รายงาน
โครงการสิ่งประดิษฐ์

เรื่อง

ซอฟต์แวร์สร้างรูปลายเส้นสามมิติ
3 Dimension Wireframe Editor

โดย

วิวัฒน์ วัฒนาวุฒิ

จฬ
วศ 15
010161

ตุลาคม 2542

โครงการสิ่งประดิษฐ์

รายงานฉบับสมบูรณ์

ซอฟต์แวร์สร้างรูปลายเส้นสามมิติ
3 Dimension Wireframe Editor

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

โดย

วิวัฒน์ วัฒนาวุฒิ

ตุลาคม 2542

- 8 มิ.ย. 2543

I19045803

กิตติกรรมประกาศ

ผู้วิจัยขอขอบคุณฝ่ายวิจัย จุฬาลงกรณ์มหาวิทยาลัยที่มีส่วนสนับสนุนในงานของผู้วิจัย โดยให้ทุนโครงการสิ่งประดิษฐ์ กองทุนรัชดาภิเษกสมโภช แก่งานสิ่งประดิษฐ์ "ซอฟต์แวร์สร้างรูปลายเส้นสามมิติ" ชิ้นนี้ ซึ่งนับได้ว่ามีส่วนช่วยให้งานวิจัยด้าน "3D Modelling" ของผู้วิจัยดำเนินไปด้วยดี

นอกจากนี้ ผู้วิจัยขอขอบคุณภาควิชาวิศวกรรมคอมพิวเตอร์ที่อนุญาตให้ใช้สถานที่และอุปกรณ์หลายอย่างที่ใช้ในงานวิจัยและการประดิษฐ์ และขอขอบคุณอาจารย์ชัชวาล วงศ์ศิริประเสริฐ และคุณทศพล ธนะทิพานนท์ และนิสิตผู้ช่วยวิจัยที่มิได้เอ่ยนามไว้ ณ ที่นี้ด้วย

ท้ายนี้ ผู้วิจัยขอขอบคุณสมาชิกในครอบครัว สำหรับกำลังใจและการสนับสนุนในทุกๆด้านมาโดยตลอด

วิวัฒน์ วัฒนาวุฒิ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อโครงการ ซอฟต์แวร์สร้างรูปลายเส้นสามมิติ
ชื่อผู้ดำเนินโครงการ วิวัฒน์ วัฒนาวุฒิ
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
จุฬาลงกรณ์มหาวิทยาลัย
เดือนและปีที่สำเร็จ ตุลาคม 2542

บทคัดย่อ

สิ่งประดิษฐ์ "ซอฟต์แวร์สร้างรูปลายเส้นสามมิติ" นี้เป็นโปรแกรมที่สามารถใช้งานในการสร้างรูปลายเส้นสามมิติ โดยใช้เทคนิคการโปรแกรมเชิงวัตถุ รูปทรงสามมิติที่สร้างขึ้นได้ถูกจัดเก็บด้วยข้อมูลพื้นฐานคือ ข้อมูลจุดและข้อมูลเส้นตรง ซอฟต์แวร์สร้างรูปลายเส้นสามมิติที่พัฒนาขึ้นสนับสนุนให้ผู้ใช้สามารถใช้คำสั่งพื้นฐานทางกราฟิกในการประมวลผลภาพลายเส้นสามมิติได้ง่าย เช่นการเคลื่อนย้ายรูป การหมุนรูป การย่อหรือขยายขนาดรูป การมองภาพแบบขยายหรือย่อ การรวมรูปลายเส้นเป็นรูปเดียวกัน การแยกรูปหลายเหลี่ยมให้เป็นเส้น เป็นต้น นอกจากนี้โปรแกรมสามารถสาธิตการแสดงผลทางจอภาพคอมพิวเตอร์ในลักษณะหลายหน้าต่าง ที่แสดงรูปทรงทั้งแบบสองมิติและสามมิติ เพื่อให้สามารถเข้าใจรูปทรงที่สร้างได้ง่าย

โปรแกรมที่ประดิษฐ์นี้ได้รับการออกแบบด้วยเทคนิคการโปรแกรมเชิงวัตถุ ทำให้สามารถเพิ่มข้อมูลพื้นฐานที่นอกเหนือจากจุดและเส้นได้อย่างเป็นระบบ จากการทดสอบพบว่า โปรแกรมนี้สามารถใช้งานได้ถูกต้อง มีประสิทธิภาพ และให้ความสะดวกต่อผู้ใช้โปรแกรม

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

Project Title	3 Dimension Wireframe Editor
Name of Inventor	Wiwat Vatanawood
	Department of Computer Engineering
	Faculty of Engineering
	Chulalongkorn University
Year	October 1999

Abstract

The "3 Dimension WireFrame Editor" program can be used to create three dimensional wireframe model. This program is developed using Object-Oriented Programming Technique. In our approach, the primitive data elements to define a wireframe object are *Point and Line*. Our wireframe editor provides a set of computer graphics primitive tools for wireframe object transformation such as translation, rotation, scaling features, zoom control, group and ungroup objects etc. Moreover, the multiple view ports are provided with 2D and 3D wireframe model drawing on computer raster monitor for the end user.

In addition, a set of new primitive data elements can be systematically added according to the advantage of using Object-Oriented Programming technique. The program was tested and it is found that the wireframe editor works correctly, efficiently and is convenient to use.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

	หน้า
กิตติกรรมประกาศ	ii
บทคัดย่อภาษาไทย	iii
บทคัดย่อภาษาอังกฤษ	iv
สารบัญ	v
รายการรูปภาพประกอบ	vi
บทที่ 1 บทนำ	
1.1 ความเป็นมาของปัญหา	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขั้นตอนการวิจัย	1
1.4 ตารางการปฏิบัติงาน	1
1.5 ประโยชน์ที่คาดว่าจะได้รับ	2
บทที่ 2 การประดิษฐ์	
2.1 หลักการเบื้องต้นในการประดิษฐ์	3
2.2 การวิเคราะห์ผังการไหลของข้อมูล	3
2.3 การออกแบบผังคลาส	4
2.4 การออกแบบแฟ้มข้อมูล	5
2.5 หลักการในการออกแบบหน้าจอใช้งาน	6
2.6 เครื่องมือพัฒนาโปรแกรม	8
2.7 การทดสอบโปรแกรม	8
บทที่ 3 โปรแกรมสร้างรูปลายเส้นสามมิติ	
3.1 การติดตั้งและระบบคอมพิวเตอร์ที่ต้องใช้	10
3.2 ขั้นตอนในการใช้งานโปรแกรม	10
3.3 ชุดคำสั่งและเครื่องมือที่ช่วยอำนวยความสะดวกต่อผู้ใช้โปรแกรม	11
3.4 ตัวอย่างการใช้งานโปรแกรม	12
บทที่ 4 สรุป	
4.1 สรุปและวิจารณ์ผลงาน	19
รายการอ้างอิง	20
ภาคผนวก ก. Method ที่พัฒนาขึ้นสำหรับคลาสเครื่องมือการจัดการจุดและเส้น	21

เลขหมู่ วฟ
วส 15
เลขทะเบียน 010161
วัน,เดือน,ปี 16 ส.ค. 43

รายการรูปภาพประกอบ

หน้า

รูปที่ 2.1 แสดงผังการไหลของข้อมูลของซอฟต์แวร์วาดภาพลายเส้นสามมิติโดยสังเขป 4
รูปที่ 2.2 ผังคลาสที่ใช้เป็นเครื่องมือในการจัดการจุดและเส้น 5
รูปที่ 2.3 ตัวอย่างแฟ้มข้อมูลที่ใช้จัดเก็บวัตถุต่างๆในซอฟต์แวร์สร้างรูปลายเส้นสามมิติ 6
รูปที่ 2.4 แสดงจอภาพหลักส่วนติดต่อผู้ใช้ 7
รูปที่ 2.5 การขยายส่วนจอภาพย่อยสามมิติให้มีขนาดเต็มจอ 8
รูปที่ 3.1 Icon ของโปรแกรมสร้างภาพลายเส้นสามมิติ 3DWFED.EXE 10
รูปที่ 3.2 ชุดคำสั่งที่มีให้ใช้งานที่เป็น Pull Down Menu ในโปรแกรม 3DWFED.EXE 11
รูปที่ 3.3 ชุดคำสั่งที่มีให้ใช้งานที่เป็น Toolbar ในโปรแกรม 3DWFED.EXE 11
รูปที่ 3.4 คำอธิบายหน้าที่ของแต่ละปุ่มใน Toolbar 12
รูปที่ 3.5 การสร้างรูปสี่เหลี่ยมใน Front View 13
รูปที่ 3.6 การสร้างรูปสี่เหลี่ยมซ้ำใน Front View และเลื่อนมาข้างๆ 13
รูปที่ 3.7 การสร้างเส้นเชื่อมใน Side View 14
รูปที่ 3.8 การสร้างเส้นเชื่อมใน Top View ให้ได้เป็น Cube 14
รูปที่ 3.9 การปรับเปลี่ยน Cube ให้เป็นแผ่นกระดานของโต๊ะ 15
รูปที่ 3.10 การปรับเปลี่ยน Cube ให้เป็นขาของโต๊ะ 16
รูปที่ 3.11 การทำ Duplication ขาของโต๊ะและวางในตำแหน่งที่ต้องการ 16
รูปที่ 3.12 วางขาโต๊ะในตำแหน่งที่ต้องการ 17
รูปที่ 3.13 ข้อมูลในแฟ้มข้อมูลของ Table.3df 17
รูปที่ 3.14 ทำการ Import โต๊ะเข้ามาเป็นตัวแรก 18
รูปที่ 3.15 ทำการ Import โต๊ะเข้าเพิ่มเป็น 3 ตัว 18

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 1
บทนำ

1.1 ความเป็นมาของปัญหา

ซอฟต์แวร์ที่ใช้ในการสร้างและปรับเปลี่ยนรูปลายเส้นสามมิติส่วนใหญ่เป็นโปรแกรมต่างประเทศ และมีขนาดใหญ่ ราคาแพง ในขณะเดียวกันถ้ามีความต้องการที่จะปรับปรุงขีดความสามารถของซอฟต์แวร์ หรือมีความต้องการในการพัฒนาให้ใช้เฉพาะกิจก็ไม่สามารถทำเองได้ ต้องรอเวอร์ชั่นใหม่ๆ ที่ออกวางจำหน่าย

ในวิชาคอมพิวเตอร์กราฟิก มีความจำเป็นสำหรับการสาธิตผลการคำนวณโยกย้าย รูปลายเส้นสามมิติให้ปรากฏในจอภาพ นิสิตอาจจะมีความต้องการในการปรับปรุง กรรมวิธีการสร้างและแสดงผลให้ดีเข้าใจได้ง่ายและมีประสิทธิภาพมากขึ้น ซอฟต์แวร์นี้จะเป็นต้นแบบให้สามารถนำไปใช้ในการปรับเปลี่ยนได้ตามต้องการ

1.2 วัตถุประสงค์ของโครงการ

เพื่อจัดทำซอฟต์แวร์เพื่อใช้สร้างรูปลายเส้นสามมิติให้นิสิตหรือผู้สนใจคอมพิวเตอร์กราฟิกสามารถนำไปใช้เป็นเครื่องมือประกอบในการสร้างรูปกราฟิกลายเส้นได้

1.3 ขั้นตอนการวิจัย

1) ออกแบบผังการไหลของข้อมูล (Data Flow Diagram) ของซอฟต์แวร์
2) ออกแบบจอภาพในการใช้งานเบื้องต้น
3) ออกแบบฐานข้อมูลในการจัดเก็บรูปลายเส้นสามมิติ
4) พัฒนาโปรแกรมส่วนการสร้างรูปลายเส้น
5) ทดสอบโปรแกรม
6) สรุปและเขียนรายงาน

1.4 ตารางการปฏิบัติงาน

เดือนที่

	ขั้นตอนการวิจัย	1	2	3	4	5	6	7	8	9	10	11	12
1	ออกแบบผังการไหลของข้อมูล	■											
2	ออกแบบฐานข้อมูล		■	■	■								
3	ออกแบบจอภาพในการใช้งาน				■	■	■						
4	พัฒนาโปรแกรม							■	■	■	■		
5	ทดสอบโปรแกรม										■	■	
6	สรุปและเขียนรายงาน											■	■

1.5 ประโยชน์ที่คาดว่าจะได้รับ

งานชิ้นนี้จะเป็นประโยชน์ต่อนิสิตหรือผู้สนใจเรียนรู้เกี่ยวกับทฤษฎีทางคอมพิวเตอร์กราฟิก ที่จะได้มีโอกาสเขียนโปรแกรมสร้างภาพลายเส้น และทำการทดลองวิธีการใหม่ได้ ซึ่งจะทำให้การเรียนรู้ได้ง่ายและสะดวกรวดเร็วมากขึ้น

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2
การประดิษฐ์

2.1 หลักการเบื้องต้นในการประดิษฐ์

การจัดทำซอฟต์แวร์สร้างรูปลายเส้นสามมิติ ทำโดยการนำเทคนิคทางคอมพิวเตอร์กราฟิกเบื้องต้นมาประยุกต์ เช่น เทคนิคการทำ Projection เพื่อให้จุดและเส้นที่กำหนดไว้ใน World Coordinate System มาแสดงบนจอภาพสองมิติ ในรูปแบบของมุมมองแบบ Top view มุมมอง Front view และมุมมอง Side view และการแสดงภาพแบบจำลองสามมิติแบบ Isomorphic เป็นต้น นอกจากนี้ยังรวมถึงการนำสมการทางคณิตศาสตร์เบื้องต้นที่ใช้ในการทำ Transformation ของจุดและเส้นจากตำรา [1] มาประยุกต์และสาธิต เช่นการทำ Translation การทำ Rotation และการทำ Scaling เป็นต้น เพื่อช่วยให้เกิดความเข้าใจได้ดีขึ้นในการเรียนวิชา Computer Graphics

แนวความคิดหลักในการประดิษฐ์ คือการที่ซอฟต์แวร์สร้างรูปลายเส้นสามมิติ สามารถใช้งานเพื่อสาธิตเทคนิคพื้นฐานทางคอมพิวเตอร์กราฟิก กล่าวคือผู้ใช้งานจะต้องสามารถเรียนรู้และเข้าใจในรูปแบบการจัดเก็บจุดและเส้นที่เป็นองค์ประกอบพื้นฐานของรูปทรงลายเส้นของวัตถุได้ง่าย และสามารถทดลองปรับเปลี่ยนค่าตัวแปรต่าง ๆได้

ขั้นตอนในการพัฒนาซอฟต์แวร์ที่จะกล่าวในบทนี้คือ การวิเคราะห์โดยใช้ผังการไหลของข้อมูล (Data Flow Diagram) และการออกแบบโดยใช้ผังคลาส (Class Diagram) ซึ่งเป็นการออกแบบเพื่อพัฒนาโปรแกรมโดยใช้เทคนิคการโปรแกรมเชิงวัตถุ (Object-Oriented Programming) โดยสังเขป

2.2 การวิเคราะห์ผังการไหลของข้อมูล

ผู้วิจัยนำเสนอผลการวิเคราะห์ระดับภาพรวม โดยใช้ผังการไหลของข้อมูลระดับที่ 1 (Data Flow Diagram Level 1) จากรูปที่ 2.1 แสดงให้เห็นการระบบย่อยที่มีอยู่ในซอฟต์แวร์ โดยแต่ละระบบย่อยจะมีหน้าที่ในการประมวลข้อมูลที่ได้รับมาและส่งข้อมูลผลลัพธ์ต่อเพื่อให้ระบบย่อยที่เกี่ยวข้อง ผู้ใช้ หรือจัดเก็บข้อมูลในแฟ้มข้อมูลต่อไป ดังนี้

ระบบย่อยที่ 1.0 *Process 3D Editing*

ในส่วนระบบย่อยนี้จะทำหน้าที่ในการสร้างรูปลายเส้นต่าง ๆ ที่ต้องการใช้ในงานการวาดรูปลายเส้นใน World Coordinate System โดยการสร้างรูปลายเส้นที่มีใหม่ สามารถนำส่วนที่เป็นรูปลายเส้นที่มีอยู่แล้วมาเพิ่มเติมและปรับแก้ไขได้ โดยไม่ต้องเริ่มทำใหม่ทุกครั้ง ดังนั้นจะเห็นได้ว่าจะมีรูปลายเส้นสามมิติแบบ Primitive ที่ได้ทำการจัดทำไว้แล้วสามารถถูกนำกลับมาใช้ใหม่ได้อีก และเมื่อสร้างรูปลายเส้นตามต้องการแล้วสามารถจัดเก็บไว้ในเพื่อใช้งานต่อไป ใน 3D Model File

ระบบย่อยที่ 2.0 Process 3D Transforming

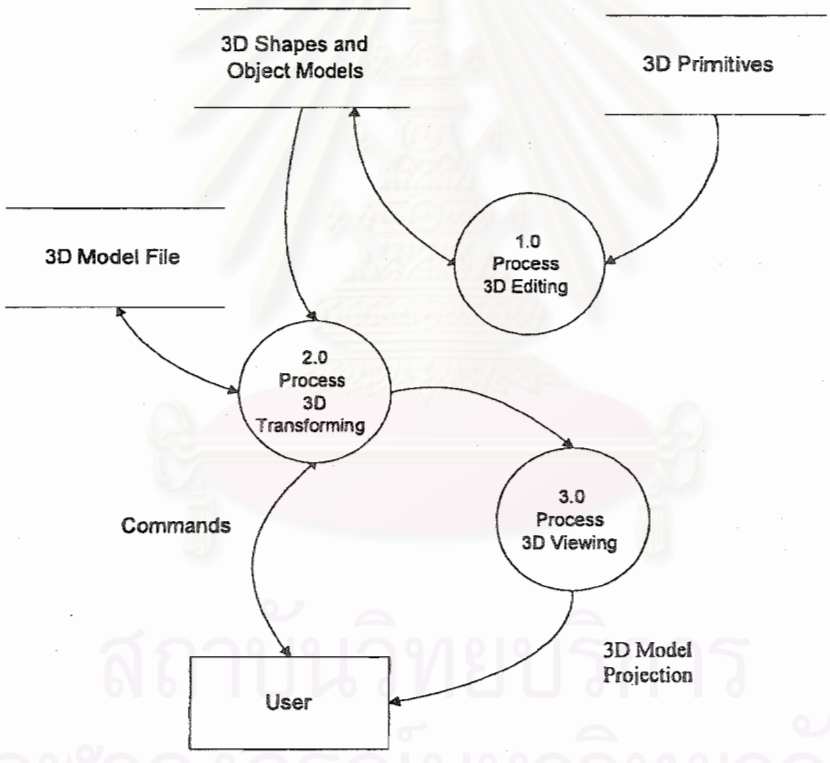
ในส่วนระบบย่อยนี้จะทำหน้าที่ในการนำรูปลายเส้นที่ต้องการโดยที่เลือกมาจากฐานข้อมูล 3D shapes and Object Models เพื่อนำมาเข้าสู่ World Coordinate System อีกครั้ง โดยผู้ใช้สามารถสร้างแบบจำลองใด ๆ

จากรูปลายเส้นที่มีอยู่ เช่นการสร้างแบบจำลองจอง วัตถุโต๊ะ หรือเก้าอี้ เป็นต้น มีขีดความสามารถในการทำกระบวนการ Transformation คือ การทำ Rotate การ Translate การทำ Scale กับวัตถุใดๆ หรือกลุ่มวัตถุใดๆ เมื่อได้องค์ประกอบแบบจำลองแล้วจึงนำไปจัดเก็บใน 3D Model File ต่อไป

ระบบย่อยที่ 3.0 *Process 3D Viewing*

ในส่วนระบบย่อยนี้จะทำหน้าที่ในการอำนวยความสะดวกในการมองภาพแบบจำลองวัตถุ โดยสามารถทำการ Map รูปลายเส้นใน World Coordinate System มาสู่ Screen Coordinate System บนจอภาพคอมพิวเตอร์ โดยใช้ View Port Mapping Technique ซึ่งเราสามารถกำหนด View Port ให้มีขนาดใดตามต้องการและสามารถทำการ Zoom in หรือ Zoom out ได้ถ้าต้องการ

ในส่วนของแบบจำลองที่เคลื่อนที่ออกนอก View Port จะถูกใช้เทคนิค 3D Clipping ตัดส่วนที่ไม่แสดงผลบนจอภาพออกไปได้



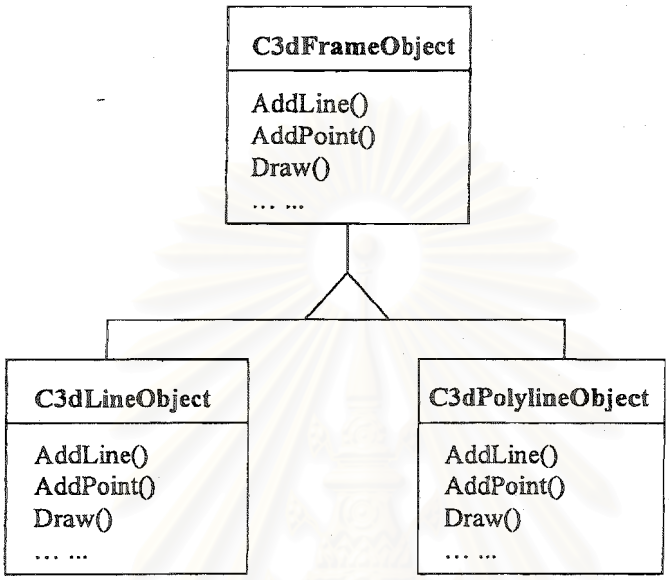
รูปที่ 2.1 แสดงผังการไหลของข้อมูลของซอฟต์แวร์วาดภาพลายเส้นสามมิติโดยสังเขป

2.3 การออกแบบผังคลาส

ผู้วิจัยได้ใช้เทคนิคการพัฒนาแบบเชิงวัตถุ โดยแบ่งส่วนของคลาสเป็น 2 กลุ่ม คือกลุ่มคลาสที่ใช้เป็นเครื่องมือในการจัดการจุดและเส้น และกลุ่มคลาสที่ใช้เป็นเครื่องมือในการสร้างส่วนเชื่อมต่อผู้ใช้ (User Interface)

พระมหากรุณาธิคุณ
ของจุฬาลงกรณ์มหาวิทยาลัย

โดยงานพัฒนาส่วนใหญ่จะเน้นในกลุ่มคลาสที่เป็นเครื่องมือในการจัดการจุดและเส้นเป็นหลัก ส่วนในกลุ่มคลาสที่เหลือจะประยุกต์มาจากคลาสมาตรฐาน MFC (Microsoft Foundation Class) [2] ทำให้สามารถปรับเปลี่ยนได้ง่ายตาม Version ของโปรแกรมแปลภาษารุ่นใหม่ในอนาคต รูปที่ 2.2 แสดงผังคลาสในกลุ่มเครื่องมือในการจัดการจุดและเส้น



รูปที่ 2.2 ผังคลาสที่ใช้เป็นเครื่องมือในการจัดการจุดและเส้น

จากรูปที่ 2.2 คลาสชื่อ C3dFrameObject เป็นคลาสหลักที่กำหนดกรอบของ Method หรือ Function ที่มีใช้งานได้ โดยจะปรากฏคลาสย่อย (Subclass) ที่จะใช้กรอบการทำงานคลาสหลัก คือ คลาสย่อยชื่อ C3dLineObject จะทำหน้าที่ในการกำหนดเส้นเดี่ยวหรือจุดที่สร้างขึ้นเป็นวัตถุ (Object) หนึ่งๆ และมี Method ที่เป็นเครื่องมือพื้นฐาน ในขณะที่คลาสย่อยชื่อ C3dPolylineObject จะทำหน้าที่ในการกำหนดเส้น Polyline ซึ่งในทางปฏิบัติใช้ในการสร้างรูปทรงหลายเหลี่ยม (Polygon) ได้และ Method ที่เป็นเครื่องมือพื้นฐานทางคอมพิวเตอร์กราฟิก เช่นกัน (ดูรายละเอียดของ Method ของกลุ่มคลาสที่แสดงในรูปที่ 2.2 ได้ในภาคผนวก ก.)

2.4 การออกแบบแฟ้มข้อมูล

แฟ้มข้อมูลที่ออกแบบเพื่อใช้จัดเก็บ Persistent Object ที่สร้างขึ้นมานั้น ผู้วิจัยได้เน้นให้สามารถเข้าใจได้ง่าย และสามารถแก้ไขได้ง่ายเช่นเดียวกัน ลักษณะของแฟ้มข้อมูลได้รับการออกแบบโดยใช้ Formatted Flat File กล่าวคือ เป็นแฟ้มข้อมูลที่มีคำสำคัญ ในการกำกับขอบเขตของข้อมูลจริง และเป็นแฟ้มข้อมูลแบบ Text File ที่สามารถอ่านเข้าใจได้ง่าย และแก้ไขได้ง่ายโดยใช้โปรแกรม Notepad ธรรมดา

รูปที่ 2.3 แสดงข้อมูลที่ใช้ในแฟ้มข้อมูลและความหมายในแต่ละบรรทัด แฟ้มข้อมูลดังกล่าวสามารถได้รับการแก้ไขได้ไม่ยากนัก เพื่อให้ผู้ใช้สามารถสร้างวัตถุ Primitive ด้วยมือได้เองโดยที่ไม่ต้องใช้โปรแกรมสร้างรูปลายเส้นสามมิติ และผู้ใช้อาจนำไปใช้งานในระบบอื่นๆได้เช่นกัน

ผู้ใช้สามารถออกแบบวัตถุใหม่ๆ ขึ้นมาได้โดยเพิ่ม ประเภทวัตถุเข้าในระบบได้ โดยใช้คำสำคัญ BEGIN และ END เป็นขอบเขตของข้อมูลวัตถุนั้น และเพิ่มคลาสย่อยที่จัดการวัตถุใหม่เข้าสู่ระบบได้ง่าย และเป็นระบบมากที่สุด

Line#	ข้อมูลในแฟ้ม	ความหมาย
1	; 3D Wireframe Editor Version 1.00 File Format	Comment
2	3DWFED 100	Signature Line
3	OBJECT 2	ระบุว่ามี 2 วัตถุในแฟ้ม
4	BEGIN LINE	เริ่มวัตถุประเภท Line
5	0 0 0	ค่า x,y,z ของจุดเริ่ม
6	100 100 100	ค่า x,y,z ของจุดสิ้นสุด
7	END LINE	จบข้อมูลวัตถุ Line
8	BEGIN POLYLINE	เริ่มวัตถุประเภท Polyline
9	POINT 4	ระบุว่ามี 4 จุดในวัตถุนี้
10	40 40 –40	ค่า x,y,z ของจุดที่ 0
11	0 30 –30	ค่า x,y,z ของจุดที่ 1
12	20 20 20	ค่า x,y,z ของจุดที่ 2
13	30 30 0	ค่า x,y,z ของจุดที่ 3
14	LINE 5	ระบุว่ามี 5 เส้นเชื่อมระหว่างจุด
15	0 1	มีเส้นจากจุดที่ 0 ถึงจุดที่ 1
16	3 2	มีเส้นจากจุดที่ 3 ถึงจุดที่ 2
17	2 3	มีเส้นจากจุดที่ 2 ถึงจุดที่ 3
18	2 1	มีเส้นจากจุดที่ 2 ถึงจุดที่ 1
19	0 3	มีเส้นจากจุดที่ 0 ถึงจุดที่ 3
20	END POLYLINE	จบข้อมูลวัตถุ Polyline

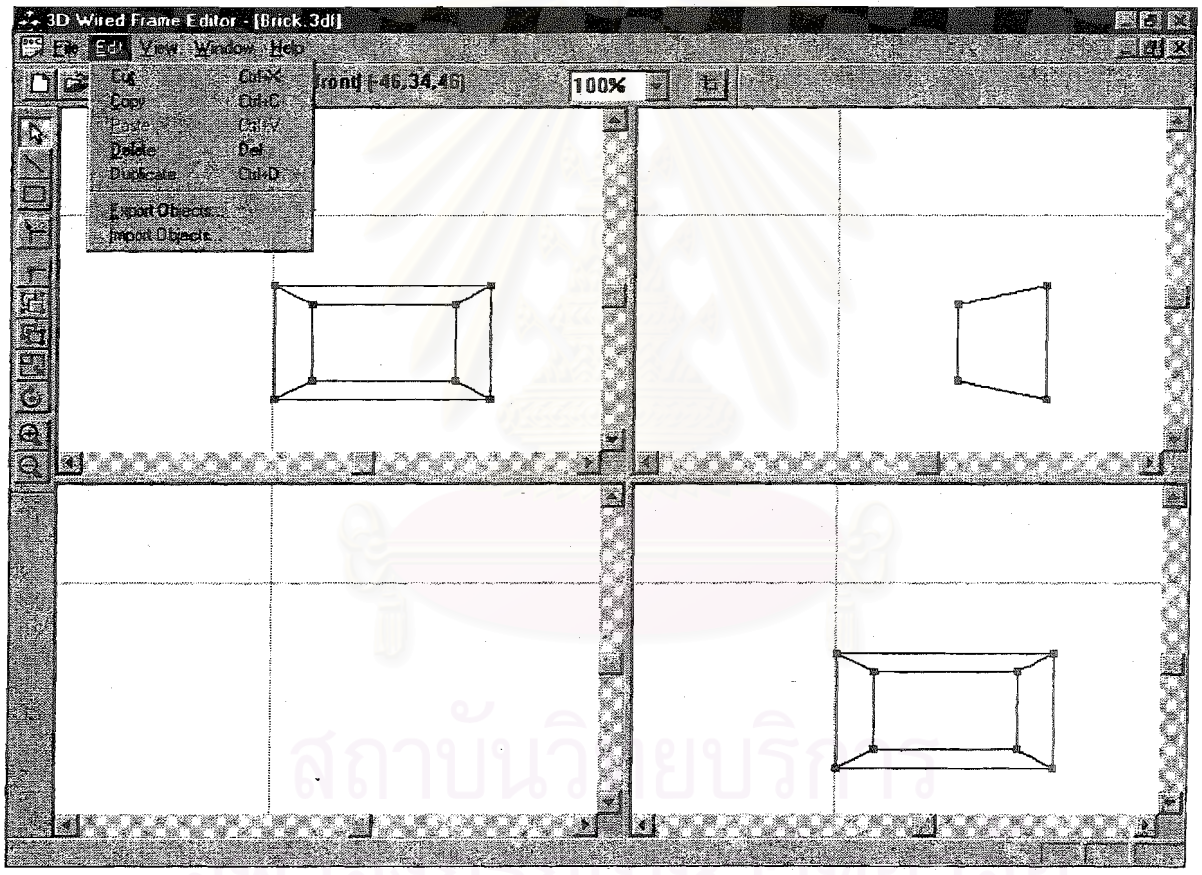
รูปที่ 2.3 ตัวอย่างแฟ้มข้อมูลที่ใช้จัดเก็บวัตถุต่างๆในซอฟต์แวร์สร้างรูปลายเส้นสามมิติ

2.5 หลักการในการออกแบบหน้าจอใช้งาน

การออกแบบหน้าจอในการใช้งาน ผู้วิจัยจอภาพหลักที่ใช้ในการสร้างและปรับปรุงรูปลายเส้นสามมิติ และการทำ Transformation ของรูปลายเส้นสามมิติ โดยที่ จอภาพหลักแบ่งออกเป็น 2 ส่วนหลัก (ดูรูปที่ 2.4) ดังนี้

1. ส่วนบริเวณแถบคำสั่ง (Command Bar Area)
   โดยที่จะเป็นบริเวณที่ใช้จัดวางปุ่มคำสั่ง หรือแถบตัวเลือกคำสั่งแบบ List Box หรือ Pull down เป็นต้น เพื่อใช้ในการทำงานหลักเช่น Load หรือ Save ข้อมูล เป็นต้น

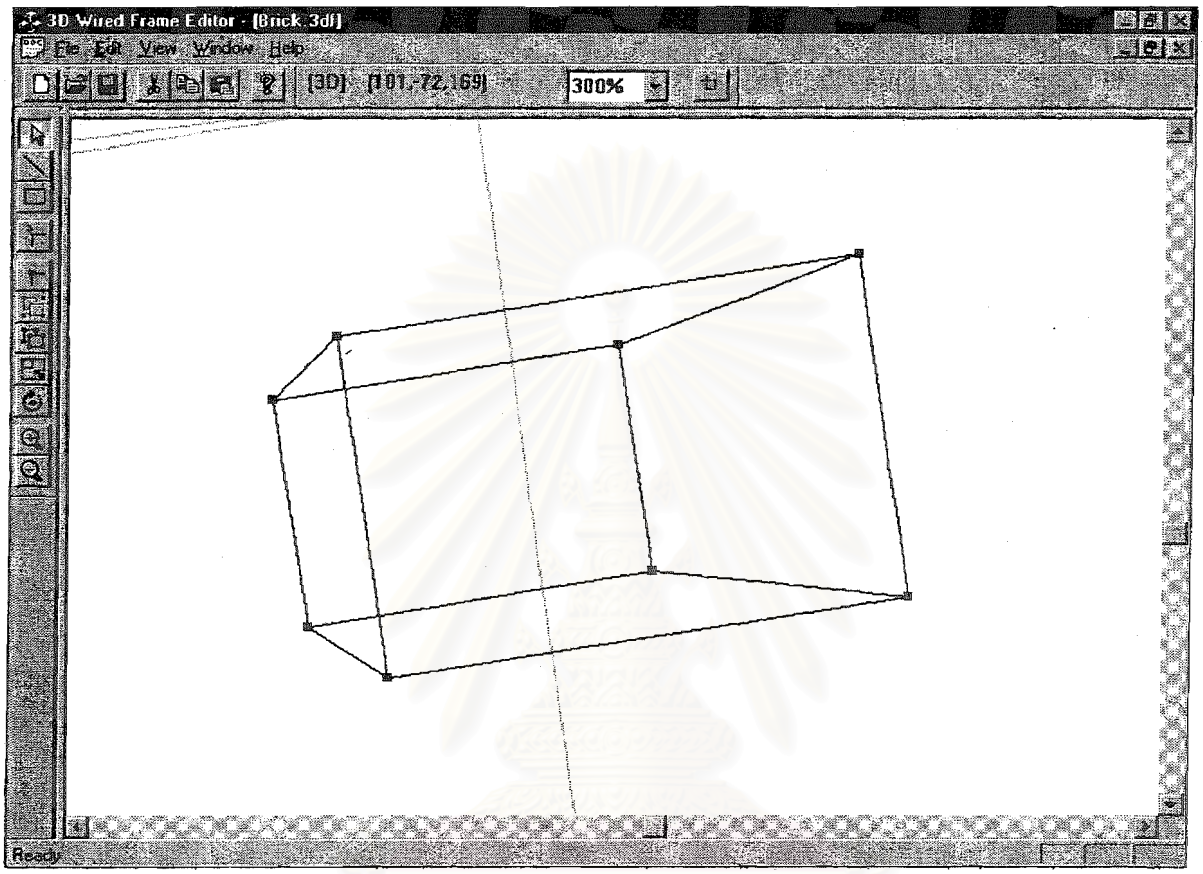
2. ส่วนบริเวณแสดงผล (View Area)
   โดยที่จะเป็นบริเวณที่ใช้แสดงผลลัพธ์ของการจัดวาง Object หรือ 3D Model และเพื่อทำ transformation และสามารถแบ่งส่วนแสดงผลออกเป็น 4 ส่วนย่อยถ้าต้องการ คือ ส่วนแสดงภาพ Front View ของรูปลายเส้นบนแกน XY ส่วนแสดงภาพ Top View ของรูปลายเส้นบนแกน XZ ส่วนแสดงภาพ Side View ของรูปลายเส้นบนแกน YZ ส่วนแสดงภาพ Isometric View ของรูปลายเส้น



รูปที่ 2.4 แสดงจอภาพหลักส่วนติดต่อผู้ใช้

จากรูปที่ 2.4 แสดงจอภาพย่อยด้านบนซ้ายคือ Front view จอภาพย่อยด้านบนขวาคือ Side view จอภาพย่อยด้านซ้ายล่างคือ Top view และจอภาพย่อยด้านขวาล่างคือ Isometric view แบบสามมิติ

ในการออกแบบเป็นจอภาพส่วนย่อย เพื่อให้ผู้ใช้สามารถเลื่อนหรือกำหนดตำแหน่งได้ในแต่ละ Plane ได้ง่ายขึ้น และผู้ใช้สามารถเลือกแสดงจอภาพย่อยส่วนใดให้ขยายใหญ่ได้ตามต้องการ เช่นในรูปที่ 2.5 แสดงการขยายส่วนจอภาพย่อยสามมิติให้มีขนาดเต็มจอ และขยายเป็น 300% ทำให้เห็นรายละเอียดได้ชัดเจนกว่าเดิมได้



รูปที่ 2.5 การขยายส่วนจอภาพย่อยสามมิติให้มีขนาดเต็มจอ

2.6 เครื่องมือพัฒนาโปรแกรม

เครื่องมือที่ใช้ในการพัฒนาคือ Microsoft Visual C++ version 5.0 และคลาสมาตรฐานที่มีอยู่ใน Microsoft Foundation Class [2]

ผู้วิจัยได้พัฒนาบนเครื่องไมโครคอมพิวเตอร์ ที่มี Microsoft Windows NT version 4.0 เป็นระบบปฏิบัติการ และมีหน่วยความจำ RAM ประมาณ 64 MB

2.7 การทดสอบโปรแกรม

ในการทดสอบโปรแกรม ผู้วิจัยได้แบ่งการทดสอบออกเป็น 3 ส่วนหลักคือ

1. การทดสอบส่วนการตอบสนองผู้ใช้
   หมายถึงการทดสอบส่วนที่ผู้ใช้ต้องมีการติดต่อกับโปรแกรม เช่นส่วนที่เป็น User Interface ส่วนที่

เป็นการคลิกจุดหรือเส้น ตลอดจนส่วนที่ทำหน้าที่ในการแสดงภาพวัตถุบนจอภาพ การย่อขยายของหน้าต่าง การรวมหรือแยกวัตถุ เป็นต้น ซึ่งส่วนใหญ่จะเป็นผลของการใช้งานคลาสมาตรฐานที่เลือกมาจาก Microsoft Foundation Class

2. การทดสอบส่วนการจัดการจุดและเส้น
หมายถึงการทดสอบผลของคำสั่งที่ผู้ใช้เลือกกระทำกับวัตถุ เช่น กลุ่มคำสั่งการเลื่อนวัตถุไปมา กลุ่มคำสั่งการหมุนวัตถุ และกลุ่มคำสั่งการย่อหรือขยายขนาดวัตถุ ตลอดจนการแสดงภาพในส่วนมุมมองแบบ Front view มุมมอง Side view และมุมมอง Top view เป็นต้น ทั้งนี้การแสดงผลสามมิติก็ได้รับการทดสอบเช่นกัน
3. การทดสอบส่วนการจัดเก็บแฟ้มข้อมูลวัตถุ
หมายถึงการทดสอบความถูกต้องของการจัดเก็บข้อมูลวัตถุต่างๆ และการตรวจสอบความถูกต้องในการสร้างวัตถุและแสดงผลวัตถุให้สอดคล้องกับข้อมูลที่มีอยู่ในแฟ้มข้อมูล อนึ่งการทดสอบจะไม่ยุ่งยากนัก เนื่องจากข้อมูลที่จัดเก็บในแฟ้มข้อมูลจะเป็น Text File ที่สามารถอ่านเข้าใจได้ง่าย

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 3
โปรแกรมสร้างรูปลายเส้นสามมิติ

3.1 การติดตั้งและระบบคอมพิวเตอร์ที่ต้องใช้

ก่อนทำการติดตั้งโปรแกรม ผู้ใช้งานต้องทำการสำรวจระบบคอมพิวเตอร์ที่ใช้ให้มีความเหมาะสมทั้งระบบปฏิบัติการและส่วนฮาร์ดแวร์ดังต่อไปนี้

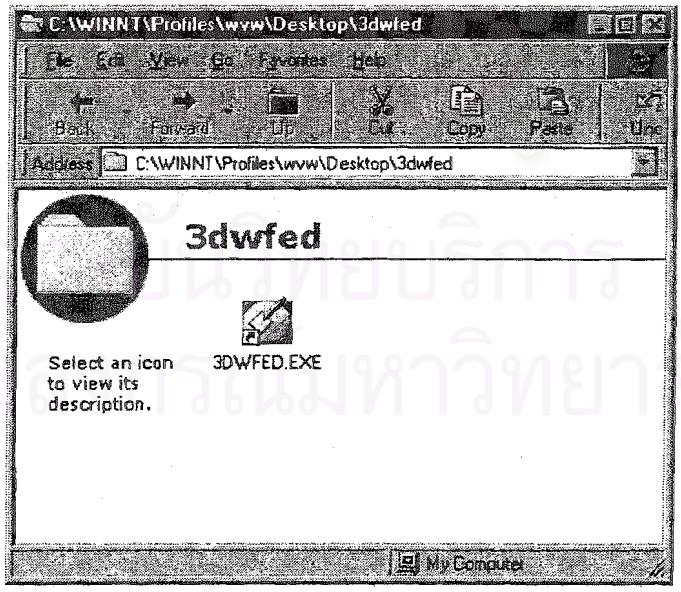
- ระบบปฏิบัติการ Microsoft Windows 98 หรือ Microsoft Windows NT 4.0
- เครื่องไมโครคอมพิวเตอร์ Pentium หรือ Compatible มีหน่วยความจำอย่างน้อย 64 MB และมีเนื้อที่ว่างในฮาร์ดดิส์กอย่างน้อย 1 MB สำหรับโปรแกรม 3DWFED.EXE พร้อมเนื้อที่สำหรับแฟ้มข้อมูลวัตถุ

การติดตั้งโปรแกรมมีขั้นตอนดังต่อไปนี้

1. สร้าง Subdirectory ชื่อ 3DWFED ในฮาร์ดดิส์กที่ต้องการ เช่น subdirectory ชื่อ C:\3DWFED
2. ทำการ Copy โปรแกรมชื่อ 3DWFED.EXE ใน Subdirectory ดังกล่าวเท่านั้น เนื่องจากโปรแกรมได้ออกแบบให้ทำงานอย่างง่าย
3. สร้าง Shortcut Icon ให้ชี้ไปที่โปรแกรม (ดูรายละเอียดการสร้าง Shortcut ได้ในคู่มือการใช้งาน Microsoft Windows ทั่วไป)

3.2 ขั้นตอนในการใช้งานโปรแกรม

ในการใช้งานโปรแกรม 3DWFED.EXE ให้เข้าสู่ Subdirectory ที่ติดตั้งโปรแกรมไว้แล้วเรียกชื่อโปรแกรมหรือคลิกที่ Icon ของโปรแกรม ดังรูปที่ 3.1

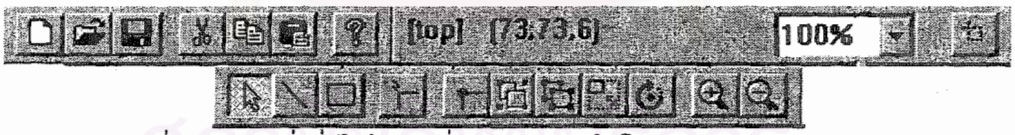


รูปที่ 3.1 Icon ของโปรแกรมสร้างภาพลายเส้นสามมิติ 3DWFED.EXE

3.3 ชุดคำสั่งและเครื่องมือที่ช่วยอำนวยความสะดวกต่อผู้ใช้โปรแกรม

ชุดคำสั่งที่มีให้ใช้งานปรากฏอยู่ในส่วนที่เป็น Pull Down Menu (ดูรูปที่ 3.2) และส่วนที่เป็น Toolbar (ดูรูปที่ 3.3)

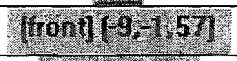
รูปที่ 3.2 ชุดคำสั่งที่มีให้ใช้งานที่เป็น Pull Down Menu ในโปรแกรม 3DWFED.EXE



รูปที่ 3.3 ชุดคำสั่งที่มีให้ใช้งานที่เป็น Toolbar ในโปรแกรม 3DWFED.EXE

ในรูปที่ 3.2 แสดงชุดคำสั่งหลักในกลุ่มของ File กลุ่มคำสั่งของ Edit กลุ่มคำสั่ง View และกลุ่มคำสั่ง Window และในรูปที่ 3.3 แสดงปุ่มคำสั่งที่ผู้ใช้สามารถเรียกใช้งานได้ง่ายในการคลิกครั้งเดียว

รูปที่ 3.4 แสดงคำอธิบายหน้าที่ของแต่ละปุ่มในรูปที่ 3.3 ให้โดยสังเขป และให้ดูตัวอย่างการใช้งานโปรแกรมในส่วนต่อไป

Icon Symbols	*Description*
	File | Open | Save
	Cut | Copy | Paste
	Help
	View Name | Coordinate (X,Y,Z)
	Zoom Control (%)
	Pointer | Line | Rectangle
	Break Line
	Join | Ungroup | Group | Scale | Rotate
	Zoom In | Zoom Out

รูปที่ 3.4 คำอธิบายหน้าที่ของแต่ละปุ่มใน Toolbar

3.4 ตัวอย่างการใช้งานโปรแกรม

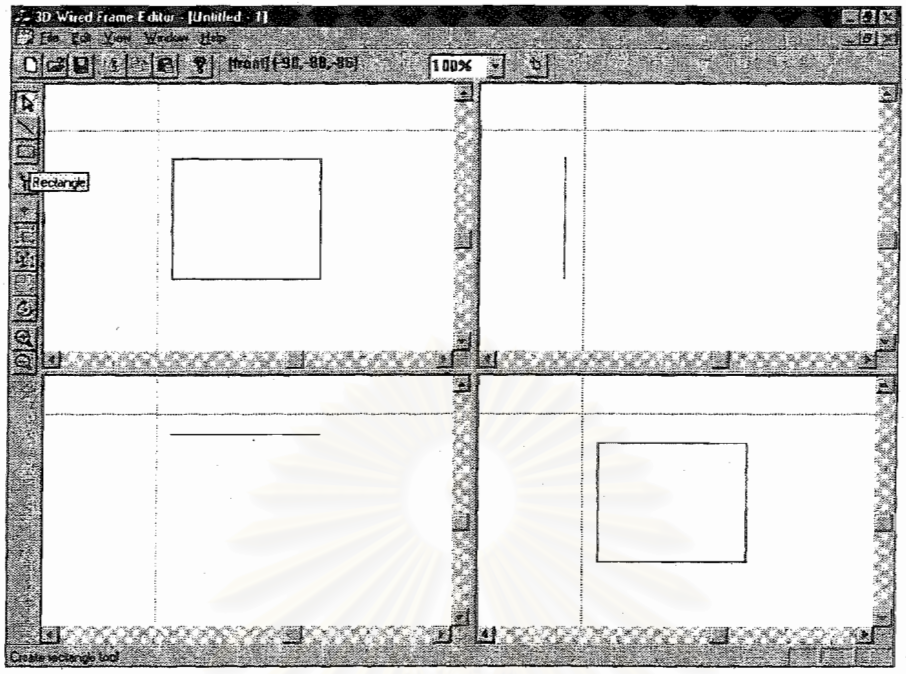
เพื่อสาธิตให้เห็นการใช้งานโปรแกรม ผู้วิจัยขอ ยกตัวอย่างการใช้โปรแกรมไว้โดยสังเขปตามรูปผังการไหลของข้อมูลที่ 2.1 คือ

- การสร้างรูปสี่เหลี่ยมลูกบาศ์ก เพื่อจัดเก็บไว้เป็น 3D Primitive Object
- การสร้างโต๊ะจากรูปสี่เหลี่ยมลูกบาศ์ก เพื่อจัดเก็บเป็น 3D Shapes and Object Model
- การสร้างบริเวณที่มีหลายโต๊ะ เพื่อจัดเก็บเป็น 3D Model Files

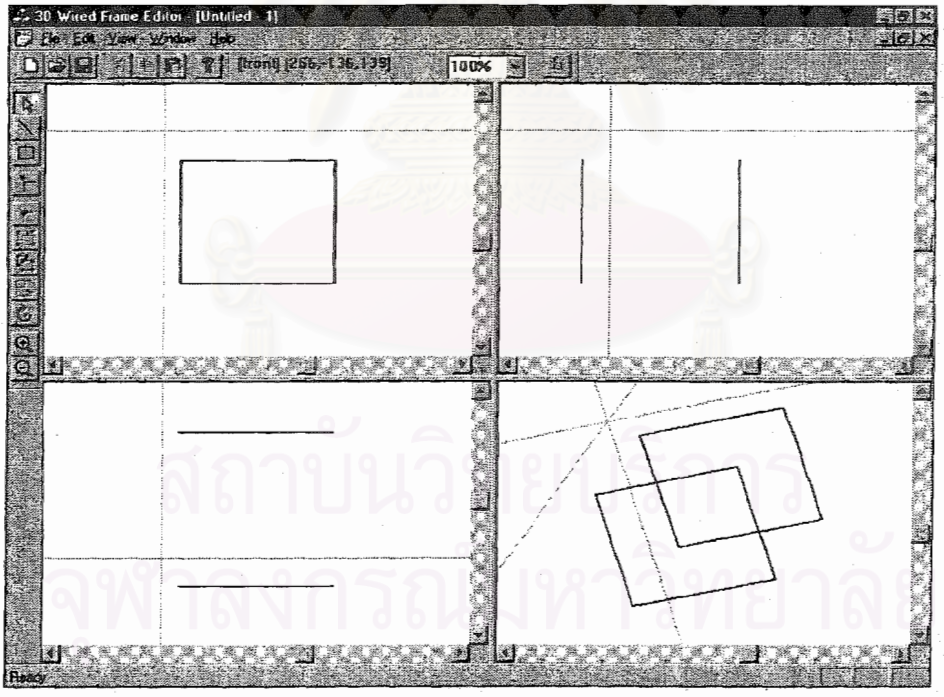
อย่างไรก็ตาม เนื่องจากในทางปฏิบัติโครงสร้างแฟ้มข้อมูลของทั้งสามชนิดจะเป็นรูปแบบเดียวกัน โดยแฟ้มข้อมูลที่อธิบายไว้ในรูปที่ 2.3 ผู้ใช้สามารถกำหนดระดับของวัตถุไว้ให้เหมาะสมกับการทำงานตามต้องการได้

การสร้างรูปสี่เหลี่ยมลูกบาศ์ก (3D Primitive Object)

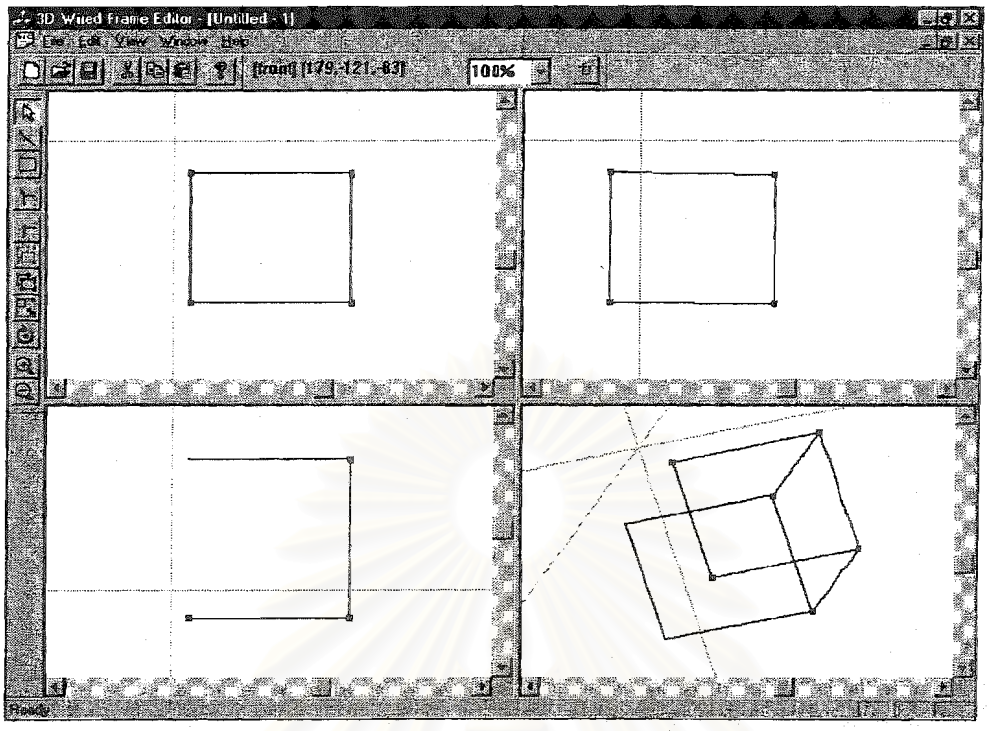
1. ใช้เครื่องมือ Rectangle ในการสร้างรูปสี่เหลี่ยมใน Front View (ดูรูปที่ 3.5)
2. คลิกเลือกรูปสี่เหลี่ยมจากข้อ 1 ใน Front View แล้วสั่ง Copy and Paste รูปทั้งสองจะซ้อนทับกันอยู่ โดยที่จะเห็นได้ยาก (ดูรูปที่ 3.6)
3. เลือกรูปสี่เหลี่ยมจาก Side View โดยการคลิกเลือกและเลื่อนไปทางขวาให้ห่างออกจากรูปแรก ใน Side View ผู้ใช้จะเห็นรูปสี่เหลี่ยมที่เลื่อนนั้นเป็นเส้นตรง ให้สังเกตจอภาพย่อยสามมิติแทน
4. เลือกเครื่องมือสร้างเส้นตรง และสร้างเส้นตรงสองเส้นเชื่อมระหว่างรูปสี่เหลี่ยมใน Side View ปรับตำแหน่งของเส้นให้ตรงกับจุดปลายของรูปสี่เหลี่ยม โดยใช้ Top View ช่วย (ดูรูปที่ 3.7)
5. เลือกเครื่องมือสร้างเส้นตรง และสร้างเส้นตรงสองเส้นเชื่อมระหว่างรูปสี่เหลี่ยมใน Top View ปรับตำแหน่งของเส้นให้ตรงกับจุดปลายของรูปสี่เหลี่ยม โดยใช้ Side View ช่วย (ดูรูปที่ 3.8)
6. จัดเก็บวัตถุไว้ในชื่อ Cube.3df เพื่อนำไปใช้ต่อไป



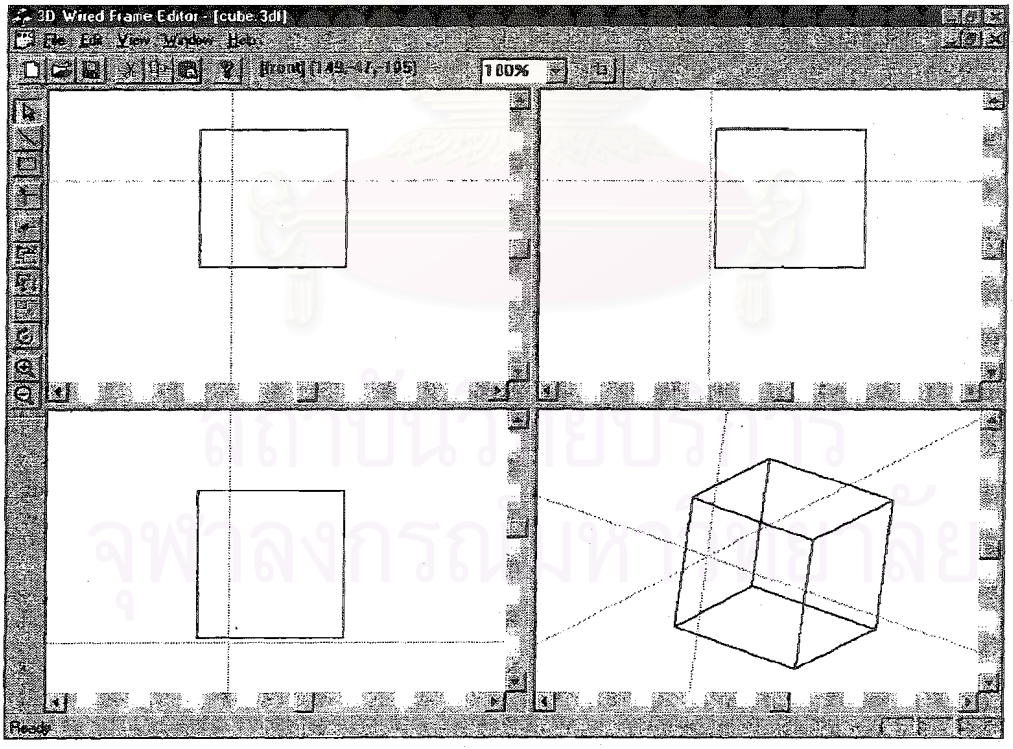
รูปที่ 3.5 การสร้างรูปสี่เหลี่ยมใน Front View



รูปที่ 3.6 การสร้างรูปสี่เหลี่ยมซ้ำใน Front View และเลื่อนมาข้างๆ



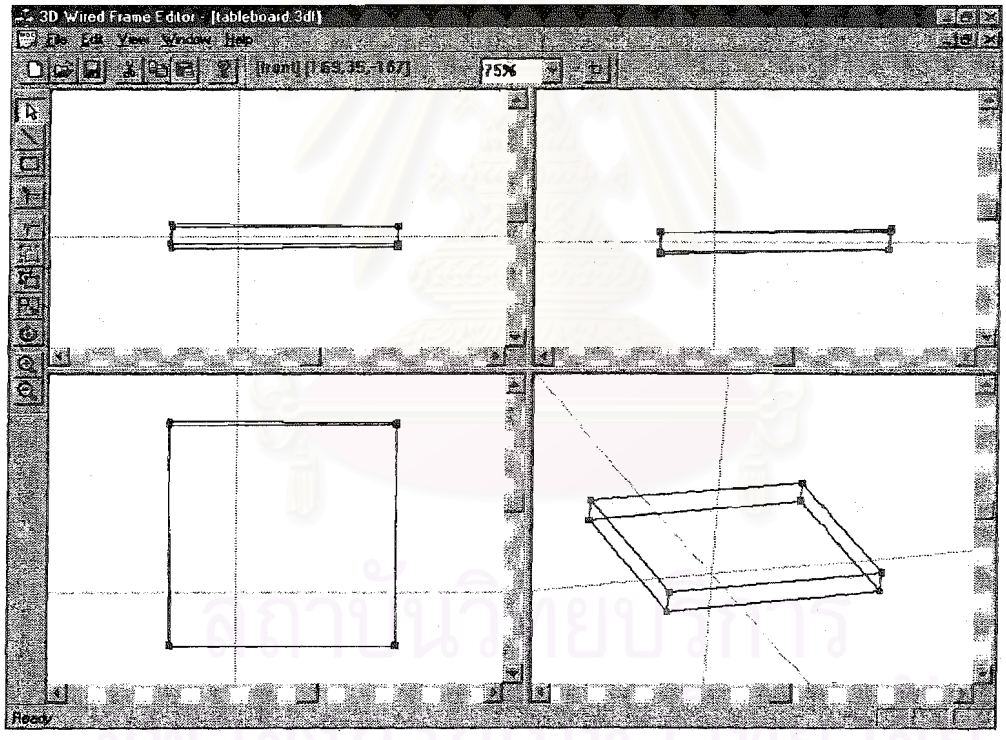
รูปที่ 3.7 การสร้างเส้นเชื่อมใน Side View



รูปที่ 3.8 การสร้างเส้นเชื่อมใน Top View ให้ได้เป็น Cube

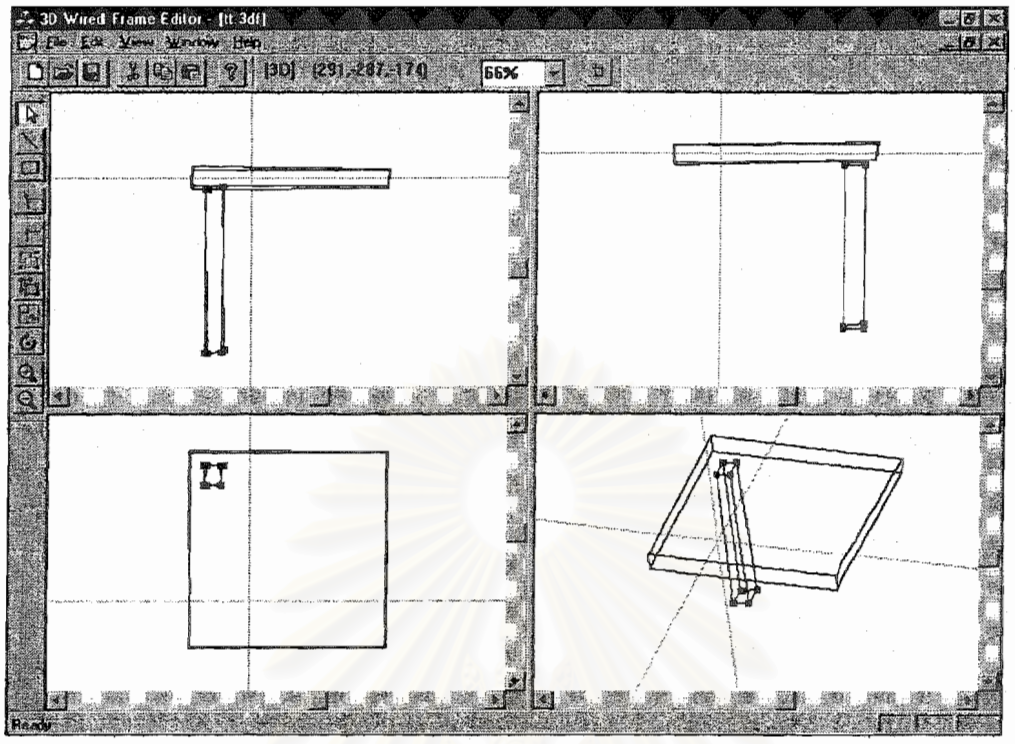
การสร้างโต๊ะจากรูปสี่เหลี่ยมลูกบาศ์ก (3D Shapes and Object Models)

1. เปิดแฟ้มใหม่
2. ใช้คำสั่ง Import Object และเลือก Cube.3df ที่สร้างไว้แล้วมา
3. ใช้เครื่องมือ Pointer ในการเลื่อนตำแหน่งให้จุดแต่ละจุดเปลี่ยนไปทำให้เป็นแผ่นกระดานโต๊ะ (ดูรูปที่ 3.9)
4. ใช้คำสั่ง Import Object และเลือก Cube.3df ที่สร้างไว้แล้วมาอีก
5. ใช้เครื่องมือ Pointer ในการเลื่อนตำแหน่งให้รูปสี่เหลี่ยมเป็นขาโต๊ะ (ดูรูปที่ 3.10)
6. คลิกเลือกขาโต๊ะ และทำการ Duplicate 3 ครั้งเพื่อสร้างขาโต๊ะเพิ่มให้ครบ 4 ขา (ดูรูปที่ 3.11)
7. เลื่อนขาโต๊ะแต่ละขาเข้ามาประกอบกับแผ่นกระดานโต๊ะ (ดูรูปที่ 3.12)
8. จัดเก็บในแฟ้ม Table.3df (ดูรูปที่ 3.13)

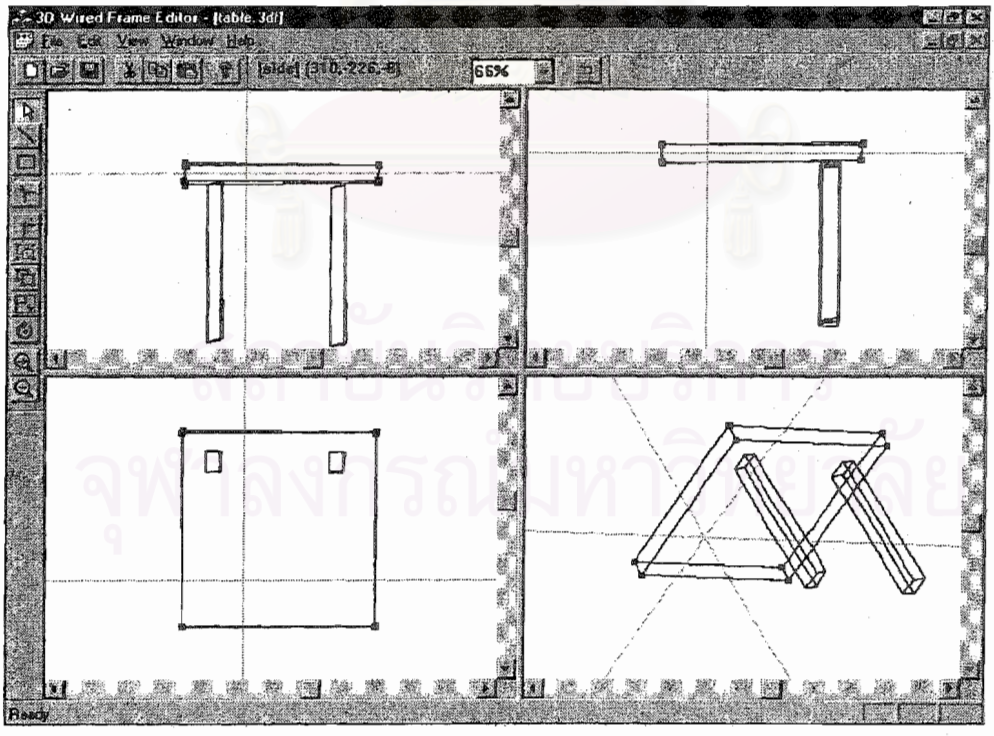


รูปที่ 3.9 การปรับเปลี่ยน Cube ให้เป็นแผ่นกระดานของโต๊ะ

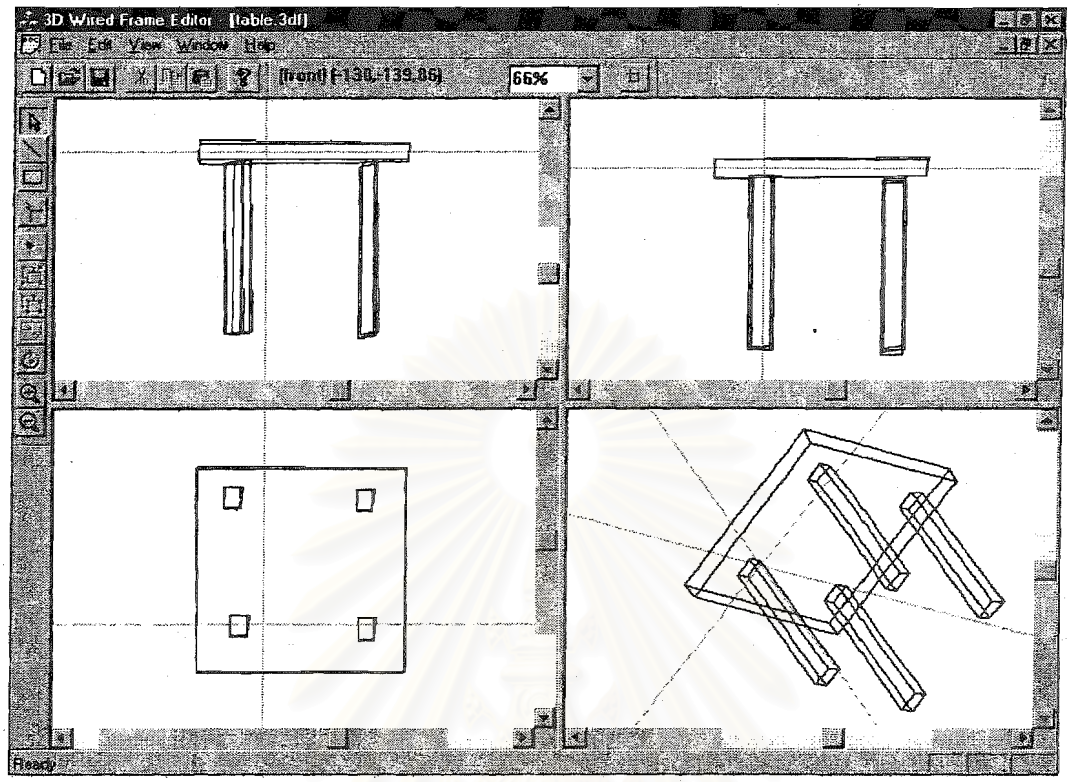
I19045803



รูปที่ 3.10 การปรับเปลี่ยน Cube ให้เป็นขาของโต๊ะ



รูปที่ 3.11 การทำ Duplication ขาของโต๊ะและวางในตำแหน่งที่ต้องการ



รูปที่ 3.12 วางขาโต๊ะในตำแหน่งที่ต้องการ

table.3df - Notepad

```
3DWFED 100
OBJECT 5
BEGIN POLYLINE
POINT 8
167 -9 186
168 11 188
-75 14 189
-76 -9 186
167 -12 -59
168 10 -58
-74 10 -59
-75 -14 -58
LINE 12
0 1
1 2
2 3
3 0
4 5
5 6
6 7
7 4
6 2
5 1
4 0
7 3
END POLYLINE
```

```
BEGIN POLYLINE
POINT 8
128 -216 161
125 -15 161
107 -10 161
108 -221 161
125 -219 136
126 -15 135
107 -10 136
107 -221 135
LINE 12
0 1
1 2
2 3
3 0
4 5
5 6
6 7
7 4
6 2
5 1
4 0
7 3
END POLYLINE
```

```
BEGIN POLYLINE
POINT 8
-24 -211 164
-27 -10 164
-45 -13 164
-44 -216 164
-27 -214 139
-26 -10 138
-45 -13 139
-45 -216 138
LINE 12
0 1
1 2
2 3
3 0
4 5
5 6
6 7
7 4
6 2
5 1
4 0
7 3
END POLYLINE
```

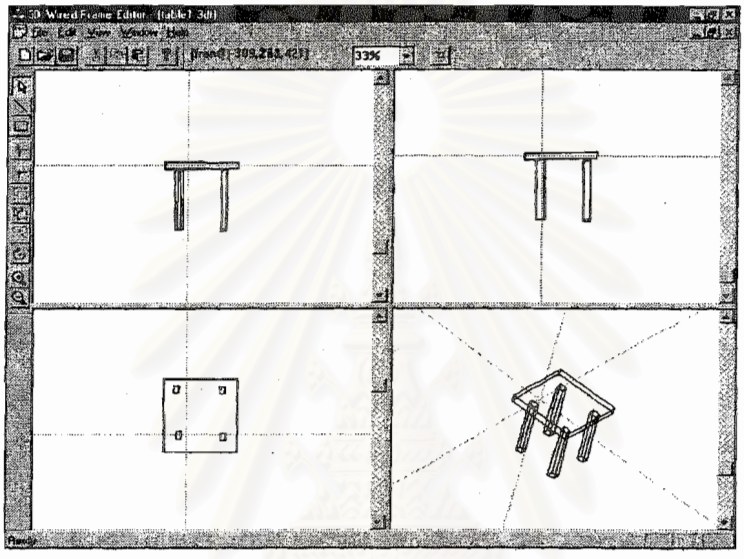
```
BEGIN POLYLINE
POINT 8
131 -211 7
128 -10 7
110 -13 7
111 -216 7
128 -214 -10
129 -10 -19
110 -13 -10
110 -216 -19
LINE 12
0 1
1 2
2 3
3 0
4 5
5 6
6 7
7 4
6 2
5 1
4 0
7 3
END POLYLINE
```

```
BEGIN POLYLINE
POINT 8
-16 -211 11
-19 -10 11
-37 -13 11
-36 -216 11
-19 -214 -14
-18 -10 -15
-37 -13 -14
-37 -216 -15
LINE 12
0 1
1 2
2 3
3 0
4 5
5 6
6 7
7 4
6 2
5 1
4 0
7 3
END POLYLINE
```

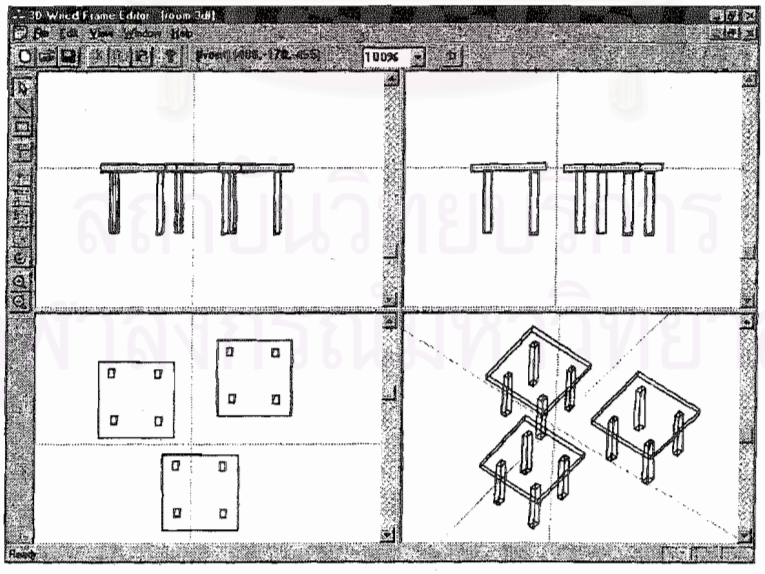
รูปที่ 3.13 ข้อมูลในแฟ้มข้อมูลของ Table.3df

การสร้างบริเวณที่มีหลายโต๊ะ

1. สร้างแฟ้มใหม่
2. ใช้คำสั่ง Import Object และเลือก Table.3df ที่สร้างไว้แล้วมา
3. ใช้เครื่องมือ Pointer ในการเลื่อนตำแหน่งโต๊ะตามต้องการ (ดูรูปที่ 3.14)
4. ใช้คำสั่ง Import Object และเลือก Table.3df ที่สร้างไว้แล้วมาอีก 2 รูป
5. ใช้เครื่องมือ Pointer ในการเลื่อนตำแหน่งโต๊ะตามต้องการ (ดูรูปที่ 3.15)
6. จัดเก็บในแฟ้ม Table.3df



รูปที่ 3.14 ทำการ Import โต๊ะเข้ามาเป็นตัวแรก



รูปที่ 3.15 ทำการ Import โต๊ะเข้าเพิ่มเป็น 3 ตัว

บทที่ 4
สรุป

4.1 สรุปและวิจารณ์ผลงาน

ผู้วิจัยได้วิเคราะห์และออกแบบซอฟต์แวร์สร้างภาพลายเส้นสามมิติ และพัฒนาโปรแกรมให้สามารถใช้งานในการสร้างรูปได้จริง โดยใช้เทคนิคการโปรแกรมเชิงวัตถุ (Object-Oriented Programming) โดยกำหนดให้มีวัตถุพื้นฐานและเครื่องมือในการสร้างภาพแบบจำลองลายเส้นสามมิติได้ในระบบ World Coordinate และสามารถแสดงผลลัพธ์บนจอภาพเครื่องคอมพิวเตอร์ทั้งลักษณะ Projection แบบสองมิติและสามมิติ ข้อมูลของจุดและเส้นตรงได้ถูกนำมาใช้เป็นวัตถุพื้นฐานเพื่อประกอบให้เป็นรูปทรงลายเส้นสามมิติตามต้องการ

แฟ้มข้อมูลที่ใช้จัดเก็บข้อมูลมีการปรับเปลี่ยนจากการออกแบบในชั้นต้นที่จะให้เป็นฐานข้อมูล มาเป็นการใช้ Formated File แทน เนื่องจากผู้วิจัยเน้นให้ผู้ใช้งานสามารถปรับเปลี่ยนข้อมูลได้โดยใช้โปรแกรม Editor ธรรมดาทั่วไป เช่น Notepad ได้ง่ายและเหมาะสำหรับการเรียนการสอน ในที่นี้ผู้ใช้งานคือนิสิต ที่จะสามารถปรับเปลี่ยนข้อมูลจุดและเส้นตามต้องการได้ง่ายกว่า ตลอดจนการติดตั้งและนำไปใช้งานสามารถทำได้ง่ายโดยที่ไม่ต้องติดตั้งระบบฐานข้อมูลเพื่อใช้งาน

โปรแกรมที่พัฒนาขึ้นมีขีดความสามารถในการใช้งานที่อำนวยความสะดวกให้ผู้ใช้ได้ดี และได้รับการทดสอบว่าสามารถทำงานได้ถูกต้อง อย่างไรก็ตามผู้วิจัยคิดว่าเครื่องมือบางอย่างยังคงต้องมีการพัฒนาเพิ่มเติมในโอกาสถัดไปคือ การปรับปรุงเครื่องมือ Select และการปรับปรุงความแม่นยำในการวางตำแหน่งวัตถุให้ง่ายขึ้น เป็นต้น

ในการจำลองวัตถุที่เห็นอยู่ตามธรรมชาติ นอกจากประกอบด้วยจุดและเส้น ยังมีเทคนิคการใช้ส่วนเส้นโค้งที่เรียกว่า Arc เพื่อให้การนำเสนอเส้นโค้งมีความมนไม่ดูหยาบ ในรูปจำลองที่มีความละเอียดต่ำ นอกจากนี้แล้วการจำลองวัตถุยังสามารถทำได้โดยใช้สมการทางคณิตศาสตร์ เช่น การทำเส้นโค้งแบบ Spline เป็นต้น โดยมีจุดควบคุมความโค้งตั้งแต่ 2 จุดขึ้นไปเพิ่มเข้ามาในการแสดงส่วนโค้ง เป็นต้น

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

รายการอ้างอิง

[1] J. D. Foley, A. V. Dam, S. K. Feiner, J. F. Hughes, "Computer Graphics – Principles and Practice", 2nd Edition, Addison-Wesley, 1990.

[2] D. J. Kruglinski, "Inside Visual C++ ", 2nd Edition, Microsoft Press, 1994.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ก.
Method ที่พัฒนาขึ้นสำหรับคลาสเครื่องมือการจัดการจุดและเส้น

คำอธิบาย Method ที่พัฒนาขึ้นสำหรับคลาสเครื่องมือในการจัดการจุดและเส้น ได้รับการบันทึกไว้เพื่อใช้เป็นคำแนะนำเพื่อให้เข้าใจและสามารถเลือก Method ที่ต้องการไปใช้ต่อได้ อย่างไรก็ตามการนำไปใช้งานใหม่นั้นจะต้องดูส่วนตัวแปรที่ใช้งานให้สอดคล้องกันด้วย รูปที่ ก-1 แสดงรายการ Method ที่มีอยู่ในระบบของคลาสหลัก C3dFrameObject และคำอธิบายโดยสังเขปมีดังต่อไปนี้

ClassName: C3dFrameObject Class

MethodName: AddLine( )

MethodType: Public

Action: ทำหน้าที่ในการเชื่อมจุดสองจุดที่มีอยู่แล้วให้เกิดเส้นตรงขึ้นใหม่อีก สำหรับวัตถุประเภท Polyline โดยจะทำการเพิ่มรายการเส้นเชื่อมไปอีกหนึ่งบรรทัด และนับจำนวนเส้นเพิ่มให้สอดคล้อง

ClassName: C3dFrameObject Class

MethodName: AddPoint( )

MethodType: Public

Action: ทำหน้าที่ในการเพิ่มจุดใหม่ให้กับวัตถุที่กำหนด การเพิ่มจุดส่วนใหญ่จะกระทำกับวัตถุประเภท Polyline โดยจะทำการเพิ่มรายการจุดไปอีกหนึ่งบรรทัด และนับจำนวนจุดเพิ่มให้สอดคล้อง

ClassName: C3dFrameObject Class

MethodName: CanAddElement( )

MethodType: Public

Action: ทำหน้าที่ในการตรวจสอบว่าวัตถุที่ต้องการเพิ่มจุดหรือเส้น สามารถรับการเพิ่มนั้นได้หรือไม่ เช่นวัตถุประเภท Polyline ที่จะต้องมีเส้นเพิ่มแต่มีจุดที่กำหนดไว้ไม่เพียงพอ เป็นต้น

ClassName: C3dFrameObject Class

MethodName: CanSplite( )

MethodType: Public

Action: ทำหน้าที่ในการแยกวัตถุประเภท Polyline ออกเป็นกลุ่มวัตถุของเส้นเดี่ยว ทำให้ผู้ใช้สามารถลบเส้นบางเส้นออก และถ้าต้องการรวมกับเป็นวัตถุเดียวกันก็สามารถทำได้โดยใช้ Method Merge( )

ClassName: C3dFrameObject Class

MethodName: Clone( )

MethodType: Public

Action: ทำหน้าที่ในการ copy วัตถุต้นแบบ ให้มีซ้ำอีกได้ เพื่อนำไปใช้ต่อได้

ClassName: C3dFrameObject Class

MethodName: CreateObject( )

MethodType: Public

Action: ทำหน้าที่ในการกำหนดวัตถุใหม่ที่จะทำการ Load ขึ้นมาใช้งาน ในการทำ Object-Oriented Programming นั้นจะมีการกำหนดวัตถุไว้ก่อน แล้วจึงทำการ Load วัตถุเข้าจริง

ClassName: C3dFrameObject Class

MethodName: Draw( )

MethodType: Public

Action: ทำหน้าที่ในการวาดวัตถุในส่วนจอภาพแบบสามมิติ ในโปรแกรมนี้จะเป็นส่วนของจอภาพย่อยที่อยู่ด้านซ้ายล่าง ผู้ใช้สามารถมองภาพมีมิติได้

ClassName: C3dFrameObject Class

MethodName: DrawFront( )

MethodType: Public

Action: ทำหน้าที่ในการวาดวัตถุในส่วนจอภาพแบบสองมิติ โดยจะแสดงเฉพาะส่วนด้าน Front เท่านั้น

ClassName: C3dFrameObject Class

MethodName: DrawSide( )

MethodType: Public

Action: ทำหน้าที่ในการวาดวัตถุในส่วนจอภาพแบบสองมิติ โดยจะแสดงเฉพาะส่วนด้าน Side เท่านั้น

ClassName: C3dFrameObject Class

MethodName: DrawTop( )

MethodType: Public

Action: ทำหน้าที่ในการวาดวัตถุในส่วนจอภาพแบบสองมิติ โดยจะแสดงเฉพาะส่วนด้าน Top เท่านั้น

ClassName: C3dFrameObject Class

MethodName: FindLineFront( )

MethodType: Public

Action: ทำหน้าที่ในการค้นหาเส้นที่ผู้ใช้เลือกโดยการคลิกเมาส์ ในส่วนจอภาพที่เป็น Front view

ClassName: C3dFrameObject Class

MethodName: FindLineSide( )

MethodType: Public

Action: ทำหน้าที่ในการค้นหาเส้นที่ผู้ใช้เลือกโดยการคลิกเมาส์ ในส่วนจอภาพที่เป็น Side view

ClassName: C3dFrameObject Class

MethodName: FindLineTop( )

MethodType: Public

Action: ทำหน้าที่ในการค้นหาเส้นที่ผู้ใช้เลือกโดยการคลิกเมาส์ ในส่วนจอภาพที่เป็น Top view

ClassName: C3dFrameObject Class

MethodName: FindPointFront( )

MethodType: Public

Action: ทำหน้าที่ในการค้นหาจุดที่ผู้ใช้เลือกโดยการคลิกเมาส์ ในส่วนจอภาพที่เป็น Front view

ClassName: C3dFrameObject Class

MethodName: FindPointSide( )

MethodType: Public

Action: ทำหน้าที่ในการค้นหาจุดที่ผู้ใช้เลือกโดยการคลิกเมาส์ ในส่วนจอภาพที่เป็น Side view

ClassName: C3dFrameObject Class

MethodName: FindPointTop( )

MethodType: Public

Action: ทำหน้าที่ในการค้นหาจุดที่ผู้ใช้เลือกโดยการคลิกเมาส์ ในส่วนจอภาพที่เป็น Top view

ClassName: C3dFrameObject Class

MethodName: GetFirstLine( )

MethodType: Public

Action: ทำหน้าที่ในการค้นหาเส้นแรกของวัตถุที่ผู้ใช้เลือก โดยที่เมื่อเส้นแรกของวัตถุพบก็จะส่งตำแหน่งการอ้างอิงกลับมาเพื่อขอข้อมูลของเส้นต่อไป

ClassName: C3dFrameObject Class

MethodName: GetLine( )

MethodType: Public

Action: ทำหน้าที่ในการอ่านข้อมูลของเส้นที่ผู้ใช้เลือก โดยใช้ตำแหน่งอ้างอิงของเส้นเป็นหลัก

ClassName: C3dFrameObject Class

MethodName: GetNextLine( )

MethodType: Public

Action: ทำหน้าทีในการเลือกเส้นที่อยู่ต่อเนื่องจากเส้นที่เลือก และส่งตำแหน่งการอ้างอิงกลับมา

ClassName: C3dFrameObject Class
MethodName: GetLineCount( )
MethodType: Public
Action: ทำหน้าที่ในการนับจำนวนเส้นที่มีอยู่ในวัตถุที่ผู้ใช้เลือก

ClassName: C3dFrameObject Class
MethodName: GetFirstPoint( )
MethodType: Public
Action: ทำหน้าที่ในการค้นหาตำแหน่งของจุดแรกที่ผู้ใช้เลือก

ClassName: C3dFrameObject Class
MethodName: GetPoint( )
MethodType: Public
Action: ทำหน้าที่ในการขอข้อมูลของจุดที่ต้องการ โดยจะได้ค่าพิกัด X, Y, Z กลับมาใช้งานต่อไป

ClassName: C3dFrameObject Class
MethodName: GetNextPoint( )
MethodType: Public
Action: ทำหน้าที่ในการหาจุดที่ต่อเนื่องจากจุดเดิม และส่งตำแหน่งจุดกลับมาใช้งาน

ClassName: C3dFrameObject Class
MethodName: GetPointCount( )
MethodType: Public
Action: ทำหน้าที่ในการนับจำนวนจุดที่ปรากฏอยู่ในวัตถุที่ผู้ใช้เลือก

ClassName: C3dFrameObject Class
MethodName: IsSelected( )
MethodType: Public
Action: ทำหน้าที่ในการตรวจสอบว่าวัตถุได้ถูกเลือก หรือคลิกหรือไม่

ClassName: C3dFrameObject Class
MethodName: JoinPoint( )
MethodType: Public
Action: ทำหน้าที่ในการรวมจุดที่มีอยู่บนจอภาพให้มาอยู่ในวัตถุเดียวกัน โดยวัตถุจะมีจุดเพิ่มเข้ามา

ClassName: C3dFrameObject Class
MethodName: Load( )

MethodType: Public

Action: ทำหน้าที่ในการ Load วัตถุให้เข้ามาสู่หน่วยความจำเพื่อใช้งานจริง หลังจากที่ได้ทำการกำหนดวัตถุไว้แล้วโดย Method CreateObject( )

ClassName: C3dFrameObject Class

MethodName: Merge( )

MethodType: Public

Action: ทำหน้าที่ในการรวมวัตถุสองวัตถุให้เป็นวัตถุเดียวกัน โดยวัตถุที่สองจะถูกรวมเข้ามาเป็นวัตถุเดียวกับวัตถุแรก

ClassName: C3dFrameObject Class

MethodName: Move( )

MethodType: Public

Action: ทำหน้าที่ในการเลื่อนวัตถุไปมาใน World Coordinate System

ClassName: C3dFrameObject Class

MethodName: MovePoint( )

MethodType: Public

Action: ทำหน้าที่ในการเลื่อนตำแหน่งของจุดหรือเส้นใดๆ ไปมาใน World Coordinate System

ClassName: C3dFrameObject Class

MethodName: Register3dFrameObject( )

MethodType: Public

Action: ทำหน้าที่ในการลงทะเบียนวัตถุที่ได้รับการ Load เข้าสู่หน่วยความจำเพื่อระบบจะได้ใช้งานได้

ClassName: C3dFrameObject Class

MethodName: RemoveLine( )

MethodType: Public

Action: ทำหน้าที่ในการลบเส้นใดๆ ที่ต้องการออกจากวัตถุจริง

ClassName: C3dFrameObject Class

MethodName: Rotate( )

MethodType: Public

Action: ทำหน้าที่ในการหมุนวัตถุ ใน World Coordinate System

ClassName: C3dFrameObject Class

MethodName: Scale( )

MethodType: Public

Action: ทำหน้าที่ในการย่อหรือขยายขนาดจริงของวัตถุ และจะมีผลจริงกับขนาดของวัตถุ ไม่เพียงการมองเห็นเท่านั้น

ClassName: C3dFrameObject Class

MethodName: Save( )

MethodType: Public

Action: ทำหน้าที่ในการจัดเก็บวัตถุไว้ในแฟ้มข้อมูล เพื่อใช้ต่อไปในคราวหน้า

ClassName: C3dFrameObject Class

MethodName: Select( )

MethodType: Public

Action: ทำหน้าที่ในการเลือกวัตถุ โดยที่ผู้ใช้จะทำการคลิกที่ภาพวัตถุบนจอภาพ

ClassName: C3dFrameObject Class

MethodName: DrawSelect( )

MethodType: Protected

Action: ทำหน้าที่ในการวาดวัตถุที่เลือกไว้โดยใช้ Method Select( ) แล้ววาดใหม่ให้มีสีที่แตกต่างกัน ผู้ใช้สามารถสังเกตได้ง่าย

ClassName: C3dFrameObject Class

MethodName: M_IsSelected

MethodType: Protected

Action: เป็นส่วนข้อมูล Array ที่แสดงสถานะว่าถูกเลือกหรือไม่

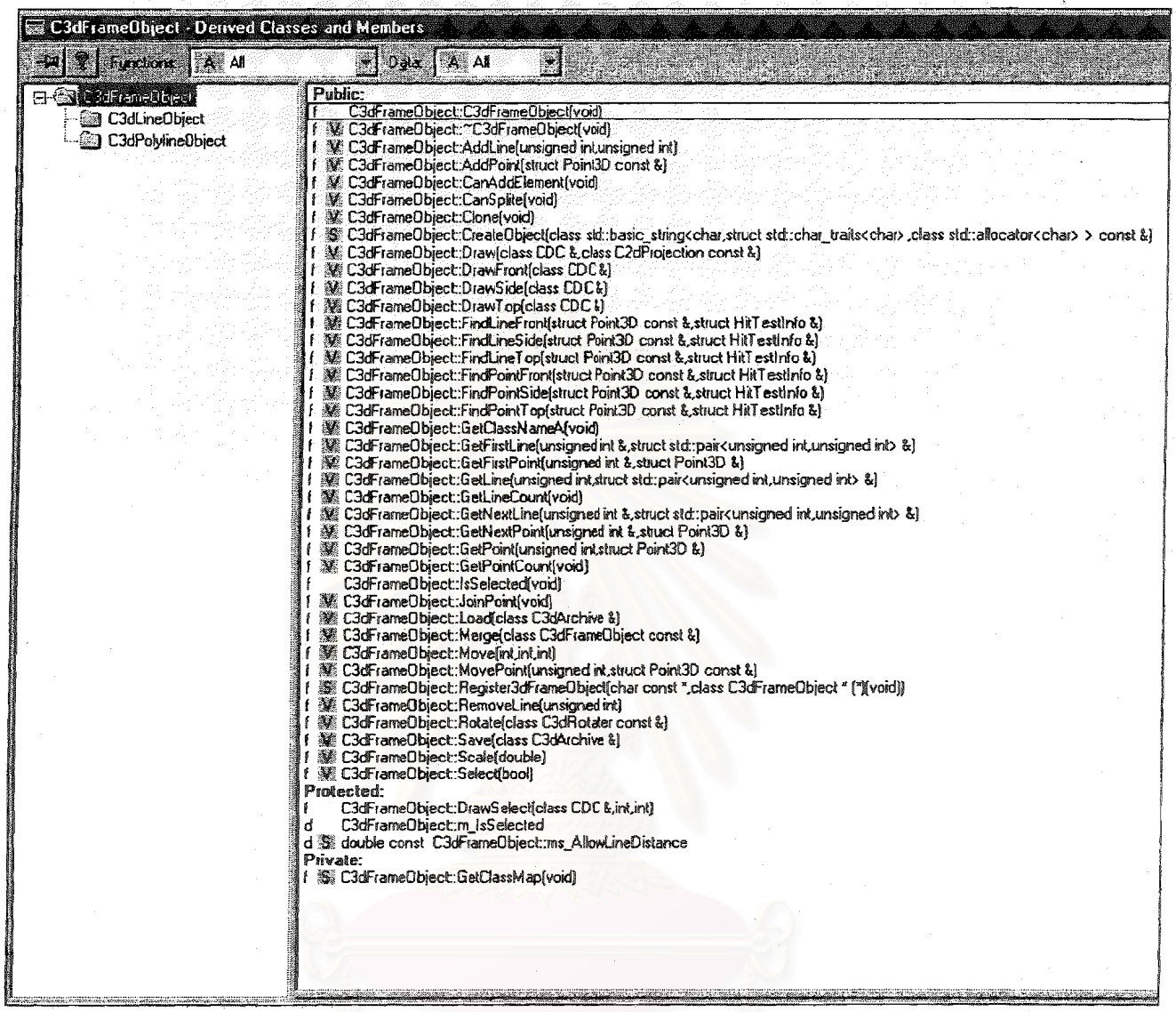
ClassName: C3dFrameObject Class

MethodName: Ms_AllowLineDistance

MethodType: Protected

Action: เป็นส่วนข้อมูลที่กำหนดค่าระยะยืดหยุ่นสำหรับการคลิกบนจอภาพ ผู้ใช้สามารถคลิกที่ตำแหน่งเส้นหรือจุดที่ได้โดยมีระยะห่างยืดหยุ่นได้

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ ก-1 รายการ Method ที่มีอยู่ในระบบของคลาสหลัก C3dFrameObject

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย