ปีการศึกษา 2530

ปริญญานิพนธ์ เรื่อง

การสร้างภาพ 3 มิติจากภาพ 2 มิติ

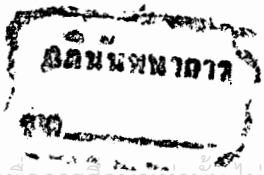
โดย

นาย พัฒน์พงษ์ มาศวงษ์ปกรณ์

อาจารย์ที่ปรึกษา

ดร. วัลลภ สุระกำพลธร

ดร. ฟูศักดิ์ ชีวะสุวิทย์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2530

เรื่อง การสร้างภาพสามมิติจากภาพสองมิติ

ผู้จัดทำ นาย พัฒน์นพงษ์ มาศวงศ์ปกรณ์

..... อาจารย์ที่ปรึกษา

(ผศ.ดร.วัลลภ สุระกำพลธร)

..... อาจารย์ที่ปรึกษา

(ผศ.ดร.ฟูศักดิ์ ชีวสุวิทย์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GENERATING 3-DIMENSION IMAGE FROM 2-DIMENSION IMAGE

Dr.Wanlop Surakampoltorn

Dr.Fusak Chevasuvit

1987

Abstract

The method to illustrate geographic surface by using contouring map is the only way that we show the real geographicsurface (3-dimension) in the form of two dimension because contouring map shows the surface only on top view. This aspect of contouring map makes small detail and uncleariness when we compare to 3-dimension geographic surface.

This thesis shows the way to generate 3-dimension geographic surface from 2-dimension geographic surface. The process to generrate the 3-D surface can be made in several steps:

Step 1 : Define regular grid on the map.

Step 2 : Read data from step 1.

Step 3 : Regenerate grid point by interpolation method.

Step 4 : Define viewpoint and generate 3-D surface.

and Step 5 : Hidden surface process.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสร้างภาพ 3 มิติจากภาพ 2 มิติ

พัฒน์พงษ์ มาศวงษ์ปกรณ์

อาจารย์ที่ปรึกษา

ดร.วัลลภ สุระกำพลธร

ดร.ฟูศักดิ์ ชีวะสุวิทย์

ปีการศึกษา 2530

บทคัดย่อ

การแสดงภาพของพื้นผิวทางภูมิศาสตร์โดยใช้แผนที่แสดงเส้นระดับ ความสูงหรือที่เราเรียกค่า คอนทัวร์แมป (contour map) นั้นเป็นเพียงการแสดงภาพ 3 มิติ (3-dimension) ในเชิง 2 มิติ (2-dimension) เท่านั้น เนื่องจากคอนทัวร์แมปเป็นภาพที่เกิดจากการมองพื้นผิวทางภูมิศาสตร์จาทางด้านบนลงมา (top view) จากลักษณะเช่นนี้ทำให้คอนทัวร์แมปไม่สามารถแสดงรายละเอียดหรือพื้นผิวภูมิศาสตร์นั้นได้อย่างชัดเจน เท่ากับภาพพื้นผิวภูมิศาสตร์ 3 มิติ จริงๆ

วิทยานิพนธ์นี้ได้นำแผนคอนทัวร์แมปซึ่งเป็นภาพเชิง 2 มิติมาสร้างเป็นภาพภูมิศาสตร์ 3 มิติ เพื่อที่ทำให้สามารถพิจารณาหรือเข้าใจถึงสภาพทางภูมิศาสตร์ได้ชัดเจนมากยิ่งขึ้น ขบวนการสร้างภาพเริ่มจาก การกำหนดตารางกริด (grid) ลงบนภาพคอนทัวร์แมป การอ่านข้อมูลบนเส้นตาราง (read data) การประมาณค่าบนจุดตัดของตารางกริด (regeneration griding point by interpolation) การกำหนดจุดมอง (viewpoint) การสร้างภาพ 3 มิติ (3-dimension generation) และการกำหนดพื้นผิวปิดบัง (hidden surface)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบาญ

บทที่ 1	บทนำ	1
บทที่ 2	หลักการพื้นฐานทางคอมพิวเตอร์กราฟิค	6
	การแปลงจุดและภาพใน 2 มิติ	6
	ระบบคู่ลำดับเสมือน (homogenous coordinate system)	11
	วิวพอร์ทของกราฟิค 2 มิติ (viewport of 2-D graphic)	11
	การแปลงจุดและวัตถุใน 3 มิติ	12
	ระบบแกนและระบบคู่ลำดับใน 3 มิติ	18
	การมองในระบบ 3 มิติ (viewing in 3-D)	26
	การทดสอบการตัดกันของเส้นตรง	31
	การทดสอบหาตำแหน่งของจุดในรูปหลายเหลี่ยม	33
บทที่ 3	แนวความคิดพื้นฐานสำหรับระบบ 3 มิติทางภูมิศาสตร์	36
	เส้นชั้นความสูง (contour line)	36
	วิธีการทำเส้นชั้นความสูงแบบวิธีอ้อม	42
	การหาจุดกริดจากเส้นชั้นความสูงด้วยวิธีการประมาณทาง คณิตศาสตร์	45
	สมการระนาบ 3 มิติ	53
	การหาระยะทางระหว่างจุดในระบบ 3 มิติ	54
	การระบายสีพื้นที่ที่กำหนด (seedfill algorithm)	54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4 แนวความคิดในการสร้างภาพ 3 มิติ 57
ขั้นตอนการทำงาน 58
บทที่ 5 การทดสอบ 62

บทที่ 6 สรุปผล 70

กิตติกรรมประกาศ 72

บรรณานุกรม 73

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

บทที่ 1

บทนำ

คอมพิวเตอร์กราฟฟิค (computer graphic) สามารถที่จะให้คำนิยามได้ว่าเป็นวิชาที่ใช้ในการสร้างและควบคุมภาพโดยใช้เครื่องคอมพิวเตอร์ จากคำนิยามพบว่า สามารถที่จะนำวิชาคอมพิวเตอร์กราฟฟิคไปใช้งานได้อย่างมากมาย คอมพิวเตอร์กราฟฟิคได้มีส่วนช่วยเพิ่มประสิทธิภาพในการแสดงข้อมูล ความสามารถในการแทนข้อมูลในรูปของภาพ (graphic image) ทั้งแบบลายเส้นและแบบเหมือนจริง (realistic image) ทำให้เราสามารถสื่อสารและเข้าใจเกี่ยวกับข้อมูลนั้นๆเพิ่มมากขึ้น จากเหตุผลดังข้างต้น ทำให้มีการนำคอมพิวเตอร์กราฟฟิคไปใช้งานด้านต่างๆ เช่น การแปลงข้อมูลจำนวนมากให้เป็นรูปแผนภูมิแสดงข้อมูลแบบต่างๆ เช่น บาร์ชาต์ (bar chart) ไพร์ชาต์ (piechart) และในรูปของกราฟ การนำคอมพิวเตอร์กราฟฟิคไปใช้แสดงการจัดรูปโมเลกุล การนำไปใช้ในการออกแบบวัตถุ อุปกรณ์หรือเครื่องมือต่างๆ การจำลองภูมิศาสตร์ (ดังรูปที่ 1) เป็นต้น

จากคุณประโยชน์อย่างมากมาย ทำให้รู้สึกสงสัยว่าทำไมวิชาการแขนงนี้จึงพึ่งมาเจริญอย่างมากในช่วง 10 ปีที่ผ่านมา ทั้งๆที่การพัฒนาทางด้านคอมพิวเตอร์มีมาประมาณ 30 ปีแล้ว เหตุผลที่สำคัญที่มีผลให้วิชาการทางด้านนี้มาเจริญอย่างมากในระยะหลัง เนื่องจากสาเหตุสำคัญ 2 ประการคือ

1) ปัญหาเรื่องการแบ่งเวลาของเครื่องคอมพิวเตอร์ขนาดใหญ่ (time sharing) (ในช่วงทศวรรษ 1970) ไม่เหมาะสมกับงานทางด้านกราฟฟิค เนื่องจากงานทางด้านกราฟฟิคมักจะเกี่ยวข้องกับข้อมูลจำนวนมาก รวมทั้งการคำนวณโดยบางครั้งต้องใช้สมการทางคณิตศาสตร์ที่ซับซ้อน ทำให้เครื่องคอมพิวเตอร์ต้องเสียเวลาในการทำงานมาก ทำให้ผู้ใช้อื่นๆไม่สามารถที่จะทำงานได้หรือต้องเสียเวลานานในการรอ

2) ระบบทางด้านกราฟฟิคยังมีอยู่น้อย และมีราคาแพงมาก ทำให้มีผู้นำมาใช้ในงานด้านต่างๆมีน้อย แต่ในช่วงทศวรรษ 1980 ได้มีการเสนอเครื่องคอมพิวเตอร์ขนาดเล็ก (microcomputer workstation) ที่มีลักษณะการทำงานโดยมีผู้ใช้เพียงคนเดียว ประกอบกับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ราคาที่ถูกลงของอุปกรณ์แห่งความจำ (memory chips IC) และตัวประมวลผลกลาง (center processing unit) รวมทั้งความสามารถในการทำงานมีประสิทธิภาพมากยิ่งขึ้น มีผลทำให้ความเจริญก้าวหน้าทางด้านคอมพิวเตอร์กราฟิคอย่างมากในระยะหลัง

ไมโครคอมพิวเตอร์กราฟิคถึงแม้จะมีใช้อย่างมากในหลายด้าน เช่น ทางด้านอุตสาหกรรม ธุรกิจ ภาครัฐบาลและทางด้านการศึกษา เราก็พอที่จะจัดแบ่งการนำไปใช้งานได้ออกเป็น 2 ระบบ คือ

1) การจัดแบ่งตามหน้าที่ที่ปรากฏ (visualization function) ได้แก่

-การแสดงที่ได้จากคำนวณที่ปรากฏออกมา ในลักษณะของกราฟิค

-การใช้งานทางด้านกราฟิคเสมือนเป็นตัวกลางของการวาดภาพ (drafting medium)

-การใช้กราฟิคแทนข้อมูลจำนวนมากๆ

-การใช้กราฟิคเพื่อดูการเปลี่ยนแปลง

-การแสดงภาพที่เป็นแนวความคิดหรือจินตนาการ

-การนำไปช่วยในการแปลข้อมูล

-การนำไปใช้แสดงภาพในการติดต่อระหว่างคนกับเครื่องจักร แทนการใช้การติดต่อแบบตัวหนังสือ (alphanumeric communication)

-การจำลองและตรวจสอบขบวนการก่อนการนำไปใช้จริง

-การจำลองภาพจริง (simulation of real world screen)

-ใช้ในการจำลองรูปแบบ (model) ทางทฤษฎี

-การสร้างงานทางด้านศิลปะ (artist design)

-การบันเทิง (entertainment)

2) การจัดแบบตามรูปแบบการประยุกต์ใช้ (application area)

จากการแบ่งลักษณะนี้ พบว่า การประยุกต์ใช้งานทางด้านกราฟิคส่วนมาก มีประกอบด้วยหน้าที่ปรากฏ (visualization function) มากกว่า 1 อย่างเสมอ การแบ่งตามลักษณะรูปแบบการนำไปใช้ ได้แก่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-การใช้งานกราฟฟิคทางด้านวิทยาศาสตร์และสถิติ (scientific and statistical graphics) ได้แก่ การนำไปใช้เกี่ยวกับการแทนข้อมูลจำนวนมาก เช่น บาร์ชาต (bar chart) ไพร์ชาต (pie chart) ทั้งในรูป 2 มิติและ 3 มิติ ในรูปของฮีตโตรแกรม (histrogram) และการหากแนวโน้มของข้อมูล เป็นต้น

-ด้านการสั่งงานและการควบคุม (command and control) ซึ่งได้แก่ การใช้งานแสดงผลแบบเวลาจริง (real time) ทางด้านการดำเนินงานทางอุตสาหกรรม ขบวนการทางเคมี หรือระบบการส่งจ่ายพลังงาน ซึ่งงานทางด้านกราฟฟิคสามารถแสดงผลของข้อมูลที่ได้และช่วยในการวิเคราะห์ข้อมูลเหล่านั้น

-ทางด้านช่วงการออกแบบ (computer aided design (cad)) คอมพิวเตอร์กราฟฟิคจะทำหน้าที่แทนกระดาษในการวาดลายเส้นเพื่อใช้ในการออกแบบ ทำให้เกิดความสะดวกอย่างมาก งานที่นิยมใช้ได้แก่ งานที่ด้านสถาปัตยกรรม วิศวกรรม และทางด้านโครงสร้าง (structure) เป็นต้น

-ทางด้านการกำหนดโครงร่าง (topological definition) ได้แก่ การทำโฟลต์ชาตโปรแกรม (flowchart program) การสร้างไดอะแกรมของขบวนการทางฟิสิกค์หรือเคมี การแทนสัญลักษณ์ของอุปกร์ทางไฟฟ้า เป็นต้น

-ทางด้านการออกแบบทางเรขาคณิต (geometric design) โดยเฉพาะอย่างยิ่งช่วยในด้านการออกแบบในเชิง 3 มิติ และยังช่วยให้ผู้ใช้สามารถ หมุนวัตถุที่ออกแบบไปตามมุมมองต่างๆ (viewpoint) อีกด้วย ซึ่งการประยุกต์ใช้ทางด้านนี้ จะทำการแสดงโครงสร้าง พื้นผิวโค้ง หรือแม้กระทั่งอุปกรณ์ทางกลชิ้นเล็กๆ ปกติแล้วการใช้งานทางด้านนี้มักต้องการระบบคอมพิวเตอร์กราฟฟิคที่สมบูรณ์ และมีความเร็วสูงในการแสดงภาพ ดังนั้นสำหรับการประยุกต์ใช้งานทางด้านนี้กับเครื่องไมโครคอมพิวเตอร์จึงค่อนข้างจำกัดมาก

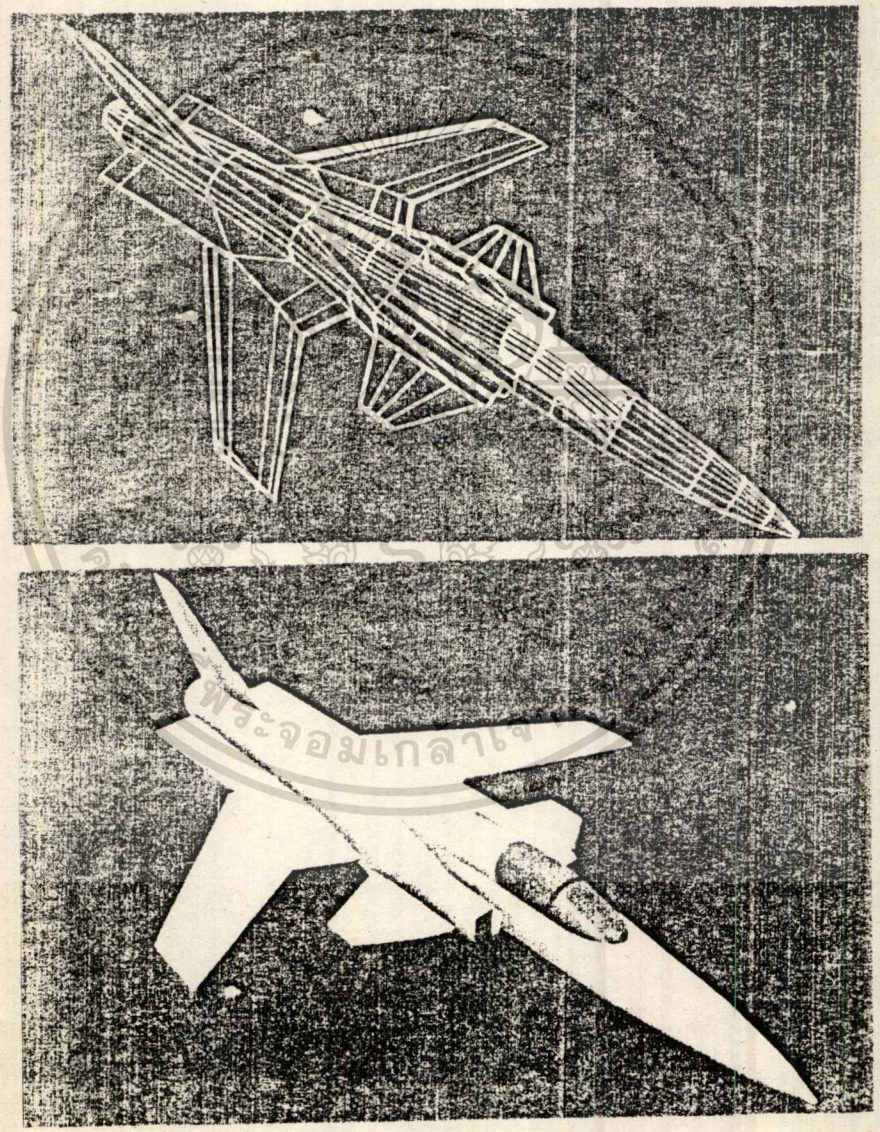
-การติดต่อสื่อสารด้วยรูปภาพ (pictorial communication) ได้แก่ การส่งภาพข้อมูลจากที่แห่งหนึ่งไปยังอีกแห่งหนึ่ง เป็นต้น

-ทางด้านศิลปะและภาพการ์ตูน (graphic arts and animation) พบว่ามีการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

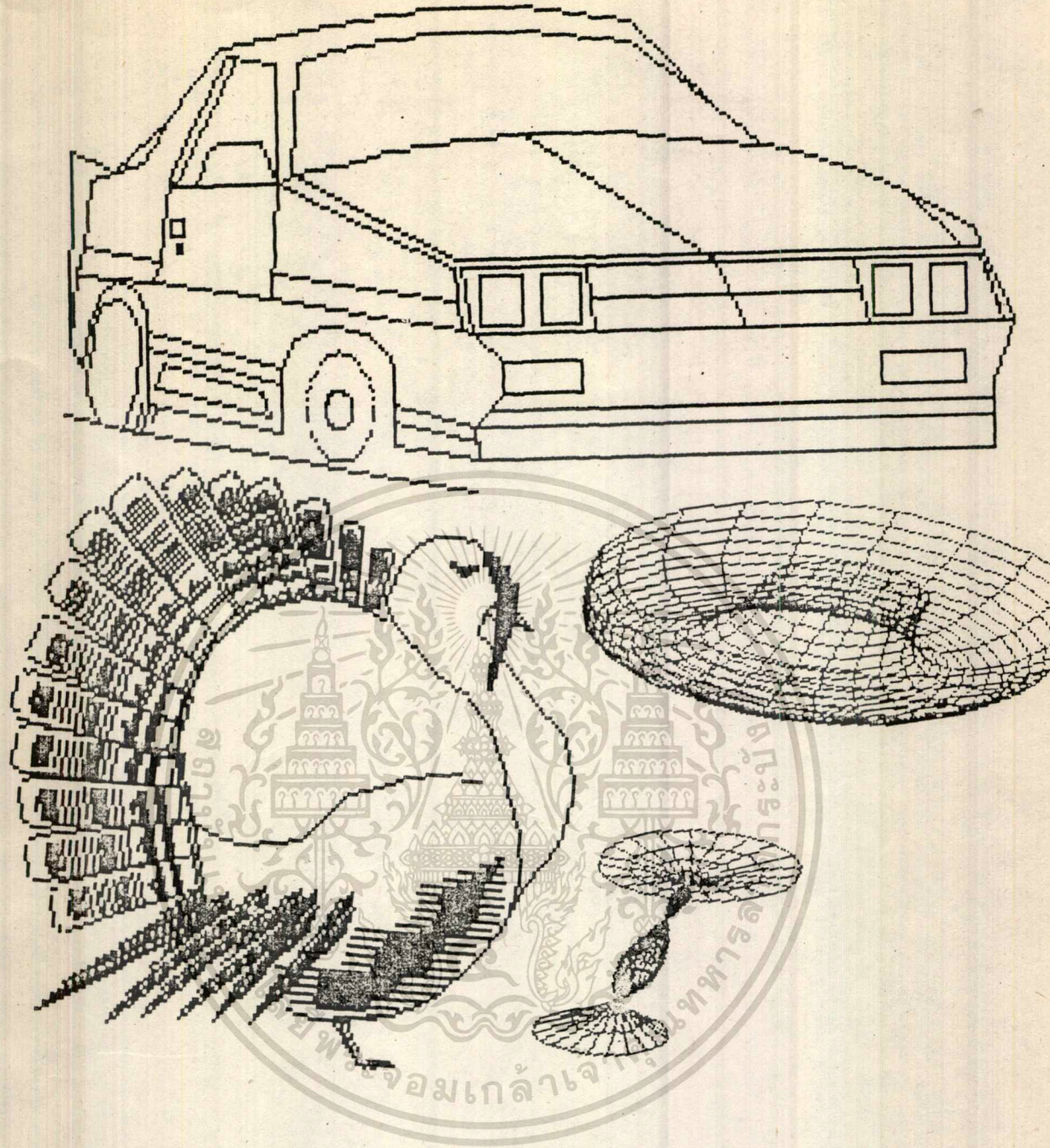
อย่างกว้างขวางมาในทางด้านนี้บนเครื่องไมโครคอมพิวเตอร์ ทางด้านศิลปะการออกแบบ และเกมส์ต่างๆ

สำหรับวิทยานิพนธ์นี้พอที่จะจัดอยู่ในหัวข้อการออกแบบทางเรขาคณิต (geometric design) เนื่องจากวิทยานิพนธ์นี้ได้สร้างภาพ 3 มิติ และให้สามารถที่จะหมุนดูพิจารณาที่จุดมองต่างๆได้



รูปที่ 1.1 การใช้คอมพิวเตอร์ช่วยในการออกแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.2 รูปแบบหนึ่งของการแสดงภาพด้วยคอมพิวเตอร์กราฟฟิค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

หลักการพื้นฐานทางคอมพิวเตอร์กราฟิค

หลักการทางคอมพิวเตอร์กราฟิคนั้นส่วนใหญ่จะอาศัยหลักการทางคณิตศาสตร์ โดยเฉพาะทางเมตริกเข้าช่วย การสร้างภาพต่างๆทางคอมพิวเตอร์กราฟิค เรามีความจำเป็นที่จะต้องเข้าใจหลักการพื้นฐานเกี่ยวกับการส่งผ่านหรือการเคลื่อนย้ายจุดบนจอภาพ ทั้งใน 2 มิติและ 3 มิติ เสียก่อน โดยจะเริ่มพิจารณาจากกรณี 2 มิติ แล้วจึงพิจารณา 3 มิติ ในหัวข้อ 2.5

2.1 การแปลงจุดหรือภาพใน 2 มิติ (transformation of points) ในการแปลงจุดเราอาศัยคณิตศาสตร์เมตริกซ์ช่วยในการแปลงจุดโดย

กำหนดให้ T = เมตริกซ์ = $\begin{bmatrix} A & B \\ C & D \end{bmatrix}$

P = จุดหรือคู่ลำดับซึ่งเป็นจุดเริ่มต้น = (x,y)

P = คือจุดที่ได้จากการแปลง = (x,y)

เราพบว่า $P = PT = (x,y)\begin{bmatrix} A & B \\ C & D \end{bmatrix}$

$= [(Ax + Cy) , (Bx + Dy)]$ (2.1)

$= (x,y)$

จากสมการข้างต้นเราพบว่า ค่า A,B,C และ D จะเป็นตัวกำหนดคุณลักษณะของการแปลง ซึ่งเราแบ่งออกได้ดังนี้ คือ

2.1.1 การแปลงเอกลักษณ์ (indentity)

โดยกำหนดให้ B = C = 0 และ A = D = 1 เราพบเมตริกซ์ T เป็นเมตริกซ์เอกลักษณ์

$(x',y') = (x,y).\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = (x,y)$

เราจะได้ว่าหลังจากการแปลงจะได้จุดเดิม

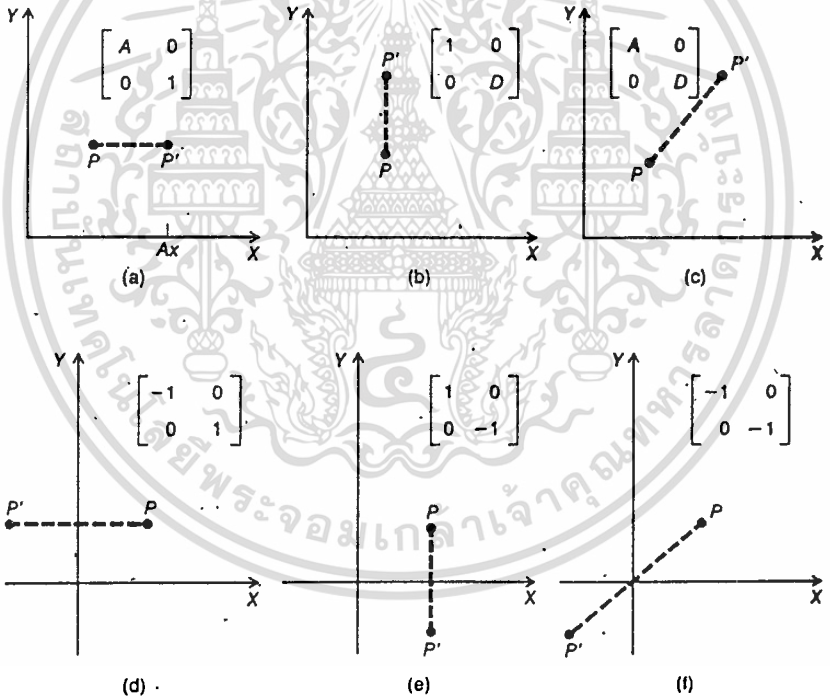
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 การแปลงขนาด (scaling)

โดยการกำหนดให้ B = C = Ø ส่วน A และ D ≠ Ø ได้

$$(x',y') = (x,y)\begin{bmatrix} A & Ø \\ Ø & D \end{bmatrix} = (Ax,Dy)$$

เราพบว่าค่า x และ y มีค่าเปลี่ยนแปลงไปขึ้นกับค่า A และ D ถ้า A = 1 การแปลงขนาดจะทำให้แนวแกน x ดังรูป 2.1 (a) และถ้า D ≠ 1 ก็ได้ลักษณะคล้ายกัน ดังรูป 2.1 (b) ถ้ากรณี A,D = 1 หรือ Ø ก็จะได้ดังรูป 2.1(c)



รูป 2.1 การแปลงของจุด

2.1.3 การแปลงแบบสะท้อน (reflection)

การแปลงแบบสะท้อนจะเกิดขึ้นเมื่อ ค่า A หรือ D หรือทั้ง 2 ตัว เป็นค่าลบ ลักษณะการเกิดจะพบว่าจุดที่ได้จากการแปลงแบบนี้จะมีลักษณะเหมือนกับจุดที่เกิดอยู่ในกระจก เพื่อประกอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเข้าใจ สมมุติให้ B=C=Ø, A=-1 และ D=1 จากสมการ 2.1 ได้

$$(x',y') = (x,y)\begin{bmatrix} -1 & Ø \\ Ø & 1 \end{bmatrix} = (-x,y)$$ (ดูรูป 2.1 (d) ประกอบ)

จากรูป 2.1(e) และ 2.1(f) เป็นกรณีต่างๆของการแปลงแบบสะท้อน

2.1.4 การแปลงแบบเชียร์ (shearing)

เป็นลักษณะการแปลงที่ให้ตัวแปรหนึ่งในคู่ลำดับมีค่าคงที่ ส่วนอีกตัวแปรหนึ่งจะเปลี่ยนค่าไป เป็นลักษณะของฟังก์ชันเชิงเส้นของตัวแปรแรก ตัวอย่าง

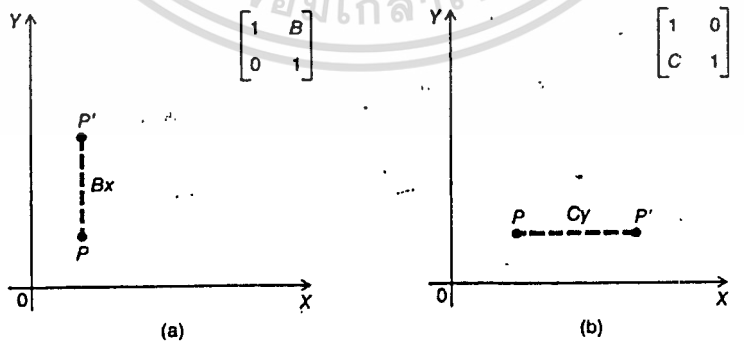
ให้ A=D=1, C=Ø และ B≠Ø จะได้

$$(x',y') = (x,y)\begin{bmatrix} 1 & B \\ Ø & 1 \end{bmatrix} = (x,Bx+y)$$ (ดูรูป 2.2(a)ประกอบ)

จากสมการที่ผ่านมา y จะแปรตามฟังก์ชันเชิงเส้นของ x

อีกกรณีหนึ่งคือ

$$(x',y') = (x,y)\begin{bmatrix} 1 & Ø \\ C & 1 \end{bmatrix} = (x+Cy,y)$$ (ดูรูป 2.2(b)ประกอบ)



รูปที่ 2.2 การแปลงแบบเชียร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5 การแปลงแบบหมุน (rotation)

จุดสามารถที่จะถูกหมุนไปเป็นมุม θ โดยเทียบกับจุดกำเนิด (origin) ได้โดยมีเมตริกซ์ T ดังสมการข้างล่าง

$$(x',y') = (x,y).T = (x,y).\begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$$

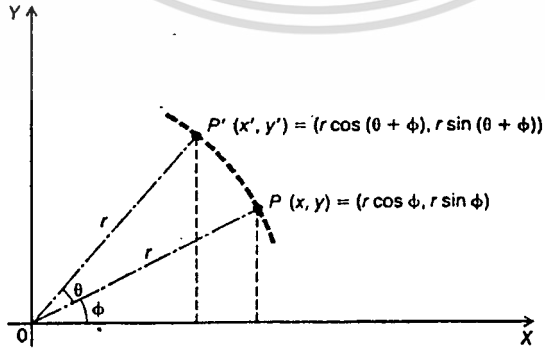
$$= (x\cos\theta - y\sin\theta \ , \ x\sin\theta + y\cos\theta) \qquad (2.2)$$

โดย θ (POSITIVE) เป็นค่าบวกหรือกล่าวอีกอย่างหนึ่งว่าเป็นการหมุนแบบทวนเข็มนาฬิกา (counterclockwise) จาก x ไป y ถ้ามีการหมุนตามเข็มนาฬิกา (clockwise) มุมต้องเป็นลบ เพราะฉะนั้นจากสมการ 2.2 แทน ด้วย $-\theta$ ได้ $\cos(-\theta) = \cos\theta$ และ $\sin(-\theta) = \sin\theta$ ได้

$$(x',y') = (x,y).\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

$$= (x\cos\theta + y\sin\theta \ , \ -x\sin\theta + y\sin\theta)$$

จากสมการ 2.2 เราสามารถพิสูจน์ได้ดังต่อไปนี้ พิจารณารูป 2.21



รูปที่ 2.21 การหมุนของจุดใน 2-D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนด P คือ จุดเริ่มต้น

P′ คือ จุดที่ได้จากการแปลง

r คือ ระยะทางจากจุดกำเนิดถึงจุด P และ P (จะมีค่าเท่ากันเพราะการหมุนเป็นลักษณะวงกลม)

จากรูปได้สมการดังนี้

$$x = r\cos\phi \quad , \quad y = r\sin\theta$$

ได้

$$x = r\cos(\phi+\theta) = r\cos\phi\cos\theta - r\sin\phi\sin\theta$$

$$= x\cos\theta - y\sin\theta$$

$$y = r\sin(\phi+\theta) = r\cos\phi\sin\theta + r\sin\phi\cos\theta$$

$$= x\sin\theta + y\cos\theta$$

2.1.6 การแปลงแบบทรานสเลชั่น (translation)

เป็นลักษณะในการแปลงค่าในคู่ลำดับโดยการบวกค่าเพิ่มเข้าไป เป็นการเคลื่อนย้ายจุดวิธีหนึ่ง สมการ 2.3

$$(x',y') = [x,y]+[Tx,Ty] = (x+Tx,y+Ty) \qquad 2.3$$

รูปตัวอย่างดูได้จากหัวข้อถัดไป (การแปลงของเส้นและวัตถุ)

2.2 การแปลงเส้นและวัตถุ (transformation of lines and objects)

การแปลงเส้นและวัตถุมีลักษณะคล้ายการแปลงจุด เพียงแต่การแปลงเส้นหรือวัตถุเป็นการนำเอาจุดทุกจุดบนเส้นมาทำการแปลงแบบต่างๆ หรือบางครั้งเรานำเพียงจุดยอดมาแปลงเท่านั้นก็พอ และจากนั้นการลากเส้นระหว่างจุดที่แปลงมา สมการ 2.4 แสดงการแปลงของเส้นหรือวัตถุ

$$\begin{bmatrix} P_1' \\ P_2' \\ \vdots \\ P_n' \end{bmatrix} = \begin{bmatrix} x_1' & y_1' \\ x_2' & y_2' \\ \vdots & \vdots \\ x_n' & y_n' \end{bmatrix} = \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \vdots & \vdots \\ x_n & y_n \end{bmatrix} \oplus \begin{bmatrix} A & B \\ C & D \end{bmatrix} \qquad (2.4)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คือ สัญลักษณ์การกระทำโดยมีค่าเท่ากับการคูณเมื่อใช้กับการแปลงแบบขนาด แบบสะท้อน แบบเชียร์ แบบเอกลักษณ์ และแบบหมุน แต่จะมีค่าเป็นบวกเมื่อใช้กับการแปลงแบบทรานสเลชั่น

2.2.1 ข้อจำกัดของการใช้ระบบคู่ลำดับมาตรฐาน (standrad coordinate system)

จากระบบคู่ลำดับที่เราได้พิจารณามาข้างต้น ซึ่งเป็นเมตริกซ์ทั่วไปขนาด 2*2 เราพบว่ามีข้อจำกัดที่สำคัญในการนำไปใช้อยู่ 2 ประการคือ

ก) จากการแปลงที่ผ่านมาทั้งหมดเรายึดเอาจุดกำเนิดเป็นเกณฑ์ ดังนั้น มีผลทำให้การแปลงแบบหมุนซึ่งแต่เดิม เรายึดจุดกำเนิดเป็นจุดหมุน ทำให้เราไม่สามารถเปลี่ยนจุดหมุนเป็นจุดใดๆได้สะดวก นอกจากนี้ยังมีผลต่อการแปลงแบบสะท้อน ถ้าเรายังใช้จุดกำเนิดเป็นเกณฑ์อยู่ การแปลงแบบสะท้อนจะทำได้เฉพาะแกน x แกน y และจุดกำเนิดเท่านั้น จะทำการแปลงแบบสะท้อนกับจุดใดๆไม่ได้

ข) การแปลงตั้งแต่หัวข้อ 2.1.1 - 2.1.5 เป็นลักษณะการคูณของเมตริกซ์ T กับคู่ลำดับเดิม แต่การแปลงในหัวข้อ 2.1.6 เป็นการบวก ทำให้ในการใช้งานจริง ซึ่งมีการแปลงหลายๆอย่างร่วมกันเกิดความยุ่งยาก ซึ่งมีสาเหตุมาจากคุณสมบัติของเมตริกซ์

จากปัญหาข้างต้นเราจึงต้องสร้างระบบคู่ลำดับระบบใหม่เรียกว่า ระบบคู่ลำดับแบบเสมือน (homogeneous coordinate system)

2.3 ระบบคู่ลำดับแบบเสมือน (homogeneout coordinate system)

ในระบบนี้คู่ลำดับ (x,y) จะถูกแทนที่ด้วยคู่ลำดับ (x,y,H) โดยกำหนดให้ H ไม่เป็น Ø จากนั้นทำการนอร์มอนไรด์ (normalize) ค่า H ได้คือ

(x,y,H) --normalize--> (x/H,y/H,1)

จากที่เรากำหนดให้ H ไม่เท่ากับ Ø และเป็นค่าคงที่ เรากำหนดให้ H = 1 เพราะฉะนั้นคู่ลำดับใหม่ที่ได้ก็คือ (x,y,1)

จากผลการใช้ระบบคู่ลำดับเสมือนทำให้เราสามารถเปลี่ยนเมตริกซ์ T จากขนาด 2*2 มาเป็นขนาด 3*3 ได้ดังสมการ 2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \longrightarrow \begin{bmatrix} A & B & 0 \\ C & D & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (2.5)$$

จากผลของเมตริกซ์ขนาด 3*3 เราสามารถเปลี่ยนการแปลงแบบทรานสเลชั่น จากการบวกมาเป็นการคูณได้ ดังสมการ 2.6

$$T(\text{translation}) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ L & M & 1 \end{bmatrix} \qquad (2.6)$$

และ $(x',y',1) = (x,y,1)T$

เราพบว่า

$x' = x+L$ และ $y' = y+M$

กรณี การแปลงแบบขนาด

$$\begin{bmatrix} A & 0 \\ 0 & D \end{bmatrix} \longrightarrow \begin{bmatrix} A & 0 & 0 \\ 0 & D & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

กรณี การแปลงแบบหมุน

$$\begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \longrightarrow \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

สำหรับการแปลงกรณีอื่นๆ ก็เช่นเดียวกัน

2.4 วิวพอร์ทของกราฟฟิค 2 มิติ (viewport of 2-D graphic)

ในการสร้างภาพ 2 มิติของวัตถุ เรากำหนดให้วัตถุ 2 มิตินั้นวางตัวอยู่ในระบบคู่ลำดับมาตรฐาน (standard coordinate system) หรือเราเรียกอีกอย่างหนึ่งว่า ระบบคู่ลำดับ

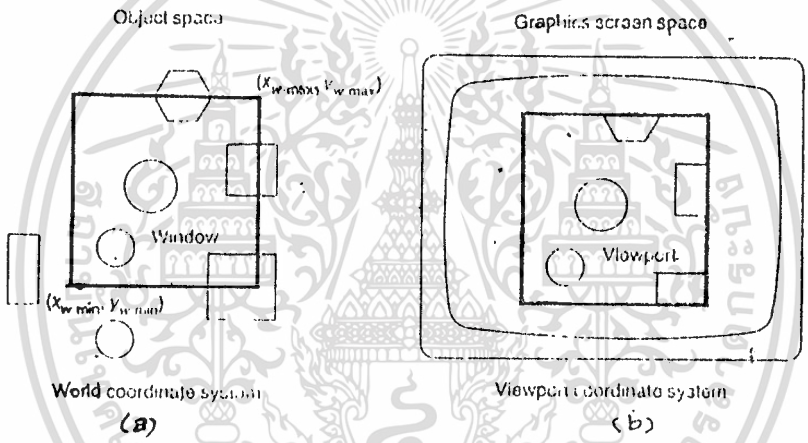
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โลก (world coordinate system) และกำหนดพื้นผิวภาพแสดงบนจอคอมพิวเตอร์ว่าเป็น คู่ลำดับจอภาพ (screen display coordinate)

จากการกำหนดข้างต้นเราจะได้กล่าวถึงเทคนิค 3 ประการ ของคอมพิวเตอร์กราฟฟิค 2 มิติ คือ

2.4.1 การทำวินโดร์คิบปิ้ง (window clipping)

ในการมองวัตถุ 2 มิติ ในระบบคู่ลำดับโลก เรามีความจำเป็นที่จะต้องกำหนดขอบเขตการมองและตัดเอาส่วนที่อยู่นอกขอบเขตการมองออกไป ดังรูป 2.3 (a)



รูปที่ 2.3 วินโดร์และวิวพอทร์

การกำหนดขอบเขตการมองของระบบคู่ลำดับโลก เราเรียกว่า วินโดร์ ส่วนการกำจัดส่วนที่อยู่นอกขอบเขตการมองเราเรียกว่า คิปปิ้ง

หลังจากมีการกำหนดวินโดร์แล้ว เราต้องกำหนดขอบเขตการมองของจอ ซึ่งเราเรียกว่า วิวพอร์ท ดังรูป 2.3 (b) การกำหนดวิวพอร์ทเป็นการกำหนดว่าส่วนที่เราแสดงออกจอภาพจะมีขนาดและขอบเขตเท่าใด

2.4.1.1 การขจัดจุด (clipping points)

จากที่ได้กล่าวมาในหัวข้อที่แล้ว เรามีวิธีการที่จะไม่แสดงจุดที่อยู่ภายนอกของวินโดว์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้ โดยพิจารณาจากรูป 2.3(a) เรากำหนดจุดต่ำสุดมุมซ้าย (X_{wmin},Y_{wmin}) และจุดสูงสุดมุมขวา (X_{wmax},Y_{wmax}) ถ้าจุด (x,y) ใดๆที่จะปรากฏในวินโดว์ได้ เมื่อทำให้เงื่อนไขของสมการ (2.7) และ (2.8) เป็นจริง

$$X_{wmin} \leqslant X \leqslant X_{wmax} \qquad (2.7)$$

$$Y_{wmin} \leqslant Y \leqslant Y_{wmax} \qquad (2.8)$$

ยังมีอีกวิธีการหนึ่งคือ การขจัดเส้น (clipping line) ซึ่งไม่ได้กล่าวไว้ในวิทยานิพนธ์เล่มนี้ หากผู้อ่านสนใจสามารถหาอ่านได้จากหนังสือหมายเลข (1) ในบรรณานุกรมของวิทยานิพนธ์นี้

2.4.2 การส่งผ่านจากวินโดว์ไปวิวพอร์ท (window to viewport mapping)

หลังจากที่เรากำหนดวินโดว์และวิวพอร์ทเรียบร้อยแล้ว ในหัวข้อนี้เราจะกล่าวถึงความสัมพันธ์ระหว่างทั้ง 2 เนื่องจากเรามีความจำเป็นต้องมีการส่งผ่านจาก วินโดว์ ไปยัง วิวพอร์ท ในการส่งผ่านกันนั้นให้พิจารณาจากรูป (2.4)

จากรูปเราสมมติให้ จุด(X_w,Y_w) มีการส่งผ่าน (mapping)ไปยังจุด (X_v,Y_v)

กำหนดจุด (X_{vmin},Y_{vmin}) และ (X_{vmax},Y_{vmax})

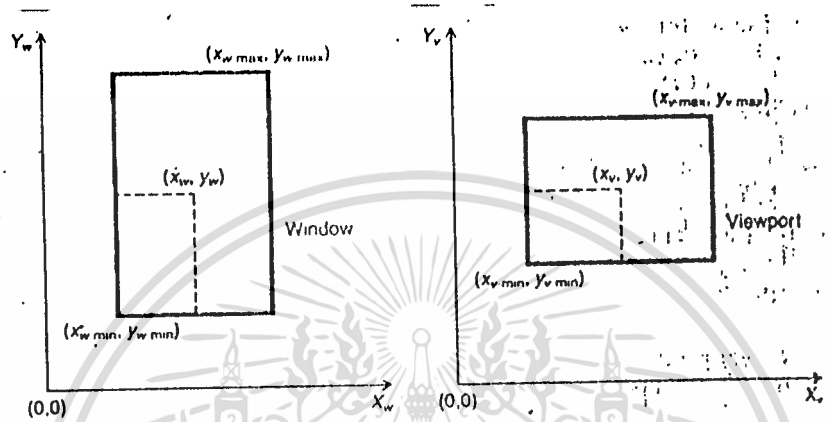
ในการส่งผ่านเราจะใช้ขบวนการเปลี่ยนแกน โดยการใช้การแปลงแบบทรานเรชั่น จากระบบคู่ลำดับจอภาพ จากนั้นทำการแปลงขนาด ดังสมการ (2.9)

$$[X_v,Y_v,1] = [X_w,Y_w,1]\cdot\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ -X_{wmin} & -Y_{wmin} & 1\end{bmatrix}\cdot\begin{bmatrix}S_1 & 0 & 0\\ 0 & S_2 & 0\\ 0 & 0 & 1\end{bmatrix}\cdot\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ X_{vmin} & Y_{vmin} & 1\end{bmatrix} \qquad (2.9)$$

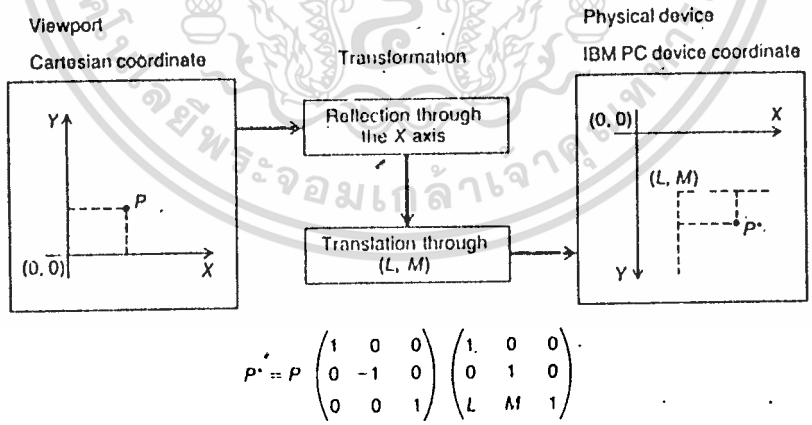
$$=[S_1(X_w-X_{wmin})+X_{vmin},\ S_2(Y_w-Y_{win})+Y_{vmin},\ 1]$$

หาค่าตัวประกอบขนาด (scale factor) S_1,S_2 ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 การส่งผ่านจากวินโดร์ไปวิวพอร์ท



$$P' = P \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ L & M & 1 \end{pmatrix}$$

รูปที่ 2.5 ขบวนการแปลงคู่ลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$S_1 = \frac{X_{vmax} - X_{vmin}}{X_{wmax} - X_{wmin}} \qquad (2.10)$$

$$S_2 = \frac{Y_{vmax} - Y_{vmin}}{Y_{wmax} - Y_{wmin}} \qquad (2.11)$$

เราจะพบว่าเราควรกำหนดให้ $S_1=S_2$ เพื่อให้รูปร่างที่ได้จากการส่งผ่านมาไม่เกิดการเปลี่ยนแปลงไป (distort)

2.4.3 คู่ลำดับของอุปกรณ์แสดงภาพโดยทั่วๆไป (จอภาพ) (generating physical device coordinate)

เนื่องจากคู่ลำดับจอภาพ (X_v, Y_v) ที่เรากำหนดมาแต่ต้นนั้นไม่สัมพันธ์กับอุปกรณ์แสดงภาพของเครื่อง IBM PC โดยตรง โดยมีความแตกต่างกันในเรื่องของการกำหนดแกน x,y ดังรูป (2.5) ดังนั้นเราจึงต้องมีการแปลงเพื่อให้เกิดความสอดคล้องโดยเราทำการแปลงสะท้อนกับแกน X จากนั้นทำการแปลงแบบทรานสเรชั่น ดังสมการ (2.12)

กำหนดคู่ลำดับ (X_s, Y_s) เป็นคู่ลำดับของอุปกรณ์แสดงภาพ (โดยเฉพาะจอของเครื่องคอมพิวเตอร์ IBM PC)

$$[X_s, Y_s, 1] = [X_v, Y_v, 1] \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ L & M & 1 \end{bmatrix} \qquad (2.12)$$

$$(X_s, Y_s, 1) = (X_v + L \ , \ M - Y_v \ , \ 1)$$

จากสมการ 2.9 และ 2.12 เราจะได้ว่า

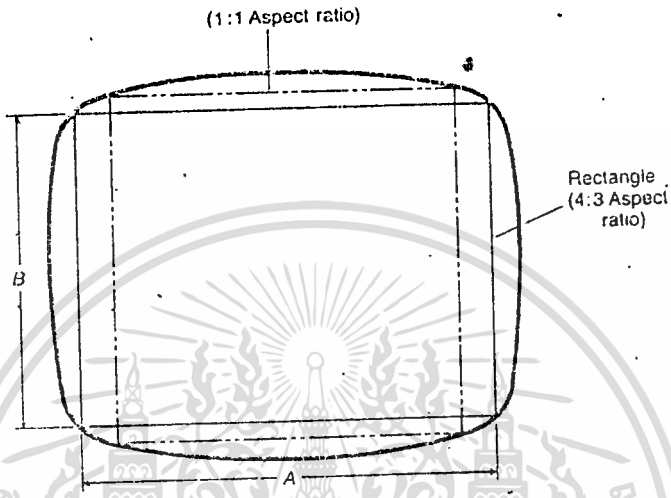
$$X_s = S_1(X_w - X_{wmin}) + X_{vmin} + L \qquad (2.13)$$

$$Y_s = S_2(Y_{wmin} - Y_w) - Y_{vmin} + M \qquad (2.14)$$

จากคุณสมบัติจากจอภาพ IBM PC เราพบว่าอัตราส่วนระหว่างด้านกว้างกับด้านยาวไม่เป็น 1 ต่อ 1 เราจึงต้องมีการเพิ่มค่าปรับแก้ให้กับค่าในแกน X เพื่อให้ภาพที่ปรากฏบนจอเกิดความสมดุลระหว่าง 2 แกน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนดให้ SCF คือ ค่าการปรับแก้ขนาดของจอ (screen scaling adjustment factor) เราสามารถพิจารณาจากตารางในรูป 2.6



Diagonal screen size	A (mm)	B (mm)	Square (mm)	Typical monitor
9″	148	111	118	Portable IBM PC
12″	216	162	172	
13″	240	180	191	IBM PC
16″	283	213	226	
19″	348	261	276	TV
20″	356	267	283	
21″	368	276	293	
23″	408	306	324	TV

รูปที่ 2.6 ขนาดและรูปแบบของจอ

ค่า SCF จะหาได้โดย

$$SCF = V/H$$

โดย $H = \frac{\text{ค่าความยาวแกนนอน (horizontal length)}}{\text{จำนวนของจุดที่มากสุดในแนวแกนนอน}}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ[illegible]เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$V = \frac{\text{ค่าความยาวในแนวแกนตั้ง (vertical length)}}{\text{จำนวนมากที่สุดของจุดในแนวแกนตั้ง}}$$

สำหรับจอ IBM PC ที่ใช้อยู่ในโหมดสี (color mode) มีค่าความละเอียด (resolution) เท่ากับ 320 จุด * 200 จุด ดังนั้น

$$H = 240mm / 320$$

และ $$V = 180mm / 200$$

ได้ค่า $$SCF = V/H = 1.2$$

ดังนั้น ค่า X_s ที่ได้จากการปรับแก้จะมีค่าเท่ากับ

$$X_s = SCF\ [S_i(X_w - X_{wmin}) + X_{vmin} + L\ ] \qquad (2.15)$$

2.5 การแปลงภาพและจุดในระบบสามมิติ (3-D transformation)

ก่อนที่เราจะได้ศึกษาถึงการแปลงในระบบ 3 มิติ เราจะขอกล่าวถึงระบบคู่ลำดับที่ใช้ในระบบ 3 มิติ เสียก่อน

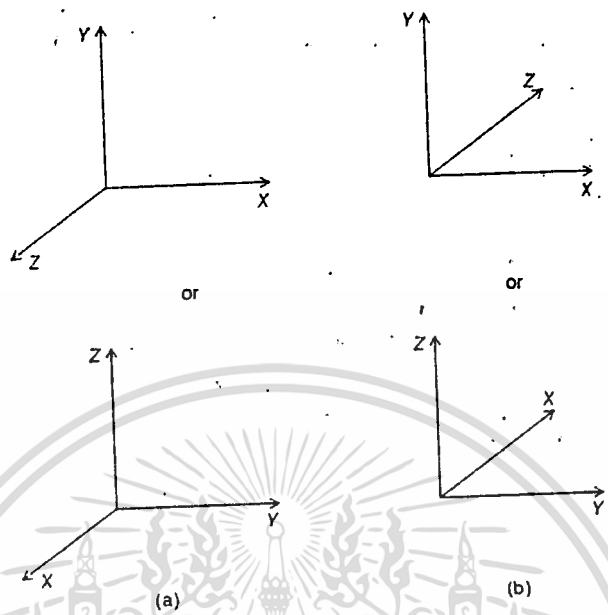
2.5.1 ระบบแกนและระบบคู่ลำดับ

ระบบแกนที่ใช้ใน 3 มิตินั้น ประกอบไปด้วย 2 ระบบคือ ระบบมือขวา (right-handed system) และระบบมือซ้าย (left-handed system) ดังรูป 2.6 (a) และ(b) ความแตกต่างของ 2 ระบบ ก็คือ แกนใดแกนหนึ่ง (เพียงแกนเดียว) ของ 2 ระบบ จะมีทิศทางตรงกันข้ามกัน สังเกตุได้จาก 2 รูปบนของรูป 2.61 จะมีทิศทางของแกน X ,Y เหมือนกัน แต่ทิศของ Z ตรงกันข้ามกัน

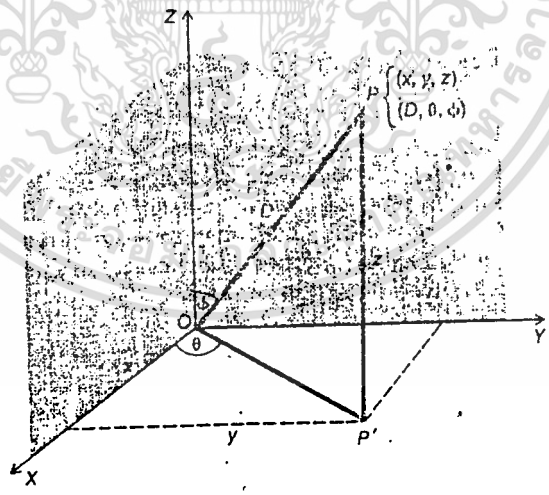
สำหรับคู่ลำดับในระบบ 3 มิตินั้น เราแบ่งออกเป็น 2 แบบ คือ

- ระบบคู่ลำดับแบบทรงเหลี่ยม (rectangular coordinate system) จุดในระบบแบบนี้จะถูกกำหนดตามระยะห่างระหว่างแกน x,y,z ดังนั้นการกำหนดจุดจะเป็น (x,y,z) ดูรูป 2.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.61 ระบบคู่ลำดับใน 3-D



รูปที่ 2.7 ความสัมพันธ์ระหว่าง 2 ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ระบบคู่ลำดับแบบทรงโค้ง (sperical coordinate system) ในระบบนี้จุดต่างๆ จะถูกแทนด้วย (D, θ , ϕ) พิจารณาจากรูป 2.7

D คือ ระยะห่างจากจุดถึงจุดกำเนิด

θ คือ มุมระหว่างเส้น $\overline{OP}$ กับแกน X

ϕ คือ มุมระหว่างเส้น $\overline{OP}$ กับแกน Z

สำหรับมุม θ จะมีการวัดแบบทวนเข็มนาฬิกา

ความจำเป็นในการกำหนดคู่ลำดับแบบทรงโค้งเกิดเนื่องมาจาก ระบบคู่ลำดับนี้มีความสะดวกและง่ายต่อการเข้าใจ เมื่อพิจารณาถึงจุดมองกับตัววัตถุในระบบ 3 มิติ ซึ่งจะได้กล่าวถึงต่อไป

2.5.2 ความสัมพันธ์ระหว่างคู่ลำดับทั้งสอง

เนื่องจากระบบคู่ลำดับทั้ง 2 แบบข้างต้น มีความจำเป็นมากในการนำไปใช้ในวิชาคอมพิวเตอร์กราฟฟิค ดังนั้น การศึกษาถึงความสัมพันธ์ระหว่าง 2 ระบบจึงเป็นสิ่งที่หลีกเลี่ยงไม่ได้ เมื่อพิจารณาจากรูป 2.7 แล้ว โดยอาศัยวิชาทางตรีโกณมิติ (trigonometry) เราสามารถที่จะแปลงจากระบบหนึ่งมาอีกระบบหนึ่งได้ดังสมการ 2.16

$$x = D.\sin\phi\ \cos\theta$$

$$y = D.\sin\phi\ \sin\theta$$

$$z = D.\cos\theta$$

และ (2.16)

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

$$\theta = \tan^{-1}(y/x)$$

$$\phi = \cos^{-1}(z/D)$$

ตัวอย่าง สมมุติว่าเรามีจุดซึ่งอยู่ในระบบคู่ลำดับทรงเหลี่ยม คือ (4,2,3) เราสามารถเปลี่ยนเป็นจุดในระบบคู่ลำดับทรงโค้งโดยเท่ากับ (5.4,27°,56°) เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.3 การแปลงแบบต่างๆในระบบ 3 มิติ

ในหัวข้อ 2.1 และ 2.2 เราได้กล่าวถึงหัวข้อการแปลงในระบบ 2 มิติ และในหัวข้อ 2.3 ได้กล่าวถึง การใช้ระบบคู่ลำดับเสมือน (homogenous system) ในระบบ 2 มิติ คู่ลำดับจะประกอบด้วย (x,y) เท่านั้น เราจึงใช้การแปลงระบบคู่ลำดับเสมือนเป็นเมตริกซ์ขนาด 3*3 แต่ในระบบ 3 มิติ คู่ลำดับประกอบไปด้วย (x,y,z) ดังนั้น เราจึงต้องขยายขนาดของเมตริกซ์ออกไปเป็นขนาด 4*4 เพื่อให้สอดคล้องกับตัวแปร z ที่เพิ่มขึ้นมา เพราะฉะนั้นการแปลงในระบบ 3 มิติ โดยใช้ระบบคู่ลำดับแบบเสมือนจะได้ดังสมการ 2.17

$$\begin{bmatrix} A & B & C & \emptyset \\ D & E & F & \emptyset \\ G & H & I & \emptyset \\ L & M & N & 1 \end{bmatrix} \qquad (2.17)$$

และสามารถแบ่งเป็นเมตริกซ์ได้ 3 ส่วน คือ

เมตริกซ์ย่อยที่ 1 ใช้สำหรับการแปลงแบบขนาด แบบเชียร์ แบบสะท้อน และแบบหมุนของวัตถุในระบบ 3 มิติ

เมตริกซ์ย่อยที่ 2 ใช้สำหรับการทรานสเลชั่นแบบเชิงเส้น (linear translation)

เมตริกซ์ย่อยที่ 3 ใช้สำหรับกรณีที่ต้องการแปลง 1 กับ 2 ในรูปของการคูณ

2.5.3.1 การแปลงแบบทรานสเลชั่น (3-D traslation) สมการที่ใช้คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$(x',y',z',1) = (x,y,z,1).\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ L & M & N & 1 \end{bmatrix} \qquad (2.19)$$

(ดูรูป 2.8 (a) ประกอบ)

2.5.3.2 การแปลงแบบขนาด (3-D scaling)

มีสมการคือ

$$(x',y',z',1) = (x,y,z,1).\begin{bmatrix} A & 0 & 0 & 0 \\ 0 & E & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (2.20)$$

ข้อสังเกตุ เราอาศัยแกนเฉียงของเมตริกซ์ (digonal) ช่วยในการแปลงขนาด

(ดูรูป 2.8 (c) ประกอบ)

2.5.3.3 การแปลงแบบเชียร์ (3-D shearing)

สมการที่ใช้คือ

$$(x',y',z',1) = (x,y,z,1).\begin{bmatrix} 1 & B & C & 0 \\ D & 1 & F & 0 \\ G & H & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (2.21)$$

(ดูรูป 2.8 (d) ประกอบ)

2.5.3.4 การแปลงแบบสะท้อน (3-D reflection)

การแปลงแบบสะท้อนเราสามารถพิจารณาได้ 3 กรณี โดยพิจารณาที่ เมตริกซ์การแปลง (T)

-การแปลงแบบสะท้อนกับระนาบ xy

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (2.22)$$

-การแปลงแบบสะท้อนระนาบ yz

$$T = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (2.23)$$

(ดูรูป 2.8 (e) ประกอบ)

-การแปลงแบบสะท้อนกับระนาบ xz

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (2.24)$$

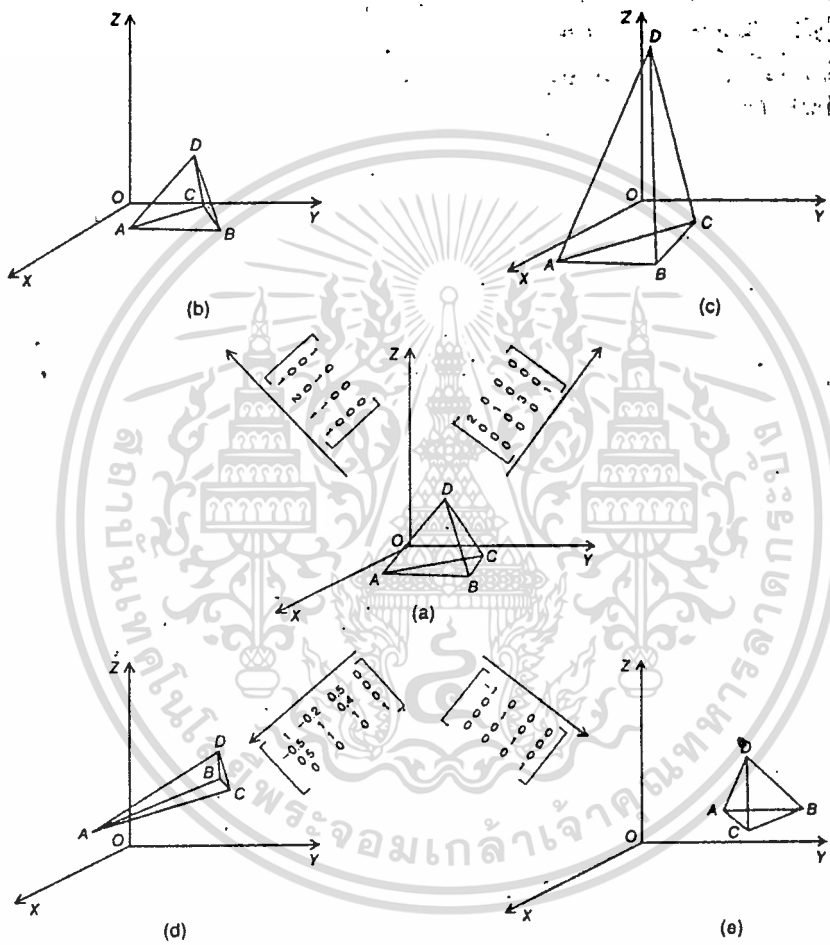
2.2.3.5 การแปลงแบบหมุน (3-D rotation)

ในการพิจารณาเรื่องการแปลงแบบหมุน เราจะพิจารณาโดยยึดจุดกำเนิดเป็นจุดหมุน (สำหรับกรณีที่ใช้จุดใดๆ เป็นจุดหมุนนั้น จะไม่ได้กล่าวในที่นี้ ผู้สนใจสามารถอ่านได้จากหนังสือ[1]) สำหรับในกรณีของ 3 มิติ เราแยกการแปลงแบบหมุนได้ 3 แกน คือ

-การแปลงแบบหมุนแกน z

$$R(\theta)_z = T = \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (2.25)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 ตัวอย่างการแปลง 3-D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

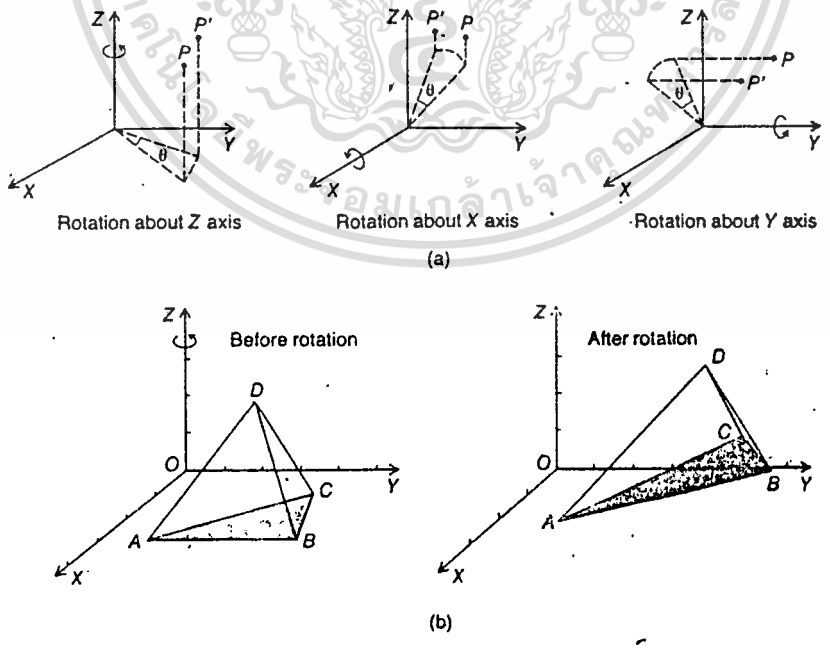
-การแปลงแบบหมุนแกน x

$$R(\Theta)_x = T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\Theta & \sin\Theta & 0 \\ 0 & -\sin\Theta & \cos\Theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.26)$$

-การแปลงแบบหมุนแกน y

$$R(\Theta)_y = T = \begin{bmatrix} \cos\Theta & 0 & -\sin\Theta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\Theta & 0 & \cos\Theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.27)$$

ดูรูป 2.9 (a) ประกอบ สำหรับตัวอย่างการแปลงแบบหมุนแสดงไว้ในรูป 2.9 (b)



รูปที่ 2.9 การแปลงแบบหมุน 3-D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อสังเกตุ การแปลงแบบหมุนในแนวแกนหนึ่งและตามด้วยอีกแนวแกนหนึ่ง จะมีค่าไม่เท่ากับการแปลงในทางกลับกันเนื่องจากเป็นการคูณแบบเมตริกซ์

จากกฎของเมตริกซ์ ให้ A,B เป็นเมตริกซ์

ได้ว่า $A.B \neq B.A$

ดังนั้น

$$R(\Theta)_x .R(\Theta)_y \neq R(\Theta)_y .R(\Theta)_x$$

2.5.4 การมองในระบบ 3 มิติ (viewing in 3-D)

ในการแปลงจุดต่างๆ ของวัตถุ ในคู่ลำดับโลก (world coordinate system) ให้เป็นคู่ลำดับของจอภาพบนเครื่องคอมพิวเตอร์กราฟฟิคนั้น เราต้องกำหนด จุดมองในระบบ 3 มิติ โดยให้จุดมอง (view point) หรือจุดที่ตามอง อยู่บนจุดกำเนิดของระบบคู่ลำดับสายตา (eye coordinate system) ซึ่งคู่ลำดับสายตามีความสำคัญมากในการแปลงค่าคู่ลำดับ (viewing transformation) ให้เป็นคู่ลำดับบนระนาบ 2 มิติ บนจอภาพ ในการแปลงคู่ลำดับโลกให้เป็นคู่ลำดับโลกให้เป็นคู่ลำดับบนระนาบ 2 มิติ ของจอภาพนั้น เราเรียกว่า การโปรเจค (projection) การโปรเจคในระบบ 3 มิติ มี 2 แบบ คือ การโปรเจคแบบศูนย์กลาง (central projection) หรือเรียกอีกอย่างหนึ่งว่า การโปรเจคแบบขนาน (parallel projection) และการโปรเจค ณ จุดมองใดๆ (projection through an arbitrary view point) ในวิทยานิพนธ์นี้จะกล่าวถึงแต่การโปรเจคแบบหลังนี้เท่านั้น เนื่องจากการโปรเจคแบบแรก เป็นการกำหนดจุดมองเพียงจุดเดียว ซึ่งต่างจากการโปรเจกแบบหลัง ซึ่งกำหนดจุดมองได้ทุกจุด

2.5.4.1 การโปรเจค ณ จุดมองใดๆ (projection through an arbitrary viewpoint)

ในการโปรเจคในระบบ 3 มิติ เรากำหนดให้ระบบคู่ลำดับโลก เป็นระบบมือขวา (right-handed system) และระบบคู่ลำดับสายตาเป็นระบบมือซ้าย (left-handed system) พิจารณาจากรูป 2.10 กำหนดให้แกนของระบบคู่ลำดับสายตา คือ X_e, Y_e, Z_e จุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

O_e คือจุดมอง แกน X_e, Y_e เป็นระบบคู่ลำดับของจอภาพ แกน x,y,z เป็นแกนในระบบคู่ลำดับโลก และจุด P คือ จุดในระบบคู่ลำดับโลก

ในการโปรเจคเราแบ่งขั้นตอนหลักออกเป็น 2 ขั้นตอนคือ

ก) ทำการแปลงจุดบนคู่ลำดับโลก (x,y,z) ไปเป็นจุดที่สอดคล้องบนคู่ลำดับสายตา (x_e, y_e, z_e)

ข) ทำการแปลงจากจุดคู่ลำดับสายตา (x_e, y_e, z_e) ไปเป็นจุดที่สอดคล้องอยู่บนคู่ลำดับจอภาพ (x_s, y_s)

ในการพิจารณาการโปรเจควัตถุ เรากำหนดจุดมองอยู่ในระบบคู่ลำดับแบบทรงโค้ง คือ (D, ϕ, θ) ดังที่ได้กล่าวมาแล้ว โดยมีขั้นตอนดังนี้ (ดูรูป 2.11 ประกอบ)

ขั้นที่ 1 การแปลงทรานสเรชั่น จุดกำเนิดในระบบคู่ลำดับโลกไปยังจุด O_e (จุดกำเนิดของคู่ลำดับสายตา) เพื่อสร้างระบบแกนใหม่ที่จุดมอง ดังรูป 2.11 (a) การแปลงใช้ T_1 โดย

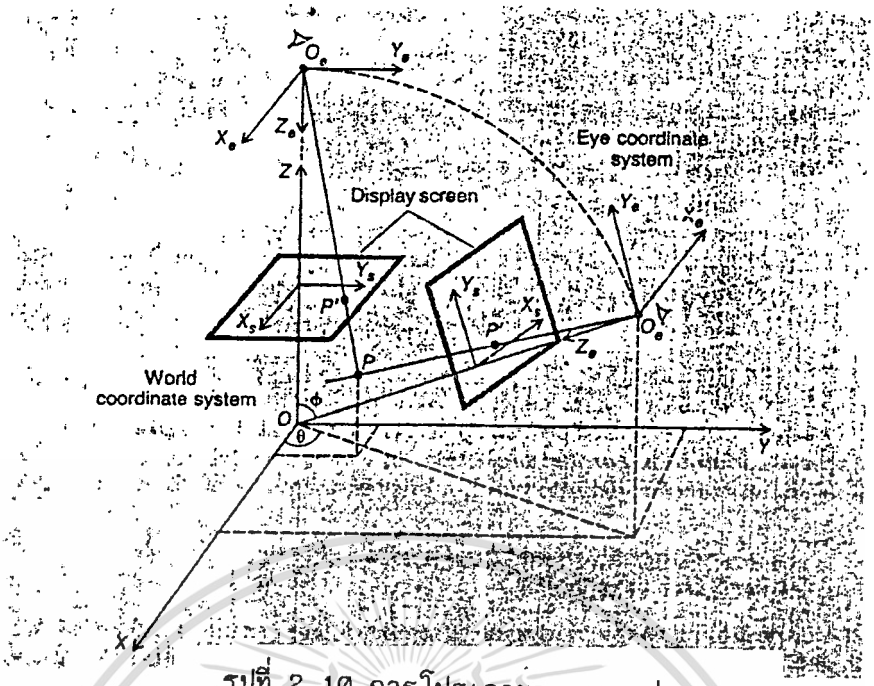
$$T_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -D\sin\phi\cos\theta & -D\sin\phi\cos\theta & -D\cos\phi & 1 \end{bmatrix}$$

ขั้นตอนที่ 2,3 และ 4 ทำการแปลงระบบคู่ลำดับใหม่ไปเป็นระบบคู่ลำดับสายตา โดยเริ่มจาก การแปลงแบบหมุนแกน z' ตามเข็มนาฬิกา ดังรูป 2.11 (b) ด้วยมุมเท่ากับ $(90 - \theta)$ จากสมการ 2.25 ได้ T_2

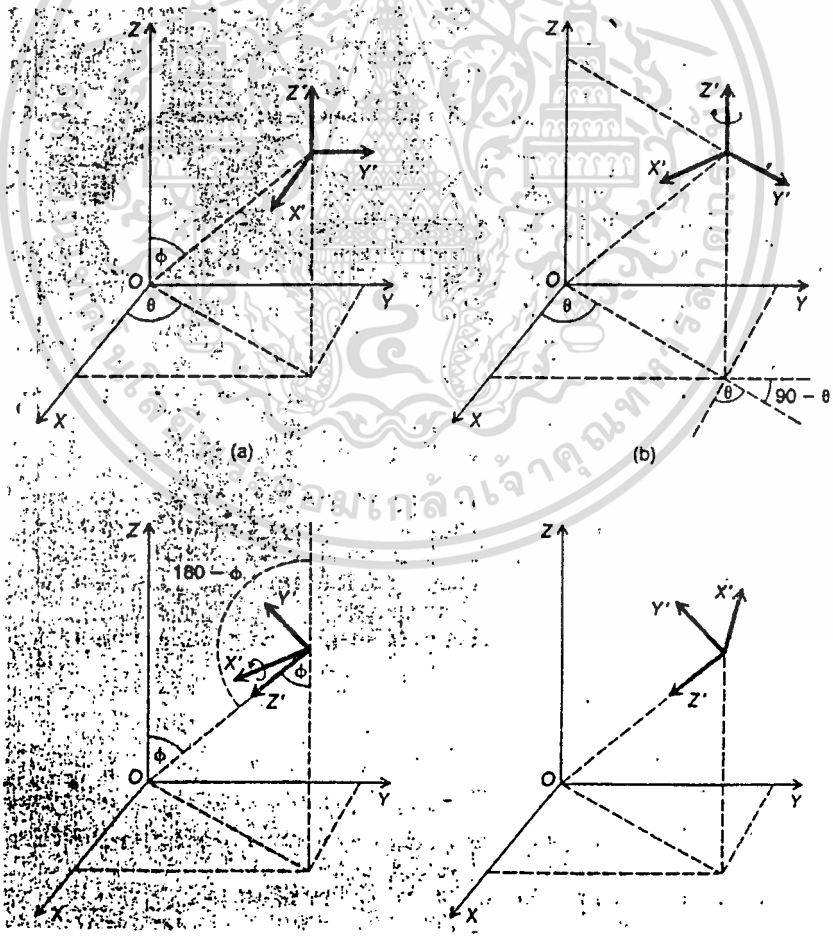
$$T_2 = \begin{bmatrix} \sin\theta & \cos\theta & 0 & 0 \\ -\cos\theta & \sin\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

จากนั้นทำการแปลงการหมุนในแนวแกน x ทวนเข็มนาฬิกาเป็นมุม $180 - \phi$ ดังรูป 2.11 (c) จากสมการ (2.26) ได้ T_3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 การโปรเจคของจุด P สู่จอภาพ



รูปที่ 2.11 4 ขั้นตอนในการแปลงการมอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$T_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -\cos\phi & -\sin\phi & 0 \\ 0 & \sin\phi & -\cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

ขั้นตอนที่ 4 ขั้นตอนสุดท้าย ทำการเปลี่ยนไปเป็นระบบมือซ้าย โดยการทำการกลับทิศ แกน x ได้ T_4 คือ (ดังรูป 2.11(d))

$$T_4 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

จากขั้นตอนทั้งหมด เราจะได้ผลลัพธ์ของการแปลงการมอง (viewing - transformation) คือ T โดย

$$T = T_1.T_2.T_3.T_4$$

$$= \begin{bmatrix} -\sin\theta & -\cos\theta\cos\phi & -\cos\theta\sin\phi & 0 \\ \cos\theta & -\sin\theta\cos\phi & -\sin\theta\sin\phi & 0 \\ 0 & \sin\phi & -\cos\phi & 0 \\ 0 & 0 & D & 0 \end{bmatrix} \qquad (2.28)$$

ทำให้ได้ผลลัพธ์ดังสมการ 2.29

$$(x_p, y_p, z_p, 1) = (x, y, z).T \qquad (2.29)$$

หรือ

$$x_p = -x\sin\theta + y\cos\theta$$

$$y_p = -x\cos\theta\cos\phi - y\sin\theta\cos\phi + z\sin\phi$$

$$z_p = -x\cos\theta\sin\phi - y\sin\theta\sin\phi - z\cos\phi + D$$

และเราสามารถหาค่า x_p, y_p ได้โดย

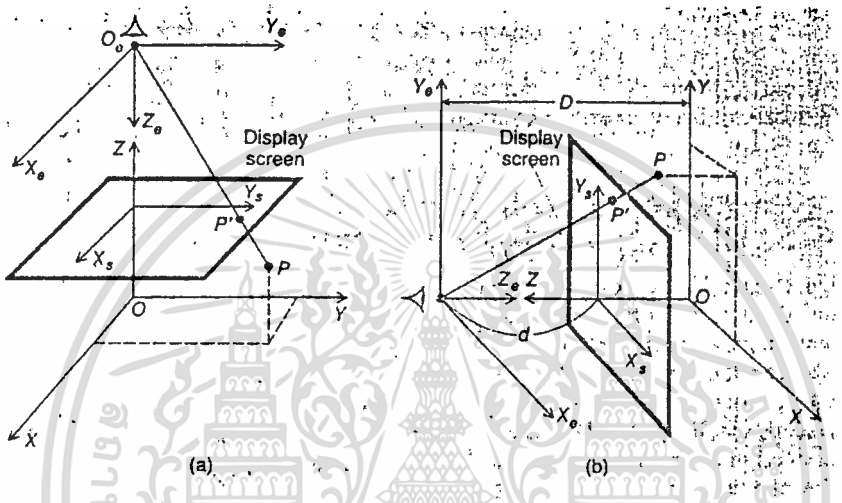
เอกสารนี้เป็นการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$x_s \doteq d(x_e/z_e) \quad , \qquad y_s = d(y_e/z_e) \qquad (2.30)$$

โดย d = ระยะห่างระหว่างคู่ลำดับจอภาพกับระบบคู่ลำดับสายตา

แต่ D = ระยะห่างระหว่างคู่ลำดับโลกกับระบบคู่ลำดับสายตา

สำหรับสมการ 2.30 ไม่ได้แสดงวิธีพิสูจน์ ผู้อื่นสามารถดูได้จากหนังสือ [1]



รูปที่ 2.12 การโปรเจคแบบขนาน

ความสัมพันธ์ระหว่างตัวแปร การกำหนด จุดมอง

มันเป็นสิ่งที่สำคัญในการทราบถึงความสัมพันธ์ หรือผลของค่า d, ϕ, θ และ D

-ค่า ϕ และ θ เป็นค่ามุมที่ใช้ในการกำหนดจุดมุม

-D คือค่าระยะจากจุดมองถึงวัตถุ โดยลากผ่านจุดกำเนิดของคู่ลำดับโลก การเปลี่ยนแปลงค่า D จะมีผลทำให้จุดมองเข้าใกล้ หรือ ออกห่างจากวัตถุ (ดูรูป 2.12 ประกอบ)

-ถ้าเราให้จุดมอง (D, ϕ, θ) มีค่าคงที่ และทำการเปลี่ยนแปลงค่า d จะมีผลต่อการเปลี่ยนขนาดของภาพบนจอภาพ เช่น ถ้าค่า d ลดลง (จอภาพ (screen) ใกล้ผู้มองเข้ามา) ขนาดของภาพบนจอจะลดลง

-เราพบว่าทั้ง D และ d จะควบคุมขนาดของภาพที่ปรากฏบนจอภาพ การปรับค่าทั้ง 2 ไปด้วยกันย่อมทำให้เกิดความยุ่งยาก ดังนั้น เพื่อแก้ปัญหานี้ เราจะเพิ่มตัวแปรเข้าควบคุมมันดัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อถัดไป

2.5.5 ระบบคู่ลำดับจอภาพเสมือน (homogeneous screen coordinate system)

เพื่อขจัดปัญหาการแปรค่าของ D กับ d เรากำหนด ตัวแปร S โดย S จะแทนศูนย์กลางขนาดจอ (centered screen size) โดยให้แกน z_e ผ่านจุดกลางของระบบ S นี้ ดังรูป 2.13

หน้าที่ของ S จะทำหน้าที่คล้ายเลนส์ในกล้องถ่ายรูป ได้ความสัมพันธ์คือ

$$x_e = (d/S)x_e / z_e \qquad (2.32)$$

$$y_e = (d/S)y_e / z_e \qquad (2.33)$$

สมมติให้จุดมองคงที่ ค่าอัตราส่วน d/S เท่านั้นที่จะควบคุมขนาดของภาพบนจอภาพ

จากที่กล่าวมาทั้งหมดเราสามารถสรุปค่า x_e, y_e ได้โดย จากสมการ 2.14,2.15,2.31 และ 2.32

$$x_e = SCF[\ (d/S)(x_e / z_e)v_x + L\ ] \qquad (2.34)$$

$$y_e = -(d/S)(y_e / z_e) \qquad (2.35)$$

เราพบว่าค่า v_x, v_y ทำหน้าเสมือนการกำหนดขอบเขตการมองของจอภาพ โดยปกติและ $v_x = v_y$ เนื่องจากเราคิดว่าระนาบจอภาพเป็นสี่เหลี่ยมจตุรัส

ค่า L,M คือค่าจุดศูนย์กลางของขอบเขตการมอง (viewport)

2.6 การทดสอบการตัดกันของเส้น (intersection test)

สำหรับคอมพิวเตอร์กราฟฟิคแล้ว นอกจากการวาดจุด (plot dots) แล้ว ยังประกอบด้วยการวาดเส้น (line) หรือลากเส้น จากวิทยานิพนธ์นี้จะพิจารณาถึงการตัดกันของเส้นตรง 2 เส้น (two line segment) พิจารณาสมการเส้นตรง

$$L : A_1x + B_1y + C_1 = 0 \qquad (A_1, B_1 \neq 0)$$

$$L' : A_2x + B_2y + C_2 = 0 \qquad (A_2, B_2 = 0)$$

เราพบว่าสิ่งที่เป็นไปได้สำหรับเส้นตรง 2 เส้น คือ

ก) เส้นตรง L และ L' ขนานกัน เราพบว่า ค่าดีเทอร์มิแนน (determinant) ของสัมประสิทธิ์เท่ากับ 0

เอกสารนี้ใช้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{vmatrix} A_1 & B_1 \\ A_2 & B_2 \end{vmatrix} = A_1B_2 - A_2B_1 = \emptyset$$

ข) เมื่อเส้นตรง 2 เส้นทับกันได้ $A_1/A_2 = B_1/B_2 = C_1/C_2$

ค) กรณีอื่นก็คือ เส้นตรง 2 เส้นตัดกัน สมมติให้ตัดกันที่จุด (x_1', y_1') เพราะฉะนั้นได้

$$x_1 = B_1C_2 - B_2C_1 / A_1B_2 - A_2B_1 \quad , \quad y_1 = C_1A_2 - C_2A_1 / A_1B_2 - A_2B_1$$

การทดสอบการตัดกันของเส้นขอบ (edge interxection test) ในการหาเราต้องการทราบว่า เส้นขอบใดบ้างที่ตัดกัน เส้นใดทับกัน และเส้นใดขนานกัน การทดสอบเริ่มจาก กำหนดเส้นตรง 2 เส้น และแต่ละเส้นมีคู่ลำดับ คือ (x_1, y_1) , (x_2, y_2) กับ (x_1', y_1') , (x_2', y_2') จากสมการเส้นตรง

$$y = y_1 + m(x - x_1) \qquad (2.36)$$

จากสมการ 2.36 ได้

$$m = (y_2 - y_1)/(x_2 - x_1) \qquad (2.37)$$

และ $$mx - y + (y_1 - x_1) = \emptyset \qquad (2.38)$$

สำหรับ (x_1', y_1') และ (x_2', y_2') ก็สามารถทำได้เช่นกัน ได้สมการ 2.39,2.40

$$m'x - y + (y_1' - x_1') = \emptyset \qquad (2.39)$$

และ $$m' = (y_2' - y_1')/(x_2' - x_1') \qquad (2.40)$$

เพื่อตรวจสอบการตัดกันของเส้นขอบ เราหาได้จากการคำนวณ ดีเทอร์มิแนน ของ m (slop)

$$\begin{vmatrix} m & -1 \\ m' & -1 \end{vmatrix} = m' - m$$

ถ้า $m'-m = \emptyset$ เราพบว่าเส้นขอบ 2 เส้นขนานกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้า $m/m' = 1 = (y_1-mx_1)/(y'_1-mx'_1)$ แสดงว่า เส้นขอบ 2 เส้น ทับกัน

ถ้า ไม่ได้กรณีทั้ง 2 กรณี ก็แสดงว่า เส้นตรง 2 เส้น ตัดกัน เราสามารถหา จุดตัดกันได้ (x_i,y_i)

โดย

$$x_i = ((y_1-y'_1) - (mx_1-mx'_1))/m'-m \qquad (2.41)$$

$$y_i = ((y_1-mx_1)m' - (y'_1-mx')m/m'-m \qquad (2.42)$$

2.7 การทดสอบว่าจุดอยู่ภายในหรือภายนอกรูปหลายเหลี่ยมที่กำหนด (containment test)

สำหรับการทดสอบว่าจุดที่กำหนดอยู่ในรูปหลายเหลี่ยม (polygons) หรือไม่นั้น มีด้วยกัน 2 วิธี คือ

- การทดสอบด้วยการคำนวณผลบวกของมุม (test by calculating a sum of angles)

และ - การทดสอบด้วยวิธีครึ่งระนาบ (test by the half-plane method)

สำหรับในวิทยานิพนธ์นี้ เราจะกล่าวถึงเฉพาะวิธีแรกเท่านั้น

2.7.1 การทดสอบด้วยการหาผลรวมของมุม

เรากำหนดจุดทดสอบมาให้ สมมติว่าคือจุด P_t ดังรูป 2.14 และกำหนดให้จุดยอดต่างๆ ของรูปหลายเหลี่ยม คือ $P_j = (x_j,y_j)$ โดย $j = 1,...n$ โดย $P_n = P_1$ เรากำหนดให้ลำดับของจุด เส้นตรงเป็นไปในทิศทวนเข็มนาฬิกา เราสามารถหาค่ามุมระหว่างเส้นตรง $\overline{P_tP_j}$ กับ $\overline{P_tP_{j+1}}$ โดยวัดจาก $\overline{P_tP_j}$ ไป เส้นตรง $\overline{P_tP_{j+1}}$ กำหนดให้เป็น α_j เราพบว่า

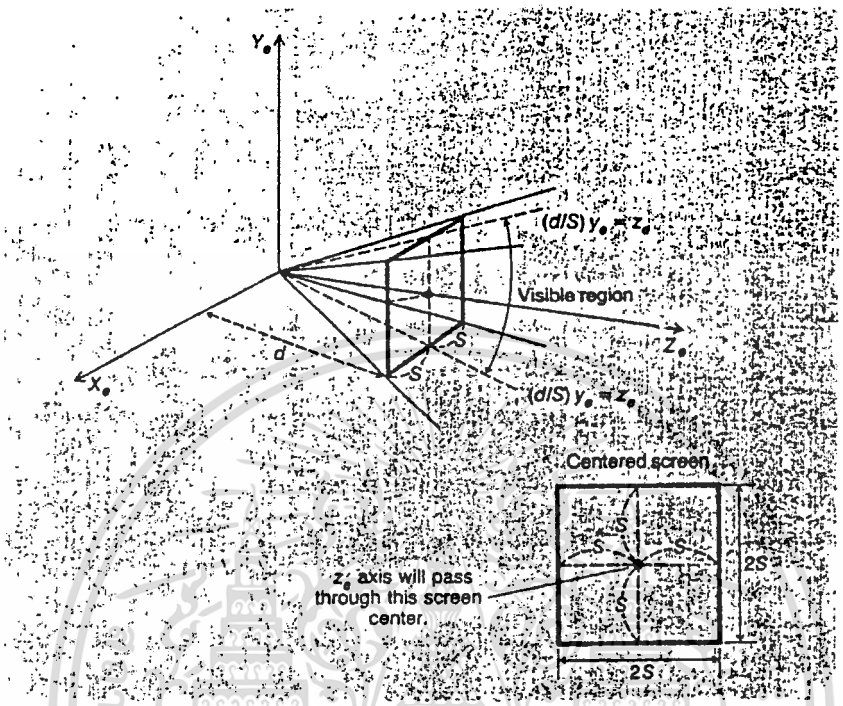
- ถ้า ผลรวของมุม $\alpha_j = 0$ ($\Sigma\alpha_j = 0$) แสดงว่าจุด P_t อยู่นอกรูปหลายเหลี่ยม
- ถ้า ผลรวของมุม $\alpha_j = 360$ ($\Sigma\alpha_j = 360$) แสดงว่าจุด P_t อยู่ในรูปหลายเหลี่ยม

จากรูป ตัวอย่าง 2.14 (a) เราสามารถหามุม ได้จาก

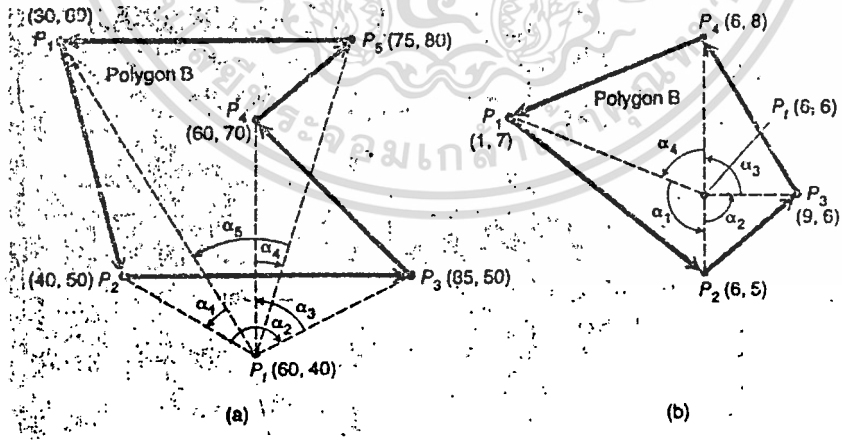
$$\overline{P_tP_1} = (30,80) - (60,40) = (-30,40)$$

$$\overline{P_tP_2} = (40,50) - (60,40) = (-20,10)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 ปิรามิดการมอง



รูปที่ 2.14 ตัวอย่างการทดสอบตำแหน่งของจุดแบบหาผลรวมของมุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หาขนาด $\overline{P_tP_1} = (-30)^2 + (40)^2 = 50$

$\overline{P_tP_2} = (-10)^2 + (10)^2 = 22.36$

จาก ทฤษฎี ผลคูณสเกลล่า (the dot product of vectors)

ได้ $\overline{P_tP_1} \cdot \overline{P_tP_2} = (-30,40).(-20,10) = 1000$

ผล $\overline{P_tP_1} \cdot \overline{P_tP_2} = (\overline{P_tP_1} \cdot \overline{P_tP_2} \cos\alpha)/|\overline{P_tP_1}| \cdot |\overline{P_tP_2}| =$

ดังนั้น $\alpha = \cos^{-1}(1000)/(50)(22.36) = 26.56$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

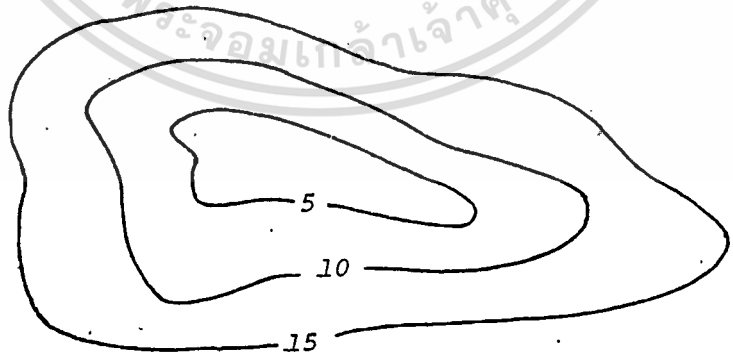
บทที่ 3

แนวความคิดพื้นฐานสำหรับระบบ 3 มิติทางภูมิศาสตร์

สำหรับวิทยานิพนธ์ฉบับนี้เราได้ให้แนวความคิดในการทำแผนที่ แสดงเส้นชั้นความสูง มาทำการจำลองเป็นภาพภูมิศาสตร์ 3 มิติ เราขอกล่าวถึงคุณสมบัติหาเส้นชั้นความสูง (contour line) และวิธีการสร้างเส้นชั้นความสูงในวิชาการสำรวจ (survey engineer)

3.1 เส้นชั้นความสูง (contour line)

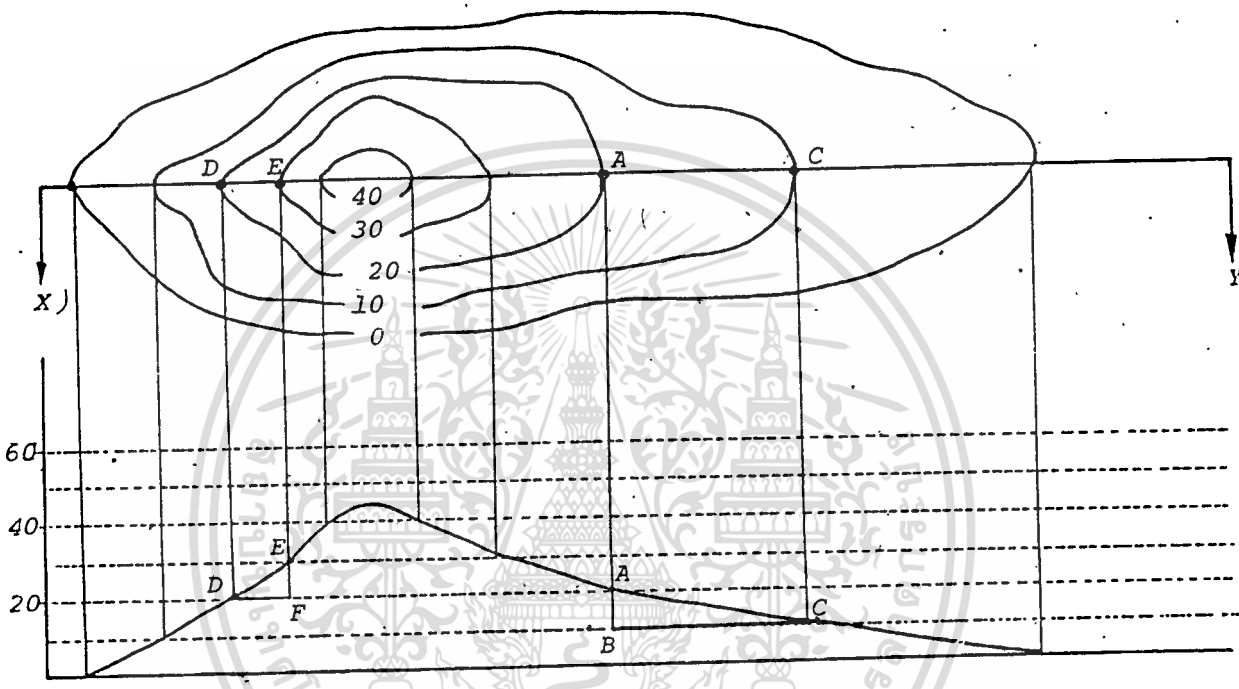
หมายถึง เส้นสมมติที่ลากผ่านจุดบนพื้นดินซึ่งมีค่าระดับเท่ากัน อาจอยู่สูงหรือต่ำกว่าฐานระดับ ลักษณะของเส้นชั้นความสูงสามารถจินจนาการได้จากระดับน้ำในอ่างเก็บน้ำ เมื่อผิวน้ำสงบราบเรียบ เส้นขอบผิวน้ำรอบอ่างเก็บน้ำจะแสดงเส้นชั้นความสูง เนื่องจากผิวน้ำมีค่าระดับเท่ากัน ถ้าหากระดับน้ำลดลง สมมติว่าลดลงอีก 5 เมตร เส้นขอบผิวน้ำรอบอ่างเก็บน้ำก็จะแสดงเส้นชั้นความสูงอีกเส้นหนึ่ง ซึ่งอยู่ระดับต่ำกว่าเส้นแรก 5 เมตร ในทำนองเดียวกัน ถ้าหากระดับน้ำลดลงเรื่อยๆ ที่แต่ละระดับซึ่งน้ำลด เส้นขอบผิวน้ำรอบอ่างเก็บน้ำ ก็จะแทนเส้นชั้นความสูงที่ระดับนั้น เส้นชั้นความสูงเป็นเส้นต่อเนื่องบรรจบครบรอบ และไม่ตัดหรือชนกับเส้นชั้นความสูงเส้นอื่น นอกเสียจากกรณีหน้าผาหรือชะง่อนเขา ลักษณะเส้นชั้นความสูงดังแสดงในรูป 3.1



รูปที่ 3.1 แสดงชั้นความสูงบริเวณที่เป็นแอ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.2 แสดงระนาบราบของเส้นชั้นความสูง และรูปตัดด้านข้างของเกาะ โดยใช้ระดับน้ำทะเลล้อมรอบเกาะเป็นฐานระดับมีค่าระดับเท่ากับ ๐ ม. ช่วงชั้นความสูง (contour interval) เท่ากับ 10 ม. หมายถึง ค่าต่างระดับระหว่างเส้นชั้นความสูงข้างเคียงมีค่า 10 ม.



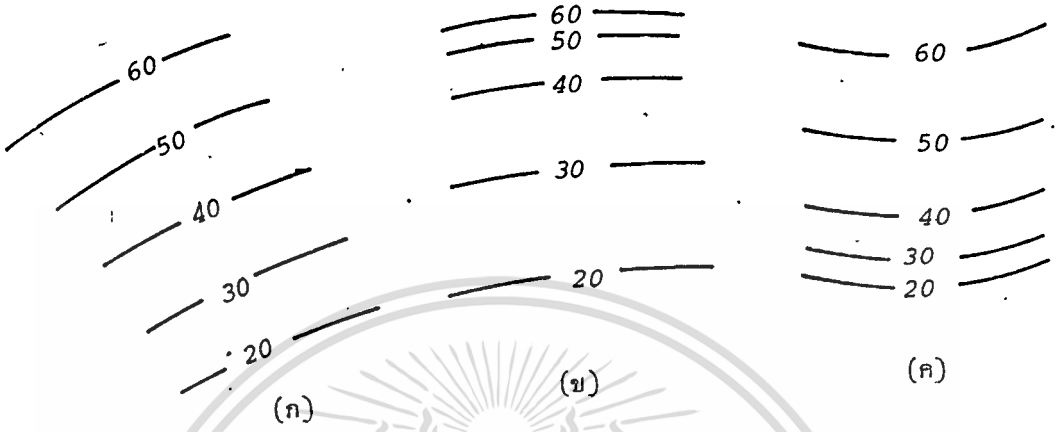
รูปที่ 3.2 รูปตัดตามแนว XY

ช่วงชั้นความสูง (contour interval) หมายถึง ความสูงระหว่างเส้นชั้นความสูงข้างเคียง ซึ่งใช้แสดงค่าต่างระดับระหว่างเส้นชั้นความสูงและมักนิยมใช้ค่าคงที่บนแผนที่นั้นๆ จากรูปที่ 3.2 ช่วงชั้นความสูงเท่ากับ 10 ม. ระยะ AB แสดงช่วงชั้นความสูง (contour interval) หรืออาจเรียก ช่วงแนวดิ่ง (vertical interval) ระยะทางตามแนวราบระหว่างเส้นชั้นความสูงทั้งสองนั้นเรียกว่า สมดุลแนวราบ (horizontal equivalent) ซึ่งแสดงด้วยระยะ BC

3.2 ลักษณะเส้นชั้นความสูง (contour characteristics)

จากตัวอย่างข้างต้นจะเห็นว่าอัตรลาดเอียงที่ชัน (steep) เส้นชั้นความสูงจะอยู่ใกล้กัน ในทางตรงกันข้าพื้นที่ราบ (flat) เส้นชั้นความสูงจะอยู่ห่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 แสดงลักษณะของเส้นชั้นความสูง

จากรูปที่ 3.3 แสดงความลาดเอียงที่แตกต่างกัน รูปที่ 3.3 (ก) แสดงความลาดเอียงที่มีอัตราลาดเอียงสม่ำเสมอ รูปที่ 3.3 (ข) ส่วนบนเส้นชั้นความสูงอยู่ใกล้กันมากกว่าส่วนที่ 3.3 (ค) เป็นความลาดเอียงแบบนูน (convex slope)

รูปที่ 3.4 แสดงลักษณะของเส้นชั้นความสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.4 (ก) แสดงลักษณะเส้นชั้นความสูงห้วงแม่น้ำ (river valley) จะมีลักษณะเหมือนรูปตัววี (v-shape) โดยปลายตัววีชี้ไปยังแหล่งต้นน้ำ เช่น ยอดเขา ส่วนรูปที่ 3.4 (ข) เส้นชั้นความสูงมีรูปตัววีแต่ชี้ลงปลายเขา แสดงทิวเขาที่ยื่นแหลมออกมา (nose or spur)

ลักษณะทั่วๆไปของเส้นชั้นความสูงสรุปได้ดังนี้

- เส้นชั้นความสูงที่มีระยะห่างกันมากและมีขนาดระยะห่างเท่า ๆ กัน แสดงว่าบริเวณนั้นเป็นพื้นที่ลาดน้อยๆและสม่ำเสมอ

-เส้นชั้นความสูงที่มีระยะชิดกันและมีระยะห่างเท่าๆกัน แสดงว่าบริเวณนั้นเป็นพื้นที่บาดชันแบะสม่ำเสมอ

-เส้นชั้นความสูงที่มีระยะชิดกันตอนบน และห่างมากขึ้นในตอนล่าง แสดงว่าพื้นที่บริเวณนั้นเป็นพื้นที่ลาดแบบเว้า

-เส้นชั้นความสูงที่มีระยะห่างกันในตอนบน และชิดกันมากขึ้นในตอนล่าง แสดงว่าพื้นที่บริเวณนั้นเป็นพื้นที่ลาดแบบนูน

-เส้นชั้นความสูงที่วงรอบยอดภูเขาสองยอดจะแสดงให้เห็นตำแหน่งของคอเขา เป็นส่วนที่ต่ำสุดของแนวเส้นเขา ซึ่งอยู่ระหว่างยอดเขาสองยอด ดังรูป 3.5

-เส้นชั้นความสูงที่มีลักษณะคล้ายรูปตัวยู (u-shape) ซ้อนๆกัน แสดงว่าบริเวณนั้นเป็นสันเนินหรือสันเขา ดังรูป 3.6

-เส้นชั้นความสูงที่มีลักษณะซ้อนกันเป็นเส้นเดียวแสดงว่า บริเวณนั้นเป็นหน้าผาชัน และเส้นตั้งฉากสั้นๆ แสดงบริเวณที่ยุบต่ำลง โดยปลายเส้นตั้งฉากจะชี้ไปยังที่ต่ำกว่า ดังรูป 3.7

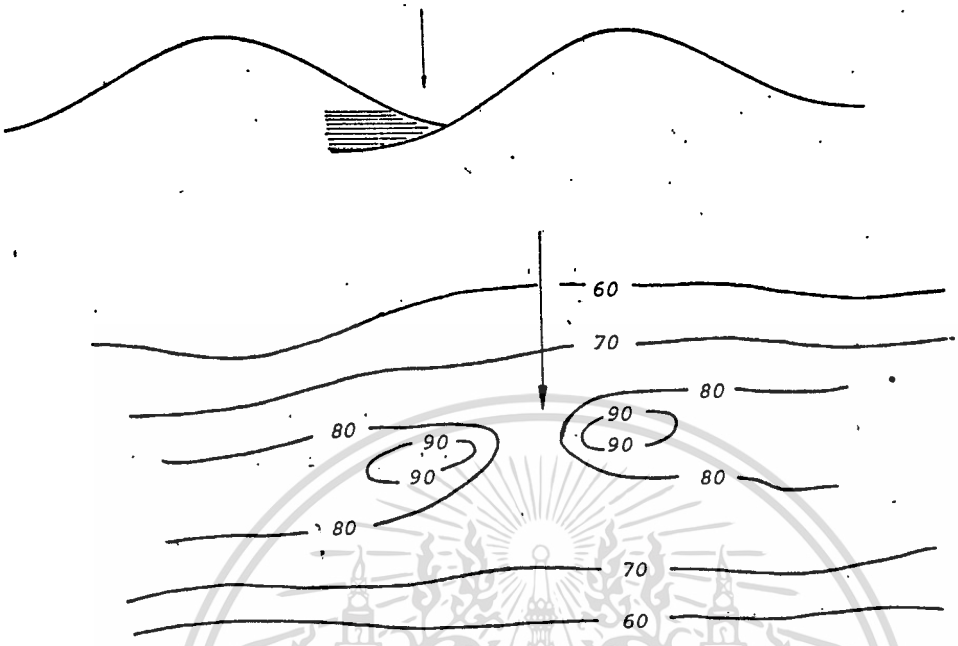
-เส้นชั้นความสูงที่มีรูปร่างคล้ายตัววีซ้อนๆกัน แสดงว่าบริเวณนั้นเป็นทางน้ำไหล ก้นตัววีจะชี้ไปในทิศทางที่มีความสูงมากกว่า ดังรูป 3.8

3.3 วิธีการทำเส้นชั้นความสูง (method of contouring)

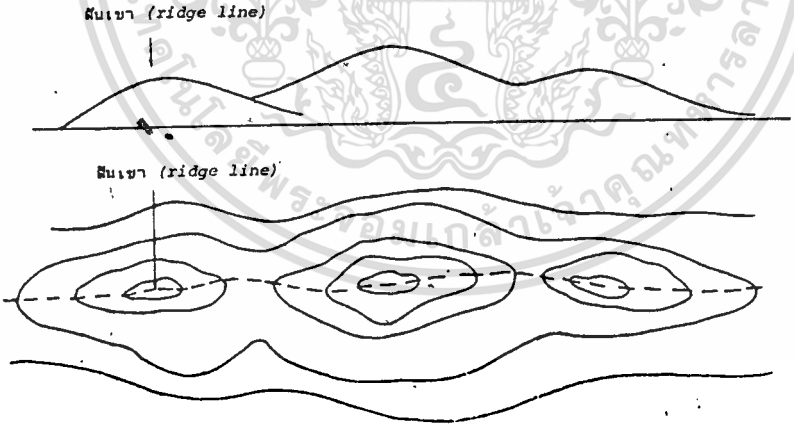
ซึ่งมีหลายวิธีที่เราเสนอในวิทยานิพนธ์ฉบับนี้ถือวิธีการทำเส้นชั้นความสูง แบบวิธีอ้อม (indirect method)

1)วิธีการทำเส้นชั้นความสูง แบบวิธีอ้อม (indirect method) เป็นวิธีการทำเส้นชั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

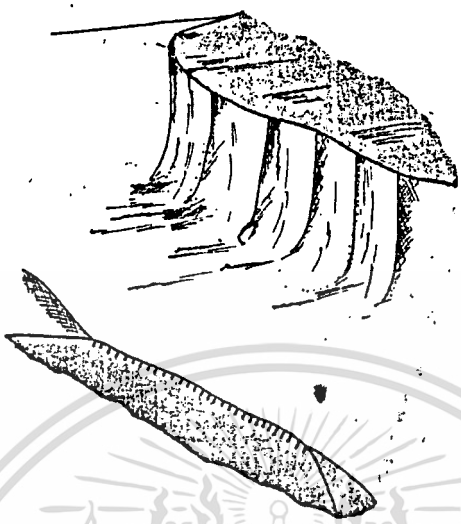


รูปที่ 3.5

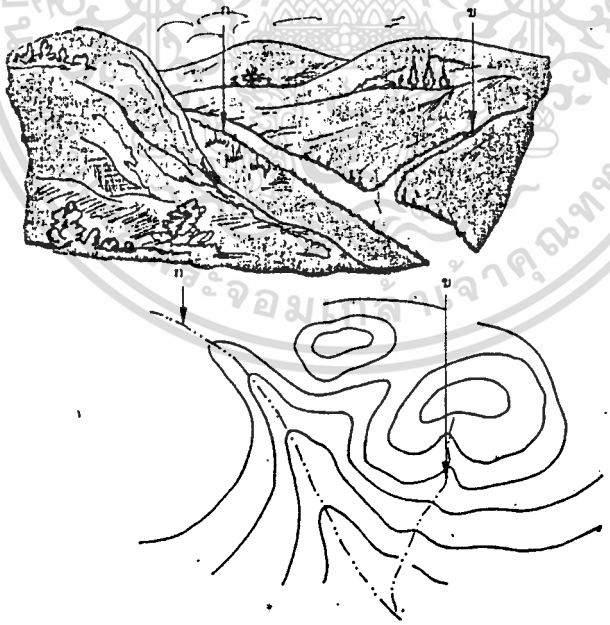


รูปที่ 3.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7



รูปที่ 3.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความสูงจากชุดของระดับจุด (series of spot levels) ซึ่งระดับจุด (spot levels) หมายถึงระดับตามจุดที่มีผลต่อการเขียนเส้นชั้นความสูง เช่น จุดเปลี่ยนค่าความลาดเอียงของพื้นดิน จุดสูงสุดหรือจุดต่ำสุดในพื้นที่ และระดับตามแนวร่องน้ำ ตำแหน่งของเส้นชั้นความสูงหาได้ด้วยการประมาณภายใน (interpolation)

วิธีการทำเส้นชั้นความสูงแบบวิดธีอ้อมมี 3 ขั้นตอน คือ

ก) สร้างตาราง (setting out a grid)

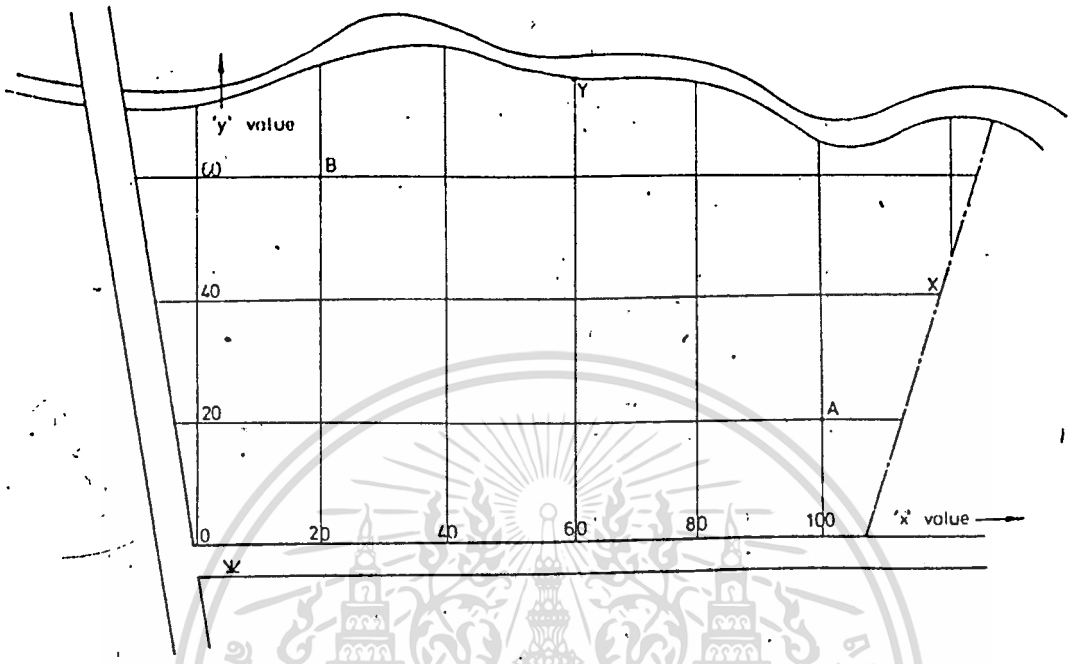
สำหรับสถานที่ดังรูปใน 3.9 เลือกด้านที่ยาวที่สุดให้เป็นเส้นฐานจากนั้นปักหมุดทุกช่วงระยะ 20 ม. ตามแนวเส้นฐานนั้น ช่วงการปักหมุดนี้ขึ้นอยู่กับช่วงชั้นความสูง (contour vertical interval) โดยทั่วไปแล้วควรปักหมุดตามแนวเส้นฐาน (baseline) ทุกช่วง 5 หรือ 10 ม. เมื่อต้องการช่วงชั้นความสูง 0.5 ม. หรือ 1 ม. การปักหมุดนี้ควรใช้ช่วงห่างมากที่สุดเพียง 20 ม.

ที่ปลายทั้งสองข้างของเส้นฐานลากเส้นตั้งฉากโดยการใช้แถบวัดระยะ (tape) หรือเครื่องส่องฉาก (prism square) ก็ได้ ปักหมุดไม้ทุกช่วง 20 ม. ตามแนวเส้นตั้งฉากทั้งสอง จากนั้นวัดระยะระหว่างหมุดไม้แต่ละคู่ของเส้นตั้งฉากทั้งสองในแนวขนานกับเส้นฐานต้องมีระยะเท่ากับระยะเส้นฐานเป็นการตรวจสอบ แล้วทำการปักหมุดไม้บนเส้นขนานกับเส้นฐานทุกเส้นโดยใช้ช่วงห่างทุก 20 ม. จะได้ตารางสี่เหลี่ยมจัตุรัสที่มีด้านยาวด้านละ 20 ม. หมุดไม้จะแทนมุมของรูปสี่เหลี่ยมจัตุรัส

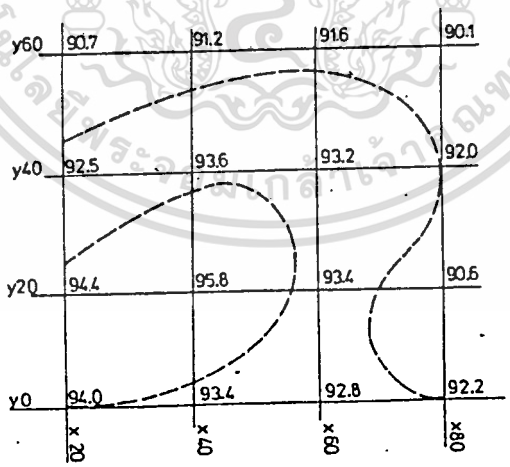
หมายเลขของหมุดไม้กำหนดชื่อตามแกน x และแกน y ดังในรูปที่ 3.9 หมุด A ใช้ชื่อย่อว่า (x 100 y 20) หมุด B ใช้ชื่อว่า (x 20 y 60) ตำแหน่งของจุดต่างๆในบริเวณกำหนดได้ด้วยจุดตัดของระยะตามแกน x กับระยะตามแกน y เช่น จุด x กำหนดได้จาก (x 116 y 40) จุด y กำหนดได้จาก (x 60 y 75) ซึ่งเป็นระบบกำหนดชื่อที่สะดวกง่าย

ข) ทำระดับ (levelling) เพื่อหาค่าระดับของหมุดไม้ทุกหมุดในตารางซึ่งปักไว้ในข้อ ก) รูปที่ 3.10 แสดงค่าระดับของหมุดไม้ที่ปักไว้เป็นตารางของบริเวณก่อสร้างซึ่งหาได้จากการทำระดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 ตารางกริด

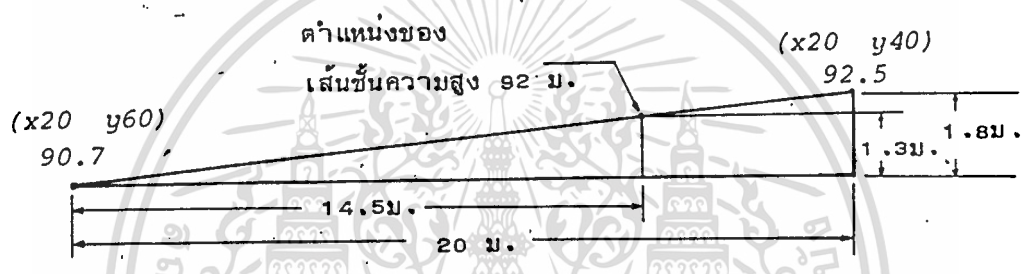


รูปที่ 3.10 ตัวอย่างการประมาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค) การประมาณภายในเพื่อเขียนเส้นชั้นความสูง (interpolating the contours) ซึ่งเป็นการประมาณค่าละเอียดของตารางคณิตศาสตร์โดยใช้ค่าต่างของช่วงตารางและสัดส่วนการเปลี่ยนค่ามี 2 วิธีคือ

1) ทางคณิตศาสตร์ (mathematically) โดยใช้การคำนวณจากสัดส่วนการเปลี่ยนค่า เช่น เส้นชั้นความสูง 92 ม. ในรูปที่ 3.10 จะต้องลากผ่านระหว่างหมุดหรือสถานี (x 20 y 40) กับหมุด (x 20 y 60)



รูปที่ 3.11

จากรูปที่ 3.17 คำนวณหาตำแหน่งของเส้นชั้นความสูง 92 ม. ในทางแนวราบได้ดังนี้

ระยะตามแนวราบระหว่างหมุดไม้ = 20 ม.

ค่าต่างระดับระหว่างหมุดไม้ = (92.5-90.7) ม. = 1.8 ม.

ค่าต่างระดับระหว่างหมุด (20,60) กับเส้นชั้นความสูง 92 ม. เท่ากับ

= (92-90.7) ม. = 1.3 ม.

ระยะทางตามแนวราบจากหมุด (20,60) ไปยังเส้นชั้นความสูง 92 ม. หาได้จากสัดส่วน

= (1.3/1.8 * 20)ม. = 14.44 ม.

การประมาณภายใน (interpolation) หาตำแหน่งของจุดต่างๆบนเส้นชั้นความสูง 92 ม. จากค่าระดับของหมุดตาราง (grid stations) สามารถคำนวณหาได้ดังแสดงในตารางที่ 3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Col.1	Col.2	Col.3	Col.4	Col.5	Col.6	Col.7
Grid Station	Grid Station	Contour Value (m)	Dist.between Col.1 & Col.2	Diff.in level Col.1 & Col.2	Diff.in level Col.1 & Col.3	Dist.between Col.1 & Col.3
x20 y60	x20 y40	92	20	1.8	1.3	(1.3/1.8)×20 = 14.44
x40 y60	x40 y40	92	20	2.4	0.8	(0.8/2.4)×20 = 6.67
x60 y60	x60 y40	92	20	1.6	0.4	(0.4/1.6)×20 = 5.00
x60 y20	x80 y20	92	20	2.8	1.4	(1.4/2.8)×20 = 10.00
x80 y00	x80 y20	92	20	1.6	0.2	(0.2/1.6)×20 = 2.50

ตารางที่ 3.1

2) ทางเขียนรูป (graphs) ซึ่งจะไม่กล่าวในที่นี้ ผู้สนใจสามารถหาอ่านจากหนังสือทางด้านวิศวกรรมสำรวจ [5]

3.4 การหาจุดกริดจากเส้นชั้นความสูงโดยใช้วิธีการประมาณทางคณิตศาสตร์ (regenerating grid point by approximating method)

ในการประมาณหาจุดกริด โดยอาศัยเส้นชั้นความสูงมีอยู่หลายวิธีด้วยกันแต่ในที่นี้เราเสนอวิธีการประมาณที่เรียกกันว่า อินเทอโปเลชั่น เนื่องจากในการจุดกริดในเส้นชั้นความสูงที่นำมานั้น จุดกริดในแต่ละแนวเส้น เส้นในแนวแกน y มีค่าความสูงที่เปลี่ยนแปลงไม่แน่นอน เราไม่สามารถหาสูตรหรือสมการทางคณิตศาสตร์มาอธิบายได้ ซึ่งในกรณีเช่นนี้วิธีการทางอินเทอโปเลชั่นดูจะเหมาะสมมาก

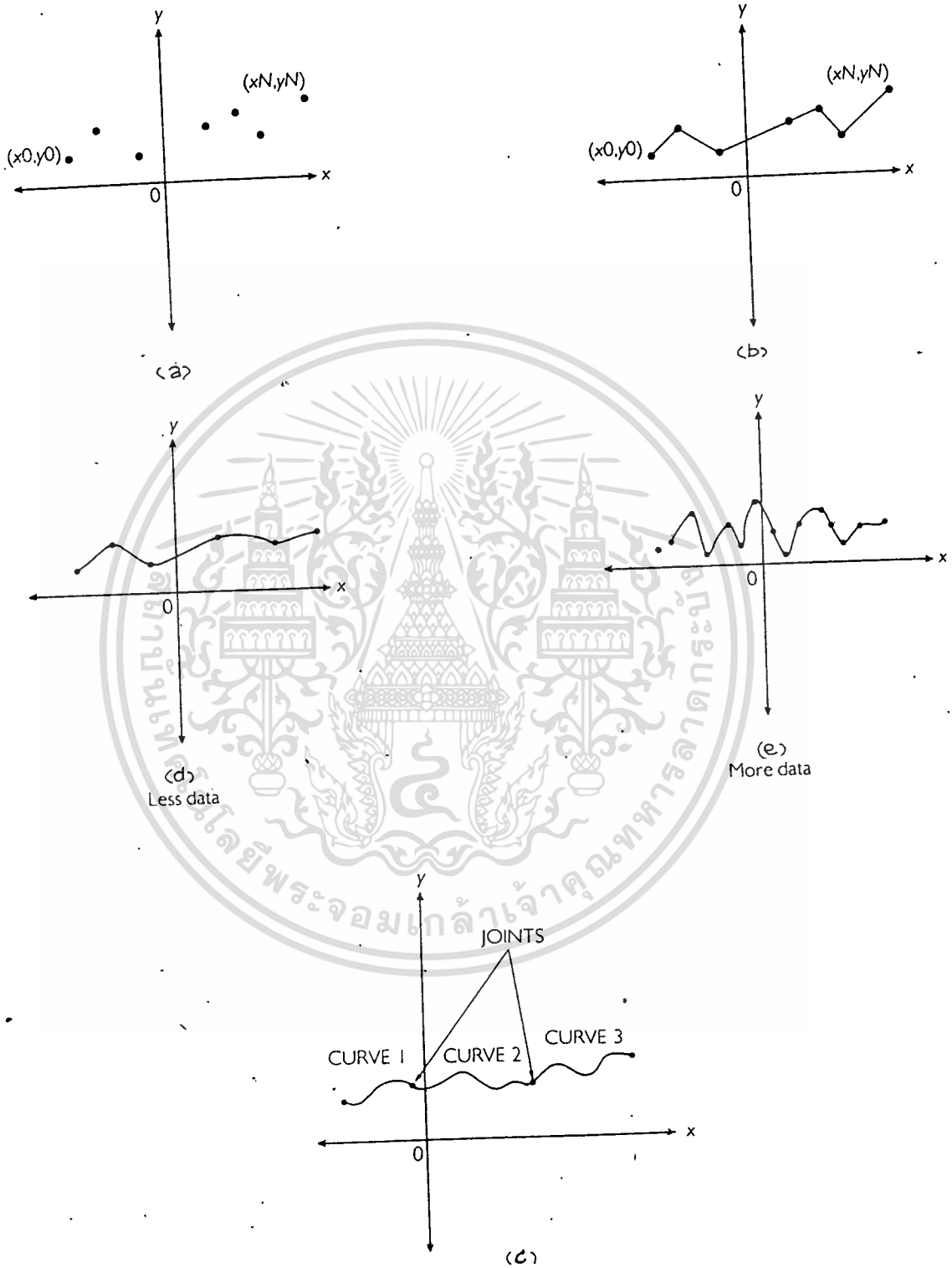
ลักษณะการประมาณแบบอินเทอร์โปเลชั่น เป็นการประมาณโดยใช้เส้นโค้งเรียบ (smooth curve) ซึ่งเป็นคุณสมบัติของการประมาณโดยใช้สมการทางโดพลีโนเมียบ (polynomial) เหตุผลที่มีการประมาณนั้น พิจารณาจากรูป 3.1(a) ซึ่งแสดงถึงความไม่ต่อเนื่องของข้อมูล ตั้งแต่ (x_o,y_o) ถึง(xn,y_n) ดังสมการ 3.1

$$f(x) = y_i \quad \text{if } x = x_i \text{ , } i=0,\ldots,n \qquad \text{สมการที่ 3.1}$$

$$= 0 \quad \text{สำหรับกรณีอื่น}$$

เราอาจประมาณโดยใช้การลากเส้นตรงระหว่างจุด 2 จุด และหาค่าจุดที่ต้องการระหว่าง 2 จุดนั้น ดังรูป 3.12(b) แต่พบว่าข้อมูลทางภูมิศาสตร์มีลักษณะที่มีการเปลี่ยนแปลงความสูงระ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หว่างจุด 2 จุดที่ทราบ จะมีลักษณะโค้งมากกว่าเป็นเส้นตรง ดังรูป 3.12(c)

3.4.1 การประมาณแบบลากรานส์อินเทอโปเลชั่น (largrange interpolation)

เป็นวิธีที่ง่ายที่สุดและนิยมใช้กันมาที่สุดวิธีหนึ่ง มีหลักการดังนี้ กำหนดให้มีกลุ่มของข้อมูล (ที่ได้จากการทดลอง เป็นต้น) อยู่ n ชุดคือ (x_i,y_i) เมื่อ $i = 1,2,\ldots,n$ ในวิธีลากรานจ์อินเทอโปเลชั่น เราจะประมาณฟังก์ชั่น $f(x_i)$ หรือ curve ที่ผ่านจุดเหล่านี้ ด้วยโพลีโนเมียบออร์เดอร์ (n-1) ดังนี้

$$P(x_i) = f(x_i) = y_i \quad , i=1,2,\ldots,n \tag{3.2}$$

โดยที่ P(x) จะเรียกว่าเป็นอินเทอโปเลติ้งโพลีโนเมียล และจะมีค่าเท่ากับ

$$P(x) = y_1L_1(x) + y_2L_2(x)+\ldots+ y_nL_n(x) \tag{3.3}$$

$L_i(x)$ ในสมการที่ (3.3) จะเป็นโพลีโนเมียลออร์เดอร์ (n-1) ดังนี้

$$L_i(x) = \prod_{\substack{j=1 \\ j\neq i}}^{n} \frac{x-x_j}{x_i-x_j} \tag{3.4}$$

$$= \frac{(x-x_1)(x-x_2)\ldots(x-x_{i-1})(x-x_{i+1})\ldots(x-x_n)}{(x_i-x_1)(x_i-x_2)\ldots(x_i-x_{i-1})(x_i-x_{i+1})\ldots(x_i-x_n)}$$

ที่จุดต่าง ๆ ของข้อมูล $x_1, x_2,\ldots,x_n$ $L_i(x)$ จะมีค่าเป็น

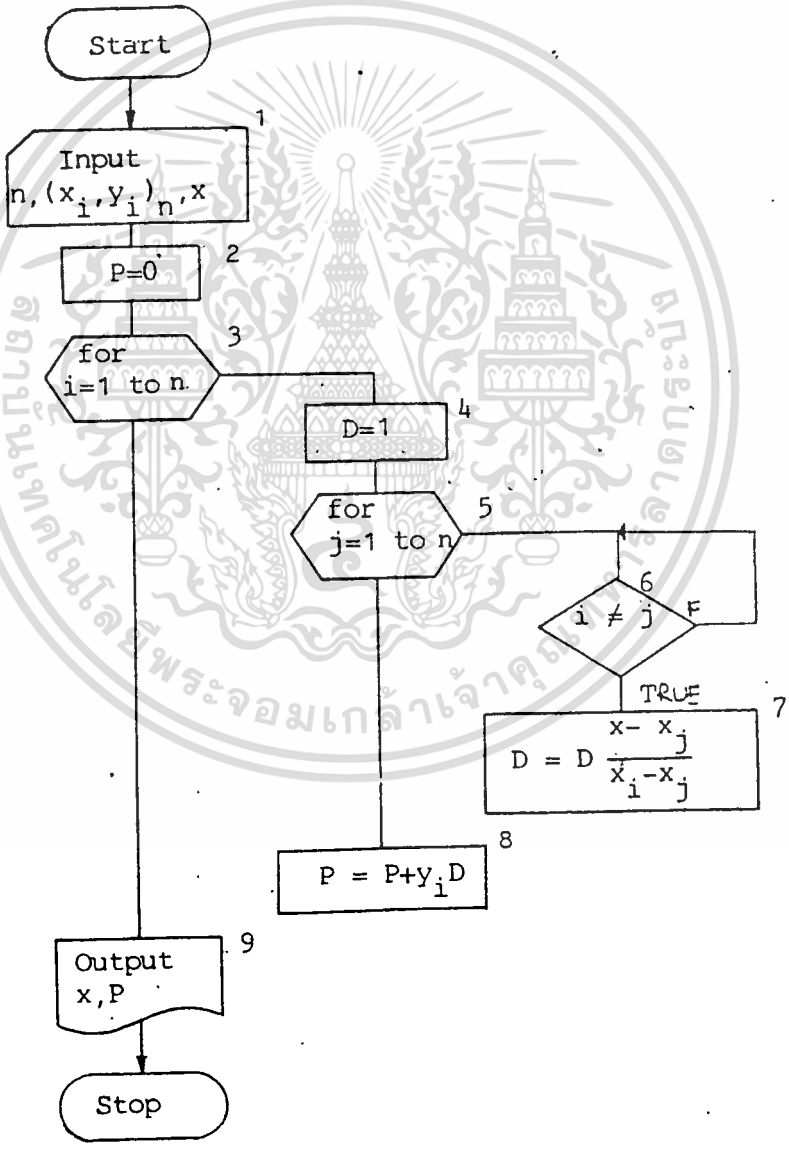
$$L_i(x_j) = \begin{cases} 1 , & i = j \\ 0 , & i \neq j \end{cases} \quad , i,j =1,2,\ldots,n \tag{3.5}$$

เมื่อแทนค่า $L_i(x)$ จากสมการที่ 3.4 ในสมการที่ 3.3 เราจะได้ลากรานจ์อินเทอโปเลติ้งโพลีโนเมียลออร์เดอร์ (n-1) ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$P(x) = \sum_{j=1}^{n} y_i L_i(x)$$

$$= \sum_{j=1}^{n} y_i \prod_{\substack{j=1 \\ j \neq i}}^{n} \frac{x-x_j}{x_i-x_j}$$



รูปที่ 3.13 โฟลท์ชาดลากรานส์อินเทอโปเรชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อดี ของการประมาณแบบนี้ เนื่องจากการประมาณแบบลากรานส์ เป็นการประมาณโดยใช้โพลีโนเมียล ขนาดออร์เดอร์ (order) เท่ากับ n-1 โดย n คือ จำนวนข้อมูลที่ทราบ(ที่ป้อนเข้ามา) เราพบว่าการประมาณแบบโพลีโนเมียล ยิ่งมีค่ามากเท่าใดความถูกต้องของการประมาณจะมีมากขึ้น แต่มีข้อกำหนดที่สำคัญคือ จำนวนข้อมูลที่ป้อนเข้ามาต้องไม่มากจนเกินไป พิจารณารูป 3.1 (d) เป็นรูปที่มีข้อมูลป้อนเข้ามาน้อยเหินไปทำให้การประมาณผิดพลาดมาก

ข้อเสีย เนื่องจากการประมาณแบบลากรานส์ใช้ขนาดออร์เดอร์เท่ากับจำนวนข้อมูลที่ป้อนเข้ามาลบด้วย1 (n-1) ถ้าขนาดออร์เดอร์มากเกินไป (n มากเกินไป) จะมีผลทำให้พฤติกรรมที่เรียกว่า การออสสิเลท (oscilatory behavior ซึ่งเป็นสิ่งที่เราไม่ต้องการ การเกิดพฤติกรรมเช่นนี้เกิดเนื่องจาก การที่เราประมาณโดยใช้ฟังก์ชันโพลีโนเมียล พิจารณาจากรูป 3.1(e) สำหรับผลของการเกิดพฤติกรรมแบบนี้ ได้แสดงไว้ในบทที่ 3 เรื่องการทดลอง

รูป 3.2 แสดงโฟลชาต ของลากรานส์อิเทอโปเลชั่น

3.4.2 คิวบิคสไปลน์อิเทอโปเลชั่น (cubic spline interpolation)

เพื่อแก้ปัญหานี้เราจะใช้วิธีพีซ-ไวสโพลีโนเมียลอินเทอโปเลชั่น (piece-wisepolynomial interpolation) แบบที่ง่ายที่สุดของวิธีนี้ก็คือจะเป็นเส้นตรงที่เชื่อมระหว่างจุดของข้อมูลสองจุดใดๆ คือ (x_i,y_i) และ (x_{i+1},y_{i+1}) วิธีนี้จะทำให้ได้ค่าประมาณที่ใกล้เคียงมากกว่าวิธีโพลีโนเมียลอินเทอโปเลชั่น แต่วิธีนี้ก็มีข้อเสียคือค่าประมาณที่ได้จะมีลักษณะไม่เรียบ และไม่มีความต่อเนื่องของเดอริเวทีฟที่แต่ละจุดของข้อมูล ซึ่งต่างกับฟังก์ชั่นในความเป็นจริงที่มีลักษณะเรียบ ดังนั้นเพื่อให้ได้ค่าประมาณที่มีลักษณะเรียบและต่อเนื่องที่แตะละจุดของข้อมูลด้วย เราจะใช้วิธีที่เรียกว่าคิวบิคสไปลน์ ในวิธีนี้ฟังก์ชั่นในระหว่างจุดของข้อมูล (x_i, x_{i+1}) จะถูกกำหนดด้วยโพลีโนเมียลออร์เดอร์ 3 คือ P(x) ซึ่งจะต้องมีคุณสมบัติดังนี้

1) P(x),P(x),P(x) มีลักษณะต่อเนื่องในระหว่าง (x_1,x_n)

2) $P(x_i) = f(x_i)$, $0<=i<=n$ นั่นคือ P(x) จะอินเทอโปเลท f(x) ในระหว่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

3) P(x) เป็นโพลีโนเมียลออร์เดอร์ 3 ในระหว่างจุดของข้อมูล (x_i, x_{i+1})

$0 <= i <= n-1$

ดิวบิคสไปลน์โพลีโนเมียล ในระหว่างจุดของข้อมูล (x_i, x_{i+1}) จะมีค่าดังนี้

$$P_i(x) = y_i + a_{1i}(x-x_i) + a_{2i}(x-x_i)^2 + a_{3i}(x-x_i)^3 \qquad 3.6$$

ค่าของสัมประสิทธิ์ a_{1i}, a_{2i}, a_{3i} จะหาได้จากสมการต่อไปนี้

$$\begin{bmatrix} 2(h_1+h_2) & h_2 & & & \\ h_2 & 2(h_2+h_3) & h_3 & & \\ & \cdots & \cdots & \cdots & \\ & & h_{n-2} & 2(h_{n-2}+h_{n-1}) & \end{bmatrix} \begin{bmatrix} a_{22} \\ a_{23} \\ \cdots \\ a_{2,n-1} \end{bmatrix}$$

$$= \begin{bmatrix} [y_2,y_3] - [y_1,y_2] \\ [y_3,y_4] - [y_2,y_3] \\ \cdots\cdots \\ [y_{n-1},y_n] - [y_{n-2},y_{n-1}] \end{bmatrix} \qquad 3.7$$

โดยที่ $a_{21} = a_{2n} = 0$

$$a_{3i} = \frac{1}{3h_i}(a_{2,i+1} - a_{2i}), \quad i = 1,2,\ldots,n-1 \qquad 3.8$$

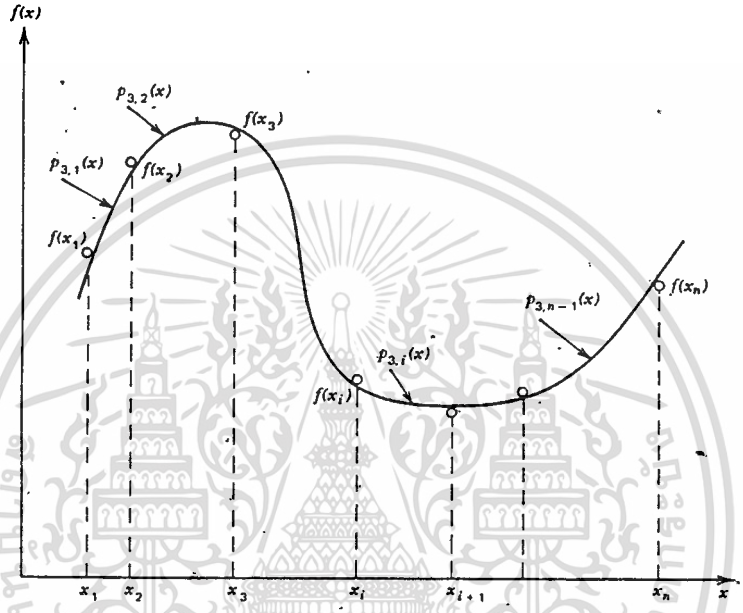
$$a_{1i} = [y_i,y_{i+1}] - \frac{h_i}{3}(2a_{2i} + a_{2,i+1}), \quad i = 1,2,\ldots,n-1 \qquad 3.9$$

และ

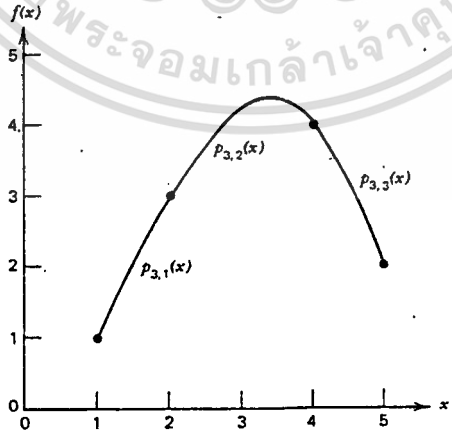
$$h_i = x_{i+1} - x_i, \quad i = 1,2,\ldots,n-1 \qquad 3.10$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อนึ่งการหาค่า $a_{22}, a_{23}, \ldots, a_{2,n-1}$ ในสมการ 3.7 อาจทำได้ง่ายๆ โดยใช้วิธีแพพพยกส่วนแบบปรับปรุงใหม่ของคลอเลสกี้ เมื่อได้ค่า $a_{22}, a_{23}, \ldots, a_{2,n-1}$ แล้วเราจะหาค่า P( ) ที่ x= ได้จากสมการที่ 3.8 คือ ถ้า. (x_i, x_{i+1}) เราจะได้ $P( ) = P_i( )$

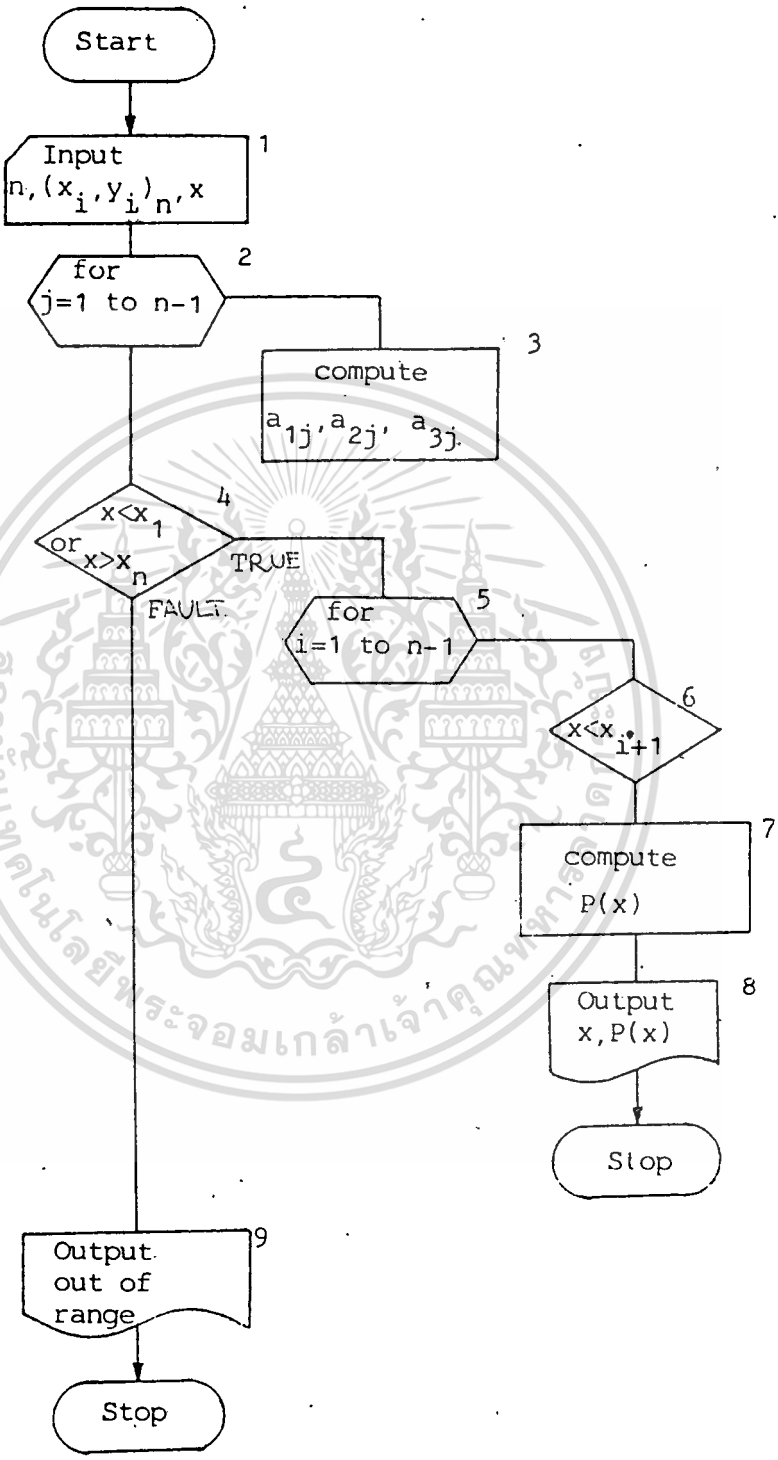


รูปที่ 3.15 รูปแบบการประมาณแบบคิวบิคสไปร์



รูปที่ 3.16 ตัวอย่างการประมาณแบบคิวบิคสไปร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 โพลท์ชาดคิวบิคสไปร์อินเทอโปเรชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 สมการระนาบ 3 มิติ (plane-equation)

เนื่องจากเราต้องมีความเกี่ยวข้องกับรูปหลายเหลี่ยม (สังเกตุตารางกริด) ซึ่งอยู่ในระบบ 3 มิติ ดังนั้นการเกี่ยวข้องกับสมการระนาบจึงเป็นสิ่งที่จำเป็นในการกำหนดตำแหน่งของรูปหลายเหลี่ยม จากสมการระนาบ 3.11

$$Ax+By+Cz+D = \emptyset \qquad (3.11)$$

เราสามารถหาสมการของระนาบได้โดยการพิจารณา ให้จุดบนระนาบ 3 จุด คือ $P_1(x_1,y_1,z_1), P_2(x_2,y_2,z_2), P_3(x_3,y_3,z_3)$ จากสมการ 3.11 ได้

$$\begin{aligned} Ax_1+By_1+Cz_1+D &= \emptyset \\ Ax_2+By_2+Cz_2+D &= \emptyset \\ Ax_3+By_3+Cz_3+D &= \emptyset \end{aligned} \qquad (3.12)$$

ถ้าจุดทั้ง 3 ไม่อยู่บนเส้นตรงเดียวกัน (not collinear) เราสามารถหาค่าสัมประสิทธิ์ A,B,C และ Dได้โดยการกำหนดค่าตัวสัมประสิทธิ์ตัวใดตัวหนึ่งให้ก่อน เช่น ให้ D=1 เราก็สามารถหาค่า A,Bและc ได้แต่ปัญหาจะเกิดขึ้นเมื่อค่า D เกิดเป็น Ø ขึ้นมา

วิธีการที่ดีกว่าอีกวิธีหนึ่งในการหาค่าสัมประสิทธิ์ A,B,Cและd โดยการกำหนด P_1, P_2, P_3 และจุด (x,y,z) เพิ่มขึ้นบนระนาบ ดังสมการ 3.13

$$\begin{aligned} Ax+By+Cz+D &= \emptyset \\ Ax_1+By_1+Cz_1+D &= \emptyset \\ Ax_2+By_2+Cz_2+D &= \emptyset \\ Ax_3+By_3+Cz_3+D &= \emptyset \end{aligned} \qquad (3.13)$$

โดยถ้าให้ผลลัพธ์ของสมการข้างต้นอยู่ในกลุ่มของสมการเสมือน (homogeneouts equation) ค่าดีเทอร์มิแนน ของสัมประสิทธิ์จะเป็น Ø ดังสมการ 3.14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

$$\begin{vmatrix} x & y & z & 1 \\ x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \end{vmatrix} = 0 \qquad (3.14)$$

ทำการขยายด้วยแฟคเตอร์ร่วม (expanding by cofactor) เราจะหาค่า A,B,C,D ได้ ดังสมการ 3.15 โดยเทียบกับสมการระนาบ

$$x\begin{vmatrix} y_1 & z_1 & 1 \\ y_2 & z_2 & 1 \\ y_3 & z_3 & 1 \end{vmatrix} - y\begin{vmatrix} x_1 & z_1 & 1 \\ x_2 & z_2 & 1 \\ x_3 & z_3 & 1 \end{vmatrix} + z\begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} - \begin{vmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{vmatrix} = 0$$

ค่าสัมประสิทธิ์ A,B,C,D จะรวมเอาเครื่องหมาย (+,-) เข้ามาด้วย เราสามารถหาค่า A ได้จากดีเทอร์มแนน โดย

$$A=\begin{vmatrix} y_1 & z_1 & 1 \\ y_2 & z_2 & 1 \\ y_3 & z_3 & 1 \end{vmatrix} = y_1(z_2-z_3)+ y_2(z_3-z_1)+ y_3(z_1-z_3) \qquad (3.16)$$

สำหรับค่าอื่นๆก็หาได้เช่นกัน สำหรับข้อจำกัดของสมการ 3.15 ก็คือ จุด P_1 , P_2 และ P_3 ต้องไม่อยู่บนเส้นเดียวกัน

3.6 การหาระยะระหว่างจุด 2 จุดในระบบ 3 มิติ

สมมติให้จุด P_1 (x_1,y_1,z_1) และ P_{21} (x_2,y_2,z_2) เป็นจุดในระบบ 3 มิติ เราสามารถหาระยะห่างระหว่างจุด 2 จุดได้จากสมการ 3.17

$$d = \sqrt{(x_1-x_2)^2+(y_1-y_2)^2+(z_1-z_2)^2}$$

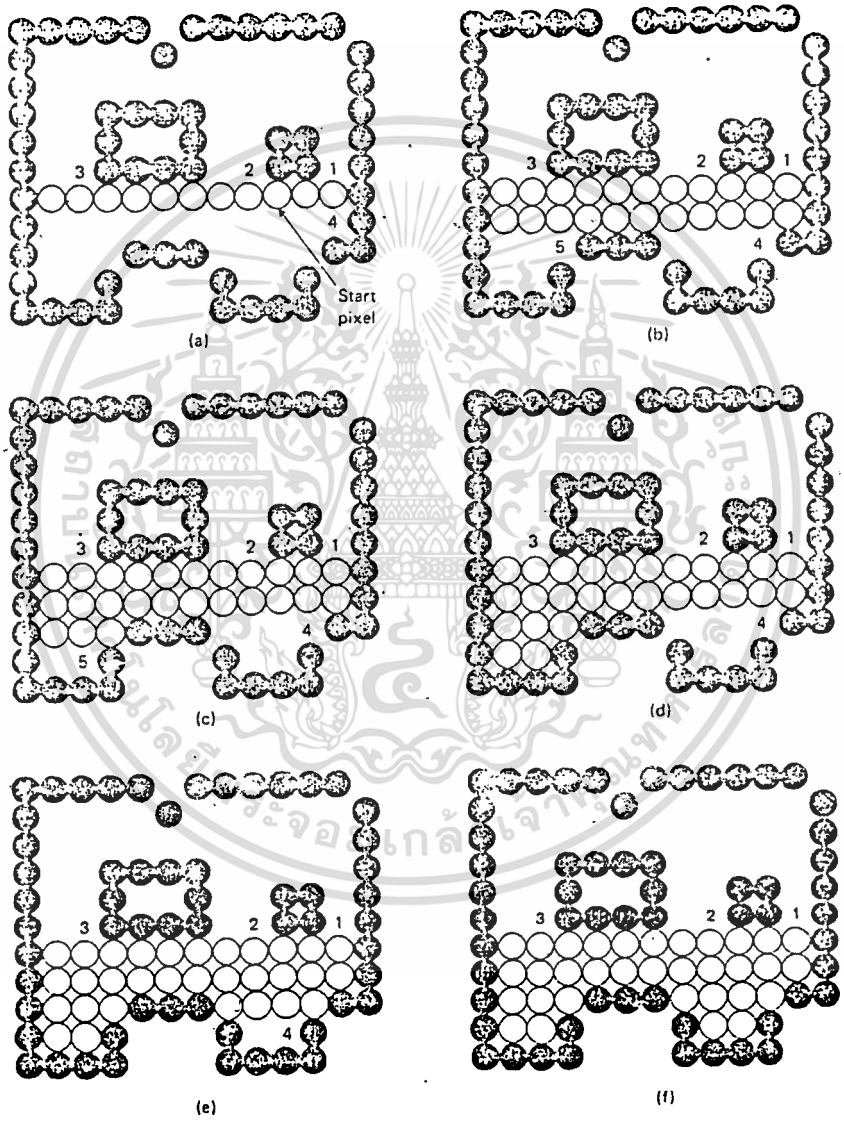
โดย d คือระยะทางระหว่างจุด 2 จุด

3.7 การระบายสีบนพื้นที่ที่กำหนด (seedfill algorithm)

ขบวนการระบายสีบนพื้นที่กำหนดนั้น มีหลายครั้งที่เรามีความต้องการที่จะกำหนดพื้นที่จำกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พื้นที่หนึ่งให้มีคุณลักษณะทางสี ที่แตกต่างจากอีกพื้นที่หนึ่ง สำหรับการระบายสีในหัวข้อนี้ เราเริ่มจากการกำหนดจุดทำการระบายสีจุดในแถบพื้นที่หลายเหลี่ยม ดังรูป 3.17



รูปที่ 3.17 วิธีการระบายสี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูป 3.17 เป็นขบวนการแสดงการทำงานของแอลกอริทึมการระบายสี โดยกำหนดจุดเริ่มต้นมาให้ (start pixel) ดังรูป 3.17(a)

สำหรับคำอธิบายหากผู้สนใจสามารถหาอ่านได้จากหนังสือ [2]

```
scan line seed fill algorithm
Seed(x, y) is the seed pixel
Pop is a function for removing a pixel from the stack
Push is a function for placing a pixel on the stack
    initialize stack
    Push Seed(x, y)
    while (stack not empty)
        get the seed pixel and set it to new value
        Pop Pixel(x, y)
        Pixel(x, y) = Fill value
        fill the span to the right of the seed pixel
        Savex = x
        x = x + 1
        if Pixel(x, y) <> Boundary value then
            Pixel(x, y) = Fill value
        else
            save the extreme right pixel
            Xright = x - 1
            x = Savex
        end if
        fill the span to the left of the seed pixel
        x = x - 1
        if Pixel(x, y) <> Boundary value then
            Pixel(x, y) = Fill value
        else
            save the extreme left pixel
            Xleft = x + 1
            x = Savex
        end if
        check that the scan line above is neither a polygon boundary nor has
        been previously completely filled.
        Savey = y
        x = Xleft
        y = y + 1
        if (Pixel(x, y) = Boundary value or
          Pixel(x, y) = Fill value) then
            x = x + 1
        else
            if x < Xright then
                seed the scan line above
                if (Pixel(x, y) <> Boundary value and
```

แอลกอริทึมการระบายสี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        x < Xright) then
          x = x + 1
        else
          push the extreme right pixel onto the stack
          Push Pixel(x - 1, y)
          continue checking in case the span is interrupted
          x = x + 1
          while (x < Xright)
            if (Pixel(x, y) = Boundary value or
              Pixel(x, y) = Fill value) then
                x = x + 1
            else
                x = x + 1
            end if
          end while
        end if
      else
        x = Savex
        y = Savey
      end if
    end if

    Check that the scan line below is not a polygon boundary nor has
    been previously completely filled.
      this algorithm is exactly the same as that for checking the scan
      line above except that y = y - 1 is subsituted for y = y + 1
  end while
  finish
```

Here the function **Pop** gets the x, y coordinates of a pixel from the stack and the function **Push** places them onto the stack.

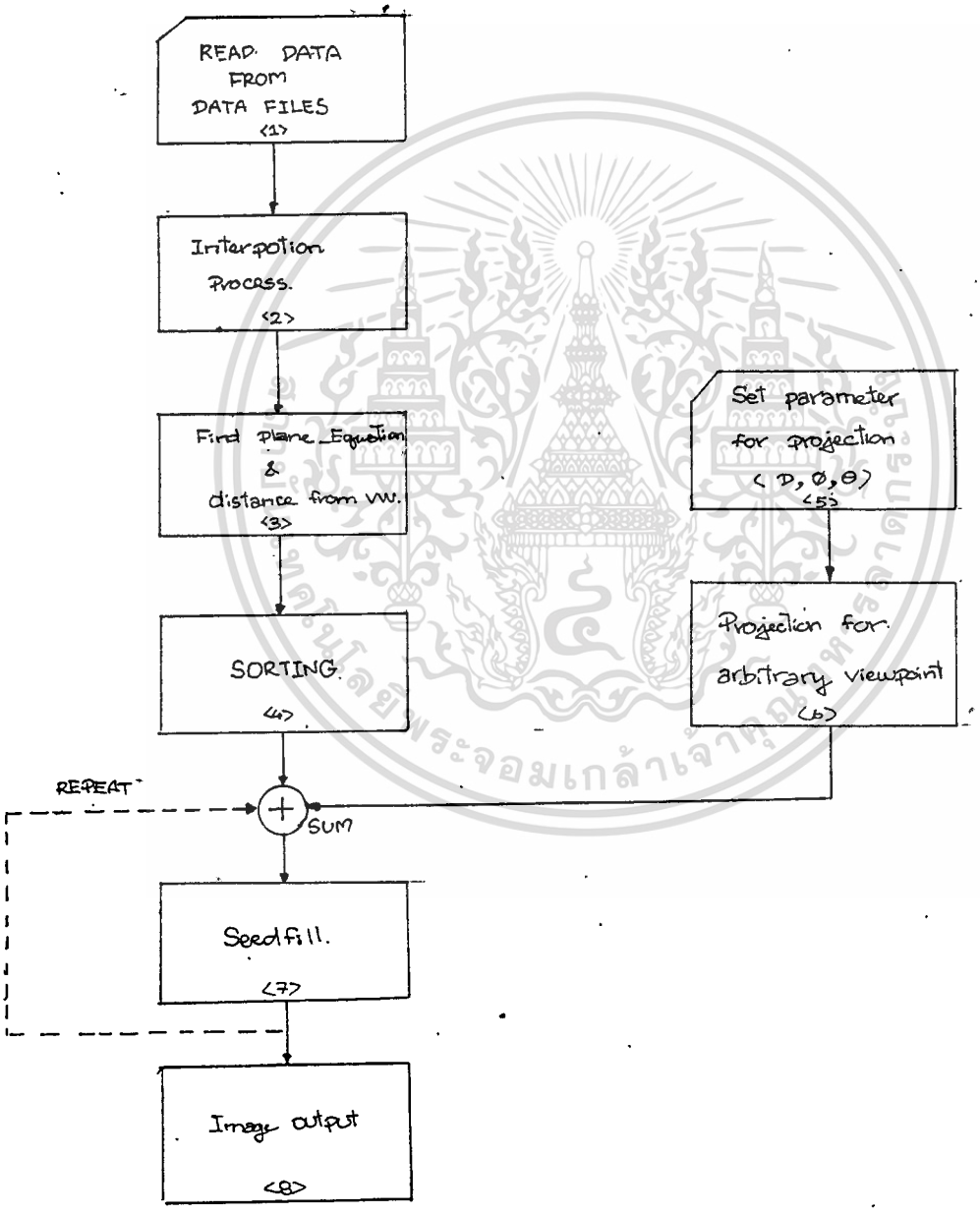
แอลกอริทึมการระบายสี (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

แนวความคิดในการสร้างภาพ 3 มิติ

บทนี้เราจะได้กล่าวถึงขั้นตอนการสร้างภาพ 3 มิติจากแผนที่แสดงเส้นชั้นความสูง พิจารณาจากบล็อกไดอะแกรม (block diagram) ข้างล่างแสดงขั้นตอนการทำงาน



รูปที่ 4.0 แสดงขั้นตอนการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการทำงาน เริ่มจาก

ขั้นแรก ทำการอ่านข้อมูลของบริเวณพื้นที่ที่ต้องการแสดงภาพ 2 มิติ ในลักษณะของค่าจุด (x,y,z) โดยค่า z คือ ค่าความสูงของเส้นชั้นความสูง x และ y คือ ระนาบพื้นราบ โดยกำหนดระยะห่างเป็นตารางกริด ดังรูป 4.1 โดยทำการอ่านข้อมูลจากไฟล์ข้อมูล (data file)

-ข้อสังเกตุ จุดกริดคือ จุดตัดของตาราง 4 เหลี่ยม คือ จุดที่เราต้องการหาค่าความสูง

จุดดอกจัน คือ จุดตัดระหว่างเส้นกริด กับเส้นชั้นความสูง เป็นค่าข้อมูลที่ป้อนเข้ามา

ขั้นตอนที่ 2 เราทำการนำข้อมูลที่ได้จากไฟล์ข้อมูล มาทำผ่านขบวนการประมาณโดยเราเลือกใช้วิธีคิวบิสสไปร์อินเทอโปเลชั่น เนื่องจากข้อมูลที่ป้อนเข้ามาหลายครั้งมีค่ามาก เพื่อป้องการเกิดพฤติกรรมการออสสิเรต จึงเลือกใช้วิธีนี้แทนการใช้วิธีลากรานซ์ อินเทอโปเลชั่น จากรูป 4.1 เราทำการประมาณเป็นเส้นๆไป โดยประมาณกับเส้นตารางที่ขนานกับแกน y

ขั้นตอนที่ 3 หลังจากที่เราประมาณหาความสูงที่จุดกริดได้ครบทุกจุดแล้ว ขั้นตอนต่อมา เรานำตารางกริดย่อย (สี่เหลี่ยมพื้นผ้าย่อยแต่ละอัน) มาหาสมการระนาบ (plane-equation) โดยใช้สมการในบทที่ 3 ที่ได้กล่าวไว้แล้ว หลังจากนั้นก็หาระยะระหว่างจุดมอง (viewpoint) กับจุดที่อยู่กึ่งกลางของสมการระนาบ ดังรูป 4.2

ค่า d แสดงถึงระยะทางระหว่างจุดมองกับจุดกลางของระนาบ

ขั้นตอนที่ 4 เป็นขบวนเรียงลำดับระยะ (d) ระหว่างจุดกลางของระนาบ ทุกๆระนาบ กับจุดมอง ณ ตำแหน่ง ตำแหน่งที่เรากำหนดขึ้น (รายละเอียดอ่านในบทที่ 2) โดยการเรียงลำดับนั้นเราจะทำการเรียงลำดับจากค่ามากไปหาค่าน้อย จากรูป 4.2 ได้ $d_3>d_1>d_2$

ขั้นตอน 5 และ 6 คือ ขั้นตอนที่เราทำการ กำหนดค่าตัวแปรในการโปรเจค และการโปรเจคจากภาพ 3 มิติ ในระบบคู่ลำดับโลก (world coordinate (x,y,z)) ซึ่งเป็นตำแหน่งของวัตถุจริง มาเป็นคู่ลำดับสายตา (eye coordinate system (x_e,y_e,z_e)) และเป็นคู่ลำดับขอจอภาพ (screen coordinate (x_s,y_s)) ซึ่งเป็นลักษณะของ 2 มิติ ดังสมการ 4.1

(x,y,z) ----------> (x_e,y_e,z_e)----------> (x_s,y_s) (4.1)

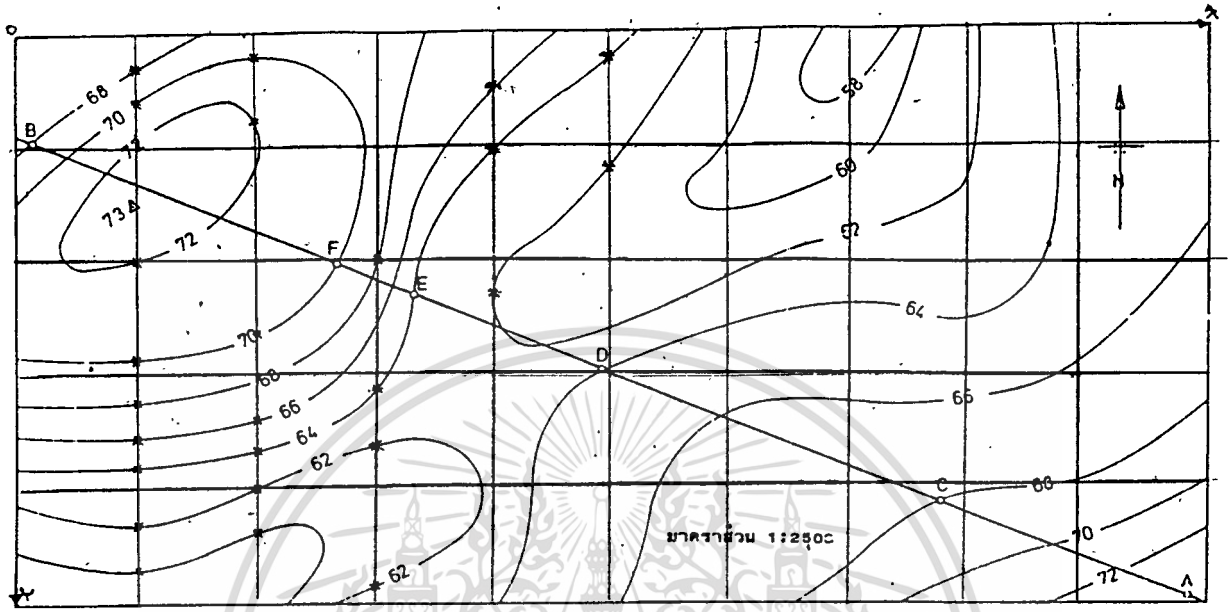
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนสุดท้าย เป็นกระบวนการระบายภาพบนจอภาพ (screen coordinate) โดยจะทำการระบายระนาบย่อยๆ แต่ละระนาบจากระนาบที่อยู่ระยะทางไกลสุด มาใกล้สุด กับ จุดมอง (d_{max} ----> d_{min})

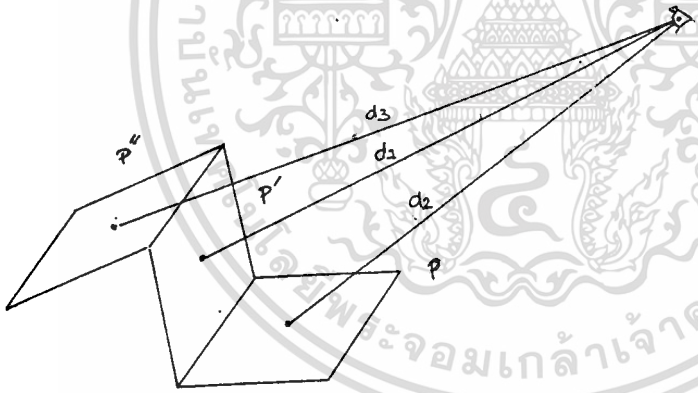
ในขั้นตอนที่ 4,5 และขั้นตอนสุดท้ายเป็นขบวนการปิดบังเส้น (hidden line) เพื่อกำจัดเส้นที่อยู่ด้านหลังจุดมองออกไป หรือเพื่อกำจัดเส้นที่เรามองไม่เห็นออกไป เพื่อให้ภาพที่ได้เหมือนจริง พิจารณา

จากรูป 4.3 สมมติให้ระนาบ P´ อยู่ไกลกว่าระนาบ P และอยู่ด้านหลังระนาบ P ขั้นแรกแอลกอริทึมการระบายจะทำการระบาย ระนาบ P´ ก่อน แล้วจึงจะระบายระนาบ P เราจะเห็นได้ว่าเส้นของระนาบ P´ ที่มองไม่เห็นจะถูกกลบลบไป ดังรูป 4.3

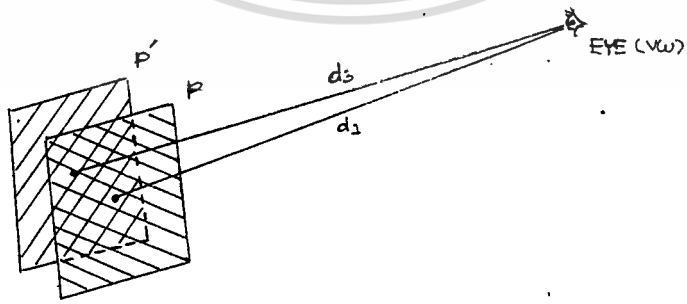
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า



รูปที่ 4.1



รูปที่ 4.2 การหาระยะทาง



รูปที่ 4.3 แสดง ระนาบที่ถูกบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

บทที่ 5

การทดสอบ

ผลการทดสอบวิธีการประมาณ

5.1 แบบลากรานส์อินเทอโปเรชั่น

5.1.1 เมื่อมีข้อมูลจำนวนน้อย เราพบว่า ค่าที่ประมาณได้มีค่าใกล้เคียงกับค่าความเป็นจริง มากกว่า แบบคิวบิคสไปส์ (ข้อมูล 12 ค่า)

```
* LAGRANGE INTERPOLATION *

* input data *

    I          X          Y
  1    0.500000D+01 0.445000D+02
  2    0.100000D+02 0.435000D+02
  3    0.150000D+02 0.426000D+02
  4    0.200000D+02 0.418000D+02
  5    0.250000D+02 0.413000D+02
  6    0.300000D+02 0.407000D+02
  7    0.350000D+02 0.402000D+02
  8    0.500000D+02 0.356000D+02
  9    0.550000D+02 0.341000D+02
 10    0.600000D+02 0.323000D+02
 11    0.650000D+02 0.305000D+02
 12    0.700000D+02 0.330000D+02

* solution *

  P( 45 )=  0.376585D+02
```

จากค่าที่ได้สำรวจ (ค่าจริง) พบว่า ที่ P(45) = 0.375D+02 ซึ่งเมื่อเปรียบเทียบกับแบบคิวบิคสไปส์ เนื่องจาก ประมาณด้วยกำลังที่สูงกว่า (order N-1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.2 เมื่อมีข้อมูลมากเกินไป (ข้อมูล 23 ตัว) พบว่าเกิด พฤติกรรมการออสสิเรท ซึ่งเป็นข้อเสีย จากค่าที่คำนวณได้มีค่ามากจนเกินความเป็นจริง

สังเกต ข้อมูลทุกตัวมีค่าน้อยกว่า 100 มาก แต่ค่าที่คำนวณได้มีค่าเกิน 100

```
* LAGRANGE INTERPOLATION *

* input data *

    I          X              Y
  1    0.500000D+01 0.445000D+02
  2    0.100000D+02 0.435000D+02
  3    0.150000D+02 0.426000D+02
  4    0.200000D+02 0.418000D+02
  5    0.250000D+02 0.413000D+02
  6    0.300000D+02 0.407000D+02
  7    0.350000D+02 0.402000D+02
  8    0.500000D+02 0.356000D+02
  9    0.550000D+02 0.341000D+02
 10    0.600000D+02 0.323000D+02
 11    0.650000D+02 0.305000D+02
 12    0.700000D+02 0.330000D+02
 13    0.750000D+02 0.345000D+02
 14    0.800000D+02 0.367000D+02
 15    0.850000D+02 0.341000D+02
 16    0.900000D+02 0.380000D+02
 17    0.950000D+02 0.374000D+02
 18    0.100000D+03 0.411000D+02
 19    0.105000D+03 0.432000D+02
 20    0.110000D+03 0.389000D+02
 21    0.115000D+03 0.399000D+02
 22    0.120000D+03 0.421000D+02
 23    0.125000D+03 0.351000D+02

* solution *

  P( 45 )=  0.103896D+03
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ข้อดีประการหนึ่งของการแปลงแบบลากรานส์ คือ สามารถประมาณจุดที่อยู่นอกขอบเขตที่ต้องการได้ (แต่ค่าที่ได้มีความถูกต้องน้อยลง) ซึ่งต่างจากแบบคิวบิควิ่งทำไม่ได้ ดูได้จากหัวข้อ ที่ 5.2.3

```
* LAGRANGE INTERPOLATION *

* input data *

    I          X              Y
  1    0.500000D+01 0.445000D+02
  2    0.100000D+02 0.435000D+02
  3    0.150000D+02 0.426000D+02
  4    0.200000D+02 0.418000D+02
  5    0.250000D+02 0.413000D+02
  6    0.300000D+02 0.407000D+02
  7    0.350000D+02 0.402000D+02
  8    0.500000D+02 0.356000D+02
  9    0.550000D+02 0.341000D+02
 10    0.600000D+02 0.332000D+02
 11    0.650000D+02 0.305000D+02
 12    0.700000D+02 0.330000D+02

* solution *

  P( 3 )=  0.499946D+02
```

5.2 แบบวิธีคิวบิคสไปร์อินเทอโปเรท

5.2.1 เมื่อมีข้อมูลน้อย (12 ตัว)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
* cubic spilne interpolation *

 *  input  data *

    I          X                Y
   1     0.5000000D+01   0.4450000D+02
   2     0.1000000D+02   0.4350000D+02
   3     0.1500000D+02   0.4260000D+02
   4     0.2000000D+02   0.4180000D+02
   5     0.2500000D+02   0.4130000D+02
   6     0.3000000D+02   0.4070000D+02
   7     0.3500000D+02   0.4020000D+02
   8     0.5000000D+02   0.3560000D+02
   9     0.5500000D+02   0.3410000D+02
  10     0.6000000D+02   0.3320000D+02
  11     0.6500000D+02   0.3050000D+02
  12     0.7000000D+02   0.3300000D+02

* solution  *

   P( 45 )=  0.3660000D+02
```

5.2.2 เมื่อมีข้อมูลมาก (23 ตัว) พบว่าค่าที่ประมาณได้ไม่เปลี่ยน เพราะว่าใช้ออร์เดอร์เท่ากัน

```
    I          X                Y
   1     0.5000000D+01   0.4450000D+02
   2     0.1000000D+02   0.4350000D+02
   3     0.1500000D+02   0.4260000D+02
   4     0.2000000D+02   0.4180000D+02
   5     0.2500000D+02   0.4130000D+02
   6     0.3000000D+02   0.4070000D+02
   7     0.3500000D+02   0.4020000D+02
   8     0.5000000D+02   0.3560000D+02
   9     0.5500000D+02   0.3410000D+02
  10     0.6000000D+02   0.3320000D+02
  11     0.6500000D+02   0.3050000D+02
  12     0.7000000D+02   0.3300000D+02
  13     0.7500000D+02   0.3450000D+02
  14     0.8000000D+02   0.3670000D+02
  15     0.8500000D+02   0.3410000D+02
  16     0.9000000D+02   0.3800000D+02
  17     0.9500000D+02   0.3740000D+02
  18     0.1000000D+03   0.4110000D+02
  19     0.1050000D+03   0.4320000D+02
  20     0.1100000D+03   0.3890000D+02
  21     0.1150000D+03   0.3990000D+02
  22     0.1200000D+03   0.4210000D+02
  23     0.1250000D+03   0.3550000D+02

   * solution  *

      P( 45 )=  0.3660000D+02
```

5.2.3 เมื่อประมาณขอบเขต ไม่สามารถทำได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

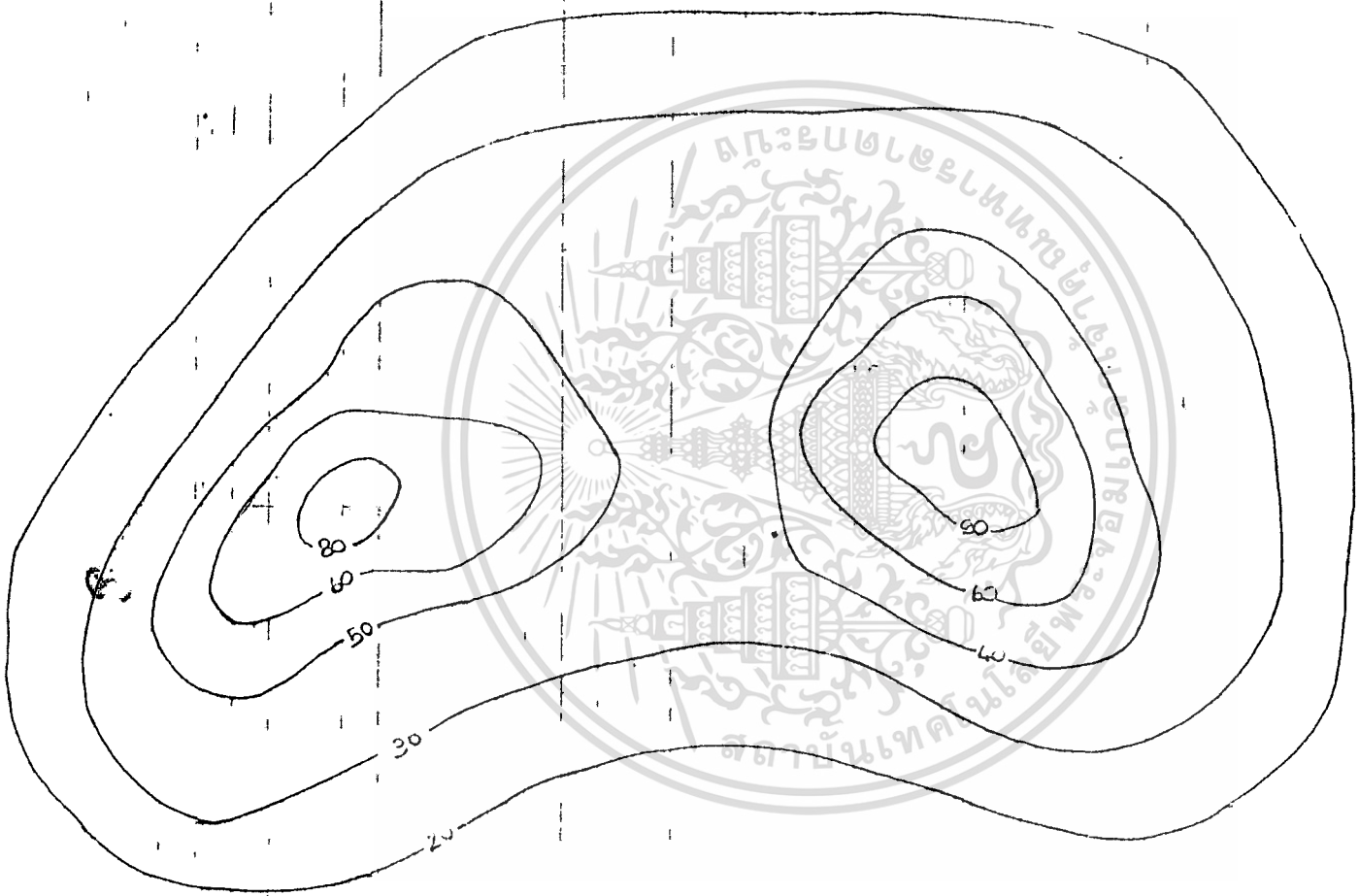
* cubic spilne interpolation *

* input data *

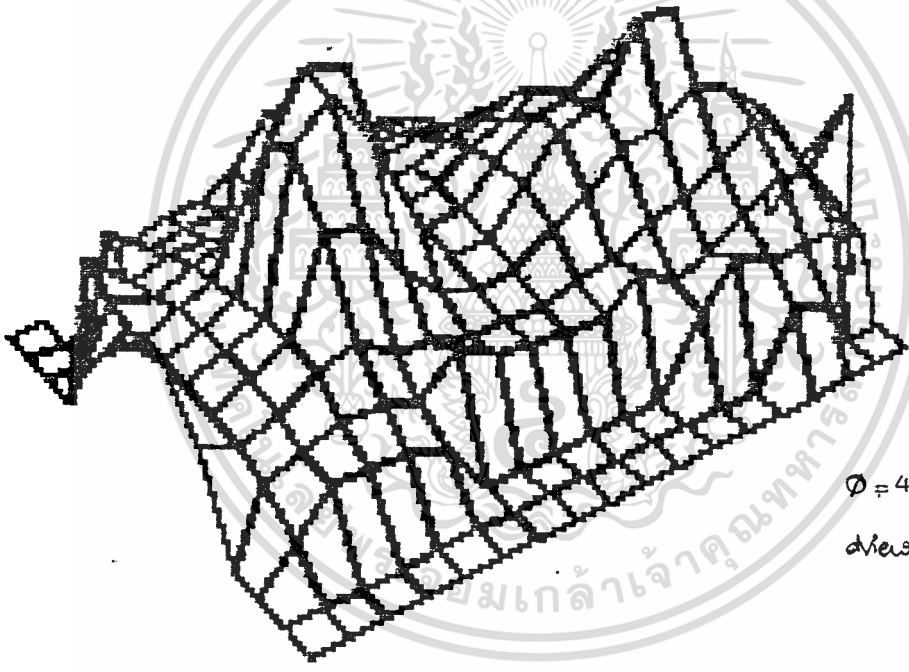
I	X	Y
1	0.5000000D+01	0.4450000D+02
2	0.1000000D+02	0.4350000D+02
3	0.1500000D+02	0.4260000D+02
4	0.2000000D+02	0.4180000D+02
5	0.2500000D+02	0.4130000D+02
6	0.3000000D+02	0.4070000D+02
7	0.3500000D+02	0.4020000D+02
8	0.5000000D+02	0.3560000D+02
9	0.5500000D+02	0.3410000D+02
10	0.6000000D+02	0.3320000D+02
11	0.6500000D+02	0.3050000D+02
12	0.7000000D+02	0.3300000D+02

* out of range *

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

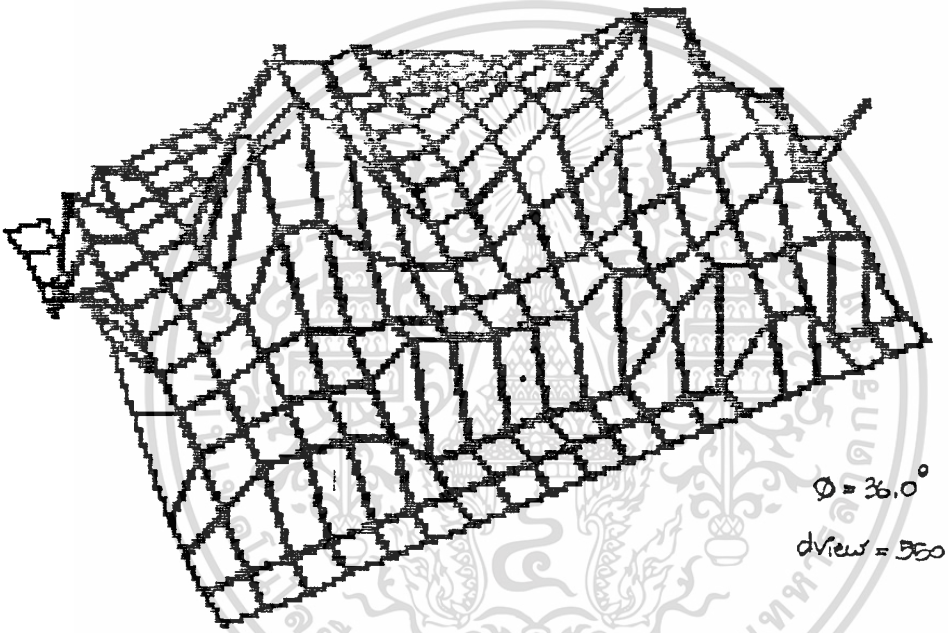


INPUT DATA (CONTOUR LEVEL)



OUTPUT IN 3-DIMENSION

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



OUTPUT IN 3-DIMENSION

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

บทที่ 6

สรุปผล

แอลกอริทึม ที่ใช้อยู่ในปริญญานิพนธ์ฉบับนี้ เริ่มต้นจาก วิธีการประมาณ ค่าความสูงของจุดกริดนั้น ผู้เขียนได้คิดหาวิธีมาจากการพิจารณา การสร้างเส้นชั้นความสูงโดยวิธีย้อม (Indirect method) ของวิชาการทางด้านวิศวกรรมการสำรวจ เป็นแนวทางโดยอาศัยบทกลับ คือ นำเส้นที่ได้จากการประมาณมาคำนวณหาจุดแทน สำหรับ แอลกอริทึมที่ใช้ในการประมาณจุดมีความสำคัญมาก วิธีหนึ่งที่ต้องการประมาณที่ถูกต้องให้มากยิ่งขึ้นคือ การใช้แอลกอริทึมการประมาณหลายแอลกอริทึม โดยแต่ละแอลกอริทึมจะมีคุณสมบัติในการประมาณข้อมูลของภูมิประเทศต่างๆกัน เช่น 3มิติประเทศที่เรียบมาก มีการสูงเพิ่มขึ้นเชิงเส้น ก็ควรใช้แบบการประมาณแบบเส้นตรง เป็นต้น สำหรับผลงานนี้ได้เสนอวิธีการประมาณเพียงวิธีเดียว คือ คิวบิสสไปท์อินเทอโปเลท์ โดยมีข้อดี และข้อเสีย ดังนี้

ข้อดี

-เป็นการประมาณเชิงโค้งแบบโค้งเรียบ โดยเหมาะกับภูมิประเทศที่เป็นเนินเขาที่ไม่มีการเปลี่ยนแปลงความสูงแบบทันทีทันใดมากนัก

-การประมาณใช้เวลาในการคำนวณน้อยและใช้ห้องความจำในการเก็บน้อย

-สามารถใช้กับข้อมูลที่มีจำนวนมากเท่าใดก็ได้

-ความถูกต้องของข้อมูลอยู่ในเกณฑ์ค่อนข้างดี (ขึ้นกับรูปแบบของข้อมูล)

ข้อเสีย

-ไม่สามารถประมาณจุดที่อยู่นอกส่วนได้

-ข้อมูลจะต้องมีมากกว่าจำนวนจุดที่ต้องการประมาณอย่างน้อย 1 จุด

-ไม่เหมาะกับการใช้ประมาณภูมิประเทศที่ราบเรียบหรือมีการเปลี่ยนระดับความสูงเป็นแบบเชิงเส้น

-เป็นการประมาณ โดยอาศัยตัวแปรเพียง 2 ตัวแปรเท่านั้น ดังนั้น ในการประมาณ เราจึงให้ค่าของ x คงที่ตลอด และค่าของ z แปรตามค่าของ y ทำให้ได้ค่าความถูกต้องลดลง เหตุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลที่ไม่ใช้การประมาณแบบ 3 ตัวแปร เนื่องมาจาก ต้องใช้คณิตศาสตร์ที่ยุ่งยากมาก

สำหรับแอบกอริทึมทางการสร้างภาพ 3 มิติ นั้น ยังมีปัญหาที่สำคัญปัญหาหนึ่งก็คือ ปัญหาการคำนวณทางสมการทางคณิตศาสตร์ที่มีเป็นจำนวนมาก เนื่องจากต้องคำนวณทุกๆจุดที่ต้องการแสดง ทำให้ดูค่อนข้างจะไม่เหมาะสมกับเครื่องไมโครคอมพิวเตอร์ เนื่องจากสาเหตุสำคัญคือ

-การสร้างภาพทางคอมพิวเตอร์กราฟฟิคบางครั้งต้องการความเร็วสูง ในการคำนวณทางคณิตศาสตร์ จึงดูจะไม่เหมาะสมกับเครื่องคอมพิวเตอร์ขนาดเล็ก (micro) สำหรับ IBM PC ควรที่จะมีการนำเอา 8087 Mathscoprocessor มาช่วยในการคำนวณ

-ความละเอียดของจอและจำนวนสีของวง เป็นส่วนที่สำคัญมาก เนื่องจาก จอที่มีคุณสมบัติความละเอียดต่ำ และจำนวนสีน้อยจะทำให้ได้ภาพที่ออกมาค่อนข้างหยาบ และการอ้างตำแหน่งของจุด (pixel) บนจอทำได้ในช่วงที่ค่อนข้างจำกัด

สำหรับแอกอริทึมการปิดบัง (hidden) ในผลงานนี้ ผู้เขียนได้คิดหาวิธีเองทำให้มีปัญหาคือไม่สามารถทำการปิดบังได้ทุกส่วนดังได้แสดงไว้ในบทที่ 5 ผู้เขียนคิดว่าควรที่จะได้มีการเพิ่มส่วนการตรวจสอบให้เพิ่มมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

กิตติกรรมประกาศ

ไม่ว่าจะเดินทางเป็นระยะทางไกลหรือใกล้ เราต้องมีก้าวแรกเสมอ ผู้ชี้ทางเดินก้าวแรกทั้ง 2 ท่าน ก็คือ อาจารย์ ดร. วัลลภ สุระกำพลธร ผู้ที่ได้ให้ความช่วยเหลือและคำแนะนำ ในการสร้างผลงานชิ้นนี้ และ ดร. ฟูศักดิ์ ชีวะสุวิทย์ ผู้ที่ได้ให้แนวความคิดเริ่มต้นและให้คำปรึกษามาตลอด

นอกจากนี้ข้าพเจ้าขอขอบคุณ บริษัท บางกอกดาต้า เซนเตอร์ จำกัด เป็นอย่างมากที่ได้เอื้อเฟื้ออุปกรณ์ และให้คำแนะนำแก่ข้าพเจ้า พี่ๆ บริษัท R & D จำกัด ที่ได้ช่วยเหลือให้วิทยานิพนธ์ฉบับนี้สำเร็จออกมาได้ รวมทั้งเพื่อนสนิท (แหลม) ที่ได้ให้คำแนะนำเกี่ยวกับโปรแกรมภาษา C เป็นอย่างมาก

พัฒน์พงษ์ มาศวงษ์ปกรณ์

12 มีนาคม 2531

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] CHAN S. PARK,"INTERACTIVE MICROCOMPUTER GRAPHICS", Addison-Wesley Publishing Company. Page 1-201.

[2] David F. Roger,"PROCEDURAL ELEMENTARY FOR COMPUTER GRAPHICS", McGraw-Hill Book Company. Page 29-51, 19-92

[3] James D. FOLEY & ANDRIES VAN DAM,"Foundamentals of interactive of interactive computer Graphics", Addison-wesley Publishing . Page 505-514, 556-569

[4] MARC BERGER,"Computer Graphic S. with pascal", The Benjamin/Cummings Publishing Company Inc , Page 251-304, 1-12, 220-231

[5] ดร. สุชาย ธนังเสถียร,วิชัย จิวังกูร," โครงสร้างข้อมูล เพื่อการออกแบบโปรแกรมคอมพิวเตอร์", สำนักพิมพ์ ซีเอ็ดยูเคชั่น จำกัด, หน้า 127-143

[6] อ.วัชรินทร์ วิทยกุล, "การสำรวจเพื่อการก่อสร้าง (Surveying for construction)", สำนักพิมพ์ พิสิกส์ เซนเตอร์ จำกัด, หน้า 126-146

[7] ดร. จเร สุธวัฒน์ปัญญา, "Numerical Computation" , คณะวิศวกรรมศาสตร์ ลาดกระบัง, หน้า 117-136

[8] ผศ. ทศพร คล้ายอุดม, "แคลคูลัส และเรขาคณิตวิเคราะห์ 3", คณะวิทยาศาสตร์ มหาวิทยาลัยรามคำแหง, หน้า 1-39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า