมตรต่อวินาที) แต่ระยะเวลาในการ รับสัญญาณจากดาวเทียมแต่ละดวงนั้นจะไม่เท่ากันเนื่องจากระยะทางไม่เท่ากัน ดังแสดงการรับสัญญาณที่ระยะต่างๆ ดังรูปที่ 2.15

ดาวเทียม 1 : ระยะเวลาในการส่งสัญญาณจากดาวเทียมดวงแรกถึงเครื่องจีพีเอส คือ 0.10 วินาที ระยะทางระหว่างดาวเทียมกับจีพีเอส คือ 18,600 ไมล์ (186,000 ไมล์ต่อวินาที × 0.10 วินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

= 18,600 ไมล์) ฉะนั้นตำแหน่งปัจจุบันก็จะสามารถเป็นจุดใดก็ได้ในวงกลมที่มีรัศมี 18,600 ไมล์ ซึ่งจะเห็นว่าดาวเทียมเพียงดวงเดียวยังไม่สามารถบอกตำแหน่งที่แน่นอนได้



รูปที่ 2.15 การรับสัญญาณจากดาวเทียม 4 ดวง

ดาวเทียม 2 : ระยะเวลาในการส่งสัญญาณจากดาวเทียมดวงที่ 2 ถึงเครื่องจีพีเอส คือ 0.08 วินาที ระยะทางระหว่างดาวเทียมกับจีพีเอส คือ 13,200 ไมล์ (186,000 ไมล์ต่อวินาที × 0.08 วินาที = 13,200 ไมล์) ฉะนั้นตำแหน่งปัจจุบันก็จะสามารถเป็นจุดใดก็ได้ในจุดตัด (Intersect) ระหว่างวงกลมจากดาวเทียมดวงแรกกับดาวเทียมดวงที่ 2

ดาวเทียม 3 : ระยะเวลาในการส่งสัญญาณจากดาวเทียมดวงที่ 3 ถึงเครื่องจีพีเอส คือ 0.06 วินาที ระยะทางระหว่างดาวเทียมกับจีพีเอส คือ 11,160 ไมล์ (186,000 ไมล์ต่อวินาที × 0.06 วินาที = 11,160 ไมล์) ฉะนั้นตำแหน่งปัจจุบันก็จะสามารถเป็นจุดใดก็ได้ในจุดตัดกัน ระหว่างวงกลมจากดาวเทียมทั้ง 3 ดวง

ข้อมูลตำแหน่งที่ได้มานั้น ยังสามารถใช้ร่วมกับโปรแกรมในเครื่องจีพีเอส เพื่อบอก จุดบนแผนที่ และแสดงตำแหน่งของเราว่าอยู่จุดใดของแผนที่ได้อีกด้วย ทั้งนี้ก็ขึ้นกับข้อมูลแผนที่ที่ติดมากับเครื่องด้วยว่ามีความแม่นยำเพียงใด โดยแผนที่พื้นฐานจะไม่ได้ติดตั้งมากับเครื่องจีพีเอส ทุกรุ่น ซึ่งอาจจะต้องซื้อแยกจากตัวเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การวัดระยะทางไปยังดาวเทียม เรียกว่า แร็งกิ้ง (Ranging) ซึ่งการคำนวณตำแหน่งต้องวัดระยะทางไปยังดาวเทียมอย่างน้อย 3 ดวง พร้อมๆ กัน และระยะทางทั้ง 3 ต้องไม่เป็นเส้นที่อยู่บนระนาบเดียวกัน ระยะทางจากดาวเทียมแต่ละดวงซึ่งก็คือวงกลมแต่ละวงนั่นเอง ถ้าหากมีวงกลม 2 วง ก็ยังไม่สามารถหาตำแหน่งที่แน่นอนได้เพราะจุดตัดของวงกลมมี 2 จุด จุดตัดก็คือเส้นรอบวงของวงกลมหรือตำแหน่งของตัวรับสัญญาณนั่นเอง แต่เมื่อมีวงกลมตั้งแต่ 3 วงขึ้นไป จะรู้ตำแหน่งที่แน่นอน เพราะจุดตัดจะมีจุดเดียว เมื่อได้ระยะทางอย่างน้อย 3 ระยะทางจากดาวเทียม 3 ดวงก็จะสามารถหาตำแหน่งได้ แต่ถ้ามีจำนวนดาวเทียมมากก็จะได้ตำแหน่งที่ถูกต้องมาก สามารถหาค่าของเวลา และความสูงได้ด้วย

การวัดระยะทางไปยังดาวเทียม 3 ดวง และรู้ตำแหน่งของดาวเทียมที่วัดระยะไปนั้น สามารถสร้างสมการได้ 3 สมการ และมีตัวไม่รู้ค่า 3 ตัว ซึ่งสามารถแก้สมการหาตัวไม่รู้ค่านั้นได้ โดยจะทราบตำแหน่งแบบ 3 มิติ คือทราบค่า X, Y และ Z เมื่อวัดระยะทางจากดาวเทียม 4 ดวงขึ้นไปก็จะได้ตำแหน่งที่ถูกต้องมากขึ้น โดยทราบสมการเพิ่มขึ้นอีก 1 ตัว คือ เวลา (T)

2.9.3 ความคลาดเคลื่อนในการใช้งานจีพีเอส

ในการใช้งานจีพีเอส ความคลาดเคลื่อนที่เกิดขึ้น อาจพิจารณาแยกได้เป็น 3 กลุ่มคือ กลุ่มเกี่ยวข้องกับดาวเทียม ได้แก่ ความคลาดเคลื่อนวงโคจรและความคลาดเคลื่อนนาฬิกาดาวเทียม กลุ่มเกี่ยวข้องกับการแพร่กระจายของสัญญาณได้แก่ ความคลาดเคลื่อนของการหักเหในชั้นบรรยากาศและการเกิดคลื่นสะท้อน กลุ่มสุดท้ายเกี่ยวข้องกับเครื่องรับสัญญาณ เช่น นาฬิกาเครื่องรับ ดังรูป

รูปที่ 2.16 ความคลาดเคลื่อนในการใช้งานจีพีเอส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9.4 ความคลาดเคลื่อนวงโคจรดาวเทียม

ความคลาดเคลื่อนวงโคจรดาวเทียมมีสาเหตุมาจากวงโคจรดาวเทียม ที่ได้มาจากข้อมูลดาวเทียมในสัญญาณที่รับได้นั้นเป็นวงโคจรที่ได้จากการคำนวณล่วงหน้า โดยอาศัยรูปจำลองของแรงต่างๆที่กระทำต่อดาวเทียมรูปจำลองที่ใช้อาจจะไม่ถูกต้องหรือไม่ละเอียดเพียงพอ เมื่อเทียบกับแรงจริงๆ ที่กระทำต่อดาวเทียมในขณะที่ทำการวัด ดังนั้นตำแหน่งดาวเทียมจากอีเฟเมอริสดาวเทียม ที่ส่งกระจายลงมาพร้อมสัญญาณดาวเทียมจึงไม่ถูกต้อง ซึ่งมีผลต่อการหาตำแหน่งสัมบูรณ์มากกว่าการหาตำแหน่งแบบสัมพัทธ์ เพราะในการหาตำแหน่งแบบสัมพัทธ์นั้น ความคลาดเคลื่อนวงโคจรจะมีผลต่อจุดปลายทั้งสองของเส้นฐานในลักษณะที่คล้ายคลึงกันและมีขนาดใกล้เคียงกัน

2.9.5 ความคลาดเคลื่อนนาฬิกาดาวเทียมและนาฬิกาเครื่องรับ

เวลาดาวเทียม หมายถึง เวลาที่อ่านได้จากนาฬิกาของดาวเทียม บนดาวเทียมแต่ละดวงมีนาฬิกาที่มีมาตรฐานสูง 4 เครื่อง สถานีควบคุมหลักสามารถควบคุมเวลาดาวเทียมให้ต่างจากเวลาจีพีเอส ไม่เกินกว่า 1 ใน 1,000 วินาที (1 ms) และควบคุมความถี่ให้มีความถูกต้องถึง 10-9 เวลาที่ถูกต้องของเครื่องรับสามารถหาได้จากสัญญาณดาวเทียมที่มีเวลาจากนาฬิกาดาวเทียมอยู่ เมื่อเครื่องรับถอดรหัสได้ก็จะรู้เวลาจีพีเอส เครื่องรับจึงไม่จำเป็นต้องมีนาฬิกาที่มีมาตรฐานสูง

2.9.6 ความคลาดเคลื่อนเนื่องจากการหักเหในชั้นบรรยากาศ

ชั้นบรรยากาศชั้นไอโอโนสเฟียร์ และชั้นโทรโพสเฟียร์มีผลต่อการหักเหคลื่นแม่เหล็กไฟฟ้า ทำให้เส้นทางการเคลื่อนที่ของคลื่นเบี่ยงเบนไป

2.9.7 คลื่นสะท้อน

คลื่นสะท้อน หมายถึง การแพร่กระจายของคลื่นที่มีการสะท้อนตั้งแต่หนึ่งครั้งขึ้นไป พื้นผิวที่สะท้อนอาจจะอยู่ในแนวดิ่ง ราบ หรือเอียงก็ได้ เช่น ผนังตึก ถนน ผิวน้ำ หรือยานพาหนะ คลื่นสะท้อนมีผลกับทั้งรหัสและคลื่น ผลของคลื่นสะท้อนเป็นสาเหตุหนึ่งที่ทำให้เกิดคลื่นหลุด (Cycle Slip) เพราะช่องรับสัญญาณไม่สามารถล็อกสัญญาณดาวเทียมได้ การรังวัดเป็นเวลานานจะทำให้ผลของคลื่นสะท้อนลดลง แต่เทคนิคของการรังวัดบางวิธีไม่สามารถวางเสาอากาศทิ้งไว้เป็นเวลานานๆ ได้ ดังนั้นจึงต้องหลีกเลี่ยงการรับสัญญาณที่มีคลื่นสะท้อนโดยเลือกจุดวางเครื่องรับที่ไม่มีพื้นผิวสะท้อนอยู่ใกล้เคียง เลือกเสาอากาศที่ออกแบบเฉพาะ เช่น เสาอากาศที่มีแผ่นกราวนด์ หรือใช้วัสดุที่ดูดซับคลื่นวางรอบเสาอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9.8 ความคลาดเคลื่อนของเครื่องรับ

ความคลาดเคลื่อนของเครื่องรับ มีสาเหตุมาจากหลายส่วน ได้แก่ สัญญาณรบกวน(Noise) ในการวัดของเครื่องรับความเอนเอียง (Bias) ระหว่างช่องรับสัญญาณ การประวิงของเฟส ความไม่เสถียรของออสซิลเลเตอร์ จุดศูนย์กลางเฟสของเสาอากาศ

2.10 มาตรฐานในการสื่อสารข้อมูลจีพีเอส

ในการที่เครื่องรับสัญญาณดาวเทียมจีพีเอสจะสามารถส่งข้อมูลไปยังอุปกรณ์อื่นได้นั้นต้องอาศัยมาตรฐานในการส่งข้อมูล โดยหน่วยงานที่ทำหน้าที่กำหนดมาตรฐานในการสื่อสารข้อมูล จีพีเอส คือหน่วยงาน NMEA (National Marine Electronics Association) ซึ่งได้มีการกำหนดให้โปรโตคอล NMEA – 0183 เป็นโปรโตคอลมาตรฐานในการสื่อสารข้อมูลจากเครื่องรับจีพีเอสไปสู่อุปกรณ์เชื่อมต่อภายนอก ซึ่งข้อมูลเหล่านี้อยู่ในรูปของรหัสแอสกี้ (ASCII Codes) จะถูกส่งผ่านทางพอร์ต EIA-422A ไปเข้าสู่คอมพิวเตอร์ผ่านทางพอร์ตอนุกร (RS-323) โดยกำหนดอัตราความเร็วในการส่งข้อมูลอยู่ที่ 4,800 บิทต่อวินาที โดยมีบิทข้อมูล 8 บิทแบบไม่มีพาริตี้บิท แต่มีบิทเริ่มต้น (Start Bit) และบิทสิ้นสุด (Stop Bit)

ตามโปรโตคอล NMEA-0183 ลักษณะข้อมูลที่ใช้ในการสื่อสารอยู่ในรูปแบบประโยคโดยที่อักขระพิเศษหมายถึงขึ้นบรรทัดใหม่ (<CR> <LF>) แต่ละส่วนในประโยคมีการขั้นด้วยเครื่องหมายจุลภาค มีการแบ่งประโยคเป็นกลุ่มรูปแบบที่แตกต่างได้แก่ ประโยค GLL, RMC, GGA, STN, TRF, และ VRW เป็นต้น และเพื่อให้เข้าใจในการแปลความหมายของประโยคในมาตรฐาน NMEA จะขอยกตัวอย่างประโยค RMC

$GPGLL,xxx.xx,a,yyy.yy,b,hhmmss.ss<CR><LF>

โดยที่ xxx.xx คือ ค่าละติจูด

a คือ ทิศของค่าละติจูด โดยที่ค่า N(North) หมายถึงทิศเหนือและ S(South) หมายถึงทิศใต้

yyy.yy คือ ค่าลองติจูด

b คือ ทิศของค่าลองติจูด โดยที่ค่า E(East) หมายถึงทิศตะวันออกและ W(West) หมายถึงทิศตะวันตก

hhmmss.ss คือ เวลาที่เป็นชั่วโมง นาที และวินาที

ตัวอย่างของประโยค GLL ที่ได้จากจีพีเอส เช่น

$GPGLL,2028.1678,N,09918.2884,E,123010<CR><LF>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากประโยค GLL ข้างต้นมีความหมายดังนี้คือ พิกัดที่รับได้อยู่ที่ละติจูดที่ 20 องศา 28.1678 ฟิลิปดา ทางทิศเหนือ ลองติจูดที่ 99 องศา 18.2884 ฟิลิปดา ทางทิศตะวันออก เวลาที่รับสัญญาณได้คือ 12 นาฬิกา 30 นาที 10 วินาที

ส่วนประโยค RMC มีข้อมูลที่ต้องการครบทุกค่า คือ ละติจูด, ลองติจูด และเวลายูทีซี (UTC time) ซึ่งเร็วกว่าในประเทศไทยอยู่ 7 ชั่วโมง, ความเร็ว (น็อตต่อชั่วโมง),องศา, วันเดือนปี ซึ่งมีลักษณะข้อมูลดังนี้

$GPRMC,hhmmss.ss,A,llll.ll,e,yyyyy.yy,f,x.x,y.y,ddmmyy,z.z,a*hh<CR><LF>

ตัวอย่างข้อมูลจริงที่ได้จาก GPS ดังเช่น

$GPRMC,140745.000,A,1343.7523,N,10046.5119,E,2.24,290.37,200807,,*03<CR><LF>

เซตข้อมูลที่ใช้งานในประโยค RMC ประกอบไปด้วย

เวลา	=	14:07:45 UTC
ละติจูด	=	1343 ลิปดา กับ 7523 ฟิลิปดา
N/S	=	เป็น ละติจูด ทางเหนือ (N)
ลองติจูด	=	10046 ลิปดากับ 5119 ฟิลิปดา
E/W	=	เป็น ลองติจูด ทางตะวันออก (W)
ความเร็ว	=	2.24 น็อต
องศา	=	290.37 องศา ทิศทางอ้างอิงจากขั้วโลกเหนือ
วันที่	=	วันที่ 20 เดือน สิงหาคม 2007

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
กราฟฟิกส์ไลบราลี OpenGL

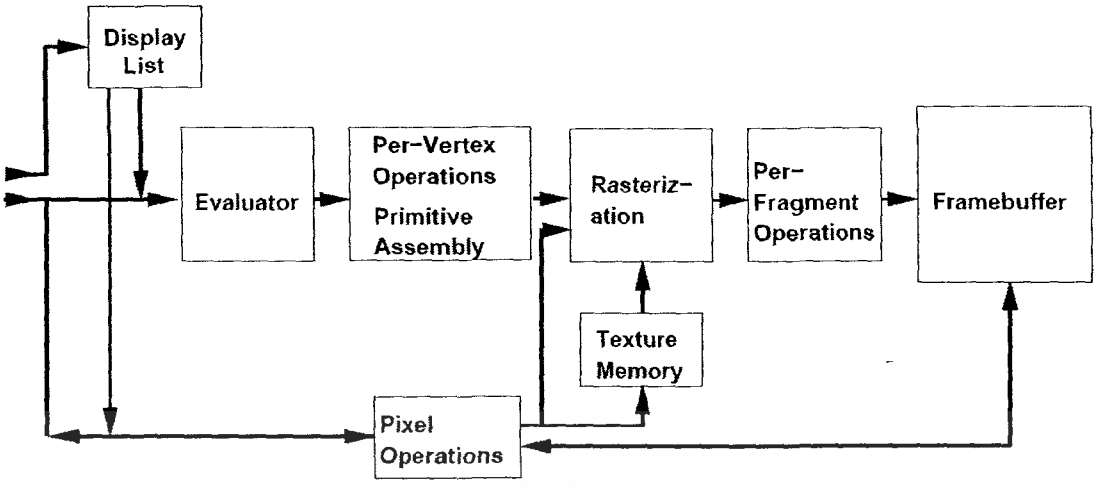
3.1 บทนำ

OpenGL (Open Graphics Library) เป็นซอฟต์แวร์ไลบราลี (Software Library) ที่ใช้ติดต่อกับฮาร์ดแวร์เพื่อการแสดงภาพกราฟฟิกส์ โดย OpenGL จะมีคำสั่งสำหรับการวาดภาพพื้นฐาน คือ จุด เส้น และรูปเหลี่ยมต่างๆ และการแสดงภาพราสเตอร์ ซึ่งคำสั่งพื้นฐานมีประมาณ 120 คำสั่งที่สามารถใช้กำหนดคุณลักษณะและควบคุมการทำงานของแอปพลิเคชัน 3 มิติ ซึ่งผู้พัฒนาโปรแกรมสามารถใช้ไลบราลี OpenGL ได้โดยไม่มีค่าลิขสิทธิ์ ทำให้มีการนำไลบราลีของ OpenGL ไปใช้งานอย่างแพร่หลายในงานกราฟฟิกส์

3.2 OpenGL

เนื่องจากโครงสร้างของ OpenGL เป็นอินเทอร์เฟซที่เป็นอิสระจากฮาร์ดแวร์ (Hardware-independent interface) และสามารถใช้ได้กับระบบปฏิบัติการหลายๆ แบบ ไม่ว่าจะเป็น Windows, Unix, IBM OS/2, Apple Mac OS หรือ Symbian S60 ก็ได้ และด้วยเหตุที่ OpenGL ถูกออกแบบให้ทำงานได้โดยไม่ยึดติดกับระบบ สามารถทำงานได้ทุกๆ แพล็ตฟอร์ม (Independent Platform) ทำให้สามารถเคลื่อนย้ายโค้ดที่สร้างเรียบร้อยแล้วไปใช้กับแพล็ตฟอร์มอื่นได้อย่างสะดวก (Portability) โดยไม่ต้องเปลี่ยนแปลงโค้ดโปรแกรมเลย การที่ OpenGL สามารถใช้ได้กับระบบปฏิบัติการที่หลากหลายนี้เอง ทำให้ OpenGL ไม่มีคำสั่งที่จัดการกับระบบปฏิบัติการเลย อีกทั้งยังไม่มีคำสั่งเพื่อรับอินพุตจากผู้ใช้อีกด้วย หน้าที่ทั้งสองอย่างนี้เป็นของผู้เขียนโปรแกรม (Programmer) ที่จะต้องออกแบบและเขียนโค้ดเพื่อให้การทำงานเป็นไปอย่างมีประสิทธิภาพ แต่อย่างไรก็ตาม ยังมียูทิลิตี้ที่ช่วยจัดการงานทั้งสองนี้หากพัฒนาโปรแกรมบนระบบปฏิบัติการแบบ Windows ยูทิลิตี้ดังกล่าวคือ GLUT(OpenGL Utility Toolkit) นอกจากนี้ OpenGL ยังไม่มีคำสั่งระดับสูงที่จะใช้วาดวัตถุ 3 มิติแบบซับซ้อน เช่น รถยนต์ อวัยวะ หรือสภาพภูมิประเทศต่างๆ สิ่งที่ OpenGL เตรียมไว้มีเพียงการสร้างรูปจำลอง 3 มิติคือรูปทรงเลขาคณิตอย่างง่ายได้แก่ จุด เส้น และรูปหลายเหลี่ยมซึ่งผู้ใช้งานจะต้องนำรูปทรงเหล่านี้มาประกอบกันเพื่อให้เกิดรูปทรง 3 มิติที่ซับซ้อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 สถาปัตยกรรมของ OpenGL

3.3 ข้อดีของ OpenGL

ในการออกแบบและการทำงานในระบบกราฟฟิกส์ มีเครื่องมือที่ช่วยใช้งานหลายเครื่องมือ แต่ละเครื่องมือมีคุณสมบัติ, ฟังก์ชัน หรือเครื่องมือที่แตกต่างกันไป เครื่องมือเหล่านั้นอาจจะมีจุดเด่น-จุดด้อยที่แตกต่างกันไป แต่ปัจจุบันเครื่องมือที่นิยมนำใช้ในการสร้างระบบกราฟฟิกส์ก็คือ OpenGL ซึ่งข้อดีที่ของ OpenGL ที่ทำให้เป็นที่นิยมมีดังนี้

1. มีประสิทธิภาพในการเร่งความเร็วแอปพลิเคชัน 3 มิติ และเกมส์ต่างๆ ในปัจจุบัน
2. สามารถใช้ข้อมูลจำนวนมากและสร้างเอ็ฟเฟ็กต์ 3 มิติในแบบเรียลไทม์ได้อย่างมีประสิทธิภาพ
3. เพิ่มการสนับสนุนอุปกรณ์ใหม่ๆ ลงไปใน OpenGL ที่ทำได้ง่ายและรวดเร็ว
4. ทำงานได้บนหลายแพล็ตฟอร์ม ทำให้การย้ายโปรแกรมประยุกต์ระหว่างแต่ละแพล็ตฟอร์มนั้นทำได้งาน และประหยัด
5. มีเสถียรภาพในการทำงานสูงสามารถใช้งานกับเครื่องเวิร์คสเตชันแบบ High End 3D และซุปเปอร์คอมพิวเตอร์ได้

3.4 รูปทรงเรขาคณิตพื้นฐาน (Geometric Primitives)

รูปทรงเรขาคณิตพื้นฐานของ OpenGL ได้แก่ จุด เส้น และรูปเหลี่ยม โดยรูปทรงทั้งสามนี้ถูกกำหนดตำแหน่งโดยคู่ลำดับ 3 มิติ (X,Y,Z) หรือคู่ลำดับโฮโมจีเนียส (X,Y,Z,W) (ปกติ W=1) ที่เรียกว่า "เวอร์เท็กซ์" (Vertex) ซึ่งใช้ 1 vertex สำหรับจุด 2 vertex สำหรับเส้น และมากกว่า 2 vertex สำหรับรูปเหลี่ยม OpenGL ได้กำหนดลักษณะของรูปเหลี่ยมที่เหมาะสมโดยใช้ข้อบังคับ 3 ข้อคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ขอบของรูปเหลี่ยมจะต้องไม่ตัดกันเอง
2. รูปทรงของรูปเหลี่ยมจะต้องไม่แหว่งเว้าเข้าไป
3. จะต้องไม่มีรูภายในรูปเหลี่ยม

แต่ไม่มีข้อกำหนดสำรับจำนวนมุมของรูปเหลี่ยม ข้อบังคับนี้ทำให้การแสดงผลมีความรวดเร็วและหากใช้รูปเหลี่ยมที่ไม่เหมาะสมก็อาจทำให้การเรนเดอร์ภาพผิดพลาดได้ อย่างไรก็ตามรูปเหลี่ยมที่ไม่เหมาะสมสามารถแบ่งเป็นรูปเหลี่ยมย่อยๆ ได้ โดยผู้ใช้หรือใช้ไลบรารี GLU ซึ่งจะมีส่วนของคำสั่งในการตัดแบ่งรูปเหลี่ยมที่ซ้อนทับด้วย Tessellation

3.5 การแปลงทางเรขาคณิต

สิ่งที่ทำให้คอมพิวเตอร์กราฟฟิกส์มีประโยชน์อย่างมากในการสร้างภาพก็คือ ความง่ายดายสะดวกสบายในการเปลี่ยนแหลงภาพที่เราวาดขึ้นมาแล้ว เช่น เปลี่ยนสเกลของกราฟ เปลี่ยนมุมมองขอบแบบตึกที่ออกแบบไว้หรือการเปลี่ยนขนาดภาพของแผนที่ เป็นต้น ทั้งหมดนี้สามารถทำงานได้ง่าย รวดเร็ว โดยใช้คอมพิวเตอร์กราฟฟิกส์ เพราะข้อมูลของรูปภาพต่างๆ หรือข้อมูลทางกราฟฟิกส์ได้ถูกจัดเก็บไว้ในคอมพิวเตอร์ เพียงแค่เปลี่ยนแปลงวิธีการแสดงภาพเหล่านี้ออกทางจอภาพเท่านั้น

3.5.1 การเลื่อนตำแหน่ง (Translation)

การเลื่อนตำแหน่ง คือการเปลี่ยนตำแหน่งของออปเจ็คไปในแนวเส้นตรงจากตำแหน่งหนึ่งไปอีกตำแหน่งหนึ่ง หรืออาจกล่าวได้ว่าเป็นการเปลี่ยนตำแหน่งจากโคออร์ดิเนตหนึ่งไปยังโคออร์ดิเนตอื่นนั่นเอง ตำแหน่งออปเจ็คที่เกิดจากการเลื่นจะต้องสัมพันธ์กับขนาดจอภาพ ทำให้ออปเจ็คอาจจะอยู่ในจอภาพ นอกจอภาพหรืออยู่ภายในบางส่วนก็ได้ เพราะขนาดของจอภาพทำให้เกิดการขริบภาพ (clipping)

$$x' = x + t_x, y' = y + ty \tag{3.1}$$

โดยสามารถเขียนอยู่ในรูปของ Matrix ได้เป็น

$$P' = \begin{bmatrix} x' \\ y' \end{bmatrix}, P = \begin{bmatrix} x \\ y \end{bmatrix}, T = \begin{bmatrix} t_x \\ t_y \end{bmatrix} \tag{3.2}$$

$$P' = P + T \tag{3.3}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.2 การย่อ/ขยายภาพ (Scaling)

การย่อ/ขยายภาพ คือการเลี่ยนขนาดของออปเจ็คให้มีขนาดใหญ่ขึ้นหรือเล็กลง ตำแหน่งออปเจ็คที่เกิดจากการย่อ/ขยาย อาจอยู่ในจอภาพ นอกจอภาพ หรืออยู่ในจอภาพเพียงบางส่วนเท่านั้นก็ได้ขึ้นอยู่กับการขริบภาพ

$$x' = x \cdot S_x, y' = y \cdot S_y \tag{3.4}$$

โดยสามารถเขียนอยู่ในรูปของ Matrix ได้เป็น

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \cdot \begin{bmatrix} x' \\ y' \end{bmatrix} \tag{3.5}$$

$$P' = S \cdot P \tag{3.6}$$

3.5.3 การหมุนภาพ (Rotation)

การหมุน คือการเปลี่ยนตำแหน่งของออปเจ็คไปในแนวเส้นโค้งของวงกลมในระนาบ xy ขนานกับโคออร์ดิเนตของแกน Z จากตำแหน่งหนึ่งไปอีกตำแหน่งหนึ่ง โดยไม่เปลี่ยนขนาดของออปเจ็คนั้น

$$\begin{aligned} x' &= r\cos(\phi+\theta) = r\cos\phi\cos\theta - r\sin\phi\sin\theta \\ y' &= r\sin(\phi+\theta) = r\cos\phi\sin\theta + r\sin\phi\sin\theta \end{aligned} \tag{3.7}$$

$$x = r\cos\phi, y = r\sin\phi \tag{3.8}$$

เมื่อแทนสมการที่ (3.8) ลงในสมการที่ (3.9)

$$\begin{aligned} x' &= x\cos\theta - y\sin\theta \\ y' &= x\sin\theta + y\cos\theta \end{aligned} \tag{3.9}$$

โดยสามารถเขียนอยู่ในรูปของ Matrix ได้เป็น

$$P' = R \cdot P \tag{3.10}$$

$$R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \tag{3.11}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 เมทริกซ์กับการแปลงทางเรขาคณิต

โปรแกรมกราฟฟิกส์ทั่วไปมักจะเกี่ยวข้องกับการแปลงทางเรขาคณิตหลายครั้ง หลายรูปแบบ ตัวอย่างเช่น ในการสร้างภาพเคลื่อนไหวจำนวนมาก หรือหมุนออปเจ็คไปยังตำแหน่งต่างๆ ในทุกๆ เฟรม

ในการแปลงทางเรขาคณิตพื้นฐานทั้งสามแบบ (การเลื่อน, การย่อ/ขยาย และการหมุน) สามารถจัดรูปแบบเป็นสมการทั่วไปได้เป็น

$$P' = M1 \cdot P + M2 \tag{3.12}$$

โดยที่ P แทนเวกเตอร์ตำแหน่งจุดเริ่มต้น

P' แทนเวกเตอร์ตำแหน่งสิ้นสุด

M1 เป็นเมทริกซ์ขนาด 2x2 ซึ่งเก็บค่าสัมประสิทธิ์การคูณ (Rotation, Scaling)

M2 เป็นเมทริกซ์ขนาด 2x1 ซึ่งเก็บพจน์ที่นำไปบวก (Translation)

การทำการแปลงด้วยสมการ (3.12) หลายครั้ง จะทำให้เกิดข้อผิดพลาดสะสมสำหรับตัวเลขจำนวนเต็ม ดังนั้นจึงจำเป็นต้องรวมพจน์ M1 และ M2 เข้าด้วยกัน หลักการของวิธีนี้ คือการแปลงทางเรขาคณิตในรูปของการคูณกันของเมทริกซ์ซึ่งทำได้โดยขยายพจน์ที่เป็นเมทริกซ์ ขนาด 2x2 -> 3x3 และ 2x1 -> 3x1 ตามลำดับ ซึ่งทำได้โดยจัดโคออร์ดิเนต (x,y) ในรูปเทอม (x_h, y_h, h) ซึ่งเรียกว่า "โฮมจีเนียสโคออร์ดิเนต" (Homogeneous Coordinates) โดยค่าพารามิเตอร์ h จะไม่เท่ากับ 0 โดยที่

$$x = \frac{x_h}{h}, y = \frac{y_h}{h} \tag{3.13}$$

ดังนั้นโฮมโมจีเนียสโคออร์ดิเนตอาเขียนใหม่เป็น $(x \cdot h, y \cdot h, h)$ โดยทั่วไป เราสามารถเลือกค่า h เป็นจำนวนจริงบวกใดๆ ได้แต่เพื่อความสะดวกมักจะเลือกให้ h = 1 ซึ่งจะได้โคออร์ดิเนตเป็น (x,y,1) สามารถเขียน Matrix ทั้ง 3 ใหม่ได้เป็น

การเลื่อน

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{3.14}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การย่อ/ขยาย

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{3.15}$$

การหมุน

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{3.16}$$

3.7 ภาพรวมแนวคิดการมองภาพ 3 มิติ

เมื่อมีการแสดงรูปทรง 3 มิติบนจอภาพ แต่ละออปเจ็คบนจอภาพถูกกำหนดเป็นชุดของพื้นผิวปิดที่ประกอบรวมกันเป็นออปเจ็ค ในบางแอปพลิเคชันอาจจะต้องกำหนดข้อมูลโครงสร้างภายในของออปเจ็คเพิ่มเติม นอกจากโปรซีเดอร์สำหรับการสร้างมุมมองพื้นผิวของออปเจ็คแล้ว ในบางครั้งกราฟฟิกส์แพ็กเกจจะต้องมีรูทีนในการแสดงโครงสร้างภายใน หรือมุมมองภาคตัดขวางของออปเจ็คทึบ ฟังก์ชันการมองจะแสดงรายละเอียดของออปเจ็คเป็นชุดของโปรซีเดอร์ที่โปรเจ็กต์มุมมองของออปเจ็คเพื่อกำหนดลงบนอุปกรณ์แสดงผล มีหลายขั้นตอนในการมองภาพ 3 มิติ เช่น รูทีนการขริบภาพ ที่เหมือนกับการมองภาพ 2 มิติ แต่ในการมองภาพ 3 มิติมีงานบางอย่างที่ไม่ได้กำหนดจอภาพการมองเห็น และสำหรับการแสดงผลให้สมจริงจะต้องมีการใช้การตกกระทบของแสง และใช้คุณสมบัติของพื้นผิวประกอบด้วย

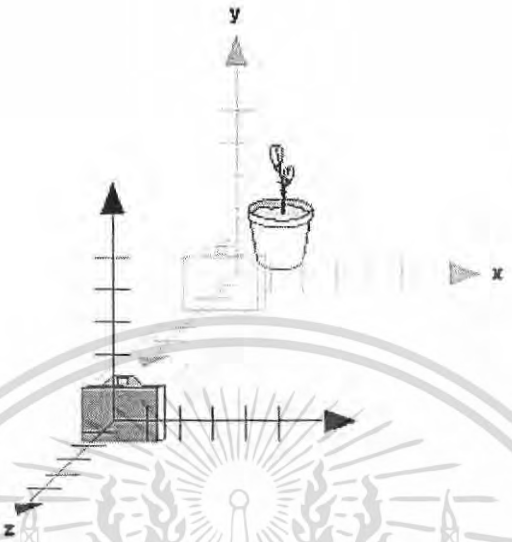
การแสดงภาพ 3 มิติ ในขั้นแรกต้องกำหนดการอ้างอิงโคออร์ดิเนตสำหรับพารามิเตอร์การมองเห็น หรือเรียกว่า "กล้อง" การอ้างอิงโคออร์ดิเนตนี้จำกำหนดตำแหน่งและแนวสำหรับระนาบการมอง (view plane หรือ projection plane) ที่สัมพันธ์กับกล้องถ่ายภาพ หลังจากนั้นรายละเอียดของออปเจ็คจะถูกแปลงไปยังโคออร์ดิเนตอ้างอิงการมอง แล้วจึงโปรเจ็กต์ภาพนั้นไปยังบนระนาย ซึ่งสามารถสร้างมุมมองของออปเจ็คบนอุปกรณ์แสดงผลในแบบของเอาต์ไลน์ได้ หรือจะประยุกต์ใช้เทคนิคของแสงและเงา รวมทั้งการเรนเดอร์พื้นผิวเพื่อให้ภาพเกิดเป็นภาพสมจริงมากยิ่งขึ้นได้

3.8 โปรเจ็กต์ชั่น (Projection)

สิ่งที่แตกต่างจากกล้องถ่ายภาพก็คือ สามารถเลือกวิธีโปรเจ็กต์ชันภาพลงบนระนาบการมอง วิธีการหนึ่งที่สามารถให้รายละเอียดของออปเจ็คทึบบนระนาบแสดงผล คือโปรเจ็กต์ชั่นพื้นผิวของออปเจ็คแบบขนานซึ่งเรียกเทคนิคนี้ว่า โปรเจ็กต์ชันแบบขนาน (parallel projection) ที่ใช้กัน

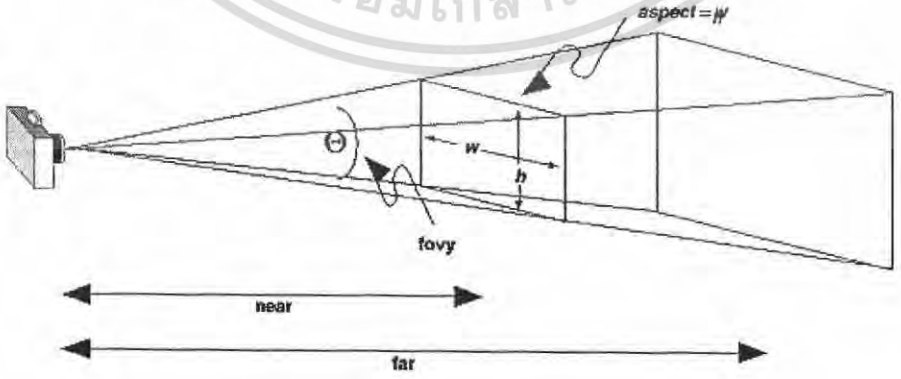
เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บ่อยในการวาดทางวิศวกรรมและสถาปัตยกรรม เพื่อแสดงออปเจ็คด้วยชุดของมุมมองให้แสดงออปเจ็คได้อย่างเที่ยงตรง



รูปที่ 3.2 การอ้างอิงโคออร์ดิเนตสำหรับการมองของภาพ 3 มิติ

วิธีการอื่นในการสร้างภาพ 3 มิติคือโปรเจ็กต์ชันไปยังระนาบการมองตามเส้นทางที่เบนเข้าหากันโปรเจ็กต์ชันนี้เรียกว่า โปรเจ็กต์ชันแบบเพอร์สเปกทีพ (perspective projection) ทำให้ออปเจ็คที่มีขนาดเท่ากัน แต่อยู่ตำแหน่งที่ไกลจากตำแหน่งการมองจะแสดงขนาดที่เล็กกว่าออปเจ็คที่อยู่ใกล้กับตำแหน่งการมอง ภาพที่สร้างจากโปรเจ็กต์ชันแบบเพอร์สเปกทีพนี้จะให้ความสมจริงมากกว่า เนื่องจากเป็นวิธีการเดียวกับการสร้างภาพของกล้องและตาของเรา



รูปที่ 3.3 มุมมองของการมองภาพแบบเพอร์สเปกทีพ

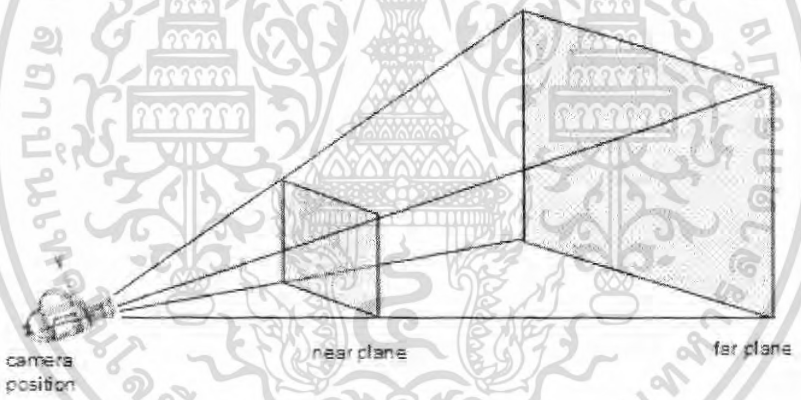
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.9 โคออร์ดิเนตการมองภาพ 3 มิติ

การสร้างเฟรมอ้างอิงการมองภาพ 3 มิติเหมือนกับการกำหนดเฟรมอ้างอิงการมองภาพ 2 มิติ ขั้นแรกต้องเลือกเอาตำแหน่งโคออร์ดิเนตทางกายภาพ $P_0 = (X_0, Y_0, Z_0)$ สำหรับจุดกำเนิดการมองซึ่งเรียกว่า จุดมอง (view point) หรือตำแหน่งการมอง (viewing position) จากนั้นจะมี view up vector (V) ซึ่งเป็นทิศทางสำหรับภาพ 3 มิติ

3.10 การทำ Frustum Culling

ในการปรับแต่งมุมมองของกล้องมุมต่างๆ สามารถทำการปรับต่างได้จากการใช้ชุดคำสั่ง gluPerspective และ gluLookAt เพื่อเป็นตัวกำหนดตัดสินในมุมมองที่เห็น มุมมอง frustum เป็นปริมาณของพื้นที่ในลักษณะเพอร์สเปกทีพ หรือเป็นรูปทรงปีรามิดตัดยอดที่จะแสดงภาพให้มองเห็นได้ในจอภาพ ค่าๆนี้สามารถหาค่าได้จากการนำรูปแบบการกำหนดมุมมองของกล้องมาคำนวณหาพื้นที่



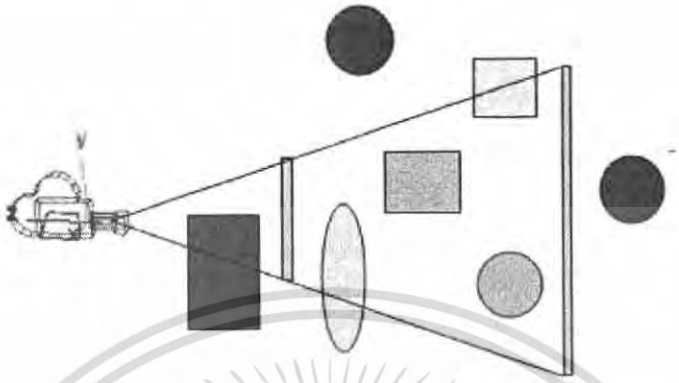
รูปที่ 3.4 ปริมาณ frustum กับค่าความสัมพันธ์ของมุมกล้อง

จากรูปที่ 3.4 ยอดแหลมของปีรามิดจะเป็นตำแหน่งของกล้องและฐานของปีรามิดจะเป็น far plane ที่เป็นตัวกำหนดระยะมองเห็นไกลได้สูงสุด รูปทรงปีรามิดตัดจะถูกตัดยอดโดย near plane ซึ่งเป็นระยะการมองเห็นใกล้สุดที่สามารถแสดงได้ โดยในพื้นที่ดังกล่าวนี้จะเรียกได้ว่าเป็นพื้นที่ frustum

วัตถุภายใน frustum ทุกๆ วัตถุสามารถที่จะถูกแสดงได้ภายในจอภาพของอุปกรณ์เอาท์พุท และบางส่วนที่ไม่ได้อยู่ภายใน frustum นี้จะไม่สามาถมองเห็นวัตถุนั้นได้ จากรูปที่ 3.5 จะเป็นตัวอย่างของวัตถุ ณ. บริเวณต่างๆ ที่อยู่ทั้งภายในและภายนอก frustum โดยวัตถุที่สามารถมองเห็นได้จะถูกแทนด้วยวัตถุที่เป็นสีอ่อน หรือสังเกตได้จากจะมีพื้นที่บางส่วนหรือทั้งหมดอยู่ภายใน

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปทรงปีรามิดตัด และวัตถุที่ไม่สามารถมองเห็นได้จะถูกแทนด้วยวัตถุที่มีสีเข้ม หรือสังเกตได้ว่าพื้นที่ของวัตถุทั้งหมดนั้นอยู่ภายนอกรูปทรงปีรามิดตัด



รูปที่ 3.5 ตัวอย่างการแสดงวัตถุที่ตำแหน่งต่างๆ ของfrustum

การทำ frustum culling มีจุดมุ่งหมายหลักคือการระบุว่าวัตถุใดๆ นั้นมีตำแหน่งอยู่ภายในหรือภายนอก frustum และจะตัวทุกๆ วัตถุที่ไม่ได้อยู่ภายใน frustum วัตถุที่อยู่ภายใน frustum เท่านั้นที่จะสามารถส่งข้อมูลและนำไปวาดออกสู่เอาท์พุทได้ โดยวัตถุที่ได้ส่งข้อมูลเพื่อนำไปแสดงผลทางเอาท์พุทจะต้องมีขั้นตอนการทำงานหรือโปรเซสเพื่อให้สามารถแสดงผลได้ หากสามารถตัดสินใจเลือกเฉพาะวัตถุที่มองเห็นได้เท่านั้น ที่จะสามารถส่งข้อมูลไปยังเอาท์พุทได้จะเป็นการประหยัดกระบวนการทำงานที่ไม่จำเป็น ทำให้สามารถแสดงผลทางเอาท์พุทได้อย่างมีประสิทธิภาพมากขึ้น อีกทั้งยังเป็นการประหยัดพื้นที่หน่วยความจำของกราฟฟิกส์การ์ดและทำให้แอปพลิเคชันสามารถทำงานได้รวดเร็วขึ้นอีกด้วย

3.11 ค่าที่ใช้คำนวณ Frustum Culling

จากการกำหนดค่ามุมกล้องโดยใช้ฟังก์ชั่นในหัวข้อ 3.7 สามารถกำหนดตัวอย่างค่าตัวแปรได้ดังนี้

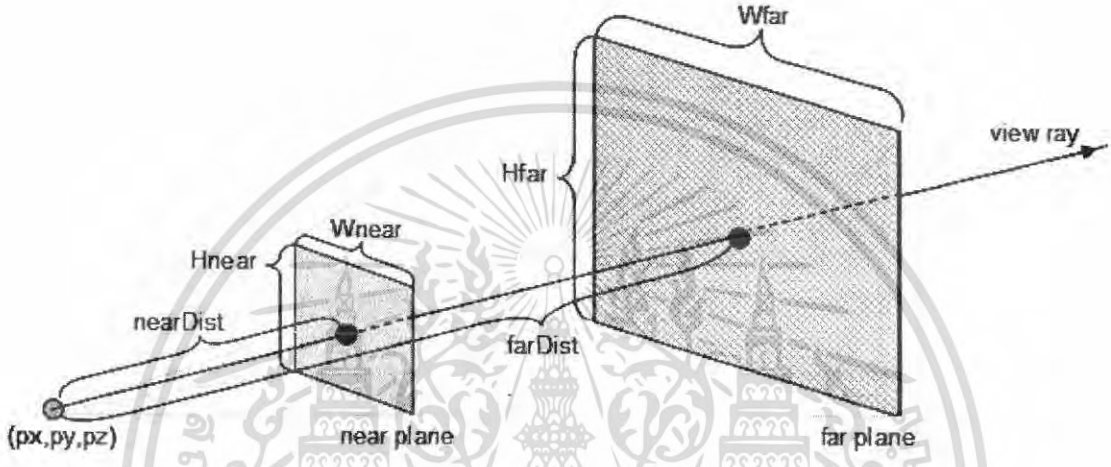
- gluPerspective(fov, ratio, nearDist, farDist);
- gluLookAt(px,py,pz, lx,ly,lz, ux,uy,uz);

โดยที่

fov	คือ	องศาการมองจากตัวกล้อง
ratio	คือ	อัตราส่วนของความกว้างต่อสูงของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

nearDist	คือ	ระยะห่างจากกล้องถึง near plane
farDist	คือ	ระยะห่างจากกล้องถึง far plane
px,py,pz	คือ	ตำแหน่งของกล้องบนโคออร์ดิเนต x,y,z
lx,ly,lz	คือ	ตำแหน่งหรือจุด focus ของกล้องบนโคออร์ดิเนต x,y,z
ux,uy,uz	คือ	เวกเตอร์ของการตั้งกล้อง



รูปที่ 3.6 ค่าพารามิเตอร์ต่างๆ ที่ใช้คำนวณ frustum(1)

จากรูปที่ 3.6 เราค่าสามารถคำนวณหาค่าต่างๆ เพิ่มได้โดยใช้สูตรพื้นฐานทางคณิตศาสตร์ ซึ่งจะเริ่มต้นจากหาตำแหน่งของ near plane และ far plane ทั้งคู่

$$Hnear = 2 * \tan(fov / 2) * nearDist \quad (3.17)$$

$$Wnear = Hnear * ratio \quad (3.18)$$

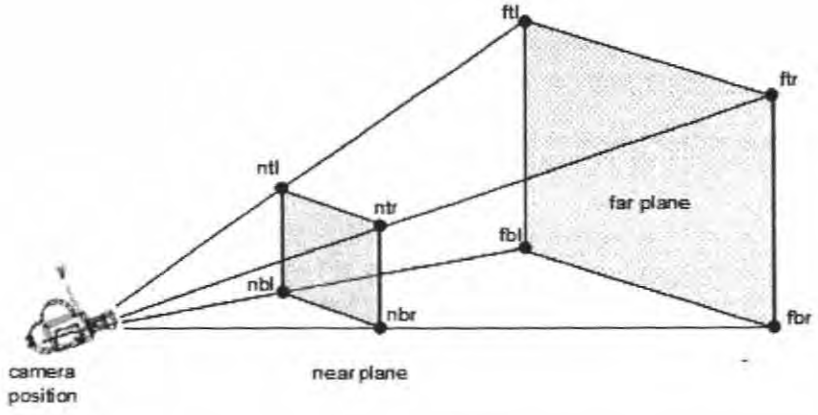
$$Hfar = 2 * \tan(fov / 2) * farDist \quad (3.19)$$

$$Wfar = Hfar * ratio \quad (3.20)$$

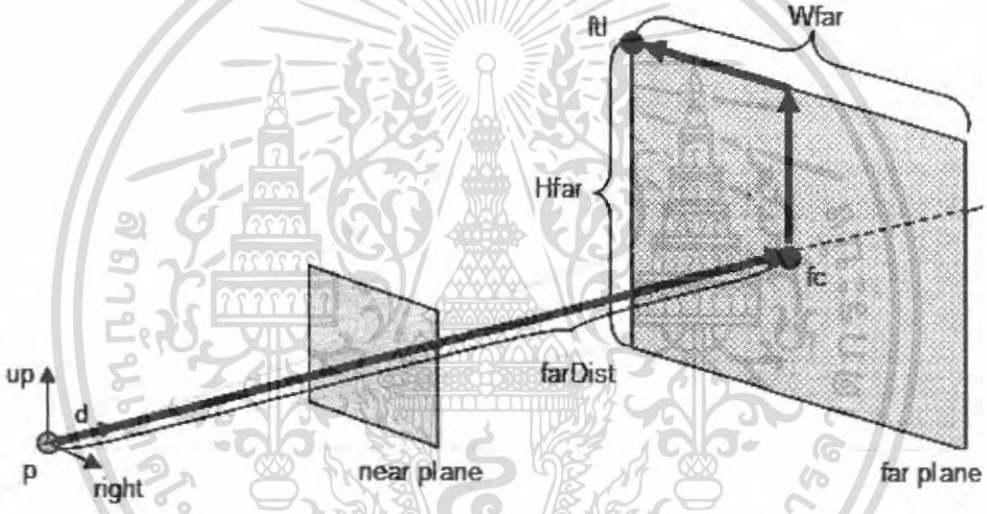
โดยที่

Hnear	คือ	ความสูงของ near plane
Wnear	คือ	ความกว้างของ near plane
Hfar	คือ	ความสูงของ far plane
Wfar	คือ	ความกว้างของ far plane

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 ค่าพารามิเตอร์ต่างๆ ที่ใช้คำนวณ frustum(2)



รูปที่ 3.8 ค่าพารามิเตอร์ต่างๆ ที่ใช้คำนวณ frustum(3)

$$fc = p + d * farDist \quad (3.21)$$

$$ftl = fc + (up * Hfar/2) - (right * Wfar/2) \quad (3.22)$$

$$ftr = fc + (up * Hfar/2) + (right * Wfar/2) \quad (3.23)$$

$$fbl = fc - (up * Hfar/2) - (right * Wfar/2) \quad (3.24)$$

$$fbr = fc - (up * Hfar/2) + (right * Wfar/2) \quad (3.25)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$nc = p + d * nearDist \quad (3.26)$$

$$ntl = nc + (up * Hnear/2) - (right * Wnear/2) \quad (3.27)$$

$$ntr = nc + (up * Hnear/2) + (right * Wnear/2) \quad (3.28)$$

$$nbl = nc - (up * Hnear/2) - (right * Wnear/2) \quad (3.29)$$

$$nbr = nc - (up * Hnear/2) + (right * Wnear/2) \quad (3.30)$$

โดยที่

p	คือ	ตำแหน่งกล้อง
d	คือ	ยูนิตเวกเตอร์ของมุมมองกล้องไปยังจุดโฟกัส
fc	คือ	far plane center
ftl	คือ	far plane top left
ftr	คือ	far plane top right
fbl	คือ	far plane bottom left
fbr	คือ	far plane bottom right
nc	คือ	near plane center
ntl	คือ	near plane top left
ntr	คือ	near plane top right
nbl	คือ	near plane bottom left
nbr	คือ	near plane bottom right

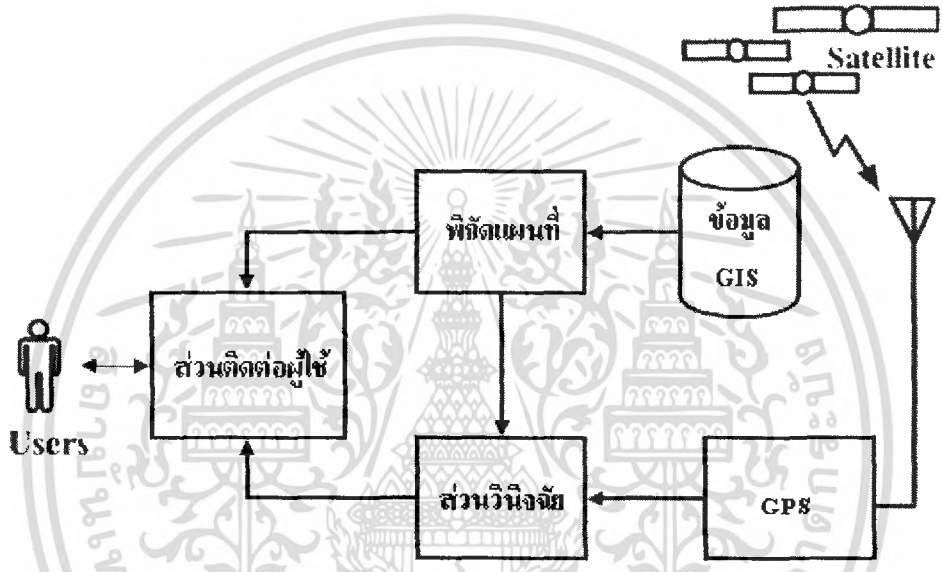
จากการคำนวณหาตำแหน่งต่างๆของ plane จะนำค่าที่ได้นี้มาคำนวณต่อหาสมการของ plane ทรงปิรามิดปิด เพื่อใช้สมการ plane นี้ตัดวัตถุที่อยู่นอก plane ทั้ง 6 ด้าน ซึ่งจะเหลือเพียงเฉพาะวัตถุที่อยู่ภายใน plane ที่จะทำการส่งค่าและนำไปประมวลผลออกเอาท์พุท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การออกแบบโครงสร้างของระบบ

4.1 การออกแบบโครงสร้างของระบบนำทางรถยนต์

โครงสร้างของระบบนำทางรถยนต์และส่วนบุคคล จะประกอบไปด้วย 5 ส่วนคือ ส่วน GPS, ส่วนวินิจฉัย, ส่วนข้อมูล GIS, ส่วนพิกัดแผนที่, ส่วนติดต่อผู้ใช้



รูปที่ 4.1 ภาพแสดงโครงสร้างของระบบ

4.2 ส่วน GPS

ส่วน GPS เป็นส่วนที่ทำหน้าที่ในการติดต่อสัญญาณกับดาวเทียมเพื่อรับค่าพิกัด ละติจูด ลองติจูด แล้วส่งค่าที่รับจากสัญญาณดาวเทียมซึ่งเป็น Format string $GPRMC ให้กับส่วนวินิจฉัย โดยข้อมูลจาก Format GPRMC มีดังนี้

$GPRMC,140745.000,A,1343.7523,N,10046.5119,E,2.24,290.37,200807,,*03<CR><LF>

เซตข้อมูลที่ใช้งานในประโยค RMC ประกอบไปด้วย

เวลา = 14:07:45 UTC

ละติจูด = 1343 ลิปดา กับ 7523 ฟิลิปดา

N/S = เป็น ละติจูด ทางเหนือ (N)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลองติจูด = 10046 ลิปดากับ 5119 ฟิลิปดา

E/W = เป็น ลองติจูด ทางตะวันออก (W)

ความเร็ว = 2.24 น็อต

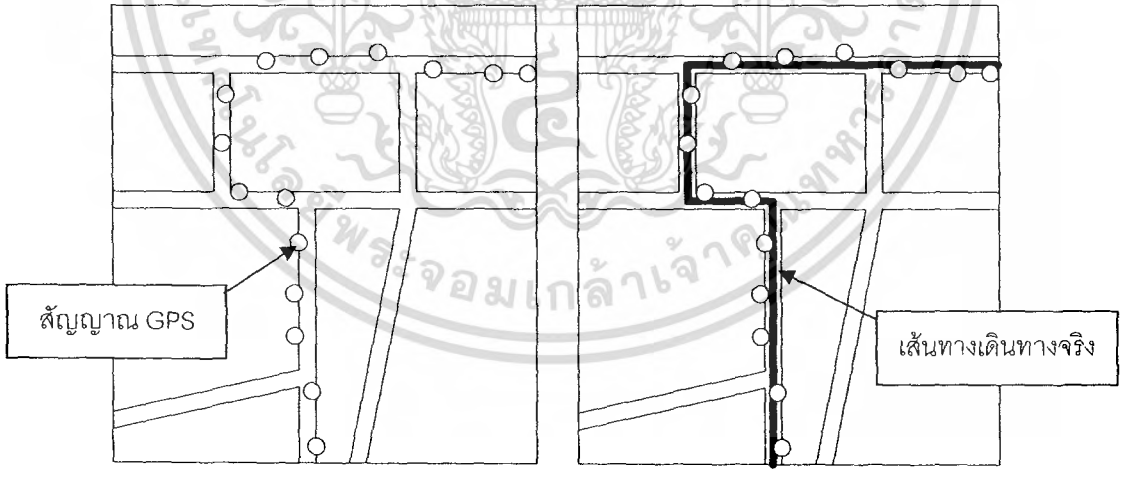
องศา = 290.37 องศา ทิศทางอ้างอิงจากขั้วโลกเหนือ

วันที่ = วันที่ 20 เดือน สิงหาคม 2007

ส่วนของระบบ GPS จะใช้เฉพาะข้อมูลการระบุตำแหน่งของสถานที่ใช้ข้อมูลส่วนบอกตำแหน่ง ละติจูด และ ลองติจูด โดยทำการกรองเอาจากข้อมูล String ที่ GPS รับได้ตามคอลัมน์ที่ 3 และ 5 โดยใช้เครื่องหมายลูกน้ำเป็นตัวแบ่งคอลัมน์ (,) จากข้อมูล องศาละติจูดลองติจูดที่ได้รับนี้จะยังไม่สามารถนำไประบุตำแหน่งในพิกัดแผนที่จากข้อมูล GIS ได้เพราะว่าข้อมูลจาก GIS ที่ได้มาเก็บข้อมูลอยู่ในระบบพิกัด UTM ซึ่งต้องทำการเปลี่ยนข้อมูลจาก ละติจูดลองติจูดให้เป็น พิกัด UTM เสียก่อนเพื่อที่จะได้อยู่ในรูปแบบเดียวกัน

4.3 ส่วนวินิจฉัย

เครื่องรับ GPS จะรับพิกัดตำแหน่งที่ประกอบด้วยละติจูดและลองติจูดและทำการเปลี่ยนให้เป็นพิกัด UTM ซึ่งแทนด้วยพิกัด (X,Y) แล้ว นำพิกัดตำแหน่งส่งมาให้ส่วนวินิจฉัยเพื่อนำไปใช้แสดงผลบนแผนที่แสดงได้ตามรูปที่ 4.2 (ก)



รูปที่ 4.2 แสดงตำแหน่งของรถบนถนน

รูปที่ 4.2 (ก) แสดงพิกัดตำแหน่งของรถที่วิ่งอยู่บนถนนที่รับมาจากเครื่องรับ GPS ในทุกๆ 1 วินาที เมื่อนำมาแสดงเป็นเส้นทางการเคลื่อนที่ของรถบนแผนที่ จะพบว่าเส้นทางวิ่งของรถนั้นคลาดเคลื่อนไปจากของแผนที่ ทั้งนี้อาจเกิดจากความผิดพลาดของเครื่องรับ GPS เอง และความ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผิดพลาดในการระบุตำแหน่งของดาวเทียมซึ่งได้กล่าวมาแล้ว รวมทั้งความคลาดเคลื่อนของแผนที่ด้วย จึงทำให้ตำแหน่ง (X,Y) ที่ได้จาก GPS คลาดเคลื่อนไปจากตำแหน่งจุดเดียวกันบนแผนที่ซึ่งควรจะเป็นจุดบนเส้นทางบนถนน สำหรับในรูปที่ 4.2 (ข) นั้นแสดงเส้นทางวิ่ง ของรถที่ควรจะเป็นบนแผนที่ โดยในแผนที่จะอ้างอิง ตาม Path (เส้น ทางที่เป็นส่วนๆของถนน) ซึ่ง หลายๆ Path ก็จะประกอบกันเป็นถนน

ดังนั้นในการแสดงเส้นทางวิ่งของรถให้ได้ถูกต้องบนแผนที่จากข้อมูลตำแหน่งของ GPS เราจะต้องทำการเปรียบเทียบ ข้อมูลทั้งสองเพื่อดูความสอดคล้องในเรื่องของตำแหน่งและทิศทางการเปลี่ยนแปลงตำแหน่ง โดยอาศัยการกำจัดสัญญาณรบกวน (Filter Noise) แล้ว พิจารณารหัสลูกโซ่ (Chain Code) เพื่อหาทิศทางของรถที่วิ่ง จากนั้นนำผลลัพธ์ที่ได้ Projection ลงบนแผนที่ด้วยวิธี Snap ซึ่งเป็นอัลกอริธึมที่หาจุดตั้งฉากบน path ที่ใกล้ที่สุดโดยข้อมูล path จากตำแหน่งบนแผนที่จริง

4.3.1 การกำจัดสัญญาณรบกวน

สาเหตุที่ต้องทำการกำจัดสัญญาณรบกวนเพราะว่าสัญญาณที่ได้จาก GPS มีความคลาดเคลื่อน สังเกตได้ว่าแม้เครื่องรับสัญญาณจะไม่ได้เคลื่อนที่แต่สัญญาณที่รับได้มีการเคลื่อนไหวเล็กน้อยซึ่งถือว่านั่นคือสัญญาณรบกวน วิธีการกำจัดสัญญาณรบกวนทำได้โดยการใช้วิธีเช็คค่าระยะห่าง (distance) ด้วยการนำเอาพิกัดตำแหน่ง (X1,Y1) และ (X2,Y2) ที่ผ่านขั้นตอนการหารหัสลูกโซ่มาแล้วจากนั้นก็นำมาเข้าสูตรเพื่อหาค่า ระยะห่าง (Distance)

$$Dis = \sqrt{(x_2 - x_1)^2 + (y - y_1)^2} \qquad (4.1)$$

โดยในขั้นตอนแรกต้องกำหนด ค่าระยะห่างที่ยอมรับได้เพื่อใช้ในการตรวจสอบ จากนั้นนำพิกัดตำแหน่ง (X1,Y1) และ (X2,Y2) มาคำนวณตาม สมการที่ 4.3 หากค่า ระยะห่าง (Distance) ที่คำนวณได้มีค่าต่ำกว่า ค่าการตรวจสอบที่กำหนดไว้ แสดงว่ายังไม่ถูกสัญญาณรบกวนก็ยังไม่ต้องทำอะไรแต่หากค่า ระยะห่าง (Distance) มีค่ามากว่า ค่าที่กำหนดไว้ ก็จะสามารถนำสัญญาณนั้นส่งไปส่วนของการวิเคราะห์ทิศทางใน รหัสลูกโซ่ต่อไปได้

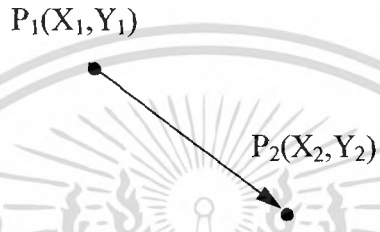
ภายในโครงงานนี้จะกำหนดค่าระยะห่างที่ใช้ในการ Filter สัญญาณรบกวนของ GPS ไว้ที่ 10 เมตร ซึ่งเป็นค่าความผิดพลาดเฉลี่ยของเครื่องรับสัญญาณ GPS ที่ใช้ในการทดลอง คือระหว่า 5-10 เมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2 รหัสลูกโซ่ (Chan Code)

เนื่องจากข้อมูลค่าพิกัดที่รับมาจาก GPS อาจจะมีความคลาดเคลื่อนจากกรณีต่างๆ ดังที่ได้กล่าวมาแล้วในบทที่ 2 ดังนั้นเราจึงต้องนำ รหัสลูกโซ่มากำจัดสัญญาณรบกวนจากนั้นใช้รหัสลูกโซ่ในการหาพฤติกรรมของรถว่ารถมีการเคลื่อนที่ในลักษณะใดเช่น โดยใช้รหัสลูกโซ่ 16 ทิศ ซึ่งเป็นวิธีการตรวจสอบทิศทางการเคลื่อนที่ของรถว่าเคลื่อนที่ไปตามแนวทิศทางใดโดยมีขั้นตอนดังนี้

คำนวณหาเวกเตอร์จากจุดค่า ตำแหน่ง (X1,Y1) กับค่าตำแหน่ง (X2,Y2) การที่เราจะคำนวณหาค่ารหัสลูกโซ่ได้ต้องมีค่าพิกัด ตั้งแต่ ตำแหน่ง 2 พิกัด ขึ้นไปโดยรับค่ามาจาก GPS

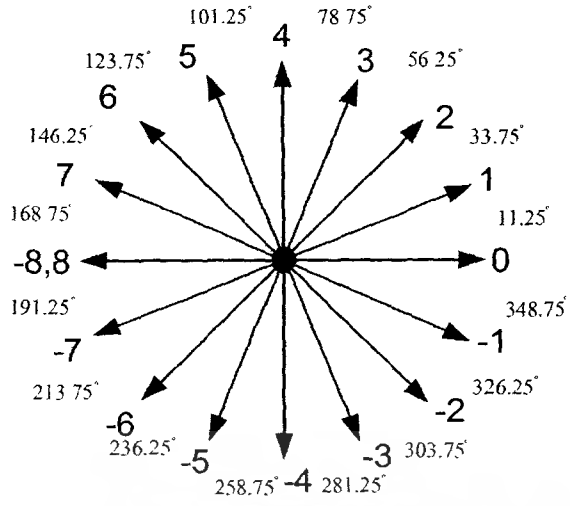


รูปที่ 4.3 แสดงการเคลื่อนที่ของรถจาก P_1 ไป P_2

$$\theta = \tan^{-1}\frac{(Y_2 - Y_1)}{(X_2 - X_1)} \qquad (4.2)$$

เมื่อได้ค่ามุมของเวกเตอร์ที่คำนวณจาก สมการที่ 4.1 มาแล้วก็จะนำค่ามุมไปทำการเปรียบเทียบเพื่อเปลี่ยนเป็นรหัสลูกโซ่ โดยรหัสลูกโซ่ (Chain Code) จะมี 16 ทิศ โดยแต่ละทิศจะแทนด้วย รหัสลูกโซ่ตัวเลขระหว่าง –8 ถึง 0 และ 1 ถึง 8 โดยเราจะกำหนดให้ค่ารหัสลูกโซ่เริ่มต้นที่ 0 เมื่อเคลื่อนที่ไปในทิศ ทางทวนเข็มนาฬิกาค่ารหัสลูกโซ่จะเพิ่มขึ้น แต่เมื่อเคลื่อนที่ไปในทิศทางตามเข็มนาฬิกาค่ารหัสลูกโซ่จะลดลง โดยในแต่ละทิศทางจะมีค่าช่วงของมุมอยู่ระหว่าง 0 ถึง 22.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 แสดงค่ามุมสำหรับรหัสลูกโซ่ 16 ทิศ

ตารางที่ 4.1 แสดงตาราง เปรียบเทียบค่ามุม ของรหัสลูกโซ่

Chain Code	ค่ามุมใน Chain Code (องศา)
0	348.75 – 11.25
1	11.25 – 33.75
2	33.75 – 56.25
3	56.25 – 78.75
4	78.75 – 101.25
5	101.25 – 123.75
6	123.75 – 146.25
7	146.25 – 168.75
8,-8	168.75 – 191.25
-7	191.25 – 213.75
-6	213.75 – 236.25
-5	236.25 – 258.75
-4	258.75 – 281.25
-3	281.25 – 303.75
-2	303.75 – 326.25
-1	326.25 – 348.75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช่น ถ้ามีค่ามุมของเวกเตอร์ที่คำนวณได้จากขั้นตอนที่ 2 มีค่าเท่ากับ 144 องศา รหัสลูกโซ่ที่ได้จะมีค่าเท่ากับ 6 ซึ่งค่า 6 เป็นค่าของรหัสลูกโซ่ที่เกิดจากค่ามุมที่อยู่ระหว่าง 123.75 องศา ถึง 146.25 องศา ดังตารางที่ 4.1

นำรหัสลูกโซ่ที่ต่อเนื่องกัน 2 อัน โดยสมมุติให้ค่ารหัสลูกโซ่ของเวกเตอร์ตัวแรกเป็น X และเวกเตอร์ตัวต่อมาเป็น Y มาหาค่าเบี่ยงเบนของรหัสลูกโซ่จากสมการ

$$Z = Y - X \qquad (4.3)$$

ซึ่งค่าเบี่ยงเบนการเคลื่อนที่นี้สามารถนำมาแปลความหมายเพื่อการวิเคราะห์หาพฤติกรรมของรถได้ โดยผลจากการคำนวณจะมีกรณีต่าง ๆ ดังต่อไปนี้

1. ถ้าเป็นการเคลื่อนที่ในทิศเดียวกันต่อเนื่องเป็นเส้นตรง ค่า Z จะมีค่าเท่ากับ 0 ซึ่งหมายความว่ารถเคลื่อนที่ตรงไปข้างหน้าไม่มีการเลี้ยว ตามรูปที่ 4.5(ก)

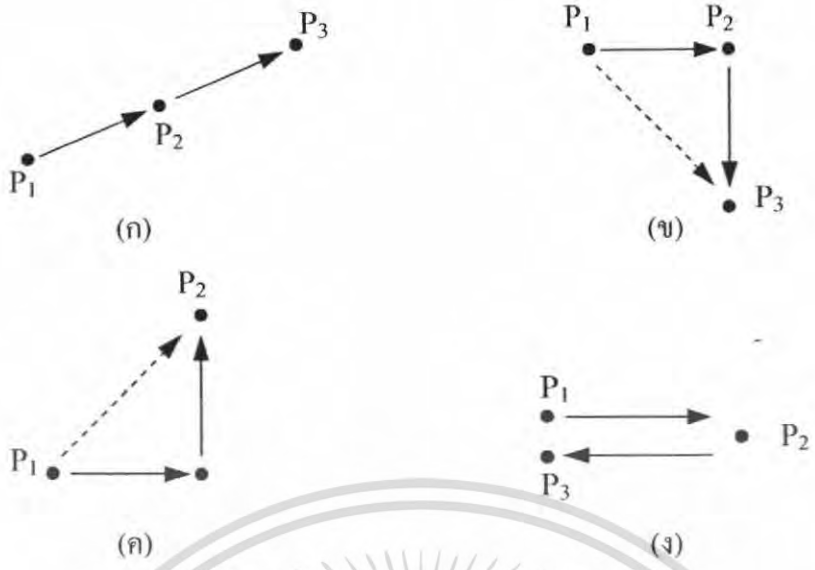
2. ถ้าเป็นการเลี้ยวขวา ค่า Z ที่คำนวณได้จะมีค่าเท่ากับ –4 ซึ่งหมายถึงขณะนั้นรถกำลังเคลื่อนที่ไปทางขวาหรือรถกำลังเลี้ยวขวานั้นเอง ตามรูปที่ 4.5(ข)

3. ถ้าเป็นการเลี้ยวซ้าย ค่า Z ที่คำนวณได้จะมีค่าเท่ากับ +4 ซึ่งหมายถึงขณะนั้นรถกำลังเคลื่อนที่ไปทางซ้ายหรือรถกำลังเลี้ยวซ้ายนั้นเอง ตามรูปที่ 4.5(ค)

4. ถ้าเป็นการเคลื่อนที่ย้อนกลับ ค่า Z ที่คำนวณได้จะมีค่าเท่ากับ –7 หรือ +7 หรือ –8 หรือ +8 ซึ่งหมายถึงขณะนั้นรถกำลังเคลื่อนที่กลับทิศทางหรือสามารถแปลความหมายว่าเป็นการกลับรถนั้นเอง ตามรูปที่ 4.5(ง)

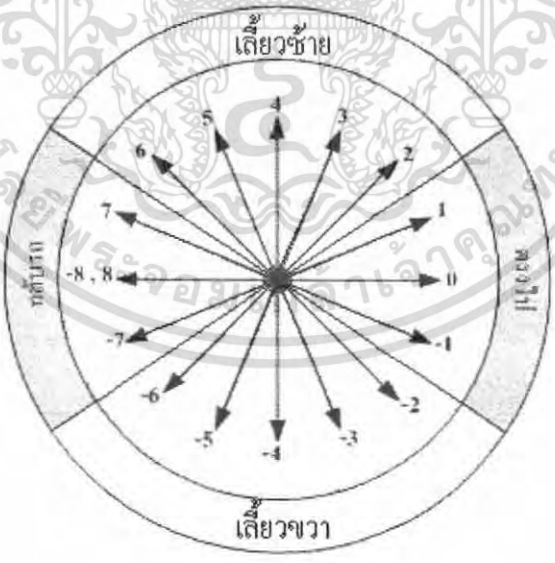
5. ถ้าผลของ Z มีค่าอยู่ระหว่าง 9 ถึง 15 ให้เอา 16 ไปลบออกหรือถ้าค่าอยู่ระหว่าง –15 ถึง –9 ก็ให้นำเอา 16 บวกเข้าไปทั้งนี้เพื่อเป็นการปรับค่า Z ให้อยู่ช่วง –8 ถึง 8 นั่นเองซึ่งการบวกหรือลบ 16 ไม่มีผลทำให้ค่า Z เปลี่ยนแปลงเพราะว่าค่า 16 มีค่าเสมือนกับการเบี่ยงเบนครบ 1 รอบ 360 องศา แต่เป็นการปรับค่า Z ให้เป็นตัวเลขที่สะดวกในการพิจารณาเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 แสดงการเบี่ยงเบนของรหัสลูกโซ่

ในการวิเคราะห์หาพฤติกรรมของรถจากการคำนวณหาค่าเบี่ยงเบนแล้วมาเปรียบเทียบเป็นพฤติกรรมนั้นสามารถสรุปได้ดังรูปที่ 4.6 ซึ่งในการใช้เทคนิคนี้ยังทำให้เราสามารถทราบถึงน้ำหนักการเลี้ยวได้อีกด้วย เช่น ถ้าค่าเบี่ยงเบนคำนวณได้เท่ากับ 4 แสดงว่าเป็นการเลี้ยวซ้าย 90 องศา หรือค่าเบี่ยงเบนมีค่าเท่ากับ 3 ก็เป็นการเลี้ยวซ้ายที่มีมุมการเลี้ยวไม่มากนัก



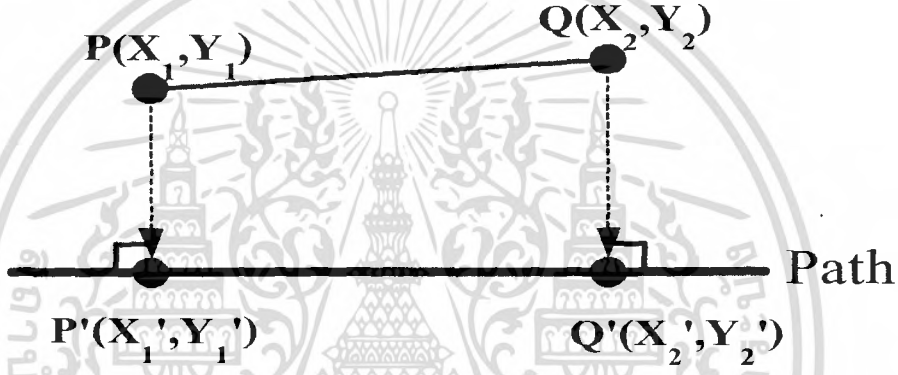
รูปที่ 4.6 การวิเคราะห์ค่าเบี่ยงเบน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าเป็นเส้นทางโค้ง ค่า Z จะเป็น ±1 ซึ่ง หมายถึงเป็นทางโค้งเล็กน้อย ส่วน ±2 หมายถึงเป็นทางโค้งปานกลาง และ ±3 เป็นทางโค้งมาก ค่าที่เป็นบวกหมายถึงโค้งทวนเข็ม (เลี้ยวซ้าย) ค่าที่เป็นลบหมายถึงโค้งตามเข็ม (เลี้ยวขวา) ส่วน ค่า ±7 และ ±8 หมายถึง การกลับรถ

4.3.3 วิธีการ Snap

วิธีการ Snap ในโครงงานนี้ใช้ในการหาจุดตั้งฉากบน path ของถนนที่ใกล้ที่สุดแล้วทำการเปรียบเทียบองศาที่ได้จากการคำนวณรหัสลูกโซ่ กับการคำนวณหาทิศทางของถนนที่ใกล้ที่สุดกับจุดที่ได้จากรหัสลูกโซ่ ทำให้จุดที่คำนวณสามารถ Snap ลงบนถนนนั้นได้จุดที่ใกล้เคียงความเป็นจริงมากที่สุดเพราะใช้ path ข้อมูลของถนนจากพิกัดแผนที่จริง



รูปที่ 4.7 แสดงตัวอย่างวิธีการ Snap

ในการจุดที่ใกล้ที่สุดกับของเส้นกับจุดที่นำมาจากรหัสลูกโซ่จะใช้การคำนวณทาง Vector กับเส้นถนนบริเวณใกล้เคียงทั้งหมดโดยจะมีการกำหนดระยะห่างการหาเส้นถนนไว้ที่ 500 เมตร ดังนั้นผลลัพธ์ที่ได้จากการหาจุดบนเส้นถนนนี้จะเป็นเส้นถนนที่ใกล้กับ จุดจากรหัสลูกโซ่ภายในรัศมี 500 เมตร และนำเอาถนนทั้งหมดนี้มาเปรียบเทียบกับทิศทางการเคลื่อนที่ของรถโดยเริ่มจากเส้นถนนที่มีรัศมีใกล้กับจุดจากรหัสลูกโซ่มาเปรียบเทียบก่อน

4.4 ส่วนข้อมูล GIS

ในงานโครงงานนี้จะใช้ข้อมูลแผนที่เป็นแผนที่ดิจิตอลที่มีมาตราส่วน 1:20,000 ครอบคลุมทุกพื้นที่ของกรุงเทพมหานครฯ ลักษณะข้อมูลในแผนที่ดิจิตอลเป็นข้อมูลแบบเวกเตอร์ โดยใช้จุดและเส้นในการแสดงลักษณะทางภูมิศาสตร์

Library Module ที่ช่วยในการติดต่อไฟล์ MapInfo นี้คือ GDAL (Geospatial Data Abstraction Library) ซึ่งเป็น Library ตัวหนึ่งที่ช่วยในการติดต่อ File Format ได้ทั้งแบบ Raster

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และ Vector โดยข้อมูลของแผนที่จะแบ่งออกเป็น Layer ซึ่งในแต่ละ Layer จะแสดงรายละเอียดเฉพาะที่เกี่ยวข้องกับ Layer นั้น เช่น หากมี Layer ที่เก็บข้อมูลเส้นทางของ ถนนหลัก ข้อมูลภายใน Layer นี้จะต้องเป็นถนนหลักเท่านั้น และภายใน Layer นี้จะประกอบไปด้วย ชนิดย่อยของการแสดงผลในแต่ละส่วน เช่น

POINT เป็นข้อมูลของจุดภายใน Layer ที่ข้อมูลนี้จะนิยมใช้ระบุสถานที่เช่นชื่อของสถานที่ หรือ ตำแหน่งสำคัญเป็น ต้น

LINESTRING เป็นข้อมูลของเส้นภายใน Layer โดยจะเป็นการรวมกลุ่มกันของจุดหลายๆ จุดตั้งแต่สองจุดขึ้นไป ข้อมูลของจุดใน LINESTRING นี้จะแสดงเป็นเส้นได้โดยการเริ่มลากเส้นผ่านจุดที่กำหนดต่อๆ กันโดยลากเส้นเป็นไปตามลำดับของข้อมูลจุด เพราะฉะนั้นหนึ่ง LINESTRING จะได้เส้นเพียงเส้นเดียวเท่านั้น หากต้องการแสดงเส้นหลายเส้นสามารถทำได้โดยใช้ LINESTRING หลายๆ ชุด

POLYGON เป็นข้อมูลของรูปหลายเหลี่ยมใน Layer ข้อมูลนี้จะแสดงในรูปของจุดหลายๆ จุดโดยจะเริ่มลากเส้นจากจุดแรกไปยังจุดสุดท้ายคล้ายๆ กับ LINESTRING ส่วนที่แตกต่างคือจะมีการเชื่อมโยงจากเส้นสุดท้ายไปยังเส้นแรกด้วย ได้รูปที่เป็นพื้นที่ปิดสามารถเติมสีลงไปในการแสดงผลชนิดนี้ได้

เนื่องจากในการใช้งานไลบรารี GDAL ยังจำเป็นต้องมีโปรเซสในการติดต่อใช้งานกับไฟล์ รวมถึงการอ่านและแปลงความหมายรูปแบบของไฟล์ ดังนั้นจึงจำได้ออกแบบรูปแบบข้อมูลที่อ่านได้จากข้อมูล MapInfo แล้วทำการสร้างไฟล์ข้อมูลที่เป็นรูปแบบของเฉพาะเพื่อทำให้สามารถเรียกใช้งานได้โดยไม่มีความซับซ้อน จากโครงงานได้ทำการเลือกเปลี่ยนไฟล์ MapInfo ให้อยู่ในรูปแบบของ Text File ซึ่งได้พัฒนาโปรแกรมที่ทำการเปลี่ยนโดยทำการอ่านเอาเฉพาะข้อมูลส่วนที่ต้องการจาก MapInfo และทำการสร้างไฟล์ Text ขึ้นมาใหม่โดยเขียนข้อมูลที่ต้องการลงไปในไฟล์นั้น

ขั้นตอนการใช้งาน GDAL for Python 2.4.2

1. ทำการ import osr จาก GDAL ในโปรแกรมที่พัฒนาเพื่อให้สามารถใช้งาน Module ของ GDAL ได้โดยใช้คำสั่ง from gdal import ogr
2. ทำการเปิดไฟล์ของ MapInfo โดยใช้ method Open() ของ ogr โดย method Open() นี้จะ return Class DataSource
3. จาก DataSource ที่ได้มา จะมีข้อมูลของ Map เป็น Layer ซึ่งอาจมีมากกว่า 1 layer โดยสามารถเลือกข้อมูลของแต่ละ layer ได้จาก Class นี้ โดยใช้ method GetLayer()
4. ภายในแต่ละ layer จะมีกลุ่มของ Feature อยู่ ตัวอย่างเช่น POLYGON LINESTRING ซึ่งเป็นประเภทของรูป Vector โดยในแต่ละชนิดจะมี Point ประกอบกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ในไฟล์ Text ที่สร้างจะประกอบไปด้วยรูปแบบดังนี้ Feature,point,point,point,….., รูปแบบที่แสดงสามารถนำไปใช้ได้โดยไม่จำเป็นต้องอ่านไฟล์ของ MapInfo อีก เนื่องจากได้ข้อมูลที่สำคัญต่อการสร้างแผนที่มาแล้ว

โครงสร้างของไฟล์ที่โครงงานนี้ใช้เก็บข้อมูล GIS มีลักษณะดังนี้

1. โครงสร้างจุด POINT, จุดX:จุดY, ชื่อจุด
2. โครงสร้างเส้น LINESTRING, จุดX1:จุดY1, จุดX2:จุดY2, … , จุดXn:จุดYn
3. โครงสร้างรูปปิด POLYGON, จุดX1:จุดY1, จุดX2:จุดY2, … , จุดXn:จุดYn

4.5 ส่วนพิกัดแผนที่

จากข้อมูลแผนที่ที่ได้จากส่วนข้อมูล GIS โดยการเปิดไฟล์และอ่านผลลัพธ์ทีละบรรทัด และนำมาเปลี่ยนแปลงข้อมูลโดยให้อยู่ในช่วงที่จุดเริ่มต้นเป็น (0,0) ซึ่งในพิกัดที่ได้จากข้อมูล GIS จะเป็นหน่วย เมตร และเริ่มพิกัดที่ (636659.88,1497316.95) ซึ่งเป็นค่าที่มากและไม่เหมาะสมต่อการนำมาคำนวณในส่วนนี้จึงได้ทำการแปลงให้อยู่ในรูปของหน่วย กิโลเมตรและทำการเริ่มพิกัดที่ (0,0) จากสูตร

$$X_{new} = \left\lfloor \frac{X - 636659.88}{1000} \right\rfloor \quad (4.4)$$

$$Y_{new} = \left\lfloor \frac{Y - 1497316.95}{1000} \right\rfloor \quad (4.5)$$

จากข้อมูลที่ทำการเปลี่ยนพิกัดจุดเริ่มต้นเป็นพิกัดใหม่แล้ว จะนำข้อมูลมาสร้างเป็นโครงสร้างแต่ละ Layer โดยในแต่ละ Layer สามารถประกอบไปด้วย Point Line และ Polygon

4.6 ส่วนติดต่อผู้ใช้

ส่วนติดต่อผู้ใช้เป็นส่วนที่ใช้ในการแสดงผลแผนที่ หาเส้นทางเดินทาง และการนำทางไปยังจุดมุ่งหมายปลายทางโดยมีรายละเอียดดังหัวข้อย่อยถัดไป

4.6.1 การแสดงผลแผนที่

การใช้งานแสดงผลภาพกราฟฟิกส์แผนที่สามารถทำได้โดยการแสดงภาพเป็นแบบ 2 มิติ และการแสดงแบบ 3 มิติ ซึ่งโดยปกติแล้วแผนที่ส่วนใหญ่มักจะนำเสนอภาพเป็นแบบ 2 มิติ ซึ่งจะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บอกได้เพียงตำแหน่งของสถานที่เท่านั้น และยังแสดงผลได้ไม่เหมือนจริงเท่าการแสดงผลแบบ 3 มิติ โดยในโครงงานนี้ใช้ไลบรารี OpenGL ในการแสดงภาพแผนที่ OpenGL นี้มีความสามารถแสดงผลได้ทั้ง 3 มิติ และ 2 มิติ การแสดงผลภาพแผนที่ในโครงงานนี้สามารถแสดงผลได้ทั้งแบบ 3 มิติ และ 2 มิติ โดยการแสดงผลในแบบ 3 มิติ ยังคงเป็นรูปแบบ เสมือน 3 มิติเท่านั้น

การแสดงผลของแผนที่จะมีแบ่งระดับของการแสดงผลอยู่ที่เป็นการช่วยประสิทธิภาพของการแสดงผลและทำให้แผนที่ดูง่ายและชัดเจนอีกด้วย โดยวิธีการนี้จะแยก Layer ของการแสดงผลออกให้แสดงในระดับของการ Zoom ที่ต่างกัน หากมีการ Zoom แผนที่ออกมาไกลๆ ภาพต่างๆที่มีขนาดเล็กควรจะตัดออกไม่แสดงผล เพราะถ้าหากยังคงแสดงผลต่อไปนอกจากจะดูไม่ชัดเจนแล้วยังเป็นการเพิ่มกระบวนการการแสดงผลให้มากขึ้นอีกด้วย

โครงงานทำการใช้ภาษา Python ในการเขียน OpenGL โดยไลบรารีที่ใช้สำหรับติดต่อกับ OpenGL API ของ Python นี้เรียกว่า pyOpenGL ซึ่งลักษณะการเขียนของ pyOpenGL จะคล้ายกับการเขียน OpenGL ใน C++ แต่ทั้งนี้การเขียนโปรแกรมภายในภาษา Python ไม่สามารถเขียนแบบอ้างอิง Pointer ได้ ดังนั้นจึงมีความแตกต่างกันเล็กน้อยสำหรับการเขียน pyOpenGL กับ OpenGL สำหรับ C++ โดยสามารถดูรายละเอียดเพิ่มเติมของ pyOpenGL ได้ผ่านเว็บไซต์ http://pyopengl.sourceforge.net/documentation/

การสร้างแผนที่โดยใช้ pyOpenGL สามารถใช้งานได้โดยการสร้าง Canvas ของ OpenGL ขึ้นมาภายใน Application นั้นๆ โดยโครงงานนี้ทำการใช้งาน OpenGL ผ่าน wxPython ซึ่ง API ที่ช่วยในการสร้าง Application ในเครื่อง PC โดย wxPython นี้มี GLCanvas ให้ใช้งานซึ่ง GLCanvas นี้สามารถช่วยในการเรียกใช้งาน OpenGL ผ่าน Canvas นี้ได้

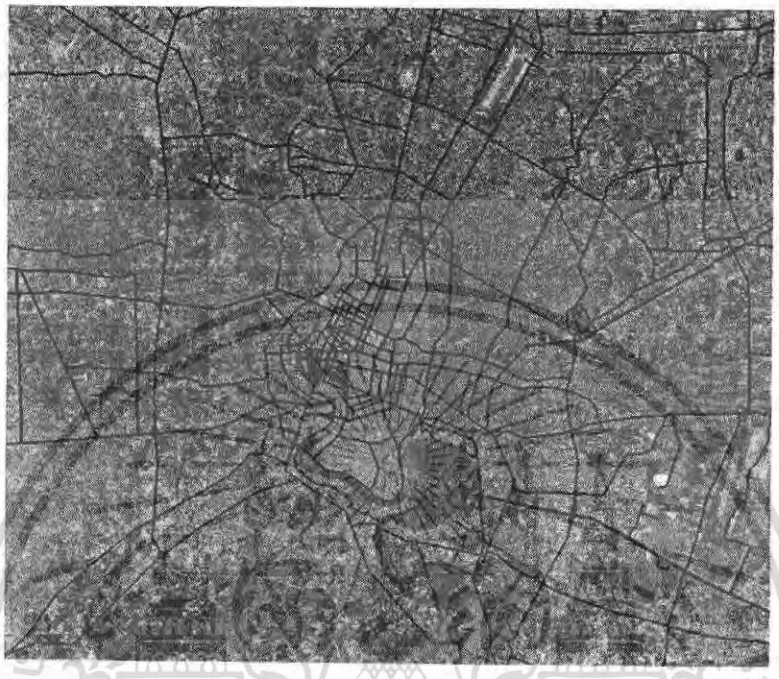
การใช้งาน GLCanvas มีขั้นตอนเบื่องต้นในการใช้งานเช่นเดียวกับการใช้งาน Canvas ทั่วไป สามารถรองรับ Event โดยผ่าน wxPython ได้ ในกรณีที่แตกต่างมีเพียงลักษณะของการวาดภาพลงใน Canvas นี้จะใช้คำสั่งของ OpenGL ในการวาดภาพ ซึ่งในการวาด Object หนึ่งๆ Object ขึ้นมาต้องทำการระบุ Vertex โดยใช้น้อยที่สุด 1 Vertex สำหรับการวาดจุด และสามารถมี Vertex เพิ่มได้มากกว่า 1 Vertex ซึ่งจะเปลี่ยนลักษณะโครงสร้างไปเป็น Line หรือ Polygon แทน ดังตัวอย่างเป็นการสร้าง Line 1 เส้นผ่านจุด Vertex 2 จุด

```
Begin(GL_LINES)          #เริ่มการวาดแบบ LINES
glVertex3f(0, 0, 0.01)   #จุด Vertex แรก
glVertex3f(1, 1, 0.01)   #จุด Vertex ที่สอง
glEnd()                  #จบการวาดแบบ LINES
```

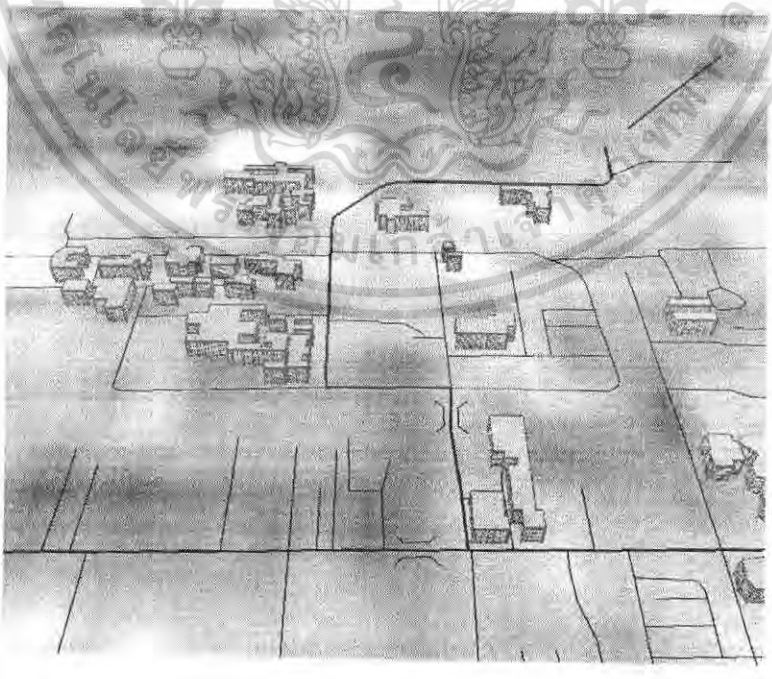
สำหรับการใช้คำสั่งการวาดของข้อมูล GIS จะใช้วิธีการวาดที่แตกต่างกันตามชนิดข้อมูลของ GIS โดยสามารถชนิดข้อมูลต่างๆ สามารถกำหนดการวาดด้วยวิธีดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลชนิด	POINT	การวาดใน GL แบบ	GL_POINTS
ข้อมูลชนิด	LINESTRING	การวาดใน GL แบบ	GL_LINE_STRIP
ข้อมูลชนิด	POLYGON	การวาดใน GL แบบ	GL_POLYGON



รูปที่ 4.8 ตัวอย่างการแสดงผลข้อมูลแผนที่โดยใช้ OpenGL แบบ 2 มิติ



รูปที่ 4.9 ตัวอย่างการแสดงผลข้อมูลแผนที่โดยใช้ OpenGL แบบ 3 มิติ

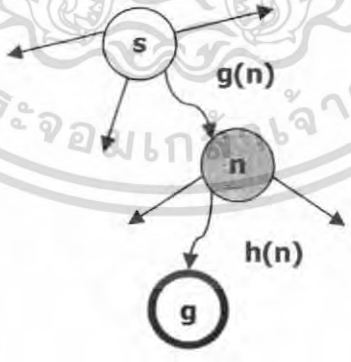
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6.2 การค้นหาเส้นทาง

การค้นหาเส้นทางบนแผนที่ เป็นการค้นหาโหนดที่เป็นจุดเชื่อมของเส้นถนน โดยที่เมื่อนำโหนดทั้งหมดกับเส้นเชื่อมโหนดก็คือ path มาต่อกันก็จะได้เส้นทางที่จะนำไปสู่จุดหมายจากที่ได้อธิบายถึงว่าการเชื่อมต่อของถนนทำให้เกิดเป็นกราฟในหัวข้อที่ผ่านมาจะเป็นผลให้ทุกๆ โหนดที่มีการเชื่อมต่อกัน สามารถเดินทางไปจากจุดหนึ่งไปยังอีกจุดหนึ่งได้โดยทั่วไปแล้วสิ่งที่เราสนใจคือการเดินทางไปในกราฟนั้น เพื่อให้ได้เส้นทางใดที่ดีที่สุด ได้แก่ มีระยะทางสั้นที่สุดใช้เวลาน้อยที่สุด เป็นต้น

การค้นหาเส้นที่ดีที่สุด มีหลักการค้นหาด้วยกันหลายวิธี ได้แก่ การค้นหาในแนวกว้าง (Breadth-First Search) ซึ่งจะให้คำตอบที่ดีที่สุดแต่ใช้เวลาในการค้นหานานมาก และใช้พื้นที่หน่วยความจำมากเกินไป ไม่เหมาะสมที่จะนำมาใช้ในทางปฏิบัติ วิธีการค้นหาที่ใช้งานได้ดีมีการนำเอา heuristic function มาช่วยตัดสินใจเลือกเส้นทางที่จะนำไปสู่จุดหมายได้โดยประหยัดทั้งเวลาและหน่วยความจำ วิธีค้นหาแบบนี้เรียกว่า การค้นหาแบบฮิวริสติก (Heuristic Search) ซึ่งมีวิธีที่เรียกว่า เอสตาร์ (A* Search) ซึ่งถือว่าเป็นวิธีที่เหมาะสมที่สุดในการทำโครงงานนี้ เพราะสามารถรับประกันว่าให้คำตอบที่ดีที่สุด (optimum solution)

ในโครงงานนี้เราได้ประยุกต์ใช้การค้นหาแบบเอสตาร์ซึ่งใช้ heuristic function f(n) = g(n) + h(n) ในการชี้นำการค้นหาไปสู่จุดหมายอย่างรวดเร็ว โดย g(n) หมายถึงการใช้ค่า (Cost) จริงที่ใช้ในการเดินทางจากโหนดเริ่มต้น s ไปจนถึงโนหดปัจจุบัน n (ซึ่งค่าใช้จ่ายในที่นี้อาจเป็นระยะทาง เวลา ฯลฯ) และค่า h(n) หมายถึงค่าใช้จ่ายประมาณการที่ใช้ในการเดินทางจากโหนดปัจจุบัน n ไปถึงโหนดจุดหมาย g ดังรูปที่ 4.10



รูปที่ 4.10 ส่วนประกอบ heuristic function f(n)

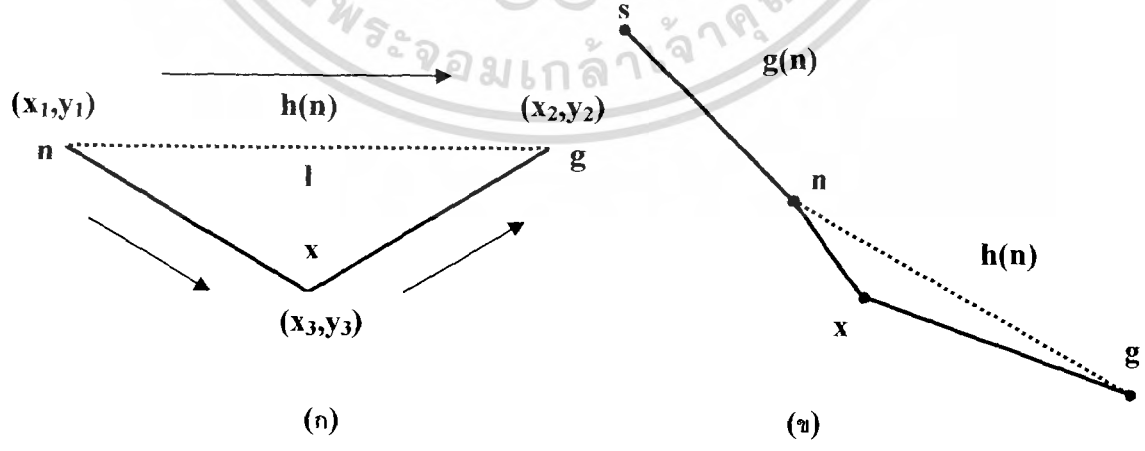
อัลกอริธึมในการค้นหาแบบเอสตาร์จะใช้ f(n) ในการตัดสินใจเลือกโหนดของกราฟเริ่มจากโหนดเริ่มต้น s เพื่อค้นหาโหนดแต่ละลูกในแต่ละขั้น ซึ่งแต่ละขั้นจะทำให้เกิดเส้นทางที่ยังค้นหาไม่จบ เส้นทางเหล่านี้มีโหนดปลาย ซึ่งแต่ละอันถือว่าเป็นโหนด n ของเส้นทางนั้นๆ ถ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เส้นทางใดในบรรดาเส้นทางเหล่านี้ให้ค่า f(n) ของโหนด n ของตนเป็นค่าน้อยที่สุดในกลุ่มโหนด n อื่นๆ อัลกอริธึมจะเลือกขยายขอบเขตการค้นหาต่อไปในเส้นทางนั้น โดยขยายการค้นหาที่โหนด n ของเส้นทางที่เลือกลงไปยังโหนดลูกของมัน ถ้ามีโหนดลูกตัวใดปรากฏในเส้นทางที่กำลังค้นหาค้างอยู่ก็จะถูกตัดทิ้ง การค้นหาโดยเอสตาร์จะทำทีละขั้นเช่นนี้ซ้ำแล้วซ้ำอีกไปเรื่อยๆ จนกระทั่งพบโหนดจุดหมาย g เส้นทางที่เกิดขึ้นเชื่อมจาก s ไป g จะถือว่าเป็นเส้นทางที่ดีที่สุดที่เลือกโดยวิธีเอสตาร์ วิธีเอสตาร์จะสามารถเลือกเส้นทางที่ดีที่สุด (optimum solution) ได้ก็ต่อเมื่อ ฟังก์ชั่นที่คิดค่าออกมาแล้วไม่ต่ำกว่าการใช้จ่ายจากโหนดเริ่มต้น s ถึงโหนดจุดหมาย g จะต้องเป็นการประมาณค่าที่ไม่เกินค่าใช้จ่ายจากโหนดปลาย n ถึงโหนดจุดหมาย g จะต้องเป็นการประมาณค่าที่ไม่เกินจากค่าใช้จ่ายยจริงจากเส้นทางที่สั้นที่สุดที่เชื่อมจาก n ถึง g สำหรับในระบบของเราจะให้ g(n) เป็นการคำนวณค่าระยะทางจริงจากจุด 2 จุดระหว่าง path ซึ่งจะเห็นว่าสอดคล้องกับเงื่อนไขข้างต้น และให้ h(n) เป็นการคำนวณระยะทางจากจุดปลาย (โหนด n) ของเส้นทางที่อยู่ในระหว่างการค้นหาให้เป็น พิกัด (x1,y1) ไปยังจุดหมาย (โหนด g) ซึ่งกำหนด พิกัด (x2,y2)

$$h(n) = \sqrt{(y_2 - y_1)^2 + (x_2 - x_1)^2} \qquad (4.6)$$

จะพบว่า h(n) เป็นการประมาณการระยะทางของเส้นทางจากโหนด n ไปยังโหนด g ซึ่งไม่มากกว่าระยะทางจริงของเส้นทางที่สั้นที่สุดที่เชื่อมจาก n ไปยัง g เสมอ เนื่องจากเส้นตรงที่เชื่อมระหว่าง n และ g ย่อมสั้นที่สุด ดูจากรูป 4.11 (ก) ให้เส้นตรงเชื่อม n และ g เป็น l) ส่วนเส้นทางในความเป็นจริงอาจมีหลายๆ path ที่เชื่อมจากโหนด n ไปยังโหนด g ซึ่ง path เหล่านี้อาจจะผ่านจุด x หรือทับเส้นตรง l ก็ได้ กรณีเส้นทางที่ทับ l ก็จะเป็นเส้นทางที่สั้นที่สุด ดังรูป 4.11 (ก)



รูปที่ 4.11 แสดงลักษณะของ function h(n) และ g(n)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6.3 การนำทางผู้ใช้งาน

การนำทางผู้ใช้งานจากส่วนวินิจฉัย ผลลัพธ์ที่ได้จะเป็นทิศทางและตำแหน่งของสัญญาณที่รับได้จาก GPS ซึ่งจะสามารถ Normalize ออกมาเป็น Vector ซึ่งบอกทิศทางของรถที่จะเคลื่อนที่ไปได้ และเมื่อนำมาผนวกกับการค้นหาเส้นทางที่หาได้จะสามารถบอกได้ว่าผู้ใช้งานควรจะเดินทางไปยังทิศทางใดโดยหาได้จากการคำนวณวิเคราะห์ Vector ออกมาเป็นองศาของทิศทางโดยคำนวณได้จากสูตรที่ 4.7

$$\theta = \tan^{-1} \frac{y_1 - y_2}{x_1 - x_2} \qquad (4.7)$$

เมื่อทำการคำนวณองศาออกมาได้ทั้งของทิศทางของรถ และทิศทางของเส้นทางต่อไปที่ได้จากการค้นหาเส้นทาง จึงนำไปหาผลต่างซึ่งผลต่างนี้เองที่จะใช้ระบุว่าการเดินทางต่อไปของผู้ใช้ควรจะเดินทางไปยังทางใดโดยโครงงานนี้ได้คำนวณผลต่างของมุมให้เป็นทางการเคลื่อนที่ดังนี้

ตารางที่ 4.2 เปรียบเทียบองศากับทิศทางการเคลื่อนที่

องศา	ทิศทางการเคลื่อนที่
33.75 ถึง -33.75	ตรงไป
33.75 ถึง 146.25	เลี้ยวซ้าย
-33.75 ถึง -146.25	เลี้ยวขวา
146.25 ถึง 213.75	กลับรถ

จากการเปรียบเทียบทิศทางของรถกับทิศทางของเส้นทางที่หาได้ จะสามารถนำใช้เป็นระบบนำเส้นทางได้ โดยการบอกเส้นทางผู้ใช้งานว่าต้องทำการเคลื่อนที่ไปยังทางใด เช่น หากอยู่ที่ทางสี่แยก และเส้นทางที่ต้องไปคือการเลี้ยวขวา ทิศทางของรถกับทิศทางของเส้นทางที่หาได้จะมีผลต่างประมาณ -45 องศา ซึ่งระบบจะตรวจได้ว่าหากได้ผลต่างเป็น -45 องศา นี้ทิศทางที่ต้องไปคือการเลียวขวา (เปรียบเทียบได้จากตารางที่ 4.2) ระบบจะทำการนำเส้นทางผู้ใช้งาน โดยทำการบอกให้ผู้ใช้งานเลี้ยวขวาเพื่อเดินทางไปยังจุดหมายปลายทางที่กำหนดไว้ตามเส้นทางที่หาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

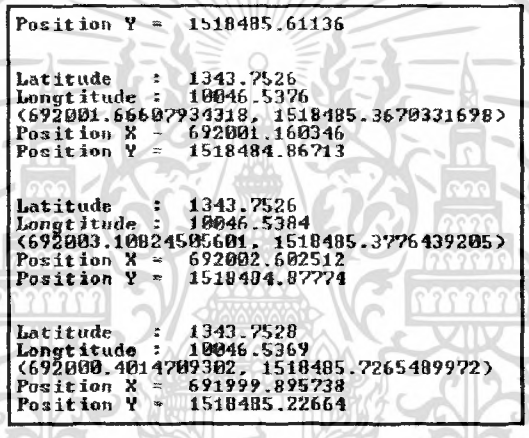
บทที่ 5

การทดลองและผลการทดลอง

ในการทดลองเราได้แบ่งส่วนการทดลองต่างตามลำดับไป

5.1 การทดลอง GPS

เป็นการแสดงการใช้งานเครื่องรับ GPS โดยเริ่มต้นทำการ เปิด port serial เพื่อติดต่อ GPS ผ่าน Bluetooth สามารถอ่านข้อมูลและกรองเอาเฉพาะข้อมูลที่สนใจและทำการแปลงค่าจากเส้นรุ้ง เส้นแวงให้เป็นค่าพิกัด (X, Y) ได้ และจากการทดสอบความสามารถในการรับสัญญาณของ GPS มีข้อจำกัดในบริเวณภายในอาคารไม่สามารถรับสัญญาณได้

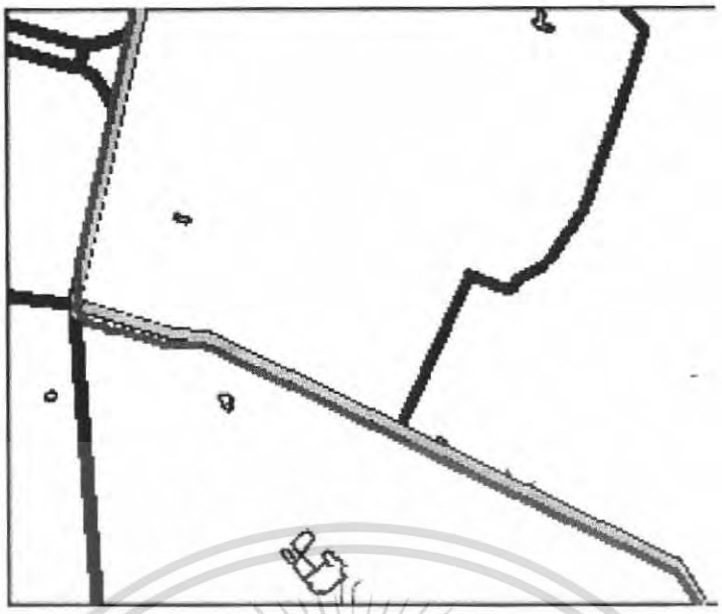


รูปที่ 5.1 การทดสอบการรับค่าของ GPS และ แปลงเป็นพิกัด (X, Y)

5.2 การทดลอง Snap เส้นทาง

การทดลอง Snap เส้นทางโดยการใช้ Chain Code ผลที่ได้สามารถวิเคราะห์ตำแหน่งในแผนที่ได้ทำให้รูปแบบการแสดงผล ของตำแหน่งมีความผิดพลาดน้อยกว่าการใช้ข้อมูลดิบที่อ่านได้ แต่หากไม่มีข้อมูลแผนที่ในพื้นที่นั้น ผลของการ Snap เข้ากับถนนจะทำให้ได้ผลลัพธ์ที่ผิดพลาด นอกจากนี้ หากพื้นที่ถนนเป็นเส้นขนานกันและมีระยะทางที่ใกล้กันมาก การใช้ Snap เส้นทางจะไม่สามารถบอกได้เลยว่าอยู่บนถนนสายไหน นี่ถือเป็นข้อจำกัดของข้อมูลที่ได้รับมา

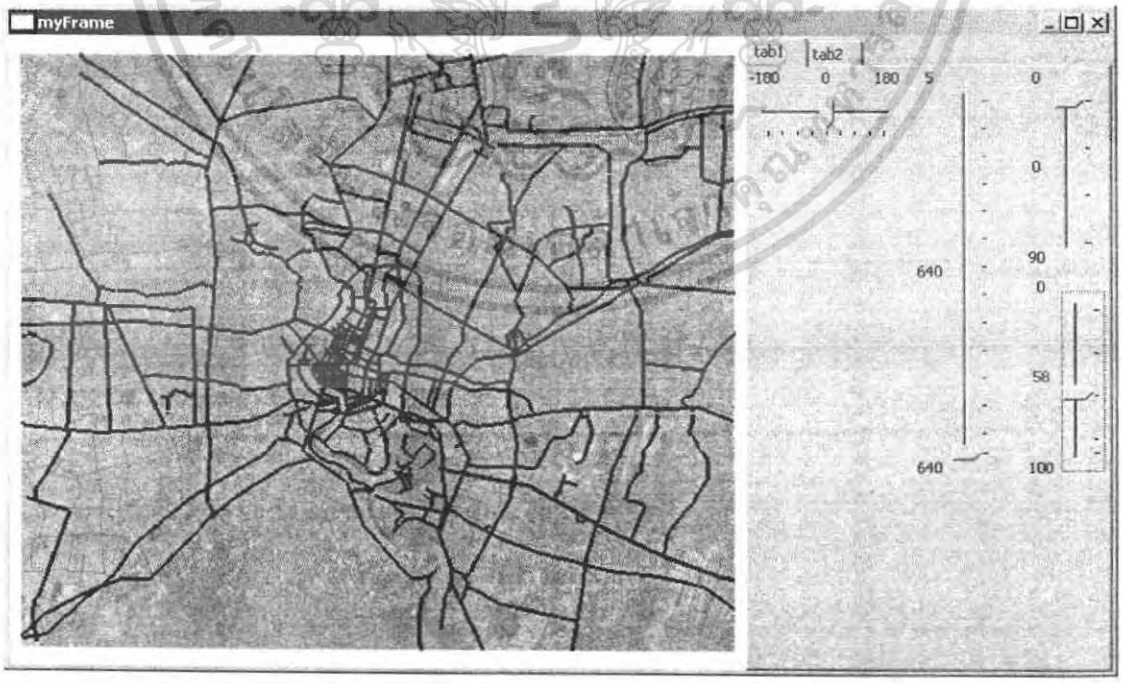
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.2 ผลเปรียบเทียบของการลดความผิดพลาดการรับค่าจาก GPS

5.3 การทดลองแสดงภาพแผนที่โดยใช้ข้อมูลแผนที่

การแสดงผลสามารถแสดงผลได้อย่างดี เนื่องจากข้อมูลเป็นชนิดเวกเตอร์ทำให้การแสดงผลเมื่อ Zoom เข้าไปในระยะใกล้จะไม่มีการเบลอของภาพ ซึ่งพบเห็นได้ทั่วไปในภาพแบบราสเตอร์

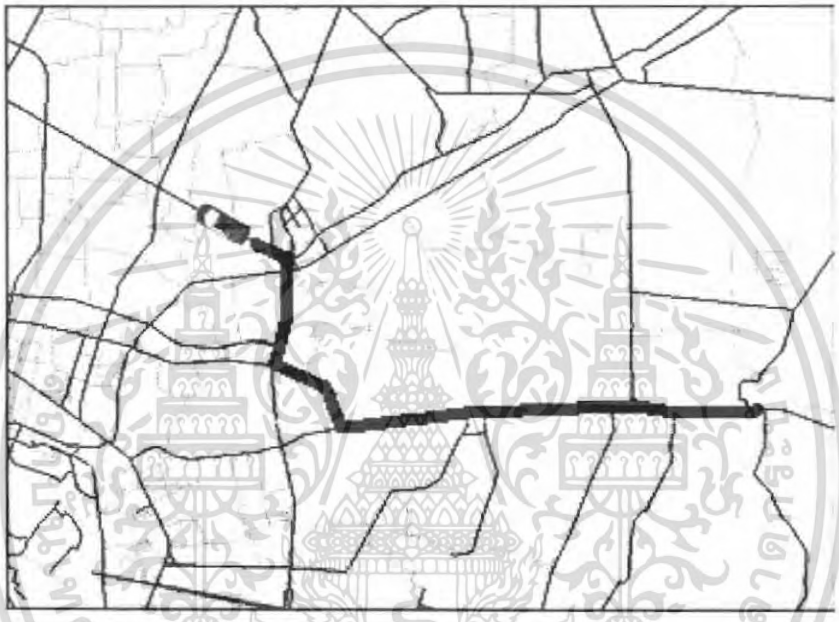


รูปที่ 5.3 แอปพลิเคชันที่ใช้แสดงแผนที่จากแบบเวกเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.4 การทดลองใช้อัลกอริทึม A* ในการหาเส้นทางที่สั้นที่สุด

ในโครงการได้สร้างอัลกอริทึม A* เพื่อใช้หาเส้นทางที่สั้นที่สุด โดยการทดสอบทำการเลือก โหนดเริ่มต้นและโหนดสิ้นสุด การทำงานของ A* จะทำการเลือกเอาเส้นทางที่ดีที่สุด โดยตัดสินใจจากผลรวมของฟังก์ชัน g() และ h() ซึ่งฟังก์ชัน g() เป็นผลรวมของค่า Cost ที่ผ่านมา และฟังก์ชัน h() เป็นฟังก์ชันที่ให้ผลลัพธ์เป็นค่าที่เข้าใกล้ศูนย์เมื่อตำแหน่งปัจจุบันเข้าใกล้ Goal จากวิธีการของ A* นี้สามารถค้นหาเส้นทางสั้นที่สุดไปยังจุดหมายปลายทางได้อย่างมีประสิทธิภาพ ดังรูปที่ 5.4



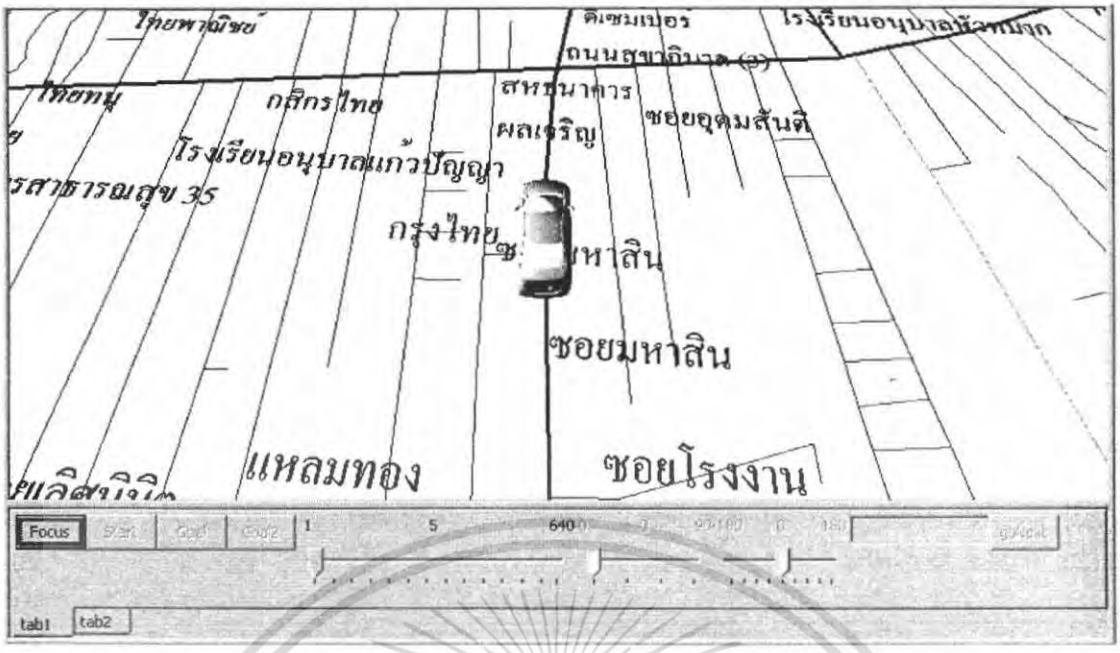
รูปที่ 5.4 ตัวอย่างการค้นหาเส้นทางสั้นที่สุดจาก พระจอมเกล้าลาดกระบัง ไป เดอะมอลล์บางกะปิ

5.5 การทดลองการนำเส้นทาง

ขั้นตอนของการใช้ระบบนำเส้นทางมี 3 ขั้นตอนคือ

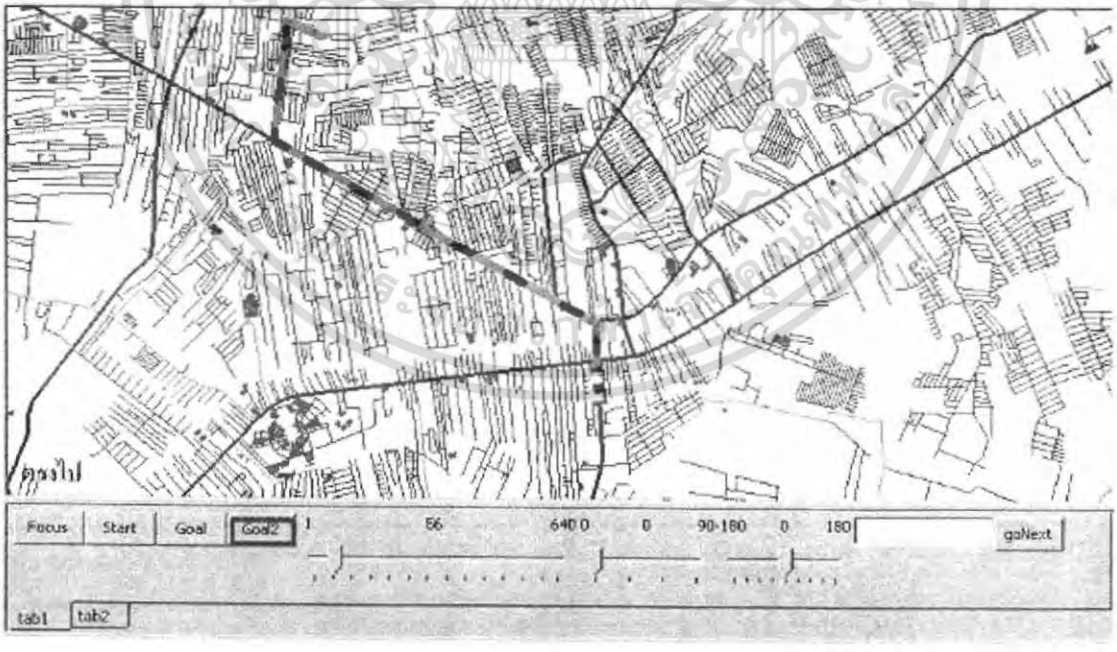
1) ทำการเปิดระบบ GPS เพื่อใช้ระบุตำแหน่งปัจจุบันของผู้ใช้งาน โดยจากตัวอย่างโปรแกรมจะมีการจำลองข้อมูล GPS อยู่ซึ่งจะทำการแสดงตำแหน่งเป็นสัญลักษณ์รถยนต์ สามารถทำการ Focus ที่จุดสัญญาณ GPS ได้โดยการกดที่ปุ่ม Focus

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.5 การแสดงผลเมื่อทำการ Focus ไปยังตำแหน่งสัญญาณ GPS

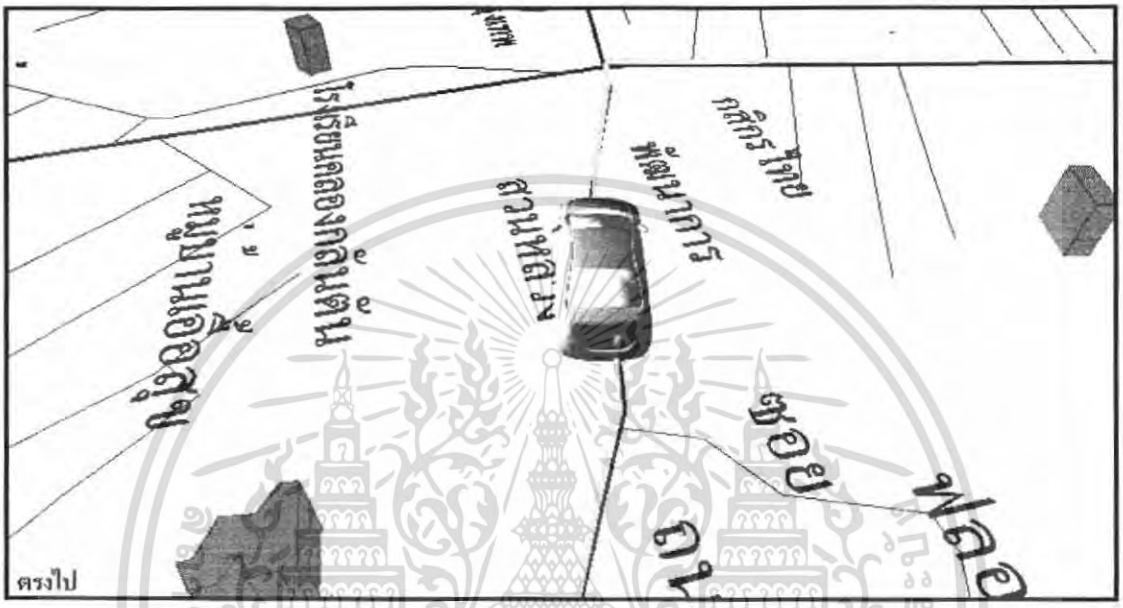
2) ทำการเลือกเป้าหมายปลายทางที่ต้องการให้นำเส้นทางไปโดยการกดปุ่ม Goal2 เพื่อเลือกเส้นทางจากตำแหน่งสัญญาณ GPS ไปยังจุดที่เลือกในแผนที่



รูปที่ 5.6 การเลือกเส้นทางจากตำแหน่ง GPS ไปยังจุดบนแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

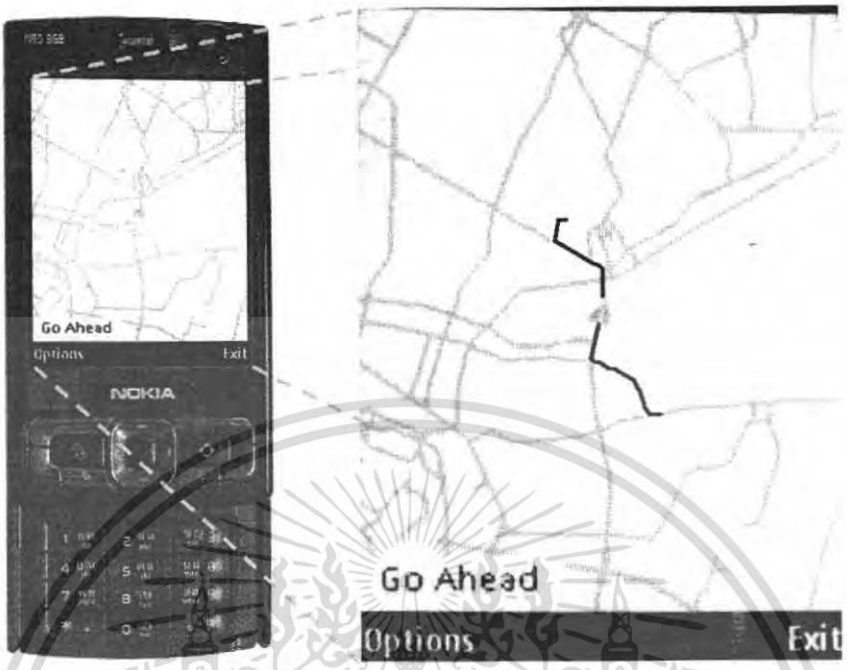
3) ใช้งานระบบนำเส้นทาง โดยระบบจะเริ่มอัตโนมัติหลังจากมีการเลือกตำแหน่งเส้นทางที่ต้องการไปในข้อที่ 2 โดยการใช้งานระบบนำเส้นทาง จะมีการแสดงเส้นทางจากจุดเริ่มต้นไปยังจุดหมายที่ต้องการไป โดยขณะที่ทำการเดินทางไปยังจุดหมายปลายทาง จะมีการแนะนำทางให้เคลื่อนที่ไปยังทิศทางใดโดยใช้เสียง และอาศัยการบอกทางนี้ทำให้สามารถทำการเดินทางไปยังจุดหมายได้อย่างสะดวกสบายมากขึ้น



รูปที่ 5.7 ภาพตัวอย่างแสดงการใช้งานระบบนำทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.6 การทดลองระบบในโทรศัพท์มือถือ N95



รูปที่ 5.8 ภาพตัวอย่างการใช้งานบนโทรศัพท์มือถือ N95

ในส่วนของระบบโทรศัพท์มือถือจะใช้วิธีการเดียวกันกับของระบบที่ใช้ใน PC แบบพกพา โดยการใช้งานประกอบไปด้วย

1. ส่วนข้อมูลจาก GPS
2. ส่วนแสดงแผนที่
3. ส่วนการค้นหาเส้นทาง
4. ส่วนการนำเส้นทาง

ซึ่งส่วนที่แตกต่างจากของระบบ PC แบบพกพาคือส่วนแสดงแผนที่ของโทรศัพท์มือถือจะใช้วิธีการแสดงเป็นภาพแบบ 2 มิติ ซึ่งสามารถลด process ในการทำงานของโปรแกรมลงได้ โดยที่ยังคงผลการทำงานได้เช่นเดียวกันกับระบบใน PC แบบพกพา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
บทวิจารณ์และสรุป

6.1 บทสรุป

ในโครงงานนี้ได้ทำการสร้างต้นแบบระบบนำทาง ประกอบด้วยส่วนแผนที่คอมพิวเตอร์ ส่วนการค้นหาเส้นทางที่ใกล้ที่สุด ส่วนทำหน้าที่แสดงตำแหน่งโดยมี GPS ซึ่งทำหน้าที่ระบุจุดพิกัด ณ. ตำแหน่งปัจจุบันของรถแล้วส่งไปยังส่วนวินิจฉัยเพื่อระบุตำแหน่งบนแผนที่ และส่วนการนำเส้นทางให้แก่รถบนแผนที่ในขณะรถวิ่งโดยใช้เสียงบอกนำเส้นทาง

6.2 วิจารณ์สิ่งที่ได้จากโครงงาน

จากการทำโครงงานทำให้เข้าใจหลักการทำงานของอุปกรณ์นำทาง ระบบพิกัดแผนที่และการระบุตำแหน่งจากดาวเทียม หลักการทำงานของ Navigator และหลักการของ AI ในการหาเส้นทางการเดินทางในระบบนำทาง

6.3 ปัญหาอุปสรรคและแนวทางแก้ไข

ปัญหาในเรื่องของการทำข้อมูล path จาก แผนที่จริงซึ่งมีจำนวนมากเป็น ทำให้ต้องใช้เวลาในการทำ ข้อมูลกราฟของแผนที่ และการพัฒนาโปรแกรมในคอมพิวเตอร์พกพามีความสามารถที่จำกัด ต่างกับการพัฒนาใน PC ทั่วไปทำให้โปรแกรมที่ทำการพัฒนาก่อนแล้วบน PC ต้องปรับเปลี่ยนให้สามารถใช้งานบน คอมพิวเตอร์พกพาได้

6.4 แนวทางการพัฒนาต่อ

พัฒนาให้ระบบมีสามารถใช้งานกับเครื่องคอมพิวเตอร์ชนิดเคลื่อนที่ได้อย่างสมบูรณ์

พัฒนาต่อให้ระบบมีความชาญฉลาดยิ่งขึ้น เช่น มีการแนะนำเส้นทางโดยวิเคราะห์จากความหนาแน่นของการจราจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

นท. ไพศาล โมลิสกลุมงคล. 2549. “คอมพิวเตอร์กราฟฟิกส์ใช้OpenGL.” กรุงเทพฯ : ดวงกมลสมัย

ประภากร ลาภประสพ. 2544. “ระบบนำร่องรถยนต์ชาญฉลาด.” วิทยานิพนธ์วิศวกรรมศาสตร์มหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

วิศิษฏ์ หิรัญกิตติ และ สราญพงศ์ หนูยิ้มซ้าย. 2550. “ระบบนำร่องรถยนต์ที่คุมเข้มพฤติกรรมการขับขี่รวมทั้งลงโทษให้เป็นไปตามกฎจราจร.” การประชุมวิชาการทางด้านวิทยาการคอมพิวเตอร์และวิศวกรรมซอฟต์แวร์.

วิศิษฏ์ หิรัญกิตติ และ สราญพงศ์ หนูยิ้มซ้าย. 2549. “ระบบนำร่องรถยนต์ที่สามารถตรวจสอบพฤติกรรมการขับขี่ที่ฝ่าฝืนกฎจราจร.” การประชุมวิชาการทางด้านเทคโนโลยีสารสนเทศ.

ประภากร ลาภประสพ วิศิษฏ์ หิรัญกิตติ และ ชม กิ้มปาน. 2543. “ระบบแผนที่ชาญฉลาด.” การประชุมเสนอผลงานระดับบัณฑิตศึกษา ครั้งที่ 1 มหาวิทยาลัยเชียงใหม่.

เกียรติศักดิ์ คิวขุนทด และ นรินทร์ เรืองศรี. 2545. “ระบบให้บริการแผนที่บนเว็บ.” ปริญญานิพนธ์ วิศวกรรมศาสตร์บัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

Barnes Scottie. 2000. “Global Positioning Systems.” USA : The Globe Pequot Press.

Jan Van Sickle. 2001. “GPS for Land Surveyors 2nd ed.” USA : Ann Arbor Press.

Rabbany Ahmed. 2002. “Introduction to GPS : the Global Positioning Systems.” USA : Artech House Inc.

Jurgen Scheible and Ville Tuulos. 2007. “Mobile Python.” San Francisco : John Wiley and Sons Inc.

Jonathan Liperi. 2001. “Python HOW TO PROGRAM Introducing XML.” USA : Prentice-Hall Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

ข้อมูลอ้างอิงการโปรแกรมในโทรศัพท์

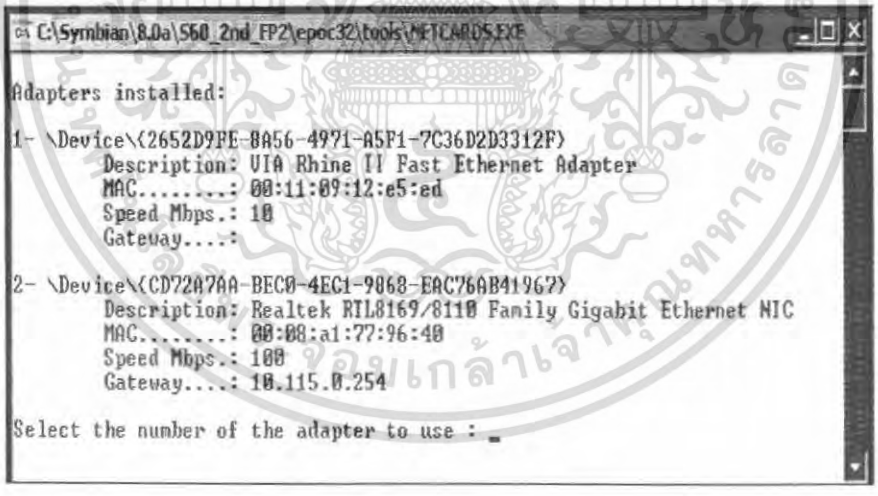
ก.1 วิธีการลง Python SDK ใน S60

ระบบปฏิบัติการของ S60 ได้มีการรองรับการเขียนโปรแกรมโดยใช้ภาษา Python ซึ่งจะต้องทำการลง pyS60 SDK ลงในโทรศัพท์มือถือก่อนจึงจะสามารถทำการโปรแกรมภาษา Python ได้

ก.1.1 ติดตั้งโปรแกรมภาษา Python บนคอมพิวเตอร์ผ่านตัว Emulator

1. ทำการ Download S60 SDK (Standard Developer Kid) จาก www.nokia.com โดยทำการสมัครสมาชิกก่อนและทำการติดตั้ง SDK ลงเครื่อง

2. ในระหว่างทำการ Install หากในเครื่องคอมพิวเตอร์มีการ์ด Network มากกว่า 1 ตัวจะมีหน้าต่าง Popup ขึ้นมาถามว่าต้องการใช้การ์ด Network ตัวไหน ให้ระบุตัวการ์ดที่ใช้งานติดต่อกับ Network เพื่อให้ Emulator สามารถติดต่อกับ Network ได้



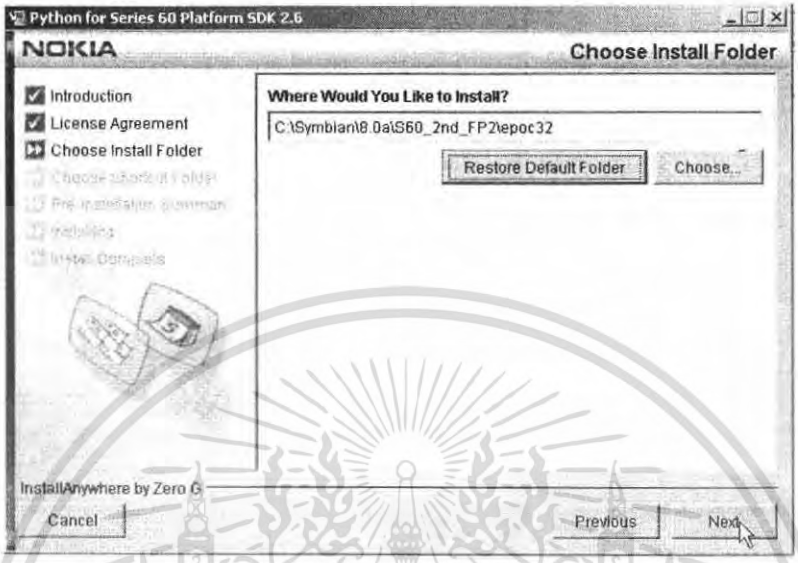
รูปที่ ก.1 ภาพแสดงตัวเลือก Network card ที่ต้องการใช้

3. จากนั้นจะมีหน้าต่างให้เลือก Device ที่จะใช้ ให้เราเลือก S60_2nd_FP2

4. โดยปกติแล้ว SDK ที่ทำการติดตั้งนี้จะรองรับการเขียนโปรแกรม ภาษา C++ หากต้องการที่จะใช้งานภาษา Python จะต้องทำการ Install SDK for Python

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. การ Install SDK for Python จะต้องทำการ Install ให้ถูก directory มิฉะนั้นจะไม่สามารถ Run ได้ซึ่ง Directory ที่จะติดตั้งต้องเป็น Directory ของ S60 SDK ที่ติดตั้งไปดังภาพและทำการดำเนินการติดตั้งต่อไปจนเสร็จการติดตั้ง



รูปที่ ก.2 ภาพแสดงการเลือก Path ที่ต้องการติดตั้ง

6. จากข้อที่ 5 เป็นการติดตั้งตัว Emulator ซึ่งจะไม่มี pyS60 มาด้วยเพราะฉะนั้นต้องทำการติดตั้ง Program ภาษา Python ลงเพิ่มเติมลงไปเพื่อให้สามารถใช้ Python ได้

ก.1.2 ติดตั้งโปรแกรมภาษา Python ลงบนโทรศัพท์มือถือ

1. ทำการติดตั้ง PC Suit software ลงบนเครื่องคอมพิวเตอร์เพื่อให้สามารถติดตั้งโปรแกรมลงบนโทรศัพท์ผ่านทางคอมพิวเตอร์ได้ และทำการต่อสาย USB ระหว่างเครื่องกับตัวโทรศัพท์

2. ทำการ Download และติดตั้ง .SIS file ไปยังโทรศัพท์โดยให้ Download pyS60 SDK และ pyS60 Shell มาและทำการติดตั้งเพื่อให้โทรศัพท์สามารถทำงานภาษา Python ได้และมี Shell ไว้ในการใช้โปรแกรม

3. การเขียน Upload ไฟล์ไปไว้ในโทรศัพท์ ทำการเชื่อมต่อกับโทรศัพท์ผ่าน File Manager ของ PC Suite และทำการ Upload ไฟล์ Scrip Python ที่ต้องการไปไว้ใน Directory ที่ทำการติดตั้ง Python ไว้ เช่น E:\Python

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก.2 OpenGL ES API

เนื่องจาก pyS60 สำหรับโทรศัพท์มือถือได้มี OpenGL ES ซึ่งมีข้อแตกต่างกับ OpenGL ของ PC ดังนั้นการใช้งาน OpenGL จะมีข้อแตกต่างกันซึ่ง OpenGL ES เป็นมาตรฐานที่ถูกกำหนดโดย Khronos Group (www.khronos.org) เป็น API ที่ช่วยในการวาดภาพกราฟฟิกส์ 2D/3D ซึ่งระบบปฏิบัติการ Series 60 นั้นรองรับ OpenGL ES version 1.0 เป็นพื้นฐาน แต่หากเป็น Series 60 version 2.6 จะรองรับกับ OpenGL ES 1.1 สำหรับข้อแตกต่างและคำอธิบายต่างๆ เกี่ยวกับ OpenGL ES นี้สามารถดูเพิ่มเติมได้ใน เว็บไซต์ดังกล่าว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้