



โปรแกรมช่วยการออกแบบภายในบ้านโดยแสดงในรูป 3 มิติ

1. นาย ชีรวัฒน์ อัสวโกลี
2. นาย พงศ์ศักดิ์ ตะวันกาญจน์โชติ
3. นางสาว เสาวณีย์ เล่าห์ขจร

ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
ภาควิชาคณิตศาสตร์ประยุกต์และวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2538

ร/พ.

5647 ร/

2538

612558849

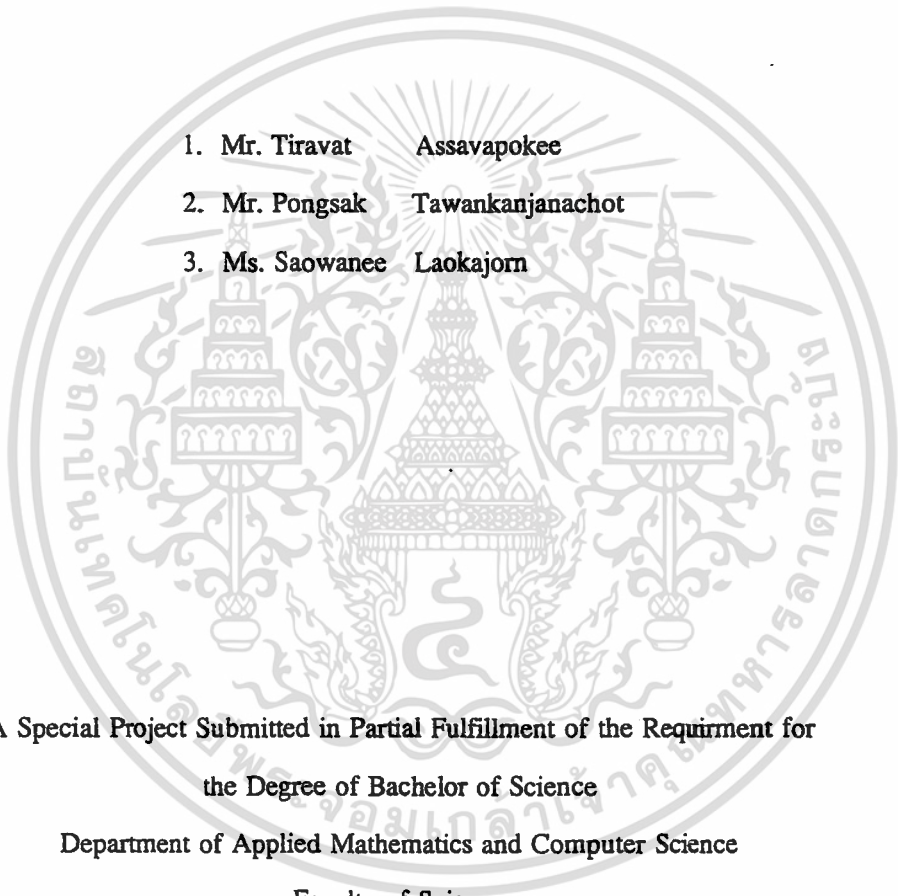
เลขหมู่.....

เลขทะเบียน.....

วันเดือนปี.....

Home Designing Program

by showing in 3 dimension

- 
1. Mr. Tiravat Assavapokee
 2. Mr. Pongsak Tawankanjanachot
 3. Ms. Saowanee Laokajorn

**A Special Project Submitted in Partial Fulfillment of the Requirement for
the Degree of Bachelor of Science**

Department of Applied Mathematics and Computer Science

Faculty of Science

King Mongkul's Institute of Technology Ladkrabang

1998

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงการพิเศษ โปรแกรมช่วยการออกแบบภายในบ้านโดยแสดงในรูปแบบ 3 มิติ
โดย นาย วีรวัฒน์ อัสวโกตี
 นาย พงศ์ศักดิ์ ตะวันกาญจน์โชติ
 นางสาว เสาวณีย์ เล่าห์ขจร
ภาควิชา คณิตศาสตร์ประยุกต์และวิทยาการคอมพิวเตอร์
อาจารย์ที่ปรึกษา รศ. ภัคคินี จิตสกุล

ภาควิชาคณิตศาสตร์ประยุกต์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยี
พระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้นำโครงการนี้เป็นส่วนหนึ่งของการศึกษา
ตามหลักสูตรวิทยาศาสตรบัณฑิต

.....
(รศ. ภัคคินี จิตสกุล)

หัวหน้าภาควิชา ฯ

.....
(อ.ศรัณย์ อินทโกสม)

ประธานกรรมการการสอบปัญหาพิเศษ

.....
(ผศ.สุนทร สุชาติเวชภูมิ)

กรรมการสอบปัญหาพิเศษ

.....
(รศ. ภัคคินี จิตสกุล)

กรรมการและอาจารย์ที่ปรึกษาปัญหาพิเศษ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทคัดย่อ

ทุกวันนี้ ความเจริญก้าวหน้าทางเทคโนโลยีของวงการคอมพิวเตอร์ปัจจุบัน ได้เข้ามา มีบทบาทในหลายสาขาวิชาชีพ ไม่ว่าจะเป็นในแง่ทางด้านธุรกิจ , วิศวกรรม , วิทยาศาสตร์ หรือแม้แต่ในงานสถาปัตยกรรม ปัญหาพิเศษนี้เป็นการนำเสนอโปรแกรมสำเร็จรูปเพื่อช่วยในการออกแบบภายในบ้านซึ่งแสดงในรูป 3 มิติ โดยใช้หลักการของการหว่านรังสี (Ray Casting) และเทคนิคเชอร์แมปปิง (Texture Mapping) ในการสร้างภาพมุมมอง 3 มิติ เพื่อความสะดวกของสถาปนิกในการออกแบบและช่วยเป็นสื่อทางความคิดจินตนาการผ่านทางโปรแกรมในรูปแบบที่ลูกค้าสามารถเข้าใจได้

ปัญหาพิเศษนี้ได้เขียนโดยใช้ Borland C++ และโปรแกรมช่วยเหลือทางด้านกราฟิก SVGA 256. BGI



Abstract

Nowaday microcomputer plays an important role in many fields such as Business , Engineering , Science and Architecture. This project presents Texture Mapping application on architect's interior design by using Ray Casting and Texture Mapping Technics to generate the 3-D visual scence for architect's convenience and communicating their idea pass program through customers.

This project is written on Borland C++ with some graphic utilities such as SVGA256.BGI



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

CONTENTS

หน้าอนุมัติ

บทคัดย่อปัญหาพิเศษภาษาไทย

บทคัดย่อปัญหาพิเศษภาษาอังกฤษ

สารบัญ	หน้า
บทที่ 1 - บทนำ	1
บทที่ 2 - ทฤษฎีและหลักการที่เกี่ยวข้อง	4
- 2.1 แนวความคิดของการหว่านรังสี (Ray Casting)	5
- 2.2 การนำหลักการหว่านรังสี (Ray Casting) มาประยุกต์ใช้กับโปรแกรม	6
- 2.2.1 หลักการมองเห็นกำแพง (Wall)	6
- 2.2.2 หลักการมองเห็นพื้น (Floor)	17
- 2.2.3 หลักการมองเห็นเพดาน (Ceiling)	20
- 2.2.4 หลักการมองเห็นวัตถุ (Object)	21
- 2.3 เรื่องเกี่ยวกับแสง	22
บทที่ 3 - การวิจัยและดำเนินงาน	24
บทที่ 4 - ผลการวิจัยและข้อเสนอแนะ	25
ภาคผนวก	
บรรณานุกรม	



บทที่ 1 - บทนำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

การพัฒนาการของคอมพิวเตอร์ในปัจจุบัน เทคโนโลยีทางด้านมัลติมีเดียได้มีการพัฒนาขึ้นอย่างรวดเร็ว มีโปรแกรมสำเร็จรูปหลายโปรแกรมที่เป็นที่นิยมใช้เทคนิคของคอมพิวเตอร์กราฟิกเพื่อความสวยงามและง่ายต่อการเรียนรู้การใช้งานของผู้ใช้โปรแกรมมือใหม่ ฉะนั้นการศึกษาและการพัฒนาโปรแกรมทางด้านคอมพิวเตอร์กราฟิกจึงเป็นที่ตื่นตัว เนื่องจากทำให้เกิดความสะดวกรวดเร็วในงานทางด้านสถาปัตยกรรม ซึ่งในช่วงระยะแรกสถาปนิกจะต้องออกแบบโครงสร้างต่างๆ ที่เป็นรูปภาพ 3 มิติ เช่น ภาพ Perspective หรือภาพในรูปแบบอื่นๆ เพื่อช่วยในการนำเสนอจินตนาการในรูปแบบที่ลูกค้าต้องการหรือสนใจลงในแปลนกระดาษ แต่ถ้าเกิดความผิดพลาดหรือไม่เป็นที่พอใจของลูกค้า สถาปนิกก็ต้องแก้ไข ตกแต่งใหม่ หรือต้องวาดใหม่ทั้งหมด เพื่อให้เป็นที่พอใจของลูกค้า ด้วยสาเหตุต่างๆ นี้จะทำให้เสียเวลาและสิ้นเปลืองแรงงานโดยเปล่าประโยชน์ ด้วยเหตุนี้เองทางคณะผู้จัดทำได้มีความเห็นพ้องต้องกันว่า ถ้าหากสามารถนำแบบแปลนที่ออกแบบในลักษณะ 2 มิติแล้วนำไปสร้างเป็นภาพ 3 มิติซึ่งเป็นลักษณะที่ทำให้ผู้ใช้โปรแกรมมีความรู้สึกเหมือนกับว่าท่านได้เข้าไปเดินอยู่ในห้องที่ออกแบบมา ถ้าอยากแก้ไขสิ่งใดก็สามารถที่จะทำได้โดยง่าย จึงช่วยให้ลดเวลาในการออกแบบของสถาปนิกลงได้มาก รูปแบบของโปรแกรมถูกพัฒนาขึ้นมาก่อให้เกิดความสะดวกสบายในหลายๆ ด้าน สามารถเรียนรู้วิธีการใช้งานได้อย่างรวดเร็ว โดยที่ผู้ใช้เองไม่จำเป็นต้องมีความรู้ทางด้านสถาปัตยกรรมศาสตร์อย่างมากมาย ช่วยให้บุคคลทั่วไปที่เป็นเจ้าของอาคารบ้านพักไม่จำเป็นต้องลงทุนไปจ้างสถาปนิกมาช่วยออกแบบให้ ซึ่งปัจจุบันค่าใช้จ่ายทางด้านนี้สูงมาก โปรแกรมสำเร็จรูปนี้จึงเป็นประโยชน์แก่ประชาชนทั่วไปไม่เฉพาะสถาปนิกเท่านั้น

สำหรับเทคนิคที่คัดเลือกมาใช้ในการสร้างภาพ 3 มิติในโปรแกรมนี้นี้คือ Ray Casting และ Texture Mapping ซึ่งทางคณะผู้จัดทำได้วิเคราะห์แล้วว่า เป็นเทคนิคที่เหมาะสม ทำให้คุณภาพของภาพที่ได้มามีลักษณะที่สมจริง ถึงแม้ว่าจะต้องอาศัยความสามารถของเครื่องคอมพิวเตอร์เพื่อช่วยในการคำนวณและสร้างภาพรวมทั้งเวลาที่ใช้ก็พอเหมาะพอควรกัน เพื่อจะได้มาซึ่งคุณภาพของผลงานที่อยู่ในเกณฑ์ที่ดี โดยรายละเอียดของเทคนิค จะได้กล่าวถึงในบทที่ 2

จากที่กล่าวมาทั้งหมด การออกแบบการตกแต่งภายในโดยใช้โปรแกรมสำเร็จรูปนี้ สามารถทำได้ครบวงจร ตั้งแต่การออกแบบกำหนดขอบเขตของห้องต่าง ๆ ในแปลน 2 มิติ ซึ่งสามารถเลือกลายกำแพง เพดาน และพื้นตามแต่ผู้ใช้ต้องการ อีกทั้งสามารถนำวัตถุวางไว้ตามจุดต่าง ๆ ในแปลน โดยที่ทั้งวัตถุ ลายพื้น ลายเพดาน และลายกำแพงนั้น มีมากมายพอสมควรที่จะให้ผู้ใช้เลือกได้ตามอัธยาศัย หลังจากนั้นสามารถตรวจสอบผลที่ได้จากแปลนที่สร้างในรูปแบบของ 3 มิติ ถ้าไม่เป็นที่พอใจก็สามารถที่จะนำแปลนดังกล่าวมาแก้ไขจนกว่าจะเป็นที่พอใจของผู้ใช้ แล้วจึงนำแบบแปลนนั้นไปสร้างของจริง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



บทที่ 2 - ทฤษฎีและหลักการที่เกี่ยวข้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงหลักการทางคณิตศาสตร์ ว่ามีวิธีอย่างไรจึงจะแปลงภาพ 2 มิติ ให้เป็นภาพ 3 มิติที่มีลักษณะใกล้เคียงกับโลกแห่งความเป็นจริงได้ โดยหลักการที่เรานำมาใช้คือ Ray Casting (การหว่านรังสี) เพราะมีการคำนวณที่ใช้เวลาไม่นานและคุณภาพของภาพที่ได้มีความละเอียดเหมือนจริง จึงเหมาะสมกับปัญหาพิเศษนี้

การหว่านรังสี (Ray Casting)

2.1 แนวความคิดของการหว่านรังสี (Ray Casting)

แนวความคิดของการหว่านรังสีอาศัยหลักการของการที่คนเรามองเห็นวัตถุสิ่งต่างๆ โดยที่สมมติให้ตาเป็นแหล่งกำเนิดแสง และเมื่อแสงที่ออกจากตาของเราไปกระทบกับวัตถุใด ก็ตามจะทำให้เรามองเห็นวัตถุนั้นๆ ดังรูปที่ 1



รูปที่ 1 แสดงการเดินทางของเส้นรังสีที่ออกจากตา

จากรูปที่ 1 เมื่อเส้นรังสีของแสงเส้นที่ 2 เดินทางผ่านมากระทบรูปหัวใจก็จะหยุดการเดินทางของเส้นรังสีที่วัตถุนั้น จึงทำให้เรามองเห็นรูปหัวใจซึ่งแสดงให้เห็นว่าทำไมเราจึงไม่เห็นรูปสี่เหลี่ยมข้าวหลามตัดที่อยู่ข้างหลังรูปหัวใจ เพราะว่าไม่มีเส้นรังสีแสงจากตาส่องไปกระทบ นั่นคือเราสามารถมองเห็นรูปวงกลม, รูปหัวใจ และรูปไบโพร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

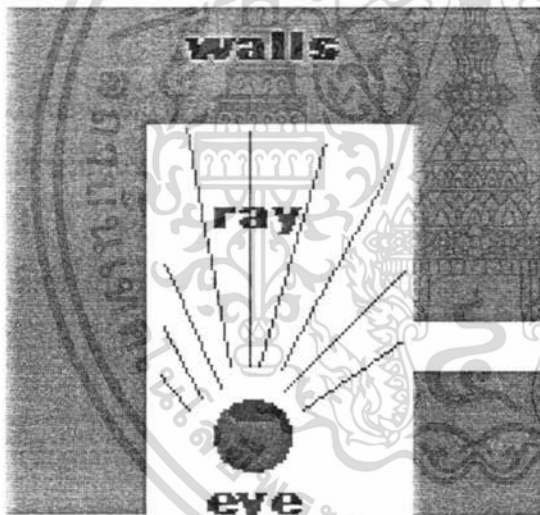
2.2 การนำหลักการหว่านรังสี (Ray Casting) มาประยุกต์ใช้กับโปรแกรม

ในปัญหาพิเศษนี้ได้แบ่งการมองเห็นออกเป็น 4 ประเภท

1. มองเห็นกำแพง (Wall)
2. มองเห็นพื้น (Floor)
3. มองเห็นเพดาน (Ceiling)
4. มองเห็นวัตถุ (Object)

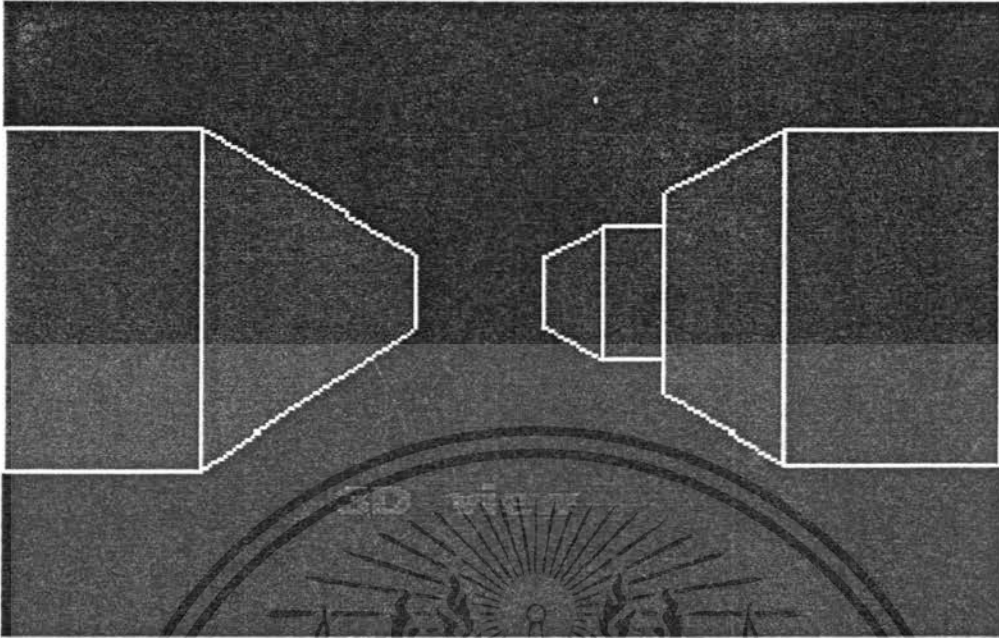
2.2.1 หลักการมองเห็นกำแพง (Wall)

กำแพงที่เราจะนำมาแสดงบนหน้าจอจะต้องเป็นรูป 3 มิติที่เราสามารถมองเห็น ซึ่งแผนที่และตำแหน่งที่อยู่ของกำแพงจะมีลักษณะเป็นดังรูปที่ 2



รูปที่ 2 แสดงเส้นรังสีที่เกิดออกจากตาไปกระทบกำแพง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3 แสดงภาพ 3 มิติของกำแพงที่ตาเรามองเห็น

สิ่งที่เราสนใจและนำมาใช้ในการคำนวณ

- แผนที่กำแพงใน 2 มิติ (Map)
- ตำแหน่งของผู้สังเกต (ระนาบพิกัด x,y)
- ทิศทางการมองของผู้สังเกต (Angle)

แผนที่กำแพงใน 2 มิติ (Map)

ในปัญหาพิเศษนี้เราใช้ Array 2 มิติในการเก็บแผนที่ของกำแพง

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$



รูปที่ 4 แสดง Array 2 มิติขนาด 3*4

รูปที่ 5 แสดงแผนที่กำแพง 2 มิติที่สร้างขึ้น

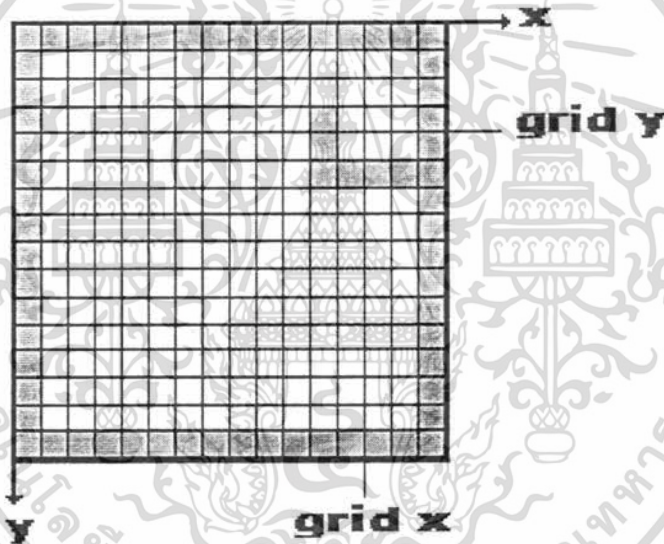
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ตัวเลขใน Array นี้อธิบายได้โดย

0 แทน พื้นที่ว่างไม่มีกำแพง

1 แทน รูปแบบของลายกำแพงซึ่งมีให้เลือกตั้งแต่ 1-15 ลาย

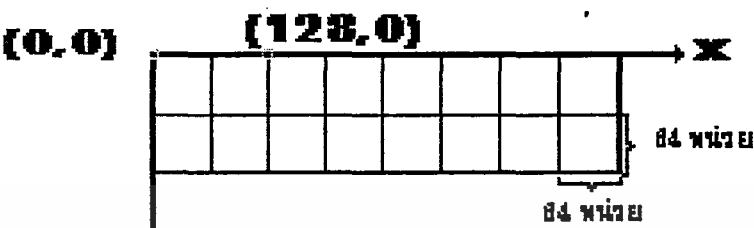
ต่อจากนั้นเราก็ทำการแปลง Array 2 มิตินี้ให้อยู่ในพิกัด x,y ที่เราใช้แสดงแผนที่เพื่อให้ง่ายต่อการคำนวณ เส้นพิกัดที่ลากมาตัดกันบนระนาบนี้มีลักษณะเหมือนตาข่าย โดยที่เส้นตามแกน x เรียกว่า $\text{grid } x$ และเส้นตามแกน y เรียกว่า $\text{grid } y$ ซึ่งเป็นดังรูปที่ 6



รูปที่ 6 แสดงแผนที่ 2 มิติบนระนาบพิกัด x,y

พิจารณาพิกัด x,y ซึ่งแทนตำแหน่งที่รังสีตกกระทบและกำหนดให้ แต่ละ Box มีขนาด 64×64 หน่วย โดยอธิบายได้ดังรูปที่ 7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7 แสดงพิกัด x,y และ Box ที่เราใช้แทนด้วยขนาดของ 64*64 หน่วย

ตำแหน่งของผู้สังเกต

จากการที่เราสามารถนำแผนที่เข้ามาในพิกัด x,y ได้แล้วเพราะฉะนั้นจะมีพิกัดที่แน่นอนของผู้สังเกตในพิกัด x,y

ทิศทางการมอง

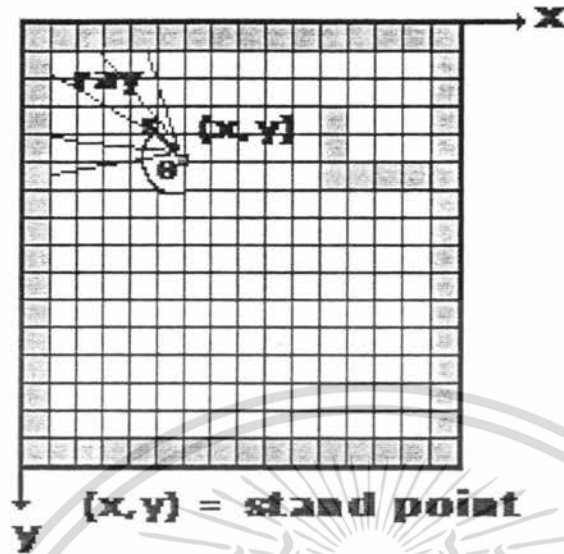
เราพิจารณาจากมุม (θ) ที่กระทำกับแกน y ถ้าหมุนตามเข็มนาฬิกาเป็นบวก (+) และทวนเข็มนาฬิกาเป็นลบ (-) ดังรูปที่ 8



รูปที่ 8 แสดงทิศทางการมองด้วยมุม (θ)

โดยที่เราจะแบ่งมุม 360° ออกเป็น 4096 ส่วน ดังนั้นมุม (θ) ที่เราใช้ในการคำนวณจึงมีความละเอียดมาก

เมื่อเรารวมตำแหน่งที่รังสีตกกระทบ, ตำแหน่งผู้สังเกต และทิศการมองเข้าด้วยกันแล้ว จะแสดงเส้นรังสีที่แสงจากตาของผู้สังเกตเดินทางไปตกกระทบที่จุดต่างๆ ได้ดังรูปที่ 9

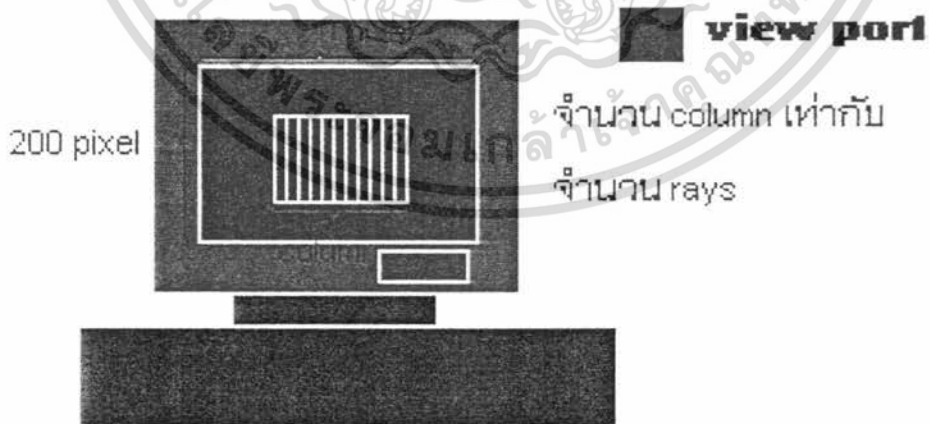


รูปที่ 9 แสดงเส้นทางเดินของรังสีแสงที่ออกจากตาผู้สังเกต

รังสีของการมองเห็น (Ray) และการหว่าน (Casting)

หลักการของการหว่านรังสีมีตาเป็นแหล่งกำเนิดแสงซึ่งจำนวนของรังสีนั้นจะมีก็เส้นก็ต้องพิจารณาจากโหมดของจอภาพที่เราได้นำมาใช้ในปัญหาพิเศษนี้คือ โหมด 13H ซึ่งเป็นโหมดที่มีความละเอียด 320*200 จุด 256 สี นั่นคือมี 320 แถว 200 หลัก เพราะฉะนั้นจำนวนรังสีที่ยังออกจากตาจึงเท่ากับจำนวนหลักของ Viewport ที่ใช้ในปัญหาพิเศษนี้ ดังรูปที่ 10

จอภาพ 320 x 200



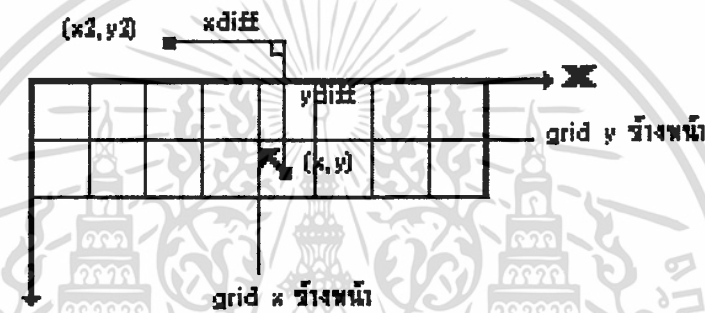
รูปที่ 10 แสดงจอภาพและจำนวนเส้นรังสีที่ยังออกจากตา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักเกณฑ์ที่ใช้ในการคำนวณเพื่อวาดภาพ 3 มิติ

พิจารณารังสี (Ray) หนึ่งเส้น, grid x, grid y และ Array 2 มิติที่เก็บรายละเอียดของภาพซึ่งภาพของภาพจะอยู่ในรูปแบบของไฟล์ .PCX สิ่งต่างๆเหล่านี้จะใช้ในการคำนวณดังนี้

ขั้นที่ 1 หาตำแหน่ง grid x และ grid y ที่อยู่ข้างหน้าผู้สังเกต ในทิศทางการมองของผู้สังเกต ซึ่งอธิบายได้ดังรูปที่ 11



รูปที่ 11 แสดงรูปของการคำนวณขั้นที่ 1

จากรูปที่ 11 เริ่มต้นด้วยการกำหนดจุดที่ไกลจากจุด (0,0) ไปตามแกน y คือ จุด (0,1024) แล้วจึงทำการหมุนไปเป็นมุม θ (มุมของการมอง) จากนั้นย้ายจุดเริ่มต้นจากจุด (0,0) ไปอยู่ที่ (x,y) จะได้จุดที่ตำแหน่ง (x_2, y_2) ในรูปที่ 11 มีการแสดงให้เห็น xdiff , ydiff

โดยหาได้จาก $xdiff = x_2 - x$;

$ydiff = y_2 - y$;

ดังนั้น ค่าความชัน (slop) = $ydiff / xdiff$

หลังจากนั้นจะคำนวณหาตำแหน่ง grid x และ grid y ที่อยู่ข้างหน้าผู้สังเกตได้โดยพิจารณาเครื่องหมายของ xdiff และ ydiff

xdiff เป็น (+) $\rightarrow$ grid x = (x & 0xffc0) + 64 ;

xdiff เป็น (-) $\rightarrow$ grid x = (x & 0xffc0) - 1 ;

ydiff เป็น (+) $\rightarrow$ grid y = (y & 0xffc0) + 64 ;

ydiff เป็น (-) $\rightarrow$ grid y = (y & 0xffc0) - 1 ;

เมื่อคำนวณหาตำแหน่ง grid x และ grid y ที่อยู่ข้างหน้าผู้สังเกตได้แล้ว จึงทำการคำนวณหาจุดตัด grid x , grid y กับ รั้วสี่

โดยให้ (xcross-x,xcross-y) คือพิกัดของจุดตัดระหว่างรั้วสี่กับ grid x

และ (ycross-x,ycross-y) คือพิกัดของจุดตัดระหว่างรั้วสี่กับ grid y

ซึ่งค่าต่างๆสามารถหาได้โดยการคำนวณเช่นเดียวกับการหาจุดตัดของเส้นตรงสองเส้น ดังนี้

xcross-x = grid x ;

xcross-y = y + slop * (grid x - x) ;

ycross-x = x + (grid y - y) / slop ;

ycross-y = grid y ;

จากนั้นก็พิจารณาระยะห่างระหว่างจุด (x,y) กับจุดตัดทั้งสองนี้ว่าจุด (x,y) ใกล้กับจุดใดมากกว่ากันหรือในอีกแง่หนึ่งรั้วสี่จะตัดกับ grid x หรือ grid y ก่อนกันซึ่งจะแทนจุดนี้ด้วย tm_column แล้วกำหนดให้ x_maze = x/64 และ y_maze = y/64

เช่น ถ้าตัดกับ grid x ก่อน พิจารณาค่าที่อยู่ใน Box แรกที่ติดกับ grid x นั้นว่าค่าตัวเลขที่อยู่ใน Array 2 มิติเป็น 0 หรือ 1-15

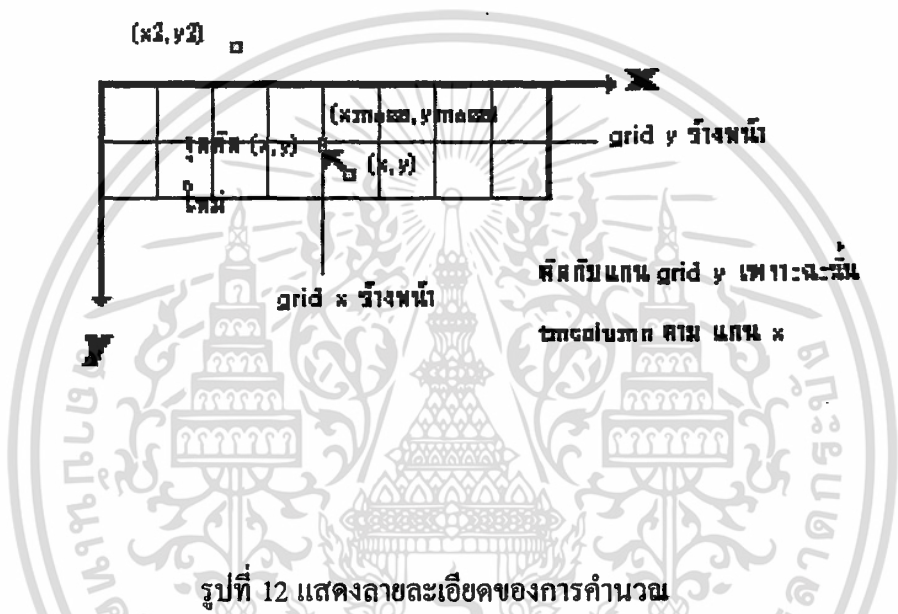
ถ้าเป็น 0 -ให้ทำการหา grid x และ grid y ต่อไปโดยเปลี่ยนค่า (x,y) จากตำแหน่งของผู้สังเกตเป็นจุดตัดครั้งแรกแล้ว

- คำนวณหาค่า x_maze,y_maze ออกมา

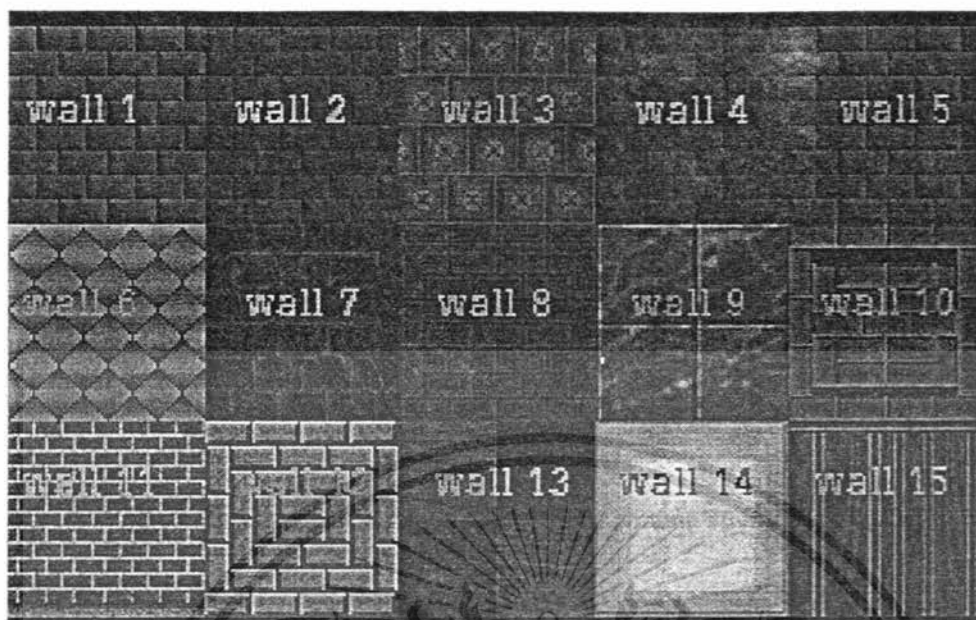
- เช็คค่า Walls[x_maze][y_maze] = ?

ถ้าเป็น 1-15 - คำนวณหาค่า `tm_column` โดย → ถ้าตัดกับ `grid x` จะคิดตามแกน `y` :
`tm_column = y & 0x3f ;`
→ ถ้าตัดกับ `grid y` จะคิดตามแกน `x` :
`tm_column = x & 0x3f ;`

ซึ่งจะอธิบายได้ดังรูปที่ 12



ในปัญหาพิเศษนี้รูปกำแพงจะเป็น โครงสร้างของไฟล์.PCX ซึ่งอยู่ในไฟล์ชื่อ Walls.
แต่เราจะไม่ใส่นามสกุล .PCX ลงไปในชื่อเพื่อป้องกันการแก้ไขดัดแปลงจากบุคคลภายนอก
แสดงได้ดังรูปที่ 13



รูปที่ 13 แสดงภาพกำแพงของไฟล์ชื่อ Walls.

เมื่อคำนวณหาค่า `tm_column` ได้และรู้ค่าหมายเลขกำแพงจาก Array 2 มิติแล้วจะทำให้เราเข้าถึงลายกำแพงตามหมายเลขที่เราเลือกโดยหาค่า `column` ของไฟล์ Walls จากค่า `tm_column` นำ `column` นั้นไปวาดลงบนจอภาพที่ `column` ที่เรียงเส้นนั้นผ่านพร้อมทั้งทำการย่อหรือขยายภาพ สมมติอ่านค่าจาก Array ได้กำแพงหมายเลข 2 ซึ่งอธิบายได้ดังรูปที่ 14



รูปที่ 14 แสดงรายละเอียดของการวาดภาพ 3 มิติบนจอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความสูงของกำแพงที่เรามองเห็น คำนวณได้จาก

$$\text{ความสูงที่เห็น} = \frac{\text{ความสูงจริง} * (\text{Viewer distance})}{\text{ระยะห่าง}}$$

และ

$$\text{จุดต่ำสุด} = \frac{\text{Viewer distance} * \text{Viewer height} + \text{Viewport center}}{\text{ระยะห่าง}}$$

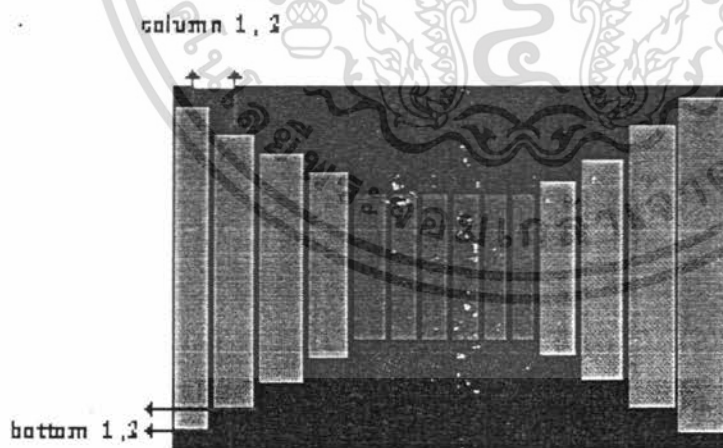
โดยการคำนวณจุดต่ำสุด คือการหาค่า Row ที่เป็น bottom ของกำแพงที่จะวาดลงบนจอภาพ

Viewer distance คือ ค่าคงที่ที่กำหนดขึ้นเพื่อแทนระยะห่างเหมือนระยะระหว่าง User กับ คอมพิวเตอร์

Viewer height คือ ค่าความสูงเสมือนของคนในจอภาพ

Viewport center คือ ค่า Row ตรงกลางของ Viewport ในจอภาพ

เพราะฉะนั้นการวาดแต่ละ column จะต้องมีการคำนวณหาความสูงที่เห็นทุกครั้งจึงทำให้ภาพที่ได้จากวิธีการนี้เป็น 3 มิติ ดังรูปที่ 15



รูปที่ 15 แสดงภาพ 3 มิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นได้ว่าภาพที่แสดงนั้น column 1 , column 2 และ column อื่นๆ มีความสูงแตกต่างกันไป เพราะขึ้นกับระยะห่าง โดยที่เป็นลายกำแพงเดียวกันทำให้เกิดภาพ 3 มิติซึ่งมีลักษณะเหมือนจริง เมื่อรังสีทุกเส้นได้ถูกคำนวณและวาดลงบนจอภาพจนครบทุก Column ใน Viewport แล้ว

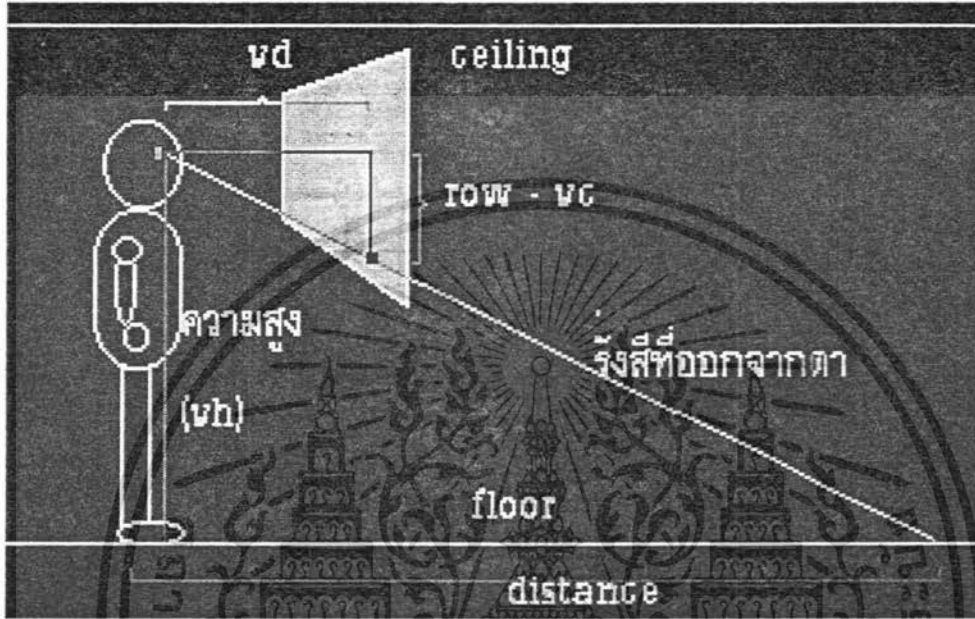
2.2.2 หลักการมองเห็นพื้น (Floor)

เราจะใช้หลักการของรังสีที่ยังออกจากตาแล้วเดินทางผ่านไปกระทบพื้น แล้วพิจารณาว่าพบจุดที่มีค่าหมายเลขสี่ตรงกับสีอะไรจากนั้นก็นำมาวาดขึ้นหน้าจอ ดังรูปที่ 16



จากรูปที่ 16 ตรงจุด จะมีพิกัดของมันบนจอภาพ นำจุดพิกัดนี้ไปคำนวณร่วมกับค่าใน Array 2 มิติของ Floor และไฟล์ภาพของพื้น จากนั้นทำการคำนวณว่า ตรงจุด ควรจะเป็นสีอะไรแล้วนำมา plot ลงบนหน้าจอก็จะทำให้เกิดพื้นขึ้น

การพิจารณาว่าตำแหน่งบนจอภาพจะเห็นจุดสีใดสามารถหาได้จากทฤษฎีดังรูปที่ 17



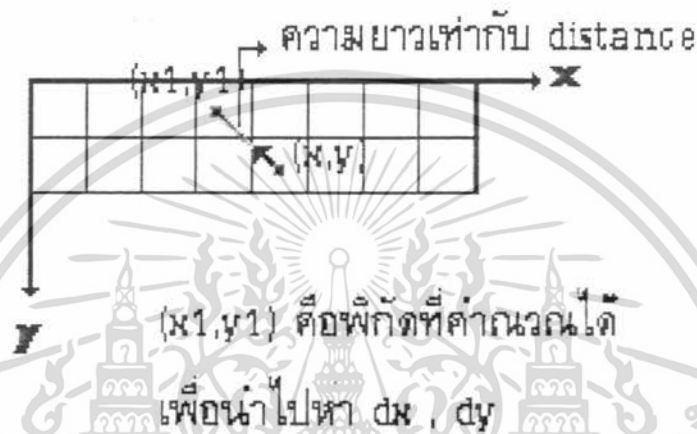
รูปที่ 17 แสดงภาพอธิบายประกอบทฤษฎีของการมองเห็นพื้น

เราจะคำนวณค่า distance ได้จากการนำหลักการของสามเหลี่ยมคล้ายเข้าช่วย จะได้ว่า

$$\text{distance} = \left[\frac{\text{Viewer distance}}{\text{Row} - \text{Viewport center}} \right] * \text{Viewer height}$$

จากนั้นเช็กในแผนที่ 2 มิติและพิกัด x, y เหมือนกันกับในกรณีของกำแพง คือ มีตำแหน่งของผู้สังเกตบนพิกัด x,y และมีทิศทางของรังสีที่ละ Column ในทำนองเดียวกัน เพียงแต่จะวาดเฉพาะบริเวณที่ไม่มีกำแพงอยู่ ในการหาจุดพิกัดที่อยู่ห่างจากตำแหน่งที่ผู้สังเกตอยู่ไปในทิศทางของรังสี โดยที่มีระยะห่างเท่ากับ distance ดังรูปที่ 18

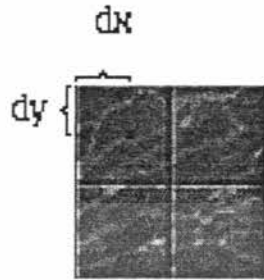
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 18 แสดงการหาจุดพิกัดที่อยู่ห่างจากตำแหน่งที่ผู้สังเกตอยู่ไปในทิศทางของรังสี

จากรูปที่ 18 จุดพิกัด (x_1, y_1) คือจุดที่เราต้องการนำไปเทียบกับไฟล์ภาพรูปพื้นเพื่อว่าตรงกับค่าหมายเลขสีใดแล้วเอานั่นไป plot ลงบนหน้าจอ ณ ตำแหน่ง (Column, Row) เมื่อเราหาจุดพิกัด (x_1, y_1) ได้เราก็สามารถหาค่า dx และ dy ได้ จากนั้น ก็ทำการคำนวณค่า $x_{maze} = x_1 / 64$ และ $y_{maze} = y_1 / 64$ นำไปตรวจดูค่าในอาร์เรย์ 2 มิติ $Floor[x_{maze}][y_{maze}]$ ว่าเป็นหมายเลขของพื้นลายใด แล้วเข้าไปในไฟล์ภาพของพื้น สมมติว่าเป็นพื้นหมายเลข 0 จะแสดงได้ดังรูปที่ 19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



pixel นี้จะถูกนำไป plot บนจอภาพ

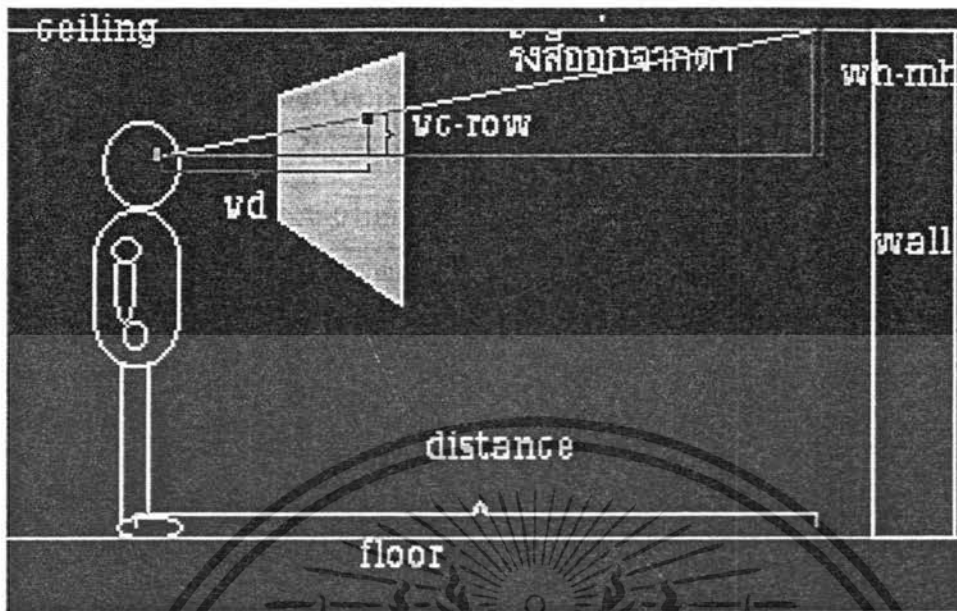
รูปที่ 19 แสดงรูปไฟล์ภาพพื้น

เมื่อทำเช่นนี้กับทุกๆจุดที่เป็นส่วนของพื้นก็จะ ได้ภาพ 3 มิติของพื้นออกมา

2.2.3 หลักการมองเห็นเพดาน (Ceiling)

ทุกอย่างใช้หลักการเช่นเดียวกับการมองเห็นพื้นแต่เปลี่ยนแค่สูตรในการหา distance

อธิบายได้ดังรูปที่ 20



รูปที่ 20 แสดงหลักการมองเห็นเพดาน

สูตรที่ใช้ในการหาค่า distance สำหรับเพดานคือ

$$\text{distance} = \frac{(\text{Wall height} - \text{Viewer height}) * \text{Viewer distance}}{\text{Viewport center} - \text{Row}}$$

เมื่อคำนวณหาค่า distance ได้ ดำเนินวิธีการทุกอย่างทำนองเดียวกับหลักการมองเห็นพื้นกับ ทุกๆจุดที่เป็นส่วนของเพดาน ก็จะทำให้เราได้รูปเพดานที่มีลักษณะเป็น 3 มิติ

2.2.4 หลักการมองเห็นวัตถุ (Object)

ในตอนนี้นับจอภาพเราจะได้ภาพ 3 มิติของกำแพง,พื้น และเพดาน ต่อไปเราจะกล่าวถึงการนำ ภาพวัตถุ (Objects) ขึ้นมาวาดบนจอภาพ และเช่นเดียวกันเราได้ทำการเก็บรายละเอียดต่างๆที่ เกี่ยวกับวัตถุว่ามีวัตถุหมายเลขใด วางอยู่ในตำแหน่งใดในบริเวณที่เราทำการตกแต่งลงใน Array 2 มิติของวัตถุคือ Obj[16][16] ซึ่งมีขนาดเท่ากับ Array ของ Walls[16][16] ,Floor[16][16] และ Ceiling[16][16] เพื่อนำข้อมูลมาใช้ในการคำนวณ โดยที่ Array ใช้เก็บเส้นทางที่ รังสีผ่านเช่นในขณะวาดพื้นรังสีผ่าน Box ใดบ้างแล้วทำการเซตค่าใน Array นี้ให้มีค่าเป็น 1 ตรง Box นั้น จากนั้นทำการเช็คดูว่าภายใน Array นี้มีจุดใดบ้างที่เป็น 1 แล้วนำตำแหน่งนั้นมา ดูค่าใน Array Obj ที่ตำแหน่งเดียวกันว่ามีวัตถุวางอยู่หรือไม่ ถ้าไม่มีก็ผ่านไปตำแหน่งอื่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ถ้ามีก็จะอ่านว่าเป็นวัตถุหมายเลขอะไร ขึ้นตอนถัดไป Program จะทำการอ่านและเขียนชื่อเกี่ยวกับ Objects หมายเลขนั้นจาก Array Objtype[60] นั่นคือสามารถวางวัตถุได้สูงสุด 60 ชิ้นซึ่งใน Objtype จะเก็บข้อมูลต่างๆเช่น หมายเลขของวัตถุที่ตรงกับหมายเลขนี้เป็นวัตถุรูปอะไร อยู่ที่ตำแหน่งใด ขึ้นต่อไปก็ทำการคำนวณหาค่า ระยะห่างระหว่างวัตถุกับตำแหน่งของผู้สังเกต หลังจากนั้นทำการคำนวณหาขนาดของวัตถุที่มองเห็นแล้วนำมาคำนวณหาตำแหน่งของจุดบนจอภาพ และเก็บข้อมูลลงใน Array ObjList [] ทำการเรียงลำดับของ Object จากหมายเลขของวัตถุที่อยู่ไกลไปหาหมายเลขของวัตถุที่อยู่ใกล้ใน Array ObjList [] แล้วเอารายละเอียดใน Array Objtype[] ของ Box พิจารณาหมายเลขที่ตรงกับค่าที่อยู่ใน Array ObjList [] มาทำการวาดลงบนจอภาพที่มี การวาดกำแพง, พื้น และเพดานไว้ก่อนแล้วโดยจะทำการวาดวัตถุที่อยู่ไกลก่อน ซึ่งภาพของวัตถุที่นำมาวาดจะเก็บไว้ในไฟล์ชื่อ Obj จากวิธีการทั้งหมดที่กล่าวมาจะทำให้เราได้ภาพ 3 มิติที่สมบูรณ์และเหมือนจริง

2.3 เรื่องเกี่ยวกับแสง

ใน Program 3D House Design นี้ได้มีการสร้างตารางแสงที่มีขนาด 32*256 โดยที่ 32 คือ ระดับของแสง 256 คือ จำนวนหมายเลขสี

ซึ่งสามารถอธิบายได้ดังนี้ ถ้ามีความเข้มของแสงเท่ากับ 16 และ หมายเลขของสีเท่ากับ 50 จะถูกเปลี่ยนไปเป็นค่าที่อยู่ในตารางแสงที่มีระดับแสงเท่ากับ 16 และมีหมายเลขสีเท่ากับ 50 ซึ่งจะได้หมายเลขสีใหม่แล้วจึงนำมา plot ลงบนหน้าจอทำให้เกิดแสงเงาขึ้น ดังนั้นตารางแสงนี้จึงถูกใช้ทุกครั้งที่มีการ plot จุดสีลงบนหน้าจอ นับว่าเป็นเทคนิคสำคัญที่ทำให้เกิดแสงเงาโดย

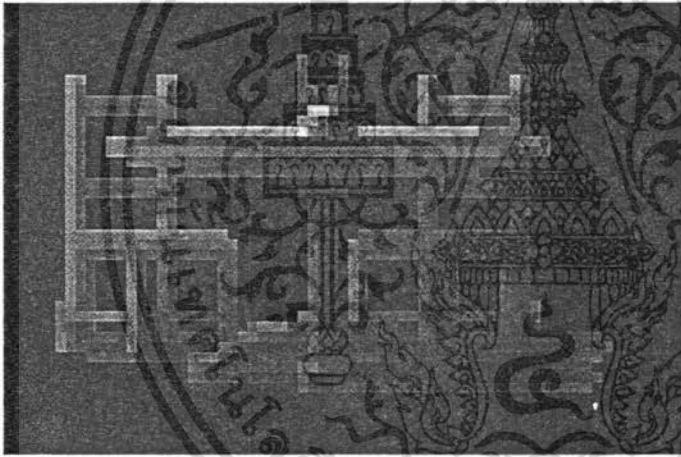
ความเข้มของแสง = ความเข้มที่ขึ้นกับระยะ + ความเข้มของแสงในห้อง
ซึ่งความเข้มของแสงในห้องที่ใช้จะเป็นค่าคงที่ตั้งแต่ 0 ถึง 32

$$\frac{\text{ความเข้มแสงที่ขึ้นกับระยะ} = \text{ความสามารถในการมองเห็น} * 3}{\text{ระยะทาง}}$$

ทั้งนี้ จากการทดสอบโปรแกรมหลายๆรอบ พบว่า ค่า 3 ที่นำมาในสูตรการคำนวณเหมาะสมที่สุดสำหรับการนำมาใช้งาน ถ้าใช้ตัวเลขที่น้อยกว่าหรือมากกว่านี้ ก็จะทำให้ได้ภาพที่มีระดับแสงต่างกันน้อยเกินไปและมากเกินไปตามลำดับ และ ความสามารถในการมองเห็นเป็นค่าคงที่ตั้งแต่ 0 ถึง 32



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



บทที่ 3 - การวิจัยและดำเนินการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

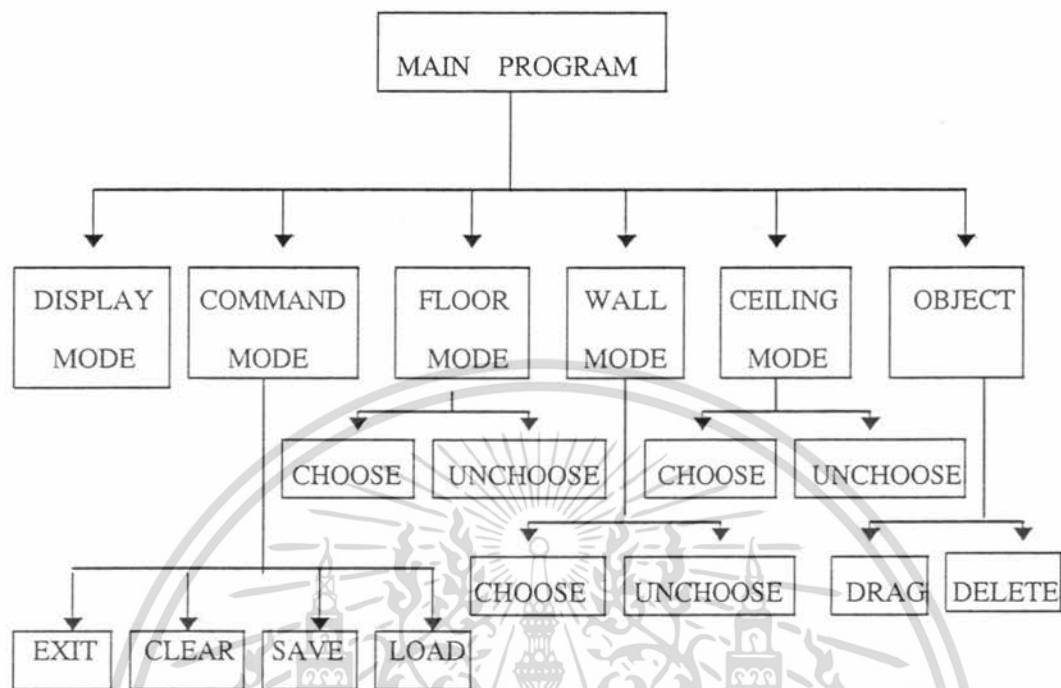
การวิจัยและการดำเนินการ

การออกแบบ

ในขั้นแรกของการออกแบบ ได้ทำการกำหนดโครงร่างขึ้นมาก่อนดังนี้

- การนำเสนอผลงานที่เป็นกราฟิกทั้งหมดในลักษณะของ GUI (GRAPHICAL USER INTERFACE)
- โหมดกราฟิกที่ใช้ในการสร้างภาพเป็นโหมด 13 ซึ่งมีความละเอียด 320*200 จุด 256 สี
- ภาพที่นำมาใช้ในโปรแกรมเป็นภาพฟอร์แมตของ PCX
- สามารถสร้างแบบแปลนในระนาบ 2 มิติ โดยใช้ระบบพิกัด X-Y
- สามารถทำการบันทึกแบบแปลนที่สร้างลงไฟล์และสามารถนำแบบแปลนเดิมซึ่งออกแบบไว้แล้วมาปรับปรุงเปลี่ยนแปลงได้
- สามารถกำหนดลายกำแพง , ลายเพดาน , ลายพื้น และวัตถุจากแบบที่มีไว้ให้เลือกได้ตามต้องการ
- สามารถกำหนดจุดเริ่มต้นของมุมมองเองได้
- ในระบบ 3 มิติ ผู้ใช้สามารถเดินไปมาเพื่อดูห้องที่ออกแบบไว้ได้

หลังจากที่กำหนดลักษณะและความสามารถของโปรแกรมแล้วจึงทำการร่างขั้นตอนการทำงานของโปรแกรมเพื่อให้สอดคล้องกับแนวทางที่ได้กำหนดขึ้นดังรูปต่อไปนี้



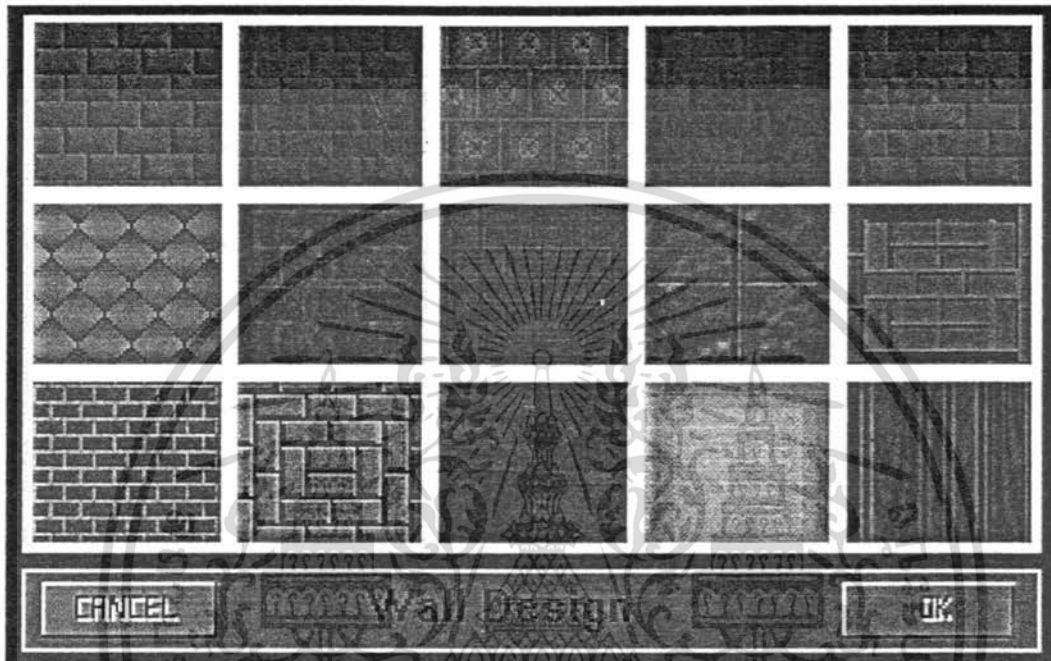
จากขอบเขตและโครงสร้างของขั้นตอนการทำงาน ทางคณะผู้จัดทำจึงออกแบบหน้าจอหลักซึ่งใช้เมาส์ในการดำเนินงาน เพื่อความสะดวกรวดเร็วและง่ายต่อการใช้งานดังนี้

- 1 :  - Exit / ออกจากโปรแกรม
- 2 :  - Floor / โหมดพื้น
- 3 :  - Walls / โหมดกำแพง
- 4 :  - Ceiling / โหมดเพดาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 5 : **SELECT** - Select / เลือกหลายจากทั้งหมด
- 6 : **↔** - Arrow / เลือกหลายถัดไป
- 7 : **MAP** - Map / โหมดปกติของแผนผัง
- 8 : **OBJECTS** - Objects / วางวัตถุลงบนแผนผัง
- 9 : **LOAD** - Load / อ่านแผนผังที่สร้างแล้วขึ้นมา
- 10 : **SAVE** - Save / เก็บบันทึกแผนผังลงดิสก์
- 11 : **CLEAR** - Clear / สร้างแผนผังขึ้นมาใหม่

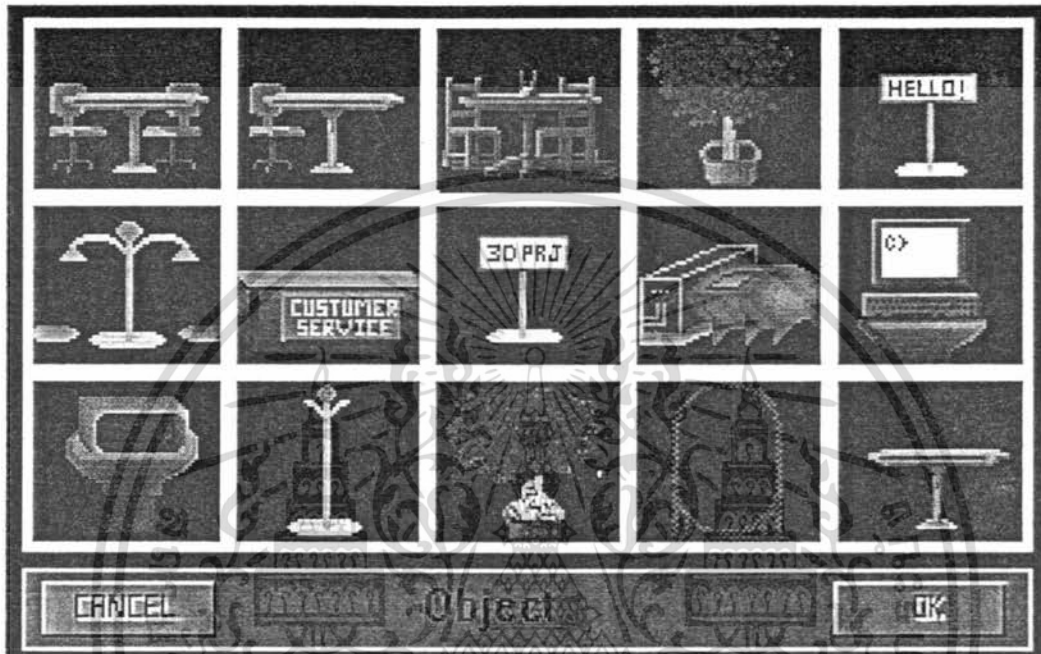
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



- หน้าจอสำหรับเลือกลายของกำแพง ลายพื้น ลายเพดาน

- 1 : รูปแบบลายของ WALLPAPER
- 2 : CANCEL ยกเลิกลายที่เลือก
- 3 : OK ยอมรับลายที่เลือก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



- หน้าจอสำหรับเลือกวัตถุที่จะนำไปวางลงในแบบแปลน

- 1 : รูปแบบของวัตถุที่จะนำไปวางในแบบแปลน
- 2 : CANCEL ยกเลิกวัตถุที่เลือก
- 3 : OK ขอมรับวัตถุที่เลือก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การพัฒนาและปรับปรุงโปรแกรม

เมื่อได้ทำการออกแบบส่วนประกอบต่าง ๆ ทั้งหน้าจอ หน้า ที่การทำงาน รวมทั้งข้อจำกัดต่าง ๆ ของโปรแกรมแล้ว จึงเริ่มทำการพัฒนาโปรแกรมให้เป็นที่กำหนดไว้ หลังจากนั้นจึงทำการทดสอบเพื่อหาข้อบกพร่อง แล้วจึงทำการแก้ไขปรับปรุงและเพิ่มเติมเพื่อความสมบูรณ์และถูกต้องของโปรแกรม



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



บทที่ 4 - ผลการวิจัยและวิจารณ์

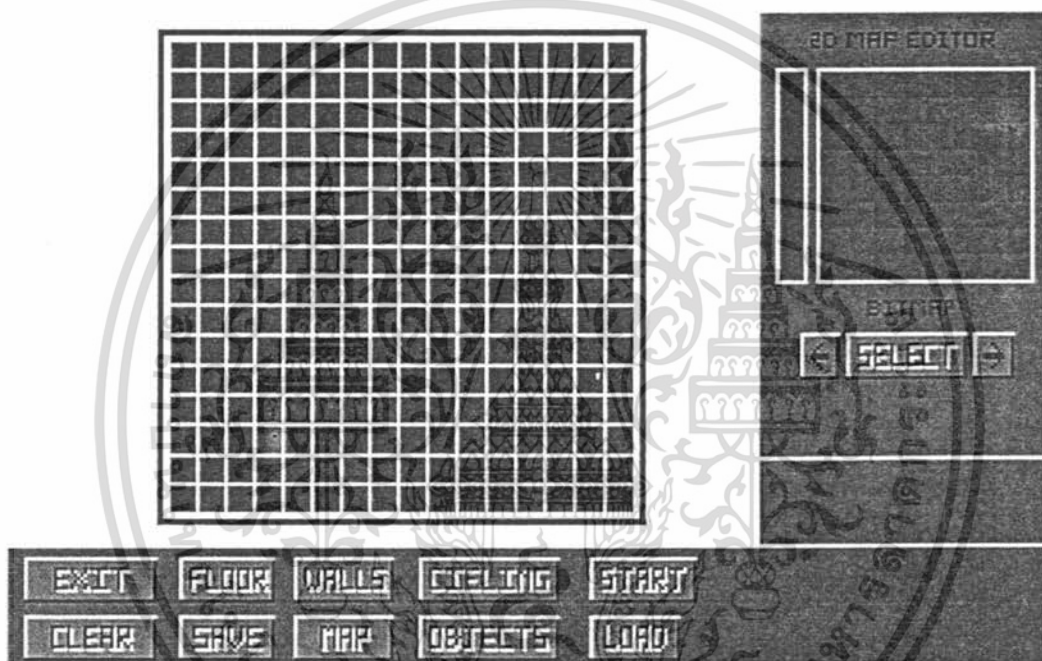
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการวิจัยและข้อเสนอแนะ

ผลการวิจัย

สามารถให้โปรแกรมมีคุณสมบัติตามที่กำหนดไว้ได้สำเร็จ และให้ผลการดำเนินงานเป็นไปตามที่คาดหวังไว้ ส่วนของการติดต่อกับผู้ใช้ (USER INTERFACE) ซึ่งออกแบบมาเพื่อให้ง่ายต่อการเรียนรู้การใช้งานไม่ยุ่งยาก ชับซ้อนดังนี้



จากรูปข้างบนเป็นการแสดงหน้าจอขณะเริ่มโปรแกรมโดยเรียกว่า “2dhouse” จากไคเรกเทอร์รี่ของโปรแกรม จะเป็นแบบแปลนใหม่ที่ยังว่างอยู่ไม่ได้มีการกำหนดหรือจัดวางอะไร ซึ่งเราสามารถทำการสร้างแบบแปลนได้ตามความต้องการ หรืออาจนำแปลนเก่าที่สร้างแล้วมาทำการแก้ไขโดยใช้เมาท์คลิกที่ปุ่ม LOAD แล้วใส่ชื่อไฟล์ที่เป็นแบบแปลนที่เราจะนำมาแก้ไขกด ENTER หลังจากนั้นก็สามารถเปลี่ยนแปลงปรับปรุงสิ่งต่าง ๆ ตามที่เราต้องการโดยใช้เมาท์คลิกที่ปุ่มต่าง ๆ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- WALL ใช้เพื่อเปลี่ยนแปลงการติดกันกำแพงและกำหนดลายกำแพงตามที่ต้องการ
- FLOOR ใช้กำหนดพื้นตามลายที่ต้องการ
- CEILING ใช้กำหนดเพดานตามที่ต้องการ
- OBJECT ใช้กำหนดการวางวัตถุตามจุดที่ต้องการ
- SAVE ใช้เก็บแบบแปลนที่สร้างลงไฟล์
- LOAD ใช้นำเข้าไฟล์ซึ่งเป็นแบบแปลนที่เคยสร้างไว้มาแสดง
- CLEAR ทำการยกเลิกแบบลักษณะแปลนที่สร้างไว้
- EXIT ใช้เพื่อออกจากโปรแกรม
- SELECT แสดงลายทั้งหมดของกำแพง เพดาน พื้นวัตถุที่มีให้เลือก
- ↑↓ เพื่อเลือกลายหรือวัตถุอันถัดไป

ข้อเสนอแนะ

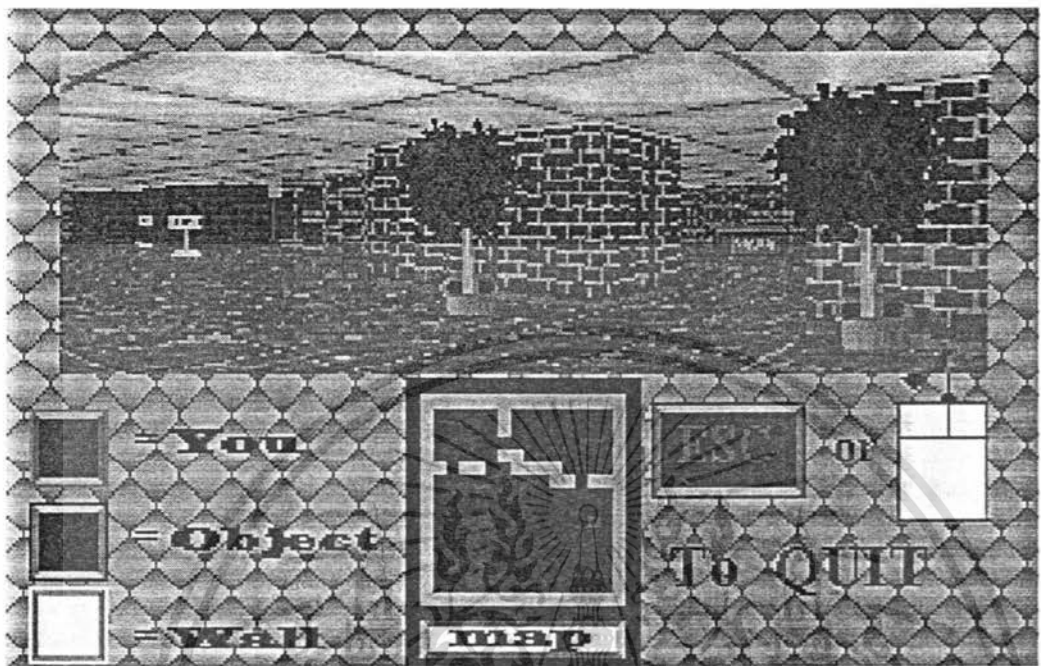
- จากโปรแกรมนี้ ยังมีความบกพร่องอยู่ในหลายจุดและควรมีการปรับปรุงดังนี้
- จำนวน ชนิด และแบบของกำแพง เฟอร์นิเจอร์ นั้นมีจำกัด ซึ่งในการใช้งานจริง ควรจะมีมากกว่านี้และควรสามารถออกแบบเพิ่มเติมเองได้ในภายหลัง
 - การสร้างภาพ 3 มิติใช้ทรัพยากรของเครื่องค่อนข้างมาก เช่น หน่วยความจำ 4 MB ขึ้นไป หากพัฒนาต่อไปควรใช้เทคนิคการจัดการหน่วยความจำ เช่น HIMEM เข้ามาช่วย
 - กำแพงมีลักษณะเป็นบล็อกใหญ่ ๆ อยู่ไม่สามารถทำกำแพงขนาดบาง ๆ ได้
 - ไม่มีประตู เพราะประตูในส่วนที่ศึกษาและสามารถทำได้เป็นไปในลักษณะของบานเลื่อน ไม่ใช่บานพับ ซึ่งในลักษณะที่เป็นบานเลื่อน จะทำให้ดูไม่สมจริง
 - สามารถออกแบบได้กับอาคารที่เป็นชั้นเดียวไม่สามารถทำเป็นหลายชั้นได้
 - ไม่มีหน้าต่างจึงทำให้ไม่มีแหล่งกำเนิดแสงที่มาจากข้างนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



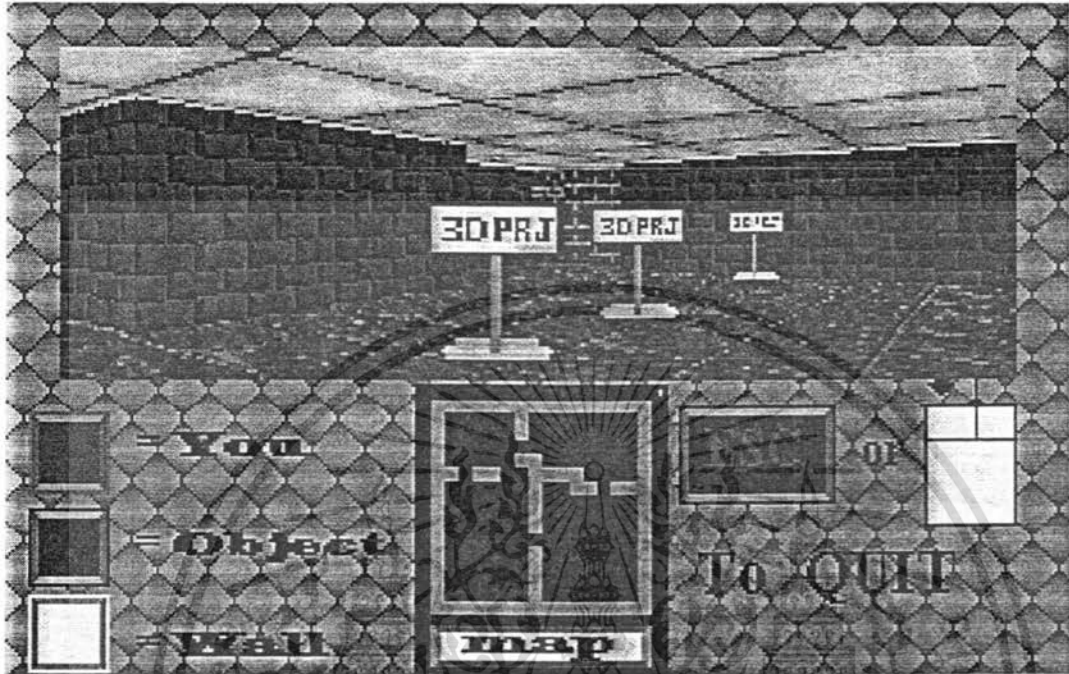
ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



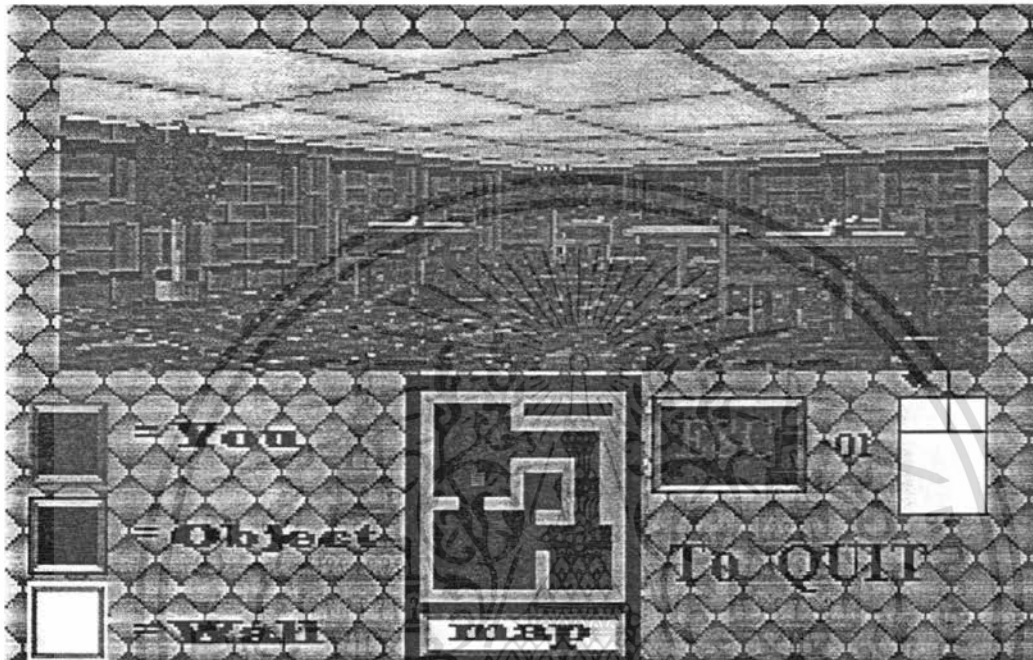
โปรแกรมช่วยออกแบบ 3 มิติ ตัวอย่างที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



โปรแกรมช่วยออกแบบ 3 มิติ ตัวอย่างที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



โปรแกรมช่วยออกแบบ 3 มิติ ตัวอย่างที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. Tom Swan Mastering Borland C++ , Sams Publishing
2. Lary L. Myers Amazing 3-D Games , Coriolis Group Books
3. Christopher Lampton Gardens of Imagination , Waite Group Press



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้