ปีการศึกษา 2531

คอมพิวเตอร์กราฟฟิก

(Computer Graphics)

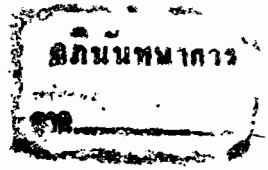
โดย

นายธนวิน ธัญญานุรักษ์

นายวินิจ พนาเมธา

อาจารย์ที่ปรึกษา

ดร. บุญวัฒน์ อัตชู



023191 -9 ส.ค. 2532

คอมพิวเตอร์กราฟฟิก

(Computer Graphics)

โดย

นาย ธนวิน ธัญญานุรักษ์ 28.1093

นาย วินิจ พนาเมฆา 28.1223

อาจารย์ที่ปรึกษา

ดร. บุญวัฒน์ อัตชู

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ประจำปีการศึกษา 2531

023191

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ปริญญานิพนธ์ปีการศึกษา 2531

ภาควิชา วิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง คอมพิวเตอร์กราฟฟิก

ผู้จัดทำ

1. นายธนวิน ชัญญานุรักษ์
2. นายวินิจ พนาเมธา

.....อาจารย์ที่ปรึกษา

(ดร. บุญวัฒน์ อัตชู)

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอมพิวเตอร์กราฟิค

ธนวิน บัญญานุรักษ์
วินิจ พนาเมธา
ดร. บุญวัฒน์ อัตชู อาจารย์ที่ปรึกษา
ปีการศึกษา 2531

บทคัดย่อ

คอมพิวเตอร์กราฟิค(Computer Graphics)เป็นสาขาหนึ่งในหลายๆสาขาวิชาทางด้านคอมพิวเตอร์ที่ได้รับการศึกษาค้นคว้าอย่างกว้างขวาง และมีการประยุกต์ใช้งานทางด้านต่างๆหลายแขนง เช่น สถาปัตยกรรมและการออกแบบ, วิศวกรรม, ธุรกิจ-และการโฆษณา, การสื่อสารและการบันเทิง ฯลฯ แขนงหนึ่งที่เป็นที่สนใจมากในขณะนี้คือคอมพิวเตอร์ช่วยในการออกแบบ(Computer Aided Design หรือ CAD) อันเป็นการนำเอาความสามารถทางด้านกราฟิคมาใช้ในการออกแบบวาดรูปต่างๆและได้มีการโปรแกรมสำเร็จรูปหลายโปรแกรม เช่น AUTOCAD, LUCID-3D, AUTOSHADE เป็นต้น ที่มีความสามารถในการวาดรูปทรงต่างๆทั้งแบบ 2 มิติและ 3 มิติ, การลบเส้นและพื้นผิวที่มองไม่เห็นใน 3 มิติ(Hidden line and Surface Removals) และการกำหนดแสง(Shading) อยู่ด้วย

ในปริญญานิพนธ์ฉบับนี้ จึงได้เสนอทฤษฎีเบื้องต้นในการแสดงภาพกราฟิคบนไมโครคอมพิวเตอร์ เพื่อเป็นแนวทางในการสร้างหรือนำเอาทฤษฎีเหล่านี้ไปประยุกต์ใช้ โดยส่วนใหญ่จะเป็นทฤษฎีทางคณิตศาสตร์ โดยเน้นด้านเรขาคณิตวิเคราะห์ (Analytic Geometry) นอกจากนี้ยังได้สร้างแบบจำลองขึ้นมาเพื่อเป็นแนวทางในการศึกษารวมทั้งนำทฤษฎีไปประยุกต์ใช้งานได้ง่ายขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Computer Graphics

Mr. Thanawin Thanyanurak

Mr. Winij Panamaeta

Dr. Boonwat Attachoo Advisor

1988

Abstarct

Computer Graphics is one of many fields in Computer which is widely studied and applied to many subjects such as Architecture and design, Engineering, Business and advertisement, Communication and entertainment etc. The interesting one is Computer Aided Design (CAD) which uses the capabilities in Graphics of microcomputer to draw pictures or models both in 2 dimension and 3 dimension, hidden-line and surface removals, shading. And there are also many packages such as AUTOCAD, LUCID-3D, AUTOSHADE etc. that can draw pictures and shapes in those capabilties.

So this thesis will present the basic and simple principles for drawing and creating pictures on microcomputer's screen in order to be guildline for application. Most of this principles are analytic Geometry. There are some models created for interested person to study for application.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

บทที่ 1 บทนำ 1

1.1 กล่าวนำ 1

1.2 วัตถุประสงค์ของโครงงาน 2

1.3 ขอบเขตของงานวิจัย 2

บทที่ 2 ทฤษฎีและหลักการ 5

2.1 การสร้างภาพในระบบ 2 มิติ 5

2.2 การสร้างภาพในระบบ 3 มิติ 16

2.3 การกำจัดเส้นและพื้นผิวที่ถูกบดบังในภาพ 3 มิติ 31

2.4 การกำหนดแสงและการให้แสงเงา 42

บทที่ 3 เนื้อหาเกี่ยวกับการทดลองและประยุกต์ใช้ทฤษฎีต่างๆ 46

3.1 ขั้นตอนและวิธีการในการพัฒนาโปรแกรม 46

3.2 ผลการทดลองที่ได้จากการพัฒนาโปรแกรม 49

บทที่ 4 สรุปและวิจารณ์ 63

กิตติกรรมประกาศ 64

บรรณานุกรม 65

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

รูปภาพ หน้า

รูป 2.1 แสดงการขยายตำแหน่ง 6

รูป 2.2 แสดงการสะท้อนกับแกน Y และแกน X ตามลำดับ 6

รูป 2.3 แสดงการหมุนจุด P ไปเป็นมุม θ 8

รูป 2.4 แสดงการเคลื่อนย้ายจุด 8

รูป 2.5 แสดงการกำหนดขอบเขตของภาพ (Clipping) 12

รูป 2.6 แสดงการกำหนดส่วนของเส้นที่มองเห็น 12

รูป 2.7 แสดงการขยายภาพ (Zooming) 15

รูป 2.8 แสดงความสัมพันธ์ระหว่างระบบ Spherical กับ ระบบ Rectangular ..17

รูป 2.9 แสดงการย้ายระบบโคออร์ดิเนต 20

รูป 2.10 แสดงการหมุนระบบโคออร์ดิเนต 20

รูป 2.11 แสดงการสัมพันธ์ระหว่างระบบ World Coordinate , Eye Coordinate และ Screen Coordinate 22

รูป 2.12 แสดงความสัมพันธ์ระหว่าง Central Projection และ Arbitrary Projection 24

รูป 2.13 แสดงขั้นตอนการย้ายแกนเพื่อกำหนดจุดในระบบ eye coordinate 25

รูป 2.14 แสดงให้เห็นว่า parameter(s) มีส่วนในการปรับแต่งจอภาพ 28

รูป 2.15 แสดงวิธีหาการมองเห็นได้ของพื้นผิว 32

รูป 2.16 แสดงภาพที่ได้จากการกำจัดเส้นและพื้นผิวที่ถูกบังออกแล้ว 34

รูป 2.17 แสดงขั้นตอนการกำจัดเส้นและพื้นผิวที่ถูกบังออก 35

รูป 2.18 แสดงการทดสอบโดย Minimax Test 36

รูป 2.19 แสดงกรณีหนึ่งของ Minimax Test ที่ fail 36

รูป 2.20 แสดงการทดสอบโดย Intersection test 40

รูป 2.21 แสดงการกำจัดเส้นที่ถูกบังออกโดยวิธี Masking Technique 40

เอกสารนี้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปภาพ หน้า

รูป 2.22 แสดงการกำจัดเส้นที่ถูกบังโดยวิธี Image Space Algorithm 40

รูป 2.23 แสดงปริมาณแสงที่สะท้อนออกมากับโคซายน์ของมุม θ 42

รูป 2.24 แสดงการสะท้อนแบบสเปคคูล่าร์ 44

รูป 3.1 แสดงภาพ 2 มิติที่ได้จากการฉายภาพ 46

รูป 3.2 แสดงภาพ 3 มิติที่ไม่กำจัดเส้นที่ถูกบังออก 47

รูป 3.3 แสดงภาพ 3 มิติที่กำจัดเส้นและพื้นผิวที่ถูกบังออก 47

รูป 3.4 แสดงภาพ 3 มิติ 2 รูปที่มีการกำจัดเส้นและพื้นผิวที่ถูกบังออก 48

รูป 3.5 แสดงแบบจำลองทรงกลมที่กำจัดและไม่กำจัดเส้นและพื้นผิวที่ถูกบังออกในทิศทาง
ต่างๆกัน 48

รูป 3.6 แสดงตำแหน่งและทิศทางของ Light source และ Viewpoint. 49

รูป 3.7 แสดงการย้ายจุด Origin ไปยัง endpoint ที่ 1 50

รูป 3.8 แสดงการหมุนระนาบ X_a Y_a ทิศทางทวนเข็มนาฬิกาเมื่อมองจากแกน Z_a ..52

รูป 3.9 แสดงการหมุนระนาบ X_b Z_b ทิศทางทวนเข็มนาฬิกาเมื่อมองจากแกน Y_b ..53

รูป 3.10 แสดงทิศทางของ Viewpoint และ Light source เมื่อระบบถูกเปลี่ยน.55

รูป 3.11 แสดงการหมุนระนาบ Y_c Z_c ทิศทางทวนเข็มนาฬิกาเมื่อมองจากแกน X_c .55

รูป 3.12 แสดงการหมุนระนาบ X_d Y_d ทิศทางทวนเข็มนาฬิกาเมื่อมองจากแกน Z_d .56

รูป 3.13 แสดงแบบจำลองที่มีการกำหนดแสง (shading model) 59

รูป 3.14 แสดงโฟล์วชาร์ตการทำงานของ shading 60

รูป 3.15 แสดงโฟล์วชาร์ตการทำงานของ mapping_to_image1 61

รูป 3.16 แสดงโฟล์วชาร์ตการทำงานของ mapping_to_image2 62

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 กล่าวนำ

คอมพิวเตอร์กราฟฟิก (Computer Graphics) เป็นสาขาหนึ่งในหลาย ๆ สาขาวิชา ซึ่งอยู่กึ่งกลางระหว่างศาสตร์ (Sciences) และศิลป์ (Arts) ซึ่งเกี่ยวข้องกับการคิดแบบวิเคราะห์ (Analytic Thinking) และการคิดแบบสร้างสรรค์ (Creative Thinking) เรามักจะนำคอมพิวเตอร์กราฟฟิกมาทำแบบจำลอง (model) ตราสัญลักษณ์ต่าง ๆ (Symbolic curves) นอกจากนี้ เรายังสามารถเขียนและระบายสีภาพต่าง ๆ, สร้างภาพเคลื่อนไหว (Animated Films) ของวัตถุต่าง ๆ ที่จินตนาการขึ้นมาให้มีลักษณะเหมือนจริงได้ ทำให้ภาพที่เราสร้างขึ้นมามีความน่าสนใจยิ่งขึ้น เหตุนี้เองทำให้คอมพิวเตอร์กราฟฟิกเข้าไปมีส่วนเกี่ยวข้องกับงานด้านต่าง ๆ เกือบทุกแขนง ได้แก่ งานด้านสถาปัตยกรรมและการออกแบบ, งานด้านวิศวกรรม, งานด้านธุรกิจและการโฆษณา, การสื่อสารและการบันเทิง, และการศึกษาทุกระดับ เป็นต้น

สิบกว่าปีที่ผ่านมา ผู้ที่ศึกษาค้นคว้าในสาขาวิชานี้มักจะเป็นกลุ่มที่ใช้เครื่องคอมพิวเตอร์ขนาดใหญ่ และอุปกรณ์แสดงผล (Display Devices) ราคาแพง เนื่องจากต้องใช้หน่วยความจำ (Memory) มาก การประมวลผลที่รวดเร็ว และการแสดงผลที่ละเอียดและสมบูรณ์ แต่ในปัจจุบัน เมื่อเทคโนโลยีทางด้านไมโครอิเลคโทรนิคส์สูงขึ้น ไมโครคอมพิวเตอร์ที่มีขนาดเล็กและราคาไม่แพง สามารถสนับสนุนงานประยุกต์ต่าง ๆ ของคอมพิวเตอร์กราฟฟิกระดับสูงได้ ความตื่นตัวในการศึกษาทดลองทางด้านนี้จึงเป็นไปอย่างกว้างขวาง อย่างไรก็ตาม การศึกษาด้านนี้ยังไปอย่างเชื่องช้ามาก เนื่องมาจากเนื้อหาที่เน้นหนักทางด้านเรขาคณิตวิเคราะห์ (Analog Geometry) โดยมีจุดสำคัญ คือ การแทนภาพ 3 มิติ (Three-dimensional Representation) ด้วยตัวกลาง 2 มิติ (Two-dimensional Medium) จึงสะท้อนให้เห็นว่าจุดเริ่มต้นของสาขาวิชานี้มาจากทางวิศวกรรม (Engineering) มากกว่าที่เป็นทางศิลป์ (Arts) ซึ่งแสดงให้เห็นว่าช่วงเวลาที่ผ่านมา สิ่งที่มีการเปลี่ยนแปลงไปก็มีแต่เพียงฮาร์ดแวร์เท่านั้นเอง

ฉะนั้นการศึกษาด้านคอมพิวเตอร์กราฟฟิกต้องมีความเข้าใจคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เทคโนโลยีในการแสดงผล (Display Technologies) และพื้นฐานทางเรขาคณิตวิเคราะห์ทั้ง 2 มิติและ 3 มิติ แล้วนำมาประกอบกับความรู้ด้านเมตริกซ์ (Matrix), โครงสร้างข้อมูล (Data Structure) และอัลกอริธึมต่างๆที่มีอยู่ ที่สุดแล้วเราก็สามารถศึกษาในสาขาวิชานี้ได้เข้าใจเป็นอย่างดี

1.2 วัตถุประสงค์ของโครงงาน

1. ศึกษาเทคนิคการสร้างแบบจำลองทางกราฟฟิกทั้ง 2 มิติและ 3 มิติ
2. ศึกษาเทคนิคการแสดงแบบจำลองที่มีหลายวัตถุซ้อนกัน (Hidden-line objects and surface removal)
3. ศึกษาทฤษฎีการกำหนดแสง (Shading)
4. รวบรวมเทคนิคต่างๆที่ศึกษานำมาสร้างเป็นแบบจำลองกราฟฟิกที่ซับซ้อน
5. พัฒนาโปรแกรมในการแสดงภาพกราฟฟิกโดยใช้เทคนิคดังที่กล่าวมาแล้ว

1.3 ขอบเขตของงานวิจัย

ขอบเขตของงานวิจัยนี้ มีเนื้อหาเกี่ยวกับวิธีการต่างๆในการแสดงภาพทางเรขาคณิตออกทางจอภาพซึ่งมีลักษณะของ 2 มิติและ 3 มิติ, การกำจัดเส้นและพื้นผิวที่ถูกบังออก, พร้อมด้วยการแสดงลักษณะของภาพที่มีลักษณะเหมือนจริงโดยการกำหนดแสงเงา โดยเนื้อหาของปริญญานิพนธ์นี้จะเน้นที่การใช้วิธีการต่างๆที่จะแสดงภาพกราฟฟิกออกจอภาพ พร้อมทั้งนำเอาวิธีการต่างๆไปประยุกต์ใช้เพื่อสร้างแบบจำลองที่มีลักษณะเหมือนจริงขึ้น

เนื้อหาปริญญานิพนธ์ฉบับนี้แบ่งออกเป็น 2 ภาคดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาค 1 : เนื้อหากล่าวถึงทฤษฎีต่างๆที่ใช้กันอยู่โดยทั่วไป ซึ่งมีเนื้อหาโดยย่อดังนี้

- การสร้างภาพในระบบ 2 มิติ ซึ่งกล่าวถึงการเปลี่ยนแปลงของจุด (point) การเปลี่ยนแปลงของเส้นซึ่งประกอบด้วยจุด 2 จุด และการเปลี่ยนแปลงของวัตถุ (object) ซึ่งประกอบด้วยเส้นหลาย ๆ เส้น การเปลี่ยนแปลงของรูปซึ่งมีหลายลักษณะได้แก่ การเปลี่ยนแปลงขนาดรูป (Scaling), การหมุนรูป (Rotation), การเคลื่อนย้ายรูป (Translation), การสะท้อนรูป (Reflection) ต่อจากนั้นเป็นการกล่าวถึงระบบนิกัดโฮโมจีเนียส (Homogeneous Coordinate System) ซึ่งนำมาใช้เพื่อการเปลี่ยนแปลงรูปรอบจุดใดๆ การกำหนดหน้าต่างแสดงภาพ (window) การย่อขยายรูป (Zooming) และการนำภาพที่สร้างขึ้นไปแสดงบนจอภาพ
- การสร้างภาพในระบบ 3 มิติ ซึ่งกล่าวถึงระบบนิกัดซึ่งมีทั้งที่เป็นแบบนิกัดมุมฉาก (Rectangular Coordinate) และนิกัดทรงกลม (Spherical Coordinate) ซึ่งมีความสัมพันธ์กัน การเปลี่ยนแปลงของรูป 3 มิติซึ่งมีลักษณะคล้าย ๆ กับการเปลี่ยนแปลงใน 2 มิติ คือ การแปลงขนาดรูป (Scaling), การหมุนรูป (Rotation), การเคลื่อนย้ายรูป (Translation), การสะท้อนรูป (Reflection) แต่สิ่งที่เพิ่มขึ้นมานอกเหนือจากนั้นก็คือ การแสดงภาพ 3 มิติบนจอภาพ 2 มิติ และการวัดความลึกของภาพ 3 มิติบนจอภาพ
- การกำจัดเส้นและพื้นผิวที่มองไม่เห็นในภาพ 3 มิติ โดยการแสดงวิธีการต่าง ๆ ในการกำจัดเส้นและพื้นผิวได้แก่ วิธีการทางเรขาคณิต (Geometric Computation), Image Space Algorithm เป็นต้น
- การกำหนดแสงและการให้แสงเงา (Shading) เมื่อมีแสงมากระทบพื้นผิวนั้น ๆ ซึ่งเป็นการกล่าวถึง หลักการทางคณิตศาสตร์ที่นำมาใช้สำหรับการกำหนดแหล่งกำเนิดแสง (Light Source) ในแต่ละพื้นผิว (Surface) ซึ่งจะมีความสัมพันธ์กับจุดมอง (Viewpoint) ในการที่จะกำหนดความเข้มของแสงบนพื้นผิวนั้นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาค 2 : เนื้อหากล่าวถึงการทดลองและประยุกต์ใช้ทฤษฎีต่างๆ โดยแบ่งเนื้อหาออกเป็นดังนี้

- การนำทฤษฎีพื้นฐานจากภาค 1 มาใช้ทดลองเขียนเป็นโปรแกรมเพื่อทดสอบ
- ขั้นตอนในการพัฒนา กล่าวถึงวิธีการและขั้นตอนในการพัฒนาโปรแกรมให้สามารถแสดงผลได้ตามต้องการ
- วิธีการใช้งาน กล่าวถึงวิธีการกำหนดแหล่งกำเนิดแสง การกำหนดมุมมองและความเข้มแสง เป็นต้น
- คำเสนอแนะเพิ่มเติม กล่าวถึงจุดที่ควรนำโปรแกรมไปพัฒนาต่อไป และการนำทฤษฎีต่าง ๆ และวิธีการต่าง ๆ ไปใช้ให้เห็นจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการ

2.1 การสร้างภาพในระบบ 2 มิติ (2-Dimensional Picture)

2.1.1 จุดในระบบแกนคาร์ทีเซียน (Cartesian Coordinate System) มีการกำหนดจุด (x,y) ใด ๆ ในระนาบนั้นด้วยเมตริกซ์ (Matrix) ขนาด 1x2 ได้เป็น [x,y] และจากคุณสมบัติของเมตริกซ์ เราจะสามารถแปลงขนาด (Scaling), หมุน (rotation), สะท้อน (reflection) จุดนั้นโดยเอาเมตริกซ์ของจุดนั้นมาคูณกับเมตริกซ์ในการแปลง (Transformation Matrix) ขนาด 2x2 ดังนี้

$$(x,y)\begin{bmatrix} A & B \\ C & D \end{bmatrix} = [(Ax + Cy), (Bx + Dy)]$$

ซึ่งจะอธิบายดังนี้

2.1.1.1 การแปลงขนาด (Scaling) กระทำโดยใช้เมตริกซ์ในการแปลง ดังนี้

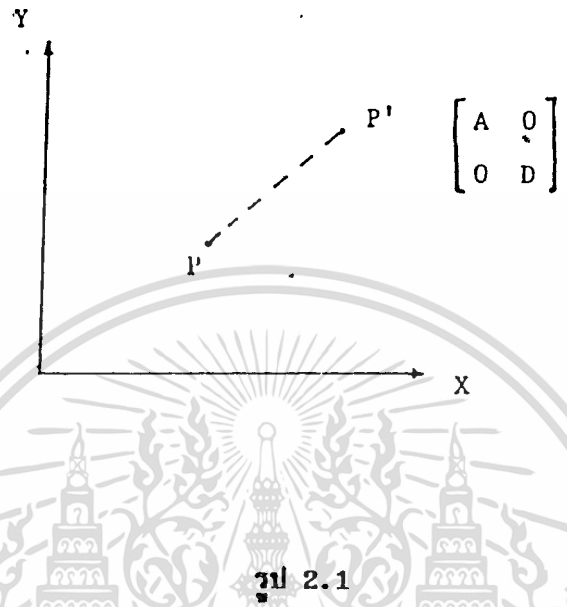
$$(x,y)\begin{bmatrix} A & 0 \\ 0 & D \end{bmatrix} = (Ax, Dy)$$

ซึ่งเป็นการขยายตำแหน่งของจุดไป A เท่าในแกน x และ D เท่าในแกน y ดังรูป 2.1

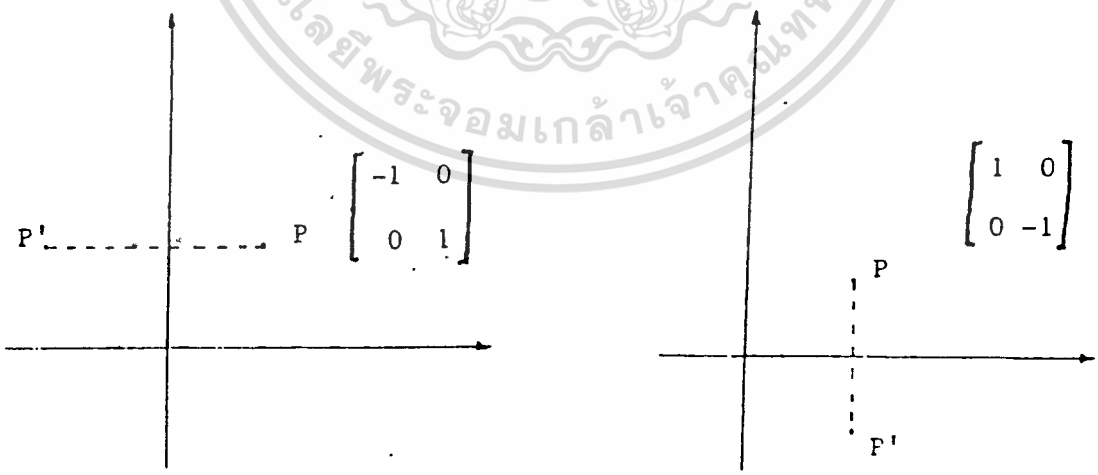
2.1.1.2 การสะท้อน (Reflection) กระทำโดยใช้เมตริกซ์ในการแปลงดังนี้

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.1



รูป 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือ $(x,y)\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} = (x, -y)$ (เป็นการสะท้อนกับแกน x)

ดังจะเห็นได้จากรูปที่ 2.2

2.1.1.3 การหมุน (Rotation) กระทำโดยใช้เมตริกซ์ในการแปลง ดังนี้

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

ซึ่งจะหมุนจุดนี้ไปเป็นมุม θ ทวนเข็มนาฬิการอบจุดกำเนิด (origin) ดังรูป 2.3

2.1.1.4 การเคลื่อนย้าย (Translation) ไม่ได้กระทำโดยนำเอาเมตริกซ์ในการแปลงมาคูณ แต่จะนำเอาระยะทางสัมพัทธ์ที่จุดจะเคลื่อนที่ไปในแนวต่าง ๆ มาบวกเข้าไป ดังนี้

$$(x,y) + (L,M) = (x+L,\ y+M)$$

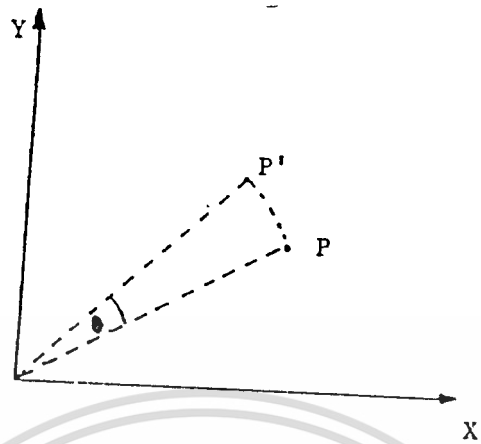
ซึ่งจะย้ายจุดไปในแนวแกน x เท่ากับ L หน่วย และไปในแนวแกน y เท่ากับ M หน่วย ดังรูป 2.4

ในการแทนจุดด้วยเมตริกซ์ ขนาด 1x2 นี้ จะเห็นได้ว่า เราไม่สามารถทำการเคลื่อนย้ายได้ โดยการใช้เมตริกซ์มาคูณเข้าไป และการกระทำจะสัมพัทธ์กับจุดกำเนิด (0,0) เสมอ

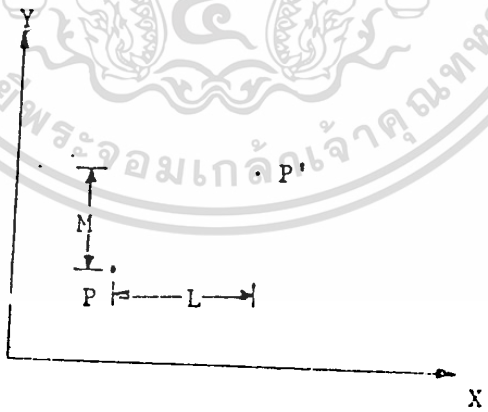
2.1.2 <u>จุดในระบบโฮโมจีเนียส (Homogeneous)</u>

ในระบบโฮโมจีเนียส เราจะใช้เมตริกซ์ขนาด 3x3 ซึ่งจะทำให้ปัญหาในการแทนจุดด้วยเมตริกซ์ขนาด 1x2 ในระบบคาร์ทีเซียนหมดไป โดยเราจะแทนจุด (x,y) ในระบบคาร์ทีเซียนด้วย (x,y,H) ในระบบโฮโมจีเนียส หรือเป็น (x/H, y/H, 1) โดย H เป็นค่าที่ไม่เท่ากับศูนย์ และถ้า H มีค่า = 1 จุด (x,y,1) ก็คือจุด (x,y) ในระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.3



รูป 2.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คาร์ทีเซียนนั่นเอง และถ้าในกรณีของจุดหลาย ๆ จุด เราจะได้เมตริกซ์เป็น

$$\begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots \\ x_N & y_N & 1 \end{bmatrix}$$

ซึ่งประกอบด้วยจุด N จุด

2.1.2.1 การเคลื่อนย้ายจุด กระทำได้ดังนี้

$$(x,y,1)\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ L & M & 1 \end{bmatrix} = (x+L,\ y+M,\ 1)$$

ซึ่งจะเคลื่อนจุดไป L หน่วยในแกน x และ M หน่วยในแกน Y ดัง รูป 2.5

2.1.2.2 การแปลงขนาด กระทำได้ดังนี้

$$(x,y,1)\begin{bmatrix} A & 0 & 0 \\ 0 & D & 0 \\ 0 & 0 & 1 \end{bmatrix} = (Ax,\ Dy,\ 1)$$

ซึ่งให้ผลเหมือนในข้อ 2.1.1.1

2.1.2.3 การหมุนจุดทำได้ดังนี้

023191

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$(x,y,1)\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} = (x\cos\theta - y\sin\theta,\ x\sin\theta + y\cos\theta)$$

ซึ่งให้ผลเหมือนข้อ 2.1.1.3

2.1.3 <u>การทำการแปลงหลาย ๆ ชนิด</u>

ถ้าเราต้องการหมุนจุดนั้นรอบ ๆ จุดอ้างอิงใด ๆ นอกจากจุด (0,0) เราจะทำได้ดังนี้

$$(x,y,1)\underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ L & M & 1 \end{bmatrix}}_{T1}\underbrace{\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{T2}\underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -L & -M & 1 \end{bmatrix}}_{T3}$$

ซึ่งเป็นการย้ายจุดก่อน แล้วจึงทำการหมุน และจึงทำการย้ายจุดกลับมาที่เดิม และเราสามารถที่จะรวมเมตริกซ์เหล่านั้นเข้าด้วยกันโดยการคูณเข้าก่อน แล้วจึงนำไปคูณกับเมตริกซ์ของจุด (Sequential transformation) เช่นในตัวอย่างจะได้ $T = T1 \times T2 \times T3$

ในกรณีของการทำการแปลงขนาด ก็จะเป็นทำนองเดียวกัน

2.1.4 <u>การนำภาพไปแสดงบนจอภาพ</u> (Screen)

เนื่องจากว่าระบบแกนที่เรากำลังพิจารณาอยู่นี้ มีค่าความยาวไม่จำกัดและมีพิกัดเป็นเลขจำนวนจริง แต่ระบบของจอภาพนี้จะมีจำนวนจุดและขอบเขตที่แน่นอน และมีพิกัดเป็นเลขจำนวนเต็มดังนั้น เราจึงต้องกำหนดขอบเขตของระบบแกนที่แน่นอน และทำการเปลี่ยนค่าของพิกัดให้เหมาะสมที่จะแสดงบนจอภาพได้

2.1.4.1 การกำหนดขอบเขตของภาพและการตัดแต่งภาพ (Window Clipping)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อมีการกำหนดขอบเขตของภาพแล้ว และปรากฏว่ามีบางส่วนของภาพนั้นอยู่นอกขอบเขตการแสดงภาพออกไป จะต้องมีการตัดแต่งภาพ (Clipping) ดังแสดงในรูปที่ 2.5

ถ้าเรากำหนดฟังก์ชันทางตรรก ดังนี้

T(I) เป็นค่าความจริงของจุด I ที่อยู่นอกขอบเขตของภาพทางด้านบน

B(I) เป็นค่าความจริงของจุด I ที่อยู่นอกขอบเขตของภาพทางด้านล่าง

R(I) เป็นค่าความจริงของจุด I ที่อยู่นอกขอบเขตของภาพทางด้านขวา

L(I) เป็นค่าความจริงของจุด I ที่อยู่นอกขอบเขตของภาพทางด้านซ้าย

ถ้าให้ จุดเริ่มของเส้นตรงเส้นหนึ่งคือ 1 และจุดปลายคือ 2 เราจะพบว่า

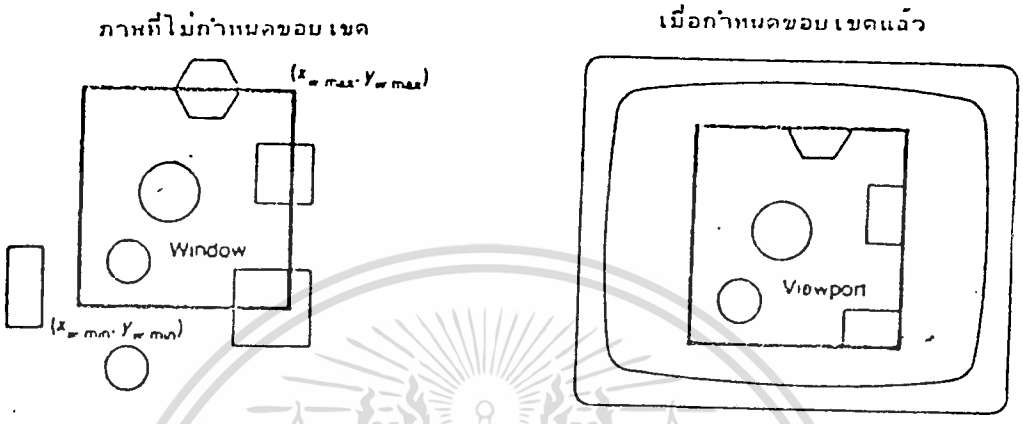
- ถ้า T(1)+B(1)+R(1)+L(1)+T(2)+B(2)+R(2)+L(2) = 0
  แล้วเส้นตรงเส้นนั้นจะอยู่ในขอบเขตของภาพคือ ไม่มีการตัดออก เช่น เส้นตรงหมายเลข 1 ในรูปที่ 2.6
- ถ้า [T(1)T(2)] + [B(1)B(2)] + [R(1)R(2)] + [L(1)L(2)] = 0
  แล้ว เส้นตรงเส้นนั้นจะไม่อยู่ภายในขอบเขตของภาพเลย ดังเช่น เส้นหมายเลข 2 และ 3 ในรูปที่ 2.6
- ถ้า [T(1)T(2)] + [B(1)B(2)] + [R(1)R(2)] + [L(1)L(2)] = 0
  แล้วจุดปลายของเส้นนั้นทั้งสองจุดจะไม่อยู่ในขอบเขตของภาพ แต่บางส่วนของเส้นตรงอาจจะปรากฏหรือไม่ปรากฏในขอบเขตของภาพก็ได้ ดังเส้นหมายเลข 5 และ 6 ในรูปที่ 2.6

นอกจากนี้ ถ้ามีจุดปลายข้างหนึ่งอยู่ในขอบเขตของภาพ และอีกปลายข้างหนึ่งอยู่นอกขอบเขตของภาพ เส้นตรงเส้นนั้นจะปรากฏเพียงบางส่วนดังเส้นตรงหมายเลข 4 ในรูป 2.6

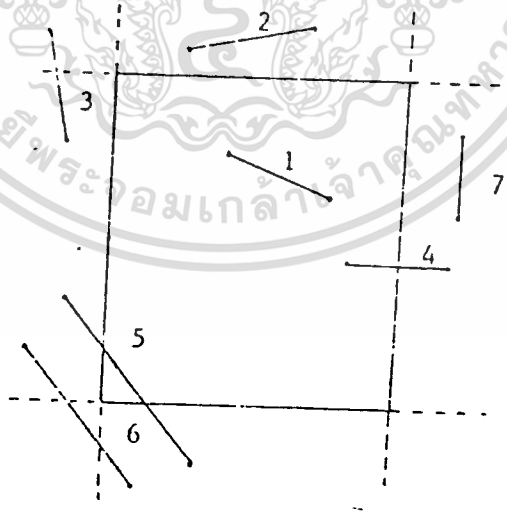
2.1.4.2 การนำภาพไปปรากฏบนจอภาพ

เนื่องจากจอภาพนั้นเราสามารถถือว่าเป็นระบบโคออร์ดิเนตใหม่จึงต้องมีการแปลง (Transform) จากระบบโคออร์ดิเนตเดิมที่กำหนดไว้ ถ้าเราให้ขอบเขตของภาพเดิมเป็น $X_w \min <= X <= X_w \max$ และ $Y_w \min <= Y <= Y_w \max$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.5



รูปที่ 1.6

รูป 2.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และในจอภาพเป็น $X_v \text{ min} <= X <= X_v \text{ max}$ และ $Y_v \text{ min} <= Y <= Y_v \text{ max}$

และเนื่องจากจอภาพมีพิกัดสูงสุดและต่ำสุดต่างไปจากระบบโคออร์ดิเนตเดิม จึงต้องใช้ตัวคูณในการแปลงขนาด (Scaling Factor) เพื่อย่อหรือขยายภาพเดิมให้ลงในจอภาพอย่างเหมาะสม ทั้งหมดนี้เราจะได้

$$(X_v, Y_v, 1) = (X_w, Y_w, 1) \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -X_w \text{ min} & -Y_w \text{ min} & 1 \end{bmatrix} \times$$

$$\begin{bmatrix} S_1 & 0 & 0 \\ 0 & S_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ X_v \text{ min} & Y_v \text{ min} & 1 \end{bmatrix}$$

$$= [S_1(X_w - X_w \text{ min}) + X_v \text{ min}, S_2(Y_w - Y_w \text{ min}) + Y_w \text{ min}, 1]$$

โดย

$$S_1 = \frac{X_v \text{ max} - X_v \text{ min}}{X_w \text{ max} - X_w \text{ min}}$$

$$S_2 = \frac{Y_v \text{ max} - Y_v \text{ min}}{Y_w \text{ max} - Y_w \text{ min}}$$

และถ้า S_1 มีค่าไม่เท่ากับ S_2 แล้ว รูปที่ได้จะบิดเบือน (Distorted)

2.1.4.3 การแสดงภาพบนจอภาพ

เนื่องจากลักษณะการวางแนวของจอภาพในเครื่องไมโครคอมพิวเตอร์บาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชนิด เช่น IBM PC จะวางแกน Y นั้นกลับหัว คือจุดกำเนิดจะอยู่ที่มุมบนซ้าย แทนที่จะเป็นมุมล่างซ้าย ดังนั้นจึงต้องมีการแปลงอีกครั้งโดยให้มีจุดกำเนิดของแกนอยู่ที่ (L,M) เราจะได้

$$(X_s, Y_s, 1) = (X_v, Y_v, 1) \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ L & M & 1 \end{bmatrix}$$

โดย $(X_s, Y_s, 1)$ เป็นจุดที่จะปรากฏบนจอภาพ และ

$(X_v, Y_v, 1)$ เป็นจุดที่จะปรากฏในคาร์ทีเซียนโคออร์ดิเนต

นอกจากนี้เนื่องจากจอภาพมักจะมีค่าจำนวนจุด (Dot) ต่อความยาวที่ต่างกันระหว่างแนวนอนและแนวดิ่ง จึงจำเป็นที่จะต้องมีตัวคูณอีกตัวหนึ่งเพื่อแก้ไขการบิดเบือนนี้ ถ้าเราให้

ความยาวในแนวนอนของจอ (Horizontal)

$$H = \frac{\text{ความยาวในแนวนอนของจอ (Horizontal)}}{\text{จำนวนจุดมากที่สุดในแนวนอน}}$$

ความยาวในแนวดิ่งของจอ (Vertical)

$$V = \frac{\text{ความยาวในแนวดิ่งของจอ (Vertical)}}{\text{จำนวนจุดมากที่สุดในแนวดิ่ง}}$$

$$SCF = \frac{V}{H}$$ ซึ่งเป็นตัวคูณที่ต้องการ

(Screen Coordinate Factor)

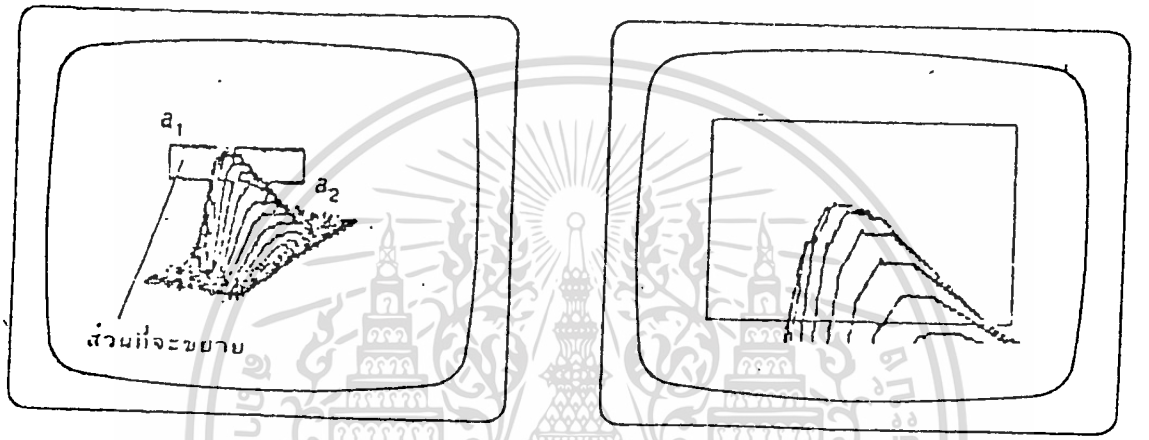
เราจะเอา SCF คูณเข้าไปเฉพาะที่ค่า X เท่านั้น

2.1.4.4 การย่อขยายภาพ (Zooming)

การย่อขยายภาพบางส่วน กระทำได้ง่ายตามหลักที่ได้กล่าวมาแล้วโดยการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนดขอบเขตการแสดงภาพเสียใหม่ เฉพาะส่วนที่ต้องการขยายแล้วจากนั้นก็นำส่วนขอบเขตการแสดงภาพนี้นไปปรากฏบนจอ เป็นขอบเขตของจอภาพสูงสุดและต่ำสุด ก็จะได้ภาพขยายขนาดตามต้องการ ตามรูป 2.7



รูป 2.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 การสร้างภาพ 3 มิติ (3-Dimensional Picture)

ระบบแกนในระบบ 3 มิติ ที่นิยมใช้กันมี 2 ชนิดคือ

- แกนพิกัดฉาก (Rectangular Coordinate) มีอาร์กิวเมนท์ (Argument) เป็น (x,y,z)
- แกนพิกัดทรงกลม (Spherical Coordinate) มีอาร์กิวเมนท์ (Argument) เป็น (D,θ,Ø)

ดังแสดงระบบแกนทั้งสองในรูปที่ 2.8

2.2.1 เมตริกซ์ในการแปลง (Transformation Matrix)

ในระบบแกน 3 มิติ ก็เช่นเดียวกันกับระบบแกน 2 มิติ คือมีการกำหนดจุดต่าง ๆ ในรูปของเมตริกซ์ ซึ่งใช้ในการขยาย, เคลื่อนย้าย, และหมุน จะมีลักษณะดังนี้

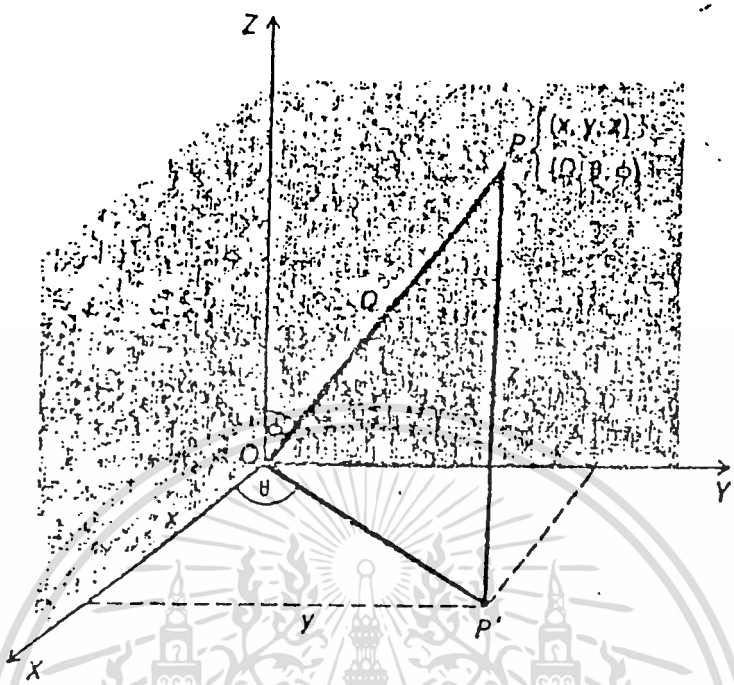
$$\begin{bmatrix} A & B & C & | & 0 \\ D & E & F & | & 0 \\ G & H & I & | & 0 \\ \text{---} & \text{---} & \text{---} & | & \\ L & M & N & | & 1 \end{bmatrix} \quad \text{ซึ่งแบ่งเป็น 3 ส่วน คือ} \Rightarrow \quad \begin{bmatrix} \multicolumn{3}{c}{\text{I}} & | & \text{III} \\ \multicolumn{3}{c}{(3\text{x}3)} & | & 4 \\ \multicolumn{3}{c}{\text{------}} & | & \text{X} \\ \multicolumn{3}{c}{\text{II}} & | & 1 \\ \multicolumn{3}{c}{(1\text{x}3)} & | & \end{bmatrix}$$

ซึ่ง ส่วนที่ 1 จะใช้ในการ ขยาย สะท้อน และการหมุน

ส่วนที่ 2 จะใช้ในการ แปลงในรูปของการเคลื่อนย้าย

ส่วนที่ 3 เพิ่มเข้ามาเพื่อให้ใช้ในการคูณได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.8

2.2.1.1 การเคลื่อนย้ายจะเป็นดังนี้

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

โดย L เป็นการเคลื่อนย้ายในแกน X

M เป็นการเคลื่อนย้ายในแกน Y

N เป็นการเคลื่อนย้ายในแกน Z

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1.2 การแปลงขนาดจะเป็นดังนี้

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

โดย A เป็นอัตราขยายในแกน X

E เป็นอัตราขยายในแกน Y

I เป็นอัตราขยายในแกน Z

2.2.1.3 การสะท้อน เป็นการสะท้อนกับระนาบจึงมีเมตริกซ์ 3 แบบคือ

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$ สะท้อนระนาบ XY ;

$$\begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$ สะท้อนระนาบ YZ

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$ สะท้อนระนาบ XZ

2.2.1.4 การหมุนเป็นการหมุนรอบแกนใด ๆ จึงมีเมตริกซ์ 3 แบบ คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

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

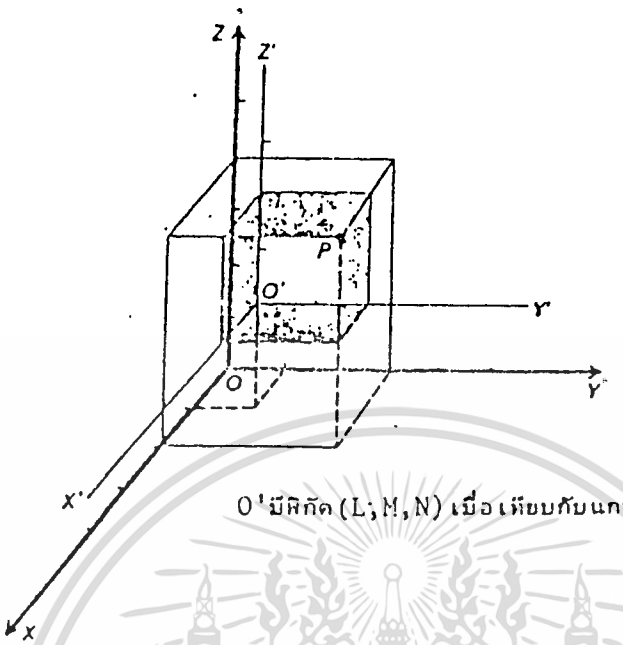
$$\begin{bmatrix} \cos\theta & 0 & -\sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$ หมุนรอบแกน Y

เมตริกซ์ในการแปลงต่างๆเหล่านี้ สามารถนำมาคูณต่อกันเพื่อให้ได้เมตริกซ์ที่ทำการแปลงหลายๆอย่างพร้อมกันได้ (Sequential Tranformation)

2.2.2 <u>การเปลี่ยนแปลงแกน</u> (Change of Coordinate System)

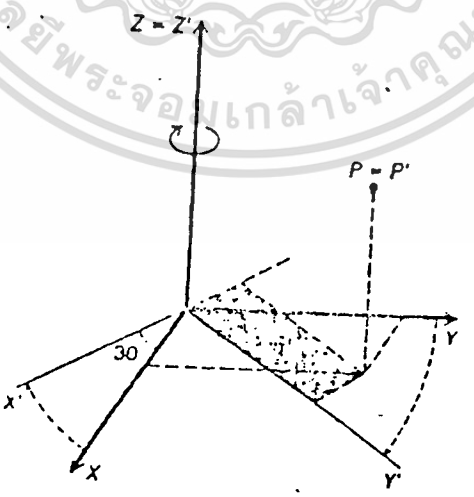
ถ้าพิจารณาให้จุดในระบบนั้นอยู่กับที่ แต่ให้ระบบแกนนั้นมีการเปลี่ยนแปลงแทน เราเรียกลักษณะนี้ว่า "การเปลี่ยนแกน" ซึ่งจะนำไปใช้ในการสร้างภาพ 3 มิติ บนระนาบ 2 มิติ การเคลื่อนย้ายแกนดังแสดงในรูป 2.9 ซึ่งจะได้เมตริกซ์ในการแปลงของจุด เช่น การย้ายแกนจากจุด (0,0,0) ไปยัง (L,M,N) จะได้เมตริกซ์ในการแปลงเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



O'มีพิกัด (L;M,N) เมื่อเทียบกับแกน (X,Y,Z)

รูป 2.9



รูป 2.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -L & -M & -N & 1 \end{bmatrix}$$

ซึ่งเป็นส่วนกลับ (Inverse) ของเมตริกซ์ในการแปลงที่ใช้ในการเคลื่อนย้าย

และการหมุนแกนตามเข็มนาฬิกาดังรูป 2.10 ซึ่งหมุนแกนไป -30 องศา ก็จะมีเมตริกซ์ในการแปลง เป็นการแปลงของจุดที่หมุนไป 30 องศา ดังนี้

$$\begin{bmatrix} \cos 30^{\circ} & \sin 30^{\circ} & 0 & 0 \\ -\sin 30^{\circ} & \cos 30^{\circ} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

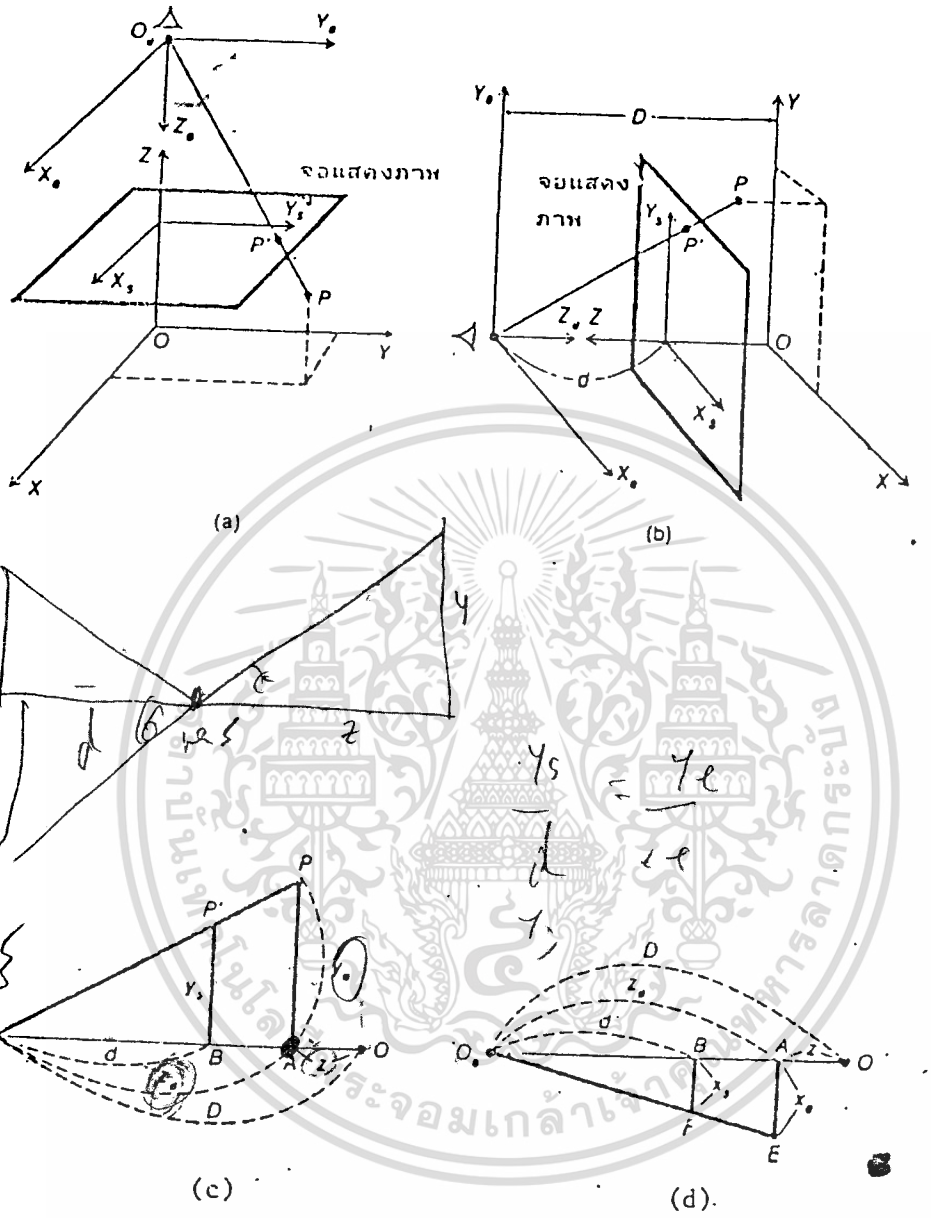
2.2.3 การมองภาพ 3 มิติ (Viewing in three dimensions)

2.2.3.1 การมองภาพจากจุดกำเนิด (Central Projection)

จากรูป 2.11 ซึ่งแสดงระบบแกนสองระบบที่มีแกน Z เข้าชนกัน ระบบหนึ่งคือระบบแกนในความจริงหรือระบบแกนของวัตถุ (World Coordinate System) และระบบแกนของตา (Eye Coordinate System) (X_e, Y_e, Z_e) ซึ่งเป็นลักษณะของการมองภาพ จะเห็นได้ว่าทั้งสองระบบมีค่าทางแกน X และ Y เท่ากัน จะต่างกันที่แกน Z เท่านั้น

แต่ระบบแกนของจอภาพนั้นมีเพียง 2 มิติ คือ แกน X และ แกน Y ถ้าเรานำเอาจอภาพ (Screen) ไปขวางกลางระหว่างจุดของวัตถุและสายตาของคน ลากเส้นเชื่อมระหว่างจุด P กับจุดสายตาดังรูป เส้นตรงเส้นนี้จะตัดกับจอภาพที่จุด P′ ซึ่งจุดนี้จะเป็นจุดของภาพที่ควรจะปรากฏบนจอ โดยแสดงดังรูป 2.11 (c) และ (d) ซึ่งเราจะได้ว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Y_s Y

$$Y_s = d\frac{}{} = d\frac{}{}$$

$$- Z_s \qquad D-Z$$

และพิจารณาเช่นเดียวกันนี้ กับ X_s เราจะได้ว่า

- X_s และ Y_s เป็นฟังก์ชันเส้นตรง (linear function) ของค่า d ดังนั้น ถ้าเราลดระยะจากจอภาพไปยังตา เราจะได้ภาพที่เล็กลง
- การเพิ่มระยะจากวัตถุไปยังจอภาพ (D) ก็จะทำให้ได้ภาพที่เล็กลง
- การเพิ่มระยะจากตาไปยังจอภาพ จน D เข้าใกล้อนันต์ และ D = d ซึ่งจะทำให้ค่าของ X_s และ Y_s ไม่ขึ้นกับค่า Z_s จะเกิดภาพในลักษณะที่เรียกว่า "การมองภาพแบบขนาน" (Parallel Projection) ภาพที่เป็นการมองภาพแบบขนาน ได้แก่ ภาพไอโซเมตริก

2.2.3.2 การมองภาพจากจุดใด ๆ (Arbitrary Projection)

การมองภาพจากจุดใด ๆ ที่ห่างเป็นระยะ D จากจุดกำเนิดของแกนจริงนั้น กระทำได้โดยการย้ายจุดกำเนิดของแกนจริงมายังจุดที่เป็นจุดมอง โดยมีแกน Z พุ่งเข้าหาจุดกำเนิด และแกน X อยู่ในแนวนอนชี้ไปทางขวาของผู้มอง แกน Y ชี้ขึ้น ดังระบบแกนระบบแกนของตา (X_e, Y_e, Z_e) ในรูป 2.12

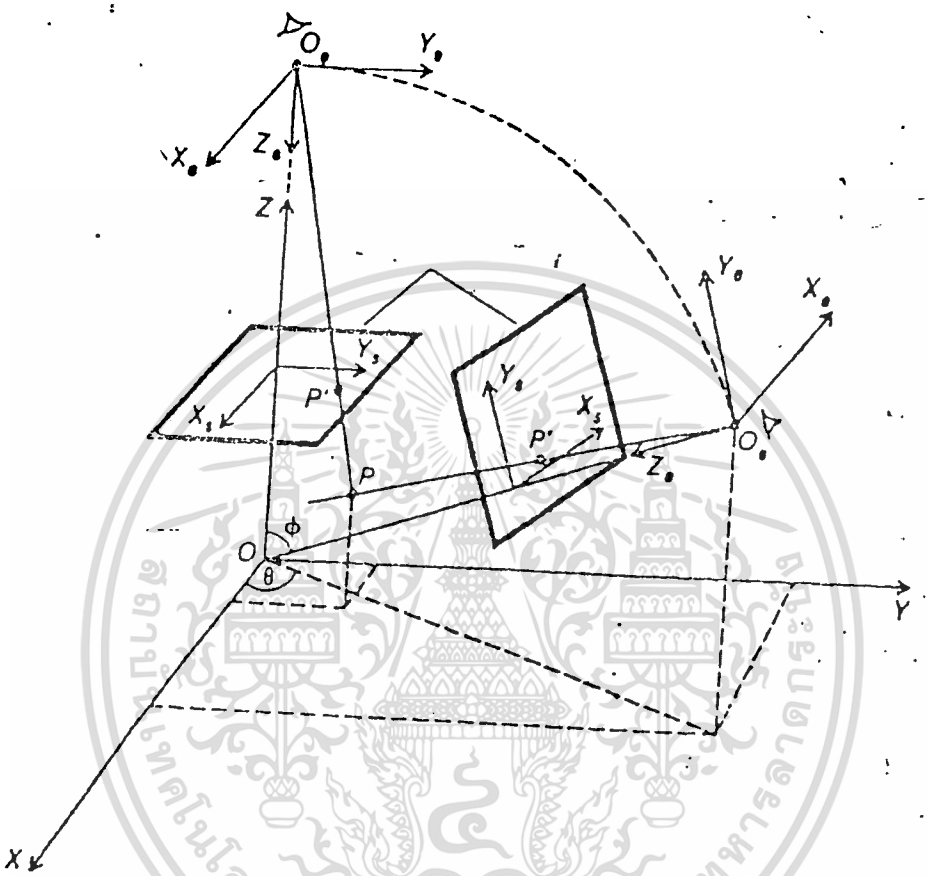
เราจะแสดงขั้นตอนการย้ายแกนของรูป 2.12 โดยจุด O_e ซึ่งเป็นจุดมองจะมีระยะห่างจากจุดกำเนิด เป็นระยะทาง D และเส้น OO_e ทำมุม θ กับระนาบ XZ และทำมุม ϕ กับแกน Z ดังแสดงในรูปที่ 2.13 และขั้นตอนการย้ายแกนมีขั้นตอนดังนี้

<u>ขั้นตอนที่ 1</u> เราจะย้ายแกนไปยังจุด O_e จะมีเมตริกซ์ในการแปลงดังนี้

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

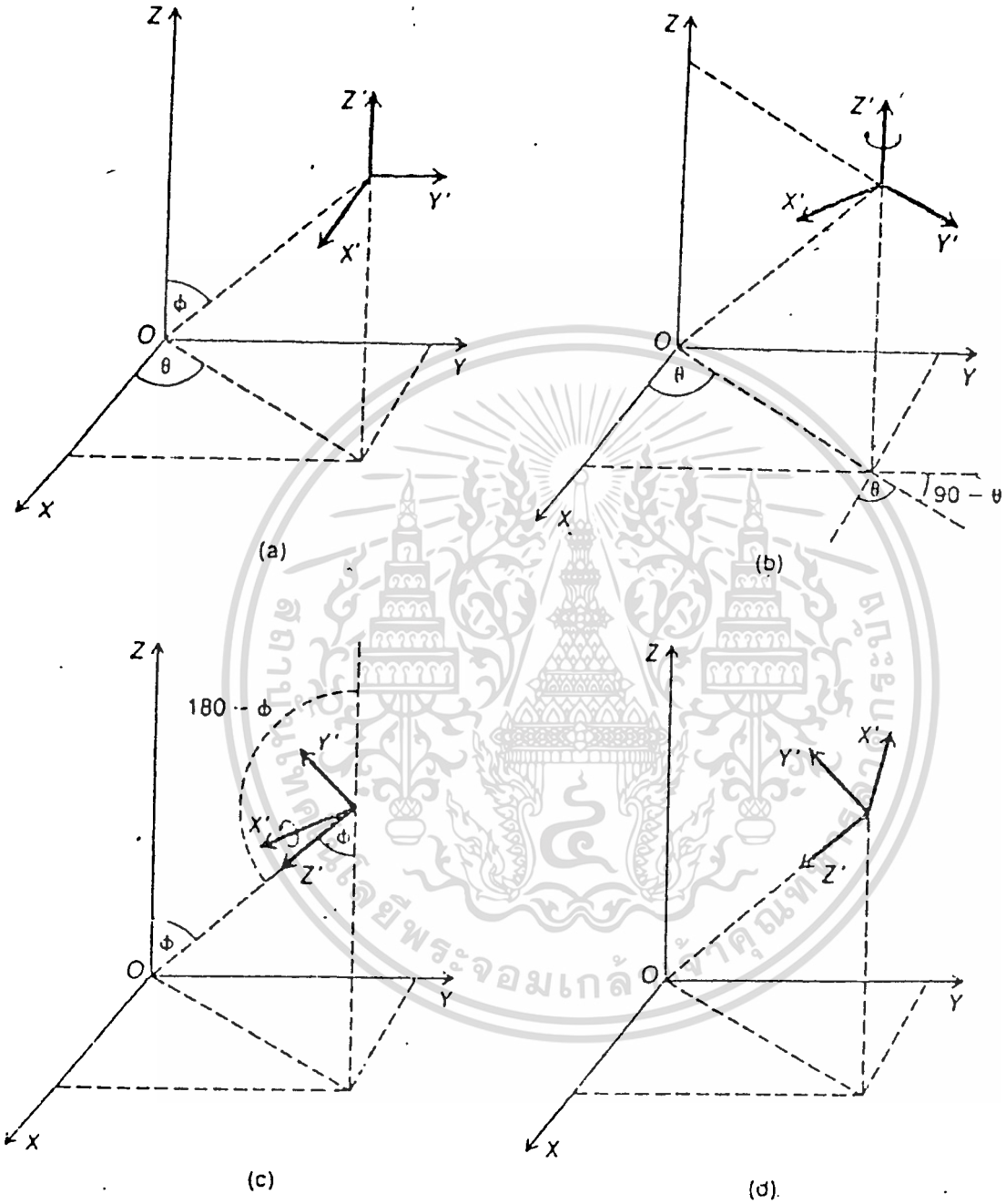
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(2)



รูป 2.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6

รูป 2.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 2 เราจะหมุนแกนรอบแกน Z ไปเป็นมุม $90^\circ - \theta$ จะได้เมตริกซ์ในการแปลง ดังรูป

$$T_2 = \begin{bmatrix} \sin\theta & \cos\theta & 0 & 0 \\ -\cos\theta & \sin\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{ดังรูป (b)}$$

ขั้นตอนที่ 3 เราจะหมุนแกนรอบแกน X' ไปเป็นมุม $180^\circ - \phi$ จะได้เมตริกซ์ในการแปลง คือ

$$T_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -\cos\phi & -\sin\phi & 0 \\ 0 & \sin\phi & -\cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{ดังรูป (c)}$$

ขั้นตอนที่ 4 เราจะแปลงโคออร์ดิเนตโดยเปลี่ยนแกน X ไปในทางตรงกันข้าม จะได้เมตริกซ์ในการแปลง ดังรูป

$$T_4 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{ดังรูป (d)}$$

จากนั้นเราจะได้ เมตริกซ์ในการแปลงรวมเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$T1 * T2 * T3 * T4 = \begin{bmatrix} -\sin\theta & -\cos\theta\cos\phi & -\cos\theta\sin\phi & 0 \\ \cos\theta & -\sin\theta\cos\phi & -\sin\theta\sin\phi & 0 \\ 0 & \sin\phi & -\cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$(X_e, Y_e, Z_e, 1) = (X, Y, Z, 1) * T$$

หรือ
$$X_e = -X\sin\theta + Y\cos\theta$$
$$Y_e = -X\cos\theta\cos\phi - Y\sin\theta\cos\phi + Z\sin\phi$$
$$Z_e = -X\cos\theta\sin\phi - Y\sin\theta\sin\phi - Z\cos\phi + D$$

และเมื่อนำไปพลอต (plot) ลงบนจอภาพ เราจะได้

$$X = d\frac{X_e}{Z_e}, \quad Y = d\frac{Y_e}{Z_e}$$

โดย d คือระยะจากจุดมองไปยังจอภาพ

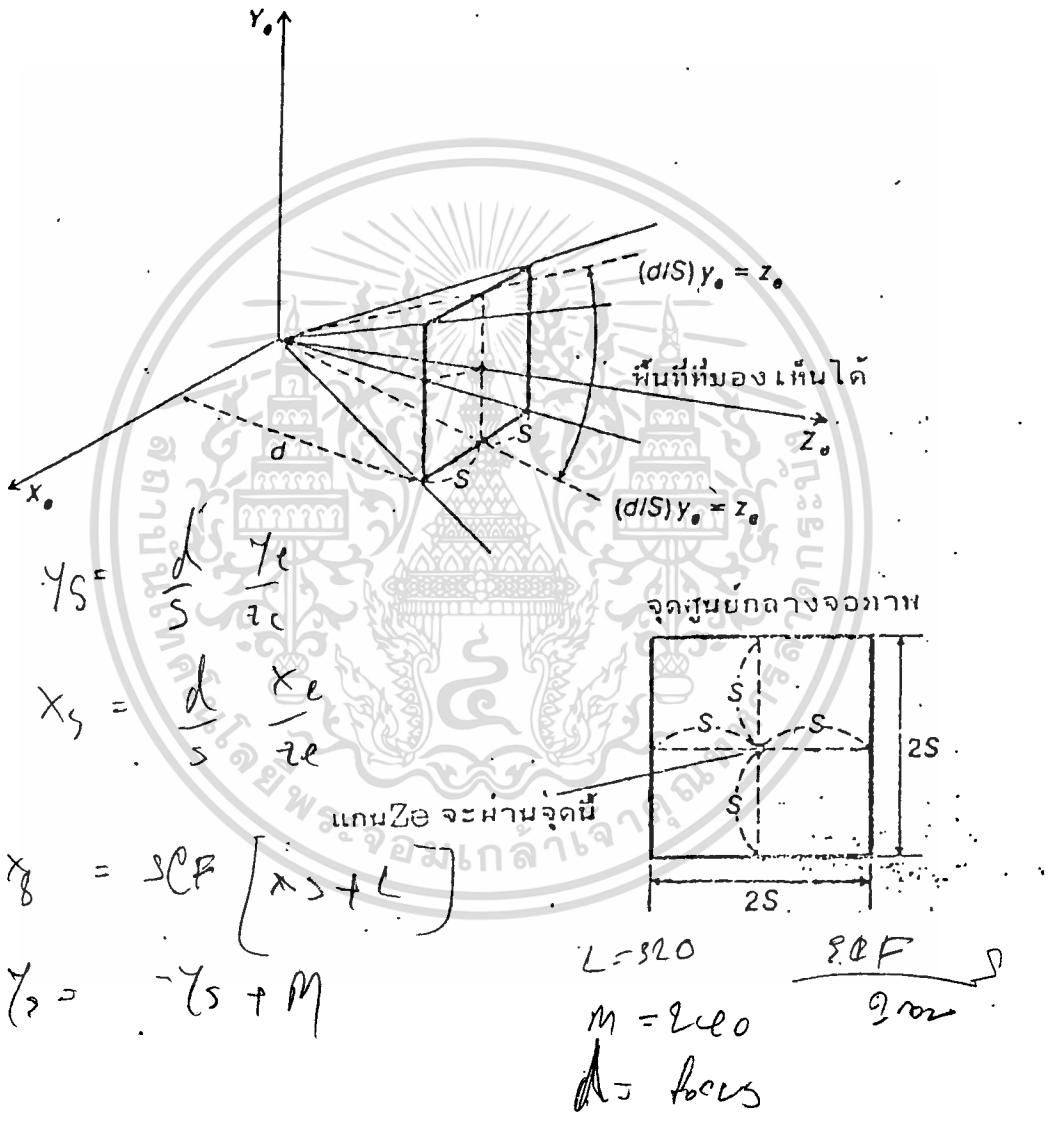
2.2.3.3 ระบบแกนจอภาพแบบโฮโมจีเนียส

ในระบบแกนจอภาพจากจุดกำเนิดเราจะใช้ ค่า D และ d เป็นตัวควบคุมขนาดของภาพ แต่ต่อไปเราจะเปลี่ยนโดยใช้อัตราส่วนของจุดกำเนิดจอภาพ กับระยะ d แทน ดังแสดงในรูป 2.14 และมีสมการดังนี้

$$x = \left[\frac{d}{S}\right]\frac{X_e}{Z_e}, \quad y = \left[\frac{d}{S}\right]\frac{Y_e}{Z_e}$$

ซึ่งอัตราส่วน d/S นี้จะมีผลต่อขนาดของรูปที่จะปรากฏบนจอ เปรียบเสมือน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นความยาวโฟกัสของกล้องถ่ายรูป ซึ่งถ้าโฟกัสยาว (เลนส์เทเลโฟโต้) รูปก็จะมีขนาดใหญ่ และถ้าโฟกัสสั้น (เลนส์ไวด์แองเกิ้ล) รูปก็มีขนาดเล็กลง

2.2.4 ความลึกของภาพ (Perspective Depth)

ในการมองภาพแบบเปอร์สเปคตีฟนั้น ความลึกหรือระยะ Z_e นี้มีผลต่อขนาดของภาพที่ปรากฏ โดยจุดที่ปรากฏนั้น ถ้ามีค่า Z_e มาก ก็จะมีค่าน้อยทำให้ขนาดของภาพเล็กลง แต่ในการมองภาพแบบขนาน ความลึกจะไม่มีผลต่อขนาดของภาพ ดังนั้น

$$X = X_e \quad \text{และ} \quad Y = Y_e$$

เนื่องจากภาพที่มองแบบเปอร์สเปคตีฟนั้น ความลึกของภาพปรากฏอยู่ จะมีผลทำให้เราไม่สามารถมองเห็นรูปทรงนั้นได้ทุก ๆ พื้นผิว อันเนื่องมาจากมุมของพื้นผิวนั้นทำกับมุมมองของสายตาเกินกว่าที่จะมองเห็นได้ (ดังจะแสดงต่อไป) หรือถูกพื้นผิวอื่นที่อยู่ใกล้สายตามากกว่าบดบังไป ดังนั้น เราจึงต้องพิจารณาค่าความลึกของแต่ละจุดที่จะพลอตลงบนจอภาพด้วย

จากสมการทั่วไปของพื้นผิว คือ

$$Ax + By + Cz + D = 0$$

ถ้าเราแทนค่า x และ y จากโคออร์ดิเนตของจอภาพลงไป เราจะได้

$$Z_e = \alpha - \frac{\beta}{Z_e}$$

โดย $\alpha = -D/C$ และ $\beta = (d/CS)(AX_e + BY_e)$

เนื่องจากค่า A,B,C,D เราไม่สามารถกำหนดได้ ดังนั้นเราจึงต้องอาศัยการแทนค่าเข้าช่วย ถ้าเราให้ Z_e ซึ่งเป็นโคออร์ดิเนต ของแกนระบบสายตา มีค่า

$$K_1 <= Z_e <= K_2$$

จากนั้นเราจะกำหนดค่า Z_e ดังนี้

$$0 <= Z_e <= 1$$

เมื่อแทนค่า Z_e ลงไปในสมการหา Z_e เราจะได้สมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$0 = \alpha - \frac{\beta}{K_1}$$

$$1 = \alpha - \frac{\beta}{K_2}$$

แก้สมการหาค่า α และ β

$$\alpha = \frac{K_2}{K_2 - K_1}, \qquad \beta = \frac{K_1 K_2}{K_2 - K_1}$$

แทนค่า หา $Z_a = \dfrac{K_2 (Z_a - K_1)}{(K_2 - K_1) Z_a}$; $0 <= Z_a <= 1$

ซึ่งเป็นความลึกของแต่ละจุด (X_a, Y_a)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การกำจัดเส้นและพื้นผิวที่ถูกบดบัง (Hidden-line and Surface Removals)

ดังที่ได้กล่าวมาแล้วว่า การมองภาพในแบบ 3 มิติ ในความเป็นจริงนั้น เราไม่สามารถที่จะมองเห็นเส้นทุกเส้นและพื้นผิวทุกด้าน อันเนื่องมาจากมีการบังของพื้นผิวที่ใกล้ตากว่า และจากจุดที่มอง (view) ไม่สามารถมองเห็นได้ ดังนั้นในหัวข้อนี้เราจะกล่าวถึงการกำจัดเส้น และพื้นผิวที่ตาไม่สามารถมองเห็น

2.3.1 พื้นผิวที่ไม่สามารถมองเห็นได้ (Invisible surface)

ถ้าเรากำหนดจุดการมองดังในรูป 2.15 ที่จุด P เรามีความจริงที่ว่า

" ถ้ามุมของเวคเตอร์ตั้งฉากกับพื้นผิวที่มองที่ทำกับเวคเตอร์ของจุดมองกับจุดที่ตั้งฉากนั้นมีค่ามากกว่า 90 องศา (ไม่คิดเครื่องหมาย) แล้ว และพื้นผิวนั้นมิได้ถูกบังด้วยพื้นผิวอื่นที่ใกล้จุดสายตากว่าแล้ว พื้นผิวนั้นจะมองไม่เห็น"

ดังนั้นจากรูป 2.15 เราจะสามารถหาเวคเตอร์ n จาก

$$n = u \times v$$

ถ้าให้ เวคเตอร์จากจุดสายตาไปยังจุดที่เวคเตอร์ n ตั้งฉากกับพื้นผิว คือ l เราจะได้ว่า

$$n \cdot l = |n|\,|l| \cos \phi$$

โดย θ จะเป็นมุมระหว่าง n กับ l ซึ่งจะต้องน้อยกว่า 90 องศา (ไม่คิดเครื่องหมาย)

ดังนั้น ถ้า $\cos \theta > 0$ แล้ว $n \cdot l$ จะต้อง $> \phi$ ด้วย

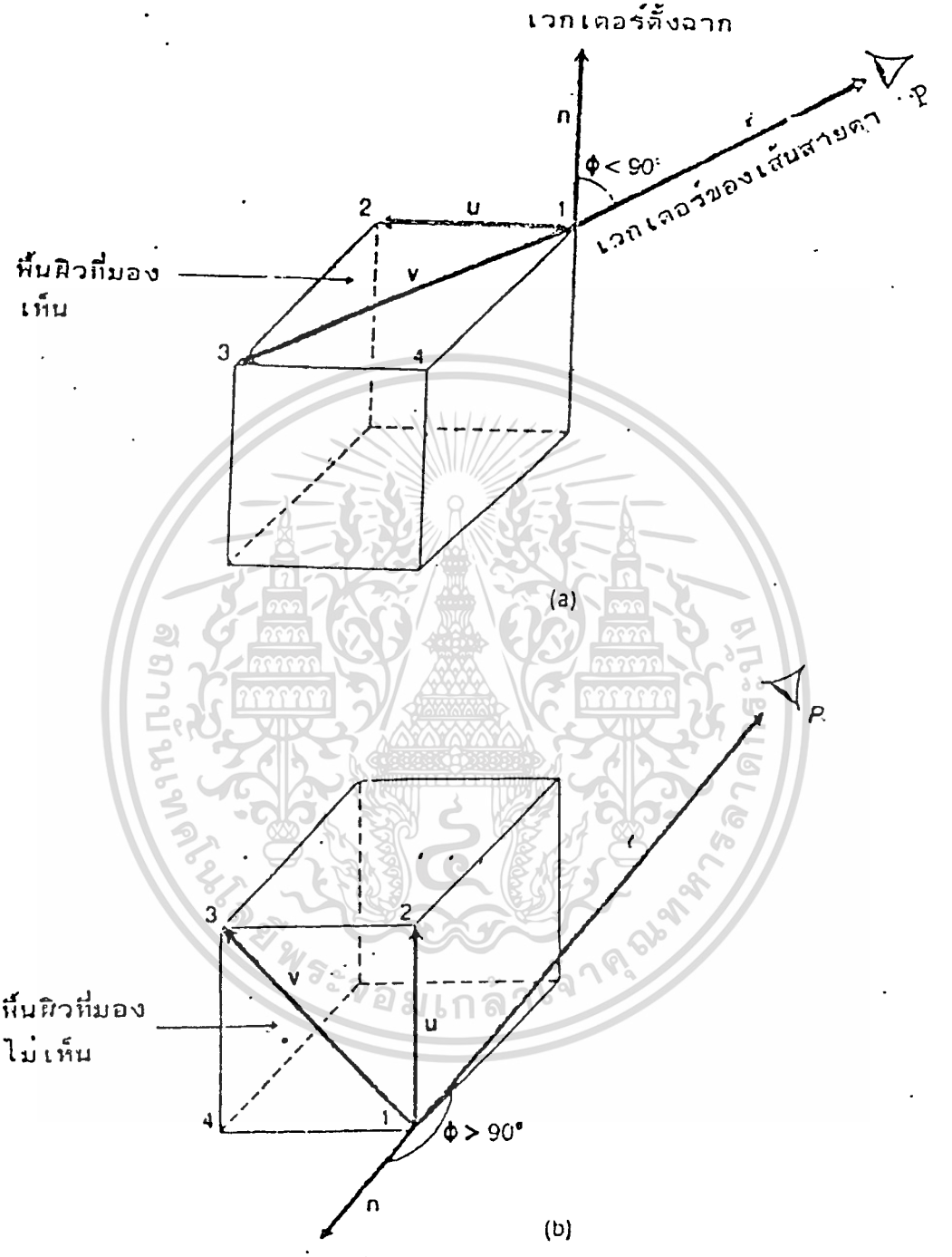
เราจึงเพียงพิจารณาค่า $n \cdot l$ เท่านั้นก็พอ ซึ่งหาค่าในทางคอมพิวเตอร์ได้ไม่ยากนัก

2.3.2 การซ้อนและบดบังกันของพื้นผิว

ถ้าเรามีการวาดภาพ 3 มิติ ของวัตถุหลาย ๆ ชิ้น หรือวาดภาพของวัตถุที่มีลักษณะซับซ้อนจนทำให้เกิดมีบางส่วนของวัตถุไปบังส่วนอื่น ๆ (ถึงแม้ว่าค่า $n \cdot l > 0$ ก็ตาม) เราจึงต้องมีการพิจารณาส่วนนี้เพิ่มเติมขึ้นมา

ก่อนอื่นเราจะพิจารณาการพลอตรูปลงบนจอภาพในลักษณะของระนาบหลายๆระนาบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่เป็นรูปเหลี่ยมต่าง ๆ กัน และมีการซ้อนกัน และกับกัน และเราจะพยายามที่จะลบเส้นที่ถูกทับออกไปจากจอภาพ ตัวอย่างของระนาบรูปเหลี่ยมต่าง ๆ ดังรูป 2.16

จากนั้น เราจะทำการแก้ปัญหาตามขั้นตอนในรูป 2.17 ซึ่งจะพิจารณาวิธีการต่างๆ ดังต่อไปนี้

2.3.2.1 การทดสอบค่ามากสุด น้อยสุด (Minimax Test)

ขั้นตอนนี้จะเป็นการทดสอบว่า รูปเหลี่ยม 2 รูปนั้นมีการซ้อนกัน (Overlap) กันหรือไม่ และตามขั้นการแก้ปัญหา ถ้ารูปเหลี่ยมใดไม่มีการซ้อนกับรูปเหลี่ยมอื่นเลย รูปเหลี่ยมนั้นสามารถวาดได้ทันที

จากรูป 2.18 เราจะได้ข้อกำหนดว่า ถ้าข้อกำหนดข้อใดเป็นจริงแล้วรูปเหลี่ยมนั้นจะไม่ซ้อนกัน

$$\max\ X_A < \min\ X_B ,$$

$$\max\ X_B < \min\ X_A ,$$

$$\max\ Y_A < \min\ Y_B ,$$

$$\max\ Y_B < \min\ Y_A$$

ในกรณีที่ทดสอบแล้ว ปรากฏว่าไม่พบว่าข้อกำหนดใดไม่เป็นจริงเลยนั้น เราจะยังไม่สามารถสรุปได้ว่า รูปเหลี่ยม 2 รูปนั้นซ้อนกัน ซึ่งจะเห็นได้จากกรณีในรูป 2.19

2.3.2.2 การทดสอบจุดตัดกัน (Intersection Test)

ขั้นตอนนี้ จะพิจารณาการหาจุดตัดของเส้นตรง 2 เส้น (ซึ่งเส้นหนึ่งจะเป็นของรูปเหลี่ยมรูปหนึ่ง) เพื่อที่จะเป็นจุดตัดของระนาบที่ซ้อนกันและเป็นจุดเริ่มของเส้นที่ถูกบัง หรือเส้นที่มองเห็นบางส่วน ซึ่งการตัดกันของเส้นตรงจะมีกรณีต่าง ๆ กัน ดังนี้

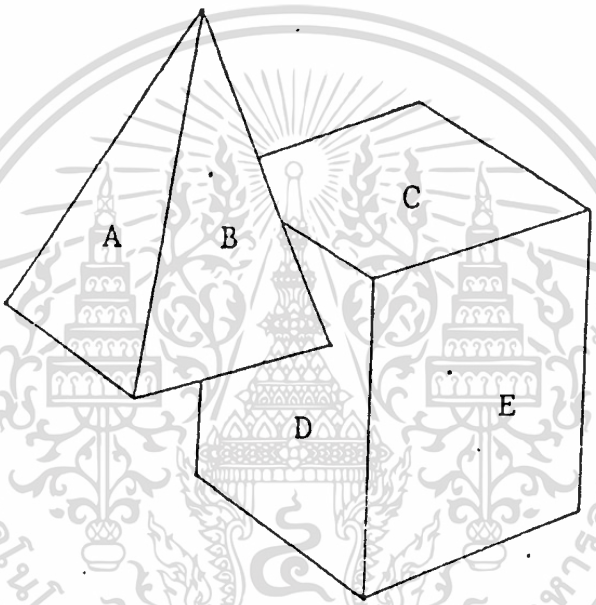
ให้ $L_1 = A_1X + B_1Y + C_1 = 0$

$L_2 = A_2X + B_2Y + C_2 = 0$

เราพิจารณาได้ 3 กรณี คือ

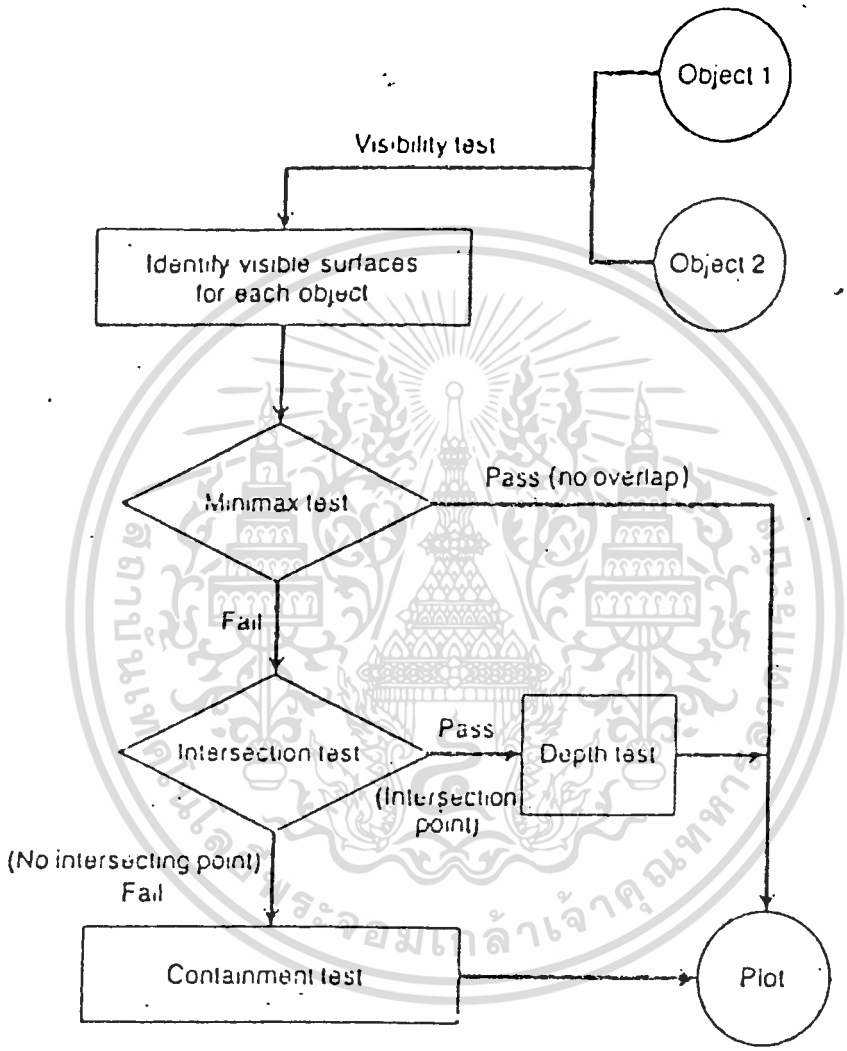
1. L_1 และ L_2 ขนานกัน ในกรณีนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



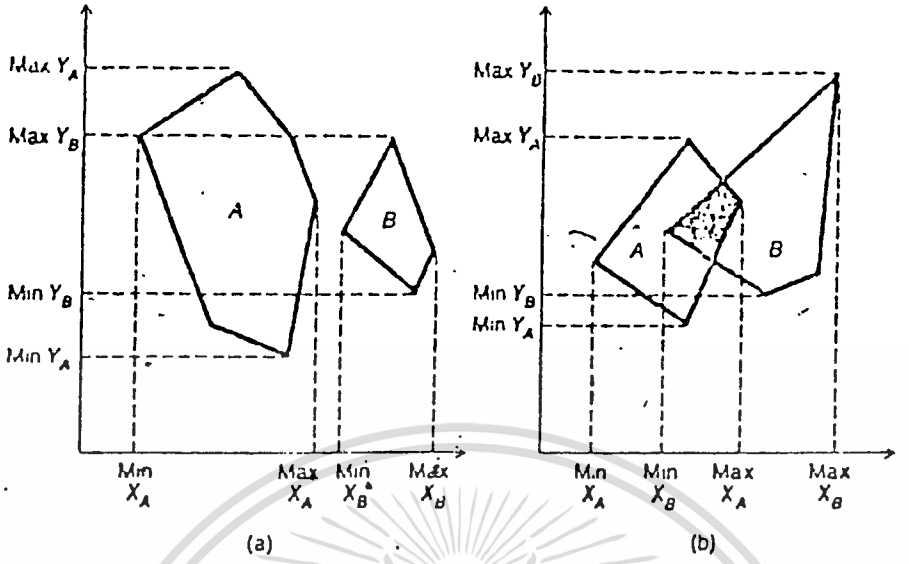
รูป 2.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.18

รูป 2.19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{vmatrix} A_1 & B_1 \\ A_2 & B_2 \end{vmatrix} = 0$$

ซึ่ง L_1 และ L_2 จะไม่ตัดกัน

2. L_1 และ L_2 ทับกัน ในกรณีนี้

$$A_1/A_2 = B_1/B_2 = C_1/C_2$$

3. นอกจากนั้นแล้ว เราจะหาจุดตัดได้โดยให้ (X_1, Y_1) เป็นจุดตัด

$$X_1 = \frac{B_1C_2 - B_2C_1}{A_1B_2 - A_2B_1}, \quad Y_1 = \frac{C_1A_2 - C_2A_2}{A_1B_2 - A_2B_1}$$

แต่ก่อนที่จะหาจุดตัดนั้น เราควรจะพิจารณาว่า เส้นตรงนั้นจะตัดกันหรือไม่ (นอกเหนือจากกรณีขนาน หรือทับกัน) เนื่องจากเส้นตรงทั้งสองแต่ละเส้น ถูกกำหนดโดยใช้จุดปลายที่แน่นอนจึงมีโอกาสที่จุดตัดจะอยู่นอกขอบเขตของเส้น 2 เส้นนั้นดังรูป 2.20 ดังนั้นเราจะใช้วิธีการทดสอบค่ามากสุด น้อยสุด (Minimax Test) ทดสอบเสียก่อน

ในทางปฏิบัติที่กำหนดจุดปลายของเส้นตรงทั้งสองมาให้ เรามักจะหาจุดตัดโดยการหาความชัน (Slope) ของเส้นตรงทั้งสองแล้วจึงนำมาแก้สมการ ดังสมการของ (X_1, Y_1) ดังนี้

$$X_1 = \frac{(Y_1 - Y'_1) - (mX_1 - m'X'_1)}{m' - m}$$

$$Y_1 = \frac{(Y_1 - mX_1)\, m' - (Y'_1 - m'x'_1)\, m}{m' - m}$$

โดย $m = (Y_2 - Y_1) / (X_2 - X_1)$

$m' = (Y'_2 - Y'_1)/(X'_2 - X'_1)$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2.3 การทดสอบการบรรจุอยู่ภายใน (Containment Test)

ขั้นตอนนี้ เป็นการพิจารณาว่า จุดใด ๆ จะปรากฏอยู่ภายในรูปเหลี่ยมรูปหนึ่งหรือไม่ ซึ่งในกรณีที่ไม่พบว่าไม่มีการตัดกันของเส้นในการทำการทดสอบการตัดกัน (Intersection Test) แล้ว ถ้าทุกจุดของรูปเหลี่ยม A อยู่ภายในรูปเหลี่ยม B แล้ว รูปเหลี่ยม A จะอยู่ภายในรูปเหลี่ยม B ทั้งหมด (Containment) เรามีวิธีการทดสอบ 2 วิธีดังนี้

- ถ้าผลรวมของมุมที่จุดที่ทดสอบ ซึ่งมีด้านของรูปเหลี่ยมเป็นด้านตรงข้ามมุมมีค่า = 360 องศาแล้ว จุดนั้นจะอยู่ในรูปเหลี่ยม
- วิธีการนี้จะเป็นการทดสอบว่า จุดใด ๆ อยู่ทางซ้าย หรือทางขวาของเส้นตรง ถ้าเรากำหนดให้เส้นตรงในรูปเหลี่ยมนั้นมีทิศทางไปตามเข็มนาฬิกาแล้ว
- ถ้าจุดนั้นล้วนอยู่ทางขวาของด้านแต่ละด้านของรูปเหลี่ยม จุดนั้นจะอยู่ภายในรูปเหลี่ยมนั้น
- ถ้าจุดนั้นอยู่ทางซ้ายของด้านใดด้านหนึ่งของรูปเหลี่ยม อย่างน้อย 1 ด้าน จุดนั้นจะอยู่นอกรูปเหลี่ยม การหาว่าจุดใดอยู่ทางซ้ายหรือทางขวาของเส้นตรงเราทำโดยใช้วิธีการที่เรียกว่า "Half Plane Method" ซึ่งอธิบายได้ดังนี้

ให้ u เป็นเวกเตอร์ของด้านของรูปเหลี่ยม และ v เป็นเวกเตอร์ของจุดปลายเริ่มต้นของด้านนั้นกับจุดที่จะพิจารณา เราจะพิจารณา u,v ว่าเป็นเวกเตอร์ใน 3 มิติ จึงเขียน u,v ได้เป็น

$$u = (X_1, Y_1, 0), \quad v = (X_2, Y_2, 0)$$

และ $u \times v = (0, 0, Z)$

ถ้า $Z > 0$ แล้ว จุดนั้นจะอยู่ทางซ้ายของเส้นตรง

และถ้า $Z < 0$ แล้ว จุดนั้นจะอยู่ทางขวาของเส้นตรง

2.3.2.4 การทดสอบความลึก (Depth Test)

หลังจากการทดสอบจุดตัดกัน (Intersection Test) หรือทดสอบการ-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรจุอยู่ภายใน (Containment Test) แล้ว เราต้องทดสอบว่า จุดใดของระนาบหนึ่งจะบังอีกระนาบหนึ่งได้หรือไม่ ในที่นี้เราจะหาค่า Z ซึ่งเป็นความลึก ณ จุดใด ๆ ของจอจากสมการของระนาบ (Plane Equation)

ถ้าที่ระนาบใด ๆ เรามีเวคเตอร์ u และ v และมีสมการระนาบเป็น

$$AX + BY + CZ + D = 0$$ แล้ว

$$(A,B,C) = u \times v$$

และ $D = - AX_1 - BY_1 - CZ_1$ โดย (X_1, Y_1, Z_1) เป็นจุดที่ทราบก่อนแล้ว

จากค่า A,B,C,D ที่หาได้ เราจะสามารถหาค่า Z ได้ในที่สุด ในระบบโคออร์ดิเนต ของจอภาพนั้นค่า Z ที่มีค่าน้อยจะอยู่ใกล้กว่าและสามารถบังจุดที่ไกลกว่าได้

การทดสอบความลึกนี้ เราจึงใช้หาว่า จุดที่ถูกรูปเหลี่ยมนี้แครอบคลุม จะบังรูปเหลี่ยมนั้น หรือจะถูกรูปเหลี่ยมนั้นบังเสีย

2.3.3 <u>การแก้ปัญหาเรื่องการบดบังเส้นและพื้นผิววิธีอื่น ๆ</u>

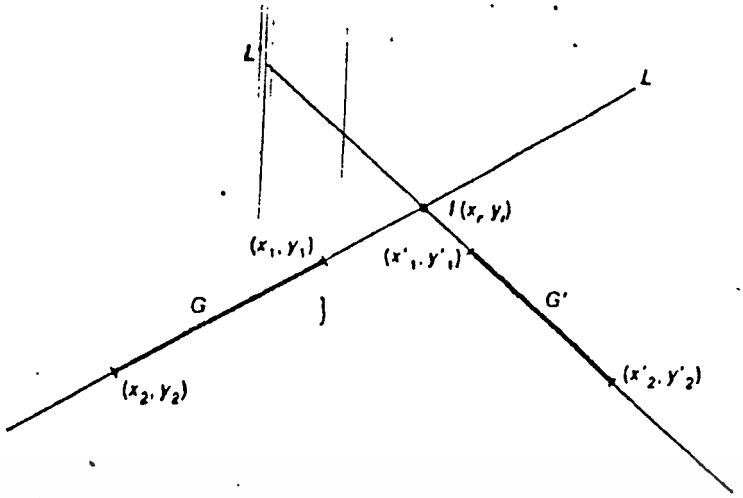
2.3.3.1 เทคนิคการมาสค์กิ้ง (Masking Technique)

วิธีการนี้ เป็นการพิจารณาค่าสูงสุด และต่ำสุดของรูปที่แต่ละจุดในแนวนอน โดยจะทำการลากเส้นตรงที่อยู่ข้างหน้าสุด (Foreground) เสียก่อน (เส้นที่อยู่ใกล้ตา หรือ Z_s มีค่าน้อยที่สุด) จากนั้นจะหาค่า Y_s min และ Y_s max ของแต่ละจุดในแนวนอนจากเส้นที่ลากลงไป เมื่อเส้นที่วาดทีหลังมีส่วนที่มีค่า Y_s อยู่ระหว่าง Y_s min และ Y_s max ของจุดนั้นแล้ว ส่วนของเส้นตรงนั้นจะไม่ถูกวาด ดังรูป 2.21

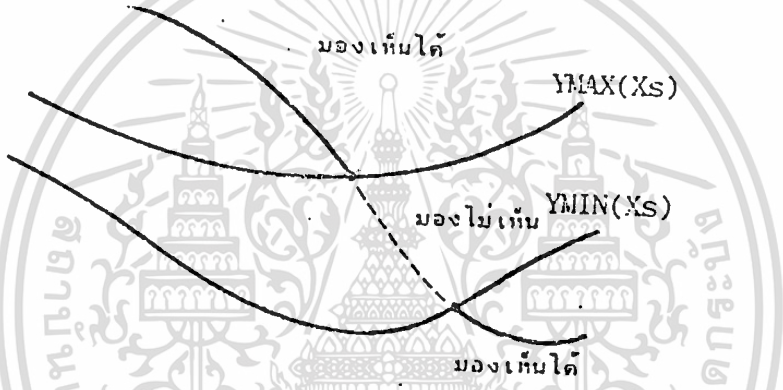
ข้อเสียของวิธีการนี้

- เราต้องเตรียมหน่วยความจำสำหรับเก็บค่า Y_s max และ Y_s min ของแต่ละจุดในแนวนอนทุกจุด เช่น จอภาพสีของ IBM Personal Computer มีจุดในแนวนอน 640 จุด จึงต้องมี Y_s max และ Y_s min อย่างละ 640 ตัว
- จำเป็นต้องมีการเรียงลำดับเส้นที่จะวาดก่อนหรือหลังซึ่งบางครั้งไม่

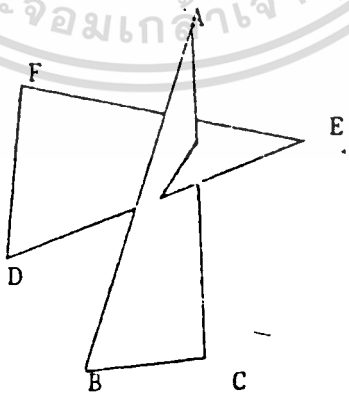
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.20



รูป 2.21



รูป 2.22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถจัดเรียงได้ เพราะอาจมีเส้นบางเส้นมีค่า Z_s ที่เหลื่อมล้ำกัน เช่น ในรูป 2.22 เส้น AB กับ DE มีค่า Z_s ที่เหลื่อมล้ำกัน

- ยุ่งยากกว่า ในการนำไปแสดงผลบนอุปกรณ์อื่น นอกเหนือจากจอภาพ เช่น พลอตเตอร์ (Plotter) หรือเครื่องพิมพ์ (Printer) เพราะเนื่องจากต้องพิจารณาการวาดทุก ๆ จุด ซึ่งต่างกับวิธีการที่แล้วคือพิจารณาเฉพาะจุดปลายทั้งสองของเส้นเท่านั้น

2.3.3.2 Image Space Algorithm

วิธีการนี้ถูกออกแบบมาให้ใช้กับจอภาพแบบราสเตอร์ (Raster Display) ซึ่งจะทำให้การพิจารณาในลักษณะของระนาบเช่นเดียวกับวิธีการทางเรขาคณิต โดยสีของแต่ละจุดบนจอภาพนั้นขึ้นกับการถูกบังของระนาบนั้น เช่น ถ้า ณ จุดนั้นเป็นจุดภายในรูปเหลี่ยมซึ่งซ้อนกันอยู่ ถ้ารูปเหลี่ยม A มีสีแดง และรูปเหลี่ยม B มีสีน้ำเงิน ณ จุดนั้น ถ้าค่าของ Z_s ในรูปเหลี่ยม A มีค่าน้อยกว่า Z_s ในรูปเหลี่ยม B จุดนั้นจะมีสีตามรูปเหลี่ยม A (Z_s อยู่ใกล้ตามากกว่า) วิธีการนี้แก้ปัญหาได้ค่อนข้างรวดเร็ว แต่ต้องใช้หน่วยความจำมาก และจะต้องมีสีประจำรูปเหลี่ยมแต่ละรูป ซึ่งจำนวนสีที่แสดงได้ในเครื่องไมโครคอมพิวเตอร์นั้นมีข้อจำกัด เช่น ใน IBM PC มีสีได้เพียง 3 สี เท่านั้น อีกประการหนึ่ง วิธีการนี้ก็ยังคงยุ่งยากในการนำไปแสดงผลลัพธ์ทางพลอตเตอร์หรือเครื่องพิมพ์ เพราะมีการพิจารณาเป็นจุด ๆ เหมือนกับเทคนิคการมาส์คกิ้ง (Masking Technique)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

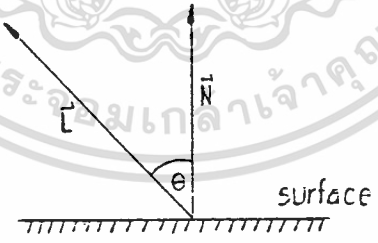
2.4 การกำหนดแสงและการให้แสงเงา (Shading)

หลังจากทำการกำจัดเส้นและพื้นผิวที่ถูกบดบังออกเรียบร้อยแล้ว ขั้นต่อไปในการสร้างภาพให้เหมือนจริง คือ การกำหนดแสง (Shading) แก่พื้นผิวที่มองเห็นซึ่งขึ้นกับแหล่งกำเนิดแสง โดยมีรายละเอียดดังนี้

การสะท้อนแสง (Reflection) ของพื้นผิวแบ่งออกเป็น 2 แบบ คือ

1. การสะท้อนแบบแพร่กระจาย (Diffuse Reflection) พื้นผิวของวัตถุทึบแสงจะแสดงคุณสมบัติที่เรียกว่า "การสะท้อนแบบแพร่กระจาย" (Diffuse Reflection) โดยแสงจะกระจายออกจากพื้นผิวเท่า ๆ กันในทุกทิศทาง ทำให้มองเห็นพื้นผิวมีความสว่างเท่ากันหมดในทุกมุมมอง

จากกฎโคซายน์ของแลมเบอร์ธ (Lambert's law) แสดงให้เห็นถึงความสัมพันธ์ระหว่างปริมาณแสงที่สะท้อนออกมากับโคซายน์ (Cosine) ของมุม θ ซึ่งเป็นมุมระหว่างทิศทาง L ชี้ไปยังแหล่งกำเนิดแสงแบบจุด (Point Source) ของความเข้ม I_p และ N ซึ่งเป็นเวคเตอร์ตั้งฉากกับพื้นผิว ดังแสดงดังรูป 2.23



รูป 2.23

จากรูป จะเห็นได้ว่าแสงที่ผู้มองสังเกตเห็นสะท้อนออกมานั้น ไม่ขึ้นอยู่กับตำแหน่งของผู้มอง

$$\text{Diffuse Illumination} = I_d = I_p k_d \cos\theta \quad \ldots\ldots(1)$$

สัมประสิทธิ์ k_d เป็นค่าคงที่อยู่ระหว่าง 0 กับ 1 และเปลี่ยนไปตามวัตถุต่าง ๆ สมมุติให้เวคเตอร์ $\vec{L}$ และ $\vec{N}$ ถูกนอร์มอลไลซ์ (Normalized) ดังนั้นเขียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมการ (1) ใหม่ได้โดยใช้ผลคูณเชิงจุด (Dot Product) ดังนี้

$$I_d = I_p k_d (\vec{L} \cdot \vec{N})$$

แต่ในสภาพแวดล้อมจริง จะมีแสงจากสภาพแวดล้อมประกอบด้วย ซึ่งเป็นแสงที่เกิดจากการสะท้อนมากมายของวัตถุหลายอย่าง ดังนั้น กฎการให้แสง (Shading Rule) จะเป็น

$$I = I_a k_a + I_p k_d (\vec{L} \cdot \vec{N})$$

I_a เป็นแสงจากสภาพแวดล้อม

k_a เป็นค่าคงที่ที่แสดงให้เห็นว่า สภาพแวดล้อมนั้นสะท้อนจากผิววัตถุมากน้อยเพียงใด

เนื่องจากแสงที่กระทบวัตถุมีลักษณะขนาน ดังนั้น ถ้าพื้นผิวมีการเกยกันอยู่จะทำให้ได้รับแสงเท่ากัน ไม่สามารถแยกได้ ดังนั้นจึงต้องใช้คุณสมบัติที่ว่า "พลังงานของแสงจะแปรผกผันกับกำลังสองของระยะทาง (R) จะได้

$$I = I_a k_a + I_p k_d (\vec{L} \cdot \vec{N})/R^2$$

แต่ทางปฏิบัติ แสงที่มาจากระยะทาง R จะมีค่าเป็นอนันต์ด้วย (เป็นการโปรเจคแบบขนาน) และสำหรับการโปรเจคแบบเปอร์สเปคตีฟ ค่า $1/R^2$ จะมีช่วงค่ากว้าง เพราะจุดมองมีค่าใกล้กับวัตถุ ดังนั้น เพื่อให้ใกล้เคียงความจริง จึงแทนค่า R^2 เป็น r+k เมื่อ

k = ค่าคงที่

r = ระยะทางจากจุดมองแบบเปอร์สเปคตีฟไปยังพื้นผิว

$$I = I_a k_a + I_p k_d (\vec{L} \cdot \vec{N})/(r+k)$$

สำหรับพื้นผิวที่เป็นสีจะหาได้จากสมการข้างต้น แยกเป็นสำหรับแสงสีน้ำเงิน, เขียวและเหลือง ด้วยค่าคงที่การสะท้อน (k_{dc}, k_{dm}, k_{dy}) ในแต่ละสี และกฎการหักล้างจะนำมาใช้ด้วย เนื่องจากการสะท้อนของแสงเป็นลักษณะของการหักล้างกัน การหักล้างกันของแสงสีทั้งสามก็จะถูกสะท้อนออกมาอยู่กับค่า k_{dc}, k_{dm} และ k_{dy} ตัวอย่างของแสงสีน้ำเงินเขียนเป็นสมการดังนี้

$$I_c = I_{mc} k_{mc} + I_{pc} k_{dc} (\vec{L} \cdot \vec{N})/(r+k)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับ I_m และ I_y ก็สามารถเขียนเป็นสมการได้ในทำนองเดียวกัน

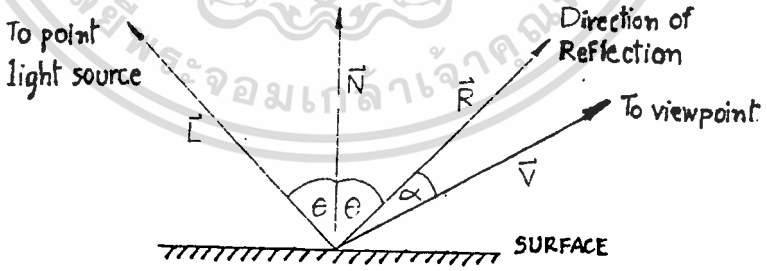
2. การสะท้อนแบบสเปคคูลาร์ (Specular Reflection) การสะท้อนแบบนี้ จะสังเกตได้บนพื้นผิวที่เป็นมันวาว เช่นการส่องแสงลงบนลูกแอปเปิ้ล จะเห็นแสงสว่างซึ่งเกิดจากการสะท้อนแบบสเปคคูลาร์ในขณะที่แสงสะท้อนมาจากส่วนอื่น ๆ เป็นการสะท้อนแบบแพร่กระจาย

ลักษณะที่แตกต่างจากการสะท้อนแบบแพร่กระจาย คือ เมื่อเปรียบจุดมองจะทำให้เห็นแสงสว่างนั้นเปลี่ยนแปลงไปด้วย เพราะผิวมันสะท้อนแสงได้ในทิศทางต่าง ๆ กัน ถ้าผิวสะท้อนนั้นเป็นแบบสะท้อนสมบูรณ์ เช่น กระจก แสงจะสะท้อนไปในทิศทางเดียวกัน จากรูป 2.24 จะต้องมีค่าเป็น 0 เท่านั้น แต่ถ้าผิวสะท้อนไม่สมบูรณ์ แสงสะท้อนจะลดลงอย่างรวดเร็วเมื่อมุม เพิ่มขึ้น

สำหรับวัตถุจริง ๆ นั้น ปริมาณแสงที่ตกกระจายแบบสเปคคูลาร์ จะขึ้นอยู่กับมุมตกกระทบ θ ถ้าให้ $W(\theta)$ คือ Specular Reflected Light จะได้

$$I = I_a k_a + I_p [k_d \cos\theta + W(\theta) \cos^n\alpha]/(r+k)$$

n ขึ้นอยู่กับพื้นผิว (ผิวสะท้อนสมบูรณ์ n จะเป็นอนันต์)



รูป 2.24

เมื่อนอร์มอลไลซ์ (normalized) $\vec{R}$ และ $\vec{V}$ แล้ว $\cos\alpha = \vec{R} \cdot \vec{V}$ ให้ $W(\theta)$ เป็นค่าคงที่ k_s จะได้สมการดังนี้

$$I = I_a k_a + I_p [k_d (\vec{L}\cdot\vec{N}) + k_s (\vec{R}\cdot\vec{V})^n]/(r+k)$$

เมื่อพิจารณาพื้นผิวที่เป็นสี เช่น สีน้ำเงิน (cyan) จะได้

$$I_c = I_{ac} k_{ac} + I_{pc} [k_{dc} (\vec{L}\cdot\vec{N}) + k_s (\vec{R}\cdot\vec{V})^n]/(r+k)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นว่า k_s ไม่ขึ้นกับสีของพื้นผิว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

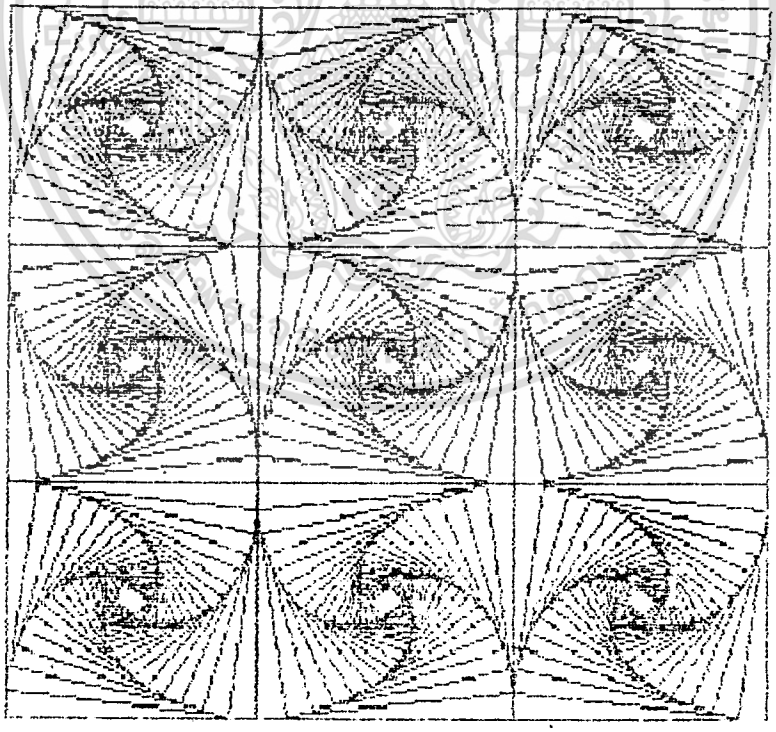
เนื้อหาเกี่ยวกับการทดลองและประยุกต์ใช้ทฤษฎีต่าง ๆ

3.1 ขั้นตอนและวิธีการในการพัฒนาโปรแกรม

ภาคการศึกษา 1 ทำการศึกษาเทคนิคต่าง ๆ ดังนี้

1. ศึกษาเทคนิคการสร้างแบบจำลองทางกราฟฟิกทั้ง 2 มิติและ 3 มิติ
2. ศึกษาเทคนิคการแสดงแบบจำลองที่มีหลายวัตถุซ้อนกัน
3. รวบรวมเทคนิคต่าง ๆ ที่ศึกษามาสร้างเป็นแบบจำลองกราฟฟิกที่ซับซ้อน

1. เมื่อศึกษาถึงเทคนิคการสร้างแบบจำลองทางกราฟฟิก 2 มิติได้ทำการสร้าง Nest Square ซึ่งมีลักษณะดังรูป 3.1

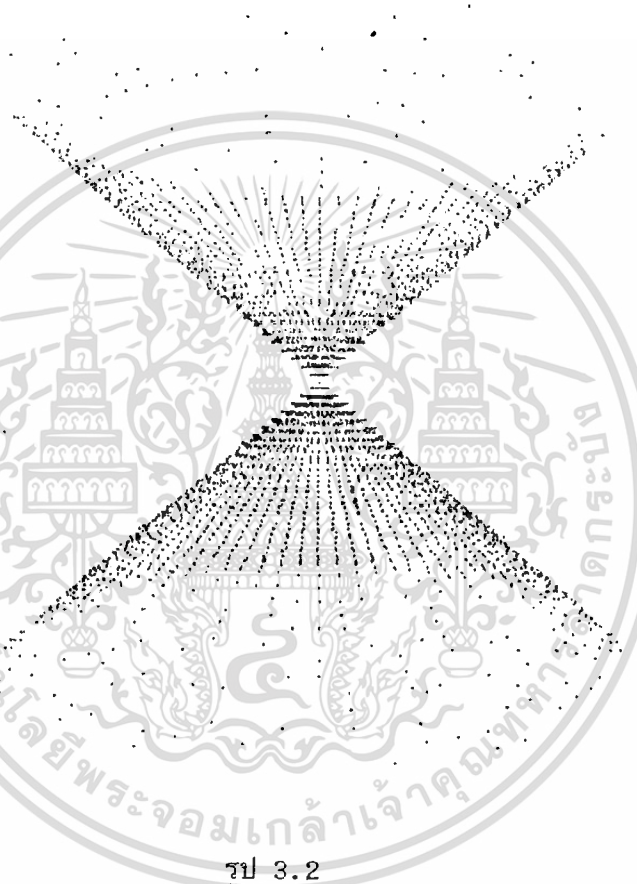


รูป 3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

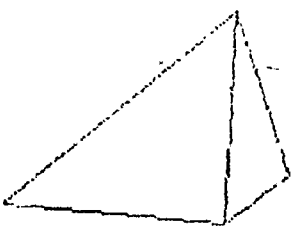
2. เมื่อศึกษาถึงเทคนิคการสร้างแบบจำลองทางกราฟฟิก 3 มิติได้ทำการสร้างภาพ

- วัตถุเดียวที่ไม่กำจัดเส้นและพื้นผิวที่ถูกบัง ซึ่งมีลักษณะดังรูป 3.2



รูป 3.2

- วัตถุเดียวที่กำจัดเส้นและพื้นผิวที่ถูกบังออก ซึ่งมีลักษณะดังรูป 3.3



รูป 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- วัตถุ 2 จัดถุที่กำจัดเส้นและพื้นผิวที่ถูกบังออก ซึ่งมีลักษณะดังรูป 3.4

รูป 3.4

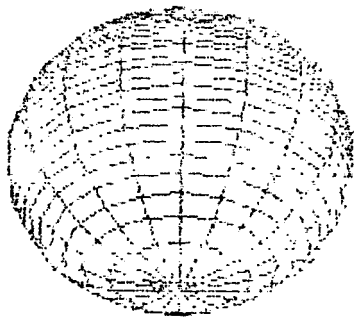
ภาคการศึกษา 2 ทำการศึกษาและสร้างแบบจำลองดังนี้

1. ทำการสร้างแบบจำลองใหม่เป็นทรงกลม ซึ่งแตกต่างจากภาคการศึกษา 1 โดยแบบจำลองนี้จะประกอบไปด้วยพื้นผิวที่ประกอบกันขึ้นมาเป็นทรงกลมจำนวนมาก เพื่อรองรับการการสร้างแบบจำลองที่เหมือนจริง กล่าวคือ ทำให้พื้นผิวแต่ละพื้นผิวมีความเข้มของแสงต่าง ๆ กัน เมื่อมีแหล่งกำเนิดแสงส่องมา

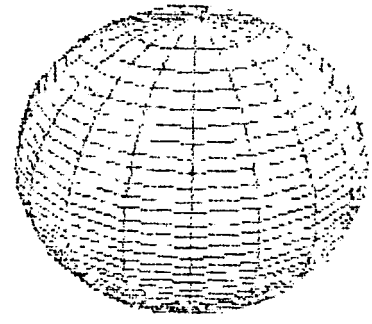
2. ทำการศึกษาการกำหนดความเข้มแสง ให้แต่ละพื้นผิว โดยสามารถกำหนดแหล่งกำเนิดแสงและมุมมองได้ตามต้องการ ในระบบพิกัดทรงกลม แสดงดังรูป 3.5

$\theta = 0, \quad \phi = 135^{\circ}$

$\theta = 0, \quad \phi = 56^{\circ}$



Hidden $\theta = 0, \quad \phi = 135^{\circ}$

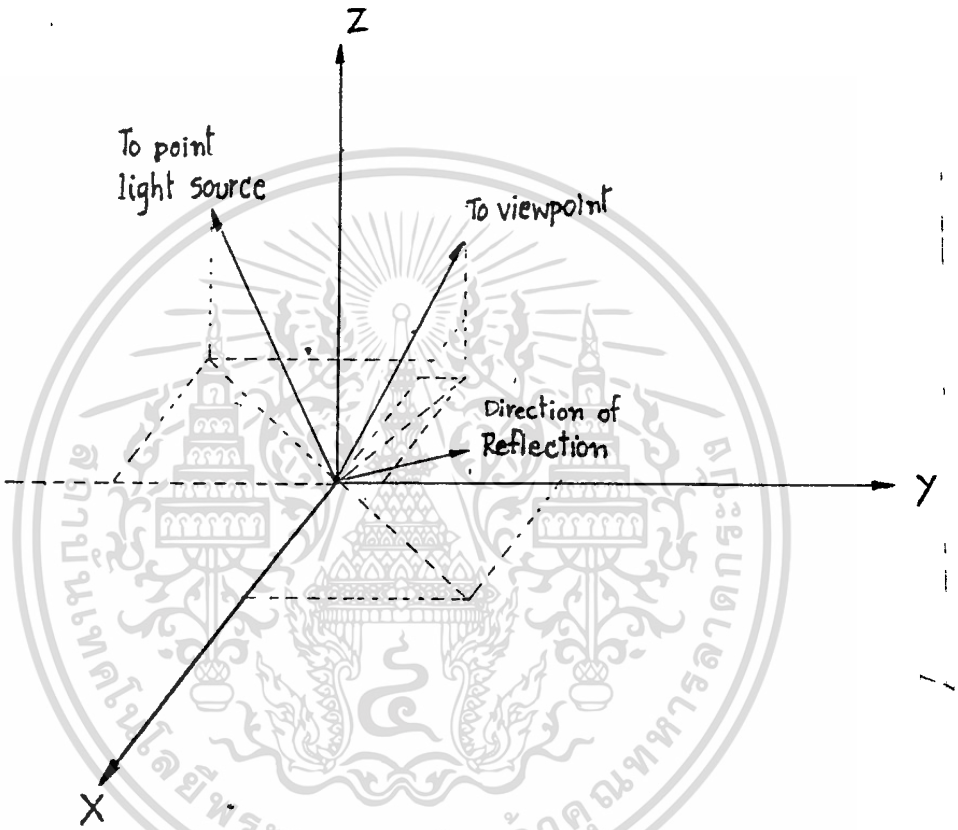


Hidden $\theta = 0, \quad \phi = 56^{\circ}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ผลการพัฒนาโปรแกรม

หลักการที่ใช้ในการกำหนดความเข้มของแสงสำหรับแต่ละพื้นผิว (Surface) ที่ประกอบกันขึ้นมาเป็นรูปทรงกลม แสดงดังรูป 3.6



รูป 3.6

เมื่อมีการกำหนดแหล่งกำเนิดแสง (Light Source) ให้อยู่ในตำแหน่งที่ปรากฏดังรูปแล้ว เราจะถือว่าจุดที่เป็นจุดกำเนิด (Origin) เป็นจุดที่แทนพื้นผิวนั้น ดังนั้นความเข้มที่ได้จากจุดนี้ ก็คือ ความเข้มของพื้นผิวนั่นเอง

ขั้นแรก ทำการสะท้อนเวคเตอร์ที่มาจากแหล่งกำเนิดแสง โดยการสะท้อนจะสะท้อนกับแกน Z

ขั้นที่สอง ทำการหาเวคเตอร์ซึ่งลากจากจุดมองใด ๆ (Viewpoint)

ขั้นที่สาม ทำการโปรเจค (Project) เวคเตอร์ที่ลากจากจุดมองนั้นลงมายังเวคเตอร์ที่สะท้อนมาจากแหล่งกำเนิดแสง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

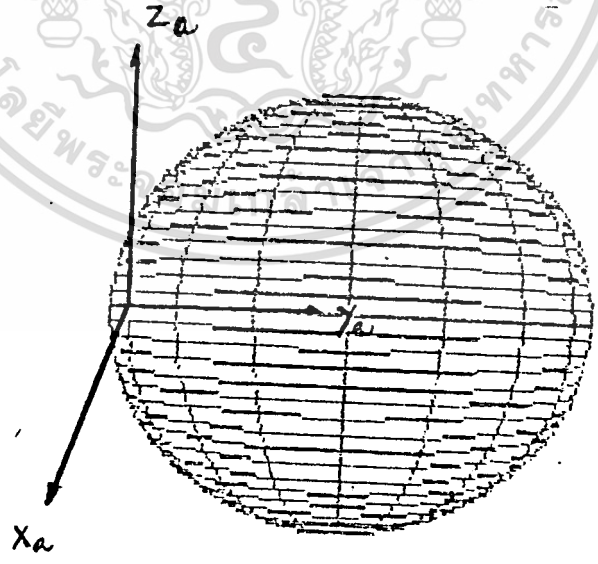
ขั้นที่สี่ กำหนดความเข้มของแสงได้จากขนาดของเวคเตอร์ที่ถูกโปรเจกลงมาแล้ว ซึ่งจะได้ว่า ขนาดเวคเตอร์ยาวกว่าจะมีความเข้มของแสง หรือความสว่างมากกว่า

ขั้นที่ห้า ทำเช่นเดียวกับข้อ 1 ถึงข้อ 4 กับทุก ๆ พื้นผิว จะทำให้สามารถกำหนดความเข้มของทุก ๆ พื้นผิว

อย่างไรก็ตาม ขั้นตอนในการกำหนดความเข้มของแสง 5 ขั้นตอน ดังที่กล่าวมานี้เป็นวิธีการใหญ่ ๆ ที่ใช้ในการพิจารณา ซึ่งในการเขียนโปรแกรมเพื่อนำภาพมาปรากฏบนจอภาพนั้นมีวิธีการที่ละเอียดอ่อนอยู่พอสมควร โดยจะกล่าวลักษณะการทำงานของรูทีน (Routine) ต่าง ๆ ในส่วนของการกำหนดความเข้มแสง (Shading) ดังนี้

1. New_System (XXX,YYY,ZZZ:Real); รูทีนนี้จะทำหน้าที่ดังนี้

1.1 ทำการย้ายระบบโคออร์ดิเนตจากตรงกลางภายในทรงกลมมาอยู่ที่จุดใด ๆ ซึ่งจุดใด ๆ นี้ก็คือ จุดที่ 1 (endpoint ที่ 1) ของแต่ละพื้นผิว (ที่ถูกกำหนดมาตั้งแต่เริ่มสร้างแบบจำลองทรงกลมนั้นแล้ว) ดังรูป 3.7



รูป 3.7

จะได้เมตริกซ์ในการแปลงดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$T_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -point[i,1] & -point[i,2] & -point[i,3] & 1 \end{bmatrix}$$

1.2 หลังจากทำตามข้อ 1.1 แล้ว โคออร์ดิเนตที่ได้จะไม่อยู่ในตำแหน่งที่ควรเป็น กล่าวคือ โคออร์ดิเนตที่ได้จะมีแกนที่ขนานกับแกนเดิมทั้งหมด ดังนั้น จึงต้องทำการหมุนระนาบ XY เมื่อมองจากแกน Z ในทิศทางทวนเข็มนาฬิกา เมื่อเริ่มจากพื้นผิวที่ 1 จะได้ว่าจะเริ่มโดยหมุนไป -90 องศาก่อน และค่อย ๆ เพิ่มขึ้นครั้งละ 18 องศาต่อ 1 พื้นผิว (ถูกกำหนดไว้ตั้งแต่เริ่มสร้างแบบจำลองทรงกลมแล้ว) ดังนั้นจะได้ว่า เมื่อหมุนไปครบรอบจะได้ความแตกต่างของรูปแบบการหมุน 20 รูปแบบ หรือ 20 พื้นผิวนั่นเอง

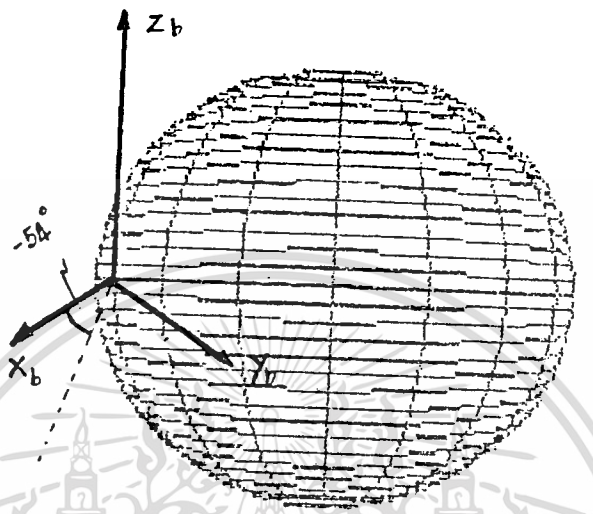
จุดประสงค์ของการทำขั้นตอนนี้ ก็เพื่อที่จะทำให้แกน X อยู่ในแนวเดียวกับเส้นลองกิจูดนั่นเอง

เมื่อทำการกำหนดพารามิเตอร์ XXX, YYY, ZZZ โดยการผ่านค่ามายังรูทีนนี้แล้ว ผลที่ได้ก็คือ จุด (XXX, YYY, ZZZ) นี้จะถูกมองโดยเปรียบเทียบกับระบบใหม่นั่นเอง ซึ่งเราสามารถกำหนดแหล่งกำเนิดแสง โดยการใส่ตำแหน่งของแหล่งกำเนิดแสง (light X, light Y, light Z) แสดงดังรูป 3.8

จะได้เมตริกซ์ในการแปลงดังนี้

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



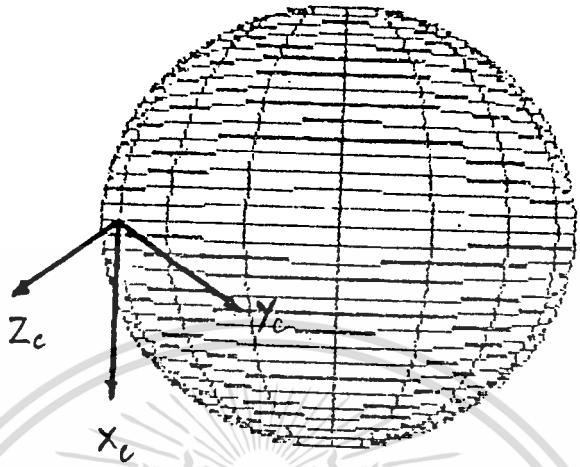
รูป 3.8

1.3 ทำการหมุนระนาบ XZ เมื่อมองจากแกน Y จะได้ว่า หมุนทวนเข็มนาฬิกาเป็นมุมต่าง ๆ เริ่มจากพื้นผิวที่ 1 ซึ่งอยู่ด้านล่างสุด ซึ่งพิจารณาแล้วต้องหมุนเป็นมุม 180 องศาเป็นจำนวน 20 ครั้ง นั่นก็คือ ในหนึ่งแถวจะประกอบไปด้วยพื้นผิว 20 พื้นผิว เมื่อขึ้นพื้นผิวที่ 21 มุมที่จะต้องหมุนทวนเข็มนี้จะลดลง 6 องศา (ถูกกำหนดไว้ตั้งแต่เริ่มสร้างแบบจำลองทรงกลมแล้ว) นั่นคือ จะต้องหมุนทวนเข็มนาฬิกา 174 องศา ทำเช่นนี้เรื่อย ๆ ไปเมื่อจะเริ่มต้นพื้นผิวที่ 41,61,...,581 ดังรูป 3.9

จะได้เมตริกซ์ในการแปลงดังนี้

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.9

จุดประสงค์ของขั้นตอนนี้ ก็เพื่อจะทำให้แกน Z เป็นแกนที่แทนเวคเตอร์ปกติ (Normal Vector) ซึ่งจะต้องตั้งฉากกับพื้นผิวทุก ๆ พื้นผิว และมีทิศทางชี้ออกจากพื้นผิวนั้น

2. Reflect_Z_Axis เป็นรูทีนที่ทำหน้าที่ในการสะท้อนเวคเตอร์ที่ลากจากแหล่งกำเนิดแสง ซึ่งแหล่งกำเนิดแสงขณะนี้เป็นแหล่งกำเนิดแสงที่เทียบกับระบบใหม่แล้ว (T_1, T_2, T_3)

3. Vector_Reflect เป็นรูทีนที่กำหนดเวคเตอร์สะท้อน โดยลากจากจุดกำเนิดของระบบใหม่ไปยังจุดซึ่งได้จากข้อ 2.

4. Viewpoint เป็นรูทีนที่ทำหน้าที่ในการแปลงจุดมองจากระบบพิกัดทรงกลม (ซึ่งถูกกำหนดมา) ให้เป็นจุดในระบบพิกัดฉาก

5. Find_Angle เป็นรูทีนที่ทำหน้าที่ในการหามุมที่จะต้องทำการหมุนแกน Y_d ให้ไปซ้อนทับกับเวคเตอร์ที่สะท้อนมาจากแหล่งกำเนิดแสง ซึ่งมุมที่กล่าวถึงนี้มีอยู่ 2 มุม

<u>มุมที่ 1</u> คือ มุม ϕ_1 ในรูป 3.10 กล่าวคือ การหมุนระนาบ Y_dZ_d เมื่อมองจากแกน X_d ในทิศทางทวนเข็มนาฬิกาเป็นมุม ϕ_1 ซึ่งหาได้จาก

ϕ_1 = arctan (ค่าของ z ของเวคเตอร์นั้น / ขนาดของเวคเตอร์ที่โปรเจคลงบนระนาบ XY) เมื่อค่าของ y ของเวคเตอร์สะท้อนนั้นมากกว่า 0

ϕ_1 = Pi - arctan (ค่าของ z ของเวคเตอร์นั้น / ขนาดของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เวคเตอร์ที่โปรเจคลงบนระนาบ XY) เมื่อค่าของ y ของเวคเตอร์สะท้อน นั้นน้อยกว่า 0

$\phi_1 = 0$ เมื่อเวคเตอร์สะท้อนนั้นมาซ้อนกับแกน Y_d ทางด้านบวกของดี

$\phi_1 = Pi / 2$ เมื่อเวคเตอร์สะท้อนนั้นอยู่บนระนาบ Z_d

และ $\phi_1 = Pi$ เมื่อเวคเตอร์สะท้อนนั้นมาซ้อนกับแกน Y_d ทางด้านลบของดี

เมื่อทำการหมุนระนาบ Y_dZ_d จากแกน X_d ในทิศทางทวนเข็มนาฬิกาเป็นมุม ϕ_1 จะได้ระบบใหม่ดังรูป 3.11 และจะได้เมตริกซ์ในการแปลงดังนี้

$$T_A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\phi_1 & -\sin\phi_1 & 0 \\ 0 & \sin\phi_1 & \cos\phi_1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

<u>มุมที่ 2</u> คือ มุม ϕ_2 ในรูป 3.12 กล่าวคือ การหมุนระนาบ X_eY_e เมื่อมองจากแกน Z_e ตามเข็มนาฬิกาเป็นมุม ϕ_2 ซึ่งหาได้จาก

ϕ_2 = arctan (ค่าของ x / ค่าของ y) เมื่อค่าของ y ของเวคเตอร์ที่สะท้อนนั้นมากกว่า 0

ϕ_2 = -arctan (ค่าของ x / ค่าของ y) เมื่อขนาดของ y ของเวคเตอร์ที่สะท้อนนั้นน้อยกว่า 0

$\phi_2 = 0$ เมื่อเวคเตอร์สะท้อนนั้นอยู่บนระนาบ Y_eZ_e

$\phi_2 = Pi / 2$ เมื่อเวคเตอร์สะท้อนนั้นซ้อนกับแกน X_e ด้านบวกของดี

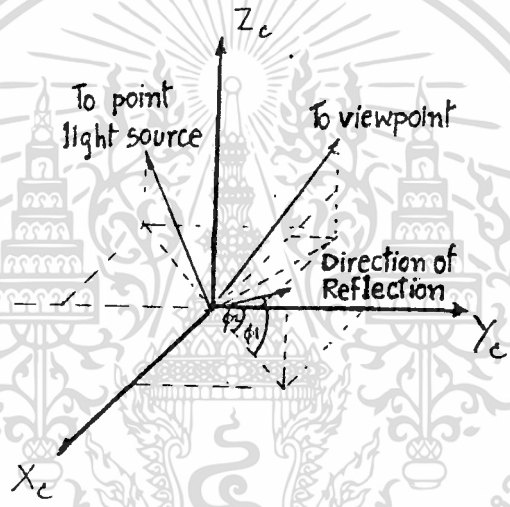
และ $\phi_2 = -Pi / 2$ เมื่อเวคเตอร์สะท้อนนั้นซ้อนกับแกน X_e ด้านลบของดี

เมื่อทำการหมุนระนาบ X_eY_e ตามเข็มนาฬิกา เมื่อมองจากแกน Z_e ในทิศทางตามเข็มนาฬิกาแล้ว จะได้ระบบใหม่ ดังรูป 3.12

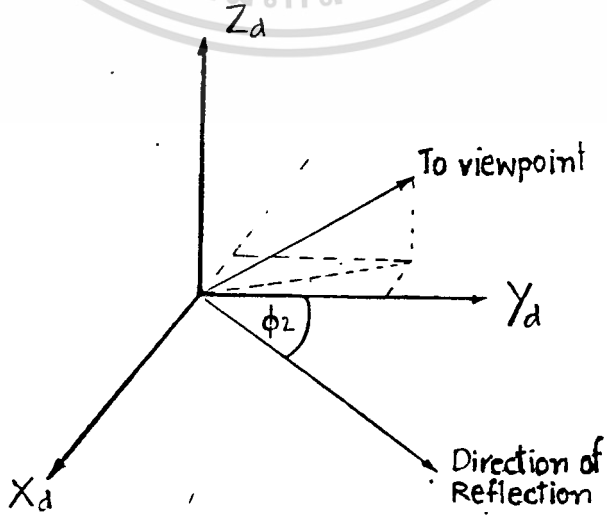
จะได้เมตริกซ์ในการแปลงดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$T_5 = \begin{bmatrix} \cos \phi_2 & \sin \phi_2 & 0 & 0 \\ -\sin \phi_2 & \cos \phi_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

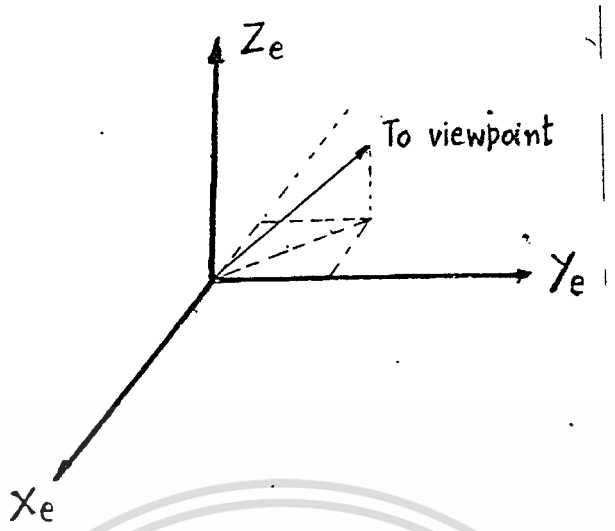


รูป 3.10



รูป 3.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.12

6. Viewpoint_Rotate_X_and_Z_New (View_X, View_Y, View_Z)

รูทีนนี้จะทำหน้าที่ในการเปลี่ยนการอ้างอิงของจุดมองให้อยู่ในระบบใหม่ล่าสุด (X_f, Y_f, Z_f) ซึ่งจะต้องผ่านขั้นตอนหลาย ๆ ขั้นตอน ดังนี้

6.1 ย้ายระบบโคออร์ดิเนตแรกเริ่มมาอยู่ที่จุดมุม (endpoint) ที่ 1 ของพื้นผิวนั้น (พื้นผิวที่ต้องการจะหาความเข้ม) ดังรูป 3.7 (T_1)

6.2 หมุนระนาบต่าง ๆ เพื่อให้อยู่ในรูปที่เหมาะสม ดังแสดงให้เห็นจากรูทีน New_System แล้ว ดังรูป 3.8 และ 3.9 (T_2 และ T_3)

6.3 เมื่อทราบค่าของมุม ϕ_1 และ ϕ_2 ดังรูป 3.10 แล้วจะทำการหมุนระนาบ Y_dZ_d เมื่อมองจากแกน X_d ในทิศทางทวนเข็มนาฬิกาเป็นมุม ϕ_1 และทำการหมุนระนาบ X_dY_d เมื่อมองจากแกน Z_d ในทิศทางตามเข็มนาฬิกาเป็นมุม ϕ_2 ดังรูป 3.11 ซึ่ง ϕ_1 และ ϕ_2 ได้มาจากรูทีน Find_Angle นั่นเอง

เมื่อสิ้นสุดขบวนการในรูทีนนี้จะได้ว่า จุดมองซึ่งแต่เดิมเคยถูกมองโดยอ้างอิงกับระบบแกนของวัตถุเดิม จะถูกมองโดยอ้างอิงกับระบบใหม่ล่าสุด (X_f, Y_f, Z_f) นั่นเอง ดังรูป 3.12

7. Projection_and_Adjust รูทีนนี้จะทำการโปรเจคขนาดของเวคเตอร์ที่ลากจากจุดมอง โดยโปรเจคมายังแกน Y แล้วนำมาเทียบกับขนาดของเวคเตอร์ที่ลากจากจุดมองนั้น จากนั้นจะนำมาปรับค่า (adjust) ให้อยู่ในระดับ (level) 0 - 255 ซึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นระดับที่สามารถนำไปปรากฏบนจอภาพ (Image Display) ได้โดยการอินเตอร์เฟสกับ DigiImage Card ซึ่งจะให้ความเข้มที่มีระดับต่าง ๆ กันถึง 256 ระดับต่อ 1 จุด และ 1 จอภาพ จะประกอบไปด้วย 256 X 256 จุด

8. Mapping_to_Image1 รูทีนนี้จะทำการกำหนดระดับความเข้มของแสงที่ตำแหน่งต่าง ๆ บนจอภาพ โดยที่ขณะนี้เราทราบแล้วว่า แต่ละพื้นผิวมีความเข้มเท่าใด (จากขั้นตอนที่ 1 - 7) วิธีการทำจะเป็นไปตามขั้นตอนต่อไปนี้

8.1 ทำการปรับอัตราส่วนของจอภาพ (monochrome) กับจอภาพ (image) ให้พอดีกัน โดยในที่นี้จะได้ค่าในแนวแกน X ของจอภาพ มีอัตราส่วนเป็น 256 / 720 และ ค่าในแนวแกน Y ของจอภาพจะมีอัตราส่วนเป็น 256 / 348

หมายเหตุ จอภาพโมโนโครมมีขนาด 720 x 348 จุด

จอภาพอิมเมจ (Image) 256 x 256 จุด

8.2 กำหนดกรอบเพื่อที่จะทำการตรวจสอบว่า แต่ละจุดภายในกรอบนั้นอยู่ภายในพื้นผิวใด หรือไม่อยู่ภายในทรงกลมเลย โดยการหาค่า X min, Y min, X max, Y max ของรูปทรงกลมนี้ (ในระบบแกนของจอภาพที่ถูกดัดแปลงด้วย SCF และ Tx, Ty เรียบร้อยแล้ว)

8.3 ทำการกำหนดค่าเริ่มต้นต่าง ๆ ในอาร์เรย์ (Array) ชื่อ TonePtr^[j,k] ด้วยค่าความสว่างสูงสุด (255)

8.4 ทำการตรวจสอบแต่ละจุด โดยเริ่มจากจุด (X min_block, Y min_block) ของกรอบที่สร้างขึ้น จนถึงจุด (X max_block,Y max_block) ของกรอบ ขณะนี้จะได้ความเข้มที่พื้นผิว 1 พื้นผิวแล้ว ซึ่งปรากฏอยู่ในรูปของค่าที่อยู่ในอาร์เรย์ชื่อ TonePtr^[j,k] แล้ว

9. Mapping_to_Image2 รูทีนนี้จะทำการส่งค่าที่อยู่ในอาร์เรย์ชื่อ TonePtr^[j,k] ออกไปยังหน่วยความจำจอภาพของ DigiImage Card โดยมีขั้นตอนดังนี้

9.1 ทำการติดต่อกับ DigiImage Card โดยการส่งค่า 01H ออกทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ต 300H พร้อมทั้งดับจอภาพ

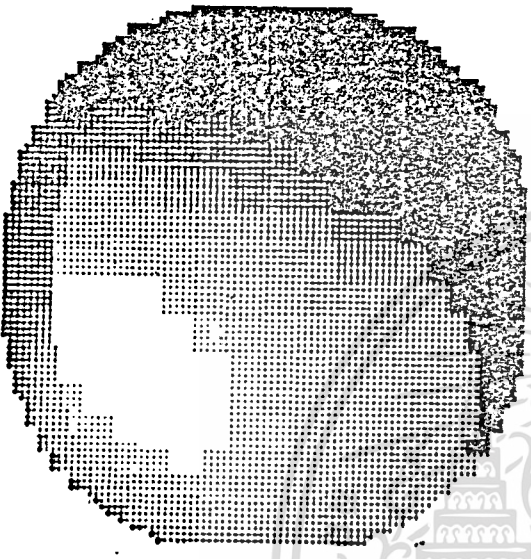
9.2 ทำการลบจอภาพหรือทำให้จอภาพมืดสนิทโดยการส่งค่า 00H ออกไปยังหน่วยความจำที่เริ่มต้นด้วย 0d000H : 0000H โดยตำแหน่ง (Adress) ต่าง ๆ ของหน่วยความจำจอภาพนี้ จะอยู่ในรูปของอาร์เรย์ชื่อ Image Screen [row,column]

9.3 ทำการส่งค่าทั้งหมดที่อยู่ในอาร์เรย์ TonePtr^[j,k] ไปไว้ที่อาร์เรย์ Image_Screen [row,column]

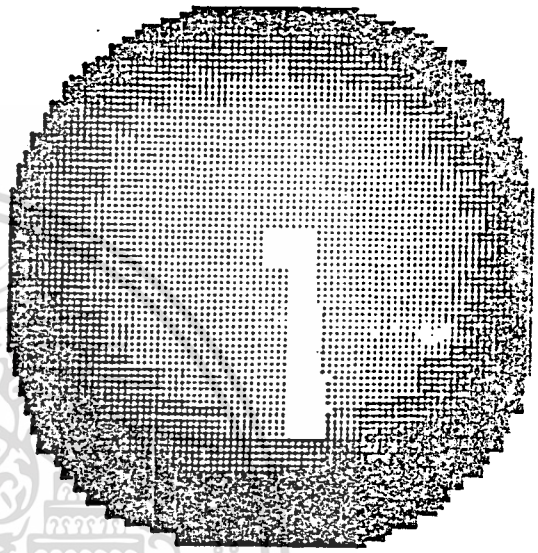
9.4 แสดงภาพด้วยการส่งค่า 08H ออกไปทางพอร์ต 0300H

จากผลการทดลองรันโปรแกรม จะได้แบบจำลองที่มีสามารถกำหนดแหล่งกำเนิดแสงและตำแหน่งของมุมมอง ได้ดังรูป 3.13

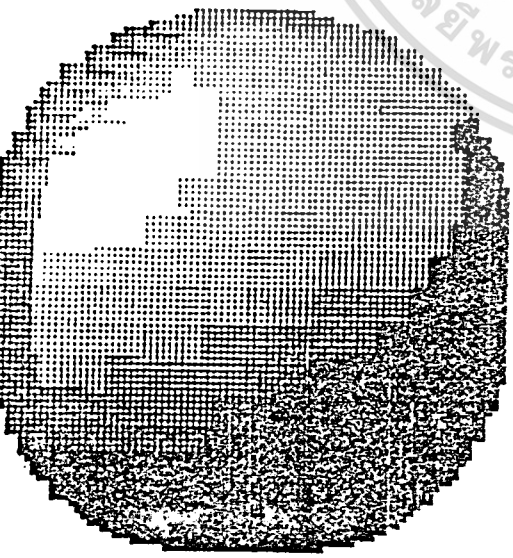
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



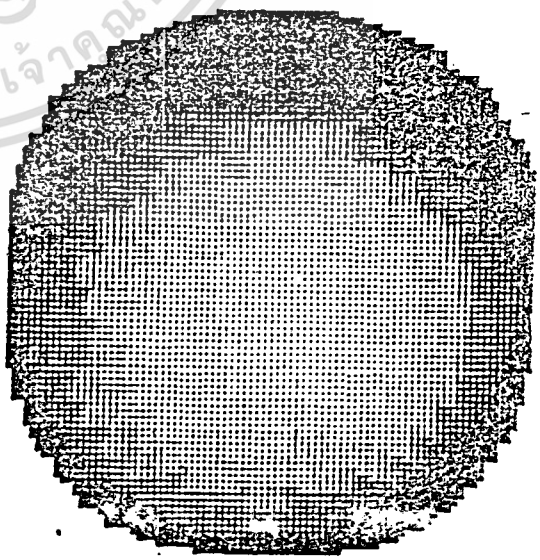
θ_VIEWPOINT = 0° , Ø_VIEWPOINT = 90°
θ_LIGHTSOURCE = -45°, Ø_LIGHTSOURCE = 135°



θ_VIEWPOINT = 56° , Ø_VIEWPOINT = 56°
θ_LIGHTSOURCE = 60°, Ø_LIGHTSOURCE = 50°

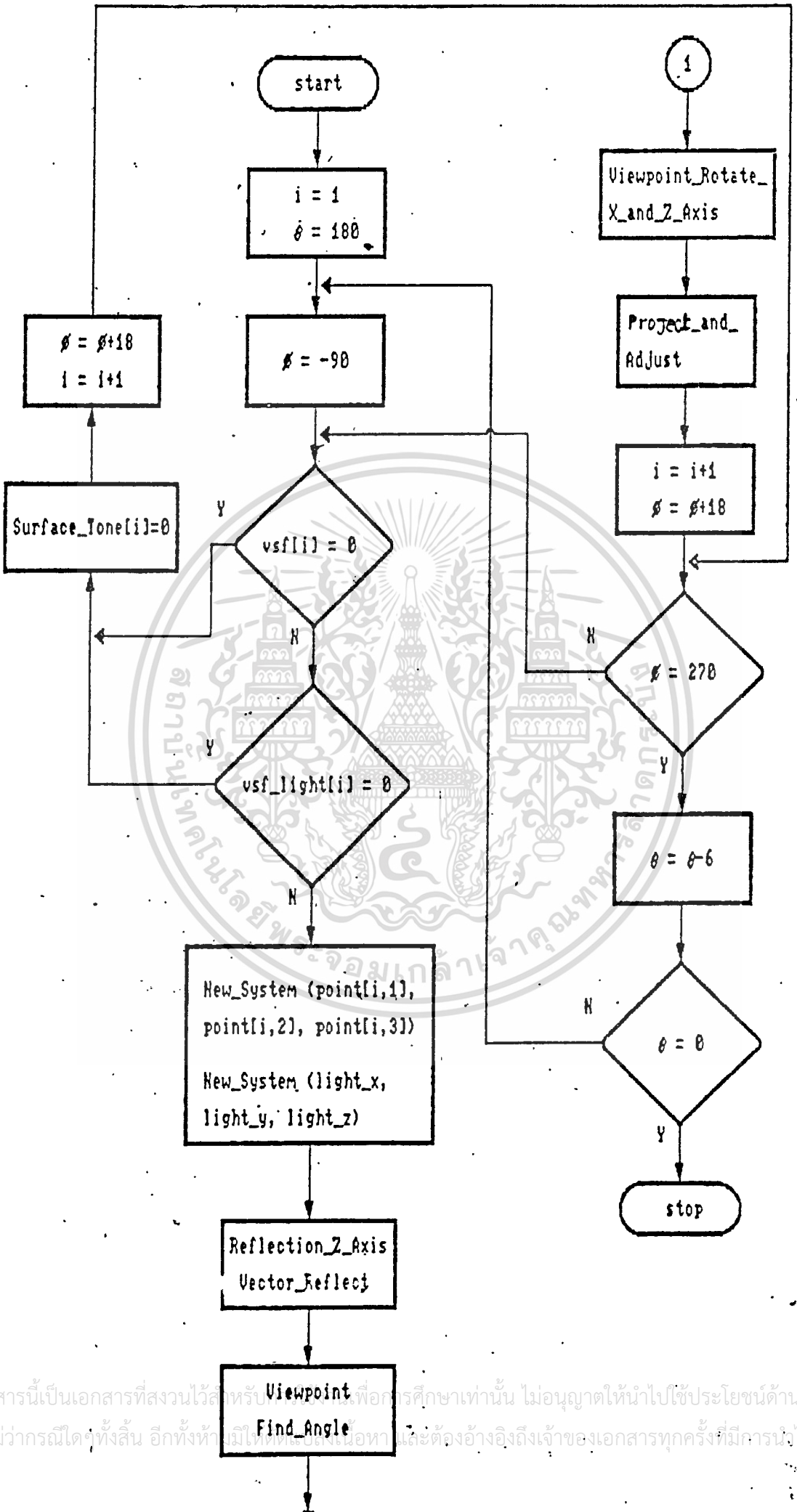


IEWPOINT = 0° , Ø_VIEWPOINT = 90°
IGHTSOURCE = -45°, Ø_LIGHTSOURCE = 45°

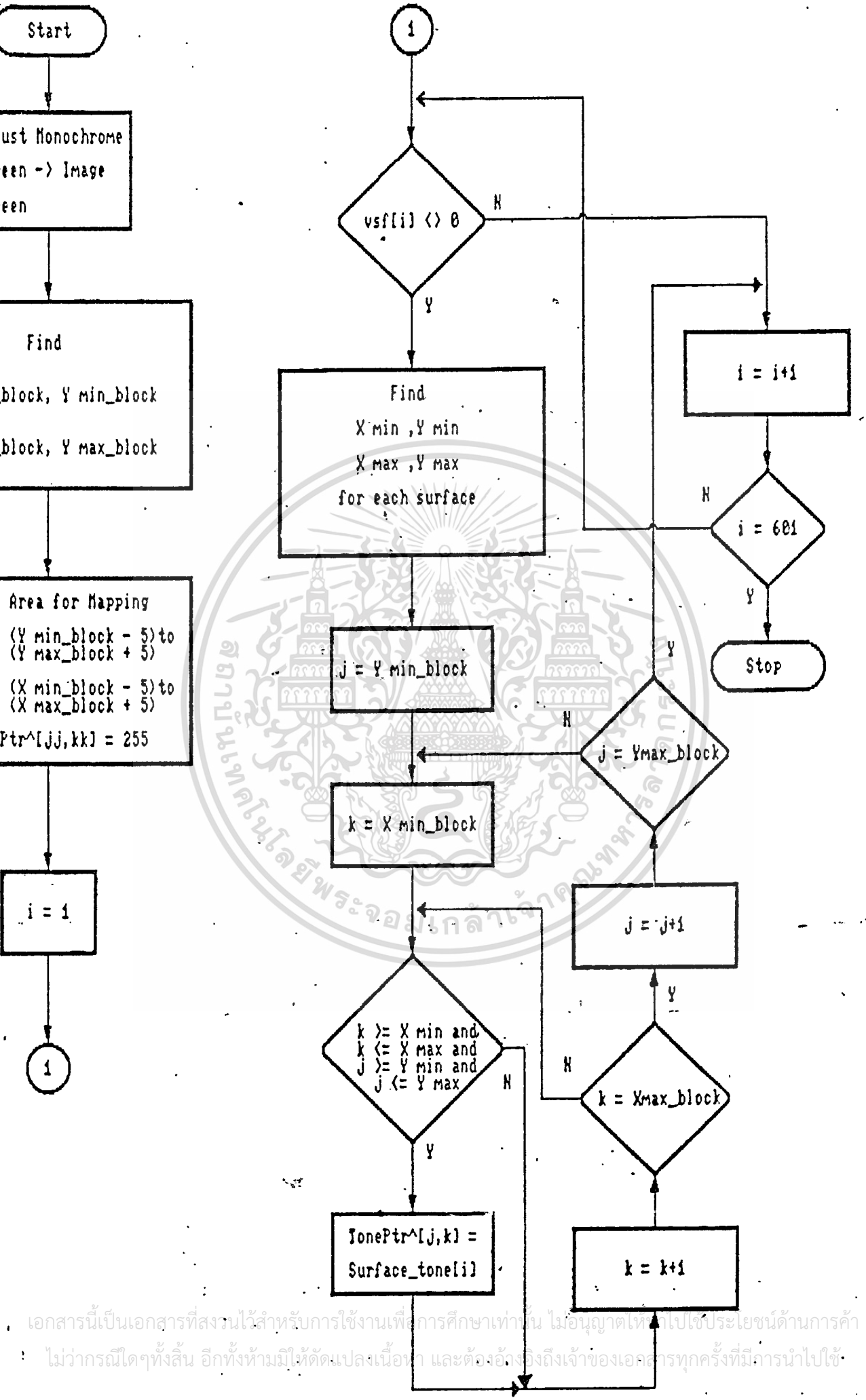


θ_VIEWPOINT = -45°, Ø_VIEWPOINT = 120°
θ_LIGHTSOURCE = -45°, Ø_LIGHTSOURCE = 135°

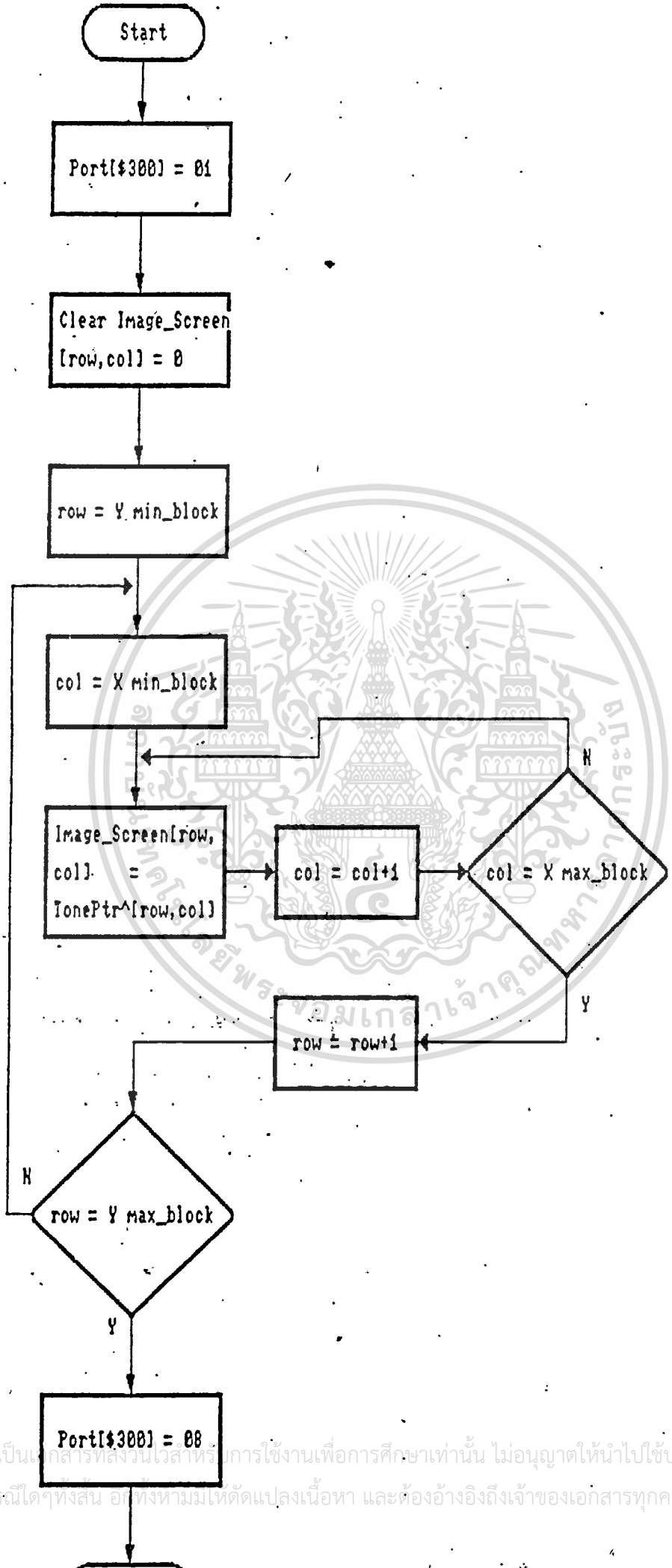
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ในเชิงพาณิชย์ ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

สรุปและวิจารณ์

ปริญญานิพนธ์ฉบับนี้ได้ถูกจัดทำขึ้นเพื่อวัตถุประสงค์ที่จะศึกษางานด้านกราฟฟิกบนไมโครคอมพิวเตอร์ โดยได้แสดงถึงทฤษฎีและวิธีการพื้นฐานต่างๆ รวมถึงทฤษฎีชั้นสูง เช่น ทฤษฎีการให้แสง (Shading) ด้วย ซึ่งเป็นภาคแรกของปริญญานิพนธ์ ส่วนในภาคหลังนั้นได้แสดงโปรแกรมต่างๆซึ่งได้นำทฤษฎีและวิธีการพื้นฐานเหล่านั้นมาพัฒนาขึ้นเป็นแบบจำลองเพื่อให้เป็นแนวทางในการศึกษาและสามารถพัฒนาให้เกิดประโยชน์ในการนำไปใช้งานจริงได้ โดยหวังว่าจะได้มีผู้นำไปพัฒนาต่อไปโดยไม่จำเป็นต้องเริ่มศึกษาใหม่กันอีกครั้งซึ่งคงจะเป็นประโยชน์ต่อผู้ที่มีความสนใจทางด้านนี้ เพราะขอบเขตทางด้านกราฟฟิคนั้นยังมีอีกมากมายที่น่าสนใจศึกษา โดยเฉพาะเมื่ออุปกรณ์คอมพิวเตอร์ที่ใช้ในงานกราฟฟิคมีราคาถูกลงและความสามารถเพิ่มขึ้น ทฤษฎีต่างๆที่มีความซับซ้อนกว่านี้ก็จะได้รับการศึกษาและพัฒนาจนสามารถนำไปใช้งานได้ในที่สุด ทำให้เกิดการประยุกต์ใช้งานได้อย่างกว้างขวางและจริงจังขึ้นกว่าที่เป็นอยู่ในปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ผู้จัดทำปริญญานิพนธ์ฉบับนี้ต้องขอขอบพระคุณบุคคลเหล่านี้ ที่มีส่วนทำให้ปริญญานิพนธ์สำเร็จลุล่วงลงไปได้

- ดร. บุญวัฒน์ อัตชู สำหรับความคิดริเริ่มของปริญญานิพนธ์ การสนับสนุนด้านวิชาการ รวมทั้งการเป็นอาจารย์ที่ปรึกษาด้วย
- คณาจารย์ ภาควิชาวิศวกรรมคอมพิวเตอร์ทุกท่าน ที่ได้ให้คำแนะนำด้านความรู้ต่าง ๆ ในการทำปริญญานิพนธ์
- วิโรจน์ สถาปนิกกุล, ศักดิ์ ว่องธนาการ, กำธร พาณิชปฐมพงศ์ และเพื่อน ๆ ทุกคนที่เป็นกำลังใจ และให้ความช่วยเหลือในโอกาสที่อำนวย

ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. กาญจนา กระแสร์ญาณ และ รศ. มนัส สังวรศิลป์, "การจำลองภาพด้วยวิธี Linear Shading", วารสารการประชุมวิชาการวิศวกรรมไฟฟ้า ครั้งที่ 10, เล่ม 2, หน้า 2-386 - 2-399.

2. Chan S. Park, "Interactive Microcomputer Graphics", Addison-Wesley Publishing Company, 458 p., 1985.

3. Steven Harrington, "Computer Graphics. A Programming Approach.", McGraw-Hill, 466 p., 1987.

4. Borland International Inc., "Turbo Pascal Reference Manual (Version 4.00)", 654 p., 1987.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้