



การคำนวณอัตราส่วนภายในของวัตถุทรงตันสำหรับการรู้จำวัตถุ 3 มิติ

โดย

นายจิณณวัตร วงศ์วิเศษ

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรมหาบัณฑิต (วิทยาการคอมพิวเตอร์)
สาขาวิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์
ปีการศึกษา 2559
ลิขสิทธิ์ของมหาวิทยาลัยธรรมศาสตร์

การคำนวณอัตราส่วนภายในของวัตถุทรงตันสำหรับการรู้จำวัตถุ 3 มิติ

โดย

นายจิณณวัตร วงศ์วิเศษ



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรมหาบัณฑิต (วิทยาการคอมพิวเตอร์)
สาขาวิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์
ปีการศึกษา 2559
ลิขสิทธิ์ของมหาวิทยาลัยธรรมศาสตร์

DISTANCE INTERIOR RATIO OF VOLUMETRIC MODEL FOR
OBJECT RECOGNITION

BY

MR. CHINNAWAT WONGWISES



A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE (COMPUTER SCIENCE)

DEPARTMENT OF COMPUTER SCIENCE
FACULTY OF SCIENCE AND TECHNOLOGY
THAMMASAT UNIVERSITY

ACADEMIC YEAR 2016

COPYRIGHT OF THAMMASAT UNIVERSITY

มหาวิทยาลัยธรรมศาสตร์
คณะวิทยาศาสตร์และเทคโนโลยี

วิทยานิพนธ์

ของ


นายจิณณวัตร วงศ์วิเศษ

เรื่อง

การคำนวณอัตราส่วนภายในของวัตถุทรงตันสำหรับการรู้จำวัตถุ 3 มิติ
ได้รับการตรวจสอบและอนุมัติ ให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรมหาบัณฑิต (วิทยาการคอมพิวเตอร์)

เมื่อ วันที่ ๒๑ กรกฎาคม พ.ศ. 2560

ประธานกรรมการสอบวิทยานิพนธ์



(ผู้ช่วยศาสตราจารย์ ดร. วิลาวรรณ รักผกาวงศ์)

กรรมการและอาจารย์ที่ปรึกษาวิทยานิพนธ์



(ดร. วิศิษฐ์ ลิ้มประเสริฐ)

กรรมการสอบวิทยานิพนธ์



(ดร. ปกป้อง ส่องเมือง)

กรรมการสอบวิทยานิพนธ์



(ดร. สรพรพฤษดิ์ มุคคทัต)

คณบดี



(รองศาสตราจารย์ ปกรณ์ เสริมสุข)

หัวข้อวิทยานิพนธ์	การคำนวณอัตราส่วนภายในของวัตถุทรงตันสำหรับการ รู้จำวัตถุ 3 มิติ
ชื่อผู้เขียน	นายจิณณวัตร วงศ์วิเศษ
ชื่อปริญญา	วิทยาศาสตรมหาบัณฑิต (วิทยาการคอมพิวเตอร์)
สาขาวิชา/คณะ/มหาวิทยาลัย	สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์	ดร. วสิศ ลิ้มประเสริฐ
ปีการศึกษา	2559

บทคัดย่อ

ในงานวิจัยชิ้นนี้ได้นำเสนอวิธีการคำนวณอัตราส่วนภายในของวัตถุ 3 มิติ ซึ่งขยายมาจากงานวิจัยการคำนวณอัตราส่วนภายในของวัตถุ 2 มิติ ในการที่จะอธิบายวัตถุซึ่งใช้แทนวัตถุ 3 มิติ ทรงตัน การคำนวณอัตราส่วนภายในวัตถุ 3 มิติ นั้นสามารถนิยามได้ดังต่อไปนี้ กำหนดเส้นตรงระหว่างตำแหน่งพิกัด 2 พิกัดในวัตถุสามมิติโดยอัตราส่วนภายในของวัตถุคือระยะทางผลรวมของเส้นตรงที่อยู่ภายในวัตถุกับระยะทางของเส้นตรง โดยนำอัลกอริทึมการตัดกันระหว่างรังสีและแผ่นพื้นผิวในการวิเคราะห์แต่ละช่วงของเส้นตรง จากนั้นนำผลลัพธ์การกระจายของของอัตราส่วนภายในมาสร้างแผนภูมิสถิติซึ่งใช้ในการอธิบายวัตถุสามมิติ โดยในการทดลองเบื้องต้นพบว่าใช้ระยะเวลาในการคำนวณนานเนื่องจากการเรียกใช้อัลกอริทึมการตัดกันระหว่างรังสีและแผ่นพื้นผิวในการคำนวณจำนวนมาก ดังนั้นเพื่อลดการเรียกใช้อัลกอริทึมจึงมีการนำข้อมูลวัตถุมาแบ่งส่วน เพื่อลดการเรียกใช้อัลกอริทึม โดยออกแบบระบบการคำนวณโดยการแบ่งวัตถุออกเป็นสองระบบคือระบบการคำนวณแบบเป็นลำดับและระบบการคำนวณแบบขนาน

คำสำคัญ: การอธิบายรูปทรง, อัตราส่วนระยะทางภายใน, การตัดกันระหว่างรังสีและแผ่นสามเหลี่ยม, การแบ่งส่วนวัตถุ

Thesis Title	Distance Interior Ratio of Volumetric Models for Object Recognition
Author	MR. Chinnawat Wongwises
Degree	Master of Science (Computer Science)
Major Field/Faculty/University	Department of Computer Science Faculty of Science and Technology Thammasat University
Thesis Advisor	Dr. Wasit Limprasert
Academic Years	2016

ABSTRACT

In this thesis, we present the method how to compute distance interior ratio of the 3D objects called 3DDIR. This thesis is extended from novel 2D shape descriptor that describe the 2D shape. To describe a 3D object represented by a volumetric model. The 3DDIR is defined as follow: construct the line between two vertices on the surface of object. The DIR of the line is the ratio of the total length of its fragments that lay inside the object to the length of the line. To find the fragments, we apply a simple ray triangle intersection to analyze the location. Then construct the histogram from distributions DIR data. From in the first step, we found the system have to compute a lot of time for construct histogram. Because the system has to use the ray triangle intersection algorithm to compute for many times that make a lot of computation time. Then we bring the idea of space partitioning to reduce the computation time. And we design the adds-on of the system called single thread partition and parallel partition.

Keywords: Shape descriptors, Distance Interior Ratio, Ray triangle intersection, Space partitioning

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้เป็นอย่างดี เนื่องจากได้รับความอนุเคราะห์จาก ดร.วสิศ ลิ้มประเสริฐ อาจารย์ที่ปรึกษาของวิทยานิพนธ์ ที่ให้ความรู้ คำแนะนำ และตรวจสอบแก้ไข ข้อบกพร่องต่างๆ เพื่อให้วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี และขอขอบคุณผู้ช่วยศาสตราจารย์ ดร. วิลาวรรณ รักภักทวงศ์ ประธานกรรมการสอบวิทยานิพนธ์ รวมถึง ดร. ปกป้อง ส่องเมือง กรรมการสอบวิทยานิพนธ์ และดร. สรรพฤทธิ์ มฤคทัต ผู้ทรงคุณวุฒิที่กรุณาสละเวลาและให้ข้อเสนอแนะเพื่อปรับปรุงแก้ไขวิทยานิพนธ์ให้มีความสมบูรณ์ยิ่งขึ้น ผู้วิจัยขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้ด้วย

ขอกราบขอบพระคุณ ดร.ณัฐสุดา เกาทัณฑ์ทอง อาจารย์ผู้จุดประกาย และให้ความรู้ คำแนะนำ และคอยช่วยเหลือเพื่อให้วิทยานิพนธ์ฉบับนี้สำเร็จไปได้ด้วยดี

ขอกราบขอบพระคุณ คณาจารย์ทุกท่านที่ได้ให้ความรู้ตลอดเวลาที่ศึกษา ขอขอบพระคุณ เจ้าหน้าที่ประจำภาควิชาวิทยาการคอมพิวเตอร์ทุกท่านที่คอยช่วยเหลือและสนับสนุนในการทำ วิทยานิพนธ์

ขอกราบขอบพระคุณพ่อและแม่ที่อนุเคราะห์และเป็นกำลังใจตลอดระยะเวลาการศึกษา และทำวิทยานิพนธ์ และขอขอบคุณเพื่อนที่ศึกษาจากมหาวิทยาลัยธรรมศาสตร์ทุกท่านที่ช่วยเหลือในการเรียนรู้ตลอดการศึกษา

นายจิณณวัตร วงศ์วิเศษ

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	(1)
บทคัดย่อภาษาอังกฤษ	(2)
กิตติกรรมประกาศ	(3)
สารบัญตาราง	(8)
สารบัญภาพ	(9)
รายการสัญลักษณ์และคำย่อ	(10)
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญ	1
1.2 วัตถุประสงค์ของงานวิจัย	4
1.3 ขอบเขตของงานวิจัย	4
1.3.1 ชุดข้อมูลที่ใช้ในการทดลอง	4
1.3.2 ชนิดข้อมูลที่ใช้ในการทดลอง	4
1.4 ประโยชน์ที่คาดว่าจะได้รับ	5
1.5 รายละเอียดโครงสร้างวิทยานิพนธ์	5
บทที่ 2 วรรณกรรมและงานวิจัยที่เกี่ยวข้อง	6
2.1 ทฤษฎีที่เกี่ยวข้อง	6

2.1.1 พื้นฐานของคอมพิวเตอร์กราฟิกส์ (Basic of computer graphics)	6
2.1.1.1 ตำแหน่งพิกัด	7
2.1.1.2 แผ่นพื้นผิว	7
2.1.1.3 วัตถุ	8
2.1.1.4 รังสี	8
2.1.2 วิธีการคำนวณที่นำมาใช้ในงานวิจัย	8
2.1.2.1 Ray Triangle Intersection	9
2.1.2.2 Axis Aligned Bounding Box	9
2.1.2.3 Minimal Ray Tracer	9
2.1.2.4 Randomized Decision Forests	9
2.1.3 วิธีการอธิบายวัตถุสามมิติ	10
2.1.3.1 วิธีการอธิบายวัตถุสามมิติด้วยกราฟ	10
2.1.3.2 วิธีการอธิบายวัตถุสามมิติด้วยลักษณะ	11
2.2 งานวิจัยที่เกี่ยวข้อง	12
2.2.1 งานวิจัยเกี่ยวกับการอธิบายรูปทรงสามมิติ	12
2.2.1.1 Shape Distribution(D2)	12
2.2.1.2 Shape Classification Using the Inner-Distance	12
2.2.1.3 Using Shape Distributions to Compare Solid Models	13
2.2.1.4 Distance Interior Ratio(DIR)	13
2.2.2 งานวิจัยเกี่ยวกับโครงสร้างข้อมูลในคอมพิวเตอร์กราฟิกส์	13
2.2.2.1 Binary Space Partition	13
2.2.2.2 K-Dimension Tree	14
2.2.2.3 Octree	14
2.3 ตารางการเปรียบเทียบงานวิจัย	15
2.3.1 ตารางการเปรียบเทียบงานวิจัยเกี่ยวกับการอธิบายรูปทรงสามมิติ	15
2.3.2 ตารางการเปรียบเทียบงานวิจัยโครงสร้างข้อมูลในคอมพิวเตอร์กราฟิกส์	16
บทที่ 3 วิธีการวิจัย	17
3.1 แนวทางการแก้ปัญหา	17

3.2	เตรียมชุดข้อมูลในการทดลอง	17
3.3	ขั้นตอนวิธีของระบบที่ 1	18
3.4	วิธีการคำนวณระยะทาง Distance Interior Ratio	19
3.5	การสร้างแผนภูมิสถิติเพื่ออธิบายวัตถุสามมิติ	21
3.6	วิธีการเปรียบเทียบความคล้ายกันระหว่างวัตถุ	22
3.7	ผลจากการทดลองของระบบที่ 1	23
3.8	ปัญหาเบื้องต้นของระบบที่ 1	24
3.9	แนวทางการแก้ปัญหาโดยวิธีแบ่งวัตถุสามมิติ	25
3.10	ขั้นตอนวิธีของระบบ 2	26
3.11	วิธีการแบ่งวัตถุสามมิติและการสร้างต้นไม้	27
3.12	ขั้นตอนในการคัดเลือกแผ่นพื้นผิวเพื่อนำไปคำนวณจุดตัด	30
3.13	ออกแบบการทำงานแบบขนาน	31
3.14	วิธีการทดลอง	32
บทที่ 4	ผลการวิจัยและอภิปรายผล	36
4.1	เปรียบเทียบกราฟผลลัพธ์เวลา	37
4.2	ผลลัพธ์แนวโน้มของการคำนวณ	40
4.3	ตรวจสอบผลลัพธ์เมื่อเพิ่มส่วนของการแบ่งวัตถุ	47
4.4	อภิปรายการทางานของระบบแบบขนาน	51
4.5	อภิปรายการขั้นตอนของการค้นหา	54
บทที่ 5	สรุปผลการวิจัยและข้อเสนอแนะ	55
5.1	สรุปผลการวิจัย	55
5.2	อภิปรายผลลัพธ์ความเร็วในการคำนวณ	56
5.3	อภิปรายแนวโน้มของการคำนวณระบบแบบขนาน	57
5.4	อภิปรายการจำแนกของระบบการคำนวณแบบขนาน	58
5.3	ข้อเสนอแนะของงานวิจัย	59

	(7)
รายการอ้างอิง	60
ภาคผนวก	63
ประวัติผู้เขียน	73



สารบัญตาราง

ตารางที่	หน้า
2.1 เปรียบเทียบงานวิจัยที่เกี่ยวข้องกับการอธิบายวัตถุ	15
2.2 เปรียบงานวิจัยเกี่ยวกับโครงสร้างข้อมูลในคอมพิวเตอร์กราฟิกส์	16
3.1 ตารางผลลัพธ์การเปรียบเทียบการจำแนกวัตถุ	22
4.1 ตารางเปรียบเทียบเวลาในการทำงานในหน่วยวินาที	39
4.2 ตารางเวลาในการทำงานของระบบในหน่วยวินาที	45
4.3 ตารางผลลัพธ์การเปรียบเทียบการจำแนกวัตถุ	47
4.4 ตารางเปรียบเทียบเวลาในการทำงานในหน่วยวินาที	49
4.5 ผลลัพธ์การทำงานของระบบโดยเปลี่ยนจำนวน processors	53
4.6 ตารางเปรียบเทียบวิธีในการค้นหา	54
5.1 ตารางเปรียบเทียบการทำงานของแต่ละระบบ	56
5.2 ตารางเปรียบเทียบการทำงานของแต่ละขั้นตอน	57
5.3 ตารางวัดผลการจำแนกและเวลาเฉลี่ย	58
6.1 ตารางเปรียบเทียบเวลาในการคำนวณของระบบ	64
6.2 ตารางเปรียบเทียบเวลาในการคำนวณของแต่ละวิธีการคำนวณ	67

สารบัญภาพ

ภาพที่	หน้า
1.1 กราฟการกระจายของรูปทรงด้วยวิธี D2	2
2.1 องค์ประกอบของวัตถุสามมิติ	7
2.2 ตัวอย่างการแปลงวัตถุเพื่อใช้ในการอธิบาย	10
3.1 ตัวอย่างขั้นตอนการทำงานของระบบ	18
3.2 ตัวอย่างการตัดกันระหว่างรังสีและแผ่นพื้นผิว	19
3.3 ตัวอย่างการแบ่งช่วงของเส้นตรง	20
3.4 กราฟวัดเวลาการทำงานในการคำนวณของ 3DDIR	23
3.5 แผนผังการทำงานโดยรวมของระบบ	25
3.6 โครงสร้างของต้นไม้ที่ทำการแบ่งวัตถุ	27
3.7 ตัวอย่างขั้นตอนการแบ่งวัตถุ	29
3.8 ขั้นตอนในการคัดเลือกเซตของแผ่นพื้นผิว	30
3.9 กระบวนการของระบบคำนวณแบบขนาน	31
3.10 ตัวอย่าง Source code ของระบบแบบขนาน	32
3.11 ตัวอย่าง Source code ของ worker function	33
4.1 กราฟแสดงความสัมพันธ์ระหว่างจำนวนแผ่นพื้นผิวและเวลาในการทำงาน	37
4.2 กราฟแสดงความสัมพันธ์ระหว่างจำนวนแผ่นพื้นผิวและเวลาในการทำงาน	38
4.3 กราฟความสัมพันธ์ระหว่างจำนวนแผ่นพื้นผิวและเวลาในหน่วยวินาที	41
4.4 กราฟความสัมพันธ์ระหว่างเวลาและจำนวนส่วนที่ถูกแบ่งในช่วงการคำนวณ	42
4.5 กราฟความสัมพันธ์ระหว่างเวลาและจำนวนแผ่นพื้นผิวในช่วงการคำนวณ	43
4.6 กราฟความสัมพันธ์ระหว่างเวลาและจำนวนแผ่นพื้นผิวตลอดระบบการคำนวณ	44
4.7 กราฟ Confuse Matrix วัดผลความคล้ายคลึงของวัตถุสามมิติ	46
4.8 กราฟความสัมพันธ์ระหว่างเวลาและจำนวนแผ่นพื้นผิวเปรียบเทียบแต่ละวิธี	48
4.9 กราฟความสัมพันธ์ระหว่างเวลาและจำนวนแผ่นพื้นผิวเปรียบเทียบแต่ละวิธี	49
4.10 แสดงผลลัพธ์การทำงานของระบบ 3DDIR Single Partition	51
4.11 แสดงผลลัพธ์การทำงานของระบบ 3DDIR Parallel Partition	52

รายการสัญลักษณ์และคำย่อ

สัญลักษณ์/คำย่อ	คำเต็ม/คำจำกัดความ
LSH	Locality Sensitive Hashing
SC	Shape Context
D2	Pairwise Distance
IDSC	Inner Distance Shape Context
DIR	Distance Interior Ratio
3DDIR	3D Distance Interior Ratio
AABB	Axis Aligned Bounding Box
BSP	Binary Space Partition
K-d tree	K-Dimension Tree
d_{ir}	อัตราส่วนระยะทางภายใน
$h(u,v)$	สมการการคำนวณกราฟสถิติ
$S(A,B)$	ค่าความคล้ายคลึงระหว่างวัตถุ

บทที่ 1

บทนำ

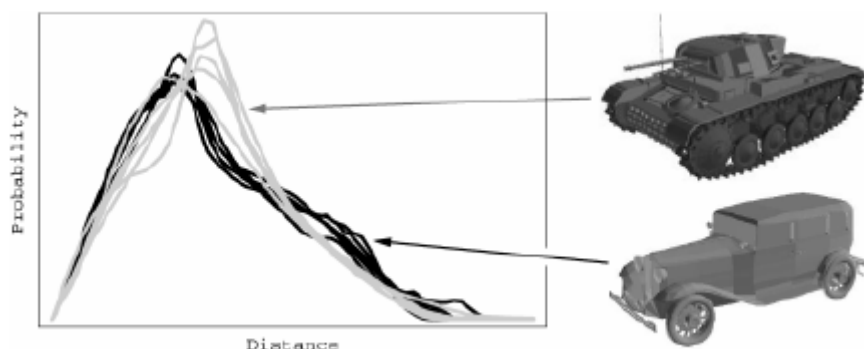
1.1 ที่มาและความสำคัญ

เนื่องด้วยความก้าวหน้าของการจำลองวัตถุและการแสดงผลนำมาซึ่งการเพิ่มขึ้นของจำนวนรูปทรง 3 มิติทั้งบนอินเทอร์เน็ตและบนฐานข้อมูลโดเมนเฉพาะ ซึ่งสิ่งนี้นำมาซึ่งการพัฒนาของระบบการค้นหาสำหรับรูปทรง 3 มิติ (12) การค้นคืนคือกระบวนการของการค้นหาและการนำส่ง ผลลัพธ์คิวรี เพื่อพิจารณาว่ารูปทรงสองคู่มีความคล้ายคลึงกันการจับคู่ที่นั้นถูกใช้ประโยชน์โดยการ คำนวณระยะทาง อย่างไรก็ตามการใช้กระบวนการจับคู่โดยตรงของคู่วัตถุที่ใช้เวลานาน เพื่อเพิ่มประสิทธิภาพการค้นคืนวัตถุการอธิบายรูปร่างของรูปทรง 3 มิติจึงมีบทบาทที่สำคัญ เนื่องจากว่า มันสามารถจัดดัชนีและใช้งานเพื่อเพิ่มประสิทธิภาพโครงสร้างของการค้นหา เช่น 'Locality Sensitive Hashing' (LSH) (13) ความท้าทายของการพัฒนาวิธีการอธิบายรูปร่าง ไม่ว่าจะ เป็นรูปร่าง 2 มิติหรือวัตถุ 3 มิติคือความสามารถในการอธิบายรูปทรงที่มีความซับซ้อนได้อย่างมีประสิทธิภาพ

ในบริบทของรูปทรง 3 มิติการอธิบายวัตถุตามลักษณะเราสามารถแบ่งประเภทได้เป็น 4 ประเภทใหญ่ๆ : 1) การอธิบายลักษณะพื้นที่(Local features) 2) การอธิบายลักษณะทั่วไป(Global features) 3) การกระจายลักษณะทั่วไป (Global feature distribution) และ 4) แผนภาพเชิงพื้นที่(Spatial map) การอธิบายลักษณะพื้นที่นั้นบรรยายรูปทรง 3 มิติรอบๆ จำนวนของจุดบนพื้นผิวอย่างเช่น 'Shape Context' (SC)(1)(4) ซึ่งวิธีการอื่นจะใช้ค่า ของเวกเตอร์ซึ่งบ่งบอกถึงข้อมูลของรูปทรง 3 มิติ

การอธิบายลักษณะทั่วไปบ่งบอกข้อมูลอย่างเช่น ปริมาณ,พื้นที่หรือขอบเขตของรูปร่าง (15) การกระจายลักษณะทั่วไปจะบรรยายเปรียบเทียบการกระจายของลักษณะทั่วไปเช่น'Pairwise Distance' (6) แผนภาพเชิงพื้นที่อธิบายถึงตำแหน่งพื้นที่ที่สอดคล้องกับชิ้นส่วนกายภาพของวัตถุ

จากการศึกษางานวิจัยที่ผ่านมา มีผู้วิจัยได้นำเสนอวิธีการบรรยายลักษณะพื้นที่รูปร่าง 2 มิติซึ่งเรียกว่า 'Inner Distance Shape Context' (IDSC) (9) ซึ่งวิธีนี้จะคำนวณโดยใช้ระยะที่ถูกระบุโดยเส้นทางที่สั้นที่สุดภายในเส้นแสดงรูปร่าง เพื่อที่จะเอาระยะภายในต้องทำการสร้างกราฟระหว่างจุดที่อยู่ในรูปร่าง 2 มิติแนวคิดนี้ของ IDSC นั้นถูกขยายเพิ่มเติมโดย 'Computing the Inner Distance of Volumetric Models for Articulated Shape Description with a Visibility Graph' (5) เทคนิคการจัดกลุ่มดำเนินการโดยตรงบนวัตถุและใช้เทคนิคโครงสร้างข้อมูลต้นไม้แปดกิ่ง (octree) เพื่อเพิ่มความเร็วในการคำนวณผลกราฟ



รูป 1.1 กราฟการกระจายของรูปทรงด้วยวิธี D2 เปรียบเทียบระหว่างรถกับรถถัง

สำหรับการอธิบายแบบการกระจายลักษณะทั่วไประยะทางระหว่างจุดสองจุดที่ถูกสุ่ม ตัวอย่างขึ้นมาบนพื้นผิวของวัตถุนั้นถูกนำมาเอาไปใช้กันอย่างแพร่หลาย วิธีการคำนวณระยะทางของยูคลิดถูกนำมาใช้ประโยชน์ในงานที่เรียกว่า 'D2' (6) ซึ่งใช้การกระจายระยะทางที่คำนวณ ด้วยกลุ่มของจุดบนพื้นผิวของวัตถุซึ่งวิธีนี้ง่ายต่อการคำนวณแต่อย่างไรก็ตามความแม่นยำนั้นมีจำกัด จากในรูปที่ 1.1 แสดงถึงการเปรียบเทียบกันระหว่างวัตถุรถกับรถถังซึ่งจากความเข้าใจวัตถุทั้ง 2 ชนิดนี้ มีความแตกต่างกันแต่การอธิบายด้วย 'D2' นั้นอธิบายว่าคล้ายกัน เหตุผลนั้นเพราะการกระจายระยะทางนั้นไม่ได้รวมพื้นที่ในแต่ละช่วงของเส้นตรง ดังนั้นวัตถุที่มีรูปร่างต่างกันสามารถมีระยะทางการกระจายที่คล้ายกัน คู่ของวัตถุที่ถูกเรียกว่า 'homometric' (11) ซึ่งได้มีงานวิจัยที่ ปรับแต่งวิธี 'D2' โดยแบ่งประเภทช่วงของเส้นระหว่างจุดสองจุดบนขอบเขตรูปร่างตามพื้นที่ของวัตถุซึ่งมีชื่อว่า 'Using Shape Distributions to Compare Solid Models' (2) ถ้าช่วงของเส้นอยู่ภายในวัตถุจะกำหนดให้เป็นช่วงเส้นภายใน ถ้าอยู่นอกก็กำหนดเป็นช่วงเส้นภายนอก อีกนัยหนึ่งถ้าช่วงของเส้นมีส่วนที่อยู่ทั้งภายในวัตถุและภายนอกวัตถุจะถูกจำแนกเป็นเส้นผสมซึ่งวิธีนี้จะทำให้การสามารถคำนวณได้อย่างอิสระมากขึ้น

จากวิธีการอธิบายรูปร่างทั้งสามแบบใน 'Using Shape Distributions to Compare Solid Models' ได้มีการนำเสนอ 'Distance interior Ratio' (DIR) (3) โดยให้ลากเส้นระหว่างจุดสองจุดบนขอบของรูปร่าง DIR ก็คือการคำนวณหาอัตราส่วนระหว่างผลรวมของช่วงเส้นที่อยู่ภายในวัตถุกับระยะทางทั้งหมดระหว่างจุดสองจุด จากนั้นนำผลที่ได้ไปสร้างเป็นกราฟสถิติความถี่(histogram) ของระยะ DIR กับระยะทางทั้งหมดซึ่งถูกใช้เพื่อแทนรูปร่างคล้ายกับวิธี

'D2' ซึ่งจากผลลัพธ์ที่ได้จากงานวิจัยของรูปร่าง 2 มิติโดยแสดงถึงประสิทธิภาพของ DIR เมื่อเปรียบเทียบกับวิธีการอธิบายแบบ 'D2'

ซึ่งในงานวิจัยชิ้นนี้ได้นำเสนอวิธีที่ขยายเพิ่มขึ้นเรียกว่า '3D Distance Interior Ratio' (3DDIR) ซึ่งใช้วิธีการคำนวณการตัดกันระหว่างแผ่นพื้นผิวกับรังสี(ray triangle intersection) (7) เพื่อหาช่วงของเส้นทางระหว่างจุดสองจุดบนรูปทรง 3 มิติซึ่งด้วยลักษณะของการคำนวณจุดตัด ระหว่างรังสีกับแผ่นพื้นผิวจะสามารถบอกทิศทางของแผ่นพื้นผิวเมื่อเมื่อถูกรังสีกระทบ ซึ่งจะสามารถหาตำแหน่งของช่วงของเส้นได้โดยจากการทดลองเบื้องต้นพบว่าการใช้วิธีการคำนวณการตัดกันระหว่างพื้นผิวกับรังสีนั้นทำให้การคำนวณ 3DDIR นั้นใช้เวลานาน เนื่องจากว่าในคำนวณแต่ละรังสีต้องทำการค้นหาแผ่นพื้นผิวที่เป็นจุดตัดซึ่งเกิดมาจากการที่ต้องตรวจสอบแผ่นพื้นผิวทุกแผ่นของวัตถุในหลายๆ รอบ จึงเป็นต้นเหตุให้การคำนวณใช้เวลานานดังนั้นจึงได้มีการนำเสนอโครงสร้างข้อมูลสำหรับแบ่งวัตถุสามมิติขึ้นเพื่อลดการคำนวณจุดตัดลง เพื่อให้การคำนวณ 3DDIR สามารถคำนวณได้รวดเร็วยิ่งขึ้น

โดยนำแนวความคิดการสร้าง Random decision forest (21) และแนวความคิดของการแบ่งวัตถุสามมิติมาประยุกต์และสร้างวิธีการแบ่งข้อมูลวัตถุสามมิติโดยนำข้อมูลแผ่นพื้นผิวของวัตถุมาทำการแบ่งวัตถุสามมิติจากนั้นในขั้นตอนการคำนวณจะเริ่มทำการคัดเลือกแผ่นพื้นผิวที่ต้องการด้วยวิธีAxis Aligned Bounding Box (AABB) และ Minimal Ray Tracer(24) ในการคัดกรองแผ่นพื้นผิวที่จะทำการค้นหาจุดตัด โดยการแบ่งวัตถุนี้มีวัตถุประสงค์เพื่อลดปริมาณการเรียกใช้อัลกอริทึมการคำนวณการตัดกันระหว่างแผ่นพื้นผิวกับรังสีเพื่อเพิ่มความเร็วในการคำนวณอัตราส่วนภายในของวัตถุสามมิติ

1.2 วัตถุประสงค์ของงานวิจัย

1.2.1 เพื่อพัฒนาประสิทธิภาพของวิธีคำนวณ Distance Interior Ratio ให้สามารถใช้คำนวณกับรูปทรง 3 มิติได้

1.2.2 เพื่อพัฒนาประสิทธิภาพของการคำนวณ Distance Interior Ratio ในระบบ 3 มิติให้สามารถคำนวณได้อย่างรวดเร็ว

1.3 ขอบเขตของงานวิจัย

1.3.1 ชุดข้อมูลที่ใช้ในการทดลอง

ข้อมูลที่ใช้ในการทดสอบงานวิจัยชิ้นนี้ ประกอบไปด้วยข้อมูลรูปทรงสามมิติที่รูปทรงหนึ่งวัตถุประกอบไปด้วยชุดข้อมูลตำแหน่งพิกัดและแผ่นพื้นผิว โดยจะทำการสุ่มตัวอย่างตำแหน่งพิกัดไม่เกิน 500 ตำแหน่ง (หากข้อมูลตำแหน่งพิกัดมีไม่ถึง 500 ตำแหน่งให้นำจุดทั้งหมดมาคำนวณ) จากนั้นจะสร้างเส้นที่เป็นไปได้ทั้งหมดโดยคำนวณได้ตั้งสมการต่อไปนี้

$$L = \frac{k(k-1)}{2} \quad (1.1)$$

ตามสมการที่ 1.1 โดย L คือจำนวนเส้นที่เป็นไปได้ทั้งหมดและ k คือจำนวนจุดที่สุ่มตัวอย่างจากนั้นนำเส้นทั้งหมดที่ได้ไปคำนวณหาอัตราส่วนระหว่างระยะทางภายในและระยะทางของเส้นตรง

1.3.2 ชนิดข้อมูลที่ใช้ในการทดลอง

ชนิดของข้อมูลที่ใช้ในการทดลองเป็นชุดของข้อมูลรูปทรง 3 มิติของ Princeton Shape Benchmark (PSB) (10) ที่ประกอบไปด้วยตำแหน่งพิกัดและแผ่นพื้นผิว โดยข้อมูลวัตถุสามมิติที่นำมาใช้ทดลองเป็นวัตถุที่เป็นทรงตันเท่านั้นคือรูปร่างของรูปทรงโดยรวมทั้งหมดต้องเป็นรูปทรงปิดทั้งหมด

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 สามารถพัฒนาประสิทธิภาพของการคำนวณ 'Distance Interior Ratio' ให้สามารถคำนวณกับรูปทรงสามมิติได้

1.4.2 สามารถพัฒนาให้วิธีการคำนวณ 'Distance Interior Ratio' ในรูปทรงสามมิตินั้นสามารถคำนวณได้อย่างรวดเร็ว

1.5 รายละเอียดโครงสร้างวิทยานิพนธ์

รายงานนำเสนอผลการศึกษาวิทยานิพนธ์ฉบับนี้แบ่งออกเป็น 5 บท ได้แก่

- บทที่1 กล่าวถึงความเป็นมาและความสำคัญ วัตถุประสงค์ขอบเขตของงานวิจัย และประโยชน์ที่คาดว่าจะได้รับ
- บทที่2 กล่าวถึงทฤษฎีและงานวิจัยที่เกี่ยวข้อง ประกอบไปด้วยพื้นฐานของคอมพิวเตอร์กราฟิกส์ วิธีการคำนวณการต่างๆที่ใช้ในงานวิจัยนี้ ชนิดโครงสร้างข้อมูลชนิดต่างๆ งานวิจัยการเปรียบเทียบวัตถุ
- บทที่ 3 กล่าวถึงวิธีการดำเนินงานวิจัย รายงานเกี่ยวกับขั้นตอนการดำเนินงานวิจัย เครื่องมือที่นำมาใช้ในงานวิจัย และวิธีที่ใช้ในการทดลองหาประสิทธิภาพเพื่อประเมินความถูกต้อง
- บทที่4 กล่าวถึงผลการทดลอง โดยกล่าวถึงผลการทดลองที่ได้ในแต่ละขั้นตอน และผลการวัดประสิทธิภาพของวิธีที่ได้พัฒนาขึ้น
- บทที่5 กล่าวถึงการสรุปผลการทดลองและข้อเสนอแนะ โดยกล่าวถึงการสรุปผลการศึกษาที่ได้จากการวิจัย พร้อมทั้งข้อเสนอแนะเพื่อปรับปรุงของงานวิจัยในอนาคต

บทที่ 2

วรรณกรรมและงานวิจัยที่เกี่ยวข้อง

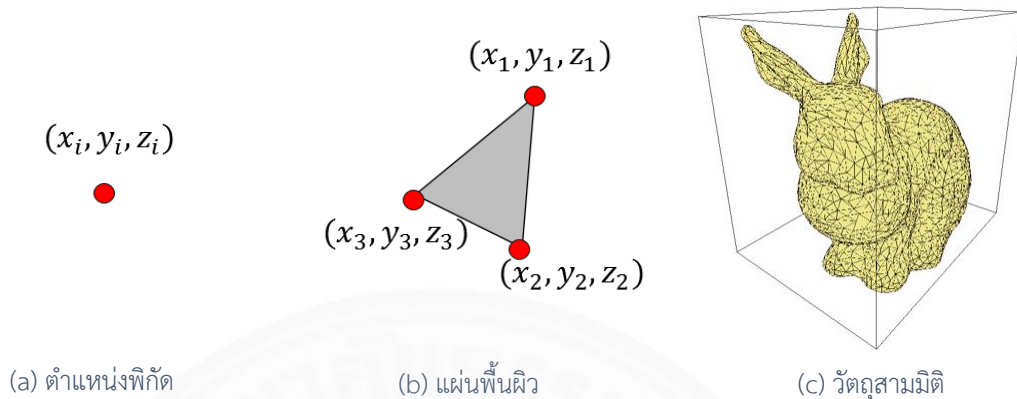
2.1 ทฤษฎีที่เกี่ยวข้อง

ในการจำแนกวัตถุที่เป็นรูปทรงสามมิติต้องอาศัยความรู้ความเข้าใจในเรื่องของคอมพิวเตอร์กราฟิกส์(Computer Graphics) และ คอมพิวเตอร์วิทัศน์(Computer Vision) ซึ่งคอมพิวเตอร์กราฟิกส์คือ องค์ความรู้ของระเบียบวิธีการแก้ปัญหาเชิงคอมพิวเตอร์(Computer Methodology) ที่แก้ปัญหาเกี่ยวกับเรื่องของภาพหรือการแสดงผลภาพ โดยเน้นการประมวลผลข้อมูลด้วยคอมพิวเตอร์ให้ข้อมูลนำเข้าเป็นข้อมูลตัวเลข ตัวอักษร หรือสัญญาณต่าง ๆ แทนตำแหน่งพิกัด สีรูปทรง ความสว่าง ซึ่งขั้นตอนแรกเริ่มต้นด้วยการสร้างแบบจำลอง (modeling) เพื่อแทนความสัมพันธ์ของข้อมูลให้คอมพิวเตอร์สามารถประมวลผลได้ตามด้วยการแปรภาพ หรือที่เรียกกันว่าการเรนเดอร์(rendering) ซึ่งจะแปรผลลัพธ์ออกทางอุปกรณ์อาทิเช่น จอภาพ หรืออุปกรณ์อื่น ๆ

ในการอธิบายวัตถุที่เป็นรูปทรงสามมิติสามารถแบ่งออกเป็น 4 ลักษณะคือ คุณลักษณะเชิงท้องถิ่น(local features) คุณลักษณะรูปร่างโดยรวม (global features) การกระจายคุณลักษณะรูปร่างโดยรวม (global features distribution) และ แผนที่เชิงพื้นที่ (spatial maps) ซึ่งงานวิจัยการคำนวณอัตราส่วนภายในของวัตถุทรงตันสำหรับการรู้จำวัตถุ 3 มิติจะต้องอาศัยความรู้ต่างๆ รวมไปถึงงานวิจัยที่เกี่ยวข้อง หลายๆด้านดังหัวข้อต่อไปนี้

2.1.1 พื้นฐานของคอมพิวเตอร์กราฟิกส์ (Basic of computer graphics)

คอมพิวเตอร์กราฟิกส์ (16) เป็น ศาสตร์องค์ความรู้หนึ่งของระเบียบวิธีการแก้ปัญหาเชิงคอมพิวเตอร์ที่แก้ปัญหาเกี่ยวกับเรื่องของการแสดงผลภาพหรือการแสดงผลภาพที่เน้นการประมวลผลข้อมูลด้วยคอมพิวเตอร์โดยนาชุดข้อมูลไม่ว่าจะเป็นตัวเลข ตัวอักษร หรือ สัญญาณต่างๆ เพื่อแทนตำแหน่งพิกัด สีรูปทรง ความสว่างเพื่อสร้างรูปทรงจำลองขึ้นมาแล้วนำไปแสดงผลผ่านทางจอภาพ หรือ อุปกรณ์แสดงผลอื่นๆ ทั้งนี้ในงานวิจัยนี้ได้ใช้รูปทรงจำลองในระบบ 3 มิติซึ่งเป็นระบบจำลองหนึ่งในคอมพิวเตอร์กราฟิกส์มีองค์ประกอบ 3 ชุดประกอบไปด้วย ความกว้าง ความยาว และความลึก ซึ่งการจำลอง 3 มิตินั้นสามารถทำให้เราสามารถจำลองวัตถุได้มีความคล้ายคลึงกับวัตถุในโลกความเป็นจริง โดยองค์ประกอบของรูปทรงจำลอง 3 มิตินั้นมีหลายองค์ประกอบซึ่งจะแสดงดังในหัวข้อต่อไปนี้



รูปที่ 2.1 องค์ประกอบของวัตถุสามมิติ

2.1.1.1 ตำแหน่งพิกัด (Vertex)

ตำแหน่งพิกัดหรือจุดในระบบสามมิติเป็นโครงสร้างพื้นฐานของรูปทรงในระบบสามมิติทั้งหมด โดยในการที่จะสร้างรูปทรงใดรูปทรงหนึ่งนั้นจะใช้ตำแหน่งพิกัดเป็นพื้นฐานที่ประกอบกันจนได้เป็นรูปร่างหรือรูปทรงหนึ่ง ในระบบสามมิติตำแหน่งพิกัดประกอบไปด้วยตัวเลขจำนวนจริงจำนวนสามชุดซึ่งเราเรียกกันว่าค่าพิกัด โดยค่าแรกเรียกว่าค่าพิกัดแกน x เป็นค่าพิกัดตามแนวความกว้าง ค่าที่สองเรียกว่าค่าพิกัดแกน y เป็นค่าพิกัดตามแนวความสูง และค่าที่สามเรียกว่าค่าพิกัดแกน z เป็นค่าพิกัดตามแนวความลึก ตามรูปที่ 2.1(a) คือตัวอย่างตำแหน่งพิกัดใดๆพิกัดหนึ่งบนพื้นที่ว่างในการจำลองของโลกสามมิติโดยปกติในคอมพิวเตอร์ข้อมูลตำแหน่งพิกัดจะถูกประกาศด้วยค่าตัวเลขจำนวนจริงสามจำนวน

2.1.1.2 แผ่นพื้นผิว (Face)

แผ่นพื้นผิว เป็นส่วนประกอบหนึ่งของรูปร่างหรือรูปทรงต่างๆ ในระบบสามมิติ ซึ่งแผ่นพื้นผิวนี้โดยทั่วไปจะประกอบไปด้วยตำแหน่งพิกัดสามตำแหน่งเชื่อมกันเป็นรูปร่างสามเหลี่ยมในระบบสามมิติตามตัวอย่างในรูปที่ 2.1(b) โดยแผ่นพื้นผิวสามารถประกอบด้วยตำแหน่งพิกัดมากกว่าสามตำแหน่งได้ขึ้นอยู่กับผู้ออกแบบวัตถุสามมิติแต่โดยพื้นฐานแล้วแผ่นพื้นผิวจะประกอบไปด้วยตำแหน่งพิกัดจำนวนสามตำแหน่ง แผ่นพื้นผิวหลายๆ แผ่นพื้นผิวหลายๆ แผ่นนั้นสามารถประกอบกันเป็นพื้นผิว (Surface) ซึ่งจะใช้เพื่อประกอบกันเป็นรูปร่างหรือรูปทรงในระบบ 3 มิติซึ่งแต่ละแผ่นพื้นผิวจะมีค่าเวกเตอร์เรียกว่า normal vector ซึ่งเป็นเวกเตอร์ที่ใช้บอกทางด้านหน้าของแผ่นพื้นผิวอยู่ด้านใด โดยดูจากการเรียงข้อมูลของตำแหน่งพิกัดบนแผ่นพื้นผิวตามกฎมือขวา เมื่อ

นิ้วหัวแม่มือชี้ไปทางทิศใด normal vector จะอยู่ในทิศทางนั้น ซึ่งเมื่อนำแผ่นพื้นผิวไปประกอบเป็นวัตถุสามมิติซึ่งวัตถุสามมิติที่ทำการจำลองขึ้นมา

normal vector ของแต่ละแผ่นพื้นผิวจะมีทิศทางเดียวกันทั้งหมดทุกแผ่นคือทิศทางพุ่งออกจากวัตถุหรือพุ่งเข้าไปภายในวัตถุโดยปกติข้อมูลแผ่นพื้นผิวในคอมพิวเตอร์จะถูกแทนด้วยค่าดัชนีที่บ่งชี้ถึงตำแหน่งพิกัดจำนวนสามตำแหน่ง

2.1.1.3 วัตถุ, รูปทรง (Object)

วัตถุคือ หน่วยใหญ่ที่สุดของโครงสร้างทั้งหมดซึ่งในคอมพิวเตอร์กราฟิกส์นั้น วัตถุนั้นจำลองจากสิ่งที่มีอยู่ในโลกจริงหรือจำลองจากสิ่งที่ไม่จริงก็ได้ซึ่งวัตถุก็สามารถเป็นชิ้นๆเดียว หรือสามารถที่จะเป็นวัตถุหลายๆชิ้นมารวมกันเป็นวัตถุเดียวได้ซึ่งวัตถุนั้นเกิดจากการประกอบกันของข้อมูลตำแหน่งพิกัดและแผ่นพื้นผิว โดยจะประกอบกันเป็นวัตถุ, รูปทรงหนึ่ง โดยสามารถแสดงให้เห็นได้ผ่านทางจอภาพหรือสื่อวัตถุอื่นๆ ด้วยวิธีการเร็นเดอร์โดยในรูปที่ 2.1(c) คือตัวอย่างวัตถุที่ถูกจำลองขึ้นมา

2.1.1.4 รังสี (Ray)

รังสีคือ โครงสร้างหนึ่งที่ใช้ในการบอกทิศทางของการจำลองในโลกสามมิติยกตัวอย่างเช่น ทิศทางมุมมองที่ใช้มองวัตถุการสะท้อนของแสงบนพื้นผิววัตถุตำแหน่งของแสงที่จำลองขึ้นมาว่าตกกระทบบนส่วนใดของวัตถุและอื่นๆอีกมากมาย ซึ่งรังสีนั้นจะใช้การคำนวณในระบบปริภูมิเวกเตอร์ในการคำนวณวิธีการต่างๆ ซึ่งการคำนวณในระบบปริภูมิเวกเตอร์นั้นเป็นระบบการคำนวณพื้นฐานของระบบคอมพิวเตอร์กราฟิกส์ ทั้งนี้รังสีนั้นมีองค์ประกอบ 2 ส่วนคือ จุดกำเนิด(Origin) และทิศทาง(Direction) โดยจุดกำเนิดก็คือตำแหน่งพิกัดใดๆ พิกัดหนึ่งบนโลกการจำลองสามมิติและทิศทางคือค่าเวกเตอร์สามมิติซึ่งเป็นค่าที่ใช้บอกทิศทางของรังสีโดย normal vector ของแผ่นพื้นผิวนั้นก็เป็นรังสีชนิดหนึ่งเช่นกัน

2.1.2 วิธีการคำนวณที่นำมาใช้ในงานวิจัย

ในงานวิจัยนี้ผู้วิจัยได้ศึกษาวิธีการคำนวณจากหลายงานวิจัยเพื่อนำมาพัฒนาโดยมีรายละเอียดดังต่อไปนี้

2.1.2.1 Ray Triangle Intersection

Ray Triangle Intersection (7) เป็นอัลกอริทึมการหาจุดตัดระหว่างรังสีกับแผ่นพื้นผิวที่มีลักษณะเป็นสามเหลี่ยมหรือแผ่นพื้นผิวที่ประกอบไปด้วยตำแหน่งพิกัดสามตำแหน่ง ซึ่งอัลกอริทึมรับข้อมูลของแผ่นพื้นผิวทั้งสามตำแหน่งและจุดกำเนิดและทิศทางของรังสีเป็นค่าตั้งต้น ผลลัพธ์ที่ได้จะได้ค่าเวกเตอร์ดังนี้ $(t \ u \ v)^T$ โดยที่ t คือระยะทางระหว่างจุดกำเนิดของรังสีกับจุดตัดบนแผ่นพื้นผิวสามเหลี่ยม (u, v) แทนตำแหน่งภายในสามเหลี่ยม ซึ่งอัลกอริทึมนี้เป็นอัลกอริทึมที่ใช้งานอย่างแพร่หลายในสาขาคอมพิวเตอร์กราฟิกส์ซึ่งอัลกอริทึมที่ใช้คำนวณได้อย่างรวดเร็วระหว่างรังสีและแผ่นพื้นผิว (7)

2.1.2.2 Axis Aligned Bounding Box

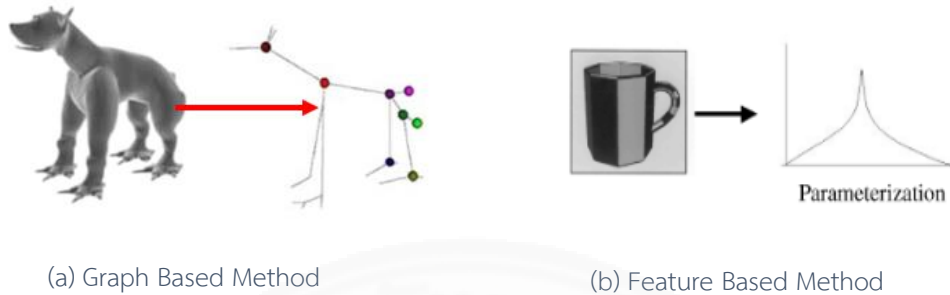
Axis Aligned Bounding Box (AABB) เป็นการสร้างขอบเขตของวัตถุขึ้นมาโดยจะนำค่าน้อยที่สุดและค่ามากที่สุดในแต่ละค่าของตำแหน่งพิกัดของวัตถุมาสร้างเป็นขอบเขต ดังตัวอย่างในรูปที่ 2.1(c) ซึ่งวิธี AABB เป็นวิธีการตรวจสอบ collision detection คือตรวจสอบการชนกันระหว่างสองวัตถุโดยจะนำขอบเขตของวัตถุสองวัตถุมาตรวจสอบว่า มีการซ้อนทับกันหรือไม่หากมีการซ้อนทับกันนั้นหมายความว่าวัตถุทั้งสองเกิดการชนกันมีฉะนั้นแล้ววัตถุทั้งสองไม่ได้ชนกัน

2.1.2.3 Minimal Ray Tracer

Minimal Ray Tracer (24) เป็นอัลกอริทึมที่คำนวณจุดตัดระหว่างรังสีกับวัตถุทรงสี่เหลี่ยม โดยมีเป้าหมายคือไม่ต้องคำนวณหาตำแหน่งพิกัดที่ตัดกันระหว่าง รังสีและ ฉากของวัตถุทรงสี่เหลี่ยม โดยจะนำค่า t ซึ่งคือระยะทางระหว่างจุดกำเนิดของรังสีและฉากมาคำนวณเพื่อตรวจสอบว่ารังสีตัดกับฉากหรือไม่ซึ่งถ้ารังสีตัดกับฉากด้านใดด้านหนึ่งอย่างน้อยหนึ่งด้าน นั้นหมายความว่ารังสีเกิดจุดตัดบนวัตถุทรงสี่เหลี่ยมมีฉะนั้นหมายความว่ารังสีไม่เกิดจุดตัดบนวัตถุทรงสี่เหลี่ยม

2.1.2.4 Randomized Decision Forests

Randomized Decision Forests (21) เป็นอัลกอริทึมการแบ่งข้อมูลจากงานวิจัย Real-Time Human Pose Recognition (21) โดยเป็นอัลกอริทึมหนึ่งี่สร้างต้นไม้การตัดสินใจ (random forests) โดยในงานวิจัยนี้ได้นำแนวความคิดของงานวิจัยนี้มาสร้างอัลกอริทึมการแบ่งวัตถุ ซึ่งจะอธิบายขั้นตอนวิธีในบทที่3 โดยจะแทนแผ่นพื้นผิวของวัตถุเสมือนข้อมูลในต้นไม้การตัดสินใจ



รูปที่ 2.2 ตัวอย่างการแปลงวัตถุเพื่อใช้ในการอธิบาย

2.1.3 วิธีการอธิบายวัตถุสามมิติ

จากการเปรียบเทียบวัตถุด้วยวิธีเปรียบเทียบตำแหน่งต่อตำแหน่งที่ทำงานโดยนำตำแหน่งพิกัดของวัตถุที่หนึ่งมาคำนวณระยะทางกับตำแหน่งพิกัดของวัตถุที่สอง ซึ่งทำให้วิธีเปรียบเทียบวัตถุนั้นใช้เวลานานเนื่องจากมีกระบวนการในการคำนวณเยอะและยังต้องระวังเรื่อง ขนาด, ตำแหน่ง, ทิศทางการหมุน ซึ่งสิ่งต่างๆเหล่านี้มีผลในการเปรียบเทียบวัตถุด้วยวิธีพื้นฐานทำให้การคำนวณนั้นใช้เวลานานไม่สามารถนำมาใช้ได้จริง

ในปัจจุบันได้มีวิธีการซึ่งช่วยให้สามารถคำนวณเปรียบเทียบวัตถุสามมิติได้อย่างมีประสิทธิภาพมากกว่าและคำนวณได้รวดเร็ว ซึ่งจากการที่ศึกษาวิธีการที่ผ่านมาสามารถจำแนกวิธีการได้ดังต่อไปนี้

2.1.3.1 วิธีการอธิบายวัตถุสามมิติด้วยกราฟ

Graph based method (12) เป็นวิธีการเปรียบเทียบวัตถุสามมิติโดยการนำวัตถุสามมิติที่ต้องการเปรียบเทียบ ไปแปลงเป็นรูปร่างกราฟโครงสร้างรูปแบบหนึ่ง ซึ่งกราฟที่แปลงนั้นจะใช้ในการอธิบายแทนวัตถุสามมิติและเวลาจะทำการเปรียบเทียบวัตถุจะนำกราฟของวัตถุทั้งสองมาเปรียบเทียบกัน โดยในรูปที่ 2.2(a) เป็นรูปตัวอย่างในการแปลงวัตถุให้เป็นกราฟ

2.1.3.2 วิธีการอธิบายวัตถุสามมิติด้วยลักษณะ

Feature based method (12) เป็นวิธีการเปรียบเทียบโดยการนำลักษณะลักษณะพื้นฐานของวัตถุเช่น ปริมาตร, การกระจายของระยะทาง, ฯลฯ ซึ่งวิธีการนี้ได้นำลักษณะพื้นฐานเหล่านี้นำมาแปลงเป็นเวกเตอร์หรือเมทริกซ์เพื่อใช้ในการอธิบายวัตถุสามมิติซึ่งวิธีการเปรียบเทียบวัตถุจะนำเวกเตอร์หรือเมทริกซ์ที่ใช้อธิบายวัตถุในการเปรียบเทียบ ซึ่งสามารถคำนวณได้อย่างรวดเร็ว โดยวิธีการอธิบายวัตถุด้วยลักษณะยังสามารถจำแนกได้ดังต่อไปนี้

- Local Feature เป็นวิธีการเปรียบเทียบวัตถุโดยนำส่วนหนึ่งของวัตถุไปแปลงเป็นเมทริกซ์ ยกตัวอย่างเช่นนำสายของกระเป่าถือโดยจะนำส่วนของสายกระเป่าไปแปลงเป็นเมทริกซ์ทดสอบจากนั้นจะนำไปเปรียบเทียบกับเมทริกซ์ของวัตถุอื่น ซึ่งถ้าหากพบว่าเมทริกซ์นั้นเหมือนกับระหว่างวัตถุแสดงว่าวัตถุที่ทำการเปรียบเทียบนั้นเป็นกระเป่าแบบเดียวกัน
- Spatial Map เป็นการสร้างแผนภูมิเชิงพื้นที่แทนตำแหน่งพื้นที่ของวัตถุ ซึ่งแผนที่จะขึ้นตรงกับพื้นที่กายภาพหรือส่วนต่างๆ ของวัตถุ ซึ่งถูกจัดในลักษณะที่รักษาความสัมพันธ์กับตำแหน่งของวัตถุ แผนที่เชิงพื้นที่โดยทั่วไปไม่คงที่สำหรับวัตถุที่มีการหมุนยกเว้นแผนภูมิที่ออกแบบมาโดยเฉพาะ ดังนั้นโดยทั่วไปจะทำการ 'normalization' ก่อนการสร้างแผนภูมิ
- Global Feature เป็นวิธีการอธิบายด้วยคุณลักษณะทั่วไปจะอธิบายลักษณะทั่วไปของวัตถุ 3 มิติ ตัวอย่างของการอธิบายรูปทรงโดยใช้วิธีนี้เป็นพื้นฐานนั้นเช่น การเก็บสถิติของขอบเขตหรือขนานของวัตถุ อัตราส่วนของขนาดต่อพื้นผิว การแปลงปริมาตรหรือขอบเขตของวัตถุ
- Global Feature Distribution เป็นวิธีการอธิบายวัตถุซึ่งต่อยอดมาจากวิธีจำแนกวัตถุด้วยลักษณะทั่วไป โดยจะนำลักษณะต่างๆไปของวัตถุเช่น การกระจายของระยะทางระหว่างตำแหน่งพิกัดของวัตถุ, ขอบเขตต่างๆของวัตถุและอื่นๆอีกมากมาย โดยในงานวิจัยที่จัดทำขึ้นนี้ก็有一部分หนึ่งของวิธีการอธิบายวัตถุวิธีนี้ โดยจะทำการคำนวณการกระจายอัตราส่วนของระยะทางภายในกับระยะทางระหว่างตำแหน่งพิกัดของวัตถุ

2.2 งานวิจัยที่เกี่ยวข้อง

ในวิทยานิพนธ์นี้จำเป็นต้องมีพื้นฐานความรู้เกี่ยวกับการอธิบายวัตถุสามมิติที่จำแนกด้วยลักษณะโดยอธิบายจากการกระจายของลักษณะทั่วไปของวัตถุและความรู้เกี่ยวกับโครงสร้างข้อมูลของวัตถุสามมิติโดยจะแบ่งงานวิจัยที่เกี่ยวข้องที่ได้ทำการศึกษามาเป็นดังต่อไปนี้

2.2.1 งานวิจัยเกี่ยวกับการอธิบายรูปร่างสามมิติ

2.2.1.1 Shape Distribution (D2)

Shape Distribution (6) เป็นงานวิจัยนี้ได้นำเสนอวิธีการและวิธีวิเคราะห์การคำนวณเพื่ออธิบายรูปร่างสามมิติหัวใจสำคัญของวิธีนี้คือแทนรูปร่างสามมิติแต่ละวัตถุด้วย 'Shape Distributions' เพื่อวัดผลคุณลักษณะของวัตถุแต่ละวัตถุแรงบันดาลใจแรกสำหรับวิธีนี้เพื่อลดปัญหาของจับคู่วัตถุเพื่อเปรียบเทียบความเป็นไปได้จากการกระจาย ซึ่งวิธีการนี้เรียบง่ายกว่าวิธีดั้งเดิมที่ผ่านมาที่จับคู่วัตถุสามมิติที่จำเป็นต้องทำการบันทึกคุณลักษณะที่เหมาะสม

ซึ่งทางผู้วิจัยได้ค้นพบความแตกต่างระหว่างการกระจายตัวของรูปร่างที่ใช้ทดลองซึ่งให้วิธีการที่มีประสิทธิภาพสำหรับการแบ่งแยกวัตถุซึ่งสามารถนำไปประเมินได้อย่างรวดเร็วและทำให้นำไปใช้เพื่อวิเคราะห์รูปร่างหาความคล้ายคลึงกันระหว่างวัตถุซึ่งงานวิจัยชิ้นนี้ได้บรรยายการจำแนกวัตถุของแบบจำลองสามมิติ โดยทั่วไปงานวิจัยนี้จะเป็นวิธีพื้นฐานในการเปรียบเทียบประสิทธิภาพในการจำแนกวัตถุสามมิติ

2.2.1.2 Shape Classification Using the Inner-Distance

Shape Classification Using the Inner-Distance (9) เป็นงานวิจัยนี้ได้นำเสนอ 'inner-distance' เพื่อสร้างการอธิบายรูปร่างขึ้นซึ่งมีประสิทธิภาพในการจับภาพวัตถุแต่ละส่วน ซึ่ง 'inner-distance' นิยามความขนาดที่สั้นที่สุดระหว่างตำแหน่งในรูปร่าง ซึ่งได้แสดงให้เห็นว่ามีประสิทธิภาพมากขึ้นกว่าวิธีการกระจายโดยการคำนวณระยะทางของยูคลิดในการจำแนกวัตถุเพราะจำแนกเป็นส่วนที่ละส่วนไป ซึ่งแสดงให้เห็นว่าข้อมูล 'inner-distance' สามารถนำมาใช้เพื่ออธิบายวัตถุได้อย่างแม่นยำสำหรับรูปภาพที่มีความซับซ้อนในรูปร่างทุกๆส่วน

นอกจากนี้ข้อมูลพื้นผิวตามเส้นทางที่สั้นที่สุดสามารถใช้เพื่อปรับปรุงการจำแนกรูปร่าง ด้วยความคิดนี้ผู้วิจัยจึงได้จำแนกวิธีออกมาสามวิธี วิธีแรกคือเชื่อมระหว่าง 'inner

distance' กับการปรับขนาดหลายมิติเพื่อใช้ในการอธิบายรูปร่าง วิธีที่สองใช้'inner-distance' เพื่อสร้างตัวอธิบายรูปร่างโดยใช้พื้นฐานของบริบทรูปร่าง ส่วนที่สามคือขยายมาจากวิธีที่สอง โดยพิจารณาข้อมูลพื้นผิวบนเส้นทางที่สั้นที่สุด ซึ่งวิธี 'inner distance' สามารถเพิ่มประสิทธิภาพการเปรียบเทียบระหว่างรูปร่างเมื่อเทียบกับอัลกอริทึมอื่น

13

2.2.1.3 Using Shape Distributions to Compare Solid Models

Using Shape Distributions to Compare Solid Models (2) เป็นงานวิจัยที่บรรยายวิธีเปรียบเทียบวัตถุโดยใช้พื้นฐานของวิธี'Shape distributions' ซึ่งเป็นวิธีที่ใช้ทั่วไปในการเปรียบเทียบระหว่างวัตถุสองมิติซึ่งในงานนี้ใช้เพื่อเปรียบเทียบประมาณวัตถุที่ถูกรูปในงานกราฟิกส์ในงานวิจัยนี้ได้ปรับเทคนิคเพื่อใช้ในการเปรียบเทียบวัตถุสามมิติซึ่งทางผู้วิจัยได้แสดงถึงวิธีการขยายพื้นฐานการกระจายตัวในการใช้ทดลองกับชุดข้อมูลและอธิบายวิธีการกระจายตัวเพื่อตรวจสอบวัตถุโดยตรง สุดท้ายได้แสดงให้เห็นว่าเทคนิคเหล่านี้สามารถนำใช้รวมกันเพื่อหาผลลัพธ์ที่ต่างๆ กันได้

2.2.1.4 Distance Interior Ratio (DIR)

Distance Interior Ratio (DIR) (3) เป็นงานวิจัยนี้ได้นำเสนอวิธีการคำนวณหาอัตราส่วนระหว่างผลรวมช่วงของเส้นที่อยู่ภายในกับระยะทางของเส้นตรงระหว่าง ตำแหน่งสองตำแหน่งบนพื้นผิวของวัตถุ 2 มิติโดยใช้ชื่อว่า 'Distance Interior Ratio' (DIR) จากนั้นนำผลลัพธ์ที่ได้ไปคำนวณและสร้างเป็นกราฟสถิติเพื่อใช้ในการอธิบายรูปร่างสองมิติซึ่งเพิ่มประสิทธิภาพมากขึ้นเพราะทำการคำนวณข้อมูลช่วงของเส้นในทุกๆ ส่วน ทำให้มีประสิทธิภาพเพิ่มขึ้นในการอธิบายวัตถุรูปร่างสองมิติ และงานวิจัยนี้ได้เป็นแรงบันดาลใจของงานวิจัยการคำนวณอัตราส่วนภายในของวัตถุทรงตันสำหรับการรู้จำวัตถุสามมิติโดยได้นำแนวคิดไปประยุกต์ให้สามารถคำนวณวัตถุสามมิติ

2.2.2 งานวิจัยโครงสร้างข้อมูลในคอมพิวเตอร์กราฟิกส์

2.2.2.1 Binary Space Partition(BSP)

Binary Space Partition (17) เป็นวิธีการแบ่งวัตถุออกเป็นสองส่วนโดยการสร้างฉากจำลองขึ้นมา โดยฉากจำลองนั้นทำการแบ่งวัตถุออกเป็นสองส่วน โดยฉากที่แบ่งสร้างโดยการนำตำแหน่งพิกัดสองตำแหน่งเป็นตำแหน่งที่ใช้ในการสร้างฉากขึ้น โดยส่วนที่แบ่งออกมาแล้วนั้นจะแทนด้วย node โดยจะทำการแบ่งแบบวนซ้ำจนไม่สามารถแบ่งได้อีก โดยฉากที่สร้างขึ้นมีลักษณะเป็นฉากสี่เหลี่ยมผืนผ้าในลักษณะที่ตั้งฉากหรือไม่ตั้งฉากกับแกนในปริภูมิเวกเตอร์สามมิติก็ได้โดยจะขึ้นอยู่กับตำแหน่งพิกัดที่เลือกในการแบ่ง โดยโครงสร้างข้อมูลชนิด

นี้มีประโยชน์ในการช่วยลดปริมาณวัตถุที่ต้องฉายภาพโดยจะเลือกจะเฉพาะส่วนที่ควรฉายขึ้นจริงเท่านั้น ซึ่งทำให้การเรนเดอร์ภาพวัตถุสามมิติทำได้รวดเร็วขึ้น โดยได้มีการใช้โครงสร้างข้อมูลนี้ในเกมสามมิติอย่างแพร่หลาย

14

2.2.2.2 K-Dimension Tree(K-d tree)

K-Dimension Tree (19) เป็นโครงสร้างข้อมูลที่ทำให้การแบ่งวัตถุออกทีละสองส่วนโดยสามารถทำการแบ่งวัตถุในมิติที่เท่าใดก็ได้โดยในแต่ละขั้นของการแบ่งในแต่ละมิติจะเริ่มแบ่งจากแกนที่หนึ่งไปเรื่อยๆยกตัวอย่างเช่น การแบ่งวัตถุในสามมิติจะแบ่งโดยเริ่มจากแกน x , y และ z ตามลำดับ โดยในแต่ละขั้นของการแบ่งจะนำค่าตำแหน่งพิกัดในส่วนที่ทำการแบ่งมาหาค่าเฉลี่ยเพื่อที่จะหาจุดที่จะสร้างฉากเพื่อทำการแบ่งวัตถุโดยจะหยุดทำการแบ่งก็ต่อเมื่อข้อมูลใน node เหลือตำแหน่งพิกัดเพียงพิกัดเดียว ซึ่งลักษณะการแบ่งจะคล้ายกับโครงสร้างข้อมูลแบบ BSP แต่ฉากของ K-d tree นั้นจะตั้งฉากกับแกนที่กำลังทำการแบ่งเสมอ

2.2.2.3 Octree

Octree (20) เป็นโครงสร้างข้อมูลชนิดหนึ่งโดยการแบ่งจะคล้ายกับโครงสร้างแบบ K-d tree โดยจะทำการหาตำแหน่งกึ่งกลางของวัตถุจากนั้นจะทำการสร้างฉากสองฉากตัดกันโดยที่ตั้งฉากกัน ซึ่งจะทำให้เกิดส่วนที่แบ่งขึ้นมาถึงแปดส่วน โดยจะทำการแบ่งแบบวนซ้ำไปเรื่อยๆ จนกว่า node ของข้อมูลจะเก็บข้อมูลตำแหน่งพิกัดเพียงตำแหน่งเดียว โดย Octree เป็นโครงสร้างข้อมูลที่นิยมใช้กันมากในงานวิจัยคอมพิวเตอร์กราฟิกส์และเทคโนโลยีการสร้างเกม เนื่องจากสามารถลดเวลาในการเรนเดอร์ภาพสามมิติได้เป็นอย่างดี

2.3 ตารางการเปรียบเทียบงานวิจัย

2.3.1 ตารางการเปรียบเทียบงานวิจัยเกี่ยวกับการอธิบายรูปทรงสามมิติ

จากการศึกษาค้นคว้างานวิจัยเกี่ยวข้องกับการอธิบายวัตถุจะสามารถสรุปได้ดังตารางต่อไปนี้

ตารางที่ 2.1 เปรียบเทียบงานวิจัยที่เกี่ยวข้องกับการอธิบายวัตถุ

ที่	ชื่องานวิจัย	ค.ศ.	เทคนิคที่ใช้	ข้อดี	ข้อเสีย
1	Shape Distributions	2002	D2	เป็นวิธีคำนวณได้รวดเร็ว	ไม่สามารถเจาะจงวัตถุได้หากมีความคล้ายกันสูง
2	Using Shape Distributions to Compare Solid Models	2002	Shape Distributions	มีความหลากหลายในการจำแนกวัตถุเพราะเก็บข้อมูลได้หลากหลาย	ไม่ได้คำนวณเจาะจงระยะทาง
3	Shape Classification Using the Inner-Distance	2007	Inner-Distance	เจาะจงเฉพาะส่วนภายในทำให้ความแม่นยำสูง	คำนวณเฉพาะส่วนที่อยู่ภายในจึงขาดข้อมูลส่วนภายนอก
4	Distance Interior Ratio	2015	DIR	คำนวณเจาะจงระยะทางทำให้มีประสิทธิภาพ	คำนวณเฉพาะในรูปร่าง 2 มิติ

2.3.2 ตารางการเปรียบเทียบงานวิจัยโครงสร้างข้อมูลในคอมพิวเตอร์กราฟิกส์

จากการศึกษาค้นคว้างานวิจัยเกี่ยวข้องกับโครงสร้างข้อมูลในคอมพิวเตอร์กราฟิกส์ จะสามารถสรุปได้ดังตารางต่อไปนี้

ตารางที่ 2.2 เปรียบเทียบงานวิจัยที่เกี่ยวข้องกับโครงสร้างข้อมูลในคอมพิวเตอร์กราฟิกส์

ที่	ชื่องานวิจัย	ค.ศ.	เทคนิคที่ใช้	ข้อดี	ข้อเสีย
1	Binary Space Partition	1969 1980	BSP	- Plane ในการแบ่งวางได้อิสระ - เร็นเดอร์ภาพได้รวดเร็ว	- แบ่งวัตถุด้วยข้อมูลแผ่นพื้นผิวได้ไม่สมดุลง่ายหากวัตถุไม่สมมาตรกัน
2	K-Dimensional Tree	1975	K-d Tree	- Plane แบ่ง Space ตั้งฉาก - ประยุกต์กับโปรแกรมค้นหาได้ง่าย	
3	Octree	1980	Octree	- สามารถหยุดการสร้างต้นไม้ได้ตามต้องการ - เร็นเดอร์ภาพได้รวดเร็ว	

บทที่ 3 วิธีการวิจัย

3.1 แนวทางการแก้ปัญหา

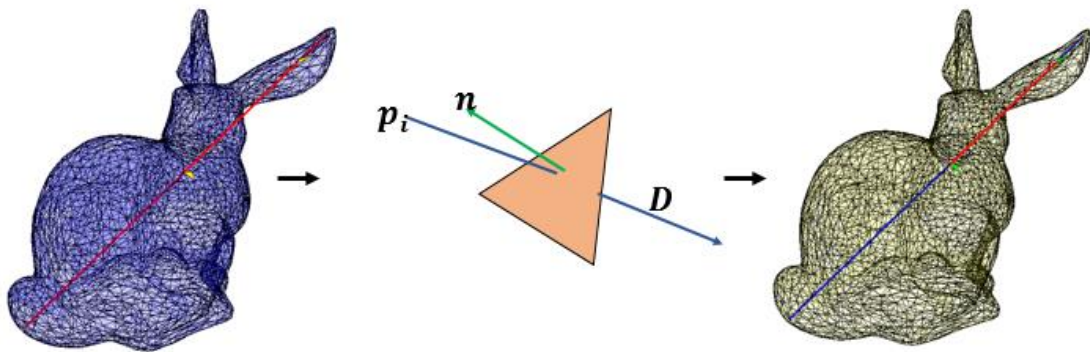
เนื่องจากการเปรียบเทียบด้วยวิธีพื้นฐานของวัตถุสามมิตินั้นใช้เวลานาน ซึ่งในปัจจุบันได้มีวิธีต่างๆมากมายซึ่งช่วยลดเวลาในการเปรียบเทียบวัตถุสามมิติและสามารถนำไปใช้ประยุกต์ในการค้นหาวัตถุสามมิติได้อย่างมีประสิทธิภาพ ซึ่งจากการที่ศึกษาวิธีการที่ผ่านมาสามารถจำแนกวิธีการได้หลักๆ ดังนี้

- Graph based method (12) เป็นวิธีการเปรียบเทียบวัตถุสามมิติโดยการนำวัตถุสามมิติที่ต้องการเปรียบเทียบไปแปลงเป็นรูปร่างกราฟ โครงสร้างรูปแบบหนึ่ง จากนั้นทำการเปรียบเทียบวัตถุสามมิติโดยนำกราฟที่ทำการแปลงไปเปรียบเทียบระหว่างวัตถุสองวัตถุ
- Feature based method (12) เป็นวิธีการเปรียบเทียบโดยการนำลักษณะลักษณะพื้นฐานของวัตถุเช่น ปริมาตร, การกระจายของระยะทาง ฯลฯ ซึ่งวิธีการนี้ได้นำลักษณะพื้นฐานเหล่านี้นำมาแปลงเป็นเวกเตอร์รูปแบบหนึ่งซึ่งจะใช้ในการเปรียบเทียบระหว่างวัตถุสามมิติซึ่งวิธีสามารถประยุกต์ใช้งานในการค้นหาได้รวดเร็ว

จากการศึกษางานวิจัยที่ผ่านมาผู้วิจัยได้สนใจวิธีการเปรียบเทียบวัตถุสามมิติโดยนำลักษณะของวัตถุมาใช้ในการเปรียบเทียบวัตถุสามมิติซึ่งจากการศึกษาได้นำแนวคิดที่ได้จากงานวิจัย Distance Interior Ratio(3) ซึ่งเป็นวิธีคำนวณการกระจายของอัตราส่วนระยะทางภายในแปลงเป็นกราฟเมทริกซ์แล้วนำมาเปรียบเทียบ ซึ่งวิธีการนี้เป็นวิธีที่ใช้ในการค้นหาวัตถุสองมิติซึ่งในงานวิจัยนี้ได้นำแนวคิดไปประยุกต์ให้สามารถเปรียบเทียบในวัตถุสามมิติ

3.2 เตรียมข้อมูลที่ใช้ในการทดลอง

ข้อมูลที่ใช้ในการทดลองเป็นข้อมูลของวัตถุสามมิติซึ่งแต่ละชุดประกอบไปด้วยข้อมูลตำแหน่งพิกัดและแผ่นพื้นผิวซึ่งประกอบเป็นวัตถุสามมิติซึ่งในการทดลองได้ใช้ชุดวัตถุสามมิติของ Princeton Shape Benchmark(PSB) (10) โดยแบ่งข้อมูลออกเป็นสามชุดโดย ชุดแรกที่ใช้ในการเปรียบเทียบเวลาในการคำนวณจำนวน 42 วัตถุ,ชุดที่สองใช้ในการดูแนวโน้มของการ



รูปที่ 3.1 ตัวอย่างขั้นตอนโดยรวมของระบบ

ทำงานจำนวน 236 วัตถุและชุดที่สามเพื่อใช้ในการวัดผลความถูกต้องและเปรียบเทียบเวลาในการคำนวณจำนวน 166 วัตถุเพื่อลดเวลาในการคำนวณในการแปลงวัตถุสามมิติเป็นกราฟเวกเตอร์ที่ใช้อธิบายวัตถุในแต่ละวัตถุได้ทำการสุ่มตำแหน่งพิกัดจำนวน 500 ตำแหน่งถ้าวัตถุที่ทำการทดสอบ มีจำนวนตำแหน่งพิกัดมากกว่า 500 ตำแหน่ง มิฉะนั้นจะไม่ทำการสุ่มถ้าวัตถุมีตำแหน่งพิกัดน้อยกว่า 500 ตำแหน่ง

3.3 ขั้นตอนวิธีของระบบการคำนวณโดยไม่แบ่งวัตถุ

ขั้นตอนแต่ละขั้นตอนของระบบเป็นดังต่อไปนี้

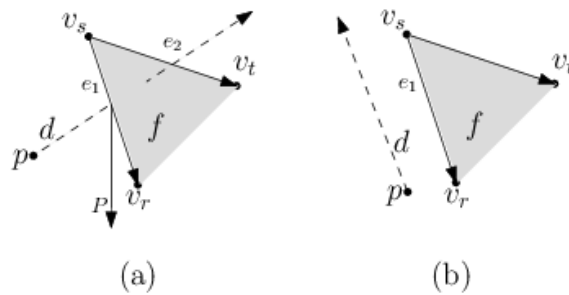
1. Input อ่านข้อมูลวัตถุสามมิติรับข้อมูลตำแหน่งพิกัดและแผ่นพื้นผิว
2. Process ที่ทำการแปลงวัตถุสามมิติเป็นกราฟเมทริกซ์มีขั้นตอนมีดังต่อไปนี้
 - (a) สุ่มตำแหน่งพิกัดบนวัตถุสามมิติและสร้างเส้นตรง (P_i, P_j)
 - (b) คำนวณระยะทาง Distance Interior Ratio ของแต่ละเส้นตรง
 - (c) บันทึกข้อมูลและนำข้อมูลไปแปลงเป็นกราฟเมทริกซ์
3. Output กราฟเมทริกซ์ที่ใช้อธิบายแทนวัตถุสามมิติ

โดยรูปที่ 3.1 จะแสดงภาพรวมของระบบการคำนวณ Distance Interior Ratio

3.4 วิธีการคำนวณ Distance Interior Ratio

ขั้นตอนในการคำนวณ Distance Interior Ratio มีขั้นตอนตาม pseudo code ดังต่อไปนี้

1. Input Sample lines (P_i, P_j) from Object
2. for line in lines do:

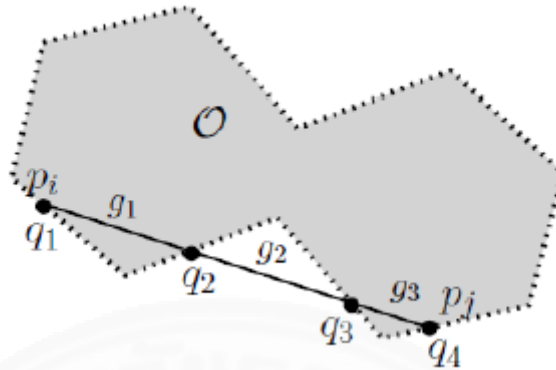


รูปที่ 3.2 ตัวอย่างการตัดกันระหว่างรังสีและแผ่นพื้นผิว

3. Use 'Ray Triangle Intersection' to find intersection points between line and Object by check every face in Object.
4. Classify the segments of the line which is laying inside or outside the Object.
5. Find Summation of distance of the segments which are laying inside the object store as d_{in} .
6. Compute the ratio by compute d_{in} and distance of the line store as d_{ir}
7. Save the d_{ir} and distance of the line to result file

ซึ่งจาก pseudo code ของระบบการคำนวณระยะทาง Distance Interior Ratio กับวัตถุสามมิติสามารถอธิบายเป็นรายละเอียดได้ดังนี้

1. ทำการสร้างเส้นตรงจากตำแหน่งพิกัดที่ทำการส่องขึ้นมา โดยเส้นตรงที่สร้างขึ้นจะสร้างรังสีขึ้นโดยกำหนดจุดกำเนิดและทิศทางของรังสี จากนั้นทำการ normalize เวกเตอร์ทิศทางของรังสีให้เป็นเวกเตอร์หนึ่งหน่วย
2. ทำการหาจุดตัดของเส้นตรงกับวัตถุสามมิติโดยนำรังสีของเส้นตรงตรวจสอบกับแผ่นพื้นผิวทุกแผ่นบนวัตถุสามมิติเพื่อหาจุดตัดว่าเกิดจุดตัดใดบ้างบนวัตถุโดยนำอัลกอริทึมการตัดกันระหว่างรังสีกับแผ่นพื้นผิวโดยลักษณะที่เกิดการตัดกันมีลักษณะเป็นดังรูปที่ 3.2 ในภาพ (a) หากมีการตัดกันแต่ถ้าหากรังสีไม่ตัดกับแผ่นพื้นผิวจะเป็นแบบใน



รูปที่ 3.3 ตัวอย่างการแบ่งช่วงของเส้นตรง

- รูปที่ 3.2 ภาพ (b) โดยจากการคำนวณของอัลกอริทึมเมื่อทำการคำนวณระหว่าง normal vector ของแผ่นพื้นผิวและเวกเตอร์ทิศทางของรังสีจะสามารถตรวจสอบว่ารังสีตัดกันได้หรือไม่โดยอัลกอริทึมจะคืนค่า t มาซึ่งค่า t คือระยะทางจากจุดกำเนิดของเส้นตรงจนถึงจุดตัดจุดหนึ่งระหว่างเส้นตรงกับวัตถุถ้าหากอัลกอริทึมคืนค่า t ที่มากกว่าศูนย์และน้อยกว่าระยะทางของเส้นตรงแสดงว่าพบจุดตัดหนึ่งจุดตัดบนเส้นตรง
3. นำจุดตัดที่ค้นหาได้ทั้งหมดไปคำนวณ Distance Interior Ratio ของเส้นตรง โดยจุดตัดที่ค้นหาได้ทั้งหมดจะแบ่งเส้นตรงออกเป็นส่วนๆ ตามในรูปที่ 3.3 เส้นตรงจะถูกแบ่งออกเป็น 3 ส่วนด้วยจุดตัดสี่จุด กำหนด $q1, q2, q3$ และ $q4$ แทนจุดตัดที่สี่จุดตามลำดับและกำหนด $g1, g2$ และ $g3$ แทนส่วนของเส้นตรงที่ถูกแบ่งด้วยจุดตัด เราจะทำการพิจารณาว่าส่วนของเส้นตรงแต่ละส่วนว่าเป็นส่วนที่อยู่ภายในหรือภายนอกวัตถุโดยการพิจารณาจากจุดตัดที่แบ่งส่วนของเส้นตรงนั้น ยกตัวอย่างเช่นส่วนของเส้นตรง $g1$ จะมีจุดตัด $q1$ และ $q2$ โดยพิจารณาจุดตัด $q1$ เวกเตอร์ทิศทางของรังสีของเส้นตรงจะพุ่งเข้าวัตถุและ normal vector ของแผ่นพื้นผิวมีทิศทางสวนทางกัน โดยสามารถตรวจสอบได้โดยการนำเวกเตอร์ทั้งสองมา dot product กันจะได้ค่าที่มากกว่าศูนย์แสดงว่าที่จุด $q1$ เป็นจุดที่เส้นตรงกำลังพุ่งเข้าวัตถุส่วนที่ตำแหน่ง $q2$ ทิศทางของรังสีจะมีทิศทางเดียวกับ normal vector ของแผ่นพื้นผิวเมื่อทำการ dot product กันจะได้ค่าที่น้อยกว่าศูนย์แสดงว่าที่ตำแหน่ง $q2$ รังสีของเส้นตรงกำลังพุ่งออกจากวัตถุดังนั้นจึงสรุปได้ว่าส่วนของเส้นตรง $g1$ นั้นเป็นส่วนที่อยู่ภายในวัตถุ ในอีกกรณีเมื่อเราพิจารณาสองส่วนของเส้นตรง $g2$ ที่ตำแหน่ง $q2$ นั้นรังสีของเส้นตรง

มีลักษณะพุ่งออกจากวัตถุจากการคำนวณและที่ตำแหน่ง $q3$ รังสีมีลักษณะพุ่งเข้าวัตถุจึงสรุปได้ว่าส่วนของเส้นตรง $g2$ นั้นอยู่ภายนอกวัตถุจากตัวอย่างทั้งสองนี้จึงสามารถสรุปเป็นทฤษฎีบท 2 ทฤษฎีดังนี้

- (a) ถ้าจุดเริ่มต้นของส่วนของเส้นตรงรังสีมีทิศทางสวนทางกับ normal vector ของแผ่นพื้นผิวและที่จุดปลายรังสีมีทิศทางเดียวกับ normal vector นั้นสรุปได้ว่าส่วนของเส้นตรงนั้นเป็นส่วนที่อยู่ภายในวัตถุ
- (b) ถ้าจุดเริ่มต้นของส่วนของเส้นตรงรังสีมีทิศทางเดียวกับ normal vector ของแผ่นพื้นผิวและที่จุดปลายรังสีมีทิศสวนทางเดียวกับ normal vector นั้นสรุปได้ว่าส่วนของเส้นตรงนั้นเป็นส่วนที่อยู่ภายในวัตถุ

4. เมื่อจำแนกส่วนของเส้นตรงทั้งหมดก็คำนวณอัตราส่วนระหว่างผลรวมระยะทางของส่วนของเส้นตรงที่อยู่ภายในและระยะทางของเส้นตรง โดยระยะทางของส่วนของเส้นตรงและระยะทางของเส้นตรงสามารถคำนวณด้วยการคำนวณระยะทางแบบยูคลิด กำหนด d_{ir} แทนอัตราส่วนระหว่างผลรวมระยะทางส่วนของเส้นตรงที่อยู่ภายในและระยะทางของเส้นตรง d_{in} แทนผลรวมระยะทางของส่วนของเส้นตรงที่อยู่ภายใน d คือระยะทางของเส้นตรง โดยสามารถกำหนดเป็นสมการได้ดังนี้

$$d_{ir} = \frac{d_{in}}{d} \quad (3.1)$$

จากขั้นตอนทั้งหมดให้เก็บข้อมูล d_{ir} และ d ของเส้นตรงทั้งหมดเพื่อทำการสร้างกราฟแผนภูมิสถิติที่ใช้อธิบายวัตถุสามมิติ

3.5 การสร้างแผนภูมิสถิติเพื่ออธิบายรูปร่างวัตถุสามมิติ

กำหนดตัวเลขจำนวนเต็มจำนวน 2 ค่า C, R ซึ่งจะทำการแบ่งเป็นพื้นที่ซึ่งมีขนาดเป็น $C \times R$ โดยแกน x จะแบ่งให้มีขนาด C บล็อกและแกน y จะแบ่งให้มีขนาด R บล็อก โดยในแต่ละบล็อกตามแกน x จะแบ่งให้มีขนาดเป็น $[\max(D) - \min(D)/c]$ และขนาดของบล็อกในแกน y ในแต่ละบล็อกมีขนาดเป็น $[\max(IR) - \min(IR)/r]$ โดยที่ D คือเซตของข้อมูลระยะทางในแต่ละเส้นระหว่างคู่ของตำแหน่งพิกัด และ IR คือเซตของข้อมูลอัตราส่วนภายในในแต่ละเส้นระหว่างคู่ของตำแหน่งพิกัด กำหนด (u, v) คือค่าดัชนีของแผนภูมิสถิติของแกน x และ y ตามลำดับ ซึ่งช่องว่างของแต่ละบล็อกนั้นสามารถเขียนได้เป็น (x_u, x_{u+1}) และ (y_v, y_{v+1}) และให้ $h(u, v)$ คือตัวเลขที่ตกลงในช่วง (x_u, x_{u+1}) และ (y_v, y_{v+1}) และเพื่อหลีกเลี่ยงปัญหาเรื่องขนาดจึงทำการ normalize

$h(u, v)$ โดยการหารด้วยจำนวนข้อมูลทั้งหมด ซึ่งจะสามารถเขียนเป็นสมการดังนี้

$$h(u, v) = \frac{2}{k(k-1)} \sum_{i=1}^k \sum_{j=1}^{k-1} f(d_{ij}, d_{ir}(i, j), (x_u, x_{u+1}), (y_v, y_{v+1})) \quad (3.2)$$

โดยที่

$$f(d_{ij}, d_{ir}(i, j), (x_u, x_{u+1}), (y_v, y_{v+1})) = \begin{cases} 1, & \text{if } d_{ij} \in [x_u, x_{u+1}] \\ & \text{and } d_{ir} \in (y_v, y_{v+1}) \\ 0, & \text{otherwise} \end{cases} \quad (3.3)$$

ซึ่งการอธิบายวัตถุสามมิติจะถูกแสดงโดย $H(P_S)$ คือแผนภูมิสถิติขนาด $c \times r$ ของ

$h(u, v)$ โดยที่ $0 \leq u \leq c - 1$ และ $0 \leq v \leq r - 1$

3.6 เปรียบเทียบความคล้ายกันระหว่างวัตถุสามมิติ

ในการเปรียบเทียบความคล้ายคลึงกัน กำหนด A คือแผนภูมิสถิติของวัตถุที่หนึ่งและ B และแผนภูมิสถิติของวัตถุที่สอง ให้ $S(A, B)$ คือระยะทางระหว่างแผนภูมิสถิตินี้ซึ่งคำนวณได้ดังนี้

$$S(A, B) = \sum_{i=0}^{r-1} \sum_{j=0}^{c-1} |h_A(i, j) - h_B(i, j)| \quad (3.4)$$

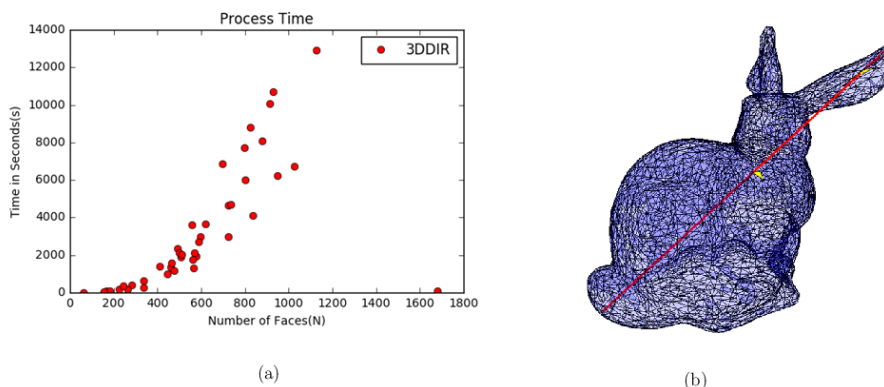
โดยตามสมการด้านบน $S(A, B)$ คือค่าความคล้ายคลึงของวัตถุสามมิติ 2 วัตถุซึ่งถ้าค่าที่คำนวณเข้าใกล้ศูนย์มากเท่าใดแสดงว่าวัตถุทั้งสองมีความเหมือนกันมาก

ตารางที่ 3.1 ตารางผลลัพธ์การเปรียบเทียบการจำแนกวัตถุ

วิธีการอธิบาย	First Tier	Second Tier	Nearest Neighbor	Average Time(s)
D2	20.24	30.73	34.34	3.28
3DDIR	22.96	34.6	42.16	32168.19

3.7 ผลลัพธ์การวัดผลประสิทธิภาพของระบบเบื้องต้น

ทำการทดลองวัดประสิทธิภาพในการจำแนกวัตถุโดยใช้ข้อมูลของวัตถุสามมิติของ PSB จำนวน 166 วัตถุโดยการทดลองที่หนึ่งทำการวัดผลประสิทธิภาพการจำแนกด้วยวิธีคำนวณวัดผลด้วย First Tier, Second Tier และ Nearest Neighbor ซึ่งอ้างอิงมาวิธีการวัดผลจากงานวิจัย 'Shape Distributions'(D2) (6) โดย First Tier คือจำนวนร้อยละของ $k - 1$ ที่วัตถุทดสอบจะคล้ายกับวัตถุในคลาสตัวมันเอง โดย k คือจำนวนวัตถุในคลาส Second Tier คือจำนวนร้อยละของ $2(k - 1)$ ที่วัตถุทดสอบจะคล้ายคลึงกับวัตถุในคลาสตัวมันเอง โดย k คือจำนวนวัตถุในคลาสและ Nearest Neighbor คือร้อยละของวัตถุทดสอบจะคล้ายคลึง



รูปที่ 3.4 กราฟวัดเวลาการทำงานในการคำนวณของ 3DDIR

กับวัตถุในคลาสเป็นอันดับแรก โดยในตารางที่ 3.1 ทำการทดสอบประสิทธิภาพกับงานวิจัย Shape Distributions(D2) (6) ซึ่งเป็นงานวิจัยมาตรฐานในการเปรียบเทียบประสิทธิภาพ ซึ่งจากผลลัพธ์ที่ได้แสดงในตารางพบว่า 3DDIR เมื่อวัดผลแบบ First Tier นั้นมากกว่า D2 ถึงร้อยละ 22.96 และ Second Tier ถึงร้อยละ 34.6 และ Nearest Neighbor นั้นวัดผลได้มากกว่า D2 ถึงร้อยละ 42.16 (25)

จากการทดลองพบปัญหาเกี่ยวกับเวลาในการสร้างกราฟสถิติสำหรับอธิบายวัตถุของ 3DDIR ตามตารางที่ 3.1 จะพบว่าเวลาเฉลี่ยในการคำนวณเพื่อสร้างกราฟนั้นใช้เวลาถึง 32168.19 วินาที ซึ่งปัญหาที่เกิดขึ้นเกิดมาจากการที่ในการคำนวณกระจายของ Distance Interior Ratio ในสามมิติ

3.8 ปัญหาเบื้องต้นของระบบ

จากการทดลองเบื้องต้นของ 3DDIR การจะสร้างกราฟสถิติเพื่ออธิบายวัตถุสามมิตินั้น ใช้เวลานาน โดยในรูปที่ 3.4 (a) คือกราฟความสัมพันธ์ระหว่างเวลาในการคำนวณเพื่อสร้างกราฟสถิติและจำนวนแผ่นพื้นผิว โดยจากกราฟจะเห็นได้ว่ายิ่งจำนวนแผ่นพื้นผิวมีจำนวนมากขึ้นสามารถทำให้เวลาในการคำนวณมากขึ้น ในรูปที่ 3.4 (b) คือรูปตัวอย่างในการหาจุดตัดกับวัตถุสามมิติโดยเส้นตรงสีแดงคือเส้นตรงที่ใช้ในการคำนวณหาเพื่อหาช่วงของระยะทาง Distance Interior Ratio แผ่นพื้นผิวน้ำเงินคือแผ่นพื้นผิวที่ไม่เกิดจุดตัดกับเส้นตรงสีแดง และแผ่นสีเหลืองซึ่งมีจำนวนน้อยคือแผ่นที่เกิดจุดตัดกับเส้นตรงสีแดง

ซึ่งจากรูปที่ 3.4 (b) จะเห็นได้ว่าจุดตัดที่เกิดขึ้นในการคำนวณนั้นเกิดขึ้นน้อยมากเมื่อเทียบกับสัดส่วนของวัตถุสามมิติโดยในขั้นตอนการคำนวณของ 3DDIR นั้นจำเป็นต้องตรวจสอบทุกแผ่นพื้นผิวทั้งหมดจึงเป็นสาเหตุที่ทำให้ในการสร้างกราฟสถิติในแต่ละครั้งใช้เวลาใน

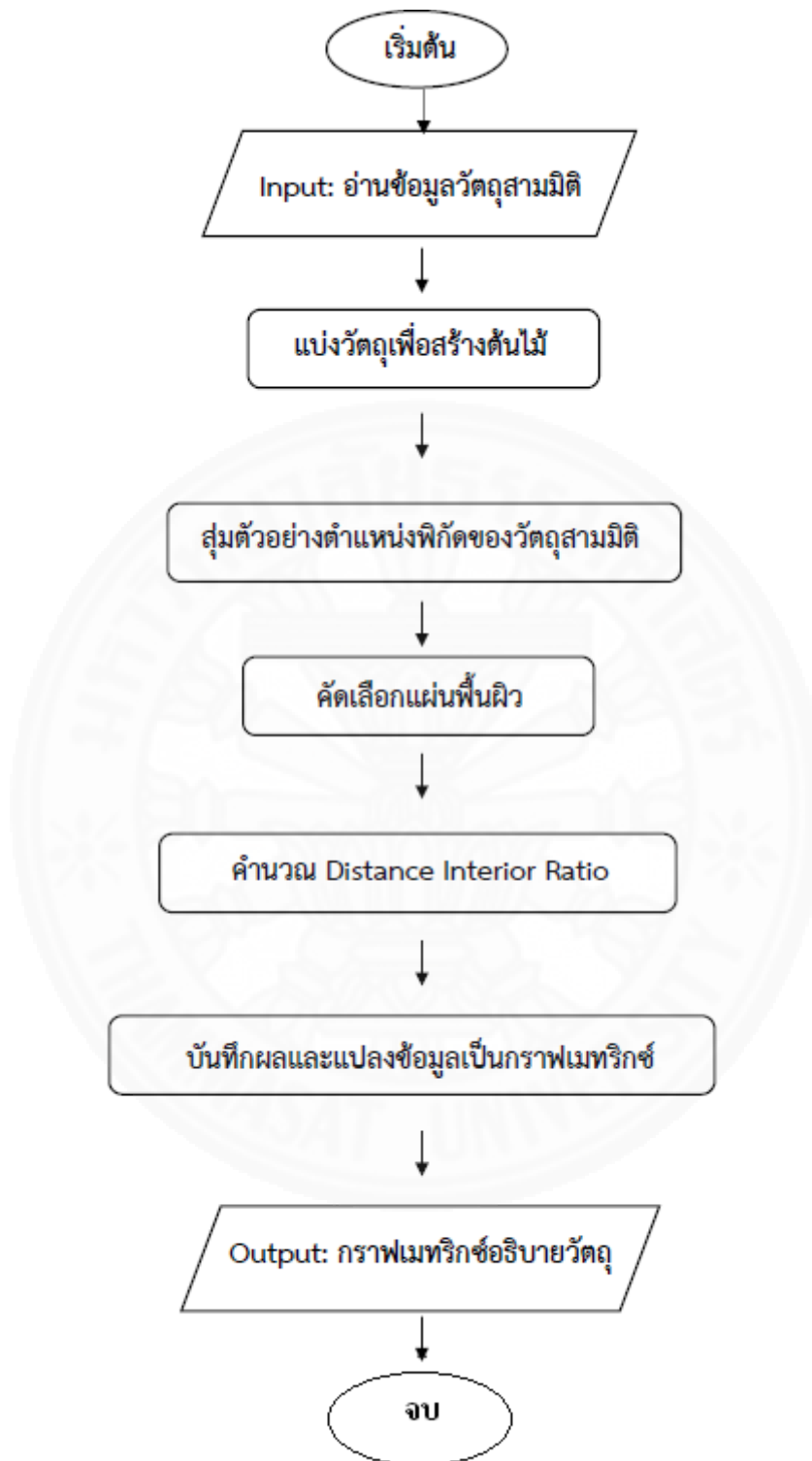
การคำนวณนาน ซึ่งจากสาเหตุของปัญหาดังกล่าวทำให้เกิดแนวคิดที่ว่าหากระบบไม่จำเป็นต้องตรวจสอบทุกแผ่นพื้นผิวมันจะช่วยลดเวลาในการคำนวณกราฟสถิติลงได้อย่างมาก จึงมีการนำโครงสร้างข้อมูลของวัตถุสามมิติมาใช้เพื่อให้สามารถลดเวลาในการคำนวณลงโดยจะกล่าวในหัวข้อถัดไป

3.9 แนวทางการแก้ปัญหาโดยวิธีการแบ่งวัตถุสามมิติ

จากปัญหาเบื้องต้นนั้นทางผู้วิจัยจึงเพิ่มส่วนการแบ่งวัตถุสามมิติเพื่อที่จะลดเวลาในการคำนวณหาจุดตัดลง โดยนำแนวคิดโครงสร้างข้อมูลในการแบ่งวัตถุสามมิติที่ได้ศึกษามาดังต่อไปนี้

- Binary Space Partition(BSP) (17) เป็นวิธีสำหรับการแบ่งวัตถุออกเป็นสองส่วนด้วย ฉากจำลองขึ้นมา โดยฉากจำลองนั้นจะทำการแบ่งพื้นที่ออกเป็นสองส่วน มีประโยชน์มากในการช่วยลดปริมาณการ render ในคอมพิวเตอร์กราฟิกส์
- K-dimension tree (K-d tree) (19) เป็นวิธีที่ทำการแบ่งวัตถุออกเป็นสองส่วนโดยคำนวณตำแหน่งพิกัดที่แบ่งวัตถุโดยการนำค่าตำแหน่งพิกัดในแกนที่จะทำการแบ่งมาหาค่าเฉลี่ยจากนั้นจะทำการแบ่งวัตถุออกเป็นสองส่วน ลักษณะจะคล้ายกับ BSP แต่ฉากของ K-d tree จะตั้งฉากกับแกนที่กำลังจะทำการแบ่งวัตถุและจะทำการแบ่งแบบวนซ้ำไปเรื่อยๆ จนกว่า node ของ K-d tree ที่ตำแหน่งล่าสุดเหลือตำแหน่งพิกัดเพียงพิกัดเดียว
- Octree (20) เป็นโครงสร้างข้อมูลคล้ายกับวิธี K-d tree โดยเป็นวิธีการแบ่งวัตถุสามมิติออกเป็นสองส่วนในการแบ่งแต่ละครั้ง โดยจะทำการแบ่งแบบวนซ้ำไปเรื่อยๆ จนกว่า node ตำแหน่งล่าสุดจะเหลือเพียงตำแหน่งพิกัดเดียว

และได้ศึกษาขั้นตอนวิธีการสร้างต้นไม้การตัดสินใจ (21) โดยจากการศึกษาทั้งหมดได้นำแนวความคิดมาสร้างโครงสร้างข้อมูลเพื่อใช้ในการแบ่งวัตถุสามมิติโดยจะบรรยายขั้นตอนวิธีและวิธีการแบ่งวัตถุในหัวข้อถัดไป



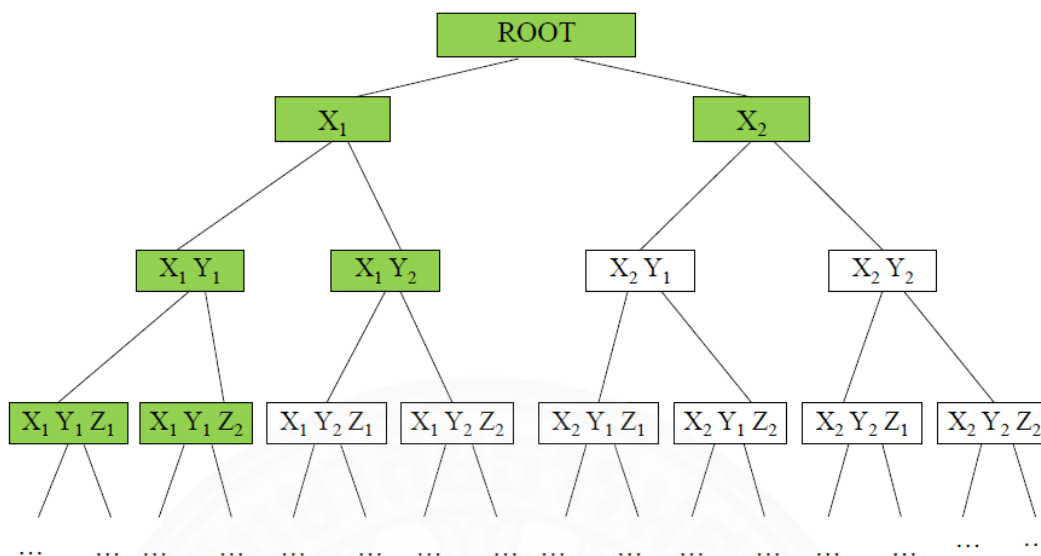
รูปที่ 3.5 แผนผังการทำงานโดยรวมของระบบ

3.10 ขั้นตอนวิธีของระบบ

โดยจากแนวทางการแก้ปัญหาในหัวข้อที่แล้ว ได้ทำการปรับปรุงขั้นตอนการคำนวณ 3DDIR ดังตัวอย่างในรูปที่ 3.5 ซึ่งสามารถอธิบายได้ดังนี้

1. Input อ่านข้อมูลวัตถุสามมิติ
2. Process ที่ทำการแปลงวัตถุสามมิติเป็นกราฟเมทริกซ์ซึ่งขั้นตอนมีดังต่อไปนี้
 - (a) แบ่งวัตถุสามมิติเพื่อสร้างต้นไม้ (ขั้นตอนการเตรียมข้อมูล (Pre-Processing))
 - (b) สุ่มตำแหน่งพิกัดบนวัตถุสามมิติ
 - (c) ค้นหาแผ่นพื้นผิว
 - (d) คำนวณระยะทาง Distance Interior Ratio
 - (e) บันทึกข้อมูลแล้วนำไปแปลงเป็นกราฟเมทริกซ์
3. Output กราฟเมทริกซ์ซึ่งเป็นตัวแทนอธิบายวัตถุสามมิติ

โดยจะทำการอธิบายรายละเอียดขั้นตอนการแบ่งวัตถุเพื่อสร้างต้นไม้ (Pre-Processing) และรายละเอียดขั้นตอนการค้นหาแผ่นพื้นผิวในหัวข้อที่ 3.11 และ 3.12 ตามลำดับ ซึ่งขั้นตอนของระบบที่ทำการพัฒนาขึ้นนี้จะเรียกว่า ‘ระบบการคำนวณแบบแบ่งวัตถุและทำงานแบบเป็นลำดับ’ (3DDIR Single-thread Partition)



รูปที่ 3.6 ตัวอย่างโครงสร้างต้นไม้

3.11 ขั้นตอนวิธีการแบ่งวัตถุและการสร้างต้นไม้

ในงานวิจัยชิ้นนี้ได้นำเสนอการแบ่งวัตถุและสร้างโครงสร้างข้อมูล เพื่อใช้ในการลดเวลาในการคำนวณของระบบ การแบ่งวัตถุคือการแบ่งวัตถุออกเป็นส่วนๆ โดยมีวิธีการแบ่งได้หลายวิธี ยกตัวอย่างเช่นการสร้าง plane มาทำการตัดแบ่งวัตถุเช่น BSP, K-d tree หรือการแบ่งโดยใช้จุดกึ่งกลางของวัตถุเช่น Octree ซึ่งการแบ่งวัตถุในงานวิจัยนี้เป็นแบบวิธีใช้เส้นในการแบ่งซึ่งหมายถึงการหาตำแหน่งที่ดีที่สุดในการแบ่งและวางเส้นตรงพาดแบ่งวัตถุออกเป็นสองส่วน ซึ่งเป็นวิธีการแบ่งคล้ายกับวิธี K-d tree ซึ่งขั้นตอนวิธีการแบ่งวัตถุจะแสดงเป็น pseudo code ดังต่อไปนี้

1. Input: set of vertices, set of faces, axis
2. Sample V from set of vertices
3. for v in V do:
 4. Check v with every face in set of faces and compute NL, NR, ND
 5. Compute score by score = $|NL-NR|+ND$ and store v that make minimum score
6. Porting object at balance vertex that make minimum score

จาก pseudo code ซึ่งเป็นขั้นตอนวิธีการในการแบ่งวัตถุโดยจะอธิบายเป็นรายละเอียดขั้นตอนการแบ่งวัตถุซึ่งมีรายละเอียดดังต่อไปนี้

1. รับข้อมูลส่วนของวัตถุเป็นตำแหน่งพิกัดและแผ่นพื้นผิวและแกนที่จะทำการแบ่ง
2. ทำการสุ่มตัวอย่างตำแหน่งพิกัดจากข้อมูลตำแหน่งที่รับเข้ามา
3. ทำการค้นหาตำแหน่งที่ดีที่สุดในการแบ่งวัตถุโดยตำแหน่งที่ดีที่สุดขึ้นกับเงื่อนไขสองเงื่อนไขซึ่งตั้งเป็นทฤษฎีบทดังต่อไปนี้
 - (a) ตำแหน่งที่ทำให้จำนวนแผ่นพื้นผิวเท่ากันทั้งด้านซ้ายและด้านขวา
 - (b) ตำแหน่งที่ทำให้จำนวนแผ่นพื้นผิวที่แบ่งได้ทั้งสองด้านมีจำนวนน้อยที่สุด (แผ่นพื้นผิวในส่วนนี้เป็นแผ่นพื้นผิวที่มีส่วนทับกับเส้นแบ่ง)

4. เมื่อได้ตำแหน่งที่ดีที่สุดก็จะทำการแบ่งวัตถุออกเป็นสองส่วน

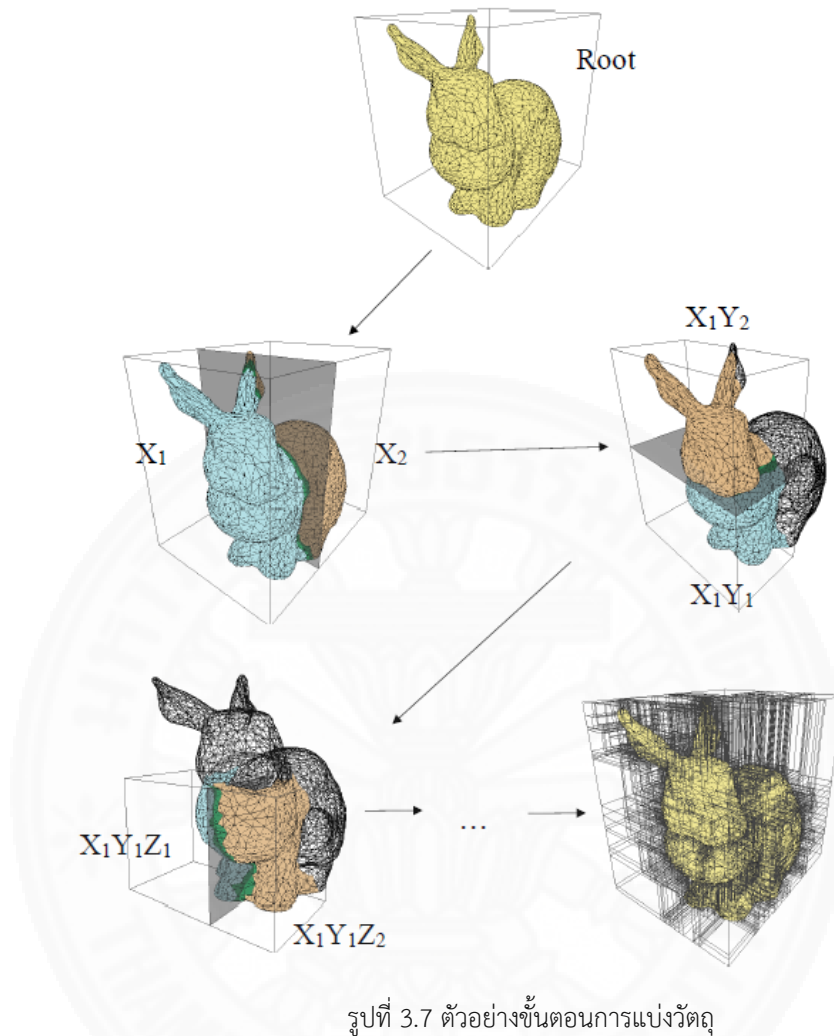
จากขั้นตอนวิธีการแบ่งเบื้องต้นจะเริ่มแบ่งวัตถุโดยเริ่มที่แกน x , y และ z ตามลำดับ

ซึ่งจากขั้นตอนการแบ่งวัตถุข้างต้นจะสามารถอธิบายขั้นตอนการสร้างต้นไม้ซึ่งมีลักษณะตามรูปที่

3.6 ซึ่งลำดับการสร้าง node จะมีลำดับแบบวิธีการค้นหาตามแนวกว้าง (breadth first search)

โดยจะอธิบายรายละเอียดขั้นตอนการสร้างได้ต่อไปนี้

1. เมื่อระบบรับข้อมูลวัตถุสามมิติให้กำหนดชุดข้อมูลนั้นเป็นรากของต้นไม้โครงสร้างข้อมูล โดยรากของต้นไม้จะทำการเก็บข้อมูลตำแหน่งพิกัด, แผ่นพื้นผิวและขอบเขตของวัตถุ
2. นำข้อมูลวัตถุในรากไปแบ่งตามแกน x ตามขั้นตอนวิธีการแบ่งวัตถุ
3. เมื่อทำการแบ่งวัตถุให้สร้าง child's node ขึ้นมาสอง node ต่อจากรากจากนั้นทำการ copy ข้อมูลตำแหน่งและแผ่นพื้นผิวที่ถูกกำหนดให้ส่งไป node แรกโดยกำหนดจากเส้นแบ่งวัตถุและทำการกำหนด boundary กับ node แรกใหม่จาก boundary ของรากและตำแหน่งที่ทำการแบ่งวัตถุ ซึ่ง node ที่สองจะทำในลักษณะคล้ายๆกัน
4. parent's node ที่เมื่อทำการแบ่งวัตถุแล้วจะทำการลบข้อมูลตำแหน่งและแผ่นพื้นผิวออกและจะเก็บข้อมูลเพียงจุดแบ่ง (threadhold) และขอบเขตของวัตถุก่อนการแบ่งเท่านั้น
5. ทำการวนซ้ำแบ่งวัตถุโดยนำ child's node เข้าคิวเพื่อแบ่งวัตถุต่อไป โดย child's node ที่เป็นผลลัพธ์จากการแบ่งแกน x จะนำไปแบ่งตามแกน y ต่อไป และเช่นเดียวกัน ผลลัพธ์การแบ่งตามแนวแกน y จะนำไปแบ่งตาม z และจะวนซ้ำกลับมาแบ่งในแกน x อีกครั้งจนกว่าจะบรรลุเงื่อนไขหยุดการสร้าง
6. ระบบจะหยุดการวนซ้ำสร้างต้นไม้เมื่อไม่มี node อยู่ในคิวการแบ่งวัตถุแล้วโดย child's node ที่จะไม่นำเข้าคิวการแบ่งวัตถุนั้นมีเงื่อนไขดังต่อไปนี้



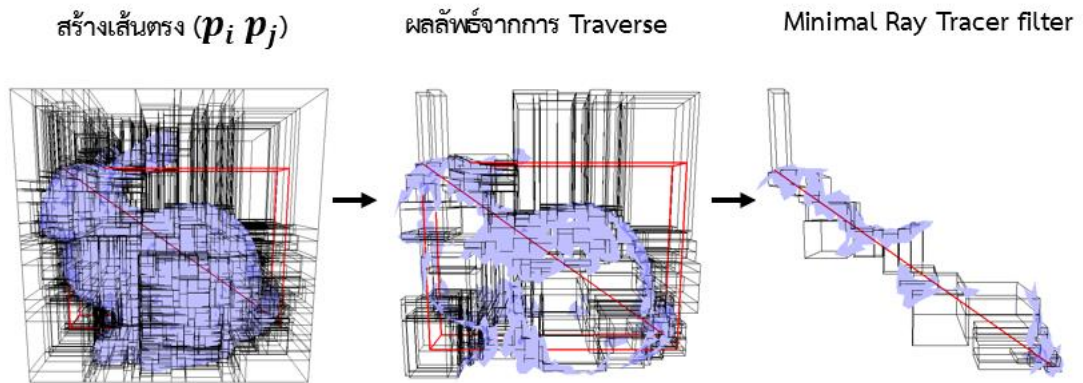
(a) เมื่อจำนวนแผ่นพื้นผิวที่เก็บใน node มีจำนวนน้อยกว่า 10 แผ่น

(b) เมื่อ node มีความสูงถึงชั้นที่กำหนดของต้นไม้

ซึ่งจากขั้นตอนวิธีการสร้างต้นไม้ที่กล่าวมาข้างต้นสามารถสรุปการเก็บข้อมูลได้ดังนี้

1. parent's node จะทำการเก็บข้อมูล boundary และจุดแบ่งของวัตถุ (threshold)
2. child's node เฉพาะ child's node ชั้นล่างสุดที่ไม่สามารถแบ่งวัตถุได้อีกตามเงื่อนไข จะเก็บข้อมูล boundary และข้อมูลแผ่นพื้นผิวเท่านั้น

ตามตัวอย่างในรูปที่ 3.7 จะเป็นตัวอย่างขั้นตอนการแบ่งวัตถุและในขั้นตอนนี้ของระบบจะเป็นขั้นตอนในการเตรียมข้อมูลวัตถุสามมิติสำหรับขั้นตอนในการค้นหาแผ่นพื้นผิว

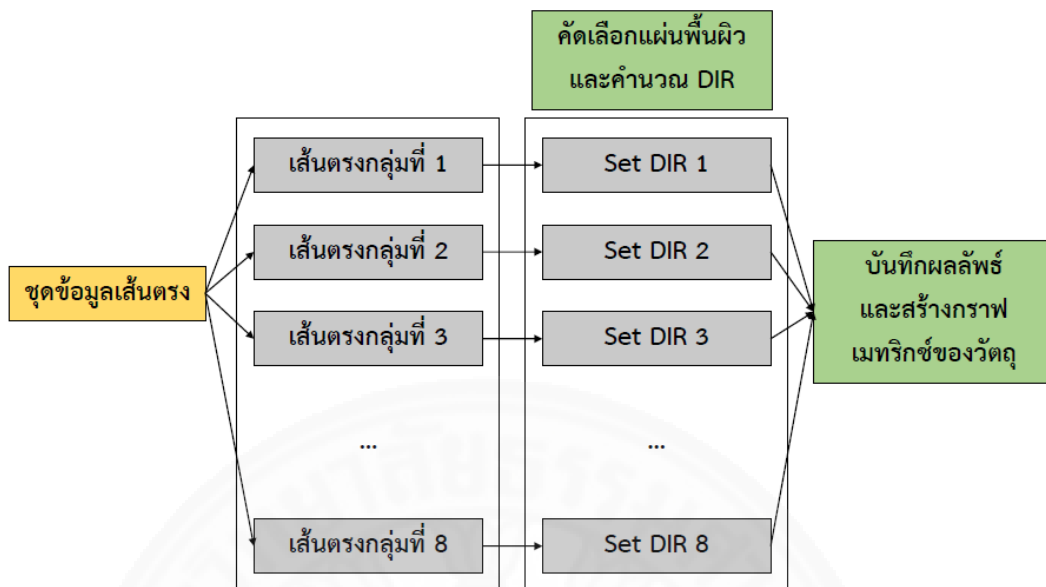


รูปที่ 3.8 ภาพรวมของกระบวนการค้นหาแผ่นพื้นผิว

3.12 ขั้นตอนการค้นหาแผ่นพื้นผิวเพื่อนำไปคำนวณหาจุดตัด

ในขั้นตอนการค้นหาแผ่นพื้นผิวนั้นจะทำการท่องต้นไม้ด้วยวิธีการค้นหาตามแนวกว้าง (breadth first search) กับต้นไม้โดยมีรายละเอียดดังขั้นตอนต่อไปนี้

1. ทำการสร้างเส้นตรง (P_i, P_j) ขึ้นมา
2. ทำการสร้าง boundary ของเส้นตรงที่กััด max และ min ของเส้นตรง
ดังตัวอย่างกรอบสีแดงในรูปที่ 3.8
3. ทำการท่องต้นไม้แบบ breadth first search โดยตรวจสอบ node ที่ต้องการด้วยการตรวจสอบ boundary ของเส้นตรงกับ boundary ของ node ซึ่งเป็นเรียกวิธีการตรวจสอบ boundary นี้ว่า Axis Aligned Bounding Box (AABB) ซึ่งเมื่อทดสอบ boundary ของเส้นตรงและ node พบว่าเกิดการ intersection แล้ว node ที่ทำการตรวจสอบมี child's node ให้นำ child's node เข้าคิวเพื่อรอค้นหาต่อไปมีฉะนั้นแล้วให้เก็บผลลัพธ์ node ไปตรวจสอบถัดไป
4. เมื่อทำการท่องต้นไม้เสร็จสิ้นผลลัพธ์ที่ได้จะการท่องจะเป็นดังรูปวัตถุตรงกลางตามตัวอย่างในรูปที่ 3.8 ซึ่งจะนำผลลัพธ์ที่ได้ไปกรองอีกขั้นตอน
5. นำผลลัพธ์ที่ได้ไปกรองด้วยวิธี Minimal Ray Tracer ซึ่งเป็นอัลกอริทึมตรวจสอบการตัดกันของรังสีและ boundary ซึ่งผลลัพธ์ที่ได้จะเป็นดังรูปทางขวามือสุดในรูปที่ 3.8
6. เมื่อได้ node ที่ทำการกรองด้วย Minimal Ray Tracer แล้วให้นำข้อมูลแผ่นพื้นผิวจากทุก node มาทำการ union ข้อมูลเข้าด้วยกันจากนั้นนำข้อมูลแผ่นพื้นผิวที่ได้ทั้งหมดไปใช้ในการคำนวณ Distance Interior Ratio



รูปที่ 3.9 กระบวนการของระบบแบบขนาน

3.13 ออกแบบการทำงานของระบบแบบขนาน

เพื่อลดเวลาในการคำนวณ ได้มีการออกแบบระบบของหัวข้อที่ 3.11 ในขั้นตอนที่ 2(c) โดยในขั้นตอนนี้ได้ออกแบบให้คำนวณ Distance Interior Ratio แบบขนาน โดยใช้จำนวน process ของ cpu จำนวน 8 process โดยในระบบได้ออกแบบให้สามารถแบ่งกลุ่มของเส้นตรงออกเป็นแปดกลุ่มและแต่ละกลุ่มโดยจะถูกส่งให้แต่ละ process ของ cpu ทำงานในขั้นตอนการคัดเลือกแผ่นพื้นผิวและคำนวณระยะทาง Distance Interior Ratio เท่านั้น ดังตัวอย่างในรูปที่ 3.7 คือภาพที่อธิบายขั้นตอนของการออกแบบระบบแบบขนาน โดยในงานวิจัยนี้จะเรียกการออกแบบนี้ว่า ระบบการคำนวณแบบแบ่งวัตถุและทำงานแบบขนาน (3DDIR Parallel Partition)

```

1 def run(name):
2
3     lines = data.off.getLines() # Get data of lines
4     t_off = data.off.t_obj # Get model data
5
6     part = len(lines)/8 # Partition lines into 8 groups
7     s = 0
8     e = part
9
10    # Create output path
11    out_path = "output/"+name+"_output"
12    # out_path = "output/Sphere_output"
13    if not os.path.isdir(out_path):
14        os.makedirs(out_path)
15
16    jobs = []
17    # Create process each of threads
18    for i in range(8):
19        if i < 7 :
20            jobs.append(Process(target=worker, args=(out_path,i,lines[s:e],data.box_tree.root,t_off,)))
21        else :
22            jobs.append(Process(target=worker, args=(out_path,i,lines[s:len(lines)],data.box_tree.root,t_off,)))
23
24        s += part
25        e += part
26
27    # Start jobs
28    for p in jobs :
29        p.start()
30
31    # Join jobs
32    for p in jobs :
33        p.join()
34
35    # Terminate jobs when finish the process
36    for p in jobs :
37        p.terminate()

```

รูปที่ 3.10 ตัวอย่าง source code ของ 3DDIR parallel partition

ตัวอย่างในรูปที่ 3.10 คือตัวอย่าง source code ของระบบแบบขนานซึ่งสามารถอธิบายเป็นรายละเอียดได้ดังนี้

เนื่องจากระบบที่ทำการพัฒนาใช้ภาษา python ในการออกแบบระบบทั้งหมดจึงนำ package เกี่ยวกับการทำ multiprocessing ของ python มาช่วยในการทำ parallel system โดยใช้ package ชื่อว่า multiprocessing และใช้วัตถุ Process จาก package multiprocessing ในการสร้าง process

จากตัวอย่างรูปที่ 3.10 ใน source code บรรทัดที่ 2 และ 3 คือการเก็บข้อมูลเส้นตรงและข้อมูลวัตถุสามมิติตามลำดับ จากนั้นจะทำการแบ่ง list ข้อมูลของเส้นตรงออกเป็น 8 กลุ่มโดยคำนวณขนาดตาม source code ในบรรทัดที่ 6 ซึ่งหากอยากเพิ่มหรือลดจำนวน process สามารถเปลี่ยนจำนวนกลุ่มที่จะทำการแบ่งได้

ในบรรทัดที่ 18 จะทำการสร้าง job ของแต่ละ process ขึ้นมาโดยสร้างขึ้นมาจำนวน 8 ครั้ง

```

1 def worker(out_path,i,lines,root,off):
2
3     out = open(out_path +"/output"+ str(i) +".csv","w")
4
5     for line in lines :
6         ans = process(line,root,off)
7         # p0 | p1 | amount_faces | DIR | inner distance | all distance
8         if ans != None :
9             out.write("%s,%s,%s,%s,%s,%s\n" % (line[0],line[1],ans[3],ans[0],ans[1],ans[2]))
10
11     out.close()

```

รูปที่ 3.11 ตัวอย่าง source code ของ worker function

ตาม

จำนวนกลุ่มซึ่งสามารถเพิ่มหรือลดจำนวน process ลงได้หากเปลี่ยนจำนวนครั้งในการสร้าง process แต่จำนวน process ที่ทำการสร้างจะต้องเท่ากับจำนวนกลุ่มของเส้นตรงที่ทำการแบ่งเสมอ ในบรรทัดที่ 27-37 จะเป็นคำสั่งเริ่มต้นการทำงานของ process ในแต่ละ process และ จบการทำงานของแต่ละ process ตามลำดับ

ในตัวอย่างรูปที่ 3.11 คือตัวอย่าง function worker ซึ่งแต่ละ process จะทำกระบวนการต่างๆ ของระบบด้วยการเรียก function process ในบรรทัดที่ 6 ซึ่งขั้นตอนของ function process จะเป็นขั้นตอนของการค้นหาแผ่นพื้นผิวและคำนวณ Distance Interior Ratio ซึ่งผลลัพธ์ที่ได้จาก function process จะบันทึกลงในไฟล์เพื่อนำไปแปลงเป็นเมทริกซ์อธิบายต่อไป

3.14 วิธีการทดลอง

ในงานวิจัยนี้ได้ออกแบบระบบเพื่อลดเวลาในการคำนวณเพื่อสร้างกราฟเมทริกซ์ในการอธิบายวัตถุโดยสามารถแบ่งระบบในงานวิจัยได้ดังต่อไปนี้

1. การคำนวณ Distance Interior Ratio ตามในขั้นตอนวิธีของระบบที่1 (3DDIR)
2. การคำนวณ Distance Interior Ratio ตามในขั้นตอนวิธีของระบบที่2 โดยออกแบบระบบให้ทำงานแบบเป็นลำดับ (3DDIR Single Thread Partition)
3. การคำนวณ Distance Interior Ratio ตามในขั้นตอนวิธีของระบบที่2 โดยออกแบบระบบให้ทำงานแบบขนาน (3DDIR Parallel Partition)

โดยได้ออกแบบการทดลองโดยมีวัตถุประสงค์เพื่อพัฒนาให้ระบบการทำการคำนวณสามารถคำนวณได้เร็วขึ้นเมื่อเทียบกับระบบที่1 3DDIR โดยได้ออกแบบการทดลองออกเป็นสามการทดลอง ซึ่งมีรายละเอียดดังต่อไปนี้

- ทดสอบเปรียบเทียบความเร็วในการคำนวณ 3DDIR ระหว่างระบบ 3DDIR, 3DDIR Single Thread Partition และ 3DDIR Parallel Partition โดยใช้วัตถุสามมิติจำนวน 42 วัตถุในการทดสอบโดยจำกัดให้แต่ละวัตถุมีจำนวนตำแหน่งพิกัดน้อยกว่าหรือเท่ากับ 500 ตำแหน่ง
- ทดสอบเพื่อดูแนวโน้มเวลาการคำนวณของระบบ 3DDIR Parallel Partition ในช่วงของการแบ่งวัตถุเพื่อสร้างต้นไม้, ช่วงการคัดเลือกแผ่นพื้นผิวและคำนวณ DIR และทดสอบเพื่อดูแนวโน้มของระบบทั้งหมด โดยจะใช้วัตถุสามมิติจำนวน 236 วัตถุในการทดสอบ
- ทดสอบความถูกต้องของระบบ 3DDIR Parallel Partition ว่าเมื่อทำการพัฒนาส่วนของการแบ่งวัตถุและคำนวณแบบขนาน แล้วจะให้ผลการจำแนกคงเดิม โดยจะเปรียบเทียบกับ 3DDIR ระบบเดิมที่ไม่ได้ทำการแบ่งวัตถุและยังทดสอบเพื่อเปรียบเทียบเวลาในการคำนวณของระบบ 3DDIR Parallel Partition ที่พัฒนาขึ้น โดยใช้ชุดข้อมูลวัตถุสามมิติจำนวน 166 วัตถุ

ซึ่งจากการทดลองทั้งหมดได้ทำการเตรียมสมรรถภาพเครื่องคอมพิวเตอร์, ระบบปฏิบัติการ และภาษาที่ใช้ในการพัฒนาโดยมีรายละเอียดดังต่อไปนี้

1. Intel® Xeon® CPU w3350 @ 2.8GHz 4 core 8 threads
 - (a) single thread ใช้จำนวน 1 thread
 - (b) parallel thread ใช้จำนวน 8 threads
2. ระบบปฏิบัติการ Windows 10
3. ภาษาที่ใช้ในการพัฒนา ภาษา Python

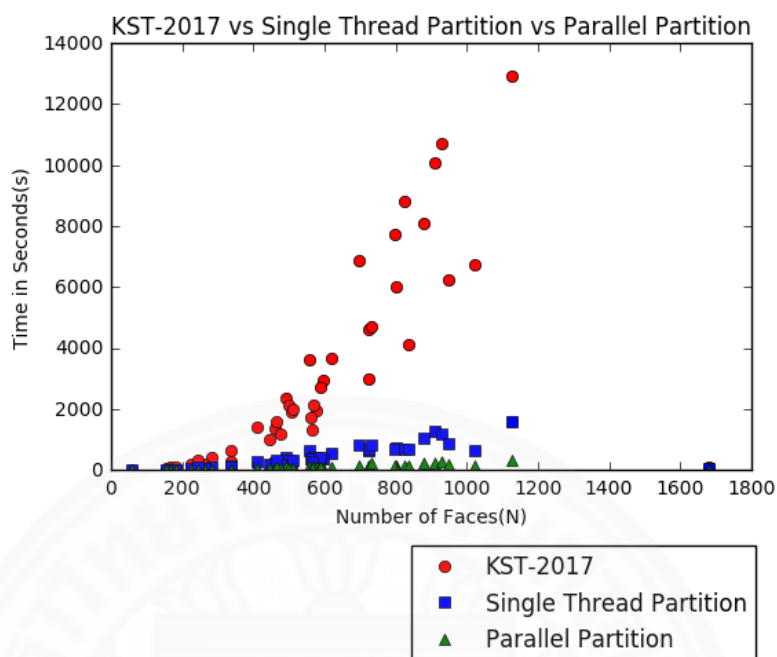
วิธีการจับเวลาในการทดลองจะใช้ package time ของภาษา python มาช่วยในการจับเวลาโดยจะใช้ function time() ของ package time ในการบันทึกเวลาเครื่องเมื่อเริ่มและสิ้นสุดกระบวนการ โดยระบบ 3DDIR single partition และ 3DDIR parallel partition จะทำการจับเวลาสองส่วนคือส่วนของการ partition และส่วนของการคำนวณ

นอกจากนี้จะมีการทำ profile เพื่อดูเวลาในการทำงานของแต่ละ function ของระบบ โดยใช้คำสั่ง 'python -m cProfile myscript.py' ในการทำ profile ของระบบ ซึ่ง profile จะแสดงเวลาที่ใช้และจำนวนครั้งที่เรียกใช้ของ function ทั้งหมดในระบบ

บทที่ 4

ผลการวิจัยและอภิปรายผล

งานวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาให้การคำนวณ Distance Interior Ratio ให้สามารถคำนวณให้รวดเร็วยิ่งขึ้น เนื่องจากว่าในขั้นตอนการคำนวณของ 3DDIR โดยไม่แบ่งวัตถุเมื่อคำนวณเพื่อสร้างกราฟสถิติแล้ววัดผลความเร็วในการคำนวณนั้นพบว่ายังใช้เวลาในการคำนวณนาน ซึ่งเป็นเพราะในการคำนวณ 3DDIR การคำนวณจุดตัดของเส้นตรงในแต่ละครั้งเราต้องเรียกใช้อัลกอริทึมการตัดกันระหว่างแผ่นพื้นผิวและเส้นตรง(7) ตรวจสอบหาจุดตัดกับแผ่นพื้นผิวทุกแผ่นในวัตถุทำให้การคำนวณใช้เวลานาน ซึ่งจะนำไปประยุกต์ใช้ในการค้นหาวัตถุสามมิติได้ยาก ดังนั้นจึงพัฒนาระบบให้ลดการเรียกใช้อัลกอริทึมการตัดกันระหว่างเส้นตรงและแผ่นพื้นผิว เพื่อลดเวลาในการคำนวณให้สามารถคำนวณได้รวดเร็วยิ่งขึ้น เพื่อให้สามารถนำระบบไปประยุกต์ใช้ในการค้นหาวัตถุสามมิติได้อย่างแท้จริง

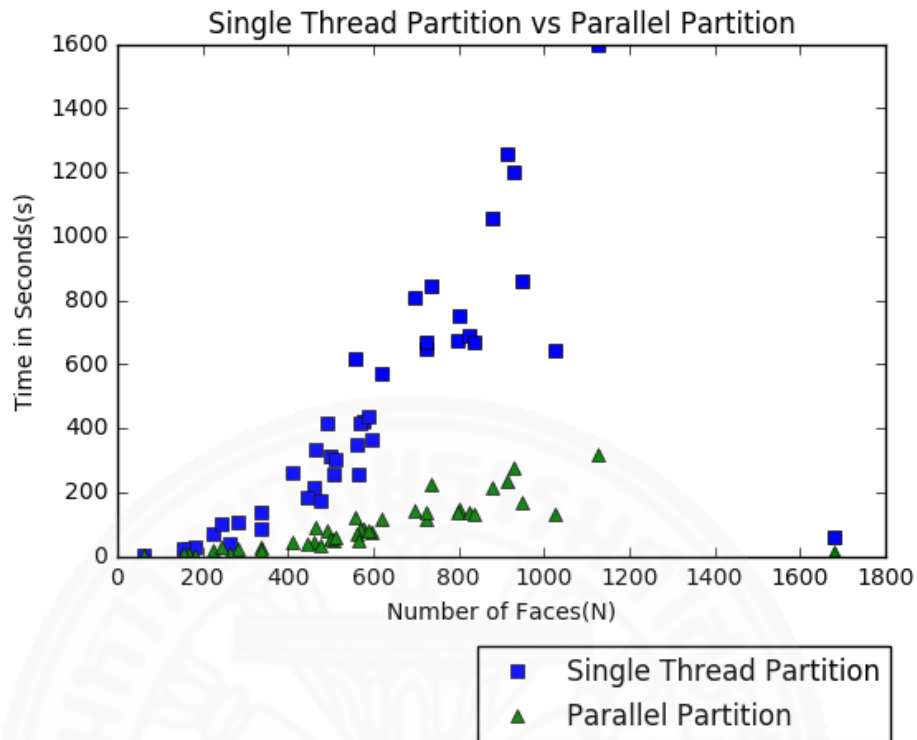


รูปที่4.1: กราฟแสดงความสัมพันธ์ระหว่างจำนวนแผ่นพื้นผิวและเวลาในการทำงานในหน่วยวินาที

4.1 เปรียบเทียบกราฟผลลัพธ์เวลา

เปรียบเทียบผลลัพธ์เวลาระหว่างระบบการคำนวณระยะทางอัตราส่วนระยะทางภายในและระยะทางของเส้นตรงโดยไม่ทำการแบ่งวัตถุ(3DDIR) และระบบการคำนวณระยะทางอัตราส่วนระยะทางภายในและระยะทางของเส้นตรงโดยทำการแบ่งวัตถุโดยในระบบที่สองจะแบ่งออกอีกสองระบบคือระบบแรกเป็นระบบที่ทำงานแบบเป็นลำดับ (3DDIR Single Thread Partition) และระบบที่สองเป็นระบบที่ทำงานแบบขนาน (3DDIR Parallel Partition) ในการเปรียบเทียบการทดลองนี้จะใช้วัตถุสามมิติของ PSB จำนวน 42 วัตถุโดยเป็นวัตถุที่มีจำนวนตำแหน่งพิกัดน้อยกว่า 500 ตำแหน่ง

โดยกราฟในรูปที่4.1 คือผลการเปรียบเทียบระหว่างสามระบบ โดยจุดแต่ละชนิดบนกราฟแทนระบบที่ใช้ในการทดลองและจุดแต่ละจุดบนกราฟจะแทนวัตถุสามมิติหนึ่งวัตถุโดยกำหนดจุดวงกลมสีแดงแทนระบบ 3DDIR, กำหนดสี่เหลี่ยมสีน้ำเงินแทนระบบ 3DDIR Single Thread Partition และสามเหลี่ยมสีเขียวแทนระบบ 3DDIR Parallel Partition จากกราฟแสดงผลการทดลองการแบ่งส่วนวัตถุและสร้างโครงสร้างต้นไม้สามารถช่วยให้การคำนวณ Distance Interior Ratio สามารถทำงานได้เร็วขึ้น เพราะการแบ่งส่วนวัตถุสามารถลดปริมาณ



รูปที่ 4.2: กราฟแสดงความสัมพันธ์ระหว่างจำนวนแผ่นพื้นผิวและเวลาในการทำงานในหน่วยวินาที

การเรียกใช้อัลกอริทึมการคำนวณการตัดกันระหว่างเส้นตรงและแผ่นพื้นผิว เพราะระบบไม่จำเป็นต้องคำนวณกับแผ่นพื้นผิวทั้งหมดโดยคำนวณเฉพาะส่วนแผ่นพื้นผิวที่คาดว่าจะเกิดจุดตัดมากที่สุด

ในกราฟรูปที่ 4.2 คือกราฟที่เปรียบเทียบการทำงานระหว่างระบบการคำนวณแบบเป็นลำดับและระบบการคำนวณแบบขนาน ซึ่งจากกราฟจะเห็นได้ว่าการคำนวณแบบขนานนั้นช่วยลดเวลาในการคำนวณลงได้เมื่อเทียบกับการคำนวณแบบเป็นลำดับ

ตารางที่ 4.1 ตารางเปรียบเทียบเวลาในการทำงานในหน่วยวินาที

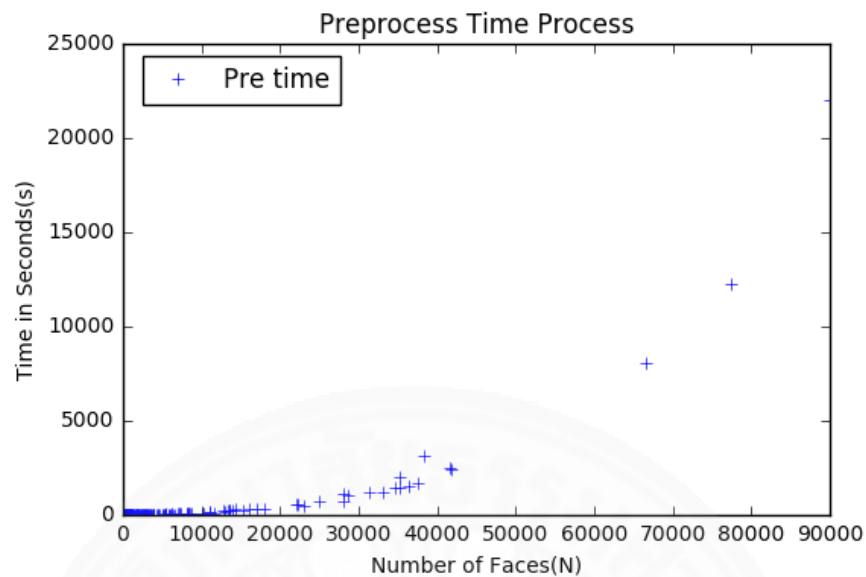
วิธี	เวลาน้อยที่สุด	เวลามากที่สุด	เวลาเฉลี่ย
3DDIR	5.672	12921.736	3284.660
3DDIR single thread partition	2.343	1595.970	450.285
3DDIR parallel partition	0.542	318.291	91.371

ตารางที่ 4.1 คือตารางแสดงผลลัพธ์เวลาของทั้งสามขั้นตอนวิธีของระบบ โดยในแถวแรกแสดงผลลัพธ์ที่ใช้เวลาน้อยที่สุดของแต่ละวิธีการทดลอง แถวที่สองแสดงผลลัพธ์ที่ใช้เวลามากที่สุดของแต่ละวิธีการทดลอง และแถวสุดท้ายแสดงผลลัพธ์ของเวลาเฉลี่ยของแต่ละวิธีการทดลอง โดยเมื่อดูจากเวลาเฉลี่ยจะเห็นว่าวิธีแบ่งส่วนของวัตถุที่ระบบทำงานแบบเป็นลำดับสามารถลดเวลาการคำนวณลงได้ประมาณ 7 เท่าเมื่อเทียบกับการคำนวณแบบไม่แบ่งวัตถุและวิธีการแบ่งส่วนวัตถุที่ระบบทำงานแบบขนานสามารถลดเวลาการคำนวณลงได้อีกถึงประมาณ 36 เท่าเมื่อเปรียบเทียบวิธีเดียวกัน

4.2 ผลลัพธ์แนวโน้มเวลาของการคำนวณ

ในงานวิจัยนี้ทำการทดสอบระบบการคำนวณโดยใช้ชุดข้อมูลวัตถุสามมิติของ PSB จำนวน 236 วัตถุโดยวัดความเร็วในการการคำนวณของระบบการคำนวณแบบขนาน (3DDIR Parallel Partition) เพื่อทดสอบและสังเกตแนวโน้มเวลาในการทำงานของแต่ละขั้นตอนของระบบ ซึ่งทางผู้วิจัยจะแบ่งการทำงานของระบบออกเป็นสองส่วนคือ ส่วนของการแบ่งวัตถุ (Pre-processing part) และ ส่วนของการคำนวณ (Computing part) ส่วนขั้นตอนในการแบ่งวัตถุเป็นส่วนที่ทำการเตรียมการทดลองก่อนที่จะทำการคำนวณ ซึ่งในขั้นตอนนี้ไม่ได้ออกแบบให้ทำงานแบบขนาน เพื่อดูแนวโน้มของขั้นตอนการคำนวณจึงทำการวัดผลเวลาในการคำนวณไว้โดยจะแสดงเป็นกราฟดังต่อไปนี้

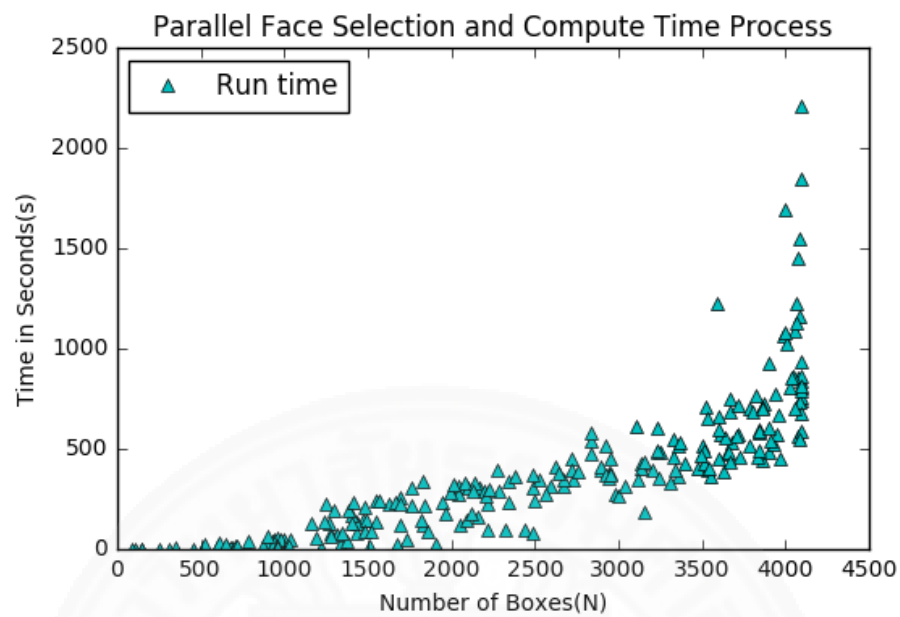
- กราฟความสัมพันธ์ระหว่างจำนวนแผ่นพื้นผิวและเวลาในขั้นตอนการแบ่งวัตถุ
- กราฟความสัมพันธ์ระหว่างจำนวนส่วนของวัตถุที่ทำการแบ่งและเวลาในขั้นตอนการคัดเลือกแผ่นพื้นผิวและคำนวณระยะทาง Distance Interior Ratio
- กราฟความสัมพันธ์ระหว่างจำนวนแผ่นพื้นผิวและเวลาในขั้นตอนการคัดเลือกแผ่นพื้นผิวและคำนวณระยะทาง Distance Interior Ratio
- กราฟความสัมพันธ์ระหว่างจำนวนแผ่นพื้นผิวและเวลาของทั้งระบบการคำนวณกราฟสถิติ



รูปที่ 4.3 กราฟความสัมพันธ์ระหว่างจำนวนแผ่นพื้นผิวและเวลาในหน่วยวินาที

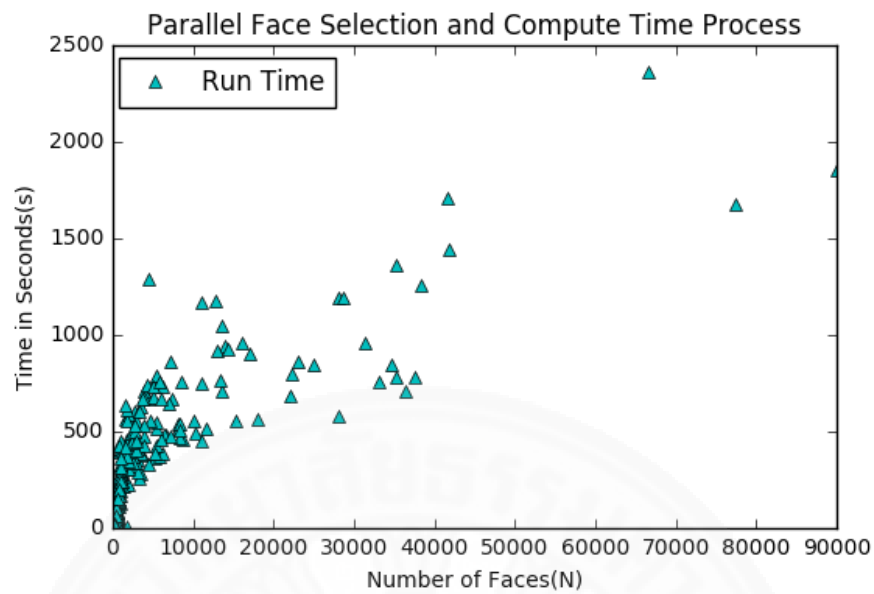
กราฟในรูปที่ 4.3 จะแสดงความสัมพันธ์ระหว่างจำนวนแผ่นพื้นผิวของวัตถุและเวลาในการคำนวณของขั้นตอนการแบ่งวัตถุ (pre-processing) ซึ่งเมื่ออภิปรายตาม pseudo code ในหัวข้อที่ 3.11 จากขั้นตอนการค้นหาจุดแบ่งวัตถุ โดยในขั้นตอนนี้ในแต่ละครั้งของการทำการค้นหาจุดแบ่งที่ดีที่สุด ต้องทำการตรวจสอบแผ่นพื้นผิวทุกแผ่นเพื่อทำการคำนวณหาจุดแบ่งที่ดีที่สุด

ดังนั้นแนวโน้มเวลาของขั้นตอนนี้จึงเป็นแบบ $O(n^2)$ เนื่องจากในขณะที่ระบบทำการค้นหาจุดแบ่งที่ดีที่สุด ในแต่ละครั้งต้องตรวจสอบแผ่นพื้นผิวใหม่เพื่อทำการคำนวณจุดแบ่งที่ดีที่สุดด้วย



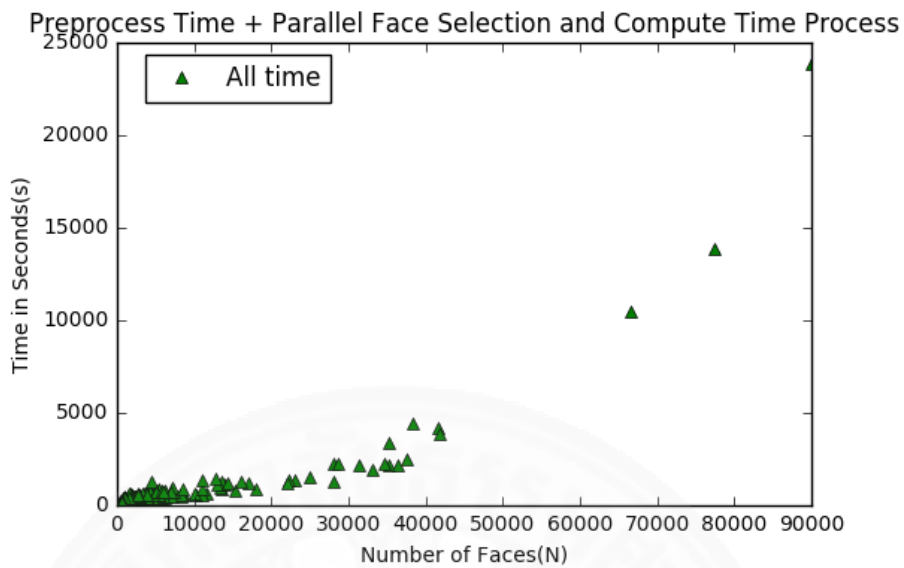
รูปที่ 4.4 กราฟความสัมพันธ์ระหว่างจำนวนแผ่นพื้นผิวและเวลาในหน่วยวินาที

กราฟในรูปที่ 4.4 ซึ่งแสดงความสัมพันธ์ของจำนวนส่วนที่ถูกแบ่งกับเวลาในการคำนวณ จะเห็นได้ว่าจำนวนส่วนที่ถูกแบ่งนั้นเมื่อมีมากขึ้นเรื่อยจะทำให้ใช้เวลาในการคำนวณมากขึ้น เนื่องจากการค้นหาทำงานโดยใช้การค้นหาแบบแนวกว้าง ดังนั้นเมื่อจำนวนส่วนที่ถูกแบ่งมีมากยิ่งขึ้นจะทำให้ใช้เวลาในการคำนวณมากยิ่งขึ้น



รูปที่ 4.5 กราฟความสัมพันธ์ระหว่างจำนวนแผ่นพื้นผิวและเวลาในหน่วยวินาที

กราฟในรูปที่ 4.5 จะแสดงความสัมพันธ์ระหว่างเวลาและแผ่นพื้นผิว โดยเมื่อจำนวนแผ่นพื้นผิวมีมากขึ้นก็จะทำให้การคำนวณใช้เวลาเพิ่มขึ้น แต่ตามแนวโน้มไม่ได้เพิ่มขึ้นสูงมากเมื่อเทียบกับวัตถุที่มีแผ่นพื้นผิวน้อยกว่า



รูปที่ 4.6 กราฟความสัมพันธ์ระหว่างจำนวนแผ่นพื้นผิวและเวลาในหน่วยวินาที

ในกราฟรูปที่ 4.6 นั้นจะแสดงความสัมพันธ์ของเวลาตลอดช่วงการคำนวณและจำนวนแผ่นพื้นผิวซึ่งจากขนาดของช่วงเวลาของกราฟนั้นจะสูงกว่ากราฟการคำนวณ เนื่องจากว่าในขั้นตอนการแบ่งวัตถุนั้นมีแนวโน้มการทำงานคล้ายกราฟยกกำลังสอง

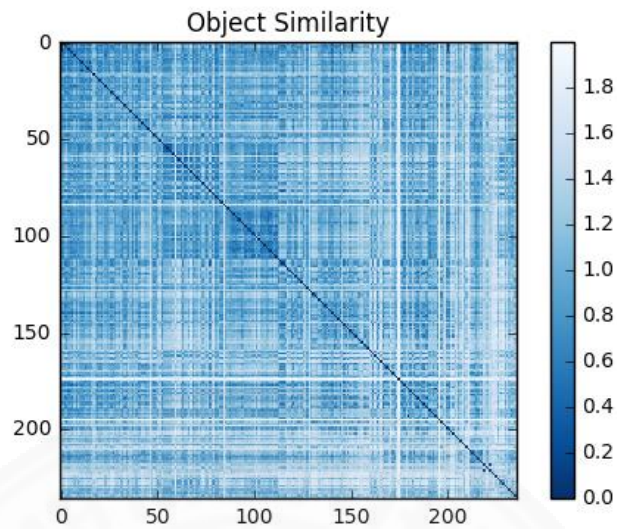
ดังนั้นจึงเป็นข้อจำกัดของระบบในงานวิจัยนี้คือระบบในการเตรียมวัตถุเพื่อที่จะใช้ในการค้นหาซึ่งใช้เวลามากพอสมควร โดยจากกราฟจะเห็นว่าเมื่อจำนวนแผ่นพื้นผิวมีมากขึ้นจะใช้เวลาในการคำนวณมากขึ้นในลักษณะคล้ายกราฟเลขยกกำลังสอง

ตารางที่ 4.2: ตารางเวลาการทำงานของระบบในหน่วยวินาที

ขั้นตอน	แบ่งวัตถุ	Search and Compute	All process
เวลาน้อยที่สุด	0.016	0.504	0.542
เวลามากที่สุด	21994.917	2362.236	23852.188
เวลาเฉลี่ย	301.997	406.358	708.355

ในตารางที่ 4.2 ได้แสดงเวลาในแต่ละขั้นตอนการคำนวณโดยแบ่งออกเป็นส่วนของ การแบ่งวัตถุ, ส่วนของการคำนวณ และเวลารวมของขั้นตอนทั้งหมด โดยจากตารางเวลาน้อย ที่สุดและเวลาเฉลี่ยจะเห็นได้ว่าขั้นตอนของส่วนการแบ่งวัตถุนั้นใช้เวลาในการแบ่งน้อยกว่าช่วง ขั้นตอนในการคำนวณ แต่เมื่อดูเวลาที่มากที่สุดจะเห็นว่าช่วงของการแบ่งวัตถุนั้นมากกว่าช่วง เวลาในการคำนวณ สาเหตุเป็นเพราะขั้นตอนในการแบ่งวัตถุมีแนวโน้มเวลาเป็น $O(n^2)$ ซึ่ง เมื่อแผ่นพื้นผิวมีจำนวนมากขึ้นการเตรียมการสร้างต้นไม้เพื่อทำการคัดเลือกแผ่นพื้นผิวก็จะ ใช้เวลานานขึ้นในลักษณะของกราฟกำลังสอง แต่แนวโน้มเวลาของการคำนวณนั้นน้อยกว่า กราฟในขั้นตอนการแบ่งวัตถุ

จึงสรุปได้ว่าถ้าหากวัตถุมีแผ่นพื้นผิวจำนวนน้อยการแบ่งวัตถุจะทำได้เร็วกว่าในขั้น ตอนการคำนวณ แต่เมื่อถึงจุดๆหนึ่งเมื่อวัตถุมีแผ่นพื้นผิวเป็นจำนวนมากก็จะทำให้ใช้เวลามาก กว่าขั้นตอนในการคำนวณ ซึ่งจากผลลัพธ์ของการทดลองนี้เราจึงพบข้อจำกัดในส่วนของการ แบ่งวัตถุซึ่งถ้าหากว่าเราสามารถลดเวลาในการคำนวณวัตถุได้เราก็จะสามารถลดเวลาในการ คำนวณลงได้



รูปที่ 4.7 กราฟความสัมพันธ์ระหว่างจำนวนแผ่นพื้นผิวและเวลาในหน่วยวินาที

นอกจากนี้ได้ทำการทดสอบโดยนำวัตถุสามมิติในชุดการทดลองนี้ที่สร้างกราฟแผนภูมิสถิติจากนั้นนำข้อมูลของกราฟที่อธิบายวัตถุทั้งหมดสร้างเป็นกราฟ Confuse Matrix ตามในรูปที่ 4.6 เพื่อแสดงความสัมพันธ์ของความคล้ายคลึงของวัตถุโดยจากกราฟในรูปที่ 4.6 สีที่เข้มคือวัตถุสองวัตถุเมื่อทำการเปรียบเทียบกันแล้ว พบว่ามีคล้ายคลึงกันมากและสีที่อ่อนลงมาคือวัตถุที่เมื่อเปรียบเทียบแล้วพบว่ามีความคล้ายคลึงในอันดับถัดมา โดยสีที่อ่อนจนเกือบขาวคือวัตถุที่ไม่มีความคล้ายคลึงกันหรือมีความคล้ายคลึงกันน้อย

ตารางที่ 4.3 ตารางผลลัพธ์การเปรียบเทียบการจำแนกวัตถุ

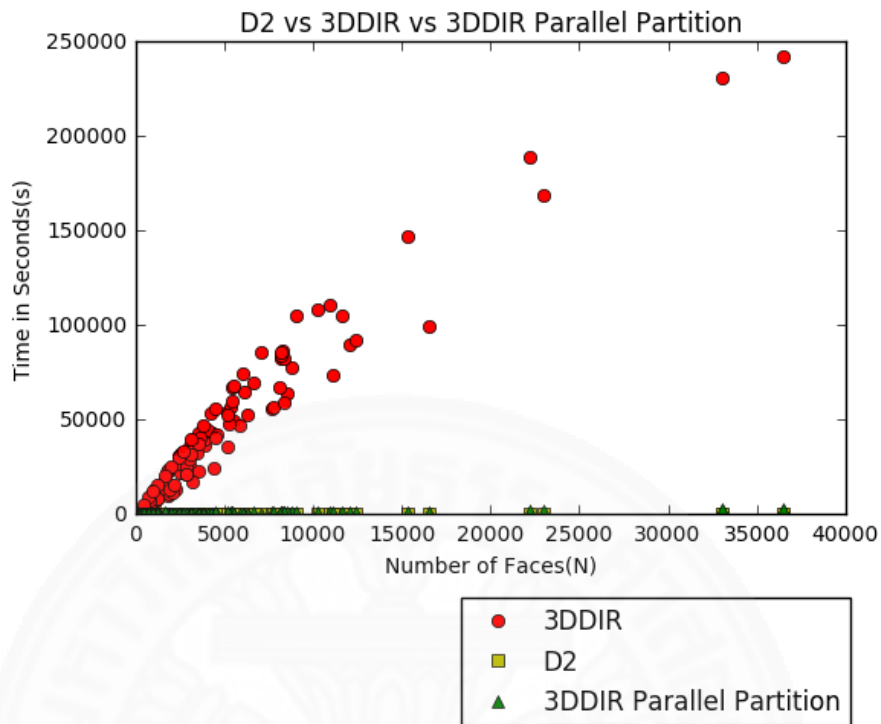
วิธีอธิบาย	First Tier	Second Tier	Nearest Neighbor	Average Time(s)
D2	18.05	26.42	33.73	3.28
3DDIR	22.96	34.60	42.16	32168.19
3DDIR Parallel Partition	19.93	30.24	41.57	380.52

4.3 ตรวจสอบผลลัพธ์ของการจำแนกวัตถุเมื่อเพิ่มขั้นตอนการแบ่งวัตถุและระบบการคำนวณแบบขนาน

ทำการทดสอบผลลัพธ์ประสิทธิภาพการจำแนกวัตถุเพื่อตรวจสอบว่า 3DDIR Parallel Partition นั้นสามารถจำแนกวัตถุได้ใกล้เคียงกับ 3DDIR เพื่อทดสอบว่าเมื่อเพิ่มส่วนของโครงสร้างข้อมูลการแบ่งวัตถุแล้วส่วนนี้จะไม่ทำให้ระบบการคำนวณ Distance Interior Ratio เกิดปัญหา โดยทำการทดสอบโดยนำวัตถุสามมิติที่ทำการทดลองใน 3DDIR จำนวน 166 วัตถุในการทดสอบและใช้วิธีวัดผลประสิทธิภาพการจำแนกวัตถุโดยวิธี First Tier, Second Tier และ Nearest Neighbor โดยอ้างอิงวิธีการวัดผลประสิทธิภาพจากงานวิจัย 'Shape Distribution'(D2) (6)

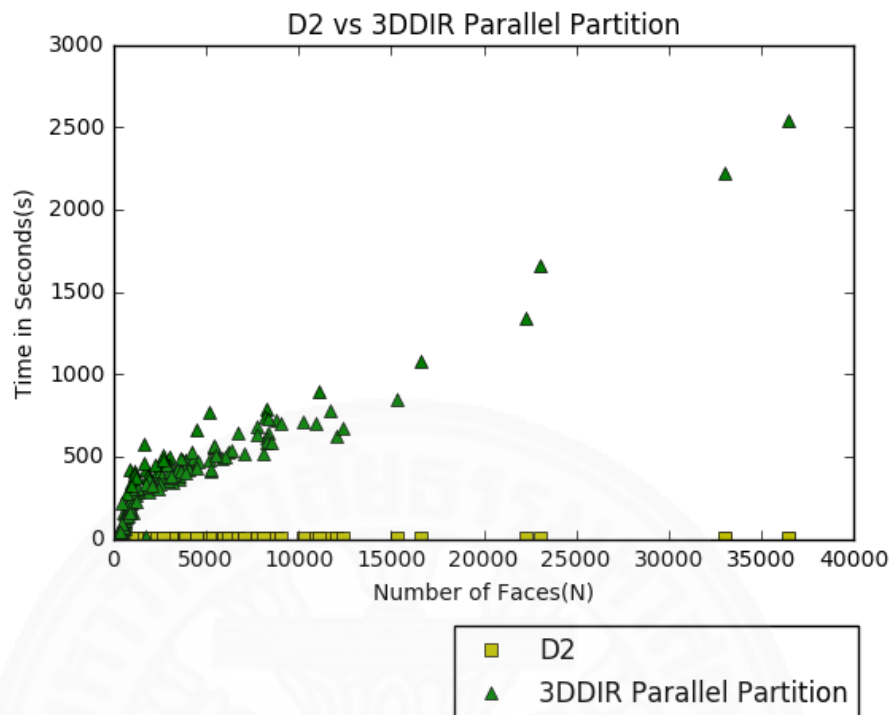
โดยผลลัพธ์ที่ได้จะแสดงในตารางที่ 4.3 โดยที่ค่า First Tier ของ 3DDIR Parallel Partition นั้นมีค่าอยู่ที่ร้อยละ 19.93, Second Tier อยู่ที่ร้อยละ 30.24 และ Nearest Neighbor อยู่ที่ร้อยละ 41.57 และเวลาเฉลี่ยในการคำนวณของ 3DDIR Parallel Partition นั้นอยู่ที่ 380.52 วินาที

ซึ่งจากผลลัพธ์ดังกล่าวในตารางนั้นจะพบว่าเมื่อเพิ่มส่วนของกระบวนการแบ่งวัตถุกับระบบ 3DDIR ยังทำงานได้เหมือนในงานทดลองเดิม โดยสาเหตุที่ค่าร้อยละดังกล่าวไม่ตรงกันนั้นเนื่องมาจากว่าในขั้นตอนการคำนวณ Distance Interior Ratio นั้นมีการสุ่มตัวอย่างตำแหน่งพิกัด ดังนั้นจึงทำให้ผลลัพธ์มีความคลาดเคลื่อนจากผลลัพธ์เดิมที่เคยวัดได้



รูปที่ 4.8 กราฟความสัมพันธ์ระหว่างจำนวนแผ่นพื้นผิวและเวลาในหน่วยวินาที

นอกจากนี้ยังทดสอบวัดผลความเร็วในการคำนวณโดยเปรียบเทียบระหว่างสามวิธี ได้แก่ D2, 3DDIR, 3DDIR Parallel Partition เพื่อดูแนวโน้มเวลาในการคำนวณ โดยแสดงในรูปที่ 4.7 คือกราฟที่แสดงผลเวลาในการคำนวณของวิธีการทั้งสามวิธีซึ่งจะเห็นว่า 3DDIR นั้นใช้เวลาในการคำนวณมากที่สุด



รูปที่ 4.9 กราฟความสัมพันธ์ระหว่างจำนวนแผ่นพื้นผิวและเวลาในหน่วยวินาที

ในกราฟที่ 4.8 แสดงถึงความเร็วในการคำนวณของ 3DDIR Parallel Partition และ D2 ซึ่งเป็นกราฟที่ขยายขนาดเพื่อแสดงแนวโน้มต่อจากกราฟในรูปที่ 4.7 ซึ่งเวลาในการคำนวณของ 3DDIR Parallel Partition นั้นใช้เวลาในการคำนวณนานกว่า D2 โดยได้สรุปเวลาในการคำนวณทั้งหมดเป็นไปดังตารางต่อไปนี้

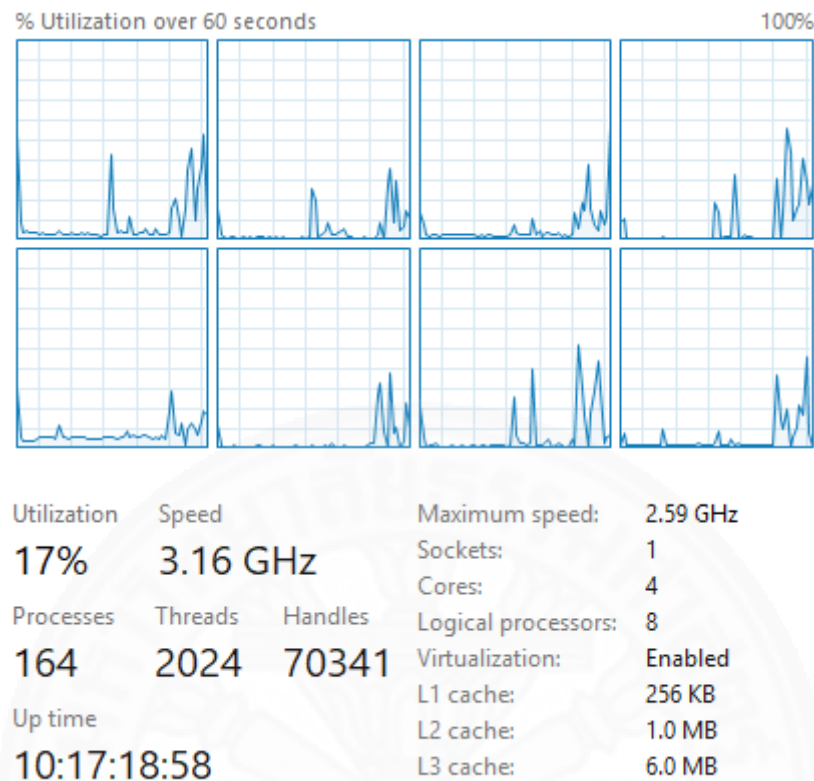
ตารางที่ 4.4 ตารางเปรียบเทียบเวลาในการทำงานในหน่วยวินาที

วิธี	D2	3DDIR	3DDIR Parallel Partition
เวลาน้อยที่สุด	0.031	5.672	1.641
เวลามากที่สุด	5.774	242456.38	2541.204
เวลาเฉลี่ย	3.279	32168.193	380.518

โดยจากตารางที่ 4.4 จะแสดงให้เห็นว่าการคำนวณแบบ 3DDIR นั้นใช้เวลามากที่สุด โดยเมื่อจากค่าเฉลี่ยที่ 32168.193 วินาทีหรือประมาณ 8.93 ชั่วโมงซึ่งหมายถึง 3DDIR นั้นใช้เวลาเฉลี่ยในการคำนวณวัตถุทดสอบทั้งหมด ถึง 8.93 ชั่วโมงและ 3DDIR Parallel Partition นั้นใช้เวลาในการคำนวณรองลงมาจาก 3DDIR ที่ 380.518 วินาทีซึ่งหมายความว่า 3DDIR

Parallel Partition ใช้เวลาเฉลี่ยในการคำนวณวัตถุทดสอบทั้งหมดที่ประมาณ 6.34 นาทีซึ่งจะเห็นว่า 3DDIR Parallel Partition ทำงานได้เร็วกว่า 3DDIR โดยระบบสามารถคำนวณได้เร็วกว่าถึงประมาณ 85 เท่า แต่อย่างไรก็ตามวิธีคำนวณเพื่อจำแนกวัตถุด้วย D2 นั้นทำงานได้เร็วที่สุดโดยใช้เวลาเฉลี่ยเพียง 3.279 วินาทีสาเหตุนั้นเนื่องมาจากว่าวิธีการคำนวณ D2 คำนวณเพียงระยะทางระหว่างตำแหน่งพิกัดภายในวัตถุเท่านั้นทำให้วิธี D2 คำนวณได้รวดเร็วที่สุด



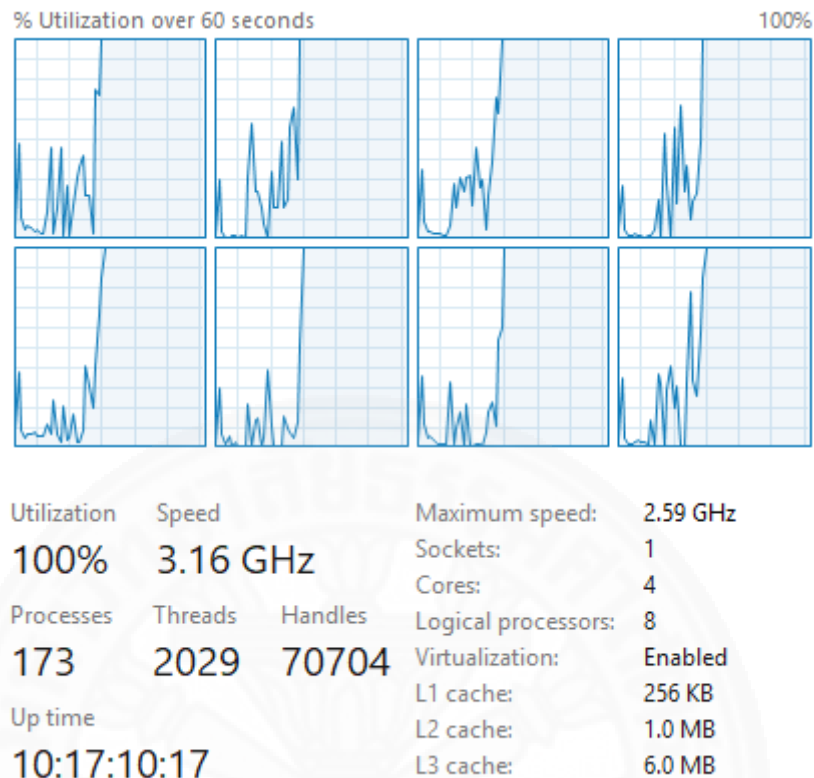


รูปที่ 4.10 แสดงผลลัพธ์การทำงานของระบบ 3DDIR Single Partition

4.4 อภิปรายการทำงานของระบบแบบขนาน

เพื่อทดสอบว่าระบบแบบขนานของ 3DDIR parallel partition นั้นทำงานแบบขนานจริงได้ทำการ Utilization เพื่อแสดงให้เห็นว่าระบบทำงานแบบขนานจริง โดยจะทำการ Utilization โดยดูจาก task manager ของระบบ OS ของ window

เนื่องจากข้อจำกัดของ OS window 10 เมื่อสั่งระบบแบบ 3DDIR single partition ให้ทำการคำนวณจะพบว่าเมื่อทำการ Utilization cpu จะมีการทำงานทุก processors การคำนวณดังตัวอย่างที่แสดงในรูปที่ 4.9 จึงทำให้พบว่าแม้ว่าระบบจะทำงานแบบเป็นลำดับแต่ OS จะช่วยให้ระบบสามารถส่งข้อมูลให้ processors อื่นทำงานได้ แต่อย่างไรก็ตาม cpu ของระบบที่ทำงานแบบเป็นลำดับจะไม่สามารถทำงานได้เต็มที่ดังในรูปที่ 4.9 จะเห็นว่า cpu ทำงานได้เพียงร้อยละ 17 เท่านั้น



รูปที่ 4.11 แสดงผลลัพธ์การทำงานของระบบ 3DDIR Parallel Partition

จากรูปที่ 4.10 เป็นการ Utilization ของระบบการคำนวณแบบขนานซึ่งจากภาพจะเห็นได้ว่า cpu มีการทำงานเต็มถึงร้อยละ 100 หมายความว่าระบบแบบขนานสามารถทำให้ cpu ทำงานได้เต็มที่ทุก processors เมื่อเทียบกับรูปการ Utilization ในรูปที่ 4.9 ที่ทำงานเพียงร้อยละ 17 จะพบว่าระบบ 3DDIR Parallel Partition สามารถทำงานแบบขนานได้จริง

นอกจากนี้ยังทำการทดสอบโดยเปลี่ยนจำนวน processors ในการคำนวณเพื่อดูแนวโน้มเวลาในการคำนวณซึ่งจะใช้จำนวน processors เป็นจำนวน 2,4,6,8 ตามลำดับโดยใช้วัตถุจำนวน 1 วัตถุในการทดสอบ ซึ่งสามารถแสดงเวลาการทำงานได้ดังตารางต่อไปนี้

ตารางที่ 4.5 ผลลัพธ์การทำงานของระบบโดยเปลี่ยนจำนวน processors

ขั้นตอน	2 processors	4 processors	6 processors	8 processors
ค้นหาและคำนวณ	115.438	73.803	64.001	66.654

จากตารางที่ 4.5 จะพบว่าเมื่อเปลี่ยนจำนวน processors จะเห็นได้ว่าเวลาจะลดลงตามจำนวน processors ที่เพิ่มขึ้น

จึงสามารถสรุปได้ว่าเมื่อเพิ่มจำนวน processors ในระบบจะสามารถทำให้ cpu ทำงานได้เต็มทุก processors ซึ่งแสดงให้เห็นว่าระบบสามารถทำงานแบบขนานได้จริงและเมื่อเพิ่มจำนวน processors ในการคำนวณจะสามารถลดเวลาในการคำนวณลงได้ แต่อย่างไรก็ตามเมื่อเพิ่ม processors มากขึ้นเวลาที่ใช้ในการคำนวณจะไม่ลดลงมากเมื่อเทียบกับจำนวน processors ก่อนหน้า

4.4 อภิปรายการขั้นตอนของการค้นหา

เนื่องจากในขั้นตอนในการค้นหาวัตถุนั้นใช้วิธีค้นหาด้วยวิธี AABB และ Minimal Ray Tracer ซึ่งเกิดข้อสงสัยว่าหากวิธี Minimal Ray Tracer เพียงอย่างเดียวจะลดเวลาในการคำนวณลงหรือไม่เนื่องจาก Minimal Ray Tracer นั้นทำการ filter ได้ดีกว่าวิธี AABB

ซึ่งได้ทำการทดสอบโดยนำวัตถุมาทดสอบจำนวน 6 วัตถุได้ผลลัพธ์ที่สามารถสรุปได้ดังตารางต่อไปนี้

ตารางที่ 4.6 ตารางเปรียบเทียบวิธีในการค้นหา

ID	AABB and Minimal Ray Tracer	Minimal Ray Tracer
m0	60.303	68.402
m1	37.533	43.194
m2	485.398	519.659
m3	209.208	234.773
m7	37.564	48.343
m8	139.722	145.974

จากข้อมูลในตารางที่ 4.6 จะเห็นได้ว่าเมื่อนำ AABB มาใช้ในการค้นหาจะใช้เวลาน้อยกว่าการค้นหาที่ไม่ใช้วิธี AABB เนื่องจากกว่า AABB มีขั้นตอนการตัดสินใจที่ง่ายไม่ต้องคำนวณเครื่องหมายคณิตศาสตร์ จึงทำให้เมื่อมีวิธี AABB ช่วยในการ filter ส่วนของวัตถุจะทำให้ลดเวลาในการคำนวณเมื่อเทียบกับการค้นหาที่ไม่มี AABB

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

งานวิจัยนี้มีจุดประสงค์เพื่อพัฒนาระบบการจำแนกวัตถุสามมิติโดยให้สามารถจำแนกวัตถุสามมิติจากฐานข้อมูลได้อย่างรวดเร็ว จากการศึกษาค้นคว้าทางผู้วิจัยได้แบ่งกลุ่มความรู้ที่ได้ศึกษาออกมาได้สามประเภทดังนี้

วิธีในการจำแนกวัตถุเป็นการพัฒนาให้ระบบคอมพิวเตอร์สามารถจำแนกวัตถุได้โดยวิธีการจำแนกจะนำวัตถุทดสอบไปเปรียบเทียบกับวัตถุในฐานข้อมูลเพื่อจำแนกว่าวัตถุทดสอบมีความคล้ายคลึงกับวัตถุใดในฐานข้อมูล ซึ่งจากการศึกษาระบบทั่วไปจะนำวัตถุไปแปลงข้อมูลเป็นตัวแทนอธิบายวัตถุเช่น กราฟ, โครงสร้าง และใช้ตัวแทนอธิบายวัตถุในการเปรียบเทียบวัตถุสองชนิด ซึ่งสามารถจำแนกได้เร็วกว่าวิธีการเปรียบเทียบวัตถุโดยการเทียบตำแหน่งซึ่งใช้เวลาในการจำแนกที่นาน ซึ่งจากการศึกษาทางผู้วิจัยสนใจวิธีการแปลงข้อมูลสามมิติให้เป็นกราฟเมตริกซ์ในการอธิบายวัตถุ

การแบ่งวัตถุเป็นการจำลองการแบ่งวัตถุออกเป็นส่วนๆ และสร้างโครงสร้างในรูปแบบต้นไม้ข้อมูล เพื่อใช้ในการเก็บข้อมูลแต่ละส่วนที่แบ่งของวัตถุซึ่งมีประโยชน์อย่างมากในการช่วยลดเวลาการคำนวณของระบบยกตัวอย่างเช่น การเร็นเดอร์วัตถุสามมิติ, การค้นหาแบบระบุส่วนของวัตถุ

การออกแบบระบบการคำนวณแบบขนาน เป็นการออกแบบให้ระบบสามารถคำนวณแบบขนานได้โดยระบบการคำนวณทุกๆ ไปนั้นเป็นการคำนวณแบบเป็นลำดับซึ่งจะคำนวณด้วยหน่วยประมวลผลเดี่ยว ซึ่งระบบแบบขนานจะแบ่งข้อมูลออกเป็นกลุ่มจากนั้นจะนำกลุ่มข้อมูลที่แบ่งกันนั้นส่งไปให้หน่วยประมวลผลหลายๆ หน่วย จากนั้นด้วยการออกคำสั่งจากระบบเพียงครั้งเดียวระบบแบบขนานจะสามารถประมวลผลข้อมูลที่ได้พร้อมๆ กันทุกหน่วยประมวลผล ซึ่งเราเรียกการทำงานของระบบในลักษณะนี้ว่า Single Instruction Multiple Data (SIMD)

5.2 อภิปรายผลลัพธ์ความเร็วในการคำนวณ

จากการทดลองเพื่อวัดผลเวลาในการคำนวณของระบบ 3DDIR, 3DDIR Single Thread Partition และ 3DDIR Parallel Partition โดยใช้วัตถุในการทดลองจำนวน 42 วัตถุโดยข้อมูลมีตำแหน่งพิกัดน้อยกว่า 500 ตำแหน่ง สามารถสรุปเป็นตารางดังต่อไปนี้

ตารางที่ 5.1 ตารางเปรียบเทียบเวลาในการทำงานของแต่ละระบบในหน่วยวินาที

วิธี	3DDIR	Single Thread Partition	Parallel Partition
เวลาน้อยที่สุด	5.672	2.343	0.542
เวลามากที่สุด	12921.736	1595.97	318.291
เวลาเฉลี่ย	3284.66	450.285	91.371

จากผลลัพธ์ที่ได้แสดงจากตารางที่ 5.1 จะพบว่าการคำนวณของระบบ 3DDIR จะมีเวลาเฉลี่ยในการคำนวณอยู่ที่ 3284.66 วินาที 3DDIR Single Thread Partition มีเวลาเฉลี่ยในการคำนวณอยู่ที่ 450.285 วินาที และ 3DDIR Parallel Partition มีเวลาเฉลี่ยในการคำนวณอยู่ที่ 91.37 วินาที ซึ่งจากผลการทดลองสรุปได้ว่า 3DDIR Single Thread Partition สามารถคำนวณได้เร็วกว่า 3DDIR ที่ประมาณ 7 เท่า และ 3DDIR Parallel Partition สามารถคำนวณได้เร็วกว่า 3DDIR ถึง 36 เท่า

จากการทดลองเพื่อดูแนวโน้มการคำนวณของระบบแบบขนาน (3DDIR Parallel Partition) โดยใช้วัตถุทดลองจำนวน 236 วัตถุสามารถสรุปได้เป็นตารางดังนี้

ตารางที่ 5.2 ตารางเวลาการทำงานของแต่ละขั้นตอนในหน่วยวินาที

ขั้นตอน	แบ่งวัตถุ	Search and Compute	All process
เวลาน้อยที่สุด	0.016	0.504	0.542
เวลามากที่สุด	21994.92	2362.24	23852.188
เวลาเฉลี่ย	301.997	406.358	708.355

จากผลลัพธ์ที่ได้แสดงจากตารางที่ 5.2 เวลาเฉลี่ยของการแบ่งวัตถุจะอยู่ที่ 301.997 วินาที และเวลาเฉลี่ยในการคัดเลือกแผ่นพื้นผิวและคำนวณอยู่ที่ 406.358 วินาที แต่เมื่อดูจาก

เวลาที่มากที่สุดจะเห็นว่าเวลาในการแบ่งวัตถุที่มากที่สุดจะอยู่ที่ 21994.92 วินาทีและเวลามากที่สุดในการคัดเลือกแผ่นพื้นผิวและคำนวณอยู่ที่ 2362.24 วินาทีซึ่งเมื่อเทียบกับระหว่างค่าเฉลี่ยของทั้งสองขั้นตอนจะพบว่าขั้นตอนการแบ่งเร็วกว่าขั้นตอนการคำนวณที่ประมาณ 1.34 เท่า แต่เมื่อเทียบกับระหว่างค่ามากที่สุดจะพบว่าขั้นตอนการคำนวณจะเร็วกว่าขั้นตอนการแบ่งที่ประมาณ 9.31 เท่า สาเหตุเนื่องมาจากว่าในขั้นตอนการแบ่งวัตถุไม่ได้ออกแบบระบบให้คำนวณแบบขนาน ดังนั้นเมื่อจำนวนข้อมูลมีมากขึ้นจึงทำให้เวลาที่ใช้ในการแบ่งวัตถุมากขึ้นซึ่งมากกว่าเวลาในขั้นตอนการคำนวณ

จากการทดลองเพื่อตรวจสอบการจำแนกของ 3DDIR Parallel Partition นั้นสามารถสรุปเป็นตารางดังนี้

ตารางที่ 5.3 ตารางผลลัพธ์การเปรียบเทียบการจำแนกวัตถุ

วิธีการอธิบาย	First Tier	Second Tier	Nearest Neighbor	Average Time(s)
D2	18.05	26.42	33.73	3.28
3DDIR	22.96	34.60	42.16	32168.19
3DDIR Parallel Partition	19.93	30.24	41.57	380.52

จากผลลัพธ์เวลาในตารางที่ 5.3 จะพบว่าเวลาในการคำนวณของ 3DDIR จะอยู่ที่ 32168.19 วินาทีเวลาในการคำนวณของ 3DDIR Parallel Partition จะอยู่ที่ 380.52 วินาทีจะพบว่า 3DDIR Parallel Partition เร็วกว่า 3DDIR ถึง 84.54 เท่า ซึ่งสาเหตุที่การทดลองนี้ความเร็วที่เปรียบเทียบที่ได้เพิ่มขึ้นเนื่องมาจากการทดลองนี้ได้เพิ่มวัตถุที่มีความละเอียดเพิ่มขึ้นเมื่อเทียบกับชุดการทดลองแรก จึงทำให้ความเร็วที่เปรียบเทียบกันของการทดลองนี้จึงเพิ่มขึ้น แต่อย่างไรก็ตามเมื่อเปรียบเทียบระหว่าง D2 และ 3DDIR Parallel Partition จะพบว่า D2 คำนวณได้เร็วกว่าถึง 116 เท่าสาเหตุเนื่องมาจากว่า D2 คำนวณเฉพาะระยะทางของเส้นตรงที่ลากระหว่างตำแหน่งพิกัดของวัตถุจึงทำให้ D2 คำนวณได้เร็วที่สุด

5.3 อภิปรายแนวโน้มของการคำนวณของระบบแบบขนาน

เพื่ออภิปรายการทำงานของระบบ 3DDIR Parallel Partition ที่พัฒนาขึ้น โดยทดสอบ เวลาในการคำนวณเพื่อดูแนวโน้มของระบบ โดยใช้ชุดข้อมูลสามมิติจำนวน 236 วัตถุจาก ผลการทดลองเมื่อพิจารณาจากกราฟในรูปที่ 4.4 แนวโน้มของเฉพาะขั้นตอนการคำนวณจะเป็น $O(\log n)$ แต่จากผลการทดลองเมื่อพิจารณาจากกราฟในรูปที่ 4.5 จะพบว่าแนวโน้มของระบบโดยรวมทั้งหมดเป็น $O(n^2)$ และจากตารางที่ 5.2 เมื่อเทียบระหว่างเวลาเฉลี่ย และเวลาที่มากที่สุดจะพบว่าในเวลาเฉลี่ยขั้นตอนการแบ่งวัตถุจะเร็วกว่าที่ 1.34 เท่า แต่เมื่อเทียบในเวลาที่มากที่สุดจะพบว่าในขั้นตอนการคำนวณจะเร็วกว่าที่ 9.31 เท่า สาเหตุเนื่องมาจากว่าในขั้นตอนการแบ่งวัตถุหากวัตถุมีจำนวนข้อมูลที่มากขึ้นเมื่อถึงจุดหนึ่งจะทำให้ระบบ ใช้เวลาในการคำนวณมากขึ้น ซึ่งยังเป็นปัญหาของระบบ โดยถ้าหากว่าสามารถลดเวลาในการ แบ่งวัตถุลงได้จะสามารถลดเวลาของทั้งระบบลงได้

5.4 อภิปรายการจำแนกของระบบการคำนวณแบบขนาน

จากการทดลองเพื่อทดสอบการจำแนกของระบบแบบขนานเพื่อตรวจสอบว่าเมื่อเพิ่ม ส่วนของการแบ่งวัตถุแล้ว จะไม่ทำให้ประสิทธิภาพในการจำแนกเปลี่ยนไปจากระบบที่ไม่ได้ ทำการแบ่งวัตถุ(3DDIR) ซึ่งจากตารางที่ 5.3 จะเห็นได้ว่าประสิทธิภาพการจำแนกของระบบ 3DDIR Parallel Partition นั้นไม่ต่างจากระบบ 3DDIR เดิมมากนัก ซึ่งสาเหตุที่ผลลัพธ์จากการ วัดประสิทธิภาพการจำแนกไม่ตรงกันกับ 3DDIR โดยคลาดเคลื่อนเล็กน้อยสาเหตุเนื่องมาจาก ว่าระบบมีขั้นตอนในการสุ่มตำแหน่งวัตถุขึ้นหากวัตถุมีตำแหน่งพิกัดมากกว่า 500 ตำแหน่ง ดังนั้นจึงทำให้ผลลัพธ์จากการวัดประสิทธิภาพการจำแนกของ 3DDIR Parallel Partition คลาดเคลื่อนกับ 3DDIR เล็กน้อย และอย่างไรก็ตามประสิทธิภาพการจำแนกวัตถุก็ยังให้ผลความ แม่นยำที่ใช้จำแนกวัตถุได้ไม่ชัดเจน ซึ่งทางผู้วิจัยยังไม่ได้ออกแบบเพื่อเพิ่มประสิทธิภาพในการ จำแนก ซึ่งในอนาคตนั้นระบบยังสามารถพัฒนาให้สามารถจำแนกให้แม่นยำขึ้นได้

5.5 ข้อเสนอแนะสำหรับงานวิจัยครั้งต่อไป

1. เพื่อพัฒนาให้ระบบสามารถคำนวณได้อย่างรวดเร็วมากขึ้น เราสามารถปรับปรุงขั้นตอนในการแบ่งวัตถุให้สามารถสร้างโครงสร้างข้อมูลได้รวดเร็วยิ่งขึ้น ซึ่งจะทำให้เราสามารถลดเวลาในการคำนวณเพื่อสร้างกราฟเมทริกซ์ได้

2. เพื่อพัฒนาระบบการจำแนกวัตถุสามมิติให้มีความแม่นยำมากขึ้น เราสามารถปรับปรุงวิธีการที่เราใช้จำแนกวัตถุเช่น อาจจะมีการแบ่งกลุ่มข้อมูลของวัตถุทดสอบ เพื่อเพิ่มประสิทธิภาพของ 3DDIR ให้สามารถจำแนกได้อย่างแม่นยำมากขึ้น



รายการอ้างอิง

1. Belongie S, Malik J, Puzicha J. Shape matching and object recognition using shape contexts. *IEEE TPAMI*. 2002; 24(4):509-522.
2. Ip CY, Lapadat D, Sieger L, Regli WC. Using shape distributions to compare solid models. In *Proceedings of the seventh ACM symposium on Solid modeling and applications*. 2002 June; 273-280.
3. Kaothanthong N, Chun J, Tokuyama T. Distance interior ratio: A new shape signature for 2D shape retrieval. *Pattern Recognition Letters*. 2016; (78): 14-21.
4. Körtgen M, Park GJ, Novotni M, Klein R. 3D shape matching with 3D shape contexts. In *The 7th central European seminar on computer graphics*. 2003 April;(3): 5-17.
5. Liu YS, Ramani K, Liu M. Computing the inner distances of volumetric models for articulated shape description with a visibility graph. *IEEE TPAMI*. 2011;33(12):2538-2544.
6. Osada R, Funkhouser T, Chazelle B, Dobkin D. (2002). Shape distributions. *ACM Transactions on Graphics (TOG)*. 2002; 21(4): 807-832.
7. Möller T, Trumbore B. Fast, minimum storage ray/triangle intersection. In *ACM SIGGRAPH 2005 Courses*. 2005 July; (7).
8. Siggelkow S, Burkhardt H. Improvement of histogram-based image retrieval and classification. In *Pattern Recognition, 2002. Proceedings. 16th International Conference*. 2002; (3): 367-370.
9. Ling H, Jacobs DW. Shape classification using the inner-distance. *IEEE TPAMI*. 2007; 29(2):286-299.

10. Shilane P, Min P, Kazhdan M, Funkhouser T. The princeton shape benchmark. In Shape modeling applications. 2004 June; 167-178.
11. Skiena SS, Smith WD, Lemke P, Reconstructing Sets From Interpoint Distances (extended abstract), The 6th Annual Symposium on Computational Geometry. 1990; 332-339.
12. Tangelder JW, Velkamp RC. (2008). A survey of content based 3D shape retrieval methods. Multimedia tools and applications. 2008; 39(3): 441.
13. Gionis A, Indyk P, Motwani R. Similarity search in high dimensions via hashing. In VLDB. 1999 September; 99(6): 518-529.
14. Ion A, Artner NM, Peyré G, Kropatsch WG, Cohen LD. Matching 2D and 3D articulated shapes using the eccentricity transform. computer vision and image understanding. 2011; 115(6): 817-834.
15. Zhang C, Chen T. Efficient feature extraction for 2D/3D objects in mesh representation. In Image Processing, 2001. Proceedings. 2001 International Conference. 2001; 3: 935-938.
16. Hughes JF, Foley JD. Computer graphics: principles and practice. Person Education. 2014; ISBN 978-0-321-39952-6.
17. Schumacker RA, Brand B, Gilliland MG, Sharp WH. Study for applying computer-generated images to visual simulation. GENERAL ELECTRIC CO DAYTONA BEACH FL APOLLO AND GROUND SYSTEMS. 1969.
18. Fuchs H, Kedem ZM, Naylor BF. On visible surface generation by a priori tree structures. In ACM Siggraph Computer Graphics. 1980, July; 14(3): 124-133.
19. Bentley JL, Multidimensional binary search trees used for associative searching. Communications of the ACM, 1975;18(9): 509-517.

20. Meagher D. Geometric modeling using octree encoding. *Computer graphics and image processing*, 1982;19(2): 129-147.
21. Shotton J, Sharp T, Kipman A, Fitzgibbon A, Finocchio M, Blake A, Moore R. Real-time human pose recognition in parts from single depth images. *Communications of the ACM*. 2013; 56(1): 116-124.
22. Li B, Lu Y, Li C, Godil A, Schreck T, Aono M, Fu H. A comparison of 3D shape retrieval methods based on a large-scale benchmark supporting multimodal queries. *Computer Vision and Image Understanding*. 2015; 131: 1-27.
23. Ankerst M, Kastenmüller G, Kriegel HP, Seidl T. 3D shape histograms for similarity search and classification in spatial databases. In *International Symposium on Spatial Databases*. 1999 July; 207-226.
24. Williams A, Barrus S, Morley RK, Shirley P. An efficient and robust ray-box intersection algorithm. In *ACM SIGGRAPH 2005 Courses*. 2005 July; 9.
25. Wongwises C, Kaothanthong N, Limprasert W. 3DDIR: The distance interior ratio of volumetric models for object recognition. In *Knowledge and Smart Technology (KST), 2017 9th International Conference*. 2017 February; 95-100.

ภาคผนวก

ในภาคผนวกจะแสดงตารางการเปรียบเทียบเวลาในการทำงานของระบบของแต่ละวัตถุและนอกจากนี้ได้แนบการทำ cprofile ซึ่งได้แนบไฟล์การทำ profile ไว้ใน cd เพื่อตรวจสอบเวลาในการทำงานของระบบแบบขนาน



ตารางที่ 6.1: ตารางผลลัพธ์เวลาในการคำนวณของ 3DDIR, 3DDIR Single Thread Partition และ 3DDIR Parallel Partition

ID	จำนวนแผ่น	3DDIR	3DDIR Single Partition	3DDIR Parallel Partition
m0	492	2350.391	414.332	77.45558501
m1	410	1421.796	258.1369998	42.62113095
m3	912	10069.426	1257.378	236.269787
m7	506	1892.571	255.454	48.85058713
m8	698	6860.881	806.9690001	142.6473341
m11	724	4629.084	647.8889999	116.674491
m12	500	2112.664	314.3109999	56.50745702
m13	1125	12921.736	1595.97	318.2910281
m15	724	2965.876	670.573	134.8590572
m21	556	3635.219	618.3460002	120.9148998
m22	620	3649.19	570.9219999	114.5671201
m23	878	8079.549	1056.44	215.4202233
m24	284	406.204	108.0939999	20.89970708
m27	1024	6742.346	640.6660001	130.796916
m30	800	6023.689	749.0049999	149.0324509
m34	462	1375.868	212.135	43.61481118
m38	930	10701.196	1199.227	274.7303581
m42	265	169.582	38.27600026	8.155002117
m43	60	5.671999931	2.342999935	0.54211092
m44	798	7739.039	674.757	136.655154
m47	165	108.063	25.51600003	5.261883021
m48	824	8806.936	689.2179999	136.233151
m50	182	86.329	26.28100014	4.840672255
m51	598	2958.366	365.1129999	75.13705993

ID	จำนวนแผ่น	3DDIR	3DDIR Single Partition	3DDIR Parallel Partition
m52	578	1962.339	420.3710001	85.55648113
m58	950	6233.914	860.994	166.5001462
m59	1680	101.266	60.53200006	12.25985193
m61	338	631.3960001	137.813	26.38246704
m62	478	1159.192	172.201	32.95472193
m64	444	994.378	180.8760002	40.26946402
m69	568	2139.762	416.4090002	84.13565612
m70	560	1740.493	350.108	68.37492919
m71	836	4122.998	670.8539999	133.9097202
m72	590	2709.982	434.0369999	81.64256788
m73	564	1323.461	257.2230001	49.41190696
m74	336	258.2349999	86.95700002	17.27915812
m75	512	2015.058	301.3530002	59.10907507
m78	734	4683.535	841.6309998	224.1000001
m79	226	179.5159998	68.2440002	18.64100003
m83	154	65.0940001	21.36000013	7.108999968
m86	245	340.3610001	100.25	28.76600004
m93	464	1583.083	333.3930001	90.20300007

ตารางที่ 6.2: ตารางผลลัพธ์ความเร็วของการคำนวณของระบบ D2, 3DDIR และ 3DDIR Parallel Partition

ID	จำนวนแผ่น	D2	3DDIR	3DDIR Parallel Partition
m0	492	1.203	2350.391	99.7190001
m1	410	1.563	1421.796	54.28999996
m2	6667	3.391	69658.23746	641.4619999
m3	912	3.172	10069.426	300.635
m4	8216	3.419	82788.7472	790.6900001
m5	8773	3.407	77469.67373	722.9659998
m7	506	0.953	1892.571	72.39500022
m8	698	2.641	6860.881	199.2819998
m9	902	3.39	7622.277651	418.312
m10	2628	3.406	22149.53933	505.6460001
m11	724	1.813	4629.084	143.8100002
m12	500	1.047	2112.664	74.7809999
m13	1125	3.811	12921.736	412.3269999
m14	1681	3.812	12100.04334	423.5210001
m15	724	2.547	2965.876	166.875
m16	5162	3.398	54353.20961	479.227
m17	3000	3.829	31653.85402	496.7590001
m18	8324	3.485	82662.27834	641.8799999
m19	5490	4.84	49310.88783	519.5209999
m20	3042	3.797	25586.57901	452.4400001
m21	556	2.281	3635.219	155.9690001
m22	620	1.578	3649.19	151.5409999
m23	878	3.078	8079.549	279.345
m24	284	0.359	406.204	30.92200017
m25	2494	3.39	21731.17386	395.4169998

ID	จำนวนแผ่น	D2	3DDIR	3DDIR Parallel Partition
m26	10274	3.397	108188.291	713.3270001
m27	1024	2.344	6742.346	163.6100001
m28	11660	4.75	104817.9999	779.8239999
m29	5878	4.016	47068.30494	493.2920001
m30	800	2.515	6023.689	182.358
m31	3548	4.493	22682.66895	358.418
m32	2246	4.281	12703.56397	392.023
m33	6164	4.16	64781.78248	530.4590003
m34	462	0.984	1375.868	55.82799983
m35	1394	4.271	14924.28424	284.6000001
m36	2531	3.829	26687.02381	409.1399999
m37	1130	3.748	12218.65984	300.9750001
m38	930	3.438	10701.196	346.5809999
m39	1156	4.125	12527.21964	388.178
m40	5390	4.094	56826.81178	559.8599999
m41	3709	4.438	39688.92615	482.6360002
m42	265	0.203	169.582	11.48399997
m43	60	0.031	5.671999931	1.641000032
m44	798	2.64	7739.039	171.0479999
m45	33072	4.128	230913.9965	2222.389
m46	1094	3.719	5523.659558	347.108
m47	165	0.187	108.063	7.969000101
m48	824	2.859	8806.936	177.688
m49	4372	3.984	24337.31823	438.8469999
m50	182	0.14	86.329	8.233999968
m51	598	1.407	2958.366	97.29700017
m52	578	1.468	1962.339	111.2719998

ID	จำนวนแผ่น	D2	3DDIR	3DDIR Parallel Partition
m56	10950	4.343	110635.3824	703.0650001
m57	36468	4.063	242456.38	2541.204
m58	950	3.563	6233.914	217.348
m59	1680	0.031	101.266	19.15700006
m60	3208	4.523	17168.30741	345.3
m61	338	0.812	631.3960001	38.8440001
m62	478	0.828	1159.192	44.6099999
m63	1792	4.188	9631.777208	302.9910002
m64	444	0.828	994.378	50.20099998
m65	1930	4.485	10345.11688	287.9099998
m66	8198	4.203	84323.18988	743.184
m68	1098	4.438	11809.55393	274.2840002
m70	560	1.843	1740.493	89.23399997
m71	836	2.516	4122.998	168.816
m72	590	1.781	2709.982	109.2190001
m73	564	1.215	1323.461	66.3440001
m74	336	0.428	258.2349999	22.84299994
m75	512	1.613	2015.058	80.37299991
m76	3164	4.145	32687.48217	389.4200001
m77	23022	4.407	168524.3326	1656.741
m78	734	3.657	4683.535	221.4189999
m80	2104	3.969	11540.08242	340.661
m81	1673	3.984	18113.62019	459.053
m82	8276	3.75	86073.96662	612.0179999
m83	154	0.188	65.0940001	6.687000036
m84	5263	3.907	47842.76027	408.9619999
m85	3907	4.329	36058.32877	483.5550001

ID	จำนวนแผ่น	D2	3DDIR	3DDIR Parallel Partition
m86	245	0.625	340.3610001	29.67199993
m87	1331	4.39	12633.04651	305.9490001
m88	3220	4.469	40029.826	406.0139999
m90	3526	4.156	42719.814	379.6429999
m91	3169	4.201	35548.051	419.908
m92	3888	4	39470.974	441.595
m93	464	1.563	1583.083	86.75
m94	6328	3.935	52383.789	532.783
m95	1908	4.25	13223.161	311.954
m97	7083	3.985	85648.691	514.595
m98	8236	4.515	85504.879	583.9550002
m99	2888	3.916	25140.209	350.095
m100	1208	3.524	8351.985	263.4850001
m101	2204	3.64	15091.214	454.642
m102	226	0.329	107.75	14
m104	462	1	1301.138	51.51600003
m105	260	0.531	292.41	21.04699993
m106	690	2.157	4208.914	146.079
m107	6052	4.428	74303.684	502.158
m108	22258	4.625	188586.638	1345.042
m109	2568	3.594	32180.113	381.8539999
m110	902	3.458	9458.062	194.1100001
m111	2442	3.703	30732.117	344.016
m112	15294	3.5	146823.169	843.5179999
m113	11092	4.047	73714.606	895.503
m114	5418	4.519	67294.12	551.9859998
m115	5399	3.407	59985.648	566.2839999

ID	จำนวนแผ่น	D2	3DDIR	3DDIR Parallel Partition
m117	4316	4.143	43066.91	487.642
m120	870	3.906	7630.15	214.454
m121	882	3.906	7931.714	234.4070001
m122	830	3.265	8315.722	183.9849999
m123	890	4.078	10161.878	222.4069998
m124	862	4.141	9815.189	222.7820001
m125	2500	4.894	31353.088	331.329
m126	1018	4.084	12901.496	289.142
m127	1796	4.234	22721.911	380.4230001
m128	853	3.265	9088.421	230.1569998
m130	1865	3.734	23358.671	358.5170002
m131	3012	4.735	35516.692	405.97
m132	4049	3.703	44851.258	420.642
m133	4253	4.609	53326.856	528.892
m134	3615	5.099	40473.818	488.6730001
m137	3254	3.594	36244.808	389.6109998
m141	3012	3.391	29524.16	409.079
m146	3460	5.242	32072.077	418.0640001
m147	8559	4.125	63600.797	584.2049999
m148	7707	4.453	55862.745	682.1430001
m267	7774	4.457	56717.778	637.0639999
m268	484	0.922	897.137	63.2190001
m270	8368	3.863	58821.893	731.7059999
m271	16564	3.547	99318.677	1080.863
m272	9039	4.015	104616.531	702.1429999
m273	942	5.172	9175.318	308.0639999
m276	4587	3.594	41808.043	467.0010002

ID	จำนวนแผ่น	D2	3DDIR	3DDIR Parallel Partition
m277	874	4.828	5617.735	187.6569998
m278	646	2.111	2408.931	125.859
m279	524	1.031	1151.77	76.0480001
m280	5504	4.313	68168.994	504.471
m282	3587	3.499	37378.355	408.845
m286	8066	3.516	67218.801	518.408
m288	1208	4.765	8200.445	230.2350001
m289	5218	5.016	35224.842	416.283
m325	2878	3.422	21354.186	462.0319998
m326	2828	4.139	20722.219	446.0800002
m327	768	2.047	3449.609	137.313
m328	12072	4.317	89815.508	624.658
m329	12378	3.453	92228.3	668.346
m330	1710	4.203	21905.791	333.158
m331	837	2.516	7589.336	162.2979999
m332	2392	5.047	30215.511	302.141
m333	2043	5.443	24802.74	320.72
m334	204	0.313	130.797	21.25
m336	3110	3.5	39402.971	372.142
m337	3846	3.758	46874.509	405.1729999
m338	3090	5.774	31339.369	381.3759999
m339	4466	4.594	40331.734	431.7510002
m382	400	1.422	898.068	96.14099979
m383	4460	3.884	55927.606	667.1900001

ID	จำนวนแผ่น	D2	3DDIR	3DDIR Parallel Partition
m387	1668	4.109	20582.935	578.8299999
m389	5170	4.453	52403.313	766.783
m391	274	1.031	700.661	46.07799983
m393	326	1.469	820.064	46.9380002
m394	1246	3.907	15592.315	370.4859998
m395	696	3.79	9307.492	277.017
m396	470	2.89	4842.948	215.641
m397	2680	5.152	33550.112	476.5639999
m398	946	4.641	12030.506	328.016



ประวัติผู้เขียน

ชื่อ นายจิณณวัตร วงศ์วิเศษ
วันเดือนปีเกิด 27 ตุลาคม 2535
วุฒิการศึกษา ระดับปริญญาตรี สาขาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์และเทคโนโลยี
มหาวิทยาลัยธรรมศาสตร์
ปีการศึกษา 2557

ผลงานทางวิชาการ

3DDIR: The distance interior ratio of volumetric models for object recognition (25)