สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เกมส์จำลองการบินของเครื่องบินสำหรับเครื่อง พ็อกเก็ตพีซี

Flight Simulator Game for PocketPC

T055084

โดย

นายปริญญา เรืองจิตรานนท์

นายรัตน เครือชาลี

อาจารย์ที่ปรึกษา

ดร. วิศิษฏ์ หิรัญกิตติ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เลขหมู่
เลขทะเบียน 55084
วัน,เดือน,ปี - 8 มี.ค. 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

b
i

ปริญญานิพนธ์ ปีการศึกษา 2546

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เกมส์จำลองการบินของเครื่องบินสำหรับเครื่องพ็อกเก็ตพีซี

Flight Simulator Game for PocketPC

คณะผู้จัดทำ นายปริญญา เรืองจิตรานนท์ รหัส 44015335

นายรัตน เครือชาลี รหัส 44015344

..... อาจารย์ที่ปรึกษา

(ดร. วิศิษฏ์ หิรัญกิตติ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เกมส์จำลองการบินของเครื่องบินสำหรับเครื่อง พ็อกเก็ตพีซี

นายปริญญา เรืองจิตรานนท์ 44015335
นายรัตน เครือชาลี 44015344
ดร.วิศิษฏ์ หิรัญกิตติ อาจารย์ที่ปรึกษา
ปีการศึกษา 2546

บทคัดย่อ

ปริญญานิพนธ์นี้ นำเสนอการพัฒนาเกมส์จำลองการบินของเครื่องบินที่เป็นภาพเคลื่อนไหว 3 มิติ โดยพยายามจำลองลักษณะทางกายภาพของเครื่องบิน และสิ่งแวดล้อม โดยคำนึงถึงคุณสมบัติบางส่วนทางพลศาสตร์ ของวัตถุในเกมส์ เช่น น้ำหนักของเครื่องบิน แรงโน้มถ่วง ความเร็วของขีปนาวุธ แรงพยุงลม ฯลฯ ซึ่งทำให้การจำลองมีความสมจริง

การแสดงผล 3 มิติของเกมส์อาศัย Diesel Engine ซึ่งเป็น API สำหรับพัฒนาภาพ 3 มิติบนเครื่องพ็อกเก็ตพีซี ส่วนการจำลองการบิน นั้นได้มาจากการดัดแปลงโปรแกรม JSBsim ซึ่งเป็นตัวสร้างสภาวะแวดล้อมทางคณิตศาสตร์ ที่ใช้แพร่หลายในโปรแกรมจำลองการบิน เช่น Flight Gear สำหรับการติดต่อผ่านทางเครือข่ายของเกมส์ อาศัยการเขียนโปรแกรมสื่อสารบน Winsock API

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flight Simulator Game for PocketPC

Parinya Ruangjittranon

Ruttana Khurchalee

Dr.Visit Hirankitti Advisor

Academic Year 2003

ABSTRACT

In this thesis we propose a development of a 3D flight simulation game. The physical reality of airplanes, environments, and objects in the game is simulated by taking some of their dynamic properties into account such as the airplanes' weights, earth gravity, missile's velocity, etc. This makes the game look more realistic.

To develop the game, we employ Diesel Engine API for the 3D rendering. This API is developed for the PocketPC platform only. The JSBsim is simplified to perform dynamic simulation of the game objects on the PocketPC, this is because the PocketPC's processor is not very powerful. The JSBsim program is very well-known and employed in many flight simulator programs such as FlightGear. Finally, to deliver an online game, the communication between game consoles is developed using the Winsock API.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ทุกอย่างที่ดูเหมือนกับโชคชะตาได้ลิขิตมา ให้พาพัดความยุ่งยากหลายประการมาสู่คณะผู้จัดทำ อุปมาเปรียบได้ดังเรือลำน้อยกลางพายุทะเลคลั่ง ซึ่งดูเหมือนจะทำให้ปริญญานิพนธ์ฉบับนี้ ต้องมีอันพิกลพิการ หรือมีอันเป็นไป เว้นหากแต่ยังมีบุคคลอีกหลายท่าน หลายฝ่าย ที่ได้คอยเกื้อหนุนค้ำจุน ให้ปริญญานิพนธ์ฉบับนี้ สำเร็จลุล่วงไปด้วยดีทั้งทางตรงและทางอ้อม โดยบุคคลหลายท่านต่อไปนี้มีส่วนช่วยอย่างมาก ในการต่อเติมเรือลำน้อยให้แข็งแกร่งพอที่จะฝ่ากระแสพายุทะเลคลั่งไปได้ โดยบุคคลแรกที่จะกล่าวถึง เป็นบุคคลที่มีส่วนสำคัญที่ทำให้ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี นั่นก็คือ ดร.วิศิษฎ์ หิรัญกิตติ อาจารย์ที่ปรึกษาปริญญานิพนธ์ ที่ให้ความเอาใจใส่ แนะนำ และคอยช่วยเหลือในทุกๆด้าน ต้องขอขอบพระคุณเป็นอย่างสูง

นอกจากนี้ต้องขอขอบคุณ เงินอุดหนุนโครงการที่ได้รับจาก สำนักงานกองทุนสนับสนุนงานวิจัย โครงการโครงงานอุตสาหกรรมสำหรับปริญญาตรี (IRPUS) ที่ให้การสนับสนุนทางด้านทุนทรัพย์ และค่าใช้จ่ายในส่วนต่างๆอีกมากมาย ตลอดจนขอขอบคุณห้องวิจัย Intelligent Communication and Transportation LAB หรือห้องวิจัย ICT ที่ให้บรรยากาศความสงบ ที่เอื้อต่อการทำวิจัย-การทำปริญญานิพนธ์และอีกหลายๆงาน ในระยะหนึ่งปีที่ผ่านมา

และสุดท้ายนี้ต้องขอบพระคุณบุคคลที่มีความสำคัญที่สุดที่ทำให้มีวันนี้ได้ ก็คือ บิดา และ มารดา อันเป็นที่เคารพและรักยิ่งซึ่งได้อบรมเลี้ยงดูมาเป็นอย่างดี พร้อมทั้งได้ให้โอกาสในการศึกษาที่ดีที่สุด และให้กำลังใจในการทำงานเสมอมา คณะผู้จัดทำขอระลึกในพระคุณ และขอกราบพระคุณเป็นอย่างสูง มา ณ ที่นี้

ปริญญา เรืองจิตรานนท์
รัตน เครือชาลี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้าที่

บทคัดย่อภาษาไทย I
บทคัดย่อภาษาอังกฤษ II
กิตติกรรมประกาศ III
สารบัญ IV
สารบัญตาราง VIII
สารบัญรูปภาพ IX
บทที่ 1 บทนำ 1
1.1 ที่มาโครงการ 1
1.2 วัตถุประสงค์ 1
1.3 ขอบเขตของโครงงาน 2
1.4 วิธีดำเนินงาน 2
บทที่ 2 พื้นฐานการแสดงภาพกราฟิก 3 มิติ 3
2.1 สภาวะแวดล้อม (Environment) ของภาพ 3 มิติ 3
2.1.1 ความรู้พื้นฐานเกี่ยวกับสี 3
2.1.2 พิกเซล (Pixel) 5
2.1.3 ชนิดของเส้น (Line type) 5
2.1.4 พื้นผิว (Texture) 6
2.1.5 การใส่สี (Fill Color) 6
2.1.6 แสง (Light) และ แหล่งกำเนิดแสง (Light Source) 7
2.1.6.1 แหล่งกำเนิดแสงชนิดจุด 7
2.1.6.2 แหล่งกำเนิดแสงขนานระยะไกล 8
2.1.6.3 แหล่งกำเนิดแสงแบบสปอร์ตไลท์ 8
2.1.6.4 แสงแบบแอมเบียนท์ 8
2.1.7 คุณสมบัติทางแสงของวัตถุ (Material) 9
2.1.7.1 คุณสมบัติของการกระจายแสง 9
2.1.7.2 คุณสมบัติของการรับแสงแอมเบียนท์ 9
2.1.7.3 คุณสมบัติของการสะท้อนแสง 9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่

2.1.7.4 คุณสมบัติการเปล่งแสง 9

2.2 วัตถุ 2 มิติ (2D Geometry Component) 9

2.2.1 จุด (Point) 9

2.2.2 เส้น (Line) 10

2.2.3 เส้นต่อเนื่อง (Pline) 11

2.2.4 วงกลม และ สมการ วงกลม 11

2.2.5 วงรี 11

2.2.6 บิตแมพ (Bitmap) 12

2.3 วัตถุ 3 มิติ (3D Geometry Component) 13

2.3.1 สี และ ความสว่างของวัตถุ 13

2.4 การมอง 13

2.4.1 การปรับภาพ และ การกวาดภาพ (Zoom and Pan) 13

2.4.2 วิว (View) 14

2.4.2.1 ขอบเขตการมองเห็น 14

2.4.2.2 การใช้เลนส์ ชนิดต่างๆ ในการมองภาพ 14

2.4.3 ภาพ เพอร์สเปกทีพ (Perspective) 15

บทที่ 3 คณิตศาสตร์พื้นฐานของระบบกราฟิก 3 มิติ 17

3.1 ระบบแกน 3 มิติ (3D Coordinate Systems) 17

3.2 จุดหรือเวกเตอร์ 17

3.3 โพลีกอน (Polygon) 19

3.4 ระนาบ (Plane) 21

3.5 เมตริกซ์ (Matrix) 21

บทที่ 4 เทคโนโลยีที่สนับสนุนการแสดงภาพ 3 มิติบนพ็อกเก็ตพีซี 25

4.1 เทคโนโลยีด้านอุปกรณ์ที่ใช้ 25

4.2 ไลบรารี กราฟิกสามมิติบนอุปกรณ์ขนาดเล็ก 26

บทที่ 5 โครงสร้างตัวเกมส์ 28

5.1 โครงสร้างของระบบ 28

5.2 รายละเอียดของการออกแบบในส่วนต่างๆ 29

5.2.1 โมเดลพลศาสตร์การบิน 29

5.2.2 วัตถุสามมิติ 29

5.2.3 โมดูลติดต่อสื่อสาร 29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่

5.2.4 ไลบรารี กราฟิกสามมิติ 29
5.2.5 โปรแกรมหลัก 30
5.3 ความสัมพันธ์ของส่วนต่างๆ 30
บทที่ 6 คณิตศาสตร์ของการจำลองการบินของเกมส์ 32
6.1 JSB Sim 32
6.2 ความเข้าใจในพิกัดของเครื่องบิน เมื่อเทียบกับพิกัดสามมิติ 33
6.3 หลักการในการสร้างเรดาห์ตรวจจับศัตรู 34
6.4 หลักการตรวจสอบมุมก้มเงย (pitch) ของเครื่องบิน 36
6.5 หลักการในการสร้างตัวตรวจจับระดับพื้นดิน (Attitude Director Indicator , ADI) 36
6.6 ผลกระทบจากสภาวะแวดล้อมภายนอก ต่อเครื่องบิน 37
บทที่ 7 ดีเซล เอนจิ้น (Diesel Engine) 39
7.1 โครงสร้างพื้นฐานของตัวดีเซล ไลบรารี 39
7.2 รายละเอียดของคลาส และ เม็ดธอด ต่างๆที่จำเป็นในการเขียนโปรแกรม 40
บทที่ 8 การเขียนโปรแกรมเน็ตเวิร์ค (TCP/IP) โดยใช้ วินซ็อค (Winsock) 44
8.1 ซ็อคเก็ต (Socket) 44
8.2 ชนิดของซ็อคเก็ต 44
8.2.1 การเชื่อมต่อแบบ ต่อเนื่อง (Connection oriented Socket) 44
8.2.2 การเชื่อมต่อแบบ ไม่ต่อเนื่อง (Connectionless Socket) 45
8.3 ทีซีพี/ไอพี 45
8.4 วินซ็อค (Winsock) 48
8.4.1 วินซ็อคและโอเอสไอโมเดล (OSI Model) 48
8.4.2 การเขียนโปรแกรมการเชื่อมต่อผ่านเครือข่าย ด้วยวินซ็อค บนวินโดว์ซีอี (Window CE) 49
8.4.2.1 เซิร์ฟเวอร์และไคลเอ็นท์ (Server and Client) 49
8.4.2.2 ฟังก์ชัน วินซ็อคโพร (Winsock Function) 50
บทที่ 9 แนวคิดของเกมส์ 51
9.1 วัตถุและผลกระทบ ต่างๆ ในเกมส์ 51
9.2 รูปแบบการเล่น 52
9.3 การทำงาน ของโปรแกรม 52
บทที่ 10 ผลการทดลอง 58
บทที่ 11 บทสรุปและวิจารณ์ 63
11.1 วิจารณ์งานวิจัย 63
11.2 สรุปงานวิจัย 63

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่

11.3 ปัญหาและการแก้ไข 64

11.4 แนวทางในการพัฒนาต่อไป 64

ภาคผนวก การติดตั้ง Embedded visual C++ และ PocketPc Emulator 66

บรรณานุกรม 67

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้าที่

ตาราง 4-1 การเปรียบเทียบ พีดีเอ ชนิดต่าง ๆ 25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

หน้าที่

รูปที่ 2-1 แม่สีทั้ง 3 สีแดง สีเขียว และ สีน้ำเงิน 3
รูปที่ 2-2 แบบจำลองแม่สี อาร์จีบี 3
รูปที่ 2-3 แบบจำลองแม่สี ซีเอ็มวาย 3
รูปที่ 2-4 สเปกตัม 4
รูปที่ 2-5 ตัวการการวางพิกเซลบนตารางกริด 5
รูปที่ 2-6 ตัวอย่างของ เส้น (Stipple line) 5
รูปที่ 2-7 ตัวอย่างของ เส้นสทริพ (Line stips) 6
รูปที่ 2-8 ตัวอย่างของ เส้นวนรอบ (Line loop) 6
รูปที่ 2-9 ระนาบฉากแสดงผิวหน้าแบน 6
รูปที่ 2-10 ระนาบฉากแสดงพื้นผิวแบบ บั๊มแมพ (Bumb map) 6
รูปที่ 2-11 แหล่งกำเนิดแสงชนิดจุด 7
รูปที่ 2-12 แหล่งกำเนิดแสงขนานระยะไกล 8
รูปที่ 2-13 แหล่งกำเนิดแสงแบบสปอร์ตไลท์ 8
รูปที่ 2-14 ระบบแกนของ วินโดว์ (Window Coordinate System) 10
รูปที่ 2-15 ระบบคาร์ทีเชียล (cartesian coordinate system) 10
รูปที่ 2-16 แสดงตัวอย่างการสร้างรูป 3 เหลี่ยม 10
รูปที่ 2-17 แสดงตัวอย่างการแปลงจากคาร์ทีเชียล มาเป็นระบบ วินโดว์ 10
รูปที่ 2-18 แสดงการสร้างเส้น 11
รูปที่ 2-19 แสดงสมการ ของวงกลมที่มีจุดศูนย์กลางอยู่จุดกำเนิด และไม่ได้อยู่จุดกำเนิด 11
รูปที่ 2-20 วงรีบนแกน x,y 12
รูปที่ 2-21 ภาพแสดงการเรียงตัวของพิกเซลในหนึ่งระนาบบิต 12
รูปที่ 2-22 ความเข้มและอัตราส่วนที่คมชัดกว่า 12
รูปที่ 2-23 ระบบ 3 มิติบน แกน คาร์ทีเชียล 13
รูปที่ 2-24 กล่อง 3 มิติ 13
รูปที่ 2-25 ขอบเขตการมอง 14
รูปที่ 2-26 ภาพที่ได้จากเลนส์มุมกว้าง 14
รูปที่ 2-27 ภาพที่ได้จากเลนส์ปกติ 15
รูปที่ 2-28 ภาพที่ได้จากเลนส์ถ่ายไกล 15
รูปที่ 2-29 ภาพที่ได้จากกล้องที่มีระยะโฟกัสอนันต์ 15
รูปที่ 2-30 ภาพที่ได้จากกล้องที่มีระยะโฟกัสปกติ 16
รูปที่ 3-1 ภาพแสดงระบบแกน 3 มิติ 17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่

รูปที่ 3-2 รูปแสดงความแตกต่างระหว่างจำนวนของรูปสามเหลี่ยมที่ใช้ในการสุ่มค่าจากวัตถุทรงกลม 20
รูปที่ 3-3 โครงสร้างข้อมูลที่ประกอบขึ้นมาเป็นพื้นผิว 20
รูปที่ 3-4 เวกเตอร์ที่ตั้งฉากกับ เวอร์เท็กซ์และเวกเตอร์ที่ตั้งฉากกับพื้นผิว 21
รูปที่ 4-1 กราฟิกไลบรารี ที่มีใช้อยู่ในปัจจุบัน 26
รูปที่ 5-1 โครงสร้างตัวเกมส์ 28
รูปที่ 5-2 รูปแบบการเชื่อมต่อเน็ทเวิร์ค 29
รูปที่ 5-3 ความสัมพันธ์ของการทำงาน 30
รูปที่ 6-1 ค่าที่ JSB Sim สามารถจำลองได้ 32
รูปที่ 6-2 แสดงแกนต่างๆ ของเครื่องบิน 34
รูปที่ 6-3 แสดงลักษณะชื่อเรียกแทนการหมุนต่างๆ ของเครื่องบิน 34
รูปที่ 6-4 แสดงคุณลักษณะของเรดาห์ 34
รูปที่ 6-5 แสดงคุณลักษณะของเรดาห์เมื่อเครื่องบิน บินกลับหัว 35
รูปที่ 6-6 แนวชี้ขึ้นของเครื่อง แบบกลับหัว และ ไม่กลับหัว 36
รูปที่ 6-7 แสดงการก้ม-เงย ของเครื่องบิน 36
รูปที่ 6-8 แสดงการควงเครื่องบิน และมุมมองหน้าปัด ADI ภายในเครื่องบิน 37
รูปที่ 6-9 ผลกระทบที่เกิดต่อเครื่องบิน 38
รูปที่ 8-1 แบบอ้างอิง ทีซีพี/ไอพี 46
รูปที่ 8-2 โพรโตคอลแสตกของ ทีซีพี/ไอพี 47
รูปที่ 8-3 แสดงความสัมพันธ์ระหว่างวินซ็อคกับโปรโตคอลสำหรับการสื่อสารอื่นๆ 48
รูปที่ 8-4 โครงสร้างการทำงานระหว่างไคลเอ็นท์กับเซิร์ฟเวอร์ 49
รูปที่ 9-1 ภาพรวมของการทำงาน ของตัวเกมส์ 53
รูปที่ 9-2 ขั้นตอนการโหลดรูปภาพ และ วัตถุสามมิติ 54
รูปที่ 9-3 กลไกการจำลองสภาวะการบิน 55
รูปที่ 10-1 หน้าจอขณะเริ่มเกมและขณะกำลังออกตัว 58
รูปที่ 10-2 หน้าจอแสดงห้องคนขับ และ เครื่องบินขณะบิน 58
รูปที่ 10-3 เครื่องบินเมื่อมองจากภายนอก 59
รูปที่ 10-4 เครื่องบินเมื่อมองจากด้านบนและด้านข้าง 59
รูปที่ 10-5 พบศัตรู และ ทำการล๊อคเป้าหมายเพื่อโจมตี 60
รูปที่ 10-6 ยิงขีปนาวุธ ติดตามเป้าหมาย 60
รูปที่ 10-7 ศัตรูโดนโจมตี และเสียหาย 61
รูปที่ 10-8 เครื่องบินถูกยิงโดยขีปนาวุธศัตรู และเครื่องบินเสียหาย 61
รูปที่ 10-9 เล่นพร้อมกันสองคนผ่านเครือข่าย 62

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ที่มาของโครงงาน

โดยพื้นฐานนั้น อุปกรณ์ พีดีเอ (PDA) หรือ เครื่องช่วยงานส่วนบุคคลแบบดิจิทัลนั้น เป็นอุปกรณ์พกพาที่ใช้ในเก็บข้อมูล เพื่อเตือนความจำ, บันทึกตารางนัดหมาย ที่มีการทำงานแบบอิเล็กทรอนิกส์ ต่างจากการจดลงสมุดทั่วๆไป แต่ในปัจจุบันนี้ พีดีเอ มีความสามารถในเรื่องของความบันเทิงเข้ามาเกี่ยวข้องด้วย ซึ่งจะนำมาใช้ ดูหนัง-ฟังเพลง เล่นเกมส์ ใช้งานอินเตอร์เน็ท หรือแม้กระทั่งใช้เป็นโทรศัพท์มือถือก็ทำได้ ซึ่งตัว พีดีเอ ที่นิยมใช้ในท้องตลาดมีหลายประเภทและหลายยี่ห้อ โดยในโครงงานของเรานี้ จะเลือกใช้ตัว พ็อกเก็ตพีซี (PocketPC) ที่ทำงานบนระบบปฏิบัติการ วินโดว์ซีอี (Window CE) ของ บริษัทไมโครซอฟ (Microsoft)

โดยในด้านความบันเทิง เช่นการใช้เล่นเกมส์นั้น พบว่า เครื่อง พ็อกเก็ตพีซี มีขีดจำกัดอยู่ หลายด้าน ไม่ว่าจะเป็นเรื่องของความเร็วในการประมวลผล, การแสดงผลของภาพ แต่ยังคงมีสมรรถนะและความสามารถที่เพียงพอ ที่จะใช้เล่นเกมส์บางประเภทที่ต้องใช้ทรัพยากรสูง ซึ่งเกมส์ที่ใช้ทรัพยากรเครื่องสูงนั้น ส่วนมากจะเป็นเกมส์สามมิติ อย่างเช่นเกมส์จำลองการบิน (Flight Simulator Game) ที่ต้องใช้ประสิทธิภาพเครื่องในการคำนวณและประมวลผลเป็นอย่างมาก ดังนั้นในโครงงานของเรานี้จะเป็นการสร้างเกมส์สามมิติขึ้นมา โดยเป็นเกมส์ที่มีความสนุกและใช้ประสิทธิภาพที่มีอยู่อย่างจำกัดของเครื่องให้ได้อย่างเต็มที่

เพราะว่าในการเขียนเกมสามมิติซักเกมหนึ่งขึ้นมานั้น จำเป็นต้องคำนึงถึงขีดจำกัดของอุปกรณ์ ที่จะใช้ในการเล่นด้วย เพราะว่าการเขียนเกมส์สามมิติ เป็นการเขียนโปรแกรมที่ทำงานคาบเกี่ยวกับทรัพยากรของระบบเป็นอย่างมาก จึงก่อให้เกิดแนวคิดในการลดการใช้ทรัพยากรให้ได้มากที่สุด แต่ให้ได้ภาพและการเล่นที่สมบูรณ์ที่สุด ซึ่งกรรมวิธีที่ใช้ในการลดทรัพยากรนี้จำเป็นต้องใช้ทั้งศาสตร์และศิลป์ในการเขียนโปรแกรมเป็นอย่างยิ่ง

นอกจากนี้ตัวเกมส์จำลองการบิน ยังมีความสามารถในการเล่นพร้อมกันสองคนได้ โดยผ่านทางเครือข่ายอินเตอร์เน็ต จึงจำเป็นที่จะต้องประยุกต์ใช้การควบคุมทางวัตถุในเชิงสามมิติ ให้สอดคล้องกับการเชื่อมต่อเครือข่ายที่มีอยู่ด้วย

1.2 วัตถุประสงค์

1. เพื่อศึกษาวิธีการเขียนโปรแกรม บนอุปกรณ์ พ็อกเก็ตพีซี
2. เพื่อศึกษาวิธีการเขียนและใช้งานโปรแกรมทางด้านสามมิติ บนอุปกรณ์ พ็อกเก็ตพีซี
3. เพื่อศึกษาวิธีการเขียนโปรแกรมติดต่อกัน ระหว่างพ็อกเก็ตพีซีสองเครื่อง ผ่านทางระบบเครือข่ายอินเตอร์เน็ท
4. เพื่อทำการแก้ไขปัญหาที่เกิดจากการสร้างกราฟิกสามมิติ บนอุปกรณ์ที่มีทรัพยากรจำกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เพื่อทำการประยุกต์ใช้ความรู้ทางด้านคณิตศาสตร์ ฟิสิกส์ เพื่อสร้างเกมส์สามมิติที่จำลองการทำงานของเครื่องบิน
6. เพื่อทำการเขียนเกมส์ที่สามารถเล่น ผ่านระบบเครือข่ายอินเตอร์เน็ทได้

1.3 ขอบเขตของโครงงาน

ขอบเขตของโครงงานคือการสร้างเกมส์สามมิติ ที่เป็นเกมส์จำลองการบิน บนเครื่อง พ็อกเก็ตพีซี ซึ่งผู้เล่นสามารถเล่นได้ทั้งคนเดียว หรือ สองคนพร้อมกันได้ โดยผ่านทางเครือข่ายอินเตอร์เน็ต

1.4 วิธีดำเนินงาน

1. ศึกษาหลักการพื้นฐานในการพัฒนากราฟิก ในระบบ 3 มิติ
2. ศึกษาวิธีการใช้งานของ อุปกรณ์ พ็อกเก็ตพีซี
3. ศึกษาการเขียนโปรแกรม Embeded Visual C++ บน อุปกรณ์ พ็อกเก็ตพีซี
4. ศึกษาการเขียนโปรแกรมผ่านระบบเครือข่าย (Network programming)
5. ศึกษาและทดลองใช้ไลบารี (Library) ที่จะนำมาช่วยในการสร้างกราฟิก 3 มิติ บนอุปกรณ์ พ็อก เก็ตพีซี
6. นำความรู้ที่ได้มาทั้งหมดมาสร้างเกมส์ 3 มิติ บนอุปกรณ์ พ็อกเก็ตพีซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
พื้นฐานการแสดงภาพกราฟิก 3 มิติ

2.1 สภาวะแวดล้อม (Environment) ของภาพ 3 มิติ

2.1.1 ความรู้พื้นฐานเกี่ยวกับสี

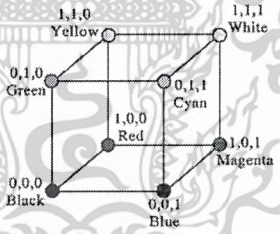
2.1.1.1 แม่สีและแบบจำลองสี

ระบบแม่สีทั้ง 3 จะประกอบด้วย สีแดง สีเขียว และ สีน้ำเงิน

รูปที่ 2-1 แม่สีทั้ง 3 สีแดง สีเขียว และ สีน้ำเงิน

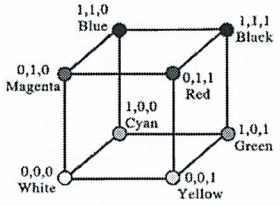
ระดับความอ่อนแก่ของแม่สี จะแสดงในตารางแม่สี ซึ่งแต่ละตัวมีความอ่อนแก่เป็นตัวเลขในช่วง ระหว่าง 0-1 โดยจะถือว่า 0 มีความสว่างต่ำสุดหรือมีความเป็นสีดำ และ 1 มีความสว่างสูงสุดจะมีความจะเป็นสีขาว

แบบจำลองแม่สี อาร์จีบี (RGB) ซึ่งแม่สี อาร์จีบี นี้จะวางแยกกัน 3 จุด



รูปที่ 2-2 แบบจำลองแม่สี อาร์จีบี

นอกจากนี้ยังมีแม่สีอีกระบบหนึ่ง คือแบบจำลองแม่สี ซีเอ็มวาย (CMY)



รูปที่ 2-3 แบบจำลองแม่สี ซีเอ็มวาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1.2 สีสัน ความสว่าง และความอิ่มตัวของสี

สีสันและความอิ่มตัวของสี เป็นความหลากหลายของสีที่เกิดขึ้นในธรรมชาติ โดยจะเกิดจากตัวแปรสำคัญอยู่ 3 อย่าง คือ สีสัน(hue),ความเปล่งแสง (luminosity),ความอิ่มตัว (saturate)

การเปล่งแสงจะเป็นตัวจัดระดับความมืดและความสว่าง ที่ปรากฏของสีใดสีหนึ่ง ระดับที่สว่างที่สุดคือ สีขาว และระดับที่มืดที่สุดคือ สีดำ

ระดับความอิ่มตัวจะเป็นตัวแสดงความสด และความหมอง ที่เกิดขึ้นจากการผสมกันของสีที่อยู่ตรงข้ามกันของวงล้อ สีที่สดที่สุดคือ สีที่ไม่ได้นำมาผสมกันเลย และสีที่หมองที่สุดคือสีผสมกันในระดับครึ่งต่อครึ่งของสีที่อยู่ตรงข้ามกันของวงล้อ

องค์ประกอบทั้ง 3 ตัวนี้รวมเรียกกันว่า เอ็กแอลเอส (XLS) เราจะพบว่าเมื่อนำแม่สีมาผสมกันจะได้สีแก่ทึบลงตามลำดับ จึงเรียกแม่สีนี้ว่า แม่สีลบ โดยสีดำเป็นอีกสีหนึ่งที่ถูกเพิ่มเข้าไปเพื่อสร้างการตัดกัน และก่อให้เกิดความทึบ ดังนั้นเพื่อประโยชน์ในการลดการสิ้นเปลือง ของการผสมของแม่สีเพื่อสร้างความทึบ เราจึงใช้สีดำ เป็นสีหลักอีกสีหนึ่ง แม่สีชุดนี้จึงเรียกรวมกันว่า ซีเอ็มวายเค (CMYK) โดยตัวเค หมายถึง สีดำ นั้นเอง

2.1.1.3 ช่วงสี และ การแปลงระบบแม่สี

ช่วงสีเป็นระยะความกว้างของสีใดสีหนึ่งที่ปรากฏออกมา โดยที่สีเหลืองเป็นสีที่มีช่วงความกว้างมากที่สุด โดยจะเห็นสีเหลืองได้แต่ไกล และสีน้ำเงินแก่ ๆ จะทำให้เห็นเป็นสีดำได้ เนื่องจากสีน้ำเงินมีช่วงสีที่แคบกว่า การกล่าวถึงช่วงสีในธรรมชาติ หมายถึง สีสเปกตัม (Spectrum Color) หรือที่มักจะเรียกว่าสีรุ้ง ซึ่งสีรุ้งเมื่อมองดูแล้วจะปรากฏเหมือนมีแค่ 7 สี แต่ความเป็นจริงจะมีช่วงความอ่อนแก่ของสีที่ ปรากฏออกมา มากมายเกินกว่าที่จะกำหนดชื่อได้ทั้งหมด

รูปที่ 2-4 สเปกตัม

การสร้างระบบสีในคอมพิวเตอร์กราฟิก จะมีความเกี่ยวข้องกันระหว่างระบบสี อาร์จีบี และซีเอ็มวาย โดยมีส่วนจำเพาะ เอ็กแอลเอส มาเกี่ยวข้องกันอย่างใกล้ชิด ทั้งช่วงสี อาร์จีบี และ ซีเอ็มวาย สามารถแสดงกลับไปกลับมาได้ โดยอาศัยสีที่บรรจุไว้ในโปรแกรมคอมพิวเตอร์กราฟิก ซึ่งทำให้สามารถสลับกันไประหว่างช่วงสีดังกล่าวได้อย่างอิสระ ทั้งนี้การแปลงจากระบบสีหนึ่งไปยังอีกระบบสีหนึ่ง เป็นการขจัดข้อมูลของช่วงสีเดิมออก แล้วแทนค่ารหัสสีที่ได้จากตารางค้นหาสีชุดใหม่ลงไปแทน

ช่วงสี ซีเอ็มวาย จะแสดงกราฟิกออกมาเป็นรูปภาพ หรือ สิ่งตีพิมพ์ต่าง ๆ, ส่วนจำเพาะ เอ็กแอลเอส มีประโยชน์ ในการแปลงระบบสีสันต่าง ๆ ให้เป็นระดับความอ่อนแก่ในภาพเฉดสีเทา (Gray Scale) ซึ่งเหมาะสำหรับการแสดงภาพขาว-ดำ, ส่วนสี อาร์จีบี เหมาะกับสื่อที่มีแสงสว่างในตัว ที่ต้องผสมแสงสีเข้าด้วยกัน ซึ่งงานเหล่านี้ได้แก่ จอสีคอมพิวเตอร์ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

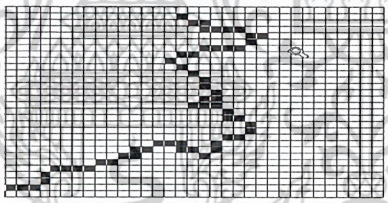
2.1.1.4 สีและระดับความคมชัดของภาพ

จำนวนสีที่แสดงออกมาให้เห็นในจอภาพเป็นผลจากจำนวนชั้นระนาบบิต (bit plane) โดยจะขึ้นอยู่กับประสิทธิภาพของคอมพิวเตอร์ และ ซอฟแวร์ที่ใช้

- ความคมชัดในระดับต่ำ หรือ ภาวะกราฟิก ซีจีเอ (Computer Graphic Adapter, CGA) ให้ความคมชัด 320 x 320 พิกเซล
- ความคมชัดในระดับกลาง หรือ ภาวะ กราฟิก วีจีเอ (Video Graphic Array, VGA) ให้ความคมชัด 640x480 พิกเซล
- ความคมชัดระดับสูง หรือ ซุปเปอร์วีจีเอ เอ (Super Video Graphic Array, SVGA) ให้ความคมชัด 800x600 พิกเซล
- ความคมชัดระดับสูงมาก หรือ เอ็กซ์วีเอ (Extended Graphic Adapter, XGA) ให้ความคมชัด 1024x768 พิกเซล จนถึง 1600x1200 พิกเซล

2.1.2 พิกเซล (Pixel)

พิกเซล เป็นหน่วยที่เล็กที่สุดของภาพกราฟิก ที่เกิดขึ้นในระบบที่ข้อมูลไม่ต่อเนื่อง ซึ่งแสดงผลบนตารางแสดงผลเมตทริกซ์สี่เหลี่ยมจัตุรัส เช่นเดียวกันกับตารางกริดหรือกระดาษกราฟ และพิกเซลที่ปรากฏออกอาจเหมือนสีที่ระบายบนตารางช่องเล็ก ๆ ซึ่งแต่ละช่องของกริดมีค่าเป็นจำนวนเต็ม

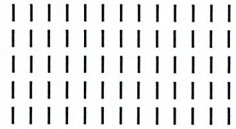


รูป 2-5 ตัวการการวางพิกเซลบนตารางกริด

2.1.3 ชนิดของเส้น (Line type)

ชนิดของเส้นจะประกอบด้วยส่วนต่าง ๆ ดังนี้คือ

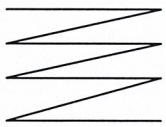
1. เส้นขีด จะแทนตัวเลขไบนารี่ ซึ่งก็คือ 0 และ 1 โดยจะใช้ 0 แทนไม่มีจุด และ 1 แทนมีจุด โดยตัวอย่างนี้จะเป็นการแทนการสลับกันระหว่างช่องว่าง และ แดช (dash)



รูป 2-6 ตัวอย่างของ เส้น (Stipple line)

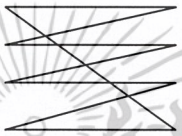
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เส้นสทริพ จะเป็นส่วนของเส้นตรงที่เชื่อมถึงกัน โดยจะมีการวาดเส้นแรก และหลังจากนั้นจะมีการกำหนดจุดตัดภายในหรือเป็นเส้นย้อนกลับ (back trace) เพื่อเริ่มต้นจุดใหม่ จากนั้นจะมีการวาดเส้นถัดไปได้เรื่อย ๆ



รูป 2-7 ตัวอย่างของ เส้นสทริพ (Line stips)

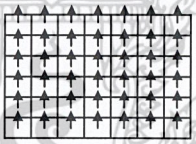
3. เส้นวนรอบจะมีลักษณะ เช่นเดียวกับ เส้นสทริพ แต่จะพบว่ามีส่วนที่เพิ่มเข้ามาก็คือ จุดที่เชื่อมถึงกันระหว่างจุดสุดท้าย และ จุดเริ่มต้น



รูป 2-8 ตัวอย่างของ เส้นวนรอบ (Line loop)

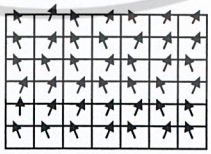
2.1.4 พื้นผิว (Texture)

พื้นผิว เป็นคุณสมบัติของผิวหน้าที่อยู่บนตัววัตถุ ซึ่งมีความแตกต่างกันโดยกายภาพ ซึ่งเราสามารถรับรู้ได้โดยการสัมผัส และ ประสบการณ์ของแต่ละบุคคล การรับรู้พื้นผิวที่มีประสบการณ์แล้วจึงไม่ต้องสัมผัสกับพื้นผิวนั้นอีก



รูปที่ 2-9 ระนาบฉากแสดงผิวหน้าแบน

จากรูปจะแสดงระนาบฉาก(normal vector) เป็นแนวเดียวกันทั้งหมด ซึ่งเป็นลักษณะของพื้นผิวแบน



รูปที่ 2-10 ระนาบฉากแสดงพื้นผิวแบบ บั๊มแมพ (Bumb map)

2.1.5 การใส่สี (Fill Color)

การวาดรูปในโปรแกรมจะมีการเรียกใช้สีแบบ อาร์จีบีโหมด ซึ่งจะมีการเก็บค่า สีแดง สีเขียว สีน้ำเงิน และ Alpha ใน อาร์จีบีโหมด สีจะเหมือน กับ ในวินโดว์ ซึ่งค่า สีแดง สีเขียว และ สีน้ำเงิน จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นการนำค่านี้มาผสมกันเกิดเป็นสีที่ต้องการ โดยค่าของสีจะมีองค์ประกอบตั้งแต่ 0-1 โดย 1 จะเป็นองค์ประกอบของสีที่มีความเข้มมากที่สุด ส่วน 0 จะน้อยที่สุด โดยการวาดภาพนั้น จำเป็นที่จะต้องทำการเคลียร์พื้นหลังเสียก่อน (Clear Background) โดยการล้างพื้นหลังนี้จะส่งผลกระทบต่อสีที่จะใส่ด้วย โดยถ้าตั้งค่าเคลียร์สีให้เป็นสีดำ จะทำให้สีที่เราจะเติมนั้น เปล่งสีได้เต็มที่จำตัวมันเอง เพราะสีดำมีค่าเป็น 0 จึงไม่เป็นตัวเพิ่มความสว่างให้แก่สีที่จะใส่ แต่ถ้าเราใช้ ค่าอื่นๆที่มากกว่า 0 ขึ้นไป จะเป็นการตั้งค่าเคลียร์สีให้เป็น สีขาวมากขึ้น ซึ่งจะทำให้สีที่เราใส่นั้น ได้รับผลกระทบจากค่าเคลียร์สีนี้ด้วย จากตัวอย่างต้องการเซ็ท องค์ประกอบ สีแดง = 0, สีเขียว = 0, สีน้ำเงิน = 1 ก็จะเป็นการตั้งค่าเคลียร์ให้เป็นสีของสีน้ำเงิน ส่วนค่า Alpha จะไม่มีผลต่อสีที่เริ่มในการตั้งค่าเคลียร์สี

หลังจากทำการเคลียร์สีเรียบร้อยแล้ว ก็จะทำการวาดรูปต่างๆ เป็นขั้นตอนต่อไป โดยการเคลียร์สีนี้ เป็นเรื่องจำเป็นอย่างมากในบางระบบที่มีการใช้งาน บัฟเฟอร์เบื้องหลัง (Background Buffer) ซึ่งเรา จำเป็นต้องเคลียร์สี และวาดรูป ต่างๆ ใน บัฟเฟอร์เบื้องหลัง ให้เรียบร้อย ก่อนที่มันจะถูกเอาไปทำเป็น ภาพ ทางเบื้องหน้า หรือ ภาพที่ปรากฏในหน้าจอ นั่นเอง

2.1.6 แสง (Light) และ แหล่งกำเนิดแสง (Light Source)

แสงในโลกแห่งความเป็นจริงนั้น มีความซับซ้อนมากมาย เพราะว่า แหล่งกำเนิดแสงในธรรมชาตินั้น จุดทุกจุดสามารถที่จะเป็นแหล่งกำเนิดแสงได้ ซึ่งหมายความว่า มันสามารถให้แสงกับวัตถุอื่นได้ แม้ว่าแสงนั้นจะสร้างขึ้นมาเอง หรือว่ารับมาจากวัตถุอื่นอีกทอดหนึ่งก็ตาม นอกจากนั้น แหล่งกำเนิดแสงสามารถที่มีรูปร่างใดๆก็ได้ไม่จำกัด จึงทำให้ในการสร้างภาพสามมิติด้วยคอมพิวเตอร์นั้น จำเป็นต้องมีแหล่งกำเนิดแสงที่ง่ายต่อการคำนวณ

2.1.6.1 แหล่งกำเนิดแสงชนิดจุด

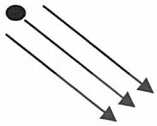
เป็นแหล่งกำเนิดแสงที่พื้นฐานที่สุด โดยเป็นแหล่งกำเนิดแสงที่ไม่มีขนาด กล่าวคือ เป็นจุดใดๆเพียงจะจุดเดียว ในพิกัดสามมิติ ที่มีทิศทางของแสงกระจากออกไปรอบ ดังนั้นการกำหนดแหล่งกำเนิดแสงชนิดนี้ จึงต้องกำหนดตำแหน่งของจุด ที่ต้องการให้เป็นตำแหน่งของแหล่งกำเนิดแสง โดยที่ไม่จำเป็นที่ต้องกำหนดทิศทางของแสง เนื่องจาก แสงจะกระจายไปในทุกทิศทาง

รูปที่ 2-11 **แหล่งกำเนิดแสงชนิดจุด**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.6.2 แหล่งกำเนิดแสงขนานระยะไกล

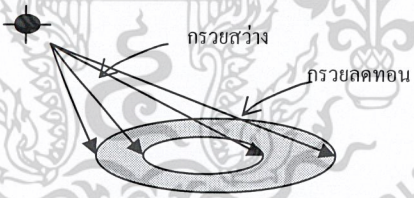
เป็นแหล่งกำเนิดแสงชนิดจุด ที่อยู่ในระยะไกล หรือ ระยะอนันต์ เนื่องจากการที่อยู่ไกลมากนี้เองจึงทำให้ทิศทางของแสงขนานกัน และมีทิศทางไปในทางเดียวกัน การกำหนดแหล่งกำเนิดแสงชนิดนี้จึงต้องการทิศทางของแสงว่ามาจากทิศทางใดเท่านั้น



รูปที่ 2-12 แหล่งกำเนิดแสงขนานระยะไกล

2.1.6.3 แหล่งกำเนิดแสงแบบสปอร์ตไลท์

เป็นการกำหนดกรวยขึ้นมาสองอัน โดยให้ยอดกรวยทั้งสองเป็นจุดเดียวกันกับแหล่งกำเนิดแสงแบบสปอร์ตไลท์ และมีแนวแกนกลางของกรวยทั้งสองเป็นทิศทางเดียวกันกับทิศทางของแสงชนิดสปอร์ตไลท์ จากนั้นจึงทำการกำหนดการลดกำลังของแสง โดยการใช้กรวยทั้งสองเป็นขอบเขตอ้างอิง ซึ่งกรวยทั้งสองจะมีมุมของยอดกรวยที่ไม่เท่ากัน ทำให้เกิดกรวยสองขนาดซ้อนกัน โดยเราจะเรียกกรวยข้างในที่เล็กกว่าว่า กรวยสว่าง (Hot spot cone) และกรวยข้างนอกที่ใหญ่กว่าจะเรียกว่า กรวยลดทอน (Fall off cone) โดยที่ภายในกรวยสว่าง จะเป็นส่วนที่แสงมีความสว่างที่คงที่ จากนั้นจะค่อยๆลดลงเมื่อพ้นออกไปจาก กรวยสว่าง จนกระทั่งมีค่าเป็นศูนย์ เมื่อพ้นจากกรวยลดทอน ออกไป



รูปที่ 2-13 แหล่งกำเนิดแสงแบบสปอร์ตไลท์

2.1.6.4 แสงแบบแอมเบียนท์

แสงแบบแอมเบียนท์ ไม่ใช่แหล่งกำเนิดแสงโดยตรง แต่เป็นแสงที่ถูกสร้างขึ้น เพื่อทดแทนปริมาณแสงที่ได้จากแสงในธรรมชาติ นั่นคือ เป็นแสงที่เพิ่มปริมาณแสงให้กับทุกๆจุดบนพื้นผิววัตถุด้วยปริมาณที่เท่าๆกัน ซึ่งปริมาณแสงที่ได้ค่อนข้างจะคงที่ และแตกต่างจากปริมาณแสงที่ได้จริงจากธรรมชาติอยู่มาก แต่ถ้าหากใช้ร่วมกับแหล่งกำเนิดแสงอื่นๆแล้ว จะให้ผลที่ดีกว่าการใช้แหล่งกำเนิดแสงตัวใดตัวหนึ่งเพียงลำพัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.7 คุณสมบัติทางแสงของวัตถุ (Material)

คุณสมบัติทางแสงของวัตถุ (Material) เป็นคุณสมบัติที่อยู่ในตัววัตถุในแต่ละชนิดและมีผลต่อความรู้สึก ในแง่ของการสัมผัส โดยมันจะเป็นข้อมูลจำเพาะของวัตถุนั้นๆ ที่จะแสดงถึงผลกระทบทางแสงที่ต่างกันออกไป ทำให้วัตถุ ดูเรียบลื่น มันวาว หรือว่า ดูหม่นหมอง ได้ต่างกันออกไป โดย คุณสมบัติเหล่านี้เกิดจากการได้สัมผัส กับ วัสดุจริงเสียก่อน เช่น สี , ความแข็ง , ความนุ่ม , โปร่งแสงและทึบแสง , ความสากหยาบ, ความเรียบ , ความวาวสะท้อนแสง เป็นต้น

2.1.7.1 คุณสมบัติของการกระจายแสง

เป็นคุณสมบัติพื้นฐานของวัตถุ ที่จะส่งผลกระทบต่อแสงโดยตรง นั่นคือ เป็นการกำหนดให้วัตถุนั้นๆ มีการดูดซับแสงมากน้อยเพียงใด ซึ่งการเปลี่ยนคุณสมบัตินี้ในสีใดสีหนึ่งจากแม่สีทั้งสาม จะทำให้วัตถุนั้น ดูดซับแสงได้ต่างต่างสีกันกัน เช่นแสงสีขาว ส่องกระทบวัตถุที่มีคุณสมบัติกระจายแสงสีแดง = 1, สีเขียว = 0, สีน้ำเงิน = 0 จะทำให้วัตถุชิ้นนั้น สะท้อนแต่แสงสีแดง และ ดูมีสีแดง และในทางกลับกัน ถ้าเราส่องแสงสีเขียว หรือสีน้ำเงิน กระทบลงบนวัตถุชิ้นนี้ มันจะไม่เกิดเป็นแสงสีใดๆ สะท้อนมาให้เห็นเลย

2.1.7.2 คุณสมบัติของการรับแสงแอมเบียนท์

เป็นคุณสมบัติของวัตถุที่ส่งผลกระทบต่อแสงแอมเบียนท์โดยตรง ซึ่งโดยปกติ แสงแอมเบียนท์จะส่องไปยังพื้นผิวใดๆ เท่าๆกันทุกๆจุด ดังนั้น ถ้าการเปลี่ยนแปลงค่าคุณสมบัติการรับแสงแอมเบียนท์ให้เปลี่ยนไป จะทำให้เกิดความหลากหลายของวัตถุได้ แม้ว่าได้รับแสงแอมเบียนท์ เหมือนๆกัน

2.1.7.3 คุณสมบัติของการสะท้อนแสง

คุณสมบัติข้อนี้เป็นการตั้งค่าให้เกิดความมันวาวของวัตถุ โดยถ้ายิ่งมีค่าสูง จะยิ่งสะท้อนแสงมาก โดยตัวอย่างที่เห็นได้ชัดเจนคือ วัตถุที่เป็นลูกบิลเลียดสีขาว กับ ลูกปิงปองสีขาว ซึ่งทั้งสองนี้ มีค่าคุณสมบัติการสะท้อนแสงที่แตกต่างกันอย่างเห็นได้ชัด เพราะว่า ลูกบิลเลียด ดูเรียบลื่นมันวาว ในขณะที่ลูกปิงปองจะไม่ค่อยดู วาว เท่าไรนัก

2.1.7.4 คุณสมบัติการเปล่งแสง

ค่าคุณสมบัตินี้ จะเป็นการตั้งให้วัตถุนั้น เปล่งแสงได้ในตัวเอง ซึ่งการเปล่งแสงของวัตถุนี้ ไม่ใช่เป็นการสร้างแสงเหมือนแหล่งกำเนิดแสง เพราะว่าคุณสมบัติการเปล่งแสงนี้ จะไม่ส่งแสงกระทบไปยังวัตถุอื่น มันเพียงแต่ดูเรืองรอง ในความมืดเท่านั้น

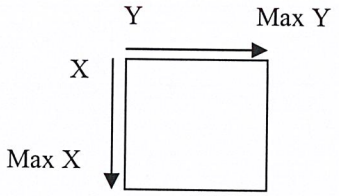
2.2 วัตถุ 2 มิติ (2D Geometry Component)

2.2.1 จุด (Point)

ตำแหน่ง x และ y ใน แกน ของ ระบบวินโดว์ (windows) นั้น การเพิ่มขึ้นของ x จะเคลื่อนที่ไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทางขวา ส่วน y จะไปทิศทางลง ซึ่งจะเป็นระบบที่โปรแกรมเมอร์ คุ้นเคยกัน

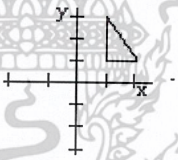


รูปที่ 2-14 ระบบแกนของ วินโดว์ (Window Coordinate System)

แต่ในระบบ คาร์ทีเชียล (cartesian coordinate system) จะแสดงดังรูป2-21และ จะพบว่าจะมีด้านลบปรากฏอยู่ ดังรูป

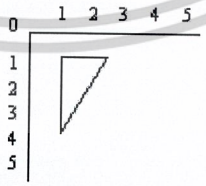
รูปที่ 2-15 ระบบคาร์ทีเชียล (cartesian coordinate system)

วัตถุ ในระบบ คาร์ทีเชียล จะต้องมีการใช้จุด ซึ่งจะมีเส้นที่ทำให้วัตถุเชื่อมถึงกัน ดังตัวอย่างรูป 2-22 จะเป็นรูป 3 เหลี่ยม ที่มีจุด (2,5) (2,2) และ (6,2) ประกอบกันเป็นสามเหลี่ยม



รูปที่ 2-16 แสดงตัวอย่างการสร้างรูป 3 เหลี่ยม

ซึ่งเราสามารถแปลงจากระบบ คาร์ทีเชียล มา เป็น ระบบ วินโดว์ ได้ดังรูป2-23 โดยจะมีการกลับด้านของ Y

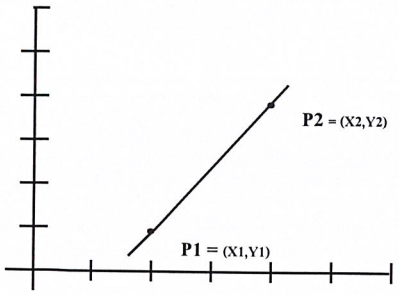


รูปที่ 2-17 แสดงตัวอย่างการแปลงจากคาร์ทีเชียล มาเป็นระบบ วินโดว์

2.2.2 เส้น (Line)

เส้นตรงเกิดจากการเชื่อมโยงสองจุดเข้าด้วยกัน ดังนั้นการวาดเส้นตรงจึงต้องกำหนดจุดเริ่มต้นของจุด ไปสู่จุดหมายหรือจุดสิ้นสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



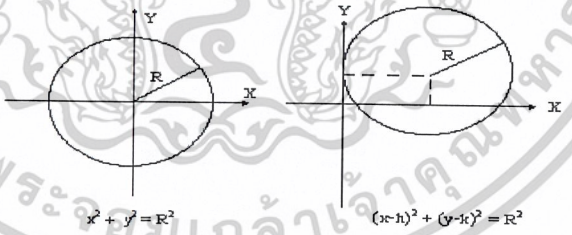
รูปที่ 2-18 แสดงการสร้างเส้น

2.2.3 เส้นต่อเนื่อง (Pline)

โพลีกอน จะเป็นส่วนหนึ่งของ เส้นต่อเนื่อง โดยสร้างจากการวาดเส้นระหว่าง กลุ่มของจุด โดยตัว Polygon นั้น อย่างน้อยต้องมี 2 เหลี่ยม แต่จะไม่มีจำนวนที่มากที่สุด เช่น Stippled Polygon , Outlined Non convex Polygons ,Triangle ,Triangle Strips ,Triangle Fans , Quadrilaterals , Quadrilaterals Strips เป็นต้น

2.2.4 วงกลม และ สมการ วงกลม

วงกลม คือ เซตของจุดทุกจุดบนระนาบ ซึ่งอยู่ห่างจากจุดคงที่จุดหนึ่งเป็นระยะทางเท่ากันเสมอ โดยเรียกจุดคงที่ว่า จุดศูนย์กลางของวงกลม และ เรียกระยะทางคงที่ ว่า รัศมีของวงกลม



รูปที่ 2-19 แสดงสมการ ของวงกลมที่มีจุดศูนย์กลางอยู่จุดกำเนิด และไม่ได้อยู่จุดกำเนิด

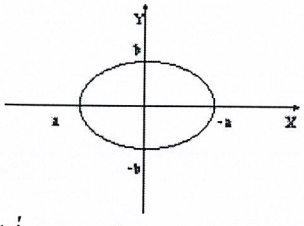
2.2.5 วงรี

วงรีที่มีจุดศูนย์กลางอยู่ที่จุด 0,0 จะมีสมการทางคณิตศาสตร์ คือ

$$F(x,y) = b^2x^2 + a^2y^2 - a^2b^2 = 0$$

ซึ่งจะแสดงดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

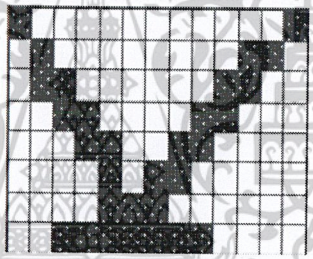


รูปที่ 2-20 วงรีบนแกน x,y

จากรูป2-36 นั้น b จะเป็นความยาวแกน y และ a จะเป็นความยาวแกน x

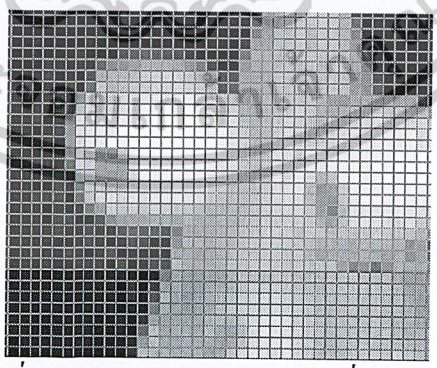
2.2.6 บิตแมพ (Bitmap)

ภาพบิตแมพที่เรียบง่ายที่สุด คือ ภาพที่มีเพียงระนาบบิตเดียว ระนาบบิตเป็นตารางของ บิตแมพเพียงหนึ่งแผ่นที่แสดงเพียงด้านเดียว แสดงสถานะในหน่วยความจำคอมพิวเตอร์ และแต่ละพิกเซล มีความหมายของข้อมูลในลักษณะ ปิด และ เปิดสวิตช์ ซึ่งเป็นข้อมูลที่ไม่อาจแบ่งออกเป็นเศษได้เลย โดยภาพกราฟิกเพียงระนาบเดียวจะแสดงผลเพียง ดำกับขาว ซึ่งตัดกันสูงมาก



รูปที่ 2-21 ภาพแสดงการเรียงตัวของพิกเซลในหนึ่งระนาบบิต

จากรูปข้างล่างจะแสดงลักษณะว่ายังพิกเซลมีขนาดเล็กยิ่งจะมีความคมชัดสูงกว่า

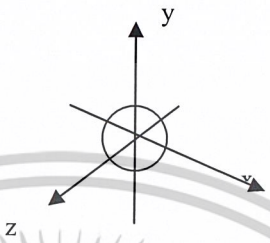


รูปที่ 2-22 ความเข้มและอัตราส่วนที่คมชัดกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 วัตถุ 3 มิติ (3D Geometry Component)

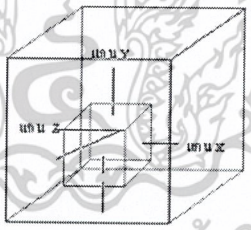
วัตถุในรูปทรง 3 มิติ มีวิธีการสร้างขึ้นมาได้ หลายวิธี นับตั้งแต่การป้อนผ่าน อุปกรณ์อินพุทต่าง ๆกันไป โดยวัตถุ 3 มิติ ในระบบ คาร์ทีเซียล นั้น จะประกอบไปด้วยระนาบต่าง ๆ ที่วางอยู่บนแกนหลัก 3 แกน คือ x,y,z ดังรูปที่ 2-41



รูปที่ 2-23 ระบบ 3 มิติบน แกน คาร์ทีเซียล

วัตถุบนโลก 3 มิติ ความจริงแล้วถูกสร้างขึ้นมาจากการกำหนดจุด ลงบนที่ว่างเปล่าก่อน เหมือนกับจุดบนดวงดาวในอวกาศที่ว่างเปล่า หากเราใช้สายตาโยงจุดดาว หลายจุดเข้าด้วยกัน ก็จะเกิดเป็นเส้นและรูปร่างในที่สุด ทั้งนี้เราจะพบความสัมพันธ์ ของตำแหน่งสังเกตการณ์ โดยการเปลี่ยนตำแหน่งดวงดาว ทั้งนี้ดวงดาวแต่ละดวงลอยอยู่ในอากาศที่เป็นที่ว่าง 3 มิติ

ในสิ่งแวดล้อมรอบตัวเรา ล้วนแต่เป็นสิ่งแวดล้อม 3 มิติ เช่นห้องเรียนจะมีลักษณะ มีด้านกว้าง , ยาว , สูง ทำให้มองเป็นวัตถุ 3 มิติ



รูปที่ 2-24 กล่อง 3 มิติ

2.3.1 สี และ ความสว่างของวัตถุ

ลวดลายของสีบนพื้นผิวของวัตถุสามมิติ เป็นภาพบิตแมพที่ได้จากการวดภาพวัตถุจริง แล้วนำมา แปลงส่งลงบนตัววัตถุสามมิติ ในโปรแกรมคอมพิวเตอร์กราฟิก หรือ เกิดจากใช้โปรแกรมคอมพิวเตอร์กราฟิก และ ความสว่าง เป็นปริมาณของแสดงที่ผิววัตถุเปล่งแสง หรือ เรืองแสดงออกมา

2.4 การมอง

2.4.1 การปรับภาพ และ การกวาดภาพ (Zoom and Pan)

- การกวาดกล้อง (Pan) จะเป็นการเลื่อนระนาบภาพไปบนตำแหน่งต่าง ๆ ในช่องว่างสามมิติทำให้ภาพเปลี่ยนไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การ ปรับภาพ (Zoom) จะเป็นค่าของกลุ่มพิกเซลที่ถูกสเกล และถูกเขียนในหน่วยความจำบัพเฟอร์ แต่ลายละเอียดในข่าวสารไม่ได้เพิ่มขึ้นแต่อย่างใด สิ่งที่เพิ่มขึ้นคือ ขนาดของ แฟ้มข้อมูล ที่มีผลต่อความคมชัดของภาพเท่านั้น

2.4.2 วิว (View)

2.4.2.1 ขอบเขตการมองเห็น

การมองภาพจะประกอบด้วยองค์ประกอบ 2 ส่วนคือ

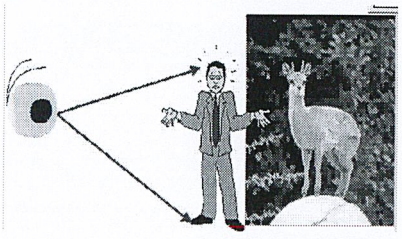
- จุดมองเป็นตำแหน่งของการมองเห็น ภาพที่อยู่บนที่ว่างสามมิติ จุดดังกล่าวสามารถกำหนดด้วยพิกัด x,y,z โดยจุดนี้อาจเทียบเคียงได้กับการมองเห็นของบุคคล หรือ ตำแหน่งของกล้องถ่ายรูป
- ขอบเขตมอง จะเป็นกรวยที่ฉายออกจากตำแหน่งสังเกตการณ์ โดยส่วนฐานของกรวยเป็นระนาบภาพ (Picture plane) ซึ่งเป็นพื้นระนาบที่มีภาพปรากฏอยู่

รูปที่ 2-25 ขอบเขตการมอง

2.4.2.2 การใช้เลนส์ ชนิดต่างๆ ในการมองภาพ

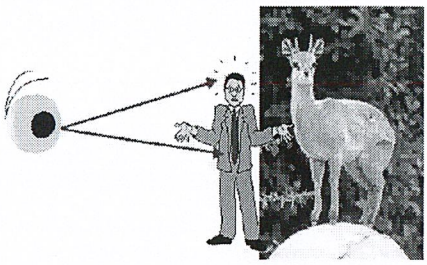
การใช้เลนส์ในโลกของความเป็นจริงนั้น มีค่าเทียบเท่ากับการปรับค่า มุมการมอง (Field of View, FOV) ในการเขียนโปรแกรมสามมิติ ซึ่งการปรับค่าเหล่านี้จะทำให้เกิดมุมมองที่ต่างกันออกไป

- ภาพจากเลนส์ปกติ หรือมีมุมการมอง เท่ากับ 60-90 องศา จะให้ภาพในมุมมองที่ได้จากสายตาของมนุษย์
- ภาพจากเลนส์มุมกว้าง หรือมีมุมการมองมากกว่า 90 องศา จะให้ภาพที่เป็นภาพพาโนรามา หรือเป็นภามพที่มีมุมมองกว้าง
- ภาพจากเลนส์ถ่ายไกล หรือมีมุมการมองน้อยกว่า 60 องศา จะให้ภาพที่มีมุมมองที่แคบ แต่จะมีความยาวโฟกัสยาว หรือพูดอีกนัยหนึ่ง คือ เหมือนกันการใช้กล้องส่องทางไกลส่องดูทิวทัศน์

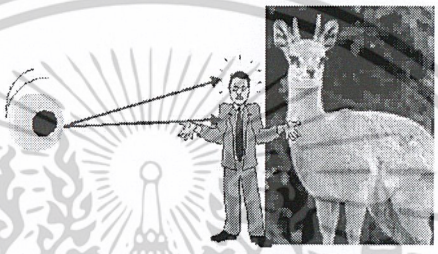


รูปที่ 2-26 ภาพที่ได้จากเลนส์มุมกว้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



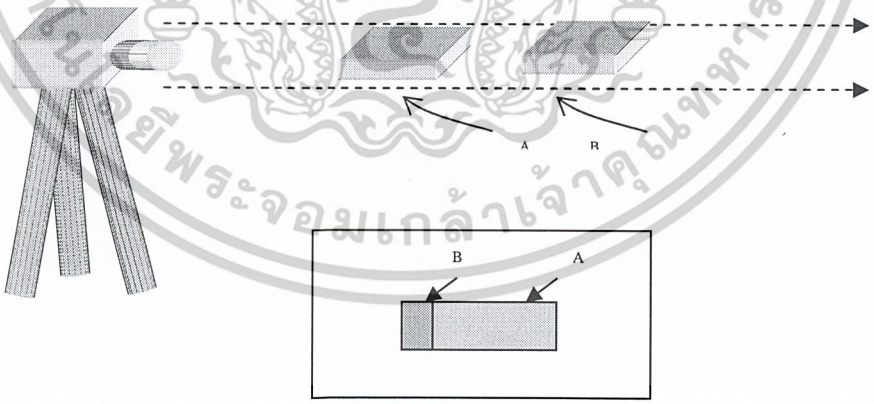
รูปที่ 2-27 ภาพที่ได้จากเลนส์ปกติ



รูปที่ 2-28 ภาพที่ได้จากเลนส์ถ่ายไกล

2.4.3 ภาพ เพอร์สเปกทีพ (Perspective)

ในการใช้กล้องที่มีโฟกัสอยู่ที่ระยะอนันต์นั้น จะได้ภาพในเชิงขนาน ซึ่งเป็นภาพที่ไม่มีมิติเชิงลึกเกิดขึ้นในภาพเลย

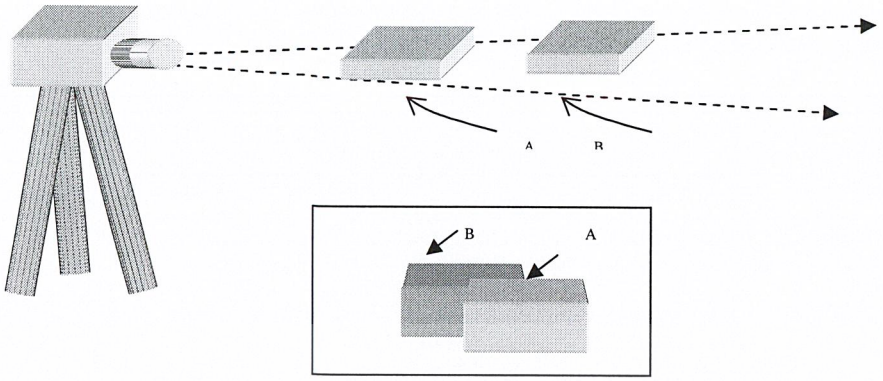


รูปที่ 2-29 ภาพที่ได้จากกล้องที่มีระยะโฟกัสอนันต์

จากรูปที่2-29 เราจะได้รูปที่ไม่มีลักษณะของมุมมองเชิงทิวทัศน์ เลย แต่จะได้ภาพฉาย (Projection) มาแทน ซึ่งลักษณะนี้ ไม่ใช่ลักษณะของภาพในเชิงสามมิติ

แต่ถ้าหากเราจัดโฟกัสให้อยู่ในระยะที่เหมาะสม เราก็จะได้ภาพเชิงสามมิติ หรือ ภาพเพอร์สเปกทีพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-30 ภาพที่ได้จากกล้องที่มีระยะโฟกัสปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

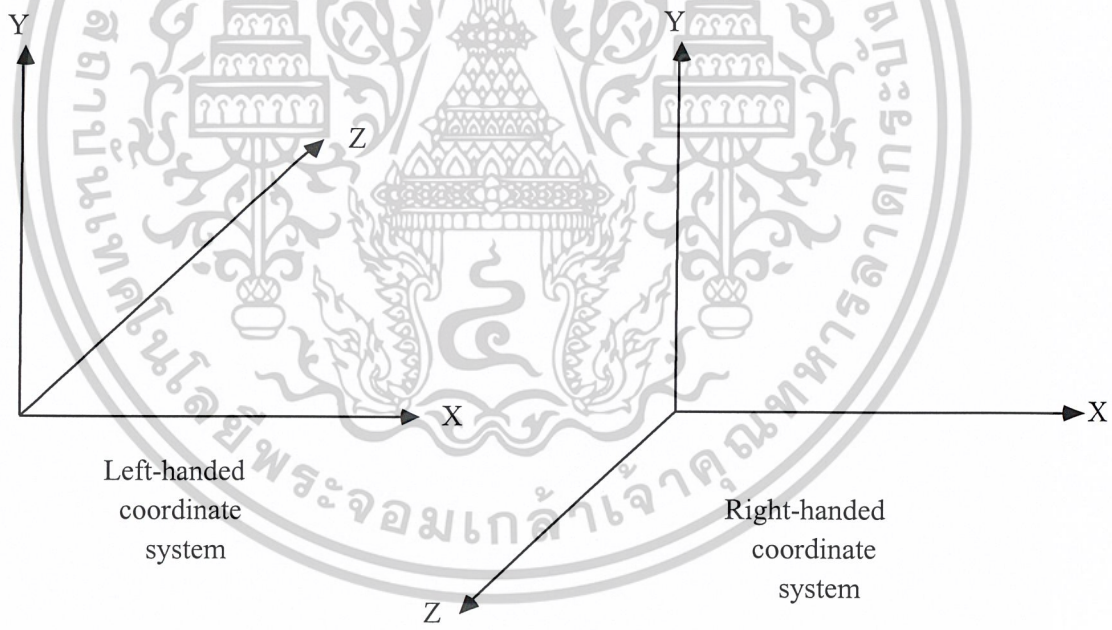
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

บทที่ 3
คณิตศาสตร์พื้นฐานของระบบกราฟิก 3 มิติ

3.1 ระบบแกน 3 มิติ (3D Coordinate Systems)

ระบบคู่อันดับที่ใช้ในโปรแกรม 3 มิติส่วนใหญ่เป็นระบบคู่อันดับคาร์ทีเซียน (Cartesian) ซึ่งมี 2 ระบบ คือ ระบบมือขวาและระบบมือซ้าย ระบบคู่อันดับทั้ง 2 ระบบนั้นแกน x ในส่วนที่มีค่าบวกจะชี้ไปทางขวามือ และแกน y ในส่วนที่มีค่าเป็นบวกก็จะชี้ขึ้นด้านบนเสมอ ส่วนทิศทางของแกน z สามารถหาได้โดยการใช้หลักมือซ้ายและมือขวาเช่นเดียวกับชื่อของระบบคู่อันดับแต่ละชนิด โดยถ้าเป็นระบบมือซ้ายก็ให้ใช้มือซ้ายและถ้าเป็นระบบมือขวาก็ให้ใช้มือขวา

วิธีหาทิศทางของแกน z เริ่มต้นแบมือออกแล้วชี้ปลายนิ้วมือไปในทิศทางของแกน x ด้านที่มีค่าเป็นบวก หลังจากนั้นให้ งอนิ้วมือจนอยู่ในทิศทางของแกน y ด้านที่มีค่าเป็นบวก ทิศทางของหัวมือคือทิศทางของแกน z ในด้านที่ มีค่าเป็นบวกซึ่งจะมีทิศทางชี้เข้าหรือออกจากตัวเราเสมอ



รูปที่ 3-1 ภาพแสดงระบบแกน 3 มิติ

3.2 จุดหรือเวกเตอร์

จุดหรือเวกเตอร์เป็นข้อมูลที่มีความสำคัญมากในกราฟิก 3 มิติ เนื่องจากข้อมูลต่างๆ ไม่ว่าจะเป็นเวอร์เท็กซ์ (Vertex) ซึ่งใช้เป็นข้อมูลในแสดงโมเดล, ตำแหน่งและทิศทางของวัตถุ และความเร็ว/แรง ทั้งหมดสามารถแทนค่าในรูปของจุดในระนาบ 3 มิติหรือเวกเตอร์ได้ทั้งหมด ในปริภูมิ 3 มิติวัตถุประกอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
55084

ด้วยความกว้าง (width) คือ ค่าในแกน x, ความยาว (height) คือ ค่าในแกน y, และความลึก (depth) คือ ค่าในแกน z ดังเช่น

$$v = (x , y, z)$$

และการแสดงค่าของจุดหรือเวกเตอร์ในรูปเมตริกซ์ จะเป็นดังนี้

$$v = [x \;\; y \;\; z]$$

เนื่องจากคำว่าจุดและเวกเตอร์มีความหมายไปในทางเดียวกัน คือ จุด หมายถึง ตำแหน่งในระบบแกน 3 มิติ เวกเตอร์ หมายถึง เส้นที่ลากจากตำแหน่งอ้างอิง (จุดกำเนิด) มายัง ตำแหน่งในปริภูมิ 3 มิติ เวกเตอร์มีคุณสมบัติมากกว่าจุดคือ สามารถบอกทิศทางได้ ในที่นี้จึงใช้สลับกันได้ แต่ถ้าใช้ในการคำนวณจะกล่าวถึงในรูปของเวกเตอร์

<u>การหาความยาวของเวกเตอร์</u>

$$\|v\| = \sqrt{x^2 + y^2 + z^2}$$

<u>การหาเวกเตอร์หนึ่งหน่วย</u> (unit vector) เวกเตอร์หนึ่ง คือ เวกเตอร์ที่มีขนาดหนึ่งหน่วยเสมอ ไม่ว่าจะอยู่ในทิศทางใดก็ตาม

$$u = \frac{v}{\|v\|}$$

เวกเตอร์หนึ่งหน่วย ที่ใช้แสดงทิศทางในแนวแกนต่างๆ ดังนี้

แกน x: i = (1, 0, 0)

แกน y: j = (0, 1, 0)

แกน z: k = (0, 0, 1)

เนื่องจากเวกเตอร์มีการนำไปใช้มากในทางฟิสิกส์ซึ่งมักจะเขียนเวกเตอร์ในรูปของค่าเวกเตอร์หนึ่งหน่วย i, j, k ทำให้เราสามารถแสดงค่าจุดในปริภูมิ 3 มิติในรูปของเวกเตอร์ได้โดยนำค่าในแต่ละแกนมาคูณกับเวกเตอร์หนึ่งหน่วยเหล่านี้ เช่น ถ้าต้องการแสดงจุด a = (3,5,2) ในรูปของเวกเตอร์ จะได้ วิธีการแสดงค่าเวกเตอร์ในรูปเวกเตอร์หนึ่งหน่วยจะมีความสำคัญในการคำนวณค่ากับเมตริกซ์ ดังสมการต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$a = 3i + 5j + 2k$$

ผลคูณเชิงสเกล่าร์ของเวกเตอร์ (Scalar product or Dot product) ผลคูณของเวกเตอร์ 2 เวกเตอร์ที่ได้ผลลัพธ์ เป็นสเกลาร์ ซึ่งค่าสเกลาร์ที่ได้แสดงค่าความยาว ของเงาที่ฉายอยู่บนอีกเวกเตอร์หนึ่ง จะเป็นดังสมการ ในระบบ 3 มิติ

$$u \cdot v = u_x \times v_x + u_y v_y + u_z v_z$$

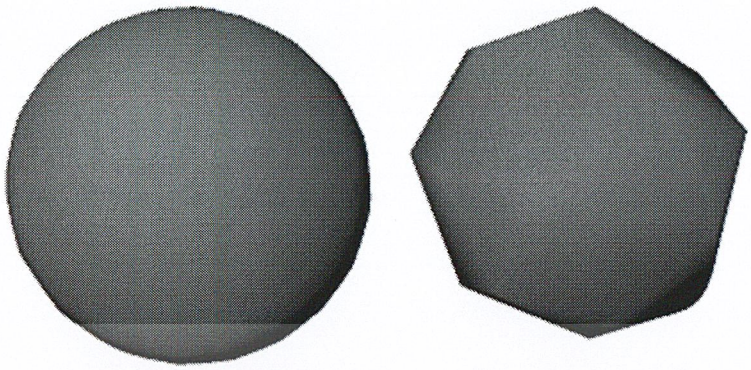
ผลคูณเชิงเวกเตอร์ของเวกเตอร์ (Cross Product) คือ ผลคูณของเวกเตอร์ 2 เวกเตอร์ที่ได้ผลลัพธ์เป็นเวกเตอร์ ซึ่งเวกเตอร์ที่ได้จะตั้งฉากกับเวกเตอร์ทั้งสองเสมอ การคูณเชิงเวกเตอร์ของ 2 เวกเตอร์ไม่มีคุณสมบัติในการสลับที่ดังนั้น v u . ไม่เท่ากับ u v . และถ้า 2 เวกเตอร์ขนานกันจะหาค่าผลคูณไม้ได้ (อาจจะได้เวกเตอร์ที่มีค่าอนันต์ (infinity) ซึ่งตั้งฉากกับ เวกเตอร์หนึ่ง) ดังสมการข้างล่าง

$$u \times v = (u_y v_z - u_z v_y, u_z v_x - u_x v_z, u_x v_y - u_y v_x)$$

3.3 โพลีกอน (Polygon)

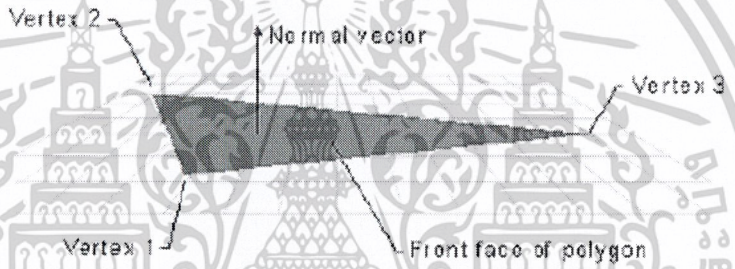
โพลีกอนเป็นพื้นฐานในสาขาวิชาคอมพิวเตอร์กราฟิก ถึงแม้ว่าในปัจจุบันจะมีเทคโนโลยีอย่างอื่น เช่น NURBS เป็นต้น แต่อุปกรณ์ที่ใช้กับเครื่องคอมพิวเตอร์ส่วนบุคคลทุกวันนี้ส่วนใหญ่ยังคงใช้หลักการวาดรูปด้วยโพลีกอน ไลบรารี่สามมิติในปัจจุบันส่วนใหญ่ ก็ใช้โพลีกอนในการสร้างภาพของวัตถุ ซึ่งโพลีกอน ก็คือพื้นผิวที่เกิดจากเส้นรอบรูป ที่มีลักษณะเป็นวงปิด (โพลีกอนสนใจเฉพาะพื้นผิวไม่สนใจปริมาตรของวัตถุ) ซึ่งเส้นรอบรูปของโพลีกอน จะมีรูปร่างเป็นรูปทรงเรขาคณิต เช่น สามเหลี่ยม, สี่เหลี่ยม เป็นต้น หลักการในการสร้างรูปทรงของวัตถุ โดยใช้โพลีกอนก็คือ การใช้พื้นผิวพื้นฐาน (primitive) ประกอบกัน ในการสร้างรูปทรงที่มีความสลับซับซ้อนมากๆ ขึ้นไป และรูปทรงที่พื้นฐานที่สุดก็คือ รูปสามเหลี่ยมนั่นเอง ดังนั้นโดยอาศัยหลักการของโพลีกอนนี้ เราสามารถสร้างพื้นผิวของวัตถุที่มีรูปร่างใดๆ ก็ได้โดยไม่มีข้อจำกัด อย่างไรก็ตาม รูปทรงที่สร้างขึ้นจากโพลีกอนจะเป็นการประมาณค่าพื้นผิวของวัตถุโดยใช้รูป สามเหลี่ยมสุ่มค่าจุดทั้งสามขึ้นมาจากพื้นผิวจริง ซึ่งถ้าจำนวนครั้งที่เราสุ่มน้อยเกินไป (มีรูปสามเหลี่ยมน้อย) ก็อาจจะทำให้วัตถุที่มีความซับซ้อนสูงๆ มีการคาดเคลื่อนไป เช่น ถ้าเราใช้รูปสามเหลี่ยมจำนวน น้อยในการสร้างรูปทรงกลมจะทำให้ได้รูปหลายเหลี่ยมแทน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-2 รูปแสดงความแตกต่างระหว่างจำนวนของรูปสามเหลี่ยมที่ใช้ในการสุ่มค่าจากวัตถุทรงกลม

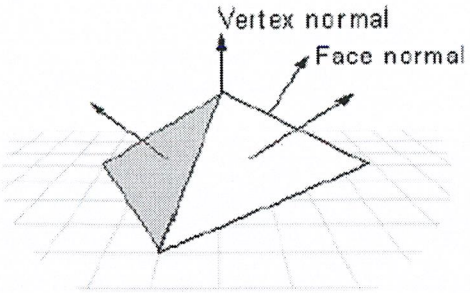
โครงสร้างข้อมูลของ โพลีกอน



รูปที่ 3-3 โครงสร้างข้อมูลที่ประกอบขึ้นมาเป็นพื้นผิว

ในการสร้างรูปสามเหลี่ยม 1 รูป จะต้องใช้จุดสามจุดในการกำหนดตำแหน่งความยาวของด้านแต่ละด้านของโพลีกอน ซึ่งจุดบนโพลีกอนมีชื่อเรียกว่า เวอร์เท็กซ์ เนื่องจากมันไม่ได้เก็บเฉพาะข้อมูลเกี่ยวกับ ตำแหน่งเท่านั้น แต่ยังประกอบด้วยข้อมูลอื่นๆ ที่ใช้กำหนดคุณลักษณะ ของพื้นผิวที่จะปรากฏอีกด้วย ข้อมูลเหล่านั้น ตัวโปรแกรมสร้างภาพสามมิติ สามารถนำไปประมวลผลเพื่อสร้างภาพที่มีความสมจริงยิ่งขึ้นได้ เช่น ค่าเวกเตอร์ตั้งฉากของเวอร์เท็กซ์ (normal vector) สามารถนำไปใช้คำนวณมุมที่แสงกระทำต่อวัตถุทำให้ ได้วัตถุที่มีสีสันเหมือนจริงยิ่งขึ้น, ข้อมูลน้ำหนักในการเบลนด์ (blend weight data) ใช้ในการทำ อนิเมชัน ให้กับวัตถุ ซึ่งทำให้ได้วัตถุที่มีการเคลื่อนไหวอย่างเป็นธรรมชาติ และคู่อันดับของเทกเจอร์ (texture coordinate) ใช้กำหนดคู่อันดับของภาพที่จะแปะบนวัตถุ เพื่อให้ได้วัตถุที่มีสีสันของพื้นผิวเหมือนจริงและเป็น ธรรมชาติมากขึ้น เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-4 เวกเตอร์ที่ตั้งฉากกับ เวอร์เท็กซ์และเวกเตอร์ที่ตั้งฉากกับพื้นผิว

3.4 ระนาบ (Plane)

เป็นข้อมูลพื้นฐานที่สำคัญอีกอย่างหนึ่ง ระนาบในทาง 3 มิติก็มีความหมายนัยเดียวกับเส้นตรงที่มีความยาวอนันต์ใน 2 มิติ หรือก็คือ สิ่งที่อยู่ในมิติ n-1 นั่นเอง ซึ่งจะแบ่งปริภูมิหรือระนาบเป็น 2 ส่วนแยกจากกันโดยเด็ดขาด สามารถนำไปงานใช้ในการคำนวณหลายๆ อย่างใน 3 มิติ เช่น การหาว่าจุดหรือโพลีกอนอยู่ในระนาบเดียวกันหรือไม่, ใช้ในการแบ่งโพลีกอนที่อยู่ในบริเวณที่จะแสดงผล (clipping) และใช้ในการคำนวณหา BSP Tree เป็นต้น โดยสมการระนาบในปริภูมิ 3 มิติ คือ

$$ax + by + cz + d = 0$$

นอกจากนั้น เรายังสามารถหาค่าอื่นๆ ที่เกี่ยวข้องกับระนาบได้ ดังนี้

1. ค่า (a , b, c) คือเวกเตอร์ที่ตั้งฉากกับระนาบ
2. ค่า d คือระยะทางตั้งฉากจากระนาบกับจุดกำเนิด
3. เมื่อนำค่าจุดใดๆ มาแทนค่าใน x, y, z สามารถหาว่าจุดนั้นอยู่ในระนาบหรือไม่ ถ้าอยู่ในระนาบจะได้ ค่าผลลัพธ์เป็น 0

3.5 เมตริกซ์ (Matrix)

เมตริกซ์ช่วยให้เราสามารถแสดงสมการหลายๆ สมการได้พร้อมๆ กัน เช่น ถ้าเราต้องการจะแก้สมการที่มีตัวแปร x, y, z ดังนี้

$$3x - 8y + 12z = 0$$
$$15x + 14y - 2z = 0$$
$$32x + 0.5y - z = 0$$

เมื่อนำค่าสัมประสิทธิ์ของสมการมาใส่ในกรอบขนาด n คูณ m ที่เรียกว่า เมตริกซ์ เมื่อ n คือจำนวนของคอลัมภ์ (vertical dimension) และ m คือ จำนวนของแถว (horizontal dimension) จะได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{array}{l} 3x-8y+12z=0 \\ 15x+14y-2z=0 \Rightarrow \\ 32x+0.5y-z=0 \end{array} \begin{bmatrix} 3 & -8 & 12 \\ 15 & 14 & -2 \\ 32 & 0.5 & 1 \end{bmatrix}$$

สัญลักษณ์ที่ใช้แทนเมตริกซ์ขนาด m*n

$$A_{m\times n} = [a_{ij}]_{m\times n} = \begin{bmatrix} a_{11} & a_{12} & . & . & . & a_{1n} \\ a_{21} & a_{22} & . & . & . & a_{2n} \\ . & . & & & & . \\ . & . & & & & . \\ . & . & & & & . \\ a_{m1} & a_{m2} & . & . & . & a_{mn} \end{bmatrix}$$

<u>การบวกเมตริกซ์</u> นิยาม

ถ้า $A=[a_{ij}]_{m\times n}$ และ $B=[b_{ij}]_{m\times n}$ แล้ว $A+B=[a_{ij}+b_{ij}]_{m\times n}$

ตัวอย่าง เช่น

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11}+b_{11} & a_{12}+b_{12} \\ a_{21}+b_{21} & a_{22}+b_{22} \end{bmatrix}$$

<u>การคูณเมตริกซ์</u> นิยาม

ถ้า $A=[a_{ij}]_{m\times n}$ และ $B=[b_{ij}]_{n\times r}$ แล้ว ผลคูณ $A\times B$ คือ เมตริกซ์ $C=[c_{ij}]_{m\times r}$ โดยที่ $c_{ij}=a_{i1}b_{1j}+a_{i2}b_{2j}+...+a_{in}b_{nj}$

ตัวอย่าง เช่น

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11}b_{11}+a_{12}b_{21} & a_{11}b_{12}+a_{12}b_{22} \\ a_{21}b_{11}+a_{22}b_{21} & a_{21}b_{12}+a_{22}b_{22} \end{bmatrix}$$

การคูณเมตริกซ์ไม่มีคุณสมบัติในการสลับที่ คือ $A\times B \neq B\times A$ ดังนั้น ต้องระวังในการคูณเมตริกซ์ ตัวอย่าง การคูณค่าของจุดกับเมตริกซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$vA = v'$$

$$[x \quad y \quad z]\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = [x' \quad y' \quad z']$$

เมตริกซ์มีการนำไปใช้ในสาขาคอมพิวเตอร์กราฟิกอย่างมาก เนื่องจากการใช้เมตริกซ์ทำให้เราสามารถทำการย้ายตำแหน่งของจุด,หมุน หรือย่อขยายได้อย่างรวดเร็วด้วยการนำจุดมาคูณหรือบวกกับเมตริกซ์ เท่านั้น แต่การใช้เมตริกซ์ 3x3 จะทำให้ได้ค่าที่ไม่ถูกต้อง เนื่องจากความไม่เข้ากันของการคูณและ การบวกในเมตริกซ์ ดังนั้นเพื่อแก้ปัญหานี้จึงต้องเพื่อองค์ประกอบในเมตริกซ์ให้เป็นเมตริกซ์ขนาด 4x4 โดยองค์ประกอบที่เพิ่มขึ้นมา เรียกว่า homogenous coordinate ซึ่งแทนด้วยตัวอักษร w

ตัวอย่างการเพิ่ม homogenous coordinate ให้กับค่าจุดในระบบคู่อันดับคาร์ทีเซียน

$$[x \quad y \quad z] \Rightarrow [bx \quad by \quad bz \quad b]$$

โดย b จะเป็นค่าใดๆ ก็ได้ที่ไม่เท่ากับ 0 (ปกติจะใช้ b = 1)

<u>การย้ายตำแหน่งจุด</u> (translation) โดยให้ $[p_x \quad p_y \quad p_z]$ เป็นเวกเตอร์แทนระยะทางที่ต้องการย้ายจุดไป จะได้เมตริกซ์ สำหรับการย้าย จุด (translation matrix) ดังนี้

$$T(p) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ p_x & p_y & p_z & 1 \end{bmatrix}$$

นั่นคือ จะต้องนำจุดที่ต้องการจะย้ายตำแหน่งมาคูณกับเมตริกซ์สำหรับการย้ายจุด ดังนี้

$$[x \quad y \quad z \quad 1]\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ p_x & p_y & p_z & 1 \end{bmatrix} = [x + p_x \quad y + p_y \quad z + p_z \quad 1]$$

<u>การหมุนตำแหน่งจุดรอบแกน</u> (rotation) การหมุนจะต้องทำโดยอ้างอิงกันแกนหมุน ซึ่งก็คือ แกน x, แกน y และแกน z เมตริกซ์ที่ใช้ในการหมุน (rotation matrix) รอบแกน x

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$R_x(\theta)=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & \cos(\theta) & \sin(\theta) & 0\\ 0 & -\sin(\theta) & \cos(\theta) & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

เมตริกซ์ที่ใช้ในการหมุน (rotation matrix) รอบแกน y

$$R_y(\theta)=\begin{bmatrix}\cos(\theta) & 0 & -\sin(\theta) & 0\\ 0 & 1 & 0 & 0\\ \sin(\theta) & 0 & \cos(\theta) & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

เมตริกซ์ที่ใช้ในการหมุน (rotation matrix) รอบแกน z

$$R_z(\theta)=\begin{bmatrix}\cos(\theta) & \sin(\theta) & 0 & 0\\ -\sin(\theta) & \cos(\theta) & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

โดย ค่า θ คือ มุมที่ต้องการหมุนจากแกนนั้นๆ

<u>การย่อขยายกลุ่มของจุด (scale)</u>

$$S=\begin{bmatrix}S_x & 0 & 0 & 0\\ 0 & S_y & 0 & 0\\ 0 & 0 & S_z & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

นอกจากการใช้เมตริกซ์เพื่อการย้ายตำแหน่งของจุด, การหมุนตำแหน่งของจุดและการย่อขยายตำแหน่งของจุด แล้วยังสามารถใช้ข้อมูลในเมตริกซ์ในการหาค่าอื่นๆ ได้อีก ดังนี้

เวกเตอร์ p แสดงความสัมพันธ์ของวัตถุกับจุดกำเนิด

เวกเตอร์ n, o และ a แสดงทิศทางในแต่ละแกนหรือในรูปเวกเตอร์หนึ่งหน่วย i, j และ k ซึ่งค่าเวกเตอร์เหล่านี้สามารถหาได้จากเมตริกซ์ในตำแหน่งต่างๆ ดังนี้

$$\begin{bmatrix}n_x & n_y & n_z & 0\\ o_x & o_y & o_z & 0\\ a_x & a_y & a_z & 0\\ p_x & p_y & p_z & 1\end{bmatrix}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
เทคโนโลยีที่สนับสนุนการแสดงภาพ 3 มิติบนพ็อกเก็ตพีซี

ก่อนที่เราจะไปค้นหาเทคนิคหรือวิธีการแสดงภาพสามมิติบนพ็อคเก็ตพีซีได้นั้น เราจำเป็นต้องดูและเรียนรู้ก่อนว่า อุปกรณ์ที่เราใช้นี้ มีคุณสมบัติอันใดบ้าง ที่จะทำให้เกิดการแสดงผล ภาพสามมิติที่ดีได้

4.1 เทคโนโลยีด้านอุปกรณ์ที่ใช้

ในขั้นต้น เราได้เลือกใช้การทำงานบน พ็อกเก็ตพีซี ที่เป็นคอมพิวเตอร์พกพาขนาดเล็ก, โดยตัว พ็อกเก็ตพีซี นั้นมีสมรรถภาพในการแสดงผล และการคำนวณที่สูงเมื่อเทียบกับอุปกรณ์พกพาชนิดอื่นๆ เช่น โทรศัพท์มือถือ หรือ ปาล์ม (Palm) ซึ่งอุปกรณ์เหล่านี้มีสมรรถภาพที่ต่ำกว่า พ็อกเก็ตพีซี ในแง่ของการใช้งานด้านความบันเทิง

ตารางเปรียบเทียบคุณสมบัติของ อุปกรณ์พกพาต่างๆ (ข้อมูลเมื่อ มิถุนายน พ.ศ. 2546 – ผู้เขียน)

	Device	Processor	RAM	Screen	Sound
PPC	Casio E-125	150 MHz VR4122	32 MB	320x240 16-bit color	16-bit stereo, 44 kHz, MP3
	Compaq iPaq 3650	206 MHz Intel StrongARM	32 MB	320x240 12-bit color	16-bit stereo, 44 kHz, MP3
	Compaq iPaq 3100	206 MHz Intel StrongARM	32 MB	320x240 4-bit grayscale	16-bit stereo, 44 kHz, MP3
	HP Jornada 548	133 MHz Hitachi SH-3	32 MB	320x240 12-bit color	16-bit stereo, 44 kHz, MP3
	HP Jornada 545	133 MHz Hitachi SH-3	16 MB	320x240 12-bit color	16-bit stereo, 44 kHz, MP3
Palm	Palm IIIc	20 MHz Motorola Dragonball	8 MB	160x160 8-bit color	Monophonic tone generator (beep)
	Palm VIIx	20 MHz Motorola Dragonball	8 MB	160x160 2-bit grayscale	Monophonic tone generator (beep)
Mobile	Nokia 3330	n/a	4.5 MB	128x128 12-bit color	16-bit stereo, 44 kHz, MP3

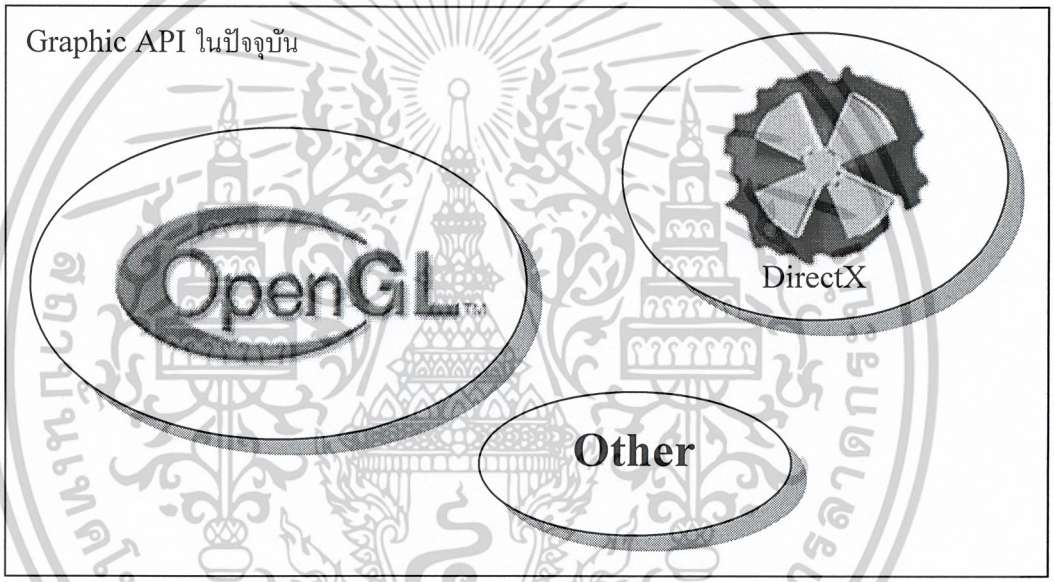
ตาราง 4-1 การเปรียบเทียบ พีดีเอ ชนิดต่าง ๆ

นอกจากนี้ ตัว พ็อกเก็ตพีซี นั้น จะใช้ระบบปฏิบัติการ Window CE ของค่ายไมโครซอฟท์ ซึ่งทางไมโครซอฟท์ ได้สนับสนุนนักพัฒนาโปรแกรมบน พ็อกเก็ตพีซี โดยแจกฟรี เครื่องมือ ที่ใช้ในการพัฒนาโปรแกรม นั่นคือ โปรแกรม Embedded Visual Tool และยังมีตัว ตัวจำลองพ็อกเก็ตพีซี (PocketPC Emulator)เพื่อให้นักพัฒนา สามารถใช้ในการทดสอบโปรแกรมที่เขียน ได้ง่ายยิ่งขึ้น โดยไม่จำเป็นต้องใช้เครื่องจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ไลบรารี กราฟิกสามมิติบนอุปกรณ์ขนาดเล็ก

ในงานการเขียนโปรแกรมทางด้านสามมิติทุกๆงาน จำเป็นต้องมีการเรียกใช้งาน ไลบรารี ท้างด้านสามมิติทั้งสิ้น หากจะกล่าวถึง ไลบรารี ที่เป็นที่รู้จัก และถือได้ว่าเป็นมาตรฐานอันหนึ่ง นั่นก็คือ OpenGL โดย OpenGL นับว่าเป็นไลบรารี ทางด้านสามมิติ ที่เป็นที่รู้จักกันอย่างกว้างขวางในวงการอุตสาหกรรมซอร์ฟแวร์ เพราะ OpenGL นี้ทั้งทำงานได้บนหลายๆแพลทฟอร์ม อีกทั้งเป็น โอเพ่นซอร์ส (Open Source) ที่อนุญาตให้ผู้ใช้สามารถนำไลบรารีไปใช้ได้ โดยไม่ต้องเสียค่าลิขสิทธิ์แต่อย่างใด และอีกตัวหนึ่งที่เป็นที่รู้จักกันดีในแวดวงเกมสามมิติ นั่นก็คือ Direct3D ของค่ายไมโครซอร์ฟ ซึ่งโดยตัวDirect3D นี้ ถึงแม้จะไม่เป็นมาตรฐานในวงการอุตสาหกรรม แต่ด้วยข้อดีที่เด่นชัดของไมโครซอร์ฟ คือ มีผู้ใช้ระบบปฏิบัติการ วินโดวส์ (Windows™) เป็นปริมาณมาก ทั่วโลก จึงทำให้ตัวไลบรารี Direct3D นั้นเป็นเหมือนกับ ข้อกำหนดพื้นฐานอันหนึ่ง ที่ผู้เขียนโปรแกรม “เกมส์” จำเป็นต้องใช้มัน ในโปรแกรมเกมส์ สามมิติของพวกเขา



รูปที่4-1 กราฟิกไลบรารี ที่มีใช้อยู่ในปัจจุบัน

ตัว OpenGL นั้นเป็น กราฟิกไลบรารี่ ที่สามารถใช้ได้หลายแพลทฟอร์มก็จริง แต่เนื่องจากมีขนาดที่ใหญ่มาก จึงไม่เหมาะกับการนำมาใช้ในอุปกรณ์ขนาดเล็กอย่างพ็อกเก็ตพีซี

ส่วนตัว Direct3D DirectX® ของค่ายไมโครซอร์ฟนั้น ถึงจะเป็นกราฟิกไลบรารี่ ที่นิยมใช้ในแพลทฟอร์มของ วินโดว์ส ที่มีอยู่มากมายนั้น แต่เนื่องจากมีขนาดใหญ่เช่นเดียวกันกับ OpenGL จึงทำให้ไม่เหมาะที่จะนำมาใช้บน PocketPC เช่นเดียวกัน

จะเห็นได้ว่า ทั้ง OpenGL และ DirectX นั้นเป็นกราฟิกไลบรารี ที่ถูกออกแบบมาเพื่อเครื่องพีซี ที่มีฮาร์ดแวร์ที่มีประสิทธิภาพของอุปกรณ์ สูงเพียงพอ, เนื่องจากสาเหตุข้างต้นนี้ เราจึงต้องหากราฟิกไลบรารีตัวอื่นๆ ที่มีขนาดเล็ก และออกแบบมาเพื่ออุปกรณ์ขนาดเล็กอย่างพ็อกเก็ตพีซี ดังนั้นเราจำเป็นที่จะต้องใช้ เอพีไอ (API) อื่นๆ ดังที่เราได้ค้นหา และ นำมาทดลองดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- GAPI (www.microsoft.com) : เป็น เอพีไอ ขนาดเล็กจากค่าย ไมโครซอฟท์ ที่เหมาะสำหรับการสร้างเกมส์บนอุปกรณ์ขนาดเล็ก, แต่เนื่องจากตัว GAPI นี้นั้น เป็น เอพีไอ 2 มิติ ที่ไม่มีฟังก์ชั่นการทำงานที่เกี่ยวกับการประมวลผลทางด้านสามมิติเลย เราจึงไม่สามารถนำมาใช้ในโครงงานของเราได้
- OpenGL|ES (www.khronos.org) : เป็น เอพีไอ (API) ที่เป็นตระกูล OpenGL ที่ออกแบบมาเพื่ออุปกรณ์ขนาดเล็กโดยเฉพาะ, โดยตัว OpenGL|ES นี้มีคำสั่งที่ครบถ้วนทั้งทางด้าน 2 มิติ , 3 มิติ ที่ใกล้เคียงกับตัว OpenGL เป็นอย่างมาก แต่ตัว OpenGL|ES นี้ ยังอยู่ในระหว่างการพัฒนา (เดือนมิถุนายน ปี พ.ศ.2546 - ผู้เขียน) เราจึงไม่สามารถนำมาใช้ในโครงงานของเราได้

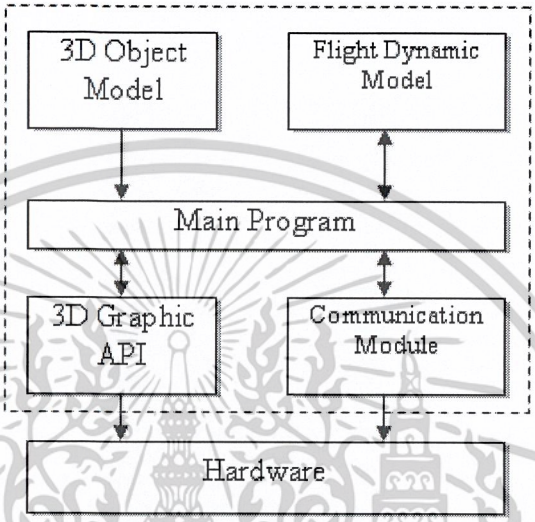
นอกจากของสองค่ายใหญ่นี้แล้ว เราจำเป็นที่จะต้องค้นหาและทดลองตัว เอพีไอ ที่เหมาะสมกับตัวงานของเรา โดยเราค้นหาจากค่ายเล็กๆ หรือจาก ผู้พัฒนาอิสระ ได้ดังต่อไปนี้

- TinyGL (fabrice.bellard.free.fr/TinyGL) : เป็น โอเพ่นซอส เอพีไอ ที่มาจากผู้พัฒนาอิสระ โดยมีโครงสร้างที่เลียนแบบมาจากตัว OpenGL ที่ตัดทอนคำสั่งบางคำสั่งที่ต้องใช้ทรัพยากรเครื่องสูงๆ ออกไป, แต่เนื่องจากเป็น เอพีไอที่มีพื้นฐานการเรียกใช้งานและการคอมไพล์อยู่บนสภาพแวดล้อมของ ลีนุกส์ (Linux) จึงทำให้เราต้องแก้ไขโค้ด เกือบทั้งหมด ซึ่งเป็นการเสียเวลาเป็นอย่างมาก เราจึงไม่เลือกใช้ตัว เอพีไอ นี้
- PocketGL (www.sundialsoft.freeserve.co.uk/pgl.htm) : เป็น เอพีไอของผู้พัฒนาอิสระที่มีชุดคำสั่งเหมือนกับ OpenGL, แต่ข้อเสียของตัว PocketGL นี้คือไม่สามารถทำงานร่วมกันกับ ตัวจำลอง พ็อกเก็ตพีซี ของ ไมโครซอฟท์ ได้ เราจึงไม่นำ เอพีไอ นี้มาใช้ในโครงงาน
- Diesel Engine (www.inmarsoftware.com) : ตัว เอพีไอตัวนี้ จะอิงโครงสร้างคำสั่งในลักษณะเดียวกันกับ DirectX แต่มีขนาดเล็กกว่า, สามารถทำงาน 2 มิติ และ 3 มิติ ได้อย่างครบถ้วนตลอดจนทั้งสามารถทำงานร่วมได้กับ ตัวจำลองพ็อกเก็ตพีซี ได้ *ดังนั้น เราจะใช้ เอพีไอ ตัวนี้ ในการทำโครงงาน*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
โครงสร้างตัวเกมส์

5.1 โครงสร้างของระบบ



รูปที่ 5-1 โครงสร้างตัวเกมส์

โครงสร้างของตัวเกมส์ประกอบไปด้วย

- ไลบรารี กราฟิกสามมิติ (3D Graphic API) : ในส่วนนี้เราจะใช้ ดีเซล เอนจิ้น ซึ่งเป็น เอพีไอ (API) ที่สร้างจากนักพัฒนาอิสระ
- โมเดลพลศาสตร์การบิน (Flight Dynamics Model) : ในส่วนนี้ เราจะสร้างรูปแบบการทำงานอย่างง่ายๆ ของผลกระทบจากสภาพแวดล้อมที่กระทบต่อตัวเครื่องบิน
- วัตถุสามมิติ (3D Object Model) : เป็นส่วนของออบเจคสามมิติ ต่างๆที่จะปรากฏอยู่ในเกม ตัวอย่างเช่น ออบเจคของเครื่องบิน เป็นต้น
- โมดูลติดต่อสื่อสาร (Communication Module) : เป็นส่วนที่ใช้ในการติดต่อสื่อสาร ระหว่างผู้เล่นสองคนเข้าด้วยกัน โดยผ่านทาง โพรโตคอล TCP/IP
- โปรแกรมหลัก (Main Program) : เป็นส่วนโปรแกรมหลักที่เป็นศูนย์กลางในการทำงานของตัวเกมส์ ซึ่งหน้าที่ของส่วนนี้คือ จะ เรียกใช้, ควบคุม, และจัดการ ส่วนต่างๆของตัวเกมส์ เพื่อให้การทำงานเป็นไปอย่างราบรื่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 รายละเอียดของการออกแบบในส่วนต่างๆ

5.2.1 โมเดลพลศาสตร์การบิน

เนื่องด้วยตัวเกมส์ในโครงงานนี้ เป็นการจำลองการบิน ดังนั้นจึงเป็นสิ่งจำเป็นที่จะต้องมีข้อมูลของตัวเครื่องบิน, สภาพแวดล้อมต่างๆ, ตลอดจนจำลองการทำงานของเครื่องบินภายใต้สภาพแวดล้อมนั้นๆ ซึ่งหน้าที่ของการทำงานในส่วนนี้ จะเกี่ยวข้องกับเรื่อง คณิตศาสตร์ ของตัวเครื่องบิน ซึ่งเราจำเป็นต้องใช้ หรือออกแบบออกแบบตัวโมเดลพลศาสตร์การ โดยเราจะใช้โมเดลการบินอย่างง่ายๆ ดังที่จะได้กล่าวในบทต่อไป

5.2.2 วัตถุสามมิติ

ตัวโมเดลต่างๆที่ปรากฏอยู่ในเกมนั้น เราสามารถสร้างได้ จากโปรแกรมทางด้านกราฟิกสามมิติทั่วไป ซึ่งในที่นี้ เราจะใช้โปรแกรม 3D Studio MAX ™ ในการสร้างโมเดลทุกๆโมเดล ในตัวเกมส์ เพราะว่า โปรแกรมนี้ เป็นโปรแกรมที่นักพัฒนาใช้กันมานานในวงการโมเดลสามมิติ จึงทำให้เรามีโมเดลวัตถุดิบมากมาย

5.2.3 โมดูลติดต่อสื่อสาร

หน้าที่ของส่วนนี้คือเป็นตัวติดต่อกับเครื่องพ็อกเก็ตพีซีอีกเครื่องหนึ่ง เพื่อเชื่อมต่อกันแบบ เครื่องต่อเครื่อง (Peer-to-Peer) จึงทำให้สามารถเล่นพร้อมกันได้สองเครื่อง ซึ่งจะมีตัวหนึ่งแสดงบทบาทของเซิฟเวอร์ และอีกตัวหนึ่งจะเป็นไคลเอนต์ ดังนั้นในโครงงานของเรา จำเป็นที่จะต้องสร้างโมดูลของการเชื่อมต่อสื่อสารไว้ด้วย

รูปที่ 5-2 รูปแบบการเชื่อมต่อเน็ทเวิร์ค

5.2.4 ไลบรารี กราฟิกสามมิติ

ในส่วนนี้ จะมีหน้าที่ในการแสดงผลภาพสามมิติของวัตถุต่างๆ ในเกมส์ ตลอดจนทำการควบคุมจัดการข้อมูลในเชิงสามมิติ ซึ่งการทำงานของไลบรารี นี้ ถือได้ว่าเป็นหัวใจสำคัญของเกมส์เลยทีเดียว ดังนั้น เนื่องจากความรู้ในเชิงคณิตศาสตร์สามมิตินั้น เป็นเพียงแนวคิดในการจัดการและควบคุมข้อมูล แต่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

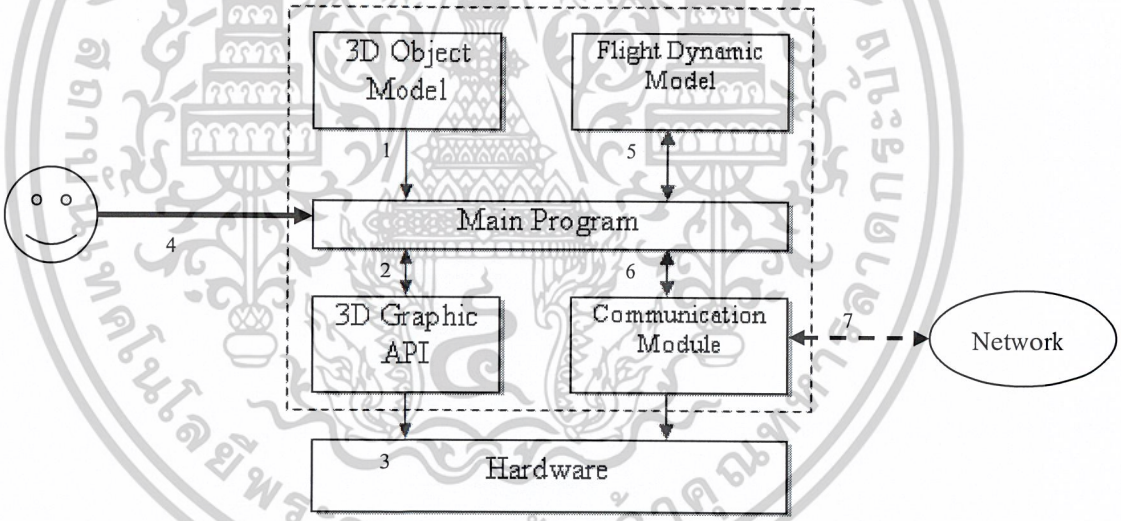
ถ้าหากจะนำมาใช้จริงนั้น จำเป็นที่จะต้องทำความเข้าใจการทำงานของไลบรารีอย่างดีด้วย จึงจะสามารถระลึกได้ว่าการควบคุมวัตถุสามมิติด้วยวิธีหนึ่งๆ จะต้องใช้คำสั่งใด ของตัว ไลบรารี

โดยในส่วนนี้ เราจะใช้ ดีเซล เอนจิ้น ซึ่งเป็นไลบรารีที่สามารถรองรับการทำงานได้อย่างครบถ้วน ไม่ว่าจะเป็นการจัดการทางด้านการควบคุมเชิงสามมิติ, การวาดรูปเพื่อแสดงผลออกหน้าจอ, ตลอดจนทั้งมีคลาสต่างๆให้ได้เลือกใช้เพื่ออำนวยความสะดวกอีกมากมาย

5.2.5 โปรแกรมหลัก

ในส่วนของโปรแกรมหลักของเรานี้ เราจะเขียนโดยใช้ภาษา C++ ผ่านทาง Embedded visual C++ ของ บริษัท ไมโครซอฟท์ ซึ่งการเขียนโปรแกรมบน พ็อกเก็ตพีซี นี้จะแตกต่างจากการเขียนโปรแกรมบน พีซี (PC)เล็กน้อย ซึ่งพื้นฐานคือเขียนโค้ดให้มีขนาดที่สั้นที่สุด และหลีกเลี่ยงการใช้ ฟังก์ชั่นที่ต้องเสียเวลาคำนวณสูง

5.3 ความสัมพันธ์ของส่วนต่างๆ



รูปที่ 5-3ความสัมพันธ์ของการทำงาน

การทำงานของเกมส์มีลักษณะการทำงานดังนี้จะ

1. ในขั้นต้นเมื่อเกมส์เริ่มทำงานตัวเกมส์จะทำการโหลดรูปภาพและข้อมูลวัตถุสามมิติเข้าไปในหน่วยความจำ และ ตัวโปรแกรมจะทำการสร้างวัตถุสามมิติเองบางส่วน จากตัวข้อมูลที่รับเข้ามา
2. การทำงานที่โดยโปรแกรมหลัก จะควบคุม และ จัดการข้อมูลเชิงสามมิติได้ดีนั้น จำเป็นที่จะต้องใช้งาน ตัว กราฟิก ไลบรารี ในการทำงานในส่วนนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. หลังจากที่การทำงานด้านสามมิติขั้นต้น ทำงานได้อย่างราบรื่นแล้ว ต่อไปก็คือจะต้องนำข้อมูลที่ประมวลผลเสร็จเรียบร้อยแล้วนั้น ไปแสดงผลออกหน้าจอ เพื่อติดต่อกับผู้ใช้
4. จากนั้น ตัวเกมส์ จำเป็นที่จะต้องรับข้อมูลจากผู้ใช้ เข้ามาสร้างผลกระทบ ให้เกิดขึ้นต่อวัตถุสามมิติต่างๆ เช่นการขับเคลื่อนเครื่องบิน เป็นต้น โดยส่วนนี้ ตัวโปรแกรมหลักของเราจะทำหน้าที่สร้างผลกระทบจากการควบคุมเหล่านั้น ให้เกิดกับตัววัตถุสามมิติเอง
5. ในการที่ตัวโปรแกรมจะสามารถสร้างค่าที่เหมาะสมในการจำลองการทำงานของเครื่องบินได้นั้น ตัวโปรแกรมหลัก จะต้องติดต่อกับ โมเดลพลศาสตร์การบิน เพื่อส่งค่าเข้าไปในโมเดล และคอยรับค่าผลกระทบต่างๆที่จะเกิดขึ้น ต่อเครื่องบิน เพื่อนำไปประมวลผลอีกทีหนึ่ง
6. ในกรณีที่มีการเล่นกันสองคน ตัวโปรแกรมหลัก จำเป็นที่จะต้องใช้การเชื่อมต่อสื่อสารกันระหว่างเครื่องสองเครื่อง ผ่านทางโมดูลการติดต่อสื่อสาร เพื่อทำการเชื่อมต่อการเล่นระหว่างเครื่องสองเครื่อง ให้เล่นด้วยกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
คณิตศาสตร์ของการจำลองการบินของเกมส์

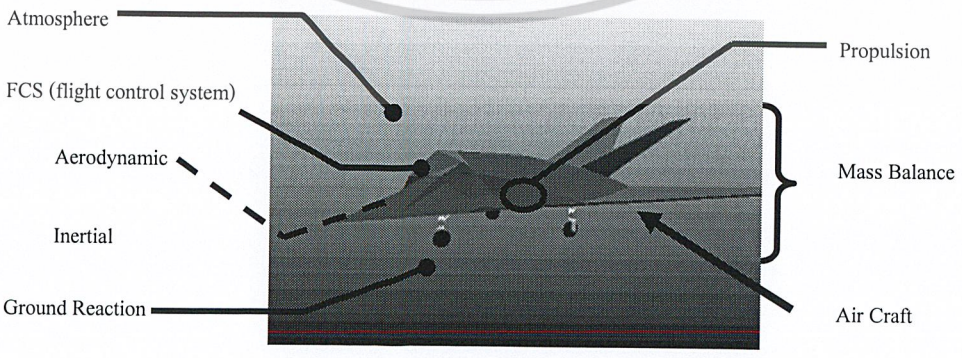
เนื่องเพราะตัวเกมในโครงงานนี้ เป็นเกมส์ที่เกี่ยวพันกับกันตัวเครื่องบิน ดังนั้น จึงจำเป็นที่ต้องเข้าใจหลักการบินพื้นฐาน ตลอดจนคุณลักษณะของตัวเครื่องบินในบางส่วนอีกด้วย ซึ่งในความเป็นจริงนั้น เราสามารถนำซอร์สโค้ด โมเดลพลศาสตร์การบิน (Flight Dynamic Model) มาประยุกต์ใช้ได้ ซึ่งตัวโมเดลพลศาสตร์การบินนี้ จะเป็นการจำลองและสร้างค่าพารามิเตอร์ทางการบินต่างๆ เช่น ความต้านอากาศ, อัตราเร่งของเครื่องบิน, สมดุลของเครื่องบิน ตลอดจนค่า ผลกระทบที่เกิดจากการบังคับต่างๆของผู้ใช้ เช่น ผลกระทบจากการบินขึ้น-ลง, การเร่งเครื่อง ฯลฯ โดยตัวโมเดลพลศาสตร์การบินที่เป็นที่รู้จัก เช่นโมเดลของ “JSB Sim” ที่เป็นฟรีโมเดล ที่ถูกใช้กันอย่างแพร่หลาย เป็นต้น

แต่ในโครงงานของเรานี้ จะใช้หลักการของคณิตศาสตร์การบินเพียงบางส่วนนำมาใช้ในตัวเกม เนื่องจากเพราะ ตัวโมเดลพลศาสตร์การบิน นั้นมีความซับซ้อนสูง จึงต้องมีความรู้ทางด้านวิศวกรรมการบินพอสมควร นอกจากนี้ตัวโมเดลพลศาสตร์การบิน มีขนาดใหญ่, มีฟังก์ชั่นการใช้งาน เกินกว่าความจำเป็นในโครงงานของเรา อีกทั้งตัวเกมของเรา เป็นเกมแนวต่อสู้ทางอากาศ ซึ่งโมเดลจำลองการบิน ไม่รองรับการจำลองการทำงานในบางส่วนได้ (เช่นผลกระทบของการยิงขีปนาวุธ) ดังนั้น เราจึงทำการศึกษาคณิตศาสตร์การบินบางส่วน เพื่อนำมาประยุกต์ใช้ในตัวเกมส์

6.1 JSB Sim

การทำงานของ JSB Sim คือ มีการสร้างสภาวะแวดล้อม, สร้างค่าต่างๆที่มีผลกระทบต่อตัวเครื่องบิน โดยขั้นต้น จะต้องมีข้อมูลทางกายภาพของเครื่องบินก่อน ซึ่งใน JSB Sim นี้ จะมีมาให้เป็นตัวอย่าง ประมาณ 10ลำ เช่น เครื่องบิน F-16, F-22 เป็นต้น จากนั้นก็ทำการรันโปรแกรม แล้ว JSB Sim ก็จำสร้างค่าต่างๆ ที่เกิดผลกระทบต่อเครื่องบินนั้นๆมาให้ ตลอดจนรับค่าจากการบังคับเครื่องบินเพื่อนำไปสร้างผลกระทบที่เกิดจากการบังคับเครื่อง อีกด้วย

โดยค่าที่ JSB Sim สร้างมาให้นี้ ยกตัวอย่างดังรูปที่ 6-1



รูปที่ 6-1 ค่าที่ JSB Sim สามารถจำลองได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าต่างๆที่จำลองมาได้ มีรายละเอียดดังนี้

- Atmosphere เป็นการจำลองสภาวะแวดล้อม ว่ามีค่าความกดอากาศเท่าไร, มีความชื้นในอากาศเป็นปริมาณเท่าไร
- FCS (Flight Control System) เป็นการจำลองหน้าปัดควบคุมต่างๆ ตลอดจนทั้งการควบคุมเครื่องที่เกิดจากมนุษย์
- Inertial ค่าความเฉื่อยของเครื่องบิน โดยค่านี้ ส่งผลให้เครื่องบินมีความหน่วงในการเคลื่อนที่และมีความหน่วงในการหยุดเคลื่อนที่
- Ground reaction เป็นผลกระทบที่เกิดจากพื้นดิน ต่อเครื่องบิน หรือเป็นแรงกด ระหว่างตัวเครื่องบินกับพื้นดิน นั่นเอง
- Propulsion เป็นค่าการขับเคลื่อนของเครื่องยนต์ ว่ามีความเร็วเท่าไร อัตราการเผาผลาญเชื้อเพลิงเป็นเท่าไร
- Mass Balance เป็นสมดุลของเครื่องบิน ในการเคลื่อนที่ในอากาศ
- Aircraft เป็นค่าพารามิเตอร์อื่นๆที่เกี่ยวกับเครื่องบิน ตลอดจนค่าพารามิเตอร์เฉพาะตัว ของเครื่องบินรุ่นต่างๆ

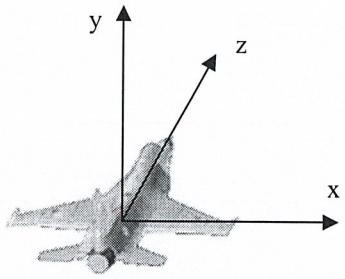
6.2 ความเข้าใจในพิกัดของเครื่องบิน เมื่อเทียบกับพิกัดสามมิติ

ถ้าพูดถึงพิกัดสามมิติ มันจะประกอบไปด้วย แกน x, y, z สามแกน และเราจำเป็นต้องเข้าใจถึงการอ้างอิงพิกัดของโลก (World Coordinate) เทียบกับพิกัดของเครื่องบินหรือวัตถุ (Local Coordinate) ซึ่งโดยปกติแล้ว เรามักจะพูดถึงคำว่า "แกน X, Y, Z" ในพิกัดของโลก แต่เมื่อถึงพูดถึงแกนที่ตัวเครื่องบินแล้ว เราจะอธิบายได้ยากกว่า เช่น เราต้องการหันหัวเครื่องบินไปในทิศทางซ้ายมือ-ขวามือ เราจะต้องกล่าวว่า "ให้หมุนเครื่องบินโดยใช้แกน y เป็นแกนศูนย์กลาง" ซึ่งนับว่าเข้าใจยากเอาการอยู่ (ดูรูปที่ 6-2 ประกอบความเข้าใจ)

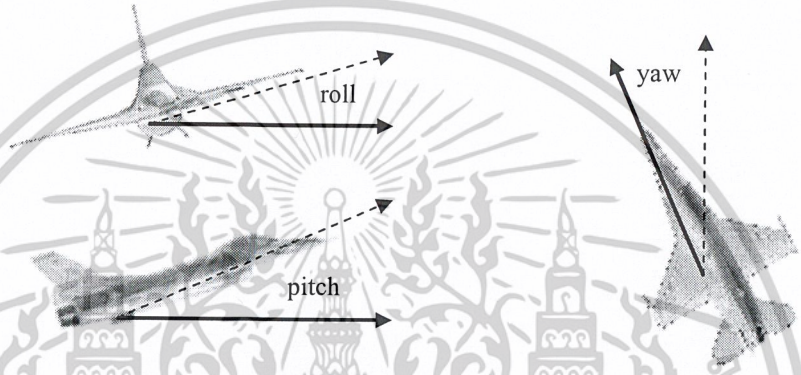
ดังนั้น ในมุมมองทางเครื่องบิน จะใช้ศัพท์ ดังต่อไปนี้ แทนการพูดถึงแกน X, Y, และ Z

- เมื่อต้องการหมุนเครื่องบินโดยใช้แกน y ของตัวเครื่องบิน เป็นแกนศูนย์กลางการหมุน เราเรียกว่า เป็นการหมุนในแนวหันเห (Yaw)
- เมื่อต้องการหมุนเครื่องบินโดยใช้แกน x ของตัวเครื่องบิน เป็นแกนศูนย์กลางการหมุน เราเรียกว่า เป็นการหมุนในแนวก้ม-เงย (Pitch)
- เมื่อต้องการหมุนเครื่องบินโดยใช้แกน z ของตัวเครื่องบิน เป็นแกนศูนย์กลางการหมุน เราเรียกว่า เป็นการหมุนในแนวควง (Roll)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



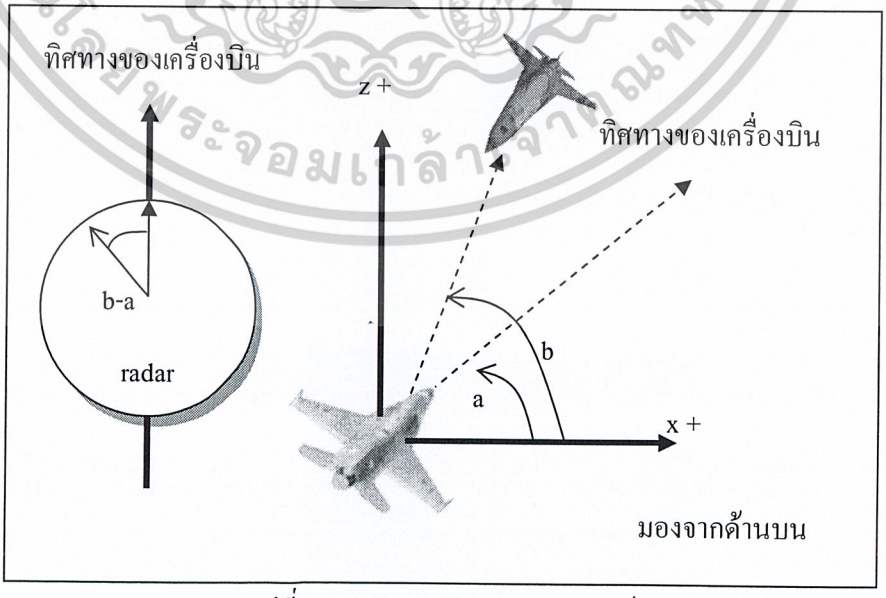
รูปที่ 6-2 แสดงแกนต่างๆ ของเครื่องบิน



รูปที่ 6-3 แสดงลักษณะชื่อเรียกแทนการหมุนต่างๆ ของเครื่องบิน

6.3 หลักการในการสร้างเรดาห์ตรวจจับศัตรู

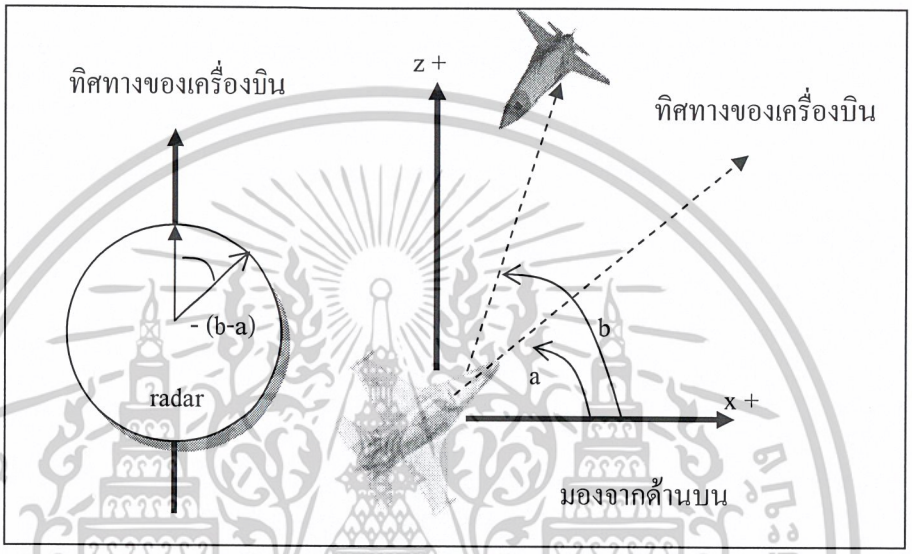
ตัวเรดาห์มีประโยชน์ในการหาตำแหน่งพิกัดของศัตรู ซึ่งในการที่จะแสดงผล ออกทางหน้าปัดให้ได้เที่ยงตรงและถูกต้องนั้น จะมีการทำงานในลักษณะดังนี้



รูปที่ 6-4 แสดงคุณลักษณะของเรดาห์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่6-4 ให้เราสมมุติว่าเครื่องบินนั้น บินได้ในระนาบเพียงสองมิติ (ระนาบ Z-X) เราต้องการหาว่า เมื่อมองจากเรดาห์ แล้ว ศัตรูนั้นจะอยู่ในทิศทาง องศาหันเห เท่าไรเมื่อเทียบกับทิศทางการบิน? จากรูปนั้น เราให้ มุม a คือ มุมของทิศทางการเคลื่อนที่เมื่อเทียบกับแกน x และให้มุม b คือ มุมของทิศทางของศัตรู เมื่อเทียบกับแกน x เราจะได้มุมของศัตรู มีค่าเท่ากับ b-a ตัวอย่างเช่น เราเคลื่อนที่ในมุม 35 องศา และศัตรูอยู่ ณ.ตำแหน่ง มุม 60 องศา เราจะได้มุมปรากฏที่หน้าปัดเรดาห์ คือ 60 - 35 = 25 องศา(เมื่อเทียบกับทิศทางการเคลื่อนที่ของเรา) หรือศัตรู อยู่ในทิศ 11 นาฬิกา เป็นต้น



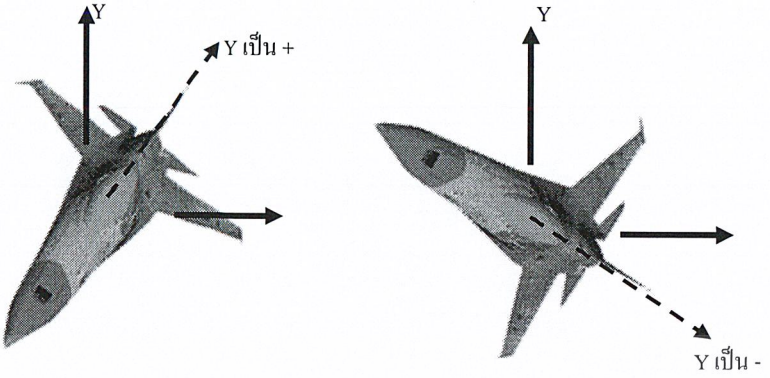
รูปที่ 6-5 แสดงคุณลักษณะของเรดาห์เมื่อเครื่องบิน บินกลับหัว

แต่ในกรณีที่อยู่ในพิกัดสามมิติ เรามีสิ่งที่ต้องทำมากกว่านั้น นั่นคือ ตรวจสอบว่าเครื่องบินของเรา กำลังบินกลับหัวอยู่หรือไม่? เพราะการบินกลับหัวจะทำให้การการคำนวณต้องซับซ้อนขึ้นเล็กน้อย ซึ่งถ้าดูตามรูปที่ 6-5 นั้น เครื่องบินที่เห็นนั้น กำลังบินกลับหัว ซึ่งให้เราสมมุติว่าเรานั่งอยู่ในเครื่องบินแล้วหาทิศทางของศัตรู จะปรากฏว่าศัตรู อยู่ในทิศ 1 นาฬิกา หรือ อยู่ทางขวามือ นั่นเอง

โดยในกรณีที่เครื่องบิน มีการบินกลับหัว เราจำเป็นที่จะต้องใช้สูตรในการคำนวนต่างกันออกไป นั่นคือจะใช้สูตร – (b- a) แทน ตัวอย่างเช่น เครื่องบินเรา บินกลับหัวอยู่ ไปในทิศทาง 25 องศา, และศัตรูอยู่ในตำแหน่ง 70 องศา(เมื่อเทียบกับแกน x) เราจะได้ค่าบนหน้าปัดเรดาห์ มีค่าเท่ากับ – (70 – 25) ซึ่งมีค่าเท่ากับ -45 องศา(เมื่อเทียบกับทิศทางการเคลื่อนที่ของเรา) หรือ อยู่ในทิศ 10 นาฬิกา นั่นเอง

ทั้งนี้ทั้งนั้น การจะตรวจสอบว่าเครื่องบินเราบินกลับหัวอยู่หรือไม่ ให้ตรวจสอบทิศทางของแนวชี้ขึ้นของเครื่องบิน ว่ามันมีค่าเป็นลบ หรือเป็นบวก เมื่อเทียบกับ แกน y ในพิกัดสามมิติ ดังรูปที่6-6

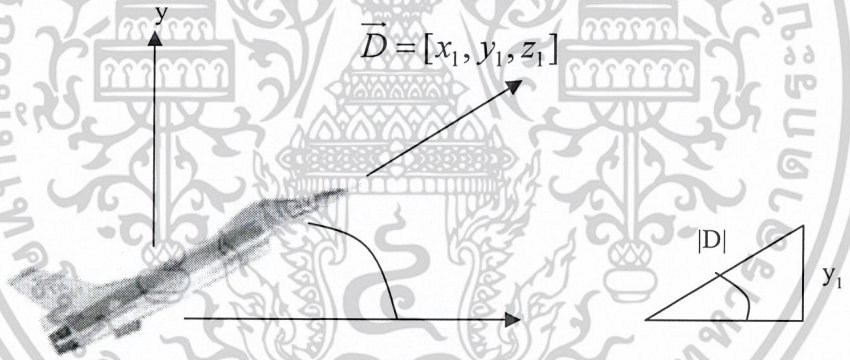
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6-6 แนวชี้ขึ้นของเครื่อง แบบกลับหัว และ ไม่กลับหัว

6.4 หลักการตรวจสอบมุมก้มเงย (pitch) ของเครื่องบิน

ในการตรวจหาทิศทางของเวคเตอร์ใดๆเมื่อเทียบกับแกนใดๆ ในพิกัดสามมิตินั้น เราสามารถทำได้จากการคำนวณทั้งสิ้น ซึ่งในกรณีของการตรวจสอบมุมก้ม-เงยของเครื่องบิน เราก็ใช้การตรวจสอบเวคเตอร์ทิศทางของเครื่องบิน เทียบกับเวคเตอร์แกน y ว่า มีค่าเป็นเท่าไร ดังรูปที่6-7



รูปที่ 6-7 แสดงการก้ม-เงย ของเครื่องบิน

เราสามารถหามุมก้ม-เงยได้จากสูตร

$$pitch\ angle = \arcsin(y_1 / |\vec{D}|)$$; $|\vec{D}|$ คือขนาดความยาวของเวคเตอร์ D

y_1 คือค่าสเกลล่า y ของเวคเตอร์ D

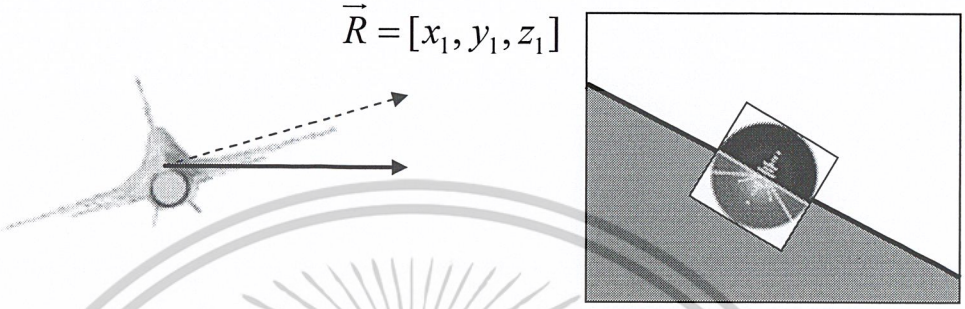
6.5 หลักการในการสร้างตัวตรวจจับระดับพื้นดิน (Attitude Director Indicator , ADI)

ตัวตรวจจับนี้ มีหน้าที่สำคัญสองประการ ประการแรกมีไว้ใช้ตรวจดูความเอียงของเครื่องบินในลักษณะการควง เมื่อเทียบกับระดับน้ำทะเล และอีกประการคือ ไว้ใช้ดูระดับความเอียงของเครื่องบินในแนวก้มเงย โดยในหน้าที่หลังนี้ จะมีลักษณะซ้ำซ้อนกับหัวข้อที่ผ่านมา (หัวข้อ 6.3) โดยแตกต่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรงที่ ตัวตรวจจับระดับพื้นดินนี้ แสดงผลออกมาในเชิงรูปภาพ ต่างกับ ตัวตรวจจับการก้ม-เงยในข้อก่อนที่แสดงผลเป็นตัวเลข

แต่อย่างไรก็ตาม ในเกมส์ของเรา จะใช้แค่คุณสมบัติประการแรกเพียงอย่างเดียว นั่นคือการตรวจจับความเอียงในการควง ของเครื่องบินเรา เมื่อเทียบกับระดับน้ำทะเล



รูปที่ 6-8 แสดงการควงเครื่องบิน และมุมมองหน้าปัด ADI ภายในเครื่องบิน

จากรูปที่6-8 จะเห็นได้ว่าเมื่อเราควงเครื่องบินไปทางซ้าย หน้าปัด ADI จะเอียงมาทางด้านขวาด้วยองศาที่เท่ากันกับองศาการควง

เราสามารถหามุมการควงได้จากสูตร

$roll\ angle = \arcsin(y_1 / |\vec{R}|)$; โดย $|\vec{R}|$ คือขนาดของเวคเตอร์ทิศทางขวามือของเครื่องบิน

; y_1 คือ ค่าสเกลล่า y ของเวคเตอร์ R

6.6 ผลกระทบจากสภาวะแวดล้อมภายนอกต่อเครื่องบิน

การจำลองสภาพแวดล้อมภายนอก ที่มีผลกระทบต่อเครื่องบินนั้น ในความเป็นจริงมีอยู่มากมาย ตั้งแต่แรงลม, แรงเสียดทานอากาศ, อัตราเร่ง, น้ำหนักตัวเครื่อง, สมดุลเครื่องบิน แต่ในเกมส์ของเรานี้สร้างสภาวะแวดล้อมจำลองอย่างง่ายๆ แค่ไม่กี่อย่าง ซึ่งก็เพียงพอแล้วในการเล่นเกมส์ บนเครื่องเล่นขนาดเล็ก เพราะว่า การคำนวณที่มากเกินไป ย่อมไม่มีประโยชน์ หากไม่สามารถรู้สึกได้ถึงผลกระทบนั้นๆ

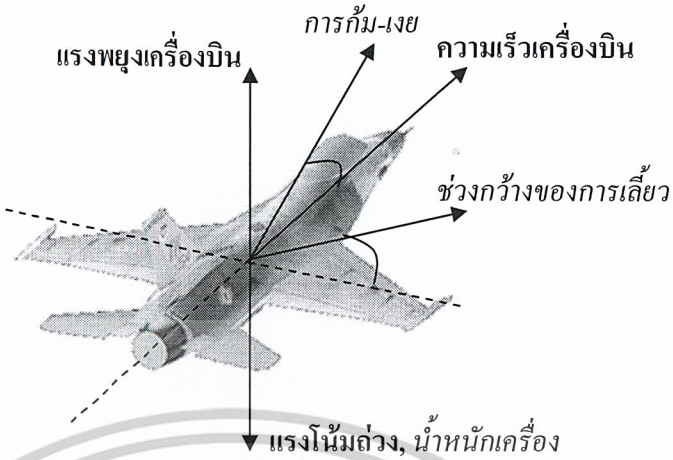
โดยผลกระทบต่างๆที่มีต่อเครื่องบินโดยตรง ในเกมส์ของเรา จะมีดังนี้

- ความเร็วเครื่องบิน
- แรงโน้มถ่วงโลก
- แรงพยุงเครื่องบิน

นอกจากนี้ ยังมีผลกระทบบางอย่างที่ไม่ได้ส่งผลต่อเครื่องบินโดยตรง แต่จะส่งผลถึงผู้เล่นนั่นคือ

- น้ำหนักเครื่อง
- การก้ม-เงย
- ช่วงกว้างของการเลี้ยว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6-9 ผลกระทบที่เกิดต่อเครื่องบิน

อันดับแรก แรงโน้มถ่วง นั้น จะเป็นเวคเตอร์ที่ชี้ลงล่างเสมอและมีค่าคงที่เสมอ ซึ่งผลกระทบของแรงโน้มถ่วง จะทำให้เครื่องบิน ค่อยๆลดเพดานบินลงมา

ต่อมาคือแรงพยุงเครื่องบิน แรงพยุงเครื่องบินนี้ เป็นแรงที่เอาชนะแรงโน้มถ่วง และทำให้เครื่องบินลอยขึ้นมาได้ ซึ่งแรงพยุงเครื่องบินนี้ เป็นเวคเตอร์ที่มีทิศทางชี้ขึ้นเสมอ และจะมีค่าต่างกันออกไปตามความเร็วของเครื่องบินที่เป็นอยู่ในปัจจุบัน กล่าวคือ ถ้ายิ่งบินเร็วมากขึ้นเท่าไร แรงพยุงเครื่องบิน ก็จะยิ่งมีมากขึ้นเป็นเงาตามตัว

อันที่สาม ที่มีผลกระทบอย่างมากกับเครื่องบิน นั่นคือ ความเร็วของเครื่องบิน โดยความเร็วของเครื่องบินนี้ จะเกิดจากตัวแปรที่สำคัญหลายตัว เช่น อัตราการเร่งเครื่องยนต์, น้ำหนักตัวเครื่องบิน ซึ่งความเร็วนี้ จะมากขึ้น หากอัตราเร่งเครื่องยนต์มีสูงขึ้น, หรือ ความเร็วของเครื่องบินจะมากขึ้น หากน้ำหนักเครื่องบินลด โดยน้ำหนักจะลดได้ เกิดจากปัจจัยสองประการ ประการแรกคือ ปริมาณขีปนาวุธที่มีอยู่ หากมีมาก จะทำให้หนักมากขึ้น และอีกประการ คือ ปริมาณเชื้อเพลิงที่เหลืออยู่ โดยหากมีเชื้อเพลิงเหลืออยู่น้อย จะทำให้บินเร็วขึ้น เป็นต้น

ส่วนผลกระทบข้างเคียงที่ส่งผลให้การบังคับเครื่องบิน นั้น ง่ายหรือ ยาก ก็คือ ช่วงกว้างของการเลี้ยว และ การก้ม-เงย ซึ่งค่าของทั้งสองตัวนี้ จะมาก จะน้อย ขึ้นอยู่กับความเร็วของเครื่องบิน กล่าวคือ หากมีความเร็วสูง ก็จะมีความคล่องตัวสูง มากขึ้น เป็นเงาตามตัว และ หากเครื่องบิน มีความเร็วต่ำ ก็จะบังคับได้ยากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7
ดีเซล เอนจิ้น (Diesel Engine)

ดีเซล เอนจิ้น นี้เป็นชื่อของ ไลบรารี่ (Library) ตัวหนึ่งในการจัดการด้านภาพสามมิติ ที่เราได้นำมาใช้ในโครงงานของเรา ซึ่งคุณสมบัติที่ดีของ ดีเซลเอนจิ้น ตัวนี้คือ ทำงานบนอุปกรณ์ขนาดเล็กได้ ไม่ว่าจะเป็นอุปกรณ์ พีดีเอ หรือว่า โทรศัพท์มือถือ

7.1 โครงสร้างพื้นฐานของตัวดีเซล ไลบรารี

โครงสร้างโดยทั่วไปของ ดีเซล ไลบรารี ในการจัดการด้านภาพสามมิตินั้น จะมีลักษณะคล้ายกันกับ Direct3D ในหลายๆด้าน เช่น ใน ไลบรารี จะใช้ระบบพิกัด มือซ้าย เหมือนกับ Direct3D นั่นคือแกน X อยู่ด้านขวามือ, แกน Y ชี้ขึ้นไปข้างบน, และแกน Z จะชี้ออกไปด้านหน้า และนอกจากนี้ ยังมีโครงสร้างการใช้งานในแบบออบเจค เหมือนกัน

ภายในตัว ดีเซล ไลบรารี มีคำสั่งต่างๆมากมาย ซึ่งไม่สามารถกล่าวในที่นี้ได้ทั้งหมด ดังนั้น เราจะแสดงเพียงโครงสร้างเพียงบางส่วนที่นำมาใช้ในโครงงานของเรา โครงสร้างของ ดีเซล ไลบรารี โดยรวมมีดังนี้

- DieselCommon: เป็นกลุ่มของคลาสที่ใช้สร้างออบเจคพื้นฐาน โดยที่ทุกๆคลาสจะจำเป็นต้องเรียกใช้มัน และยังมีคลาสย่อยภายในอีกมากมายที่เป็นโครงสร้างการเก็บข้อมูลของการเขียนโปรแกรมเชิงสามมิติ เช่น เก็บคลาสที่ไว้ใช้สร้างอ๊อบเจคที่เป็นเมตริกซ์ ชนาด 4 x 4 , คลาสของเวคเตอร์สองมิติ-สามมิติ เป็นต้น
- DieselGraphic: เป็นกลุ่มของคลาสที่ใช้ในการจัดการด้านรูป ในงานสองมิติ เช่น คลาสในการเก็บข้อมูลพื้นผิว (Texture) และนอกจากนี้ ยังเป็นคลาสหลักในการเริ่มต้นทำงานต้นการทำงานของโปรแกรม
- Diesel3D: เป็นกลุ่มของคลาสที่ใช้ในการจัดการด้านสามมิติทุกด้าน เช่นคลาสของกล้อง (Camera), คลาสของการสร้างแสง, คลาสของการสร้างคุณสมบัติของวัตถุ (Material), คลาสของการสร้างวัตถุสามมิติ เป็นต้น
- DieselInput: เป็นกลุ่มของคลาสที่ใช้ในการจัดการเกี่ยวกับอุปกรณ์ อินพุต ทั้งหลาย ไม่ว่าจะเป็นเมาส์, คีย์บอร์ด, แป้นหน้าจอมือถือ เป็นต้น โดยคลาสนี้ เราไม่ได้ใช้ในโครงงานของเราเพราะว่าในงานของเราสามารถใช้ความสามารถของตัวภาษา ควบคุมอุปกรณ์อินพุตแทนได้
- DieselSound: เป็นกลุ่มของคลาส ที่ใช้ในการสร้างเสียงให้แก่โปรแกรม ซึ่งโครงงานของเราก็ไม่ได้ใช้งานคลาสนี้เช่นเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2 รายละเอียดของคลาส และ เม็ดธอด ต่างๆที่จำเป็นในการเขียนโปรแกรม

จากที่กล่าวไว้ในหัวข้อ 7.1 นั้น เราจะอธิบายถึงคลาสที่สำคัญๆ ในการทำโครงงานของเรา

กลุ่มของ DieselCommon

ชื่อคลาส: CDieselVector2

หน้าที่: เป็นคลาสที่ใช้ในการสร้างเวคเตอร์ออบเจค โดยจะเป็นเวคเตอร์ในพิกัดสองมิติเท่านั้น

ดาต้าเมมเบอร์: FLOAT x, y

เมมเบอร์: Add ,Sub ,Mul ,Scale,Div,Dot ,LengthSq ,Length ,GetAngle ,Neg ,Set,Normalize ,IsEqual

ชื่อคลาส: CDieselVector3

หน้าที่: เป็นคลาสที่ใช้ในการสร้างเวคเตอร์ออบเจค โดยจะเป็นเวคเตอร์ในพิกัดสามมิติเท่านั้น

ดาต้าเมมเบอร์: FLOAT x, y, z

เมมเบอร์: Add, Sub, Mul ,Scale,Div ,Dot ,LengthSq ,Length, GetAngle ,Neg ,SetNormalize , IsEqual ,Setlength ,CreateBetween ,Vectors ,Lerp ,operator = ,operator -

ชื่อคลาส: CDieselMatrix4

หน้าที่: เป็นคลาสที่ใช้ในการสร้างเมตริกซ์ ออบเจค โดยจะเป็นเมตริกซ์ 4 x 4 ที่ใช้ในระบบ Homogenous

ดาต้าเมมเบอร์: FLOAT m00 - m33

เมมเบอร์:CDieselMatrix4 ,Identity ,Set ,RotateX ,RotateY ,RotateZ ,RotateXYZ ,RotationAxis ,Translate,Scale ,MakePerspective ,MakeLookAt ,Add ,Sub ,Mul ,Div ,Inverse ,Transpose ,IsIdentity ,GetRight ,GetUp ,GetDir ,GetPos ,SetRight ,SetUp ,SetDir ,SetPos ,operator = ,operator* ,IsEqual ,operator == ,operator -

กลุ่มของ CDieselCommon

ชื่อคลาส: CDieselSurface

หน้าที่: เป็นคลาสที่เก็บพื้นผิวไว้สำหรับไว้ใช้ แปะบนวัตถุต่าง ๆ

เมมเบอร์: Create , Release ,Load ,LoadFromPack ,LoadFromResource ,LoadFromMemory , Save ,Lock ,Unlock ,GetDC ,ReleaseDC ,Blt ,BltFast ,BltRotate ,BltEffect ,GradientFill ,Fill , Fade ,FadeRGB ,AlphaBlend ,AlphaBlendRGB ,Negate ,SetPixel ,GetPixel ,SetPixelDIB ,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GetPixelDIB ,SetPieceDIB ,SetSubFrames , GetSubFrame ,GetSubFrameWidth , GetSubFrameHeight ,GetSubFrameCount ,SetColorKey ,SetAutoColorKey ,GetColorKey , GetSurfaceDesc , GetFormat ,GetWidth ,GetHeight , SetClipRect ,GetClipRect , MakeColor ,Restore ,IsLost ,GetDirectDrawSurface

ชื่อคลาส: IDieselApplication

หน้าที่: เป็นคลาสหลักที่ใช้กับ แอพพริเคชั่นของ Diesel โดยจะนำค่ากลับต่างๆ จากระบบปฏิบัติการ

เมมเบอร์: Startup , StartupFromHwnd ,ChangeDisplayMode ,Shutdown ,AppFatalError ,Run , GetBack ,GetDisplayModes ,GetDisplayDevices ,SetAppTitle ,GetAppTitle ,GetAppPath , BuildFilepath ,LockWindowSize ,SetResources ,GetWindow ,SetActive , IsActive , IsWindowed ,IsInitDone ,GetFps ,GetDirectDraw ,SetPalette ,GetPalette , GetPaletteLookupTable ,GetClosestFromPalette ,BltToScreen ,MsgBox ,GetFrameTime , GetDWordFrameTime ,StartTimer ,StopTimer ,DoFlip ,OnMessage ,OnSize ,OnMove , OnMouseButton ,OnFlip ,OnGetTime ,OnRestore ,OnInitDone ,OnExit ,OnKeyDown , OnChar ,OnCommand ,GetApp ,GetDirectXVersion ,SetLastHresultError , GetLastHresultError ,HresToText ,WinErrorToText ,DieselErrorToText

กลุ่มของ CDiesel3D

ชื่อคลาส: CDiesel3DCamera

หน้าที่: คลาสที่จัดการเกี่ยวกับกล้อง เช่น หมุนกล้อง, พลิกกล้อง ต่างๆ เป็นต้น

เมมเบอร์: Startup , Shutdown ,Update ,Render ,SetProjectionParams ,GetProjectionParams , GetViewMatrix ,GetProjectionMatrix ,SetViewMatrix ,SetProjectionMatrix ,SetProjection , Duplicate

ชื่อคลาส: CDiesel3DDevice

หน้าที่: คลาสที่จัดการเกี่ยวกับอุปกรณ์ 3 มิติ เช่น การ์ดแสดงผลต่าง ๆ

เมมเบอร์: Startup ,Shutdown ,DrawPrimitive ,DrawPrimitiveVB ,DrawIndexedPrimitive , DrawIndexedPrimitiveVB ,DrawNurbs , DrawIndexedNurbs ,DrawSprite , DrawPrimitiveStream ,DrawIndexedPrimitiveStream ,SetStreamSource ,GetStreamSource , SetIndices ,GetIndices ,BeginScene ,EndScene ,ClearRenderTarget ,ClearZBuffer , SetTransform ,GetTransform ,TransformVectors ,SetLight ,GetLight , LightEnable , SetMaterial ,GetMaterial ,SetTexture ,GetTexture ,SetTextureState ,GetTextureState , SetTexturingType ,SetRenderTarget ,GetRenderTarget ,SetLighting ,GetLighting ,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SetAmbientLight ,GetAmbientLight ,SetSpecular ,GetSpecular , SetAlpha ,GetAlpha , SetFog ,GetFog ,SetFogParams ,GetFogParams ,SetFillMode , GetFillMode , SetZBuffering ,GetZBuffering ,SetZBufferWrite ,GetZBufferWrite ,GetZBuffer , SetViewport ,GetViewport ,Get3DDevice ,GetDirect3DDevice ,GetDirect3D

ชื่อคลาส: CDiesel3DLight

หน้าที่: คลาสที่จัดการเกี่ยวกับแสง เช่น กำเหนิดแสง ต่าง ๆ เช่น ไดเรกชั่น, สปอตร์ไลท์ และ สีต่างๆ ที่ได้จากแสง

ดาต้าเมมเบอร์: eDET_LIGHTTYPE eType ,CDiesel3DColor m_cDiffuse , CDiesel3DColor m_cSpecular ,CDiesel3DColor m_cAmbient ,CDieselVector3 m_vPosition , CDieselVector3 m_vDirection ,FLOAT m_fRange ,FLOAT m_fTheta ,FLOAT m_fPhi

เมมเบอร์: SetDirectional ,SetPoint ,SetSpot ,operator=

ชื่อคลาส: CDiesel3DMaterial

หน้าที่: คลาสกำหนดลักษณะพื้นผิวต่างๆ เช่นแสงสะท้อน ,ความหม่น หรือ ความโปร่งใสโปร่งแสง เป็นต้น

ดาต้าเมมเบอร์: CDiesel3DColor m_cDiffuse ,CDiesel3DColor m_cAmbient , CDiesel3DColor m_cSpecular ,CDiesel3DColor m_cEmissive ,FLOAT m_fShiness

เมมเบอร์: SetToDevice ,SetTexture ,GetTexture ,SetTextureState ,GetTextureState , SetTexturingType ,GetTexturingType ,operator =

ชื่อคลาส: CDiesel3DObject

หน้าที่: เป็นคลาสหลักในการสร้างวัตถุ 3 มิติต่าง ๆ

เมมเบอร์: Shutdown ,Render ,AddFaceMesh ,GetFaceMeshes ,Duplicate

ชื่อคลาส: CDiesel3DParitcleEmitter

หน้าที่: เป็นคลาสที่ใช้ในการผลิต พาติเคิล 3 มิติ ต่าง ๆ เช่นหมอก ควัน การแตกตัวต่างๆ

เมมเบอร์: Cube ,Sphere ,Cylinder ,Cone ,Torus

ชื่อคลาส: CDiesel3DScene

หน้าที่: เป็นคลาสใช้ในการเก็บวัตถุ 3 มิติต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมมเบอร์: Startup ,Shutdown ,Render ,Update ,Load ,LoadFromResource , LoadFromMemory ,Save ,FindObject ,SubclassObject ,GetActiveCamera , SetActiveCamera ,RemoveObject ,GetRootObject ,SetRootObject ,GetMaterials , GetLights , GetVertexBuffers ,GetIndexBuffers ,GetTextures , GetCameras , GetAnimators , GetVertexBuffer ,GetIndexBuffer ,GetMaterial ,GetTexture ,GetLight , GetCamera ,GetAnimator ,AddVertexBuffer ,AddIndexBuffer ,AddMaterial ,AddTexture , AddLight ,AddCamera ,AddAnimator ,OptimizeAllVertexBuffers ,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8
การเขียนโปรแกรมเน็ตเวิร์ค (TCP/IP) โดยใช้ วินซ็อค (Winsock)

ในบทนี้จะเป็นการเขียนโปรแกรมเน็ตเวิร์คระหว่าง เซิร์ฟเวอร์ (Server) กับไคลเอนท์ (Client) ด้วยการใช้โปรโตคอล ทีซีพี / ไอพี (TCP/IP) ใน วินซ็อค (Window Socket API)

8.1 ซ็อคเก็ต (Socket)

ซ็อคเก็ตถูกกำหนดหรือนิยามไว้ว่า เป็นคู่ของการสื่อสาร หรือ คู่ของโพรเซส (หรือ เธรด) โดยที่การสื่อสารบน เน็ตเวิร์คใช้คู่ของซ็อคเก็ต สำหรับแต่ละโพรเซส

สำหรับซ็อคเก็ตประกอบไปด้วย ไอพี แอดแดรส (IP Address) กับหมายเลข พอร์ท (Port Number) โดยทั่ว ๆ ไป ซ็อคเก็ตใช้สถาปัตยกรรมไคลเอนท์ เซิร์ฟเวอร์ ๆ จะรอการเข้ามาตามการขอร้องของ ไคลเอ็นท์โดยการฟังที่ พอร์ท เฉพาะ เมื่อการขอร้องได้รับ เซิร์ฟเวอร์ก็จะยอมรับการเชื่อมต่อจากซ็อคเก็ตไคลเอนท์ เพื่อให้สมบูรณ์ในการเชื่อมต่อ

เซิร์ฟเวอร์ที่สร้างการบริการเฉพาะ เช่น เทลเน็ท (Telnet), เอฟทีพี (Ftp), จดหมายอิเล็กทรอนิกส์ (e-mail) และ เฮดทีทีพี (Http) จะฟัง (listen) ที่พอร์ทมีชื่อ เช่น เซิร์ฟเวอร์ เทลเน็ท จะฟังที่พอร์ต 23, เซิร์ฟเวอร์ เอฟทีพี จะฟังที่พอร์ต 21 หรือ เซิร์ฟเวอร์ เฮดทีทีพี จะฟังที่พอร์ท 80 เป็นต้น

หมายเลขพอร์ททั้งหมดที่ต่ำกว่า 1024 จะถูกพิจารณาว่าเป็นพอร์ตที่มีชื่อเสียง เราสามารถใช้พอร์ทเหล่านี้เพื่อสร้างการบริการตามมาตรฐานได้

8.2 ชนิดของซ็อคเก็ต

ชนิดของซ็อคเก็ตมีอยู่ 3 ชนิด คือ

- การเชื่อมต่อแบบ ต่อเนื่อง (Connection oriented Socket)
- การเชื่อมต่อแบบ ไม่ต่อเนื่อง (Connectionless Socket)
- รอพ์ ซ็อคเก็ต (Raw Socket)

8.2.1 การเชื่อมต่อแบบ ต่อเนื่อง (Connection oriented Socket)

เป็นซ็อคเก็ตการเชื่อมต่อแบบต่อเนื่อง ที่อนุญาตให้โพรเซสเชื่อมต่อ กับโพสเซสระยะไกล (Remote) ซึ่งใช้โปรโตคอล ทีซีพี (TCP: Transmission Control Protocol) ดังนั้นด้วยวิธีการนี้ทำให้ข้อมูลเชื่อถือได้ เมื่อการเชื่อมต่อได้เกิดขึ้น โพรเซสก็จะมีการส่งข้อมูลกลับไปจนกระทั่งฝั่งใดฝั่งหนึ่งหรืออื่น ๆ มีการปิดการเชื่อมต่อ ชนิดของซ็อคเก็ตนี้ บางครั้งเรียกว่า สตรีมซ็อคเก็ต (Stream Socket) ทั้ง เอฟทีพี (FTP) และ เฮดทีทีพี (HTTP) ใช้ซ็อคเก็ตแบบนี้ในการสื่อสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.2.2 การเชื่อมต่อแบบ ไม่ต่อเนื่อง (Connectionless Socket)

หรือเรียกอีกอย่างหนึ่งว่า ดาต้าแกรมเป็นซ๊อคเก็ตแบบไม่ต่อเนื่อง และนำมาใช้เป็นประโยชน์ในการส่งเมสเสจนั้น ๆ ซึ่งไม่สามารถสนับสนุนส่วนหัว ดังนั้นจึงพิจารณาการเชื่อมต่อแบบต่อเนื่องที่ซ็อคเก็ตปลายทางถูกตรวจสอบเมื่อแพ็กเก็ตถูกส่งต่อไป

ซ็อคเก็ตแบบไม่ต่อเนื่อง เปรียบเสมือนกับการบริการของไปรษณีย์ ที่ผู้ส่งจดหมายไปตามที่อยู่แล้วใส่ในกล่องรับจดหมาย ผู้ส่งจะไม่ทราบว่าผู้รับได้รับจดหมายหรือไม่ ซ็อคเก็ตแบบนี้นิยมใช้กันในเซิร์ฟเวอร์ ดีเอ็นเอส (DNS: Domain Name System) ที่ซ็อคเก็ตดาต้าแกรมในการตอบสนองต่อการขอร้องที่เข้ามามาก ๆ

นอกจากนี้จะใช้ดาต้าแกรมซ็อคเก็ตในการกระจาย (Broadcast) เมสเสจ หรือ มัลติคาส (Multicast) เพื่อไปยังปลายทางหลาย ๆ แห่งพร้อมกัน ซึ่งเหมือนกับการกระจายเสียงวิทยุ หรือ โทรทัศน์

8.3 ทีซีพี/ไอพี

ระบบเครือข่ ายที่ใช้ โปรโตคอลมาตรฐานชื่อทีซีพี /ไอพีในการสื่อสารผ่านระบบเพื่อติดต่อกับเครื่อง คอมพิวเตอร์อื่นๆโปรโตคอลทีซีพี/ไอพีนั้นประกอบด้วยส่วนที่สำคัญ 2 ส่วนก็คือ

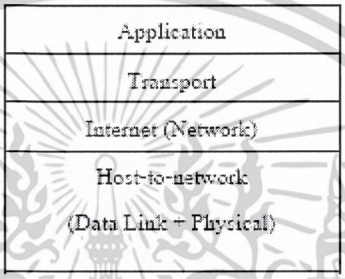
- ทีซีพี(Transmission Control Protocol: TCP)
- ไอพี (Internet Protocol: IP)

ในการติดต่อสื่อสารกัน จริงๆ แล้วคงจะไม่ สามารถเห็นขั้นตอนการทำงานของระบบได้ เพราะเป็น การทำงานของซอฟแวร์และฮาร์ดแวร์แต่สามารถอธิบายเพื่อให้เข้าใจโปรโตคอลทีซีพี-ไอพีซึ่งมีส่วนประกอบดังนี้

- ตำแหน่งไอพี (IP Address) สำหรับการรับส่งข้อมูลในระบบอินเตอร์เน็ตจะถูกกำหนดและอ้างอิงด้วยหมายเลขประจำเครื่องนั้นก็คือ ตำแหน่งไอพีซึ่งในระบบอินเตอร์เน็ตจะมีเครื่องคอมพิวเตอร์ เป็น จำนวนมากที่อยู่ในระบบ ดังนั้นการที่จะใช้ตำแหน่งไอพีอาจจะไม่สะดวกจึงได.มีการเปลี่ยนมาใช้เป็นชื่อ ในความเข้ากันก็คือชื่อโดเมน (Domain name) โดยทั้งหมดนี้อยู่ในระบบบริการชื่อ (Name Services) ซึ่งเป็นการอ้างอิงชื่อแทนหมายเลขนั้นเอง
- ลักษณะเร้าติ้ง (Routing Configuration) ข้อดีของโปรโตคอลทีซีพี-ไอพีก็คือในการกำหนดเส้นทาง สำหรับการรับส่ง ที่สามารถเลือกเส้นทางในการรับส่งข้อมูลได้อย่างอัตโนมัติหากถ้าเกิดเส้นทาง บ้างเส้นทางเสียหาย ระบบกลไกในการกำหนดเส้นทางสำหรับการรับส่งข้อมูลของโปรโตคอลทีซีพี-ไอ พีก็จะเลือกเส้นทางให้เหมาะสมถูกต้องให้สามารถรับส่งข้อมูลได้
- โปรโตคอล (Protocol), พอร์ท (Ports), ซ็อกเก็ต (Sockets) เป็นช่องทางสำหรับกำหนดทิศทางของ การรับส่งข้อมูลนอกเหนือจากที่จะต้องกำหนดหลังจากตำแหน่งไอพี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทีซีพี/ไอพี จึงเป็นโพรโตคอลมาตรฐานที่มีการใช้งานกันอย่างแพร่หลายในทุกๆระดับ ทั้งเครือข่ายเฉพาะขนาดเล็กตลอดจนเครือข่ายในบริเวณกว้าง ทีซีพี/ไอพีผ่านการออกแบบให้เป็นอิสระจากคอมพิวเตอร์ทั้งฮาร์ดแวร์และระบบปฏิบัติการ กลไกภายในมีความเชื่อถือได้สูงและทำงานได้แม้ในภาวะที่ผิดปกติ รวมทั้งยังสามารถเลือกเส้นทางได้ในการส่งข้อมูลได้แม้บางเส้นทางเกิดการชำรุด ระบบการสื่อสารข้อมูลในเครือข่ายคอมพิวเตอร์ประกอบไปด้วยฮาร์ดแวร์ และซอฟต์แวร์จึงมีการแบ่งแยกออกเป็นส่วนๆเพื่อช่วยลดความซับซ้อน ชื่อ ทีซีพี/ไอพี มีที่มาจากโพรตคอลสองโพรโตคอลคือ TCP – Transmission Control Protocol และ IP – Internet Protocol แต่ ทีซีพี/ไอพี ไม่ได้มีส่วนประกอบเพียงแค่สองโพรโตคอลเท่านั้นแต่แบ่งออกเป็นระดับต่างๆดังรูปที่ 6.3



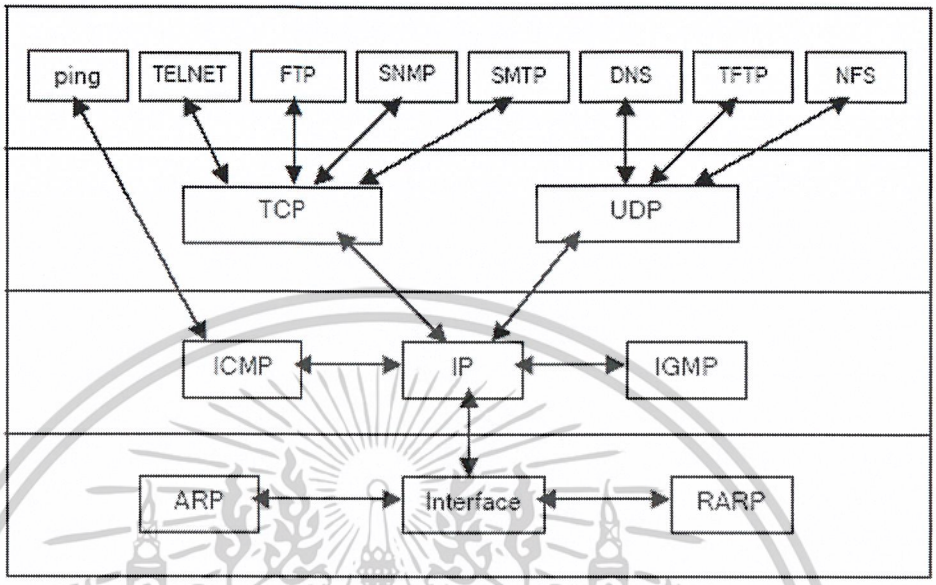
รูปที่ 8-1 แบบอ้างอิง ทีซีพี/ไอพี

โฮสทูเน็ทเวิร์ค (Host-to-network) ในระดับชั้นนี้จะมีการทำงานใน 2 ระดับคือ Physical และ Data Link

- ฟิสิกคอล (physical layer) ชั้นของการกำหนดคุณสมบัติฮาร์ดแวร์ เช่นคุณสมบัติทางกลอย่างหัวต่อ ชนิดสายสื่อสาร คุณสมบัติทางไฟฟ้าอย่างลักษณะสัญญาณ อัตราเร็ว เป็นการกำหนดวิธีการสูงข้อมูลในระดับบิต ตัวอย่างเช่น RS232 และ X.21
- ดาต้าลิ้งค์ (data-link layer) ในระดับชั้นนี้จะทำงานด้านการเชื่อมโยงเข้ากับสายสื่อสาร ตัวอย่างมาตรฐานในระดับชั้นนี้เช่นอีเทอร์เน็ตและโทเคนริง Internet ทำหน้าที่เลือกเส้นทางเพื่อส่งข้อมูลระหว่างสถานีต้นทางและปลายทาง ตัวอย่างเช่นโพรโตคอล ไอพี
- ชั้นทรานสปอร์ต (transport layer), ทำหน้าที่จัดเตรียมการส่งข้อมูลระหว่างสถานีต้นทางและปลายทาง โดยการสถาปนาการเชื่อมต่อและรักษาการเชื่อมต่อตลอดจนยกเลิกการเชื่อมต่อเมื่อสิ้นสุดกระบวนการ และอาจยังต้องทำหน้าที่รักษาความถูกต.องของข้อมูล มีโพรโตคอลในระดับชั้นนี้อยู่สองโพรโตคอลคือ ทีซีพี และ ยูดีพี
- แอพพริเคชั่น (Application layer) เป็นระดับชั้นการทำงานของโปรแกรมประยุกต์ตัวอย่างโพรโตคอลเช่น FTP ,SMTP, TELNET

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานหนึ่งๆนั้นไม่ได้ใช่โปรตคอลพร้อมกันทั้งหมด แต่ใช้งานสัมพันธ์กันไปในแต่ละระดับชั้น การซ้อนทับของโพรโตคอลจากชั้นบนลงไปชั้นล่างเรียกว่าโพรโตคอลแสตกดังรูปที่ 6.4



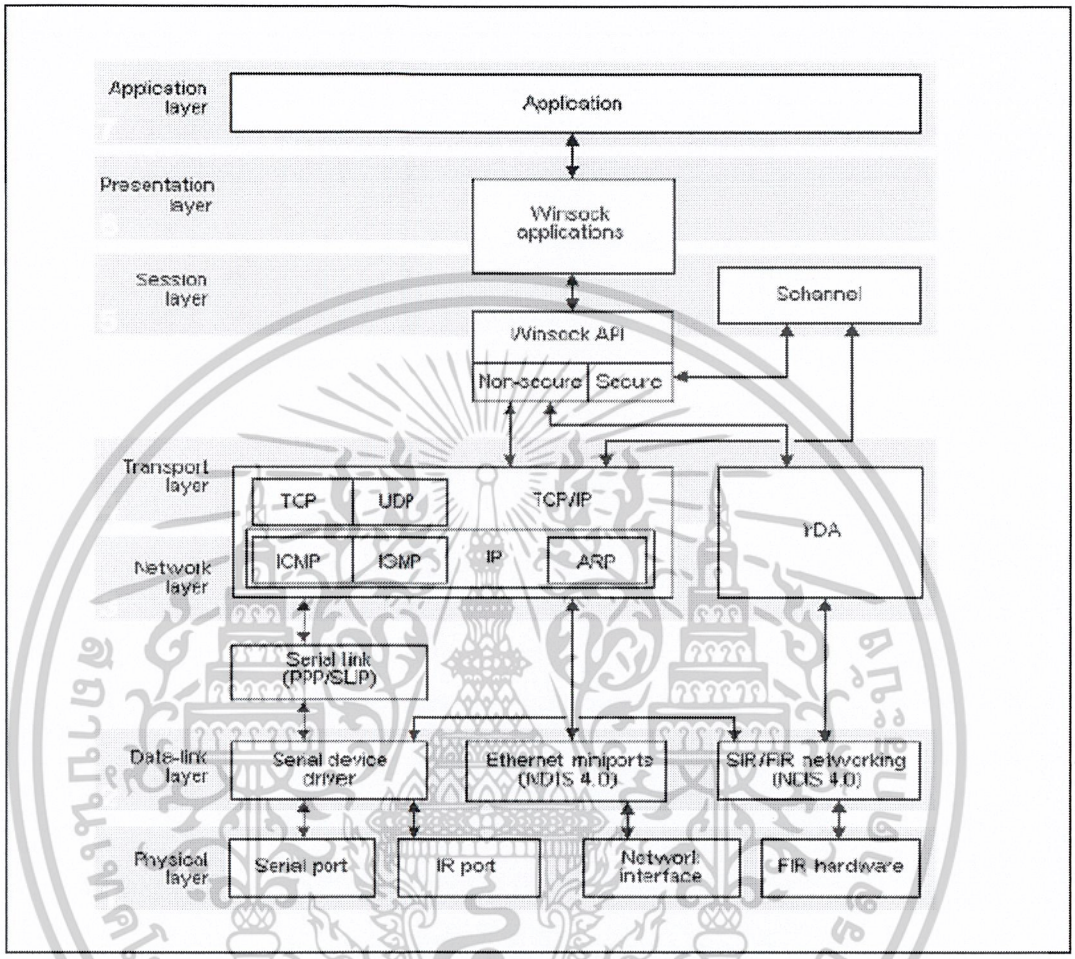
รูปที่ 8-2 โพรโตคอลแสตกของ ทีซีพี/ไอพี

ไอพีเป็นโพรโตคอลที่อยู่ในระดับชั้น เน็ทเวิร์ค เป็นแกนสำคัญในโพรโตคอลแสตกเนื่องจากทั้งยูดีพีและทีซีพีต่างก็ต้องใช้โพรโตคอลไอพีเพื่อเลือกเส้นทางในการส่งแพ็กเก็ต ข้อมูลจะถูกส่งจากโพรโตคอลชั้นบนสุดลงมาโพรโตคอลชั้นล่างสุดในสถานีส่งและถูกเปลี่ยนเป็นสัญญาณทางไฟฟ้าส่งผ่านสื่อตัวกลางไปยังสถานีปลายทาง ที่สถานีปลายทางก็จะรับสัญญาณแล้วส่งผ่านไปยังโพรโตคอลระดับบน เมื่อข้อมูลผ่านไปในแต่ละระดับชั้น โพรโตคอลในชั้นนั้นจะผนวกข้อมูลที่ใช้ในการทำงานของโพรโตคอลระดับนั้นเรียกว่าส่วนหัวของโพรโตคอล (Protocol Header) เข้าไปกับส่วนข้อมูลและเรียกกระบวนการนี้ว่าเอ็นแคปซูเลชั่น(Encapsulation) ข้อมูลจะถูกส่งไปยังส่วนล่างกว่าโดยโพรโตคอลชั้นล่างจะมองว่าข้อมูลที่ส่งมาทั้งหมดนั้นเป็นข้อมูลทั้งหมดโดยไม่ทราบว่าส่วนใดเป็นข้อมูลจริงหรือส่วนหัวของโพรโตคอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.4 วินซ็อค (Winsock)

8.4.1 วินซ็อคและโอเอสไอโมเดล (OSI Model)



รูปที่ 8-3 แสดงความสัมพันธ์ระหว่างวินซ็อคกับโปรโตคอลสำหรับการสื่อสารอื่นๆ

ในไอเอสโอ/โอเอสไอโมเดล (ISO/OSI Model) นั้นวินซ็อคจะทำงานที่อินเตอร์เฟสระหว่างชั้นเซสชั่น (Session Layer) กับชั้นทรานสปอร์ต (Transport Layer) โดยวินซ็อคทำหน้าที่เป็นตัวกลางระหว่างแอพพลิเคชั่นกับ โปรโตคอลสำหรับการส่งข้อมูลและเป็นทางส่งข้อมูลอีกด้วย รูปที่ 4.1 แสดงความสัมพันธ์ระหว่างวินซ็อคกับโปรโตคอลวินโดว์ซีอีสำหรับการติดต่อสื่อสาร (Window CE communication protocols) อื่นๆ ในขอบเขตของ ไอเอสโอ/โอเอสไอโมเดล

โดยวินซ็อคจะทำให้แอพพลิเคชั่นที่พัฒนาในไอเอสโอ/โอเอสไอชั้นบนมีความง่ายขึ้นโดยจะเก็บเฉพาะ รายละเอียดข้อมูลของเครือข่ายที่จะแลกเปลี่ยนกับชั้นที่ต่ำลงมาเท่านั้น นอกจากนี้วินซ็อคยังได้เพื่อความสามารถ ให้โปรแกรมเมอร์สามารถเขียนโปรแกรมการติดต่อระหว่างชั้นบน (5-7) กับชั้นล่าง (1-4) ได้อีกด้วย ข้อมูลของวินซ็อคแอพพลิเคชั่นจะถูกทำเป็นแพ็กเก็ตและส่งผ่านเครือข่ายโดย ชั้นทรานสปอร์ต (Transport layer), ดาต้าลิ้งค์ (data-link layer) และ ฟิสิกคอล (physical layer)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

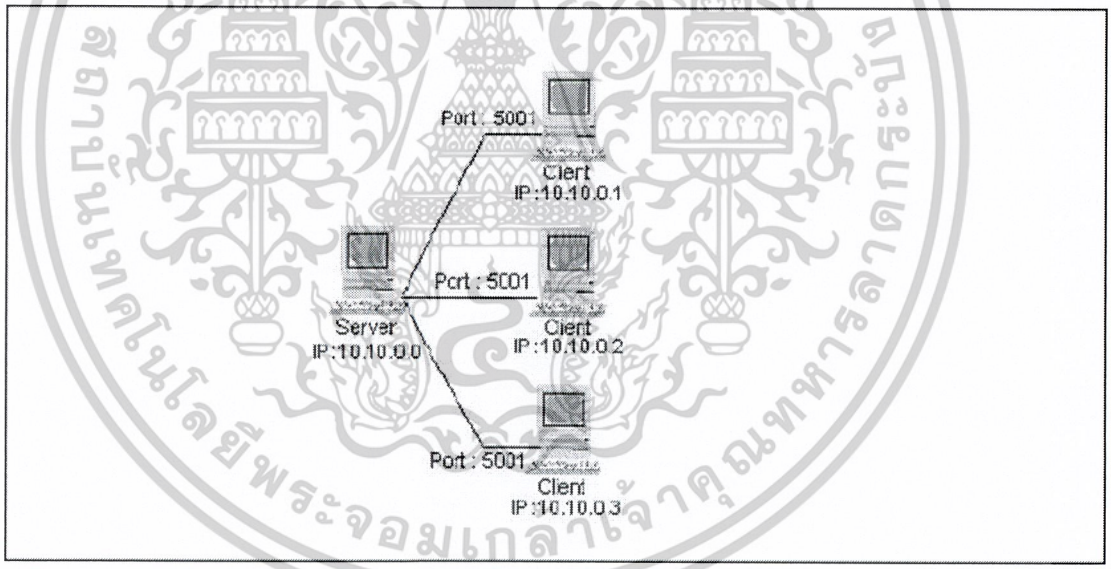
8.4.2 การเขียนโปรแกรมการเชื่อมต่อผ่านเครือข่าย ด้วยวินซ็อค บนวินโดว์ซีอี (Window CE)

ในปัจจุบันการสื่อสารผ่านระบบเครือข่าย (Network)รวมทั้งระบบอินเตอร์เน็ต (Internet) เป็นที่แพร่หลาย มากในแง่ของการใช้งาน, ใช้บริการและอำนวยความสะดวกในชีวิตประจำวันต่างๆเนื่องจากระบบสื่อสารที่รวดเร็ว และสามารถเชื่อมโยง เข้ากับหลายๆระบบได. ทำให้สามารถนำมาประยุกต์ใช้ งานผ่านระบบเครือ ข่ายได้หลาก หลายรูปแบบ ซึ่งในบทความนี้จะพยายามเน้นไปประยุกต์ใช้ ในการควบคุมอุปกรณ์หรือโปรแกรมต่างๆ เรื่องที่ ต้องรู้ในการที่จะเขียนโปรแกรมผ่านระบบเครือข่ายมีดังนี้

8.4.2.1 เซิร์ฟเวอร์และไคลเอ็นท์ (Server and Client)

สิ่งที่เราจะพูดถึงในการเขียนโปรแกรมเพื่อใช้ในระบบเครือข่าย (Network) จุดหลักๆของระบบจะ

แบ่ง ฝ้ายที่ต้องติดต่อรับส่งข้อมูลระหว่างกันออกเป็น 2 ส.วน คือเซิร์ฟเวอร์ (Server) และไคลเอ็นท์ (Client) ซึ่งในการ ใช้งานจริงอาจมีส่วนประกอบอื่นๆอีก ซึ่งจะขอไม่กล่าวถึงเนื่องจากต้องการให้เห็นภาพและเข้าใจง่ายขึ้นจึงยกแค่ 2 ส่วนนี้มากล่าว



รูปที่ 8-4 โครงสร้างการทำงานระหว่างไคลเอ็นท์กับเซิร์ฟเวอร์

เซิร์ฟเวอร์จะเป็นส่วนทำหน้าที่เสมือนกองอำนวยการ, ประชาสัมพันธ์, เมสเซทเจอร์ ร่วมถึงผู้จัดให้กับระบบ ส่วนไคลเอ็นท์จะเป็นส่วนร้องข้อมูลจากเซิร์ฟเวอร์ โดยเซิร์ฟเวอร์ในที่นี้จะเป็นส่วนที่เก็บข้อมูล, จัดการ, บริหารข้อมูลหรือทรัพยากรของระบบ เพื่อให้ฝ่ายไคลเอ็นท์สามารถใช้บริการได้

เซิร์ฟเวอร์และไคลเอ็นท์ต่างก็จะต้องมีตำแหน่งไอพี , พอร์ท (Port) โดยทั้งสองฝ่ายจะสามารถติดติอถึงกันได้จะต้องอยู่ในช่องทางเดียวกัน ซึ่งเราสามารถกำหนดหมายเลขของพอร์ทได้ ทั้งนี้โปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ติดต่อนั้นจะต้อง อ้างอิงหมายเลขของพอร์ททุกครั้งเนื่องจากในระบบมีโปรแกรมมากมายที่กำลังติดต่อกันอยู่

8.4.2.2 ฟังก์ชัน วินซ็อคโพร (Winsock Function)

ในส่วนของวินโดว์โค้ดอ็อฟเจ็ค (Windows Code Object) หรือวินซ็อค นั้นมีฟังก์ชันสำหรับกระทำติดต่อสื่อสารทั้งทางฝ่ายเซิร์ฟเวอร์และไคลเอ็นท์ โดยที่

- Create ใช้สำหรับการสร้างซ็อคเก็ตใหม่
- GetLastError เป็นการเอาค่าสถานะผิดพลาดการกระทำครั้งสุดท้ายที่ล้มเหลว
- GetPeerName เป็นการเอาค่าที่อยู่ ของซ็อคเก็ตอื่น ๆ ที่เชื่อมต่อกัน
- GetSockName เอาค่าที่อยู่ของซ็อคเก็ตโลคอล
- Accept ยอมรับการเชื่อมต่อบนซ็อคเก็ต
- Bind ทำการไบนด์ที่อยู่โลคอลกับซ็อคเก็ต และใช้กับสตรีมซ็อคเก็ตเท่านั้น
- Close ปิดซ็อคเก็ตและถ้าออบเจ็กต์นี้ไม่ได้ใช้งาน ตัวทำลาย (Destructor) จะเรียกฟังก์ชันนี้ให้
- Conect สร้างการเชื่อมต่อกับ ซ็อคเก็ต อื่น ๆ และ ใช้กับสตรีมซ็อคเก็ต
- Listen สร้างซ็อคเก็ตเพื่อฟังการเข้ามาการขอร้องการเชื่อมต่อและใช้กับสตรีมซ็อคเก็ต
- Receive รับข้อมูลซ็อคเก็ต ใช้กับสตรีมซ็อคเก็ตและดาต้าแกรมซ็อคเก็ต
- ReceiveFrom รับข้อมูลดาต้าแกรมและเก็บที่อยู่ที่มาไว้ในโครงสร้าง SOCKADDR
- Send ส่งข้อมูลบนซ็อคเก็ตที่ถูกเชื่อมต่อใช้ได้ทั้งสตรีม และ ดาต้าแกรมซ็อคเก็ต
- SendTo ส่งข้อมูลไปยังปลายทางโดยเฉพาะใช้ได้ทั้งสตรีมและ ดาต้าแกรมซ็อคเก็ต
- ShutDown ทำให้การเรียกฟังก์ชัน Send หรือ Receive ไม่สามารถใช้ได้บนซ็อคเก็ต
- OnAccept ถูกเรียกบนซ็อคเก็ตที่กำลังฟัง (Listening Socket) เพื่อให้สัญญาณว่าการขอร้องการเชื่อมต่อจากแอพพลิเคชั่นอื่น ๆ กำลังรอคอยพร้อมที่จะถูกยอมรับ
- OnConnect ถูกเรียกบนซ็อคเก็ตที่กำลังต่อ (Connecting Socket) เพื่อให้สัญญาณว่าการเชื่อมต่อกับแออพลิเคชั่นอื่น ๆ ได้เรียบร้อยสมบูร์แล้ว และแอพพลิเคชั่นพร้อมที่จะส่งรับเมสเสจบนซ็อคเก็ตทั้งสอง
- OnSend แจ้งให้ทราบว่าซ็อคเก็ตนี้สารารถพร้อมจะส่งข้อมูลได้โดยการเรียกฟังก์ชัน Send
- OnReceived แจ้งให้ทราบว่าซ็อคเก็ตนี้มีข้อมูลอยู่ในบัฟเฟอร์ และพร้อมที่จะถูกดึงมาใช้โดยการเรียกฟังก์ชัน Receive
- OnClose แจ้งให้ทราบว่าซ็อคเก็ตนี้ที่แอพพลิเคชั่นอีกฝั่งหนึ่งของการเชื่อมต่อมีการปิดแล้วหรือ เกิดการสูญเสียการเชื่อมต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 9
แนวคิดของเกมส์

เกมส์นี้ เป็นเกมส์แนวการขับเครื่องบินไล่ล่า ที่มีการต่อสู้กัน โดยมีการขับเครื่องบิน, การค้นหาศัตรูโดยพึ่งค่าพารามิเตอร์ต่างๆ, การโจมตีศัตรู, การรักษาสมดุลการบินให้ได้ตลอดรอดฝั่ง ซึ่งองค์ประกอบเหล่านี้เองที่เราจำเป็นที่จะต้องเขียนโปรแกรมขึ้นมา โดยจุดประสงค์เพื่อให้เป็นเกมส์ที่สร้างความสนุกสนานแก่ผู้เล่น อีกทั้งเป็นเกมส์ที่มีความสวยงามของภาพและมีความสมจริง อีกด้วย

ในขั้นต้น เราจะต้องมีโครงสร้างโดยรวมของวัตถุและผลกระทบต่างๆ ที่มีอยู่ในเกมส์ และสร้างความสัมพันธ์ของวัตถุต่อวัตถุขึ้นมาก่อน เช่น ขีปนาวุธ มีไว้โจมตีศัตรู, ภูเขาเป็นสิ่งที่ห้ามชนหรือห้ามกระทบโดยเครื่องบิน จากนั้นค่อยสร้างความสัมพันธ์ของวัตถุต่อผลกระทบ ตัวอย่างเช่น ความเร็วของเครื่องบิน ส่งผลกระทบต่อความเร็วของขีปนาวุธ เป็นต้น ซึ่งรายละเอียด ความสัมพันธ์เหล่านี้เอง ที่เราจำเป็นที่จะต้องสร้างมันขึ้นมา เพื่อให้เป็นเกมส์ที่มีความครบถ้วนสมบูรณ์

9.1 วัตถุและผลกระทบ ต่างๆ ในเกมส์

ในเกมส์นี้ มีวัตถุ, และผลกระทบต่างๆ ที่มีความสัมพันธ์กันดังต่อไปนี้

- ความเร็ว: เป็นผลกระทบที่ส่งผลให้วัตถุใดๆ สามารถเคลื่อนที่ไปข้างหน้าได้ โดยยิ่งมีความเร็วมาก ก็ยิ่งเคลื่อนที่ได้ระยะไกลมากกว่า ในเวลาที่เท่ากัน
- แรงโน้มถ่วง: เป็นผลกระทบต่อวัตถุที่อยู่อากาศทุกชิ้น โดยแรงโน้มถ่วงนี้ เป็นแรงเวคเตอร์ที่มีทิศทางชี้ลงในแนวแกน y- ซึ่งผลกระทบของแรงโน้มถ่วง จะทำให้วัตถุทุกชิ้น ตกลงมายังพื้นดิน
- แรงพยุงเครื่องบิน: เป็นผลกระทบที่มีต่อเครื่องบิน โดยแรงพยุงเครื่องบินนี้ ได้รับผลกระทบมาจากความเร็ว อีกทอดหนึ่ง โดยถ้าเครื่องบินยิ่งมีความเร็วมากเท่าไร แรงพยุงเครื่องบินก็ยิ่งมีมากขึ้นเท่านั้น ซึ่งทำให้เครื่องบิน สามารถบินขึ้นไปได้
- ความสามารถในการเลี้ยวหักเห ของเทหวัตถุ: เป็นความสามารถในการเลี้ยวของวัตถุทุกชิ้นที่อยู่ในอากาศ โดยปกติแล้ว ความสามารถในการเลี้ยวของเครื่องบิน นั้น จะแปรผันกันกับความเร็วของเครื่องบินด้วย กล่าวคือ ยิ่งเครื่องบินมีความเร็วมากเท่าไร ก็สามารถเลี้ยวหักเหได้มากขึ้นเท่านั้น ส่วนความสามารถในการเลี้ยว ของขีปนาวุธนั้น มีค่าคงที่หนึ่งๆ ซึ่งขีปนาวุธนี้ มีความสามารถในการติดตามศัตรูได้ ซึ่งมันจะเลี้ยวไปตามทิศทางที่ศัตรูอยู่ ดังนั้น ถ้าต้องการหลบจากการถูกยิงโดยขีปนาวุธ ก็จำเป็นต้องอยู่นอกวิถีการเลี้ยวหักเหของขีปนาวุธ
- ความสามารถในการก้ม-เงย ของเทหวัตถุ: มีลักษณะคล้ายกันกับความสามารถในการเลี้ยวหักเห นั่นคือเครื่องบินสามารถก้ม-เงยได้มากขึ้น เมื่อมีความเร็วมากขึ้น , แต่ตัวขีปนาวุธจะมีค่านี้คงที่
- น้ำหนักวัตถุ: ค่าน้ำหนักของวัตถุนี้ จะส่งผลกระทบต่อความเร็วของเครื่องบิน โดยน้ำหนักของเครื่องบิน จะแปรผันกันกับ ปริมาณน้ำมัน และ จำนวนขีปนาวุธ นั่นคือยิ่งมีน้อยเท่าไร ก็จะมีความเร็วมากขึ้นเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ปริมาณน้ำมัน: ปริมาณน้ำมันนั้น แปรผันกลับ กันกับความเร็ว นั่นคือ หากเครื่องยิ่งมีปริมาณน้ำมันน้อยเท่าไร ก็จะยิ่งบินได้เร็วมากขึ้นเท่านั้น และ ยิ่งเครื่องบินเร็วมากขึ้นเท่าไร ก็จะเผาพลาญปริมาณน้ำมัน ให้ลดลงเร็วเท่านั้น
- ปริมาณขีปนาวุธ: จะลดลงหากผู้เล่นได้ยิงขีปนาวุธออกไป และปริมาณขีปนาวุธ จะส่งผลแปรผันกลับ กับน้ำหนักเครื่องบิน
- พลังชีวิต: เป็นค่าที่ใช้วัดความเสียหายที่เกิดขึ้นของเครื่องบิน หากมีค่าเป็น ศูนย์ นั่นคือ ไม่สามารถเล่นได้อีกต่อไป
- ภูเขา: เป็นสิ่งกีดขวางเพียงอย่างเดียวในเกม ซึ่งเราต้องหลีกเลี่ยงการชนภูเขา เพื่อป้องกันการเสียหาย
- ขีปนาวุธ: เป็นอาวุธเพียงอย่างเดียวในเกม ที่ใช้ในการโจมตี
- เครื่องบิน: เป็นวัตถุที่มีผลกระทบในตัวเกมมากที่สุด และมีผลกระทบที่เกิดขึ้นโดยตรงจากตัวผู้เล่น

9.2 รูปแบบการเล่น

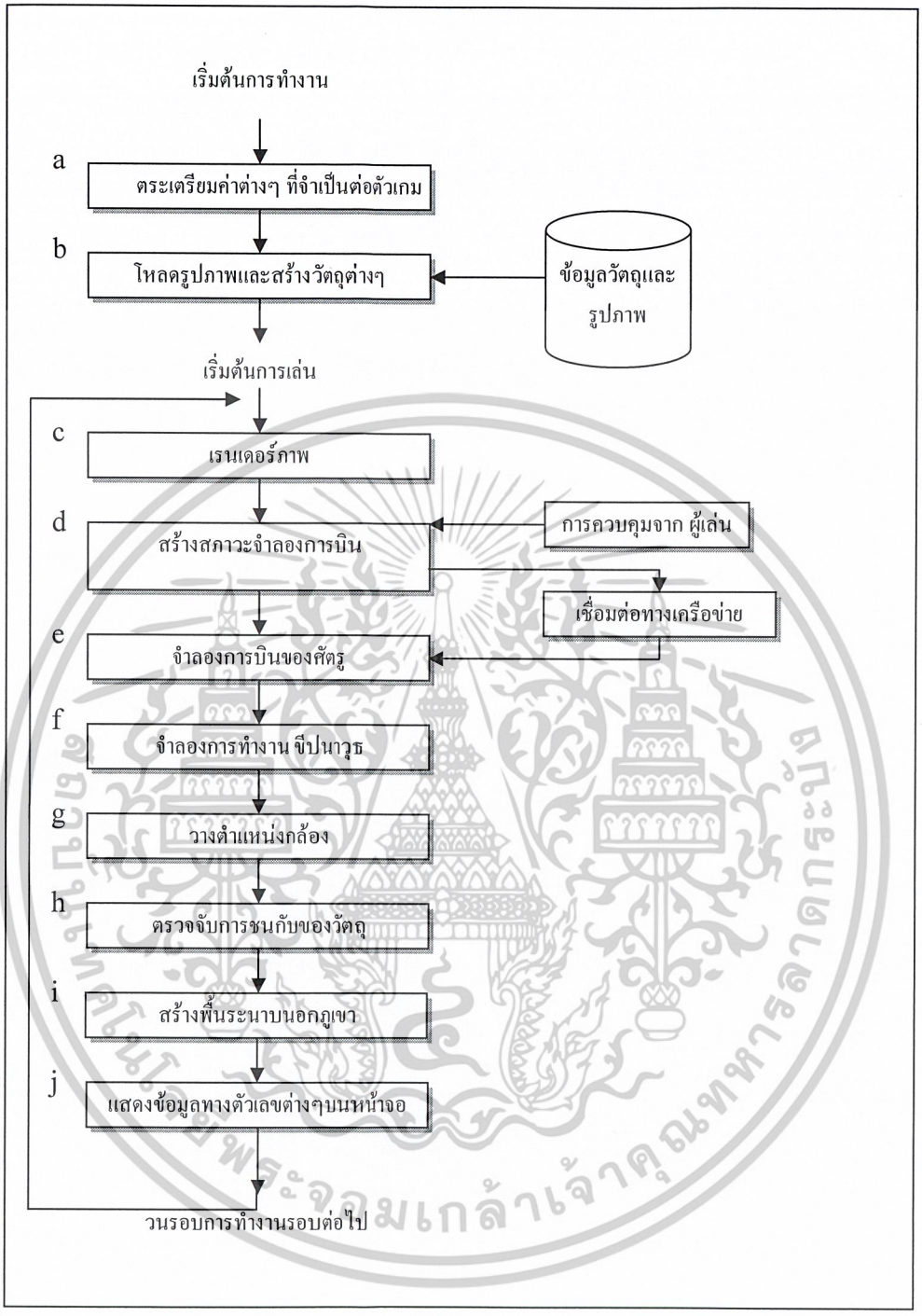
งานหลักก็คือ กำจัดศัตรูให้ได้มากที่สุดเท่าที่จะทำได้ โดยมี ปริมาณเชื้อเพลิง, ปริมาณขีปนาวุธ, และพลังชีวิต เป็นข้อจำกัดที่จะทำให้เกมนั้น เล่นยากขึ้น ซึ่งการที่จะเล่นให้ได้ดีที่สุดนั้น จำเป็นต้องคำนึงถึงการขับเครื่องบิน ที่ใช้ความเร็วที่เหมาะสม เพื่อเป็นการประหยัดเชื้อเพลิง และ เป็นการง่ายในการเล็งหาศัตรู นอกจากนี้ ผู้เล่น จำเป็นที่จะต้องคำนึงถึงมุมและระยะในการยิงขีปนาวุธ ด้วย ว่ามีมุมและระยะการยิงที่เพียงพอหรือไม่ ที่จะทำให้ขีปนาวุธ วิ่งเข้าสู่เป้าหมาย อีกทั้ง จำเป็นที่ต้องระวังการโจมตีกลับของเครื่องบินฝ่ายศัตรูอีกด้วย

9.3 การทำงาน ของโปรแกรม

จากรูปที่ 9-1 นั้นการทำงานโดยภาพรวมของโปรแกรมเกมส์ จะมีการทำงานโดยละเอียดในแต่ละส่วน ดังต่อไปนี้

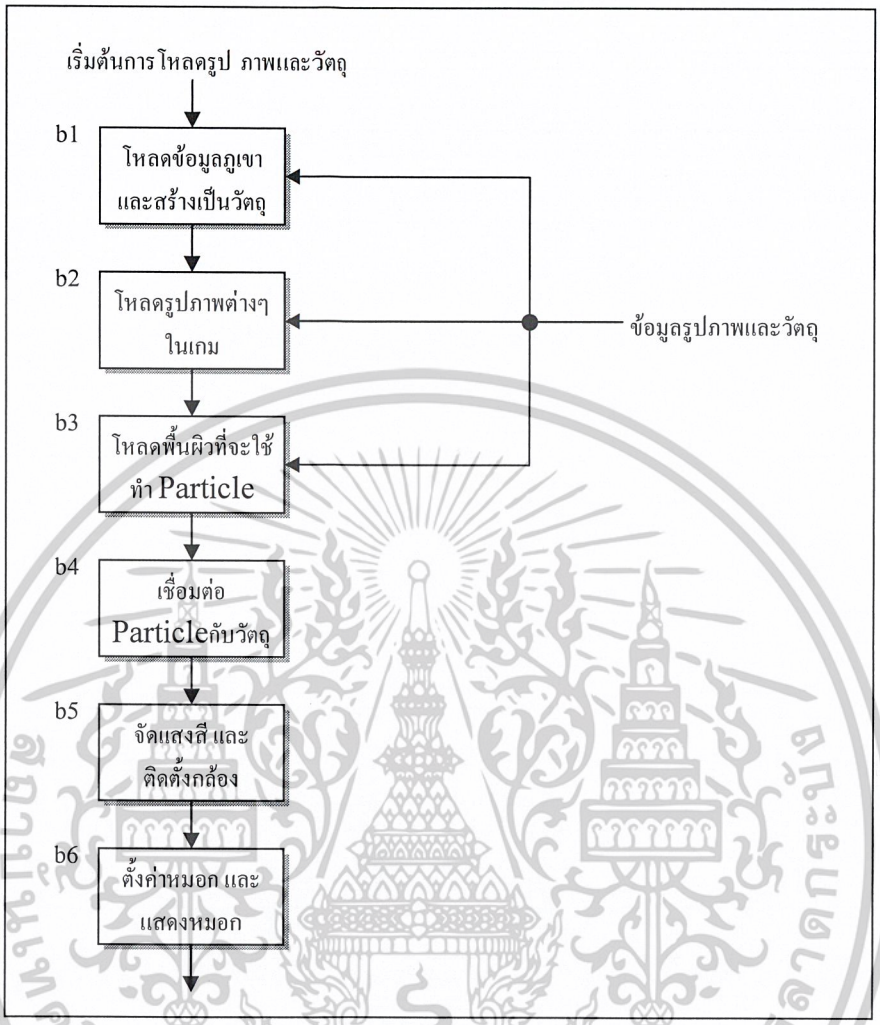
- a) ขั้นตอนตระเตรียมสิ่งต่างๆที่จำเป็นต่อตัวเกม: ในขั้นตอนนี้ จะเป็นการตั้งค่าเริ่มต้นต่างๆ เช่น ปริมาณเชื้อเพลิง, จำนวนขีปนาวุธ เป็นต้น ตลอดจนเป็นการตั้งค่าให้ตัวเกมส์สามารถทำงานได้อย่างราบรื่น
- b)โหลดรูปภาพและสร้างวัตถุต่างๆ: คือการโหลดรูปภาพที่เกี่ยวข้อง ขึ้นมาเก็บไว้ในหน่วยความจำทั้งหมด และทำการโหลดวัตถุสามมิติที่เกี่ยวข้องขึ้นมาทั้งหมด ดังรูปที่ 9-2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 9-1 ภาพรวมของการทำงาน ของตัวเกมส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 9-2 ขั้นตอนการโหลดรูปภาพ และ วัตถุสามมิติ

จากรูปบล็อกการทำงานย่อยๆ b1 ถึง b6 นั้นจะมีการทำงานดังนี้

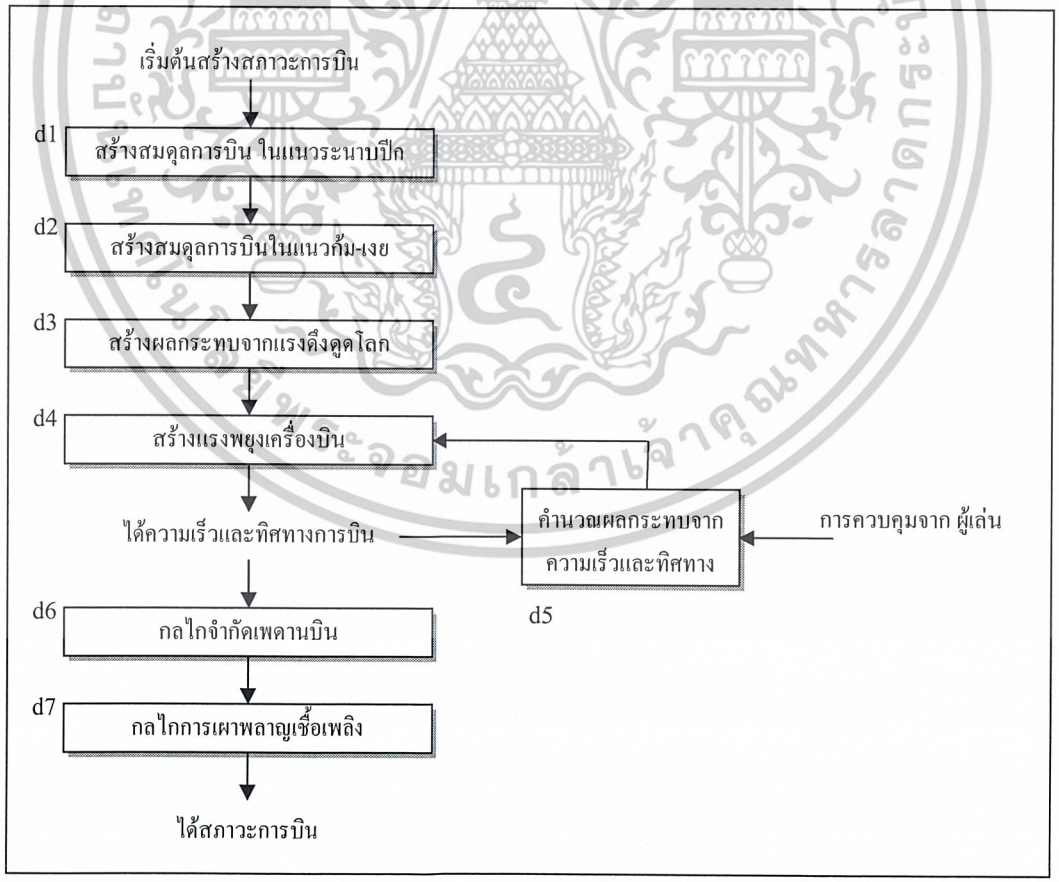
- b1)โปรแกรมจะทำการอ่านเท็กซ์ไฟล์ (Text file) ที่เก็บรายละเอียดของเวอร์เท็กซ์ ของภูเขา แล้วนำมาสร้างเป็นโพลีกอนวัตถุของภูเขา ตลอดจนสร้างพื้นผิวและใส่สีให้เหมือนภูเขาจริงเท่าที่จะทำได้
- b2) ต่อจากนั้นจะโหลดรูปภาพสองมิติ และ พื้นผิว เท็กซ์เจอร์ ที่จะใช้ประกอบกับวัตถุสามมิติขึ้นมาเก็บไว้ในหน่วยความจำ เพื่อที่จะได้ใช้งานยามต้องการ
- b3) หลังจากนั้นก็จะโหลดภาพเคลื่อนไหว ที่จะใช้ในการสร้างพาร์ติเคิล (สร้างเป็นละอองไอพ่นของเครื่องบิน และ จรวดขีปนาวุธ) โดยการโหลดข้อมูลทั้งหมดนี้ จะโหลดมาจากไฟล์ภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- b4)หลังจากทำการสร้างวัตถุที่เกี่ยวข้องทั้งหมด ก็จะทำการเชื่อมต่อวัตถุที่เกี่ยวข้องเข้าด้วยกัน เช่นเชื่อมพาร์ติเคิล ให้ติดบริเวณท้ายเครื่องบินหรือท้ายจรวดขีปนาวุธ (เพื่อแสดงเป็นไอพ่น) เป็นต้น
- b5) จากนั้น จะทำการติดตั้งกล้องเข้าไปกับตัววัตถุต่างๆ เพราะว่า การเคลื่อนไหวของกล้อง ในเกมนั้น มีมากมายหลายมุม จึงจำเป็นที่จะต้องติดตั้งกล้องในจุดต่างๆเพื่อแสดงหลากหลายอิริยาบถของการบิน
- b6) สุดท้าย ก็จะทำการปรับแต่งค่าหมอก(Fog) ให้เกิดแก่ตัวเกม โดยค่าหมอกนี้ จะทำให้ตัวเกมมีความสมจริงมากขึ้น เพราะหมอกนี้จะทำให้เกิดมิติในความลึกของตัวเกม กล่าวคือ จะทำให้วัตถุที่อยู่ไกลออกไปมองไม่ชัดเท่ากับวัตถุที่อยู่ใกล้

หลังจากที่ทำการโหลดข้อมูล และ สร้างวัตถุต่างๆครบถ้วนแล้ว ก็จะเข้าสู่กลไกของการควบคุมการเคลื่อนไหวที่เกิดขึ้นในตัวเกม

- c) เรนเดอร์ภาพ: ในกระบวนการนี้ เป็นหน้าที่ของโปรแกรมที่จะแสดงผลออกสู่หน้าจอ หลังจากที่ได้วางตำแหน่ง ของวัตถุต่างๆ ลงบนตำแหน่งต่างๆเรียบร้อย
- d) สร้างสภาวะจำลองการบิน: กลไกในการจำลองนี้ มีการทำงานภายในดังรูปที่ 9-3



รูปที่ 9-3 กลไกการจำลองสภาวะการบิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกลไกการจำลองสภาวะการบินนี้จะมีองค์ประกอบหลายอย่างที่เกี่ยวข้อง โดยถ้าดูจากรูปที่ 9-3 นี้ มีการทำงานย่อยๆ ตั้งแต่ d1 ถึง d7 ซึ่งแต่ละอันจะมีการทำงานดังนี้

- d1-d2) โดยขั้นต้น จะเป็นการสร้างสมดุลการบิน ในแนวก้ม-เงย และแนว ควงซ้าย-ขวา เพื่อให้เครื่องบิน กลับมาบินอยู่ในแนวระนาบได้โดยอัตโนมัติ
- d3) ต่อมาคือจะหาค่าผลกระทบของแรงดึงดูดโลกต่อตัวเครื่องบิน ทำให้ทิศทางเครื่องบินนั้น บินลงสู่พื้นดิน
- d4) ต่อจากนั้น ก็นำค่าความเร็วของเครื่องบิน มาหาแรงพยุงเครื่องบินเพื่อให้เกิดสภาวะลอยได้ของตัวเครื่องบิน
- d5) หลังจากได้ความเร็วและทิศทางทั้งหมดแล้ว ก็จะนำมาคำนวณหามุมก้ม-เงย และมุมการเลี้ยวหักเห ที่เครื่องบินสามารถจะเลี้ยวได้ โดยผู้ใช้ จะเป็นคนบังคับการเลี้ยว และ การก้ม-เงย ของเครื่องบินเอง
- d6) เมื่อทุกอย่างสมบูรณ์แล้ว ก็จะตรวจสอบเพดานบิน ว่าเครื่องบินนั้น มีเพดานบินสูงเท่าไร แล้วก็จะสร้างข้อจำกัด ในความสูงของการบินขึ้นมา
- d7) สุดท้ายคือนำความเร็วของเครื่องบิน มาหาอัตราการเผาผลาญเชื้อเพลิง

เมื่อถึงจุดๆนี้แล้วเราก็จะได้สภาวะการบินที่สมบูรณ์ พร้อมที่จะนำไปแสดงผลได้ หรือเมื่อเล่นผ่านเครือข่าย เราก็จะได้ผลลัพธ์ของการเคลื่อนที่ ส่งไปให้ผู้เล่นอีกคนหนึ่งเพื่อนำไปประมวลผลในเครื่องผู้เล่นคนนั้นต่อไป

- e) เมื่อจัดการกับผลกระทบต่างๆของเครื่องบินแล้ว ต่อไป จะเป็นการจัดการเกี่ยวกับเครื่องบินของศัตรู ว่าอยู่ตำแหน่งใด มีทิศทางไปทางไหน ซึ่งส่วนนี้ ตัวเกมส์จะเป็นผู้กำหนดขึ้นมาเอง แต่ถ้าในกรณีที่เล่นผ่านเครือข่าย ก็จะเป็นการรับค่าจากผู้เล่นอีกคนหนึ่งแทนการกำหนดขึ้นมาเอง
- f) การทำงานส่วนต่อไป คือการจำลองการทำงานขีปนาวุธ เมื่อมีการปล่อยขีปนาวุธออกไป โดยการควบคุมขีปนาวุธนี้ คือการทำให้ขีปนาวุธมีการโจมตีแบบติดตาม คือสามารถเลี้ยวเข้าหาศัตรูได้
- g) การทำงานขั้นต่อไปคือ วางตำแหน่งกล้อง , เนื่องเพราะว่าเครื่องบินของเราไม่ได้อยู่กับที่ และเพื่อมุมการมองที่หลากหลาย เราจึงต้องวางตำแหน่งกล้อง และ หันตำแหน่งกล้องไปยังตำแหน่งที่ต้องการ
- h) ตรวจจับการชนกันของวัตถุ: ส่วนนี้จะเป็นการตรวจสอบวัตถุทั้งหมดว่ามีการชนกันเกิดขึ้นหรือไม่ เพื่อนำไปสร้างเหตุการณ์ต่างๆ ที่เหมาะสม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

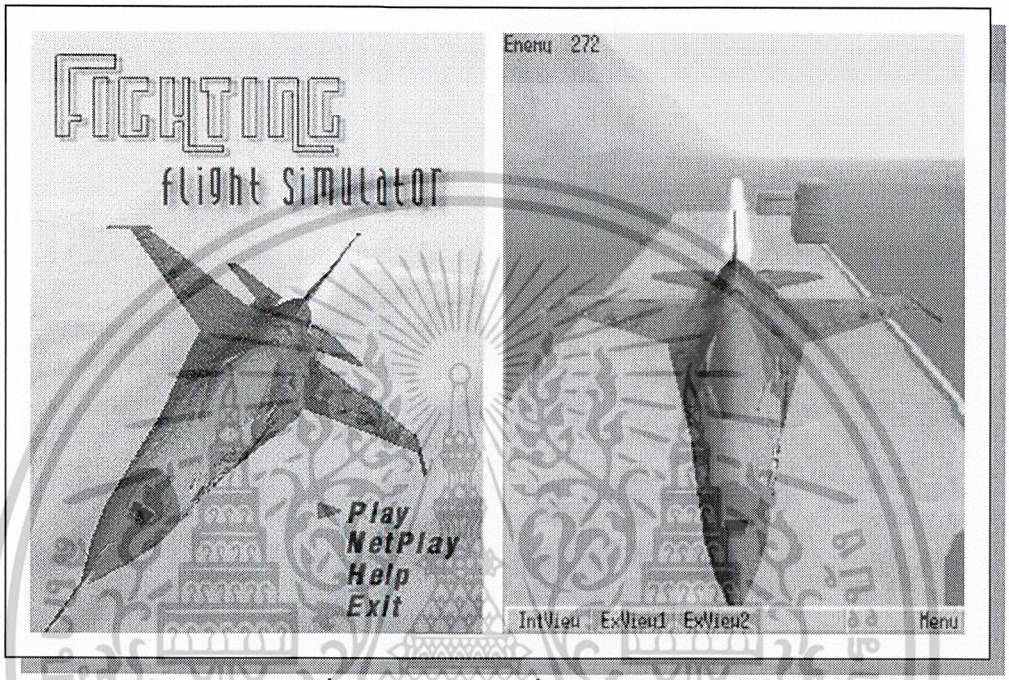
- i) สร้างระนาบ นอกภูเขา: จะเป็นการสร้างระนาบ นอกขอบเขตของภูเขา ซึ่งระนาบนี้มีขนาดกว้างใหญ่ในระดับหนึ่ง ที่จะทำให้ วัตถุภูเขา นั้นมองดูแล้วไม่อยู่อย่างโดดๆ ในโลกสามมิติเพียงวัตถุเดียว
- j) แสดงค่าต่างๆทางตัวเลข สู่หน้าจอ: สุดท้ายคือการวาดรูป แสดงค่าพารามิเตอร์ในการบินต่างๆที่สำคัญให้แค่ผู้ใช้ดู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

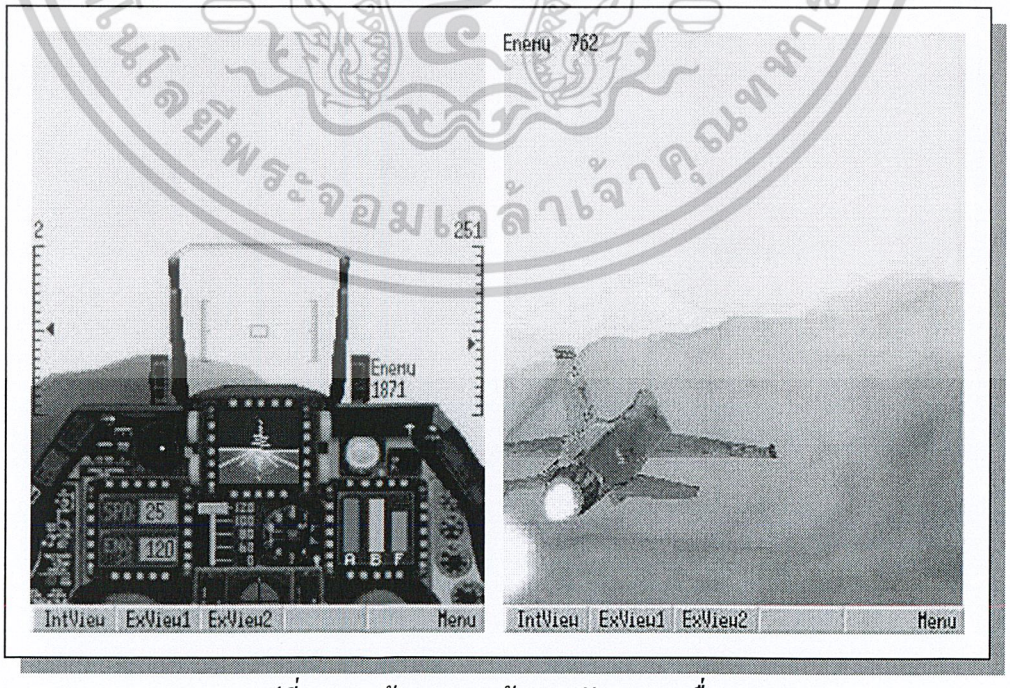
บทที่ 10

ผลการทดลอง

ตัวเกมส์ มีลักษณะรูปร่างหน้าตาเป็นดังรูปต่อไปนี้

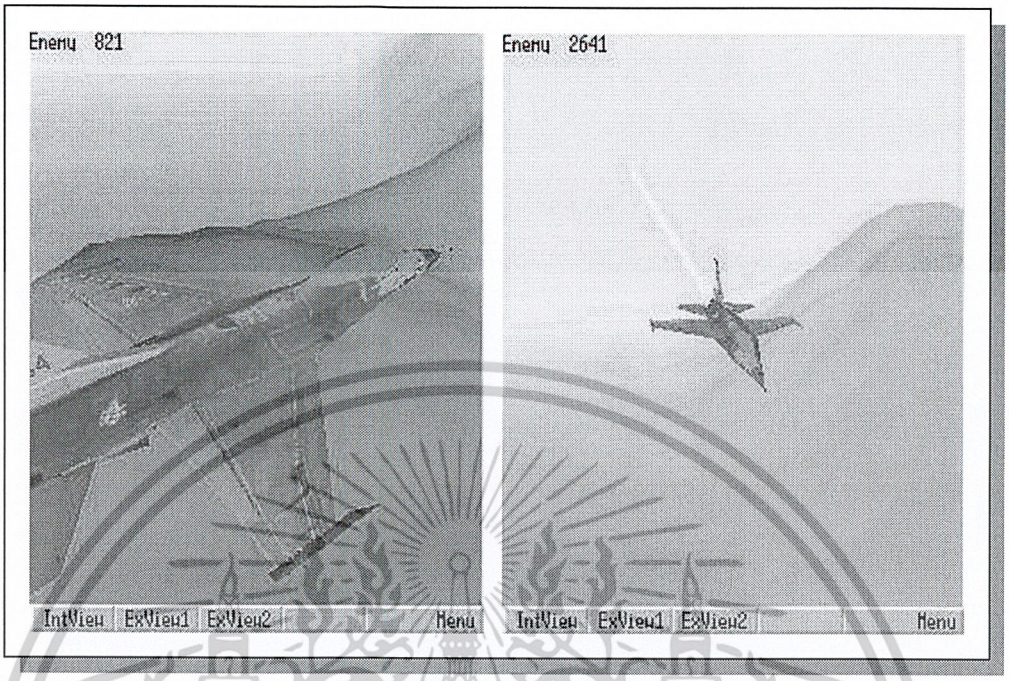


รูปที่ 10-1 หน้าจอขณะเริ่มเกมและขณะกำลังออกตัว

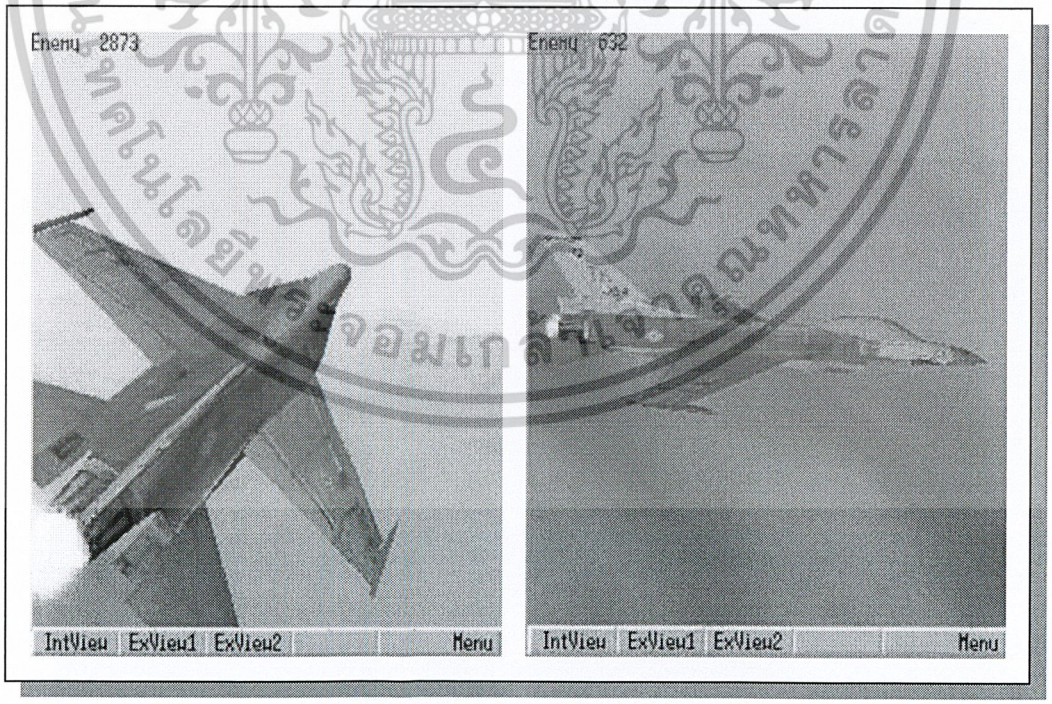


รูปที่ 10-2 หน้าจอแสดงห้องคนขับ และ เครื่องบินขณะบิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

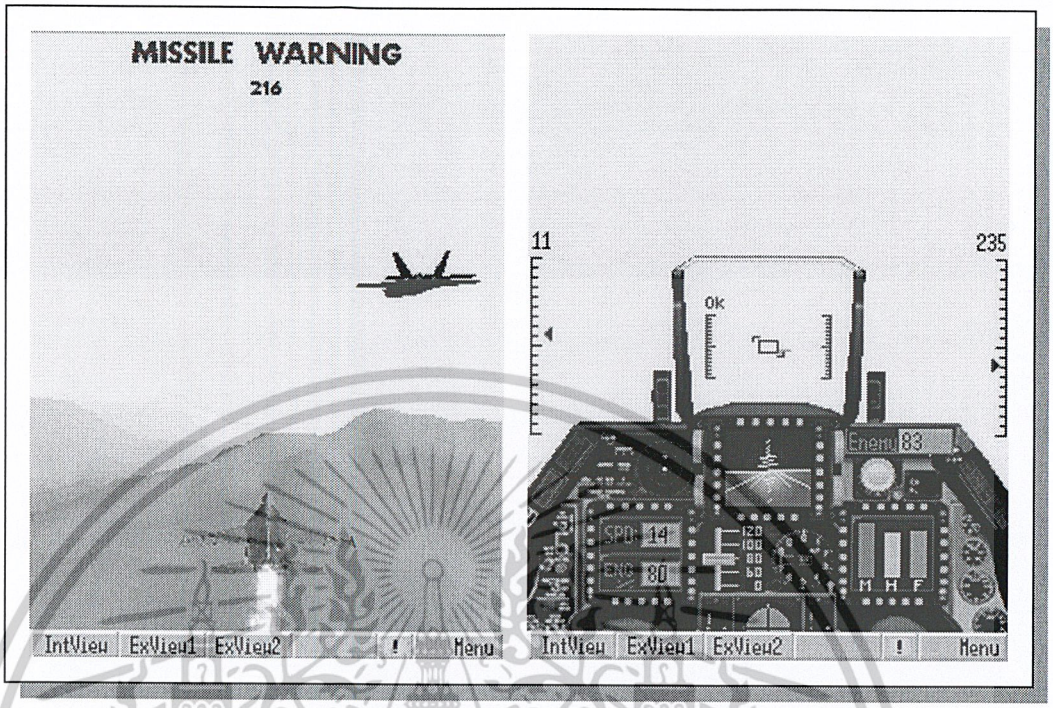


รูปที่ 10-3 เครื่องบินเมื่อมองจากภายนอก

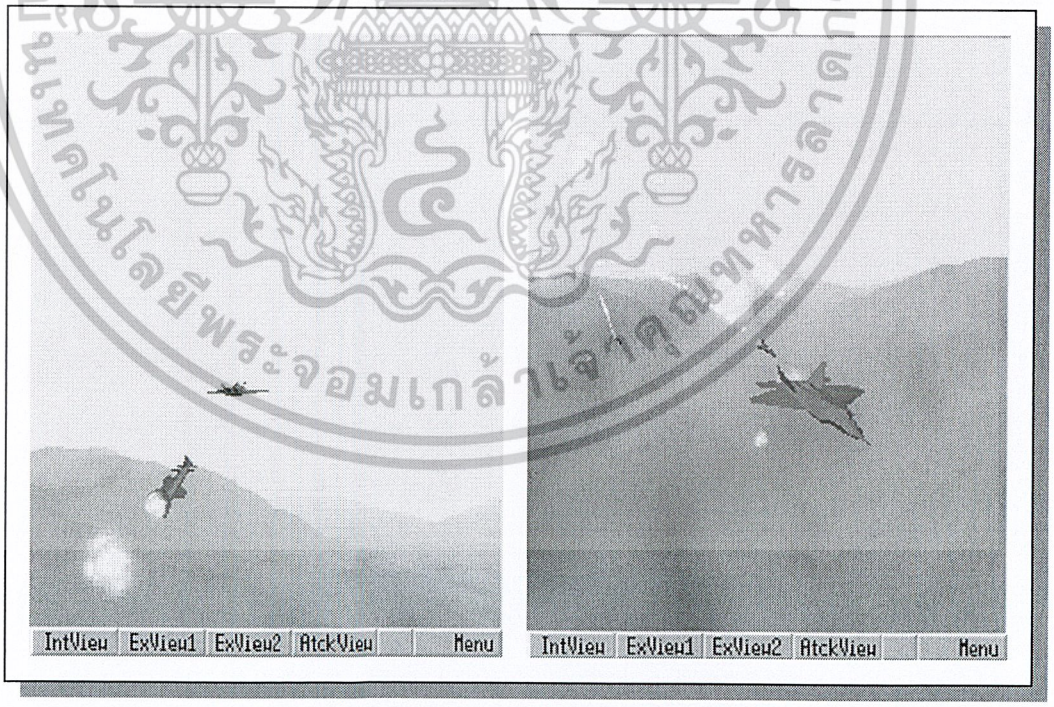


รูปที่ 10-4 เครื่องบินเมื่อมองจากด้านบนและด้านข้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

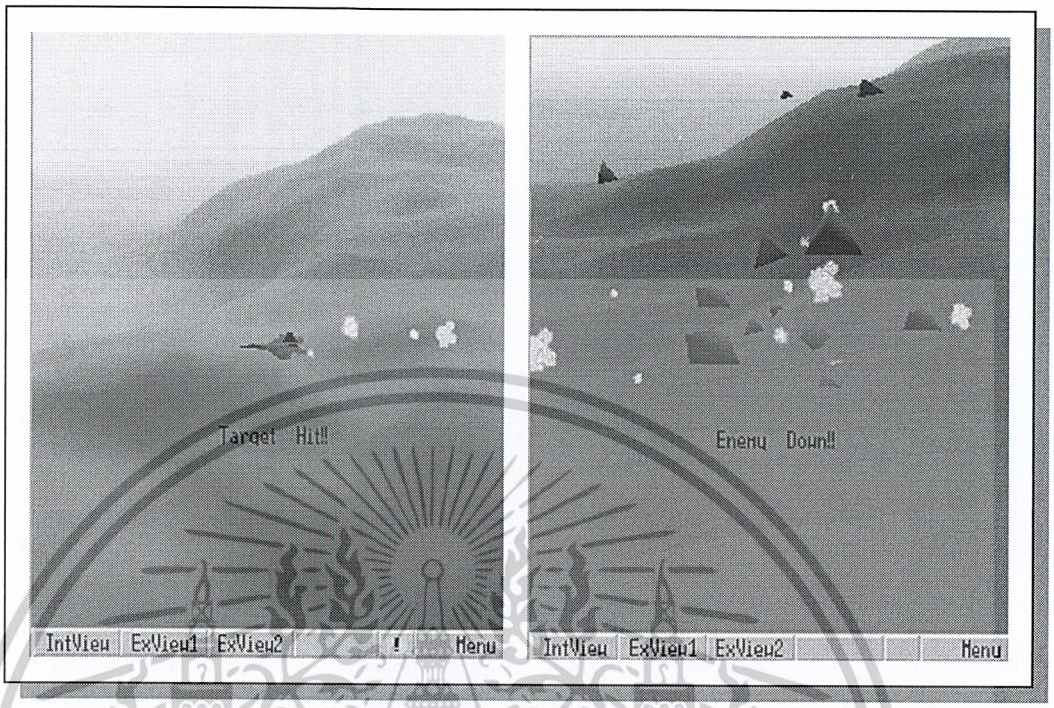


รูปที่ 10-5 พบศัตรู และ ทำการล๊อคเป้าหมายเพื่อโจมตี

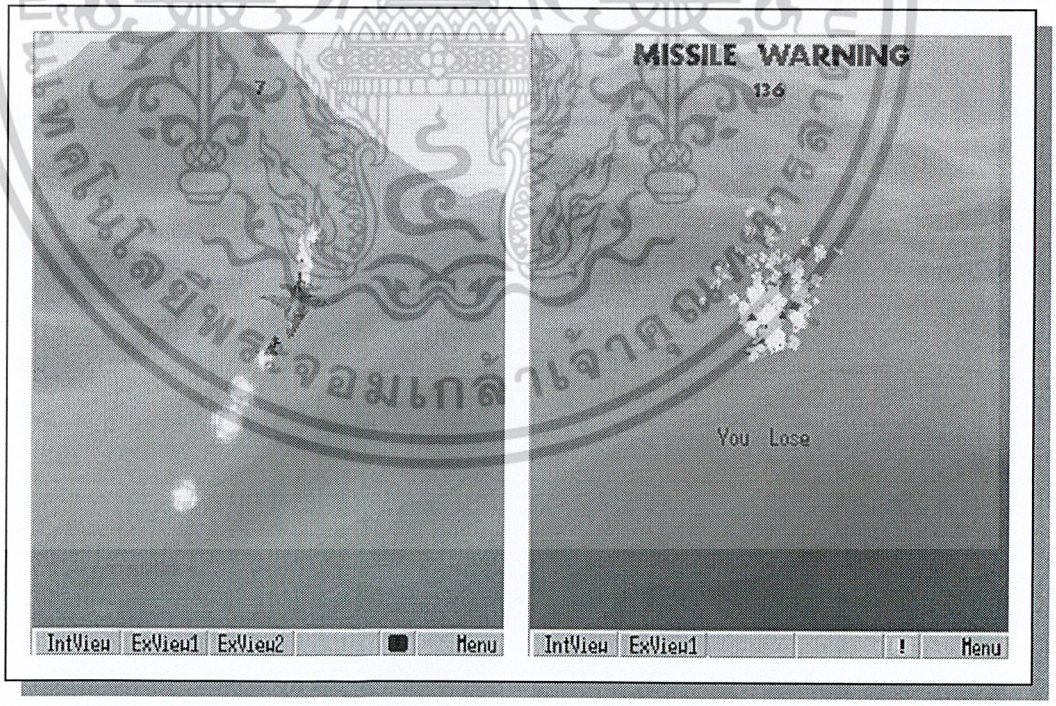


รูปที่ 10-6 ยิงขีปนาวุธ ติดตามเป้าหมาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

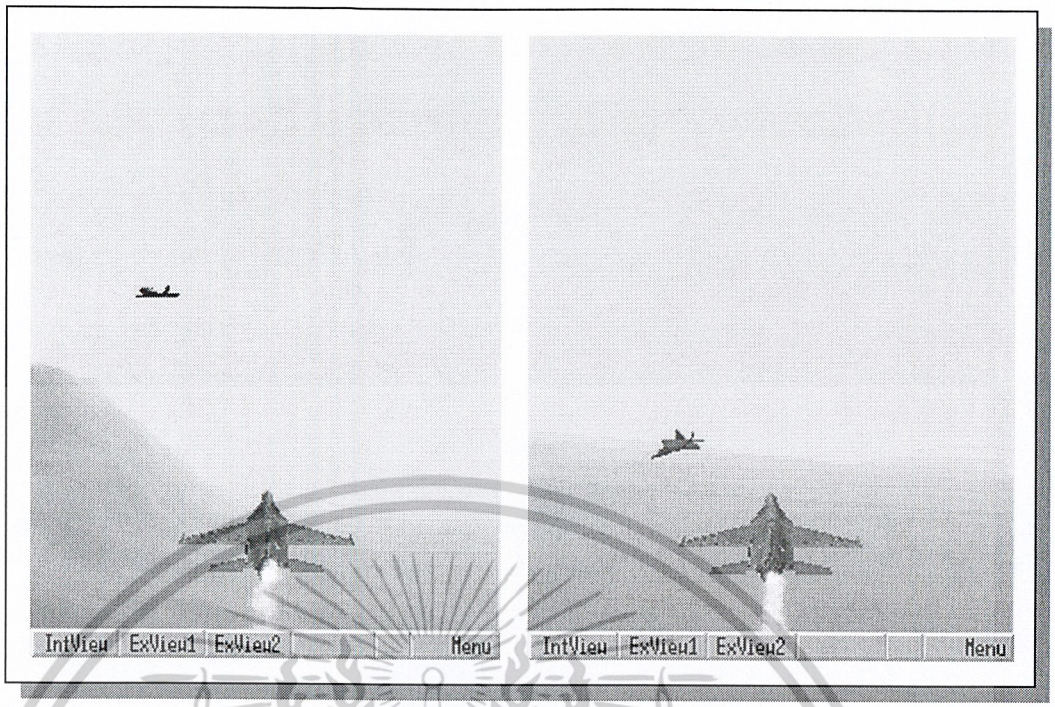


รูปที่ 10-7 ศัตรูโดนโจมตี และเสียหาย



รูปที่ 10-8 เครื่องบินถูกยิงโดยขีปนาวุธศัตรู และเครื่องบินเสียหาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 10-9 เล่นพร้อมกันสองคนผ่านเครือข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 11
บทวิจารณ์และสรุป

11.1 วิจารณ์งานวิจัย

ในโลกของเกมส์ สามมิติ นั้นเปลี่ยนไปเร็วมากเมื่อเทียบกับเกมแนวอื่นๆ ที่ไม่ค่อยเปลี่ยนแปลงกันมากเท่าไรนัก โดยความเปลี่ยนแปลงในตัวเกมส์สามมิตินั้น มักจะเปลี่ยนแปลงกันในเรื่องของฮาร์ดแวร์ มากกว่า อัลกอริทึ่มที่ใช้เสียเป็นส่วนใหญ่

ถ้าจะกล่าวถึงงานวิจัยของโครงงานนี้ ก็เปรียบเสมือนเป็นการค้นหาวิธีที่ดีที่สุดที่จะแสดงผลให้ได้สวยที่สุด ภายใต้ฮาร์ดแวร์ที่มีข้อจำกัดอย่างสูง ซึ่งทำให้การค้นหา “กรรมวิธี” และ ค้นหาอัลกอริทึ่มดีๆ ในการจัดการนั้น มักจะคิดค้น หรือ ค้นหาได้อย่างยากเย็น และบางที อาจจะยากกว่าการหา ฮาร์ดแวร์ตัวใหม่ (ที่ออกใหม่มาเรื่อยๆ และถูกลงเรื่อยๆ) ซักตัวหนึ่งก็เป็นได้

ดังนั้นจากผลการทดลอง เราจะเห็นได้ว่าโครงงานนี้ สามารถทำงานและแสดงผลตัวเกมส์ได้ใกล้เคียงกันกับตัวเกมส์บนเครื่องพีซีพอสมควรอยู่ นั่นคือ ถ้าเราต้องการพัฒนาเกมส์สามมิติให้ดีขึ้น ก็สามารถทำได้โดยไม่ยากนัก เพราะว่าเราได้เรียนรู้ที่จะไม่พึ่งฮาร์ดแวร์ มากเท่ากับการพึ่งอัลกอริทึ่มในการจัดการแทน

11.1 สรุปงานวิจัย

โครงงานนี้เป็นโครงงานซอร์ฟแวร์เพื่อความบันเทิง ประเภทเกมส์ ถึงแม้จะดูว่าเป็นอุตสาหกรรมที่มีการเจริญเติบต่อค่อนข้างต่ำในเมืองไทย แต่ก็นับว่านี่ก็ถือเป็นก้าวแรกในการพัฒนาเกมส์ 3 มิติ บนพ็อกเก็ตพีซี ในประเทศ เนื่องจากปัจจุบันนี้เกมส์จำพวกนี้ ส่วนมากจะเขียนโดยบริษัทจากต่างประเทศ ซึ่งปีหนึ่ง ๆ เราจะต้องสูญเสียเงินตราเพื่อนำไปซื้อเกมส์ที่คนไทยก็พัฒนาเองได้เหล่านี้ โดยโครงงานนี้เป็นโครงงานที่พยายามทดสอบศักยภาพของการเขียนและรันเกมสามมิติ ตลอดจนใช้ความสามารถในเรื่องเครือข่าย บนเครื่องพ็อกเก็ตพีซี โดยชี้ให้เห็นถึงความสามารถในการใช้พ็อกเก็ตพีซี นำมาเล่นเกมส์ที่สนุกและสวยงามได้โดยไม่แพ้เครื่องพีซีเลย ซึ่งอาจจะกระตุ้นให้เกิดการพัฒนาเกมส์ บน พ็อกเก็ตพีซีโดยคนไทยและอุตสาหกรรมเกมส์ในประเทศมากขึ้น จนบางทีอาจจะสามารถเทียบชั้นกับของต่างประเทศได้ ซึ่งเมื่อถึงจุดนั้นแล้ว “เกมส์” อาจเป็นสินค้าส่งออกที่สำคัญของประเทศก็เป็นได้ ดังนั้นโครงงานนี้จึงเป็นเสมือนจุดเริ่มต้นของการพัฒนาเกมส์ ที่พัฒนาโดยใช้หยิบเอาเทคโนโลยีที่คนมองข้ามหรือไม่ได้ใส่ใจ มาสร้างสรรค์ในมุมมองใหม่ๆ เพื่อค้นหาความเป็นไปได้ในเชิงอุตสาหกรรม เหมือนกันกับการเอาเกมส์สามมิติ มาพัฒนาลงบนเครื่องพ็อกเก็ตพีซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11.3 ปัญหาและการแก้ไข

ปัญหาที่พบ และเป็นปัญหาหลัก อันดับแรก ก็คือ ทรัพยากรอันจำกัดของเครื่องพ็อกเก็ตพีซี ที่มีอยู่อย่างจำกัด ไม่ว่าจะเป็นความสามารถในการคำนวณ, เนื้อที่หน่วยความจำ ฯลฯ ที่ทำให้การพัฒนา เกมส์ที่ใช้กราฟิกสามมิตินั้น ต้องมีขีดจำกัด ในการแสดงผล ซึ่งทำให้เราต้องใช้กรรมวิธีที่แตกต่างกันไป ในการชดเชยข้อจำกัดเหล่านั้น เพื่อทำให้ได้ภาพที่มีข้อบกพร่องน้อยที่สุด นอกจากนี้ ปัญหารองลงมาก็คือ ขีดจำกัดของอุปกรณ์อินพุต เอาท์พุต ที่มีขนาดเล็ก ตัวอย่างคือ ตัวพ็อกเก็ตพีซี นั้นไม่มีคีย์บอร์ด มีเพียงคีย์แพดเล็กๆให้กด บวกกับ สไตล์ลัส (Stylus) อันหนึ่ง แทนการคลิ๊กซ้ายของเมาส์ โดยไม่มีปุ่มคลิ๊กขวาอีกต่างหาก ซึ่งจำเป็นที่จะต้องออกแบบเกม ให้เหมาะสมกับข้อจำกัดที่ว่านี้ และปัญหาอื่นๆที่พบก็คือเรื่องของการเชื่อมต่อเครือข่าย ที่พัฒนาได้ยาก บนตัว จำลองพ็อกเก็ตพีซี

การแก้ไขในแนวทางของเรานั้น เราแก้ไขปัญหาเรื่องทรัยากรอันจำกัด ด้วยการใช้ลูกเล่น เชิงจิตวิทยา มาชดเชยความไม่สมจริงของภาพกราฟิกสามมิติแทน ซึ่งการใช้ลูกเล่นเชิงจิตวิทยานี้เป็นเรื่องที่พบกันได้บ่อยในการเขียนเกมส์สามมิติอยู่แล้ว ตัวอย่างเช่น การใช้พื้นผิว เทกซ์เจอร์ แทนการใช้วัตถุ โพลีกอนชนิดนั้นจริงๆ (เช่นการเขียนกำแพงอิฐบล็อก ที่ไม่จำเป็นต้องสร้างโพลีกอนที่มีลักษณะรูปทรงอิฐ มาต่อๆกันให้เสียเวลา แต่กลับใช้การแปะ เทกซ์เจอร์ รูปอิฐ เข้าไปบนพนังเรียบๆแทน เป็นต้น

ส่วนปัญหาเรื่องขีดจำกัดของอุปกรณ์ ที่จะใช้ในการควบคุมเกมนั้น เรามีทางเลือกสองทาง นั่นคือ ทางเลือกแรก เราจะสร้างอุปกรณ์ควบคุมเกมจากภายนอก เช่น จอยสติ๊ก ฯลฯ และอีกแนวทางคือ ออกแบบให้โปรแกรม มีความเหมาะสมกับอุปกรณ์ อินพุตเอาท์พุตที่มีอยู่ เช่นไม่ออกแบบให้ต้องคลิ๊กหน้าจอเกมบ่อยๆหรือแรงๆ เพราะว่าอาจก่อให้เกิดความเสียหายแก่หน้าจอพ็อกเก็ตพีซีได้ เป็นต้น ซึ่งการพัฒนาเกมในโครงงานนี้ เลือกที่จะใช้วิธีหลัง นั่นคือใช้อุปกรณ์ อินพุต เอาท์พุตพื้นฐานที่มีอยู่ แต่จะออกแบบให้ลดจำนวนการกด ให้น้อยที่สุด

ปัญหาทางด้านเครือข่ายที่พบก็คือ ไม่สามารถติดต่อเครือข่ายจากตัว จำลองพ็อกเก็ตพีซีได้ ซึ่งวิธีแก้ไขของเรา คือ สร้างการติดต่อแบบปรกติๆ ด้วพีซี จนมั่นใจว่าใช้งานได้จริง จึงค่อยนำไปใส่เข้าไปในโค้ดของตัวเกมส์

11.4 แนวทางในการพัฒนาต่อไป

จากความสามารถของตัวเกมส์ที่ได้กล่าวมาแล้ว จะเห็นได้ว่าโปรแกรมเกมส์นี้ยังมีสิ่งที่ต้องพัฒนาอีกมากเพื่อให้เกมส์มีความสมบูรณ์และตรงกับความต้องการของผู้เล่นเกมส์ สิ่งที่ควรพัฒนามีดังนี้

1. เพิ่มความสามารถของกราฟิก (ถ้าความสามารถของ พ็อกเก็ตพีซี ในยุคต่อไปมีความสามารถเพิ่มมากขึ้น)
2. พัฒนาเกมส์ให้สามารถเล่นได้มากกว่า 2 คนในระบบเน็ทเวิร์ค ตลอดจนสร้างการเชื่อมต่อในรูปแบบอื่นๆ นอกจากการเล่นผ่าน แลน เช่นทำให้เล่นได้ผ่าน บลูธูท (Blue tooth)เป็นต้น
3. เพิ่มความสมจริงให้เกมส์โดยเพิ่มความสามารถในตัวโมเดลพลศาสตร์การบิน ให้มากขึ้นกว่าเดิม (Flight dynamics model)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เพิ่มความละเอียดของตัวเกมส์ เช่น รูปแบบเครื่องบิน , อาวุธที่ใช้เล่นในเกมส์ หรือ แผนที่ให้มากขึ้น ตลอดจนประยุกต์ใช้กฎใหม่ในการเล่น เพื่อสร้างความแปลกใหม่แก่ตัวเกมเป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

การติดตั้ง Embedded Visual C++ และ PocketPc Emulator

ขั้นตอนการติดตั้ง

ในขั้นตอนการติดตั้ง Embedded Visual C++ เพื่อใช้ในการเขียนโปรแกรมบน PocketPc และ ตัว Emulator ของ PocketPc จะต้องทำตามขึ้นตอนดังต่อไปนี้

1. จะต้องทำการดาวน์โหลดโปรแกรมทั้งหมดจาก
   http://www.microsoft.com/windowsmobile/resources/downloads/developer/default.mspx
   ซึ่งจะประกอบไปด้วย
   - eMbedded Visual C++ 4.0 เพื่อใช้ในการเขียนโปรแกรมบน PocketPc
   - eMbedded Visual C++ 4.0 Service Pack 2 ตัวอัพเดต eMbedded Visual C++ 4.0
   - Microsoft ActiveSync 3.71 เพื่อใช้ในการเขียนโปรแกรมติดต่อ PocketPc จริง ๆ
   - Microsoft Pocket PC 2003 SDK เป็น Emulator ของ PocketPc
2. ต่อไปจะเป็นการเริ่มขึ้นการการติดตั้งโปรแกรมต่าง ๆ ที่ดาวน์โหลดมา โดยขึ้นแรกให้ uninstall ตัวโปรแกรม eMbedded Visual C++ 3.0 หรือ Emulator version ที่เก่ากว่า version 2003 ออกก่อน
3. โปรแกรมแรกที่เราจะต้องทำการติดตั้ง คือ โปรแกรม Microsoft ActiveSync 3.71 ซึ่งถ้าเรามี Visual Studio .NET 2003 อยู่แล้ว เราต้องทำการ install ใหม่หรือ repair มัน โดยจะเป็นโปรแกรมที่ใช้เชื่อมต่อ Pc กับ PocketPc จริง ๆ และสามารถจำลองการเชื่อมกับตัว Emulator ของ PocketPc โดยต้อง install ก่อน The Pocket PC 2003 SDK เสมอ
4. ต่อไปทำการติดตั้ง โปรแกรมที่ใช้เขียนโปรแกรม บน PocketPc ซึ่งก็คือ eMbedded Visual C++ 4.0 และ ตัว อัพเดตของมัน ซึ่งเป็น Service Pack 2 โดยจะต้องทำการติดตั้งก่อน The Pocket PC 2003 SDK เสมอ

   Note: ถ้าจะทำการติดตั้งบน Windows Server 2003 อาจทำให้มี ไดอะล็อคบ็อค แจ้งขึ้นมาว่า Emulator driver ไม่สามารถมองเห็นได้ เราควรทำการยอมรับว่า มองไม่เห็น และ ดำเนินขั้นตอนต่อไป
5. ทำการติดตั้ง Pocket PC 2003 SDK เป็นอันจบขั้นตอนสุดท้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] นิรุท อำนวยศิลป์, "คู่มือการเขียนโปรแกรม Microsoft Visual C++ Version 6.0 ฉบับเพื่อการใช้งานจริง ", พิมพ์ครั้งที่ 4, กรุงเทพ: บริษัท ซัคเซส มีเดีย จำกัด, 2543

[2] ยุทธนา ลีลาศวัฒนกุล, "คู่มือการเขียนโปรแกรม Visual C++ 6.0 ฉบับโปรแกรมเมอร์", กรุงเทพ: อินโฟเพรส, 2544

[3] พรพล สาครินทร์ม, "พื้นฐานการก้าวสู่โลก 3 มิติ 3D Graphics", กรุงเทพ: บริษัท ซัคเซส มีเดีย จำกัด

[4] กฤษฎา แก้วมณี, "เข้าใจหลักการสร้างงาน 3 มิติระดับมืออาชีพ",กรุงเทพ: บริษัท ซัคเซส มีเดีย จำกัด

[5] Andre LaMothe (1999): "Tricks of the Windows Game Programming Gurus", Sams, 9 1999.

[6] Mark DeLoura (2000): "Game Programming Gems", Charles River Media, 8 2000.

[7] Mark DeLoura (2001): "Game Programming Gems 2", Charles River Media, 8 2001.

[8] "DieselEngine" , http://www.3darts.fi/mobile/de.htm

[9] "PocketPc Programming", http://miketeo.net/links/pocketpc.html

[10] "MSDN Library", http://msdn.microsoft.com/library

[11] "Download PocketPc Emulator and Embedded Visual C++", http://www.microsoft.com/windowsmobile/resources/downloads/developer/default.mspx

[12] "Tutorial Source code", http://www.codeproject.com/ce/

[13] "Pocket PC Developer Network", http://www.pocketpcdn.com/

[14] "GAME PROGRAMMING TUTOR FOR POCKET PC", http://www.pocket-g.com/prx/buildtools.html

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้