การปรับแก้และเปลี่ยนรูปของแบบจำลองฟันใน 3 มิติเพื่อช่วยงานทันตกรรม

นายปฐมพงศ์ พันนุรัตน์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

มหาวิทยาลัยเทคโนโลยีสุรนารี

ปีการศึกษา 2550

EDITING AND DEFORMING 3D MODEL

IN ORTHODONTICS

Pathomphong Phannurat

A Thesis Submitted in Partial Fulfillment of the Requirements for

the Degree of Master of Engineering in Computer Engineering

Suranaree University of Technology

Academic Year 2007

การปรับแก้และเปลี่ยนรูปของแบบจำลองฟันใน 3 มิติเพื่อช่วยงานทันตกรรม

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

คณะกรรมการสอบวิทยานิพนธ์

(รศ. ดร.กิตติศักดิ์ เกิดประสพ)
ประธานกรรมการ

(อ. ดร.ปรเมศวร์ ห่อแก้ว)
กรรมการ (อาจารย์ที่ปรึกษาวิทยานิพนธ์)

(ผศ. ดร.พิชโยทัย มหัทธนาภิวัฒน์)
กรรมการ

(ศ. ดร.ไพโรจน์ สัตยธรรม)
รักษาการแทนรองอธิการบดีฝ่ายวิชาการ

(รศ. น.อ. ดร.วรพจน์ ขำพิศ)
คณบดีสำนักวิชาวิศวกรรมศาสตร์

ปฐมพงศ์ พันนุรัตน์ : การปรับแก้และเปลี่ยนรูปของแบบจำลองฟันใน 3 มิติเพื่อช่วยงานทันตกรรม (EDITING AND DEFORMING 3D MODEL IN ORTHODONTICS) อาจารย์ที่ปรึกษา : อ. ดร.ปรเมศวร์ ห่อแก้ว, 93 หน้า.

คอมพิวเตอร์ช่วยงานทางการแพทย์ (Computer-aided Medical) มีบทบาทสำคัญอย่างมากในการบำบัดรักษาและวินิจฉัย ปัจจุบัน คอมพิวเตอร์ถูกนำมาใช้ประโยชน์ในทางการแพทย์มากขึ้น ทั้งในด้านการผ่าตัดศัลยกรรมรวมทั้งด้านทันตกรรมจัดฟัน (Orthodontics) ซึ่งการทำทันตกรรมจัดฟันในแต่ละครั้ง ทันตแพทย์จะศึกษาและวิเคราะห์จากภาพถ่ายเอกซเรย์ด้านหน้าของใบหน้าแล้วร่างเค้าโครงของใบหน้า กราม ฟัน พร้อมทั้งคำนวณระยะและมุมต่าง ๆ ด้วยตนเอง ซึ่งเป็นกระบวนการที่ต้องอาศัยความเชี่ยวชาญเป็นพิเศษ ดังนั้นจึงมีความจำเป็นที่ต้องใช้คอมพิวเตอร์เข้ามาช่วยเพื่ออำนวยความสะดวกให้กับทันตแพทย์ และช่วยให้การรักษามีประสิทธิภาพมากขึ้น โดยการแสดงผลภาพเป็นสามมิติ (3D Reconstruction) เพื่อให้มีความเหมือนจริงกับฟันของผู้เข้ารับการรักษาและสามารถจำลอง (Simulation) รูปแบบการจัดเรียงของฟันแต่ละระยะว่าจะเปลี่ยนไปอย่างไรในระหว่างการรักษา นอกจากนี้ข้อมูลที่ได้สามารถนำไปช่วยในการออกแบบวัสดุจัดฟัน

งานวิจัยนี้จึงได้เสนอเทคนิควิธีการปรับแก้และเปลี่ยนรูปของแบบจำลองฟันใน 3 มิติ ซึ่งพัฒนาขึ้นเพื่อช่วยงานทันตกรรมจัดฟัน โดยเริ่มจากกระบวนการลดทอนรายละเอียดของวัตถุ (Decimation) ซึ่งมีความจำเป็นเนื่องจากแบบจำลองฟันทางการแพทย์ส่วนใหญ่มีความละเอียดสูงทำให้สิ้นเปลืองเวลาในการแสดงผล กระบวนการถัดไปคือ การตัดพื้นผิวของวัตถุ (Surface cutting) ซึ่งเป็นการแยกโครงสร้างของฟันและเหงือกออกจากกัน หลังจากนั้นจะเป็นการวิเคราะห์การบิดตัวของเหงือกโดยอาศัยแบบจำลองมวลสปริง (Mass-spring model) เพื่อจำลองการจัดฟัน ซอฟต์แวร์ต้นแบบและเทคนิควิธีซึ่งเป็นผลจากงานวิจัย สามารถเพิ่มประสิทธิภาพในการบำบัดรักษา และสามารถผนวกรวมกับซอฟต์แวร์ระบบภาพทางการแพทย์เพื่อช่วยงานทันตกรรมและเผยแพร่สู่กลุ่มเป้าหมายในเชิงพาณิชย์

สาขาวิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนักศึกษา_____

ปีการศึกษา 2550 ลายมือชื่ออาจารย์ที่ปรึกษา_____

PATHOMPHONG PHANNURAT : EDITING AND DEFORMING 3D MODEL IN ORTHODONTICS. THESIS ADVISOR : PARAMATE HORKAEW, Ph. D., 93 PP.

ORTHODONTICS SIMULATION/MESH DECIMATION/SURFACE CUTTING/ MASS-SPRING MODEL

Computer-aided Medical has important role in diagnosis and treatment planning. Computer system can be useful in surgery treatment include orthodontics. Traditionally, the orthodontist use cephalometric projections to plan their treatments which need specialist. However, it is not convenient to store and compare orthodontic patient record of each patient for each period. Therefore, computer technology is used to simulate in orthodontics on 3-dimention space for increasement efficiency treatment. It can simulate the teeth alignment for each period. In addition, output data can used to design orthodontic material such as brackets.

We present an Orthodontics treatment simulator, which has three components: Mesh Decimation, Surface Cutting and Mass-spring Model. Medical model is generated form medical software which results of the model has a large size.Models of this size are genally not practical since rendering speeds, cost computation and memory requirements are proportional to the number of polygons in the model. Mesh decimation uses local operations on geometry and topology to reduce the number of triangles in our models. The extraction of tooth structure from the gingival surface use surface cutting technique. We apply mass-spring system to soft tissue

deformation in 3D orthodontic simulation, the movement of which is evaluated using the numerical integration of the fundamental law of dynamics. The software phototype and algorithms can increase efficiency orthodontic treatment and it can be integrated with medical software for distributed to commercial software.

School of Computer Engineering Student's Signature_____

Academic Year 2007 Advisor's Signature_____

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงด้วยดี ผู้วิจัยขอกราบขอบพระคุณ บุคคล และกลุ่มบุคคลต่าง ๆ ที่ได้กรุณาให้คำปรึกษา แนะนำ ช่วยเหลือ อย่างดียิ่ง ทั้งในด้านวิชาการ และด้านการดำเนินงานวิจัย ดังนี้

อาจารย์ ดร. ปรเมศวร์ ห่อแก้ว อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ให้คำแนะนำปรึกษา และช่วยชี้แนะแนวทางการแก้ปัญหา รวมทั้งช่วยตรวจทาน และแก้ไขวิทยานิพนธ์เล่มนี้จนเสร็จสมบูรณ์

ดร. จันทร์จิรา สินทนะโยธิน นักวิจัยที่ปรึกษาโครงการทุนสถาบันบัณฑิตวิทยาศาสตร์และเทคโนโลยีไทย ที่ให้คำแนะนำด้านวิชาการ และให้กำลังใจแก่ผู้วิจัยมาโดยตลอด

สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ และศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ ที่สนับสนุนเงินทุนในการวิจัย

ศูนย์เทคโนโลยีทางทันตกรรมขั้นสูง ที่สนับสนุนข้อมูลที่ใช้ในการทดสอบ

ขอขอบคุณ คุณสิทธิชัย ป้อมทอง คุณณัฐพล พันนุรัตน์ คุณอภิชัย ฤทธิ์ธงชัยเลิศ และบัณฑิตศึกษาสาขาวิชาวิศวกรรมคอมพิวเตอร์ทุกท่าน ที่ให้คำปรึกษาและช่วยเหลือผู้วิจัยด้วยดีมาโดยตลอด

สุดท้ายนี้ ผู้วิจัยขอขอบคุณคณาจารย์ทุกท่านที่ได้ประสิทธิ์ประสาทวิชาความรู้ต่าง ๆ ทั้งในอดีต และปัจจุบัน และขอกราบขอบพระคุณบิดา มารดา ที่ให้ความรัก กำลังใจ การอบรมเลี้ยงดู และส่งเสริมการศึกษาเป็นอย่างดีมาโดยตลอด จนทำให้ผู้วิจัยประสบความสำเร็จในชีวิตเรื่อยมา

ปฐมพงศ์ พันนุรัตน์

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....ก
บทคัดย่อภาษาอังกฤษ.....ข
กิตติกรรมประกาศ.....ง
สารบัญ.....จ
สารบัญตาราง.....ช
สารบัญรูป.....ซ

บทที่

1 บทนำ.....1
1.1 ความสำคัญและที่มาของปัญหาการวิจัย.....1
1.2 วัตถุประสงค์การวิจัย.....5
1.3 ขอบเขตการวิจัย.....6
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....8
1.5 โครงสร้างของวิทยานิพนธ์.....8
2 ปริทัศน์วรรณกรรมและงานวิจัยที่เกี่ยวข้อง.....9
2.1 กระบวนการจำลองการจัดฟัน (Orthodontic Simulation).....9
2.2 การลดทอนรายละเอียดของวัตถุ (Mesh Decimation).....14
2.3 การตัดพื้นผิว (Surface Cutting).....19
2.4 การสร้างแบบจำลองมวลสปริง (Mass-spring Model).....26
2.5 สรุปงานวิจัยที่เกี่ยวข้อง.....31
3 ระเบียบวิธีวิจัยและกรอบแนวคิด.....32
3.1 ระเบียบวิธีวิจัย.....32
3.2 ข้อมูลที่ใช้ในการวิจัย.....33
3.2.1 โครงสร้างแฟ้มข้อมูล STL.....33
3.2.2 ข้อมูลแบบจำลองขากรรไกรบนและขากรรไกรล่าง.....35

สารบัญ (ต่อ)

หน้า

3.3 ขั้นตอนวิธีการจำลองการจัดฟัน 36
3.3.1 โครงสร้างข้อมูล 36
3.3.2 เทคนิคการลดทอนรายละเอียดของวัตถุ (Mesh Decimation) 38
3.3.3 เทคนิคการตัดพื้นผิว (Surface Cutting) 42
3.3.4 เทคนิคการสร้างแบบจำลองมวลสปริง (Mass-Spring Model) 54
3.4 สรุปวิธีดำเนินการวิจัย 65
4 การทดสอบและอภิปรายผล 66
4.1 ผลการทดสอบการลดทอนรายละเอียดของวัตถุ (Mesh Decimation) 66
4.2 ผลการทดสอบการตัดพื้นผิวของวัตถุ (Surface cutting) 72
4.3 ผลการทดสอบการสร้างแบบจำลองมวลสปริง (Mass-spring model) 79
4.4 การอภิปรายผล 81
5 สรุปผลการวิจัยและข้อเสนอแนะ 82
5.1 สรุปผลการวิจัย 82
5.2 ข้อจำกัดของงานวิจัย 84
5.3 ข้อเสนอแนะในงานวิจัยต่อไป 84
รายการอ้างอิง 86
ภาคผนวก
ภาคผนวก ก. บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่ 91
ประวัติผู้เขียน 93

สารบัญตาราง

ตารางที่ **หน้า**

1.1 แสดงราคาของตัวอย่างชุดซอฟต์แวร์สำหรับช่วยงานทางทันตกรรม 3
2.1 เทคนิควิธีการสร้างข้อมูลข้อมูลการบำบัดรักษาของผู้ป่วยแบบสามมิติ (Mah and Bumann, 2001) 10
3.1 รูปแบบแฟ้มข้อมูล STL (Binary) 35
4.1 แสดงผลการทดสอบขั้นตอนวิธีลดทอนรายละเอียดกับแบบจำลองขากรรไกรล่าง 66
4.2 แสดงผลการทดสอบขั้นตอนวิธีลดทอนรายละเอียดกับแบบจำลองขากรรไกรบน 66
4.3 แสดงผลการทดสอบการสร้างแบบจำลองมวลสปริงด้วยวิธีออยเลอร์ 78
4.4 แสดงผลการทดสอบการสร้างแบบจำลองมวลสปริงด้วยวิธีรุงเก-คุตตาอันดับสี่ 78

สารบัญรูป

รูปที่ **หน้า**

1.1 กระบวนการวิจัยการประมวลผลภาพทางการแพทย์เพื่อการบริหารจัดการผู้ป่วย (ปรับปรุงจาก The Research Council of Norway, 2006) 2
1.2 แผนผังแสดงส่วนประกอบของโปรแกรม CephSmile.....4
1.3 แผนผังแสดงส่วนประกอบของซอฟต์แวร์ต้นแบบ 5
1.4 แสดงผลการลดทอนรายละเอียดของแบบจำลองฟัน 7
1.5 แสดงผลการแยกองค์ประกอบของฟันและเหงือกออกจากกัน (a) แสดงผลการเปลี่ยนรูปร่างของเหงือกเมื่อฟันเลื่อนตำแหน่ง จะเห็นได้ว่า บริเวณเหงือกที่ติดกับฟันมีการเปลี่ยนรูปร่างตามทิศทางการเลื่อนของฟัน (b) 7
2.1 โครงสร้างของระบบคอมพิวเตอร์ช่วยจำลองการจัดฟัน (Li et al., 2005) 12
2.2 ส่วนโปรแกรมพื้นฐานของระบบจำลองการจัดฟัน.....12
2.3 แสดงผลของการจำลองการเคลื่อนของฟัน (Rodrigues et al., 2006) 13
2.4 ประเภทของจุดบนโครงข่าย (Schroeder et al., 1992) 15
2.5 ระยะทางจากจุดกับระนาบเฉลี่ย (a) การสร้างสามเหลี่ยมใหม่ด้วยวิธีการแบ่งระนาบ (b) (Schroeder et al, 1992) 15
2.6 เทคนิคการยุบหรือลดด้านของสามเหลี่ยม (Garland and Heckbert, 1998).....18
2.7 ขั้นตอนการทำงานของ Cutting callback (ปรับปรุงจาก Bruyns and Senger, 2001) 19
2.8 แสดงเส้นทางการตัดที่ลากผ่านด้านของสามเหลี่ยม (a) และแสดงการสร้างสามเหลี่ยมใหม่ (b) ปรับปรุงจาก (Bruyns and Senger, 2001) 20
2.9 ส่วนประกอบของสามเหลี่ยมที่ถูกตัดผ่านด้วยระนาบ (Pintilie and McInerney, 2003) 21
2.10 แสดงการแบ่งสามเหลี่ยมที่เส้นรอบขอบตัดผ่าน (Pintilie and McInerney, 2003) 22
2.11 แสดงจุดที่ถูกเพิ่มในเส้นทางการตัด (a) การสร้างจุดตัดบนเส้นรอบขอบ (b) และการปรับปรุงโครงข่ายที่จุดสุ่มบนเส้นรอบขอบ (c) (Pintilie and McInerney, 2003) 22
2.12 แสดงการเริ่มต้นกระบวนการตัดโครงข่าย (a) และการสร้างโครงข่ายใหม่ (B) (HuyVeit et al., 2006) 23

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

2.13 แสดงองค์ประกอบของฟันปลอม (Chi et al., 004) 25
2.14 การเลื่อนตำแหน่งของ snake (Ji et al., 2006) 25
2.15 ประเภทของสปริง สำหรับแบบจำลองมวลสปริง (Provot, 1995) 27
2.16 แบบจำลองกล่องที่สามารถเปลี่ยนรูปได้ (a) ภายใต้แรงโน้มถ่วงด้วยการกำหนด ค่าแปรที่ต่างกัน (b) รายละเอียดเดียว (c) รายละเอียดหลายระดับโดยกำหนดค่า k=l/1 และค่าคงที่การหน่วงดังสมการที่ 2.8 (Paloc et al., 2002) 29
3.1 แสดงกระบวนการสร้างแฟ้มข้อมูล STL 33
3.2 แสดงทิศทางของสามเหลี่ยม (a) และกฎการเชื่อมโยงจุด (b) 34
3.3 แบบจำลองของขาไกรกรรบน (ขวา) และแบบจำลองของขาไกรกรรล่าง (ซ้าย) 36
3.4 แสดงโครงสร้างข้อมูลของโครงข่ายแบบสามเหลี่ยม.....37
3.5 แสดงโครงสร้างข้อมูลของด้านสามเหลี่ยม (ซ้าย) และโครงสร้างข้อมูลของแบบจำลองมวลสปริง (ขวา) 37
3.6 แสดงกระบวนการลดทอนรายละเอียดของวัตถุ 38
3.7 ประเภทจุดบนโครงข่ายสามเหลี่ยม.....39
3.8 ประเภทจุดอย่างง่าย (Simple vertex) (a) และFeature edge (b) ดูรายละเอียดในเนื้อหา 39
3.9 ระนาบเฉลี่ยในรูปแบบจุดและเวกเตอร์ตั้งฉาก 40
3.10 แสดงการสร้างสามเหลี่ยมใหม่ (b) จากช่องว่างที่เกิดจากการดึงจุดออก (a) 42
3.11 แสดงกระบวนการตัดพื้นผิว.....43
3.12 เทคนิค crossing test (a), upward edge (b) และ downward edge (c) 47
3.13 แสดงจุด Landmark ที่ไม่อยู่บนพื้นผิว (a) และผลลัพธ์ (b) 49
3.14 แสดงจุดขอบของฟันที่ตัดแล้ว (a) และการกำหนดจุดตัดของฟัน กรณีที่ ฟันมีโครงสร้างติดกัน (b).....52
3.15 แสดงจุดกำหนดขอบเขตการตัด (a) และการเชื่อมโยงจุดขอบเขตการตัด (b) 52
3.16 แสดงเส้นทางการตัดลากผ่านด้านสามเหลี่ยม กรณีที่จุด landmark ทั้งสอง 53
3.17 แสดงกระบวนการสร้างแบบจำลองมวลสปริง 55

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

3.18 แสดงจุดขอบของแบบจำลองฟันและเหงือก 56
3.19 แสดงการกำหนดตัวแทนของแบบจำลองมวลสปริง 56
3.20 แสดงการเข้าสู่สภาวะสมดุลย์ของระบบมวลสปริง 64
3.21 แสดงการจำลองการเปลี่ยนแปลงรูปร่างของเหงือก 65
4.1 แสดงเวลา (วินาที) ที่ใช้ในการลดทอนรายละเอียดของแบบจำลอง ขากรรไกรล่างและบน 68
4.2 แสดงเวลา (มิลลิวินาที) ที่ใช้ในการแสดงผลแบบแบนราบ (flat shading) ของแบบจำลองขากรรไกรล่างและบน 68
4.3 แสดงเวลา (มิลลิวินาที) ที่ใช้ประมวลผลสำหรับขั้นตอนวิธีการสร้างกราฟ ด้านสามเหลี่ยม 69
4.4 เปรียบเทียบแบบจำลองขากรรไกรล่างที่ถูกลดทอนรายละเอียดระดับต่าง ๆ 70
4.5 เปรียบเทียบแบบจำลองขากรรไกรบนที่ถูกลดทอนรายละเอียดระดับต่าง ๆ 71
4.6 ฟันแต่ละซี่ของแบบจำลองขากรรไกรล่าง (ขวา) และบน (ซ้าย) 72
4.7 แสดงระยะห่างระหว่างจุด landmark บนพื้นผิวของแบบจำลองที่ถูกลดทอน รายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม 72
4.8 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันซี่ A ของแบบจำลอง ขากรรไกรบนที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม 73
4.9 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันซี่ B ของแบบจำลอง ขากรรไกรบนที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม 74
4.10 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันซี่ C ของแบบจำลอง ขากรรไกรบนที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม 74
4.11 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันซี่ D ของแบบจำลอง ขากรรไกรบนที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม 75
4.12 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันซี่ A ของแบบจำลอง ขากรรไกรล่างที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม 75

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

4.13 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันซี่ B ของแบบจำลองขากรรไกรล่างที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม 76
4.14 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันซี่ C ของแบบจำลองขากรรไกรล่างที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม 76
4.15 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันซี่ D ของแบบจำลองขากรรไกรล่างที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม 77
4.17 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันทั้ง 4 ซี่ของแบบจำลองขากรรไกรล่างที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม 78
4.18 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันทั้ง 4 ซี่ของแบบจำลองขากรรไกรบนที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม 78
4.19 แสดงการเปลี่ยนแปลงรูปร่างของแบบจำลองเหงือกซึ่งใช้วิธีออยเลอร์ในแต่ละรอบ 80
4.20 แสดงการเปลี่ยนแปลงรูปร่างของแบบจำลองเหงือกซึ่งใช้รุงเก-คุตตาอันดับสี่ในแต่ละรอบ.....80

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มาของปัญหาการวิจัย

ความเจริญก้าวหน้าของการประมวลผลภาพ (Image processing) ได้พัฒนาและใช้ประโยชน์กันอย่างกว้างขวางสำหรับงานในหลาย ๆ ด้าน ตัวอย่างเช่น ทางด้านสื่อสารโทรคมนาคม Lin and Ming (2008) นำเทคนิคการประมวลผลภาพมาใช้กำจัดผลกระทบจากการฉายแสงที่คลาดเคลื่อนบนรูปที่มีความละเอียดสูง ทางด้านเทคโนโลยีอาหารและเกษตรกรรม Lee et al. (2008) ใช้เทคนิคการเปรียบเทียบภาพของลูกอินทผลัมเพื่อปรับปรุงคุณภาพของผลิตภัณฑ์ และ Boese et al. (2008) ได้วิเคราะห์โรค Zostera marina ซึ่งเป็นโรคที่เกิดกับหญ้าปลาไหล (eelgrass) ด้วยภาพถ่ายทางด้านดาราศาสตร์ Luz, Berry, and Roos-Serote (2008) ได้เสนอวิธีการตรวจจับกลุ่มเมฆด้วยภาพถ่ายทางอากาศ ทางด้านชีวเคมี Toropova et al. (2008) ได้ทำการสร้างแบบจำลองโครงสร้างยีน RNA ในแบคทีเรีย MS2 และทางด้านการแพทย์ Yun et al. (2008) ได้นำเสนองานวิจัยเกี่ยวกับการวินิจฉัยโรคเบาหวานที่ตาในแต่ละระยะโดยใช้ภาพถ่ายเรตินา

การพัฒนาระบบภาพทางการแพทย์ (Medical Imaging) เพื่อช่วยในการวินิจฉัยและบัดรักษาในปัจจุบันระบบภาพทางการแพทย์ ได้วิวัฒนาการมาจากการเป็นเครื่องมือเพื่อการวินิจฉัย ไปสู่การบูรณาการในขั้นตอน การบำบัดรักษา และ การวิจัยทางการแพทย์ ดังนี้

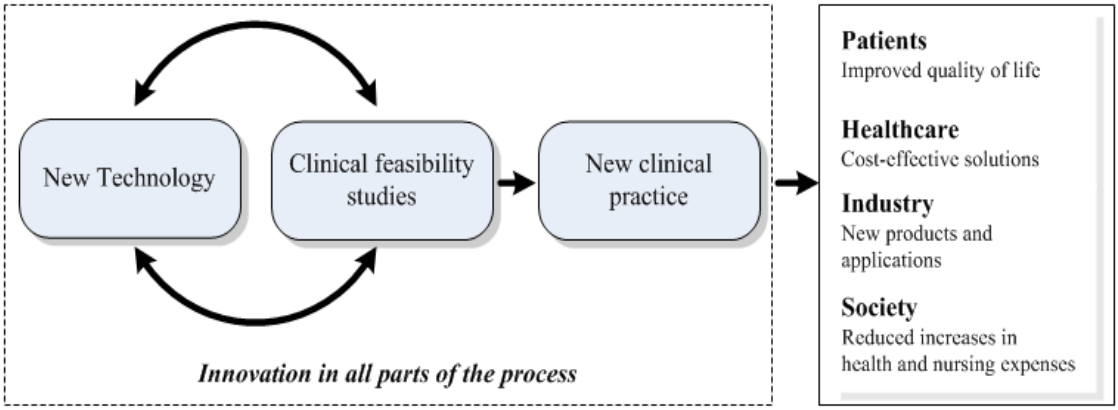
การบำบัดรักษา

1) บริหารจัดการผู้ป่วย (Patient Management) The Research Council of Norway (2006) ได้เสนอกระบวนการวิจัยทางระบบภาพทางการแพทย์เพื่อการบริหารจัดการผู้ป่วยให้มีประสิทธิภาพ แสดงดังรูปที่ 1.1

2) สนับสนุนขั้นตอนการบำบัดรักษาอย่างต่อเนื่อง (Therapeutic Intervention) Swanson, Alvord, and Murray (2004) ได้สร้างแบบจำลองของสมองเพื่อวิเคราะห์การเคลื่อนที่และการขยายตัวของเซลล์เนื้องอกในสมอง อย่างต่อเนื่องเป็นระยะเวลา 640 วัน และ Rahman et al. (2006) ศึกษาขั้นตอนการบำบัดรักษาอย่างต่อเนื่องในการโรครักษามะเร็งเต้านมที่แพร่กระจายเข้าสู่กระดูก (Breast cancer bone metastasis) ด้วยสารอินโดล-ทรี-คาร์บินอล (indole-3-carbinol) โดยอาศัยการปลูกถ่ายเซลล์มะเร็งในหนู

3) สนับสนุนการใช้หุ่นยนต์ร่วมสำหรับการศัลยกรรมในบริเวณจำกัด (Minimal Invasive Surgery) Hayashibe et al. (2006) นำหุ่นยนต์มาช่วยในการจำลองการผ่าตัดด้วยการคำนวณการ

เลียนแบบการเคลื่อนของมนุษย์และการใช้ระบบการประมวลผลภาพทางการแพทย์ในการจำลองการผ่าตัดด้วย และ Worn and Muhling (2001) ได้นำระบบคอมพิวเตอร์และหุ่นยนต์มาใช้ในการผ่าตัดกะโหลกศีรษะและใบหน้า



รูปที่ 1.1 กระบวนการวิจัยการประมวลผลภาพทางการแพทย์เพื่อการบริหารจัดการผู้ป่วย (ปรับปรุงจาก The Research Council of Norway, 2006)

การวิจัยทางการแพทย์

1) การวิจัยและคิดค้นเวชภัณฑ์รักษา (Drug Designs) และการบำบัดแผนใหม่ Taylor et al. (2008) คิดค้นเวชภัณฑ์รักษาจากโปรตีน cAMP โดยจำลองการรวมตัวของโมเลกุลโปรตีน และ Chappell et al. (2008) ได้ศึกษาแบบจำลองการเคลื่อนที่ของยาซึ่งรวมตัวกับเซลล์ไซเคิล เพื่อวิเคราะห์ผลกระทบที่มีต่อเซลล์ของมนุษย์ ผลที่ได้สามารถนำไปใช้ช่วยด้านการออกแบบยา
2) การสร้างแบบจำลองทางชีวะกลศาสตร์ (Biomechanical Modeling) เพื่อศึกษาด้านกายวิภาค และสรีรวิทยา (Computational Anatomy and Physiology) Wang et al. (2005) ได้ศึกษาการสร้างแบบจำลองทางชีวะกลศาสตร์ของกระดูกไขสันหลังโดยใช้เทคนิควิธี Marching-cubes และ Leadini et al. (2006) พัฒนาซอฟต์แวร์ประมวลผลภาพสามมิติ สำหรับการวิเคราะห์การเคลื่อนที่ของระบบกล้ามเนื้อ และกระดูก

แต่อย่างไรก็ตาม การพัฒนาระบบภาพทางการแพทย์ในประเทศไทย เพื่อช่วยวิเคราะห์และวางแผนรักษาทางทันตกรรมจัดฟัน (Orthodontics) ยังมีไม่มากนัก (Thanwadee and Parun, 2002) ในขณะที่ปัจจุบันมีประชาชนจำนวนมากเข้ารับการรักษาทางทันตกรรมจัดฟัน เพื่อการแก้ไขปัญหา

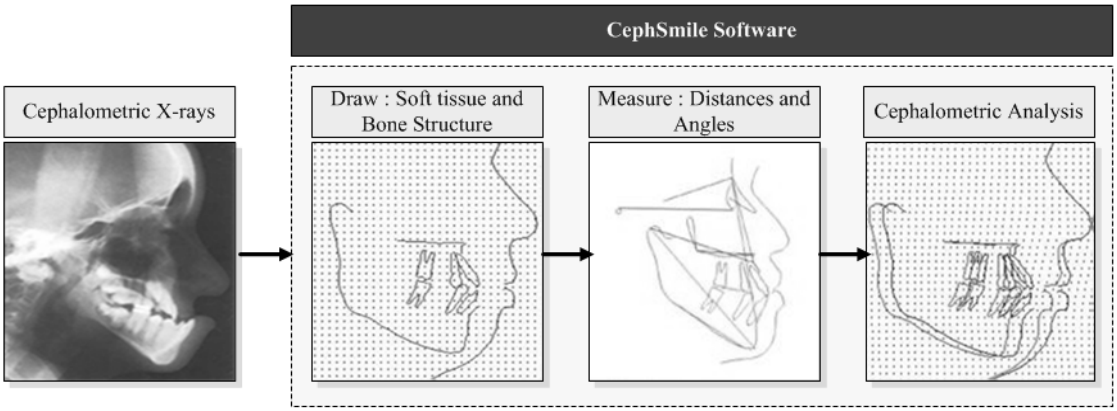
เกี่ยวกับสุขภาพช่องปากและบุคลิกภาพ เช่น ปัญหาฟันเก ยื่น ห่าง ซ้อน หรือฟันไม่สบกัน เป็นต้น การจัดฟันเป็นการแก้ไขการเรียงตัวของฟันที่ขึ้นไม่เป็นระเบียบให้กลับเข้าสู่ตำแหน่งที่ควรจะเป็น เพื่อให้การสบระหว่างฟันบนและฟันล่างมีความสัมพันธ์ที่ถูกต้องซึ่งจะเป็นผลดีต่อการบดเคี้ยวอาหาร รวมทั้งปรับรูปร่างลักษณะใบหน้าให้ดูดีขึ้นด้วย ซึ่งข้อมูลที่ทันตแพทย์ใช้วางแผนจัดฟัน คือ ภาพถ่ายรังสีกะโหลกศีรษะด้านข้าง แล้วทำการร่างเค้าโครงของใบหน้า กราม ฟัน พร้อมทั้งคำนวณระยะ และมุมต่าง ๆ ด้วยอุปกรณ์เครื่องเขียน เพื่อวิเคราะห์ความสัมพันธ์ของโครงสร้างใบหน้าและฟัน (Chanjira Sinthanayothin, 2005) ซึ่งเป็นวิธีที่ใช้เวลามากและไม่สะดวกต่อการจัดเก็บข้อมูล ตลอดจนไม่สามารถเปรียบเทียบข้อมูลของผู้ป่วยรายเดียวกันที่เข้ามาทำการจัดฟันในแต่ละช่วงระยะเวลาได้ การจัดฟันยังต้องอาศัยชุดซอฟต์แวร์และอุปกรณ์บางตัวในราคาสูงจากต่างประเทศเพื่อช่วยในการจัดฟันด้วย ดังแสดงในตารางที่ 1.1 แถมยังอาจไม่เหมาะสมกับการจัดฟันของคนไทยที่มีสรีระแตกต่างกัน

ตารางที่ 1.1 แสดงราคาของตัวอย่างชุดซอฟต์แวร์สำหรับช่วยงานทางทันตกรรม

บริษัท	ซอฟต์แวร์	ราคา ($)
Sirona Dental Systems, Inc	SIDEXIS XG	12,500.00
perfectByte	PerfectByte Ortho	6,000.00
Practice Works, Inc	Kodak Dental Systems	4,995.00
Dentrix	Dentrix Dental Systems	1,895.00
FYI technolies	Dr. Ceph	1,595.00
Patterson-Eagle Soft	Patterson-Eagle Soft	1,440.00

ตัวอย่างการพัฒนาซอฟต์แวร์เพื่อช่วยงานทันตกรรมจัดฟันในประเทศไทย คือ โปรแกรม CephSmile ซึ่งพัฒนาโดยศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติร่วมกับคณะทันตแพทย์ศาสตร์ มหาวิทยาลัยมหิดล CephSmile เป็นโปรแกรมคอมพิวเตอร์ที่ช่วยวาดรอยของรูปร่างกะโหลกศีรษะใบหน้าด้านข้าง ทั้งที่ส่วนที่เป็นโครงสร้างกระดูก และเนื้อเยื่อ (Soft tissue) จากจุดสำคัญในแผ่นภาพรังสีเซฟาโลเมตริก (Cephalometric X-ray) ที่กำหนดโดยทันตแพทย์ คำนวณค่าของมุมและระยะต่าง ๆ ที่เกิดขึ้น การวิเคราะห์ค่าเซฟาโลเมตริก (Cephalometric Analysis) มีความสำคัญและใช้วิเคราะห์โครงสร้างลักษณะใบหน้าด้านข้าง ดังแสดงในรูปที่ 1.2 เพื่อเป็นเครื่องมือช่วยให้ทราบถึงปัญหาความผิดปกติของโครงสร้างกระโหลกศีรษะใบหน้าและฟัน ด้วยเทคนิคทางด้านวิธีวิเคราะห์เชิงตัวเลข (Numerical method) ต่าง ๆ ได้ถูกนำมาประยุกต์ใช้ใน

ขั้นตอนวิธีการหาเส้นโค้งที่เหมาะสมระหว่างจุดที่กำหนด (Curve Fitting) เพื่อให้ได้เค้าโครงของใบหน้าจากจุดกำหนด (landmark) เพียงไม่กี่จุด ตลอดจนการคำนวณหาค่ามุม และระยะต่าง ๆ ตามหลักการทางตรีโกนมิติอย่างอัตโนมัติ และทำการเปรียบเทียบกับค่ามาตราฐานของคนไทยโดยเฉพาะ

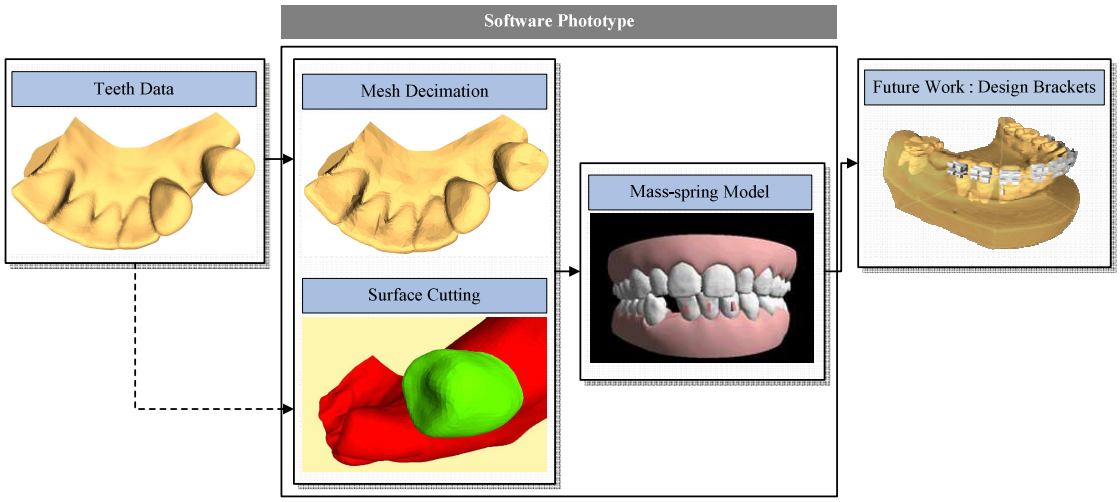


รูปที่ 1.2 แผนผังแสดงส่วนประกอบของโปรแกรม CephSmile

เทคนิคการประมวลผลภาพสามมิติถูกนำมาประยุกต์ใช้ เพื่อช่วยงานทางการแพทย์ ในด้านต่าง ๆ เช่น การจำลองการวางแผนและรักษาโรคทางตับ โดยใช้ในขั้นตอนของการฝึกฝนและการวางแผนการผ่าตัดตับ ซึ่งแบบจำลองของตับจะสามารถเปลี่ยนรูปร่างได้เมื่อมีการกระทำการผ่าตัด (Sojar et al., 2004) เช่นเดียวการจำลองการวางแผนการผ่าตัวหัวใจ โดยสร้างแบบจำลองของหัวใจในสามมิติ ซึ่งแบบจำลองของหัวใจนั้นจะมีการเปลี่ยนรูปร่างอยู่ตลอดเวลาเนื่องจากการบีบและคลายตัว เพื่อสูบฉีดโลหิตไปยังส่วนต่าง ๆ ของร่างกาย (Trunk et al., 2007) การจำลองการผ่าตัดปากแหว่งเพดานโหว่ (Clef LIPS) เป็นปัญหาการผิดปกติของกระดูกริมฝีปาก เพื่อวางแผนการผ่าตัดซึ่งต้องสร้างแบบจำลองของผิวหนังบนใบหน้ารวมถึงริมฝีปาก โดยการผ่าตัดจะแก้ไขริมฝีปากให้กลับเข้ารูปเป็นปกติ (Schendle et al., 2005) การแสดงผลภาพสามมิติจากเครื่องอัลตราซาวน์ เพื่อสังเกตลักษณะและพฤติกรรมของทารกภายในครรภ์ และตรวจหาสิ่งผิดปรกติที่เกิดกับทารก การสร้างภาพสามมิติของเต้านม เพื่อวิเคราะห์หาเซลล์มะเร็ง (Sakas, 2002) แต่สำหรับการประยุกต์เทคนิคการประมวลภาพสามมิติเพื่อใช้งานในด้านทันตกรรมยังมีไม่มากนัก

ดังนั้นงานวิจัยนี้จึงมุ่งที่จะศึกษาและพัฒนาขั้นตอนวิธีเพื่อช่วยงานทางทันตกรรมจัดฟัน โดยใช้คอมพิวเตอร์เข้ามาช่วยในขั้นตอนการวิเคราะห์และวางแผนรักษาทางทันตกรรมจัดฟัน ซึ่งจะช่วยลดเวลาการบำบัดรักษาและอำนวยความสะดวกให้กับทันตแพทย์ ช่วยให้การรักษามี

ประสิทธิภาพมากขึ้นซึ่งเป็นประโยชน์ต่อผู้เข้ารับการรักษา ส่วนประกอบของซอฟต์แวร์ต้นแบบแสดงดังรูปที่ 1.3 โดยการแสดงผลภาพในรูปแบบสามมิติ (3D Reconstruction) เพื่อช่วยให้มีความเสมือนจริงกับฟันของผู้เข้ารับการรักษา และสามารถจำลอง (Simulation) รูปแบบการจัดเรียงของฟันแต่ละระยะว่าจะเปลี่ยนไปอย่างไรระหว่างการบำบัดรักษา นอกจากนี้ข้อมูลที่ได้จากการวิเคราะห์สามารถนำไปช่วยในการออกแบบวัสดุจัดฟันได้



รูปที่ 1.3 แผนผังแสดงส่วนประกอบของซอฟต์แวร์ต้นแบบ

1.2 วัตถุประสงค์การวิจัย

งานวิจัยนี้มีวัตถุประสงค์หลัก คือ ศึกษาเทคนิคการประมวลผลภาพทางการแพทย์ 3 มิติสำหรับงานทันตกรรมจัดฟัน ซึ่งจะเน้นด้านจำลองการจัดฟัน และการเปลี่ยนแปลงรูปร่างของเหงือก โดยแบ่งเป็นวัตถุประสงค์ย่อยดังนี้

1) พัฒนาโปรแกรมต้นแบบสำหรับการวางแผนการจัดฟัน เพื่อช่วยให้การบำบัดและรักษาของแพทย์มีประสิทธิภาพเพิ่มขึ้น และเป็นประโยชน์ต่อผู้เข้ารับการรักษา
2) เพื่อศึกษาและพัฒนาขั้นตอนวิธีการลดทอนรายละเอียดของวัตถุ (Mesh decimation) การตัดพื้นผิวของวัตถุ (Surface cutting) และการจำลองการเปลี่ยนแปลงรูปร่างของแบบจำลองเชิงสปริง (Mass-spring model)
3) เพื่อเปรียบเทียบ และค้นหาการลดทอนรายละเอียดของแบบจำลองฟันที่เหมาะสมสำหรับการจำลองการจัดฟัน

4) เพื่อประสานความร่วมมือระหว่างหน่วยงานวิจัยหลัก เช่น สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ (สวทช.) และสถาบันการศึกษาต่าง ๆ ซึ่งงานวิจัยนี้ได้รับการสนับสนุนเงินทุนจากสำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติในโครงการทุนสถาบันบัณฑิตและวิทยาศาสตร์ไทย (TGIST)
5) สามารถนำผลลัพธ์ของงานวิจัยซึ่งอยู่ในรูปของขั้นตอนวิธี ผนวกร่วมกับระบบปฏิบัติการร่วม สำหรับการประมวลผลภาพของมหาวิทยาลัยเทคโนโลยีสุรนารี (SUT Imaging Platform) และใช้ในการพัฒนาซอฟต์แวร์ทางการแพทย์เพื่องานทันตกรรมจัดฟัน

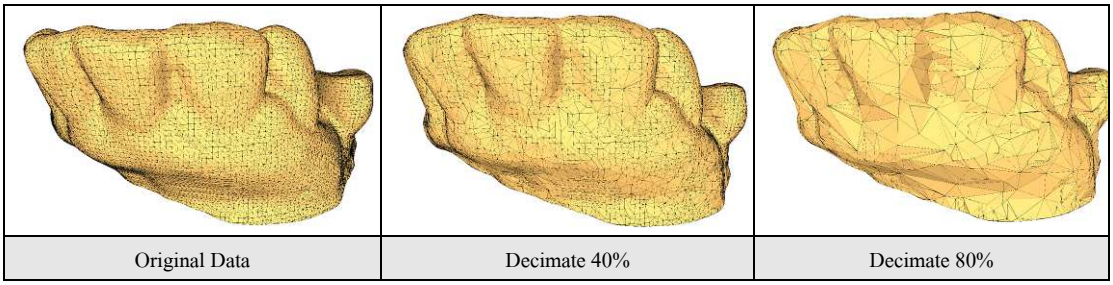
1.3 ขอบเขตการวิจัย

งานวิจัยนี้มีจุดมุ่งหมายที่จะพัฒนาแนวทางและขั้นตอนวิธีการประมวลผลภาพ 3 มิติเพื่อใช้งานในทางการแพทย์เป็นหลัก โดยเน้นการช่วยงานด้านทันตกรรมจัดฟัน ซึ่งอาจสรุปลักษณะสำคัญของเทคนิคการจำลองการจัดฟันสำหรับงานวิจัยนี้ได้ดังต่อไปนี้

1) สามารถบริหารจัดการแฟ้มข้อมูลภาพและแบบจำลองมาตรฐาน ซึ่งแฟ้มข้อมูลที่ใช้ในงานวิจัยนี้คือ แฟ้มข้อมูลแบบเอสทีแอลไบนารี (STL binary) ซึ่งเป็นแฟ้มข้อมูลของแบบจำลองฟันแบบ 3 มิติ ที่ได้มาจากเครื่องมือขึ้นรูปสามมิติ (Rapid Prototype, RP) (Lin, Ma, and Chua, 2001)
2) สามารถแสดงผลแบบจำลองฟันและการเปลี่ยนมุมมองในแบบ 3 มิติ เพื่อช่วยให้การวางแผน การจัดฟันสมจริงและสะดวกต่อการวิเคราะห์ การเปลี่ยนมุมมองรวมถึงการย่อ/ขยาย หมุน และเลื่อนแบบจำลองฟัน
3) สามารถจำลองการเลื่อนตำแหน่งของแบบจำลองฟันและการเปลี่ยนรูปของแบบจำลองเหงือก การจำลองการเลื่อนตำแหน่งของแบบจำลองฟัน เป็นการจัดเรียงฟันในแต่ละซี่เพื่อให้ฟันแต่ละซี่เรียงตัวในตำแหน่งที่เหมาะสม ช่วยให้ทันตแพทย์สามารถวางแผนการรักษาได้อย่างถูกต้อง เมื่อมีการเลื่อนตำแหน่งของฟัน จำเป็นต้องมีการจำลองการเปลี่ยนรูปของเหงือกด้วย ซึ่งเป็นการจำลองในขณะที่ฟันมีการเลื่อนตำแหน่ง เพื่อวิเคราะห์ว่าการเลื่อนตำแหน่งของฟันจะส่งผลกระทบต่อการเปลี่ยนรูปร่างของเหงือกอย่างไร และเพื่อให้การจำลองการรักษาทางทันตกรรมดูเสมือนจริงโดยอ้างอิงกับหลักการด้านกลศาสตร์คำนวณ (Computation Mechanics)

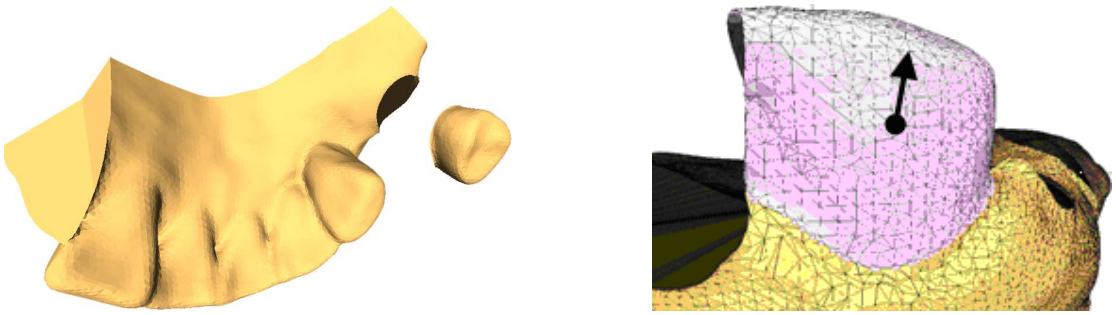
ในงานวิจัยนี้จะทำการศึกษาและพัฒนาขั้นตอนวิธีสำหรับการจำลองการจัดฟันใน 3 มิติ ในส่วนการแสดงผลใช้ไลบรารีและชุดคำสั่งของ OpenGL® ข้อมูลทดสอบ คือ ข้อมูลของฟัน ซึ่งเป็นแฟ้มข้อมูลแบบเอสทีแอลไบนารี โดยทั่วไปแล้วข้อมูลแบบจำลองฟันจะมีขนาดใหญ่และ

รายละเอียดสูง ดังแสดงในรูปที่ 1.4 ดังนั้นจำเป็นจะต้องลดรายละเอียดของข้อมูลเพื่อลดเวลาการแสดงผล โดยหลักเกณฑ์ที่ใช้ในการพิจารณาความเหมาะสมของการลดรายละเอียด ประกอบด้วย เวลาที่ใช้ในการแสดงผล เวลาประมวลผล และวิเคราะห์การจำลองการจัดฟันในขั้นตอนต่าง ๆ ขั้นตอนถัดไปคือ การแยกองค์ประกอบของฟันและเหงือกออกจากกัน เพื่อนำเหงือกไปสร้างแบบจำลองเชิงสปริงที่สามารถเปลี่ยนรูปร่างได้ ดังแสดงในรูปที่ 1.5(a) ซึ่งใช้ขั้นตอนวิธีการตัดพื้นผิวของวัตถุ ซึ่งเป็นการกำหนดจุดลงบนพื้นผิวของแบบจำลองฟัน เพื่อกำหนดขอบเขตการตัดพื้นผิว ขั้นตอนสุดท้ายคือ การเรียงตำแหน่งของแบบจำลองฟัน ซึ่งการเรียงตำแหน่งของฟันจะทำให้รูปร่างของแบบจำลองเหงือกเปลี่ยนไปด้วย ดังแสดงในรูปที่ 1.5(b) ขั้นตอนนี้ใช้ขั้นตอนวิธีการเปลี่ยนรูปของแบบจำลอง โดยเลือกใช้เทคนิคการเปลี่ยนรูปของแบบจำลองเชิงสปริง



Original Data	Decimate 40%	Decimate 80%

รูปที่ 1.4 แสดงผลการลดทอนรายละเอียดของแบบจำลองฟัน



(a) (b)

รูปที่ 1.5 แสดงผลการแยกองค์ประกอบของฟันและเหงือกออกจากกัน (a) แสดงผลการเปลี่ยนรูปร่างของเหงือกเมื่อฟันเลื่อนตำแหน่ง จะเห็นได้ว่าบริเวณเหงือกที่ติดกับฟันมีการเปลี่ยนรูปร่างตามทิศทางการเลื่อนของฟัน (b)

1.4 ประโยชน์ที่คาดว่าจะได้รับ

งานวิจัยนี้เป็นงานวิจัยสำหรับช่วยการบำบัดรักษาในการทำทันตกรรมจัดฟัน ผลลัพธ์ที่ได้อยู่ในรูปของขั้นตอนวิธี (Algorithm) และโปรแกรมต้นแบบ ที่จะนำไปพัฒนาต่อในการช่วยบำบัดรักษา ซึ่งเมื่อนำไปพัฒนาต่อสามารถจะเผยแพร่ไปสู่กลุ่มเป้าหมายอันได้แก่ ทันตแพทย์ ซึ่งประโยชน์ที่คาดว่าจะได้รับจากงานวิจัยคือ โปรแกรมที่ทำหน้าที่วิเคราะห์และประมวลผลภาพ 3 มิติ และช่วยในการวางแผนการจัดฟัน โดยการแสดงผลภาพแบบ 3 มิติและสามารถจำลองการจัดฟันได้ เพื่อให้การทำทันตกรรมจัดฟันมีประสิทธิภาพ ทั้งนี้ยังมีประโยชน์โดยตรงต่อผู้เข้ารับการรักษาเอง

นอกจากนี้ องค์ความรู้ที่ได้รับจากโครงการวิจัยนี้จะสามารถใช้เป็นพื้นฐานในการพัฒนาซอฟต์แวร์ในด้านการบำบัดรักษาระหว่างการทำทันตกรรมจัดฟันของแพทย์

โดยสรุป หน่วยงานที่สามารถนำผลการวิจัยไปใช้ประโยชน์ สามารถแจกแจงได้ดังต่อไปนี้

1) โรงพยาบาล สถาบัน หน่วยงานด้านสาธารณสุข และวิทยาศาสตร์สุขภาพ
2) มหาวิทยาลัย และสถาบันการศึกษา ที่มีการเปิดสอนวิชาสาขาแพทยศาสตร์
3) องค์กรที่ดำเนินการวิจัยด้านเทคโนโลยีการบำบัด และออกแบบเวชภัณฑ์รักษา
4) คลินิกทันตกรรม

1.5 โครงสร้างของวิทยานิพนธ์

รายละเอียดของงานวิจัยในส่วนที่เหลือจะกล่าวถึงในบทถัดไป ดังนี้ บทที่ 2 จะเป็นการกล่าวถึงวรรณกรรมและรายละเอียดของงานวิจัยที่เกี่ยวข้องกับงานวิจัยนี้ โดยจะกล่าวถึงเทคนิควิธีการจำลองการรักษาทางทันตกรรม ขั้นตอนวิธีการลดทอนรายละเอียดของวัตถุ ขั้นตอนวิธีการตัดพื้นผิว และขั้นตอนวิธีการสร้างแบบจำลองเชิงสปริง ในบทที่ 3 กล่าวถึงรายละเอียดและขั้นตอนการวิจัย ประกอบด้วย ขั้นตอนวิธีการลดทอนรายละเอียดของวัตถุ ขั้นตอนวิธีการตัดพื้นผิว และขั้นตอนวิธีการสร้างแบบจำลองเชิงสปริง สำหรับบทที่ 4 เป็นผลการทดสอบที่ได้จากงานวิจัยนี้ โดยจะแสดงผลการทดสอบที่ได้จากขั้นตอนวิธีต่าง ๆ ในรูปแบบของรูปภาพ ตารางหรือกราฟ ส่วนบทที่ 5 เป็นการสรุปและอภิปรายผลการทดลอง ปัญหาและอุปสรรคที่พบ และข้อเสนอแนะในการวิจัยต่อไป

บทที่ 2
ปริทัศน์วรรณกรรมและงานวิจัยที่เกี่ยวข้อง

ความก้าวหน้าทางด้านเทคโนโลยีสารสนเทศ และระบบคอมพิวเตอร์ที่มีประสิทธิภาพสูงขึ้น ทำให้มีการนำคอมพิวเตอร์มาช่วยในด้านการบำบัดรักษาทางการแพทย์ เพื่ออำนวยความสะดวกและเพิ่มประสิทธิภาพในการบำบัดรักษาของแพทย์ ตัวอย่างการนำคอมพิวเตอร์มาช่วยในด้านการบำบัดรักษาทางการแพทย์ คือ การนำคอมพิวเตอร์มาใช้ในการประมวลผลภาพในทางการแพทย์เพื่อช่วยในการบำบัดรักษา การประมวลผลภาพทางในการแพทย์เป็นการประยุกต์การประมวลผลภาพมาใช้ กระบวนการและเทคนิควิธีในการประมวลผลภาพนั้นมีจำนวนมาก เนื้อหาในบทนี้จะอธิบายตัวอย่างการนำเทคนิควิธีที่ใช้ในการประมวลผลภาพ เพื่อประยุกต์ใช้ในด้านการบำบัดรักษาของแพทย์ โดยในหัวข้อที่ 2.1 จะกล่าวถึงเทคนิคและกระบวนการจำลองการจัดฟัน อันเป็นการให้ข้อมูลเบื้องต้นของงานจำลองการจัดฟัน หัวข้อที่ 2.2 จะกล่าวถึงเทคนิคและขั้นตอนวิธีการลดทอนรายละเอียดของวัตถุ (Mesh decimation) ซึ่งเป็นขั้นตอนในการเตรียมข้อมูลให้มีขนาดที่เหมาะสมและเพื่อเป็นการเพิ่มประสิทธิภาพสำหรับงานจำลองการจัดฟัน หัวข้อที่ 2.3 เป็นการกล่าวถึงเทคนิคและขั้นตอนวิธีของการตัดพื้นผิวของวัตถุ (Surface cutting) ที่ใช้ เพื่อใช้ในการแยกองค์ประกอบของแบบจำลองฟันและเหงือกออกจากกัน หัวข้อที่ 2.4 กล่าวถึงเทคนิคและขั้นตอนวิธีการสร้างแบบจำลองเชิงสปริง (Mass-spring Model) เพื่อวิเคราะห์การเปลี่ยนแปลงรูปร่างของเหงือก ซึ่งเป็นผลมาจากการเลื่อนตำแหน่งของแบบจำลองฟันในกระบวนการจัดเรียงฟัน และหัวข้อที่ 2.5 กล่าวสรุปงานวิจัยที่เกี่ยวข้องกับการปรับแก้และเปลี่ยนรูปของแบบจำลองฟันใน 3 มิติ เพื่อช่วยงานทันตกรรม

2.1 กระบวนการจำลองการจัดฟัน (Orthodontic Simulation)

เทคโนโลยีสารสนเทศ และระบบคอมพิวเตอร์ ถูกนำมาประยุกต์ใช้เพื่อช่วยงานทางการแพทย์ในด้านต่าง ๆ มากมาย เช่น ด้านการตรวจและวินิจฉัย ด้านการวางแผนการบำบัดรักษา เป็นต้น การวางแผนการบำบัดรักษา จะรวมถึงการจำลองการรักษา กระบวนการจำลองการจัดฟันได้ประยุกต์ใช้เทคนิควิธีการประมวลผลภาพ งานวิจัยที่เกี่ยวข้องกับกระบวนการจำลองการจัดฟันได้แก่

Spencer et al. (1987) อธิบายถึงอุปกรณ์ขากรรไกรจำลอง (articulator) สำหรับจำลองรูปแบบการเคลื่อนที่ของขากรรไกร เพื่อสังเกตและศึกษาผลกระทบที่เกิดจากเปลี่ยนแปลงบนสันฐานวิทยา (morphology) ของฟัน ความซับซ้อนและความแม่นยำของกลอุปกรณ์ขากรรไกร

จำลองเป็นผลมาจากอุปกรณ์บานพับอย่างง่ายที่สามารถปรับมุมซึ่งทำให้เกิดการเคลื่อนที่แบบเลื่อนที่ในระบบสามมิติ ขากรรไกรสามารถพับเป็นรูปตัวยูได้ด้วยข้อต่อยึด (condyle) ซึ่งถูกยึดไว้กับขากรรไกรบนและล่าง การเคลื่อนที่ของข้อต่อยึดช่วยในการวิเคราะห์โครงสร้างกระดูกของร่องฟัน (fossa) เอ็นยึดของปลอกหุ้ม (the ligament of the capsule) เส้นประสาท (the nerves) และความสัมพันธ์ของกล้ามเนื้อกับข้อต่อ ขากรรไกรนั้นถูกควบคุมด้วยการเคลื่อนที่ของข้อต่อยึดแต่ละตัว การเคลื่อนที่นี้อาจจะถูกปรับด้วยการเรียงตัวของฟันหน้า เพื่อสร้างอุปกรณ์จำลองทางทันตกรรม เช่น ตัวฟัน (crown) สะพานฟัน (bridge) หรือฟันปลอม ทิศทางการเคลื่อนที่ที่ฟันบนและฟันล่างสัมพันธ์กัน ทิศทางการเคลื่อนที่นี้คือ เส้นโค้งในระบบสามมิติ อุปกรณ์ขากรรไกรจำลองจะจำลองทิศทางการเคลื่อนที่แบบเส้นโค้งดังกล่าว ผลลัพธ์จะแสดงผลออกทางจอภาพซึ่งเป็นค่าการวัดต่าง ๆ

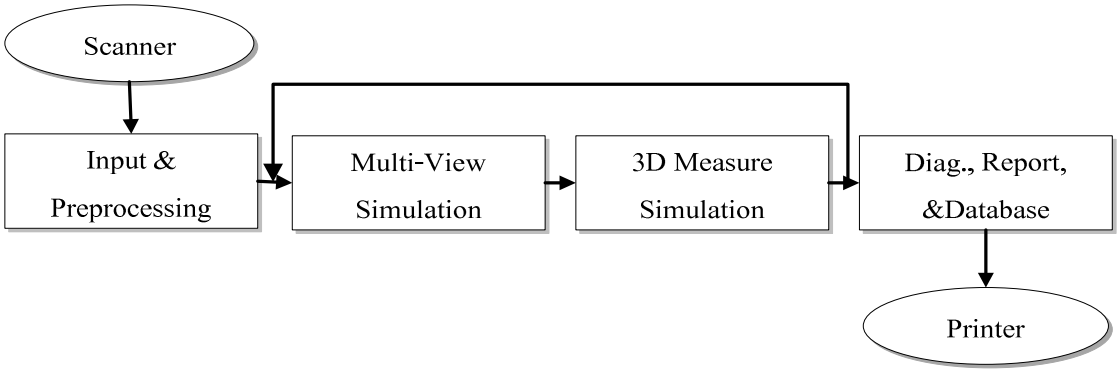
Mah and Bumann (2001) ได้ทำการศึกษาเทคโนโลยีการสร้างข้อมูลการรักษาของผู้ป่วยทางทันตกรรมในรูปแบบของข้อมูลสามมิติ โดยทั่วไปแล้วข้อมูลจะอยู่ในรูปของรูปถ่าย ภาพรังสีเอกซเรย์ และข้อมูลอื่น ๆ ซึ่งต้องจัดการด้วยตนเอง เกิดปัญหาความยุ่งยากในกระบวนการปรับปรุงข้อมูลให้เป็นปัจจุบัน ปัจจุบันเทคนิคด้านการประมวลผลภาพได้นำมาประยุกต์ใช้ในงานทันตกรรม ซึ่งรวมถึงภาพสามมิติของใบหน้า กระดูก ข้อมูลตำแหน่ง และการเคลื่อนที่ของฟันในสามมิติ เทคโนโลยีที่ถูกนำมาใช้ในการสร้างข้อมูลสามมิติ เริ่มต้นจากการใช้เครื่องเอกซเรย์คอมพิวเตอร์ (Computed Tomography, CT) สร้างภาพสามมิติของใบหน้าและฟัน ด้วยข้อจำกัดด้านความถูกต้องด้านทรัพยากรของเครื่อง ทำให้เกิดความผิดพลาดด้านตำแหน่ง เพื่อแก้ไขปัญหาดังกล่าวได้ใช้เทคนิคการสร้างภาพสามมิติของใบหน้าและฟันจากภาพรังสีเอกซเรย์ศีรษะด้านข้าง (cephalograms) เทคนิควิธีการสร้างข้อมูลการบำบัดรักษาของผู้ป่วยแบบสามมิติ เพื่อช่วยงานทันตกรรม แสดงได้ดังตารางที่ 2.1

ตารางที่ 2.1 วิธีการสร้างข้อมูลการบำบัดรักษาของผู้ป่วยแบบสามมิติ (Mah and Bumann, 2001)

กระบวนการ	วิธีการที่ใช้อยู่ในปัจจุบัน	เทคโนโลยีใหม่ (ผู้ผลิต/ตัวแทนจำหน่าย)	วัตถุประสงค์
Imaging the craniofacial skeleton	Cephalograms	Anatomic Reconstruction (Acuscape) Cone Beam 3D CT (ZeroBase-USA)	3D skeletal images
Recording the dentition	Impressions and Plaster models	3D dental scanner (OraMetrix Intl)	3D model of the dentition
Imaging the face	Photography	3D image acquisition (Eyetronics Corp, Leuven)	3D facial images and analysis
Recording mandibular relation and motion	Wax bite, Articulator	3D ultrasonic motion (Zebris Medizintechink GmbH, Tubingen)	3D mandibular relation and motion

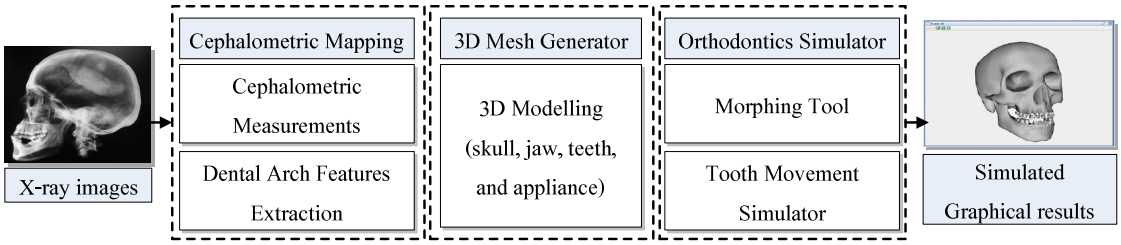
Vlaskalic and Boyd (2001) ได้นำเสนอเทคนิคใหม่ในการบำบัดรักษาทางทันตกรรมจัดฟัน ซึ่งได้ร่วมมือกับบริษัทอะไลน์เทคโนโลยี (Align Technology Inc.) โดยที่ใช้อุปกรณ์จัดฟันแบบใหม่ที่มีลักษณะโปร่งใส สามารถถอดออกง่ายและสะดวกในการสวมใส่ เรียกว่า “Aligner” การรักษาแบบเสมือนจริงในแต่ละระยะของการจัดฟันได้ใช้ซอฟต์แวร์ไคลน์เช็ค (ClinkCheck™) และติดต่อสื่อสารกับผู้เชี่ยวชาญผ่านระบบอินเทอร์เน็ต ในแต่ละช่วงของการรักษาจะถูกประเมินโดยผู้เชี่ยวชาญและคอมพิวเตอร์จะช่วยจำลองการเลื่อนตำแหน่งของฟันเพื่อใช้ในการสร้างวัตถุจัดฟันแบบใส แต่ละชิ้นของวัตถุจัดฟันแบบใสจะใช้ได้ประมาณ 7-10 วัน ความสามารถของซอฟต์แวร์ของ Invisalign™ สามารถใช้เพื่อออกแบบวัตถุจัดฟันซึ่งสามารถจัดฟันได้อย่างถูกต้องด้วยระบบภาพคอมพิวเตอร์สามมิติ

Li et al. (2005) ศึกษาและพัฒนาระบบคอมพิวเตอร์ช่วยจำลองการจัดฟันและวินิจฉัยสำหรับทันตแพทย์ โครงสร้างของระบบ แสดงดังรูปที่ 2.1 เริ่มต้นจากสร้างโมเดลสามมิติจากเครื่องสแกนเนอร์ เครื่องสแกนเนอร์สามมิติสามารถสร้างโครงข่ายได้หลายรูปแบบไฟล์ ระบบสามารถรองรับไฟล์ได้ 3 รูปแบบ ได้แก่ 3DS (3D Studio Mesh), STL (Stereo lithography format), และ OBJ (Wavefront object format) ด้วยเทคโนโลยีของเครื่องสแกนเนอร์สามมิติทำให้โครงข่ายที่ได้มีความละเอียดสูงมาก เป็นปัญหาสำหรับการแสดงผล เพื่อช่วยให้การแสดงผลเร็วขึ้น จำเป็นต้องลดรายละเอียดของโครงข่าย ซึ่งใช้เทคนิควิธีการจัดกลุ่มของจุดบนโครงข่าย (Vertex clustering) จากนั้นนำข้อมูลฟันดังกล่าวเข้าสู่กระบวนการจำลองการจัดฟัน ซึ่งการแสดงผลเป็นแบบสามมิติและสามารถแสดงผลได้หลายรูปแบบ ทันตแพทย์วัด และคำนวณค่าต่าง ๆ ในสามมิติ การแสดงผลบนหน้าต่างโปรแกรมใช้ไลบารีของ OpenGL® (Kempf et al., 1999) การวัดและคำนวณค่าต่าง ๆ ระบบได้เตรียมเครื่องมือวัดหลายประเภท ได้แก่ เครื่องมือตรวจสอบการสบกันของฟันบนและฟันล่าง เครื่องมือวัดขนาดของฟันแต่ละซี่ โดยวัดจากระนาบขนานและระนาบฉาย เครื่องมือวัดค่าความโค้งของฟัน เครื่องมือสำหรับแยกฟันแต่ละซี่ โดยใช้เทคนิคการแยกด้วยค่าความโค้ง (curvature) และวอเตอร์เฉด (watershed) หรือแนวกั้น และเครื่องมือจัดเรียงฟัน ขั้นตอนสุดท้ายคือ การวิเคราะห์และวินิจฉัย ด้วยเครื่องมือช่วยแสดงผลลัพธ์ที่ถูกต้องสำหรับการจัดฟัน หลังจากทันตแพทย์กระทำการวัดและคำนวณค่าความโค้ง คอมพิวเตอร์สามารถจัดเรียงฟันให้สอดคล้องกับข้อมูลทางสถิติ ซึ่งสามารถพิมพ์รายงานการวินิจฉัยได้



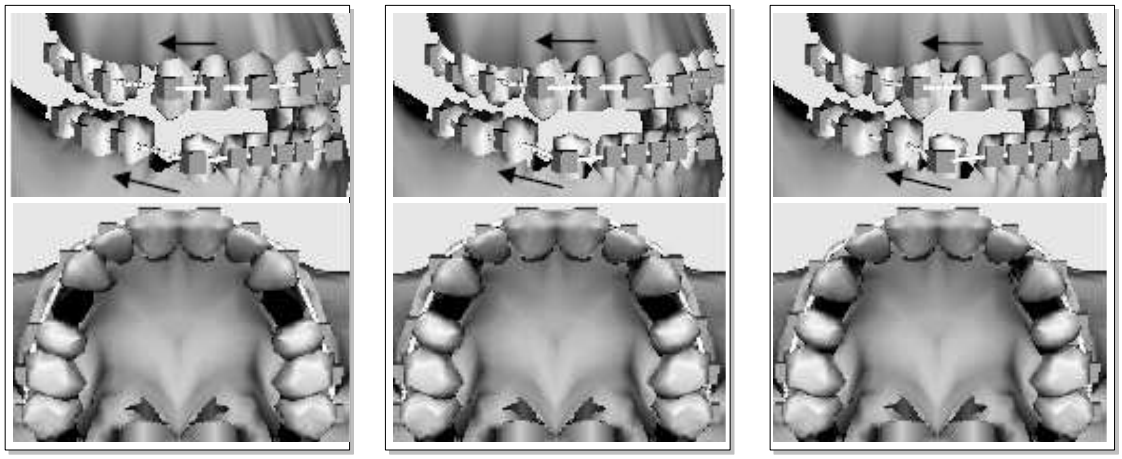
รูปที่ 2.1 โครงสร้างของระบบคอมพิวเตอร์ช่วยจำลองการจัดฟัน (Li et al., 2005)

Rodrigues et al. (2006) ได้พัฒนาซอฟต์แวร์โอเพนซอร์สจำลองการรักษาทางทันตกรรมจัดฟัน ชื่อว่า “J-Ortho” ซึ่งพัฒนาด้วยภาษาจาวา และประกอบด้วย 3 ส่วนโปรแกรมพื้นฐาน ได้แก่ กระบวนการจัดภาพถ่ายรังสีเอกซเรย์ของกระโหลกศีรษะด้านข้าง กระบวนการสร้างโครงข่ายสามมิติ และกระบวนการจำลองการรักษาทางทันตกรรม ดังรูปที่ 2.2 กระบวนการจัดการกับภาพถ่ายรังสีเอกซเรย์ของกระโหลกศีรษะด้านข้างเป็นการตอบสนองต่อการจัดการภาพเอกซเรย์ และการวัดค่าของแบบจำลองฟันไปยังจุดบนโครงสร้างแบบจำลองฟันและขากรรไกรล่าง เพื่อใช้ในกระบวนการสร้างโครงข่ายสามมิติ เครื่องมือจำลองการเลื่อนของฟัน และเครื่องมือหลอมภาพ (Morphing Tool) เป็นมอดูล (module) ของกระบวนการจำลองการรักษาทางทันตกรรม ระบบช่วยจำลองการจัดฟันนี้ช่วยให้ผู้เชี่ยวชาญจำลอง และมองเห็นการจัดฟันในแต่ละระยะของการบำบัดรักษา การได้มาของข้อมูลและการสร้างแบบจำลองสามมิติ ระบบสามารถสร้างแบบจำลองมาตรฐานต่าง ๆ ได้แก่ กระโหลก ขากรรไกรบน ขากรรไกรล่าง และฟัน การสร้างแบบจำลองฟันนั้นจะประกอบด้วยตัวฟันและรากฟัน ซึ่งใช้เทคนิคเส้นโค้งสไปลน์ (spline curve) แบบจำลองสามมิติสร้างจากภาพเอกซเรย์มีรายละเอียดไม่สูงมาก ทำให้ใช้เวลาสำหรับการแสดงผลน้อย



รูปที่ 2.2 ส่วนโปรแกรมพื้นฐานของระบบจำลองการจัดฟัน

ส่วนประกอบในกระบวนการจำลองการจัดฟัน จะรวมถึงเครื่องมือการหลอมภาพ (Morphing Tool) ซึ่งพัฒนาจากขั้นตอนวิธีของเบียร์ (Beier's Algorithms) (Beier and Neely, 1992) ซึ่งเปลี่ยนรูปร่างเรขาคณิตของปากบนภาพที่จัดเก็บไว้ในระหว่างตรวจวินิจฉัยของผู้เชี่ยวชาญ การหลอมภาพ (Morphing) เป็นกระบวนการเปลี่ยนรูปของวัตถุในแต่ล่ะช่วงเวลา และการรวมตัวกันของวัตถุสองวัตถุ กระบวนการทำงานคือ เริ่มต้นกำหนดภาพก่อนการรักษาและภาพหลังการรักษา จากนั้นจะสร้างภาพในแต่ละช่วงเวลาจากภาพก่อนการรักษาเพื่อนำไปสู่ภาพหลังการรักษา ทำให้สามารถวิเคราะห์การเปลี่ยนแปลงที่เกิดขึ้นในแต่ละช่วงเวลาได้ การจำลองการเคลื่อนที่ของแบบจำลองฟัน สามารถเลื่อน และปรับหมุนแบบจำลองฟันเพื่อให้ฟันอยู่ในตำแหน่งที่ถูกต้องได้ ดังแสดงในรูปที่ 2.3 ระบบได้สร้างวัสดุจัดฟัน (Bracket) มาให้ด้วย



รูปที่ 2.3 แสดงผลของการจำลองการเคลื่อนของฟัน (Rodrigues et al., 2006)

กระบวนการจำลองการจัดฟันจากงานวิจัยที่ได้กล่าวมาแล้วนั้น สามารถสรุปได้ว่า การจำลองการจัดฟันต้องสามารถจัดเรียงฟัน ด้วยการหมุนหรือการเลื่อนตำแหน่ง ซึ่งกระบวนที่จำเป็นคือการแยกองค์ประกอบของฟันออกจากเหงือก และในกระบวนการจัดเรียงฟันต้องสามารถจำลองการเปลี่ยนรูปร่างของเหงือกได้อีกด้วย เทคนิควิธีด้านการประมวลผลภาพที่ศึกษาต่อไปคือ การลดทอนรายละเอียดของวัตถุ (Mesh decimation) การตัดพื้นผิว (Surface cutting) ใช้สำหรับแยกองค์ประกอบของฟันออกจากเหงือก และการสร้างแบบจำลองมวลสปริง (Mass-spring model) ใช้สำหรับจำลองการเปลี่ยนรูปร่างของเหงือก

2.2 การลดทอนรายละเอียดของวัตถุ (Mesh Decimation)

การลดทอนรายละเอียดของวัตถุเป็นงานที่มีความสำคัญอย่างมากในกระบวนการสร้างภาพกราฟฟิก 3 มิติ เนื่องจากแบบจำลองสามมิติที่ได้มีรายละเอียดสูง เป็นผลให้เกิดการสิ้นเปลืองเนื้อที่จัดเก็บและเวลาในการประมวลผลจำนวนมาก อีกทั้งแต่ละเทคนิคยังมีข้อจำกัดที่แตกต่างกันออกไป ดังนั้นจึงมีผู้สนใจศึกษาและพัฒนาเทคนิคการลดทอนรายละเอียดของวัตถุอย่างกว้างขวาง

Schroeder, Zarge, and Lorensen (1992) ได้นำเสนอแนวคิดและขั้นตอนวิธีการลดทอนรายละเอียดของวัตถุ งานวิจัยนี้ได้นิยามการลดทอนรายละเอียดของวัตถุว่า “เป็นกระบวนการลดจำนวนสามเหลี่ยมของโครงข่ายแบบสามเหลี่ยม (Triangle mesh) และยังรักษาลักษณะที่สำคัญของวัตถุนั้นไว้ด้วย”

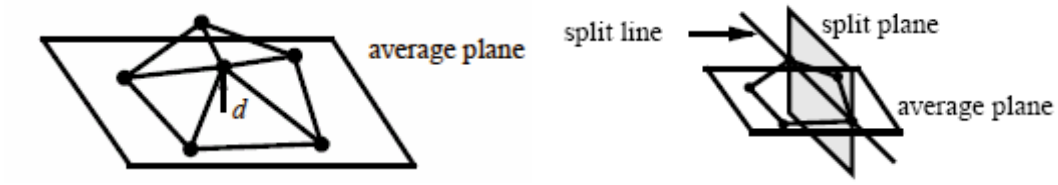
กระบวนการลดทอนรายละเอียดของวัตถุ แบ่งออกได้เป็น 3 ขั้นตอน คือ

1) กระบวนการจำแนกจุดบนโครงข่าย เพื่อพิจารณาว่าจุดประเภทใดที่สามารถดึงออกจากโครงข่ายได้
2) กระบวนการพิจารณาจุดที่เหมาะสมสำหรับการลดทอนรายละเอียด แล้วดึงจุดนั้นออกจากโครงข่าย
3) กระบวนการสร้างสามเหลี่ยมใหม่จากช่องว่างที่เกิดจากการดึงจุดออกจากโครงข่าย

ประเภทของจุดบนโครงข่าย สามารถจำแนกได้ 5 ประเภท ได้แก่ Simple, Complex, Boundary, Interior Edge และ Corner ดังแสดงในรูปที่ 2.4 *Simple vertex* คือ จุดที่ใช้สามเหลี่ยมสองรูปร่วมกันกับจุดข้างเคียง (Neighbor vertex) โดยที่จุดนั้นต้องไม่ใช่จุดที่อยู่บนเส้นขอบ (Boundary vertex) หรือมุม (Corner vertex) สำหรับ *Complex vertex* จะมีจุดข้างเคียงที่ไม่ได้ใช้สามเหลี่ยมสองรูปร่วมกัน และจุดนั้นต้องไม่ใช่จุดที่อยู่บนเส้นขอบด้วย ส่วน *Boundary vertex* เป็นจุดที่อยู่บนเส้นขอบของโครงข่าย ซึ่งจะมีจุดข้างเคียงที่ไม่ได้ใช้สามเหลี่ยมสองรูปร่วมกัน *Interior edge vertex* คือ จุดที่มีด้าน (edge) สองด้านที่สามเหลี่ยมประกอบของด้านนั้นทำมุมแหลมต่อกัน ส่วน *Corner vertex* คือ จุดที่มีด้าน (edge) สามด้านที่สามเหลี่ยมประกอบของด้านนั้นทำมุมแหลมต่อกัน ซึ่งประเภทของจุดที่ไม่สามารถดึงออกจากโครงข่ายได้ คือ *Complex vertex*

เกณฑ์การพิจารณาเลือกจุดที่สามารถดึงออกจากโครงข่ายได้ พิจารณาจากจุดที่ไม่ใช่จุดประเภท *Complex vertex* และพิจารณาระยะห่างระหว่างจุดนั้นกับระนาบเฉลี่ย ดังแสดงในรูปที่ 2.5(a) ถ้าระยะห่างน้อยกว่าค่าผิดพลาด จุดนั้นสามารถดึงออกจากโครงข่ายได้ เมื่อจุดถูกดึงออกจากโครงข่ายจะทำให้เกิดช่องว่างขึ้น จำเป็นต้องสร้างสามเหลี่ยมใหม่ โดยใช้วิธีการแบ่งของระนาบ (split plan) ซึ่งจะแบ่งและสร้างสามเหลี่ยมภายในช่องว่างไปจนกว่าช่องว่างนั้นจะประกอบไปด้วยสามเหลี่ยมทั้งหมด ดังแสดงในรูปที่ 2.5(b)

รูปที่ 2.4 ประเภทของจุดบนโครงข่าย (Schroeder et al., 1992)



(a) (b)

รูปที่ 2.5 ระยะทางจากจุดกับระนาบเฉลี่ย (a) การสร้างสามเหลี่ยมใหม่
ด้วยวิธีการแบ่งระนาบ (b) (Schroeder et al., 1992)

Hoppe et al. (1993) ได้เสนอแนวคิดในการทำการลดทอนรายละเอียดของวัตถุ โดยใช้ฟังก์ชันพลังงานเข้ามาช่วย เพื่อให้โครงข่ายหรือวัตถุนั้นมีจำนวนจุดน้อยและมีรูปร่างลักษณะคล้ายเคียงกับโครงข่ายหรือวัตถุเดิม โดยฟังก์ชันพลังงานมีสูตรดังสมการที่ 2.1 โดยที่ V คือ ตำแหน่งของจุดบนวัตถุ และ K คือ ความสัมพันธ์ระหว่างจุด (vertices) หรือด้าน (edges) และสามเหลี่ยม (facets) ในพจน์แรกของฟังก์ชันพลังงานเป็นค่าพลังงานระยะทาง (E_{dist}) ซึ่งนิยามด้วย ค่าผลรวมระยะทางจากจุด $X = x_1, \ldots, x_n$ กับโครงข่าย โดยค่าพลังงานนี้จะเพิ่มขึ้นเมื่อมีการเลื่อนตำแหน่งห่างออกไปจากตำแหน่งเดิม ซึ่งหมายความว่า ขั้นตอนวิธีนี้จะรักษารูปร่างลักษณะเดิมของวัตถุนั้นไว้ สำหรับพจน์ที่สอง คือ พลังงานตัวแทนของโครงข่าย (Representation Energy, E_{rep}) ซึ่งค่านี้จะส่งผลกระทบต่อการลดทอนรายละเอียดของวัตถุด้วย พจน์ที่สามของสมการพลังงานคือ E_{spring} เป็นค่าพลังงานสปริงของโครงข่าย ซึ่งมีค่าเริ่มต้นเท่ากับศูนย์ ค่าคงที่ของสปริง (k) ควรมีค่าน้อย ๆ ในทุกรอบของขั้นตอนการลดรายละเอียดค่าคงที่นี้จะลดลง

$$E(K,V) = E_{dist}(K,V) + E_{rep}(K) + E_{spring}(K,V) \tag{2.1}$$

การปรับค่าพลังงานนี้สามารถแก้ปัญหาสองข้อคือ ปัญหาการลดค่าพลังงานตัวแทนโดยการดึงหรือแทนที่จุด และปัญหาการลดพลังงานระยะทาง จุดที่ถูกดึงออก เพิ่ม หรือแทนที่สามารถเปลี่ยนรูปได้ 3 รูปแบบ คือ ยุบเส้นขอบ (edge collapse) แบ่งเส้นขอบ (edge split) และสับเปลี่ยนเส้นขอบ (edge swap) สังเกตว่าการแบ่งเส้นขอบและการสับเปลี่ยนเส้นขอบไม่ทำให้ค่าพลังงานตัวแทนลดลง แต่อาจจะทำให้ค่าพลังงานระยะทางหรือพลังงานสปริงลดลง

Cignoni, Rocchhini, and Scopigno (1998) ได้สร้างโปรแกรม Metro ซึ่งเป็นเครื่องมือเปรียบเทียบรายละเอียดของโครงข่ายแต่ละระดับโดยการคำนวณปริมาตรของโครงข่าย คณะวิจัยได้กล่าวถึงความจำเป็นสำหรับการใช้เครื่องมือนี้ว่า “งานวิจัยที่เกี่ยวข้องกับการลดทอนรายละเอียดของวัตถุยังคงขาดสูตรและความรู้ในการนิยามค่าความผิดพลาด” ดังนั้นงานวิจัยนี้จึงนำเสนอเครื่องมือการทดสอบหาค่าความผิดพลาดในการลดทอนรายละเอียดของวัตถุ Metro เป็นเครื่องมือที่จำเป็นสำหรับผู้ที่ทำวิจัยเกี่ยวกับการลดทอนรายละเอียดของวัตถุ เพราะสามารถแสดงผลลัพธ์เชิงตัวเลขเพื่ออธิบายความแตกต่างระหว่างโครงข่ายดังเดิมกับโครงข่ายที่ถูกลดทอนรายละเอียด การประมาณค่าระยะห่างระหว่างโครงข่ายสามารถคำนวณได้จากสมการที่ 2.2 การหาค่าระยะทางโดยประมาณใช้เทคนิควิธีการตรวจสอบความเปลี่ยนแปลง (scan conversion) ของสามเหลี่ยมภายในโครงข่ายหรือสุ่มการกระจายระหว่างพื้นผิวเพื่อคำนวณหาค่าเฉลี่ยด้วยวิธีอินทิเกรตของมอนตริคาร์โล (Monte Carlo integration) โปรแกรมจะให้ผลลัพธ์สองค่า ค่าแรกคือ ค่าระยะทางมากที่สุด E ซึ่งค่านี้คำนวณได้ระยะทางระหว่างจุดและพื้นผิวโดยประมาณ ส่วนค่าที่สองคือ ระยะทางเฉลี่ย (E_m) ซึ่งคำนวณได้จากการอินทิเกรตพื้นผิวของระยะทางต่อพื้นที่ของพื้นผิวเดิม โปรแกรม Metro ใช้ค่าระยะทางโดยประมาณเพื่อคำนวณค่าที่สำคัญมากมายด้วยค่าเชิงตัวเลขเพื่อแสดงความแตกต่างระหว่างโครงข่ายที่ถูกลดทอนรายละเอียดกับโครงข่ายเดิม

$$e(p,S) = \min_{p' \in S} d(p,p') \tag{2.2}$$

เมื่อ p คือ จุดบนโครงข่าย

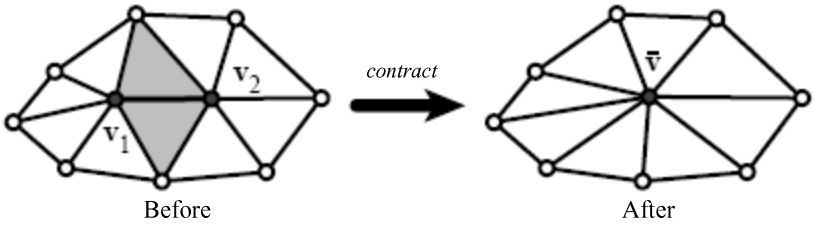
S คือ พื้นผิวของโครงข่าย

d คือ ระยะห่างระหว่างจุด

Lindstrom and Turk (1998, 1999) ได้คำนวณค่าผิดพลาดสำหรับทุกจุดเคียงข้าง และการยุบเส้นขอบด้วยค่าผิดพลาดที่น้อยที่สุดในจุดนั้น เพื่อรักษาลักษณะที่สำคัญของปริมาตรและเส้นขอบ อีกทั้งยังมีกระบวนการปรับปรุงปริมาตร เส้นขอบ และรูปร่างสามเหลี่ยมภายในโครงข่าย กระบวนทั้งหมดนี้ทำเพื่อลดค่าความผิดพลาดทางเรขาคณิตเฉลี่ย งานวิจัยนี้ได้เลือกใช้เทคนิคการยุบเส้นขอบ (edge collapse) สำหรับการลดทอนรายละเอียดของวัตถุ เนื่องจากเทคนิคการยุบเส้นขอบถูกนำมาใช้อย่างมากเพื่อแทนที่เทคนิคการลบจุด (vertex removal) ด้วยเหตุที่ว่ากระบวนการยุบเส้นขอบนั้นสามารถรักษาลักษณะทางเรขาคณิตของวัตถุได้ดีกว่า และไม่จำเป็นต้องสร้างสามเหลี่ยมใหม่จากช่องว่างที่เกิดขึ้นจากการลบจุดออก งานวิจัยนี้แสดงวิธีหาจำนวนจุดที่มีค่าผิดพลาดสำหรับการลดทอนรายละเอียดของวัตถุ โดยใช้โปรแกรม Metro ประมาณค่าความผิดพลาดทางเรขาคณิตเฉลี่ยด้วยระดับของรายละเอียดของวัตถุสามมิติที่แตกต่างกัน อีกทั้งยังทดสอบการสิ้นเปลืองของหน่วยความจำ จากการทดสอบสามารถสรุปได้ว่า การไม่สิ้นเปลืองหน่วยความจำในกระบวนการยุบเส้นขอบช่วยปรับปรุงประสิทธิภาพในการลดทอนรายละเอียดของขั้นตอนวิธีอื่น ๆ ได้

Godi and Manocha (1998) เริ่มต้นศึกษาค้นคว้าเทคนิควิธีการสร้างแบบจำลองของวัตถุอย่างง่ายด้วยบีสไปลน์ (B-spline) งานวิจัยประกอบด้วย 3 ส่วนด้วยกัน คือ กระบวนการสร้างตัวแทนของวัตถุแบบซีจากโครงข่ายเริ่มต้น ซึ่งประกอบด้วยแผ่นสไปลน์แบบสามเหลี่ยม (triangular spline patch) กระบวนการรวมเส้นขอบของแผ่นสามเหลี่ยมและกระทำการลดรายละเอียดด้วยวิธีการหารูปแบบที่สอดคล้องกัน (pattern matching) ของแผ่นสามเหลี่ยมข้างเคียงจากตารางการลดทอนรายละเอียด และกระบวนการแปลงกลับสู่สภาพเดิมของวัตถุ

Garland and Heckbert (1998) เสนอขั้นตอนวิธีการลดทอนรายละเอียดของวัตถุซึ่งสามารถใช้วิเคราะห์กับพื้นผิวแบบเปิด (non-manifold surface) ได้อีกด้วย โดยใช้เทคนิคการยุบหรือลดด้านของสามเหลี่ยม (Edge contraction) ดังแสดงในรูปที่ 2.6 ซึ่งด้านที่ถูกเลือกจะถูกย่อให้เป็นจุดเดียว และสามเหลี่ยมที่แรเงาจะถูกดึงออกระหว่างการยุบเส้นขอบ (contraction) พิจารณาจุดที่จะดึงออกจากโครงข่ายด้วยค่าผิดพลาดกำลังสอง (quadric error) ซึ่งเป็นผลรวมของสองเท่าของระยะทางระหว่างจุดนั้นกับระนาบที่เกิดจากสามเหลี่ยมที่ล้อมรอบจุดนั้น ขั้นตอนวิธีนี้จะรวมถึงการพิจารณาค่าสีที่แต่ละจุดหรือพื้นผิวของวัตถุด้วย ในขั้นตอนการลดทอนรายละเอียดนั้นการเลือกตำแหน่งของจุดใหม่ระหว่างจุดสองจะพิจารณาจากค่าผิดพลาดกำลังสอง ขั้นตอนแรกคือ สร้างลิสต์ของคู่จุด ซึ่งคู่จุดที่สามารถลดหรือย่อได้ ต้องเป็นด้านของสามเหลี่ยมหรือระยะทางของคู่จุดนั้นต้องน้อยกว่าค่าความถี่ที่กำหนด (threshold parameter) เมื่อค่าผิดพลาดกำลังสองของคู่จุดนั้นมีค่าน้อย ๆ ก็สามารถลดคู่จุดนั้นได้ ข้อดีของเทคนิคการยุบด้านของสามเหลี่ยมคือ ช่วยลดความผิดพลาดทางเรขาคณิตของโมเดล



รูปที่ 2.6 เทคนิคการยุบหรือลดด้านของสามเหลี่ยม (Garland and Heckbert, 1998)

งานวิจัยของ Garland and Heckbert นั้นแตกต่างจากงานวิจัยของ Schroeder et al., (1992) โดยเฉพาะขั้นตอนการลบจุดออกจากโครงข่าย ซึ่งใช้วิธีการร่วมจุดสองจุดให้เป็นจุดเดียวซึ่งอาจจะไม่เป็นตำแหน่งเดียวกันกับตำแหน่งเดิมของจุดใดจุดหนึ่ง แต่เป็นการแทนที่ตำแหน่งใหม่ในพื้นผิวเพื่อลดความผิดพลาดทางเรขาคณิตของวัตถุนั้น

Hoppe (1998) ได้พัฒนาวิธีการใหม่ในการสร้างโครงข่ายแบบก้าวหน้า (Progressive mesh, PM) โครงข่ายแบบก้าวหน้านี้ใช้เทคนิคการแปลงโครงข่ายแบบยุบเส้นขอบและการแบ่งจุด (vertex split) ซึ่งการแบ่งจุดจะมีผลตรงข้ามกับการยุบเส้นขอบ การแบ่งจุดจะสามารถสร้างจุดสองจุดจากจุดเดียวได้ หลักการสำหรับการแปลงโครงข่ายสำหรับเป็นตัวแทนของโครงข่ายแบบก้าวหน้า คือ การปรับปรุงโครงข่ายจะเกิดจากกระบวนการยุบเส้นขอบ และการสร้างโครงข่ายใหม่เกิดจากกระบวนการแบ่งเส้นขอบ กระบวนการทั้งสองนี้ไม่ได้สร้างความเสียหายภายในโครงข่าย แต่กระบวนการเหล่านี้มีประโยชน์ 2 ข้อ คือ โครงข่ายสามารถบีบอัดไปสู่รายละเอียดต่ำสุดได้ แต่สามารถแปลงกลับคืนสู่โครงข่ายแบบเดิมได้ด้วยกระบวนการแบ่งเส้นขอบ งานวิจัยนำเสนอวิธีการใหม่ซึ่งถูกใช้ในกระบวนการสร้างตัวแทนของโครงข่ายแบบก้าวหน้า ไม่เพียงแต่การพิจารณาความผิดพลาดทางเรขาคณิตของพื้นผิวของโครงข่าย แต่ยังรวมถึงการพิจารณาถึงพลังงานรวม ซึ่งกล่าวในงานวิจัยการปรับปรุงโครงข่าย (Mesh Optimization) ของ Hoppe เอง เทคนิควิธีนี้ประยุกต์ใช้เทคนิคการยุบเส้นขอบ สำหรับแต่ละรอบของกระบวนการยุบเส้นขอบโปรแกรมจะเขียนลำดับของการแบ่งเส้นขอบลงแฟ้มข้อมูล แฟ้มข้อมูลนี้จะถูกใช้เมื่อต้องการจะแปลงกลับสู่โครงข่ายเดิม

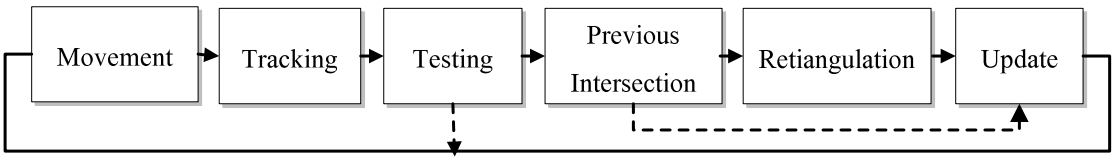
กระบวนการลดทอนรายละเอียดของวัตถุของงานวิจัยนี้ ได้ประยุกต์มาจากงานวิจัยของ Schroeder et al. ซึ่งเป็นขั้นตอนวิธีที่เหมาะสมกับการลดทอนรายละเอียดของแบบจำลองฟันสำหรับการจำลองการรักษาทางทันตกรรมจัดฟัน เพราะเป็นขั้นตอนวิธีที่สามารถประยุกต์ใช้ได้ง่ายและสะดวก โดยเฉพาะการคำนวณค่าระยะทางระหว่างระนาบกับจุด เพื่อพิจารณาเลือกจุดที่ดึงออกจากโครงข่าย เหตุผลของการลดทอนรายละเอียดของวัตถุในงานวิจัยนี้คือ การลดเวลาสำหรับการแสดงผลและการประมวลผลเท่านั้น ซึ่งไม่ได้คำนึงถึงค่าความผิดพลาดทางเรขาคณิตของแบบจำลองฟันที่ถูกลดทอนรายละเอียดมากนัก จึงไม่จำเป็นต้องใช้เทคนิควิธีที่ซับซ้อน

2.3 การตัดพื้นผิว (Surface Cutting)

กระบวนที่สำคัญสำหรับการจำลองการรักษาทางทันตกรรมจัดฟัน (Orthodontic simulation) คือ การแยกองค์ประกอบของฟันออกจากองค์ประกอบอื่น เช่น เหงือก เพื่อใช้สำหรับการจัดเรียงฟัน การแยกองค์ประกอบของฟันนั้นใช้เทคนิควิธีการตัดพื้นผิว ซึ่งเป็นเทคนิควิธีที่ใช้กันอย่างกว้างขวางและแต่ละเทคนิควิธีที่ถูกประยุกต์ใช้ต่างมีข้อจำกัดที่แตกต่างกันออกไป ตัวอย่างการประยุกต์ใช้เทคนิควิธีการตัดพื้นผิวมีดังนี้

Bruyns and Senger (2001) ได้เสนอเทคนิควิธีในการตัดพื้นผิวของโมเดล เป็นการตัดพื้นผิววัตถุแบบ real-time โดยมีกระบวนการทั้งหมด 4 กระบวนการ คือ

1) Placement คือ กระบวนการกำหนดจุดเริ่มต้นของการตัดพื้น เมื่อผู้ใช้คลิกเมาส์เพื่อกำหนดจุดเริ่มต้น ซึ่งจุดที่เมาส์คลิกเป็นจุดซึ่งอยู่บนระบบ 2 มิติ การคลิกเมาส์จะได้รังสีที่ฉาย (ray pick) ที่ฉายไปยังวัตถุใน 3 มิติ แต่จุดของวัตถุนั้นอยู่ในระบบ 3 มิติ ดังนั้นจึงต้องค้นหาจุดที่คลิกบนหน้าแสดงผลกับจุดของวัตถุในระบบ 3 มิติ โดยใช้วิธีการฉาย (Projection) และหาจุดบนวัตถุที่ใกล้กับเส้นรังสีฉายที่สุด
2) Activation คือ กระบวนการกำหนดให้เครื่องมือตัด (cutter) ทำงานได้ สามารถลากผ่านพื้นผิวของวัตถุได้
3) Cutting คือ กระบวนการตัดพื้นผิวของวัตถุ ซึ่งมีขั้นตอนการทำงานดังรูปที่ 2.7



รูปที่ 2.7 ขั้นตอนการทำงานของ Cutting callback (ปรับปรุงจาก Bruyns and Senger, 2001)

a) Movement : การวาดเส้นทางในการตัวพื้นผิวลงในพื้นผิวของโมเดล โดยจะลากผ่านด้าน (Edge) ของรูปสามเหลี่ยม

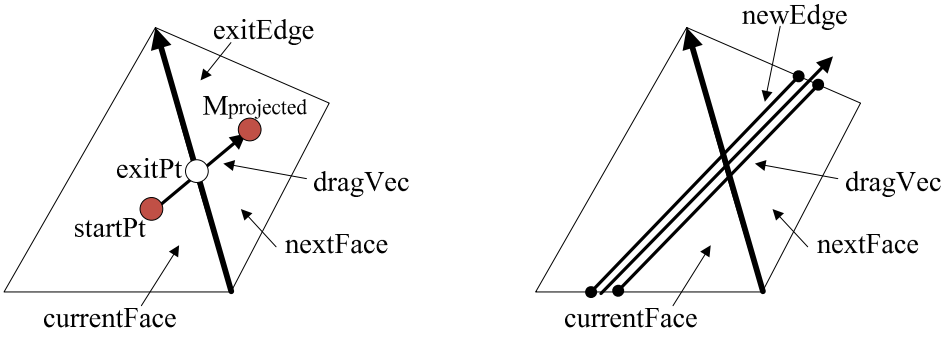
b) Tracking : วิธีการหาด้านที่ถัดจากด้านเริ่มต้น โดยใช้เวกเตอร์ฉาย (projection vector) ที่ได้จากการลาก (dragVec) บนด้านล่าสุดที่ลากผ่าน ทำให้สามารถทำนายได้ว่าด้านถัดไปจะอยู่ด้านซ้ายหรือขวาของด้านตั้งต้น และเวกเตอร์ฉายที่ได้จากการลากบนเวกเตอร์ที่ได้จากการคูณเวกเตอร์ระหว่างด้านล่าสุดกับเวกเตอร์ทิศทาง (normal vector) ของสามเหลี่ยมปัจจุบัน ทำให้สามารถทำนายได้ว่าจะเลื่อนไปข้างหน้าหรือหลังสามเหลี่ยมปัจจุบัน ผลลัพธ์ที่ได้คือลำดับของด้านทั้งหมดที่ถูกลากผ่าน

c) Testing : ทดสอบว่าเส้นที่ลากผ่านด้านของสามเหลี่ยมจริงหรือไม่ ดังรูปที่ 2.8(a)

d) Retriangulation : การสร้างรูปสามเหลี่ยมใหม่ขึ้นมาเพื่อให้เกิดรอยตัด โดยกำหนดให้จุดที่ลากผ่านบนแต่ละด้านขึ้น 2 จุด แล้วสร้างด้านใหม่ ดังรูปที่ 2.8(b)

e) Updating : การปรับปรุงโครงสร้างข้อมูลของโครงข่าย

4) Deactivation คือ กระบวนการหยุดการทำงานของเครื่องมือตัด

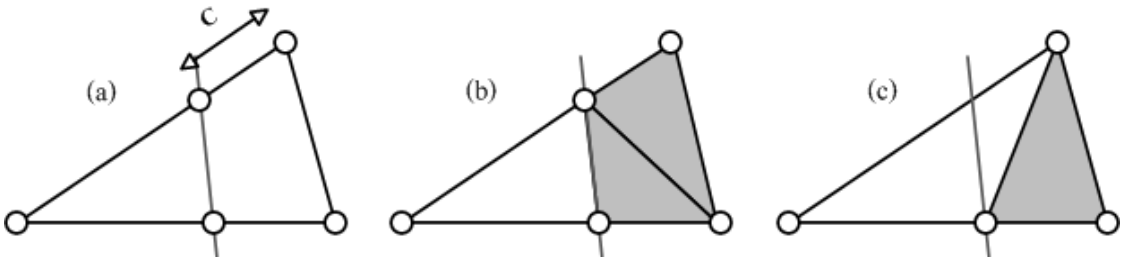


รูปที่ 2.8 แสดงเส้นทางการตัดที่ลากผ่านด้านของสามเหลี่ยม (a)
และแสดงการสร้างสามเหลี่ยมใหม่ (b)
(ปรับปรุงจาก Bruyns and Senger, 2001)

Lindenbeck et al. (2002) พัฒนาโปรแกรม TRICUT เพื่อตัดโครงข่ายสามเหลี่ยมโดยใช้ไลบารี RAPID (Robust and Accurate Polygon Interference Detection), TRIANGLE และ VTK® (Visualization Toolkit) (Schroeder et al., 2002) ไลบารี RAPID สำหรับจัดการการชนกันระหว่างโครงข่าย (collision detection) และไลบารี TRIANGLE สำหรับสร้างสามเหลี่ยมใหม่ ขั้นตอนวิธีทั้งหมดถูกพัฒนาบนไลบารี VTK® การตัดจะเกิดจากการตัดกันระหว่างโครงข่าย เพื่อโต้ตอบกับ

ที่ตัดกัน สำหรับแต่ละคู่ของสามเหลี่ยมต้องกำหนดจุดตัด จากนั้นสร้างเส้นของการตัดระหว่างโครงข่ายเพื่อสร้างสามเหลี่ยมใหม่ ลบสามเหลี่ยมที่โผล่พ้นเกินขึ้นมาและบันทึกโครงข่ายผลลัพธ์ไว้ ขั้นตอนการอ่านและบันทึกข้อมูลโครงข่ายใช้ไลบารี VTK® ส่วนขั้นตอนกำหนดคู่สามเหลี่ยมที่ตัดกันใช้ไลบารี RAPID และขั้นตอนการสร้างสามเหลี่ยมใหม่ใช้ไลบารี TRIANGLE โปรแกรม TRICUT มีประโยชน์สำหรับการจัดการกับพื้นผิวของรูปทรงในด้านต่าง ๆ เช่น การศึกษาโครงสร้างทางธรณี การทำเหมืองแร่และปิโตรเลียม เป็นต้น

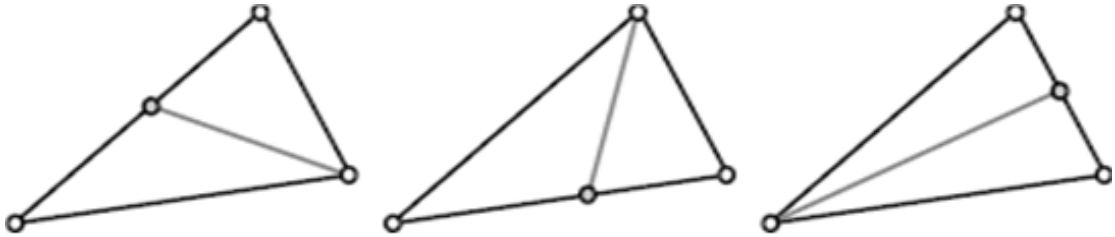
Pintilie and McInerney (2003) เสนอต้นแบบระบบการวางแผนผ่าตัดซึ่งจำลองการตัดและการจัดการตำแหน่งของวัตถุแบบแข็ง อย่างเช่น กระโหลก ด้วยการจำลองแบบเสมือนจริงและสามารถโต้ตอบกับผู้ใช้ได้ ระบบนี้สนับสนุนการตัดด้วยระนาบ (planar) และเส้นรอบขอบ (contour) ทั้งหมดนี้จะกระทำกับโครงข่ายสามเหลี่ยมปิด การตัดด้วยเส้นรอบขอบสามารถตัดได้ทั้งเส้นรอบขอบเปิดและปิด การตัดด้วยระนาบสร้างการตัดโดยประมาณผ่านโครงข่ายสามเหลี่ยมด้วยระนาบที่กำหนดโดยผู้ใช้ ขั้นตอนแรกคือ จำแนกประเภทของจุดบนโครงข่าย ออกเป็น 3 ประเภท คือ จุดเหนือระนาบ จุดใต้ระนาบ และจุดภายในระยะทาง p ของระนาบ ระยะทาง p คือ ค่าความผิดพลาดที่ยอมรับได้ในการประมาณการตัด ขั้นตอนที่สองคือ แบ่งสามเหลี่ยมที่ระนาบตัดผ่านซึ่งจะประกอบด้วยจุดเหนือระนาบและจุดใต้ระนาบอย่างละจุด การแบ่งสามเหลี่ยมคือ การสร้างเหลี่ยมใหม่ด้วยวิธีการของ Delaunay ถ้าระยะทาง $c > p$ จะเพิ่มจุดใหม่สองจุดและสามเหลี่ยมสองรูป มิฉะนั้นจะเพิ่มจุดเพียงจุดเดียวและเพิ่มสามเหลี่ยมรูปเดียว แสดงดังรูปที่ 2.9



รูปที่ 2.9 ส่วนประกอบของสามเหลี่ยมที่ถูกตัดผ่านด้วยระนาบ (Pintilie and McInerney, 2003)

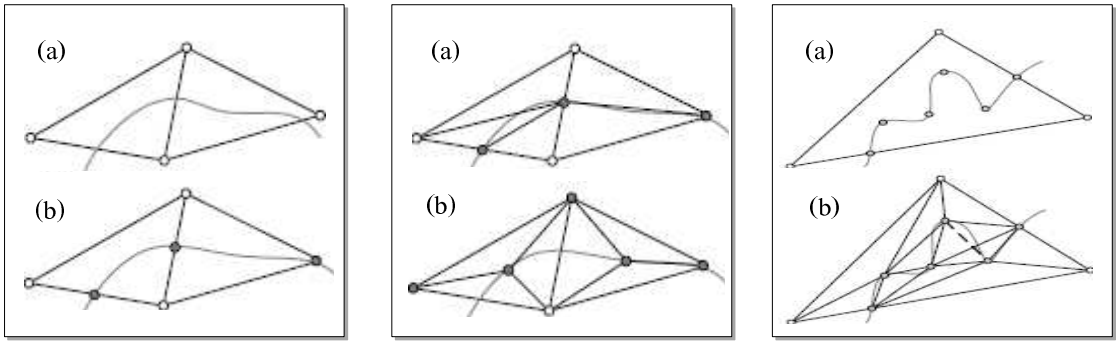
หลักการตัดด้วยเส้นรอบขอบเป็นการคำนวณหาจุดที่เส้นรอบขอบลากผ่าน ประกอบด้วยขั้นตอน 4 ขั้นตอน คือ

1) แบ่งโครงข่ายในบริเวณของการตัด การแบ่งโครงข่ายจะเกิดขึ้นเมื่อเส้นทางการตัดตัดนั้นตัดผ่านด้าน และจุดของสามเหลี่ยม ดังรูปที่ 2.10



รูปที่ 2.10 แสดงการแบ่งสามเหลี่ยมที่เส้นรอบขอบตัดผ่าน (Pintilie and McInerney, 2003)

2) เปลี่ยนเส้นรอบขอบการตัดเพื่อให้เส้นทางการตัดวางตัวอยู่บนพื้นผิวของโครงข่าย เป็นกระบวนการเพิ่มจุดบนด้านของสามเหลี่ยมที่เส้นรอบขอบลากผ่าน ดังรูปที่ 2.11(a)
3) สุ่มเส้นรอบขอบการตัดเพื่อสร้างจุดตัด เพื่อให้เส้นทางการตัดไม่เบี่ยงเบน ดังรูปที่ 2.11(b)
4) ปรับปรุงโครงข่าย เป็นการสร้างสามเหลี่ยมใหม่ ซึ่งเป็นผลจากการเพิ่มจุดสุ่มบนเส้นรอบขอบ ดังรูปที่ 2.11(c)

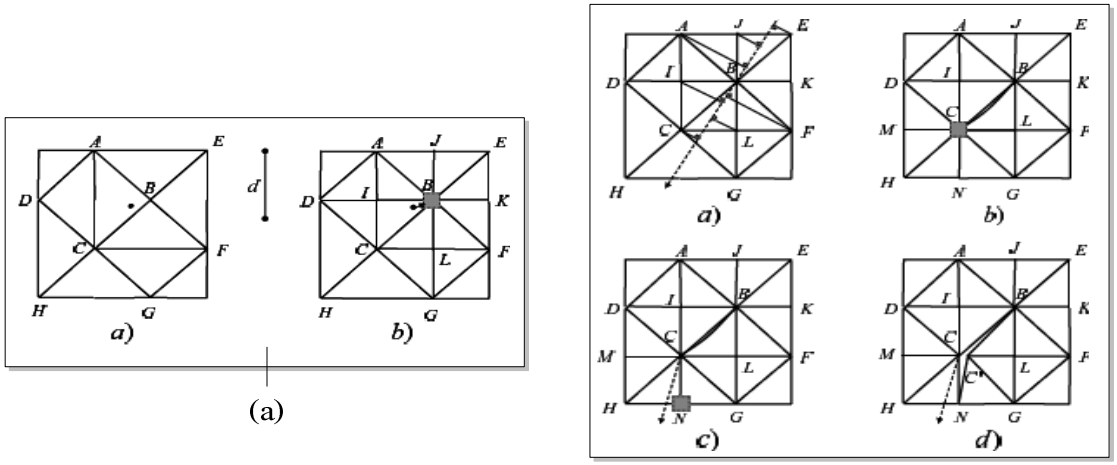


(a) (b) (c)

รูปที่ 2.11 แสดงจุดที่ถูกเพิ่มในเส้นทางการตัด (a) การสร้างจุดตัดบนเส้นรอบขอบ (b) และการปรับปรุงโครงข่ายที่จุดสุ่มบนเส้นรอบขอบ (c)
(Pintilie and McInerney, 2003)

HuyVeit, Kamada, and Tanaka (2006) เสนอเทคนิคการตัดพื้นผิวด้วยวิธีการแบ่งขนาดของด้านที่ยาวที่สุดแบบย้อนกลับ (Backward Longest-Size Refinement) ซึ่งเป็นขั้นตอนวิธีของไรวารา (Rivara, 1996) ใช้สำหรับการแบ่งในโครงข่ายแบบสามเหลี่ยม ความต้องการของการแบ่งโครงข่ายต้องมีคุณสมบัติ คือ สามเหลี่ยมต้องมีลักษณะคล้ายกัน (conforming) รูปร่างเหมาะสม (well shaped)

และรอยแบ่งต้องเรียบ (smooth) ลักษณะคล้ายกันของสามเหลี่ยมภายในโครงข่ายคือ โครงข่ายนั้นต้องไม่มีทางแยกรูปตัวที (T-junctions) โครงข่ายที่รูปร่างเหมาะสมคือ มุมภายในของสามเหลี่ยมต้องอยู่ในช่วง 0 ถึง π องศา ขั้นตอนวิธีการตัด เริ่มต้นจากค้นหาจุดใกล้เคียงกับจุดที่เครื่องมือผ่าตัดกำหนดโดยวิธีการแบ่งขนาดด้านที่ยาวที่สุดแบบย้อนกลับ (Backward Longest-Size Refinement) แสดงดังรูปที่ 2.12(a) เพื่อกำหนดสามเหลี่ยมแรกในการแบ่ง จากนั้นพิจารณาเลือกจุดถัดไปโดยเลือกจากจุดที่อยู่ข้างเคียงกับจุดเริ่มต้นซึ่งมีระยะทางใกล้ที่สุดกับเส้นทางการตัด แล้วสร้างสามเหลี่ยมใหม่จากด้านที่แบ่งและสร้างจุดขึ้นมาใหม่ด้วย แสดงดังรูปที่ 2.12(b)



(a)

(b)

รูปที่ 2.12 แสดงการเริ่มต้นกระบวนการตัดโครงข่าย (a)
และการสร้างโครงข่ายใหม่ (b) (HuyVeit et al., 2006)

การงานวิจัยนี้สามารถสรุปได้ว่ากระบวนการตัดโครงข่ายนั้นประกอบด้วย กระบวนการสร้างโครงข่ายใหม่และสร้างจุดใหม่ จากผลการทดลองที่ได้ยังพบว่าผลลัพธ์ที่ได้จากการตัดมีความเรียบ ถึงแม้เส้นทางการตัดจะตัดผ่านจุดของสามเหลี่ยมไม่ได้ตัดผ่านด้านของสามเหลี่ยม สาเหตุที่รอยตัดเรียบเพราะมีการสร้างสามเหลี่ยมใหม่ขึ้นมาอย่างมากระหว่างเส้นทางการตัด ซึ่งเป็นข้อเสียของขั้นตอนวิธีนี้

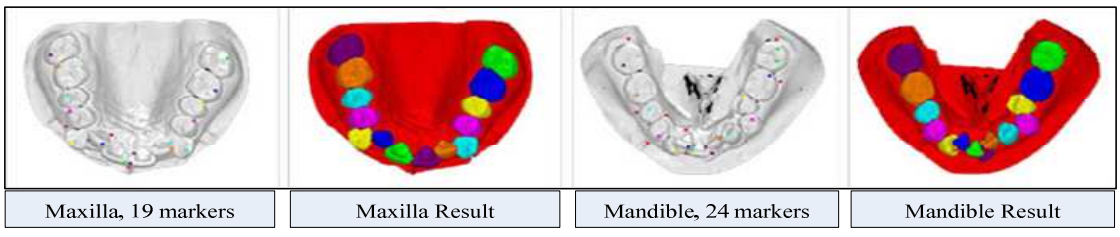
Wong et al. (1998) ได้เสนอเทคนิควิธีการตัดปริมาตรของโครงข่าย ซึ่งเป็นการตัดด้วยเส้นรอบขอบ (contour) โดยการกำหนดจุดต่าง ๆ ซึ่งจุดที่กำหนด เรียกว่า “seed point” และหาระยะที่สั้นที่สุดจากจุด “seed point” กับจุดทั้งหมดบนวัตถุ โดยใช้ Dijkstra’s algorithms เพื่อสร้างเส้นทางการตัด เมื่อได้เส้นรอบขอบแบบปิด (Closed contour) แล้วขั้นตอนต่อไปคือ กระบวนการตัดปริมาตร

เมื่อตัดแล้วจะได้พื้นผิว 2 แบบ คือ พื้นผิวภายนอก และพื้นผิวที่ถูกตัด ซึ่งในส่วนของพื้นผิวที่ถูกตัดจำเป็นต้องสร้างโครงข่ายขึ้นมาใหม่ การหาพื้นที่ส่วนที่ถูกตัด เริ่มต้นจากขอบของโครงข่ายที่อยู่ติดกับเส้นรอบขอบไปสู่จุดที่อยู่ภายในโครงข่าย โดยใช้ฟังก์ชันประเมินค่า (cost function) ดังสมการที่ 2.3 พจน์แรกของสมการจะเป็นกำหนดให้พื้นผิวที่ตัดเรียบ สำหรับพจน์ที่สองทำให้พื้นผิวเป็นพื้นผิวที่มีค่าเท่ากัน (iso-surface) โดยที่ M คือ จำนวนของจุดบนปริมาตร (voxel) และ I คือ ค่าความเข้มของจุด

$$C_M = \sum_{(x,y,z)\in M} [-\alpha |\nabla I(x,y,z)| + \beta S(x,y,z)] \tag{2.3}$$

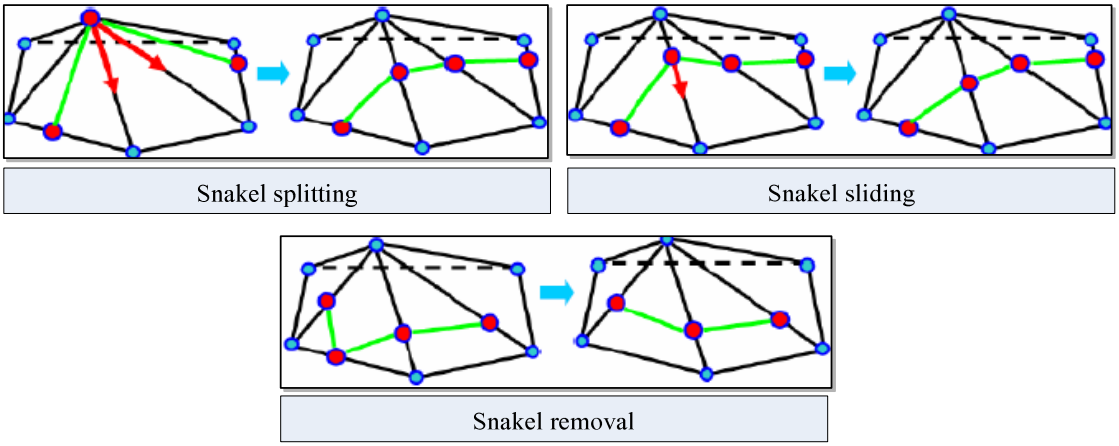
หลังจากสิ้นเสร็จกระบวนหาพื้นผิวที่ถูกตัด จะได้ข้อมูลเกี่ยวกับรูปร่างของโครงข่าย และรอยต่อ (junction) ทั้งหมดของโครงข่าย ปัญหาที่เกิดจากการใช้เส้นรอบขอบในการตัดโครงข่ายคือไม่สามารถแยกโครงข่ายที่ตัดออกเป็นสองส่วนได้ สาเหตุเกิดจากปัญหาจากรูปร่างลักษณะของโครงข่าย ลักษณะของโครงข่ายที่ทำให้เกิดปัญหาดังกล่าวคือ โครงข่ายรูปทรงคล้ายห่วงยาง (torus)

Chi et al. (2004) เสนอการแยกองค์ประกอบของโครงข่ายด้วยจุดควบคุม (maker-controlled) ผู้ใช้สามารถเพิ่มจุดควบคุมเพื่อให้ได้ผลลัพธ์ตามที่ต้องการ นอกจากจะใช้จุดควบคุมสำหรับการแยกองค์ประกอบโครงข่ายแล้วยังใช้เทคนิควิธีการประมาณค่าความโค้งและ region-growing การกำหนดจุดควบคุมจำเป็นต้องกำหนดอย่างน้อยสองจุด ซึ่งเป็นส่วนที่ต้องการตัดและส่วนที่เหลือ จำนวนของจุดควบคุมเป็นตัวกำหนดจำนวนองค์ประกอบที่จะแยก งานวิจัยนี้ได้ทำการทดสอบแยกองค์ประกอบของฟันปลอมด้วย ดังรูปที่ 2.13 ซึ่งจะเห็นได้ว่าถ้าแบบจำลองฟันมีความซับซ้อน เช่น มีฟันซ้อนกันมาก จำเป็นต้องใช้จำหนดจุดควบคุมเพิ่มขึ้นด้วย ซึ่งแบบจำลองฟันล่าง (mandible) มีฟันซ้อนกันมากกว่าแบบจำลองฟันบน (maxilla) ทำให้ใช้จุดควบคุมมากกว่า โดยใช้จุดควบคุมทั้งหมด 24 จุด ขณะที่แบบจำลองฟันบนใช้ 19 จุดควบคุม



รูปที่ 2.13 แสดงองค์ประกอบของฟันปลอม (Chi et al., 2004)

Ji et al. (2006) เสนอขั้นตอนวิธีการตัดโครงข่ายอย่างง่าย โดยผู้ใช้สามารถใช้เครื่องมือการตัดที่ง่ายต่อการใช้งาน ผู้ใช้สามารถตัดองค์ประกอบที่สำคัญจากโครงข่ายด้วยการวาดด้วยมือบนโครงข่ายนั้น ขั้นตอนการตัดโครงข่ายนี้ใช้เทคนิควิธีการปรับปรุงขั้นตอนวิธีการแยกองค์ประกอบของภาพ (region-growing) ซึ่งขอบการตัดสามารถปรับปรุงได้อัตโนมัติหรือผู้ใช้สามารถปรับได้ง่าย ผู้ใช้จำเป็นต้องกำหนดส่วนที่ต้องการตัดและส่วนที่เหลือด้วยการวาด ขั้นตอนการแยกองค์ประกอบใช้วิธีการประมาณด้วยค่าความโค้ง (curvature estimation) และ region-growing การปรับปรุงขอบการตัดให้เรียบขึ้นโดยอัตโนมัติใช้ขั้นตอนวิธีของ Bischoff et al. (2004) คือ snake หรือ active model contour (Kass et al., 1988) ซึ่งประกอบด้วย จุดและขอบ การปรับปรุงขอบการตัดจะพิจารณาจากค่าพลังงานภายในและภายนอกของ snake โดยที่ค่าพลังงานภายในคำนวณได้จากค่ามุมที่กระทำกับด้านเคียงข้างและค่าพลังงานภายนอกคำนวณจากค่าความโค้ง การเลื่อนตำแหน่งของ snake มี 3 แบบ คือ การแบ่ง (snakel splitting) การเลื่อน (snakel sliding) และการลบ (snakel removal) ดังรูปที่ 2.14 การปรับปรุงขอบการตัดโดยผู้ใช้สามารถทำได้โดยการวาดเส้นขอบด้วยตนเองลงบนโครงข่าย



รูปที่ 2.14 การเลื่อนตำแหน่งของ snake (Ji et al., 2006)

กระบวนการตัดพื้นผิว ประกอบด้วยขั้นตอนที่สำคัญ คือ การกำหนดเส้นทางการตัดและการปรับปรุงโครงข่ายใหม่ เพื่อให้รอยตัดที่ได้เรียบจำเป็นต้องตัดผ่านขอบของสามเหลี่ยมด้วย การกำหนดเส้นทางการตัดสามารถสร้างได้การลากเส้นหรือการกำหนดจุดบนพื้นผิว เพื่อความสะดวกในกระบวนการกำหนดเส้นทางการตัดรอบขอบฟัน ต้องกำหนดจุดตัดรอบขอบฟันแล้วสร้างเส้นทางการตัด สำหรับการค้นหาส่วนของฟันสามารถใช้เทคนิควิธีการแยกองค์ประกอบของภาพโดยเลือกใช้วิธี Region growing ซึ่งเป็นวิธีการที่เหมาะสม เพราะเนื่องจากทราบจุดที่กำหนดขอบเขตของการตัดแล้ว เพียงแค่เลือกจุดเริ่มต้น (start seed vertex) เพื่อค้นหาจุดต่าง ๆ ที่เป็นองค์ประกอบของฟัน ซึ่งจุดที่เป็นเงื่อนไขการหยุดค้นหาก็คือจุดที่เป็นขอบเขตของการตัดนั่นเอง

2.4 การสร้างแบบจำลองมวลสปริง (Mass-spring Model)

สำหรับขั้นตอนการจัดเรียงฟัน เพื่อวางแผนการรักษาทางทันตกรรมจัดฟัน กระบวนการที่จำเป็นนอกจากกระบวนการแยกฟันออกจากเหงือก คือ กระบวนการจำลองการเปลี่ยนแปลงรูปร่างของเหงือกระหว่างที่จัดเรียงฟัน เทคนิคการจำลองการเปลี่ยนแปลงรูปร่างของวัตถุมีมากมายหลายเทคนิควิธี แต่เทคนิควิธีที่ใช้กันอย่างมากและสามารถจำลองการเปลี่ยนแปลงรูปร่างของวัตถุได้อย่างเสมือนจริง นั่นคือ เทคนิคการสร้างแบบจำลองมวลสปริง ซึ่งงานวิจัยที่ประยุกต์ใช้เทคนิคการสร้างแบบจำลองมวลสปริงเพื่อศึกษาการเปลี่ยนแปลงรูปร่างของวัตถุมีดังนี้

Terzopoulos et al. (1987) ได้ริเริ่มประยุกต์นำเอาทฤษฎีความยืดหยุ่น (elasticity) มาใช้อธิบายพฤติกรรมของวัตถุที่เปลี่ยนรูปร่างได้ (deformable models) เช่น เสื้อผ้า กระดาษ ยางยืด และโลหะที่โค้งงอได้ วัตถุจะเปลี่ยนรูปร่างก็ต่อเมื่อมีแรงกระทำ ประกอบด้วย แรงโน้มถ่วงของโลก แรงสปริงซึ่งเป็นแรงกระทำระหว่างจุดบนวัตถุ แรงหนืดจากของไหล และแรงที่เกิดจากการชนกัน เนื่องจากแรงที่กระทำมีลักษณะต่อเนื่อง แต่การจำลองพฤติกรรมของวัตถุสามารถกระทำได้ในแต่ละช่วงเวลานั้น (discretized) ด้วยการประยุกต์ใช้เทคนิคไฟไนต์อีเลเมนต์ (finite element) การจำลองพฤติกรรมยืดหยุ่นจะจำลองในแต่ละช่วงเวลา แรงลัพธ์ทั้งหมดคำนวณได้จากแรงที่เกิดจากความเร่ง แรงต้านของสปริง (damping) และแรงกระทำของสปริงระหว่างมวล ดังสมการที่ 2.4 ซึ่งแก้สมการด้วยวิธีการของ Runge-Kutta (Ascher et al., 1998)

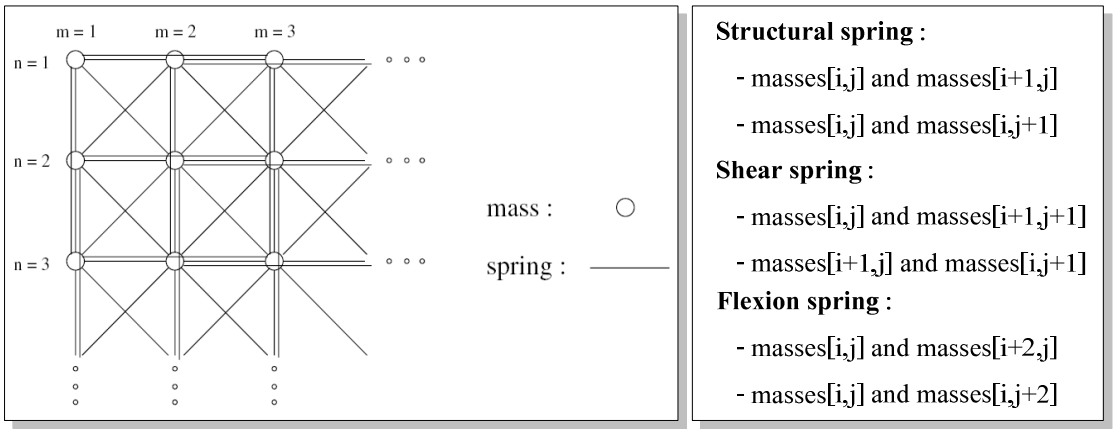
$$M\frac{\partial^2 r}{\partial t^2} + C\frac{\partial r}{\partial t} + K(r)r = f \tag{2.4}$$

เมื่อ M คือ เมตริกซ์ของมวลสปริง (mass matrix)

C คือ เมตริกซ์ของค่าความหน่วงสปริง (damping matrix)

K คือ เมตริกซ์ของค่าคงที่สปริง (stiffness matrix)

Provot (1995) ได้เสนอแบบจำลองทางกายภาพ (physically-based model) สำหรับการจำลองการเคลื่อนไหวของเสื้อผ้า ซึ่งพัฒนามาจากแบบจำลองที่สามารถเปลี่ยนรูปร่างแบบยืดหยุ่น (elastically deformable model) แบบจำลองของเสื้อผ้าประกอบด้วยโครงข่ายของมวลและสปริง โดยแบ่งประเภทของสปริงออกเป็น 3 ประเภท คือ สปริงโครงสร้าง (structural spring) สปริงเฉือน (shear spring) และสปริงงอ (flexion spring) แสดงดังรูปที่ 2.15



รูปที่ 2.15 ประเภทของสปริง สำหรับแบบจำลองมวลสปริง (Provot, 1995)

แรงภายในของสปริงสามารถคำนวณได้จากแรงกระทำของสปริงรอบข้างดังสมการที่ 2.5 งานวิจัยนี้ยังได้ศึกษาผลกระทบของสภาพยืดหยุ่นยิ่งยวด (Super-elastic) ต่อการเคลื่อนที่ของแบบจำลองเสื้อผ้า ซึ่งพบว่าจุดหรือมวลที่อยู่ใกล้กับจุดกำเนิดแรงจะเปลี่ยนแปลงตำแหน่งและรูปร่างอย่างรวดเร็ว วิธีแก้ปัญหาดังกล่าวคือ การเพิ่มค่าคงที่ของสปริง การคำนวณแรงที่กระทำกับจุดแต่ละจุดในแต่ละช่วงเวลา คำนวณได้จากสมการกฎพื้นฐานของเคลื่อนที่ของนิวตัน ซึ่งสามารถหาปริพันธ์แบบชัดแจ้ง (explicit integration) ด้วยวิธีการของ Euler (Hairer et al., 2006) ดังสมการที่ 2.6

$$F_{\text{int}}(\mathbf{P}_{i,j}) = -\sum_{(k,l)\in R} \mathbf{K}_{i,j,k,l}\left[\mathbf{l}_{i,j,k,l} - l^{0}_{i,j,k,l}\frac{\mathbf{l}_{i,j,k,l}}{\left\|\mathbf{l}_{i,j,k,l}\right\|}\right] \tag{2.5}$$

เมื่อ **P** คือ ตำแหน่งพิกัดของจุดมวล

K คือ ค่าคงที่สปริง

l คือ ความยาวของสปริงที่ถูกยืดหรือหด

l^0 คือ ความยาวเริ่มต้นของสปริง

$$\begin{cases} a_{i,j}(t+\Delta t) = \frac{1}{\mu}F_{i,j}(t) \\ v_{i,j}(t+\Delta t) = v_{i,j}(t) + \Delta t a_{i,j}(t+\Delta t) \\ P_{i,j}(t+\Delta t) = P_{i,j}(t) + \Delta t v_{i,j}(t+\Delta t) \end{cases} \tag{2.6}$$

เมื่อ P คือ ตำแหน่งพิกัดของจุดมวล

a คือ ความเร่งของจุดมวล

v คือ ความเร็วของจุดมวล

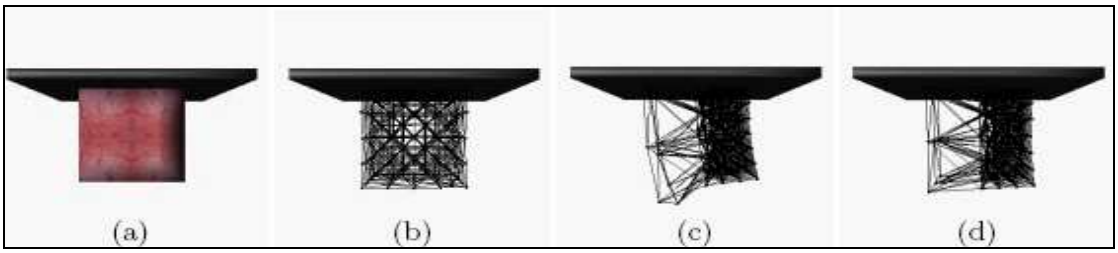
μ คือ มวลของจุดมวล

Nedel and Thalmann (1998) ได้จำลองการเปลี่ยนรูปร่างของกล้ามเนื้อด้วยระบบมวลสปริง (Mass-spring system) แบบจำลองกล้ามเนื้อสามารถแบ่งออกเป็น 2 ระดับ คือ เส้นและรูปร่างของกล้ามเนื้อ เส้นกล้ามเนื้อจะแสดงแรงที่กระทำกับกล้ามเนื้อบนกระดูก ขณะที่รูปร่างกล้ามเนื้อถูกใช้ในการจำลองพื้นผิวของแบบจำลองที่เหมาะสมกับขอบของภาพการแพทย์ ระบบมวลสปริงใช้ชนิดสปริงแบบใหม่ เรียกว่า “angular spring” การแก้สมการการเคลื่อนที่ของกล้ามเนื้อใช้เทคนิควิธี 4th order-Runge Kutta

ระบบการจำลองพฤติกรรมของเสื้อผ้า (cloth simulation) ส่วนใหญ่ ช่วงเวลาที่ใช้ในการจำลองในแต่ละรอบใช้ช่วงเวลาน้อย ๆ เพื่อหลีกเลี่ยงความไม่แน่นอนของการวิเคราะห์เชิงตัวเลข เป็นผลให้การจำลองต้องใช้เวลานาน จากปัญหาดังกล่าวทำให้ Baraff and Witkin (1998) เสนอวิธีการจำลองพฤติกรรมของเสื้อผ้าด้วยช่วงเวลามาก ๆ แต่การวิเคราะห์เชิงตัวเลขยังมีความแน่นอนอยู่และสามารถจำลองได้อย่างรวดเร็ว การจำลองที่รวดเร็วนี้ เริ่มต้นจากการเลือกวิธีการหาปริพันธ์โดยปริยาย (implicit integration) ซึ่งปรับปรุงวิธีการ Conjugate gradient สำหรับแก้สมการการ

เคลื่อนที่ ซึ่งงานวิจัยที่ได้กล่าวมาข้างต้นเลือกใช้เทคนิควิธีการหาปริพันธ์แบบชัดแจ้ง (explicit integration) เช่น วิธีการของออยเลอร์ (Euler) หรือรุงเก-คุตตา (Runge-Kutta) วิธีการแบบชัดแจ้งนี้ไม่เหมาะสมกับการจำลองพฤติกรรมของเสื้อผ้า เพราะวิธีการนี้ต้องการช่วงเวลาแต่ละขั้นตอน (time step) ที่น้อยมาก ๆ

Paloc et al. (2002) ได้เสนอแบบจำลองมวลสปริงที่ประกอบด้วยรายละเอียดหลายระดับ (multiresolution) และได้ศึกษาพฤติกรรมที่เหมาะสมของแบบจำลองมวลสปริงสำหรับโครงข่ายสามเหลี่ยม เช่น การแกว่งของแบบจำลองภายใต้แรงโน้มถ่วง ดังรูปที่ 2.16



รูปที่ 2.16 แบบจำลองกล่องที่สามารถเปลี่ยนรูปได้ (a) ภายใต้แรงโน้มถ่วงด้วยการกำหนดค่าตัวแปรที่ต่างกัน (b) รายละเอียดเดียว (c) รายละเอียดหลายระดับโดยกำหนดค่า $k=1/l$ และค่าคงที่การหน่วง(c) รายละเอียดหลายระดับโดยกำหนดค่าคงที่สปริงดังสมการที่ 2.7 และค่าคงที่การหน่วงดังสมการที่ 2.8 (Paloc et al., 2002)

จากการศึกษาพบว่าพฤติกรรมดังกล่าวขึ้นอยู่กับค่าตัวแปรต่าง ๆ ของแบบจำลองมวลสปริง ซึ่งประกอบด้วย ค่ามวลสปริง (point mass, m) ค่าคงที่สปริง (spring stiffness, k) และค่าคงที่การหน่วง (spring damping, c) มวลสปริงสามารถคำนวณได้จากค่าปริมาตรที่เกิดจากจุดเคียงข้างดังสมการที่ 2.7 ค่าคงที่สปริงคำนวณได้จากอัตราส่วนของปริมาตรต่อความยาวดังสมการที่ 2.8 และค่าคงที่การหน่วงสามารถคำนวณได้ดังสมการที่ 2.9

$$m_i = \frac{D\sum_j V_j}{4} \tag{2.7}$$

$$k_i = \frac{E\sum_j V_j}{l^2} \tag{2.8}$$

$$c_i = \frac{2\sqrt{kM}}{l} \tag{2.9}$$

เมื่อ D คือ ความหนาแน่นของวัตถุ

V คือ ปริมาตรสี่ระนาบของสามเหลี่ยมข้างเคียง

E คือ มอดูลัสความยืดหยุ่น

M คือ มวลรวมของสปริง

Metaaphanon and Kanongchaiyos (2005) ได้จำลองพฤติกรรมของเสื้อผ้าสำหรับการใช้คอมพิวเตอร์ช่วยออกแบบเสื้อผ้าและเครื่องแต่งกาย โดยประยุกต์ใช้แบบจำลองมวลสปริง ซึ่งแบ่งประเภทของสปริงออกเป็น 3 ประเภท ข้อที่ 2 ของนิวตัน การแก้สมการการเคลื่อนที่ใช้วิธีการหาปริพันธ์ของ 4^{th} order-Runge Kutta ดังสมการที่ 2.10

$$\begin{aligned}
k_1 &= hf(x_0, t_0) \\
k_2 &= hf(x_0 + k_1/2, t_0 + h/2) \\
k_3 &= hf(x_0 + k_2/2, t_0 + h/2) \\
k_4 &= hf(x_0 + k_3, t_0 + h) \\
x(t_0 + h) &= x_0 + (k_1 + 2k_2 + 2k_3 + k_4)/6
\end{aligned} \tag{2.10}$$

Debunne et al. (2001) ได้เสนอขั้นตอนวิธีที่เสถียรและมีประสิทธิภาพสำหรับการจำลองของแบบจำลองมวสปริง ขั้นตอนการหาปริพันธ์ได้ประยุกต์ใช้เทคนิคการหาปริพันธ์โดยปริยายจากสมการที่ 2.11 ซึ่งเป็นเทคนิคการหาปริพันธ์แบบชัดแจ้งของ Euler แต่สำหรับวิธีการหาปริพันธ์โดยปริยายของ Euler นั้นแทนที่จะคำนวณหาแรงที่เวลา t ใด ๆ ก็เปลี่ยนเป็นคำนวณหาแรงที่เวลา $t + \Delta t$ ดังสมการที่ 2.12

$$\begin{cases} x(t + \Delta t) = x(t) + \Delta t v(t) \\ v(t + \Delta t) = v(t) + \Delta t F(t)/m \end{cases} \tag{2.11}$$

$$\begin{cases} x(t + \Delta t) = x(t) + \Delta t v(t + \Delta t) \\ v(t + \Delta t) = v(t) + \Delta t F(t + \Delta t)/m \end{cases} \tag{2.12}$$

แบบจำลองมวลสปริง ประกอบด้วย จุดมวลและสปริง การจำลองการเปลี่ยนรูปร่างหรือตำแหน่งของจุดสามารถคำนวณได้จากสมการกฎการเคลื่อนของนิวตัน โดยแรงที่กระทำกับสปริงประกอบด้วย แรงภายในและแรงภายนอก แรงภายใน คือ แรงกระทำของสปริง ส่วนแรงภายนอก

ได้แก่ แรงโน้มถ่วง แรงลม และแรงต้านของสปริง ซึ่งงานวิจัยนี้จะใช้เฉพาะแรงต้านของสปริงเท่านั้น เพราะกระบวนการจำลองเปลี่ยนแปลงรูปร่างของเหงือกเกิดจากการจัดเรียงตัวของฟันซึ่งเป็นแรงภายในของสปริงเท่านั้น การแก้สมการการเคลื่อนที่สามารถแก้ได้ด้วยวิธีการหาปริพันธ์ทั้งแบบปริยายและชัดแจ้ง งานวิจัยนี้แก้สมการด้วยวิธีการหาปริพันธ์แบบชัดแจ้ง (Explicit integration) ซึ่งสามารถแก้สมการได้อย่างง่ายและรวดเร็ว

2.5 สรุปงานวิจัยที่เกี่ยวข้อง

จากที่ได้กล่าวมาทั้งหมดนั้น จะเห็นได้ว่าการประยุกต์เทคนิควิธีการประมวลผลภาพทางการแพทย์ถูกนำมาใช้ช่วยงานทางด้านทันตกรรมอย่างมากมาย ซึ่งมีประโยชน์ในขั้นตอนการวางแผนและบำบัดรักษาผู้ป่วยทางทันตกรรม การจำลองการจัดฟันจะแสดงผลบนระบบสามมิติเพื่อช่วยให้ผู้เชี่ยวชาญสามารถวางแผนการรักษาได้อย่างถูกต้องในแต่ละช่วงของการรักษา เทคนิควิธีการประมวลผลภาพที่ถูกใช้สำหรับงานทันตกรรม ได้แก่ การลดทอนรายละเอียดของวัตถุ การตัดพื้นผิว และการสร้างแบบจำลองมวลสปริง การลดทอนรายละเอียดของวัตถุถูกนำมาใช้เพราะแบบจำลองสามมิติของฟันส่วนใหญ่มีรายละเอียดสูง ทำให้เกิดปัญหาในการแสดงผลและการจัดเก็บ ซึ่งได้เลือกใช้เทคนิควิธีการลบจุดออกจากโครงข่าย (vertex removal) โดยพิจารณาจากค่าระยะทางระหว่างจุดนั้นกับระนาบเฉลี่ย ถ้าค่าระยะทางนั้นมีค่าน้อย ๆ สามารถจะลบจุดนั้นออกจากโครงข่ายได้ เพราะจะไม่ทำให้ลักษณะสำคัญของวัตถุเสียไป เมื่อจุดถูกลบออกจะทำให้เกิดช่องว่างจำเป็นที่จะต้องสร้างสามเหลี่ยมใหม่ สำหรับกระบวนการตัดพื้นผิว ใช้สำหรับการแยกองค์ประกอบของฟันและเหงือกออกจากกันเพื่อจำลองการจัดเรียงฟัน เริ่มต้นจากการกำหนดจุดลงบนพื้นผิวของแบบจำลองฟันเพื่อสร้างเส้นทางการตัด สามารถเลือกส่วนที่เป็นองค์ประกอบของฟันได้ด้วยเทคนิควิธี Region growing ในกระบวนการจัดเรียงฟันจำเป็นต้องศึกษาการเปลี่ยนรูปร่างของเหงือกด้วยเทคนิควิธีการสร้างแบบจำลองมวลสปริง โดยกำหนดให้จุดบนโครงข่ายเป็นจุดมวลของสปริงและด้านของสามเหลี่ยมแต่ละรูปถูกแทนด้วยสปริง ค่าคงที่ต่าง ๆ ที่เกี่ยวข้องกับระบบมวลสปริงสามารถคำนวณได้จากสมการที่ได้กล่าวมาแล้ว และสามารถแก้สมการการเคลื่อนที่เพื่อหาตำแหน่งของจุดบนโครงข่ายได้ด้วยวิธีการหาปริพันธ์แบบชัดแจ้ง ซึ่งรายละเอียดของขั้นตอนต่าง ๆ และวิธีการดำเนินการวิจัยทั้งหมดนั้นจะได้กล่าวถึงในบทถัดไป

บทที่ 3
ระเบียบวิธีวิจัยและกรอบแนวคิด

งานวิจัยนี้มีจุดมุ่งหมายที่จะประยุกต์ใช้และพัฒนาขั้นตอนวิธีต่าง ๆ ของการประมวลผลภาพเพื่อช่วยงานด้านทันตกรรมจัดฟัน โดยการพัฒนาโปรแกรมจำลองการจัดฟัน (Orthodontic simulation) ซึ่งแสดงผลในสามมิติ เพื่อช่วยวางแผนและบำบัดรักษาทางทันตกรรมจัดฟัน ขั้นตอนวิธีที่นำมาพัฒนา ได้แก่ การลดทอนรายละเอียดของวัตถุ (Mesh Decimation) การตัดพื้นผิวของวัตถุ (Surface Cutting) และการสร้างแบบจำลองมวลสปริง (Mass-spring Model) รายละเอียดเนื้อหาในบทนี้ประกอบไปด้วยระเบียบวิธีวิจัย ปรากฏในหัวข้อที่ 3.1 คำอธิบายข้อมูลที่ใช้ในการทดสอบปรากฏในหัวข้อที่ 3.2 และหัวข้อที่ 3.3 เป็นรายละเอียดวิธีการที่ใช้ในการพัฒนาขั้นตอนวิธีแต่ละขั้นตอนวิธี

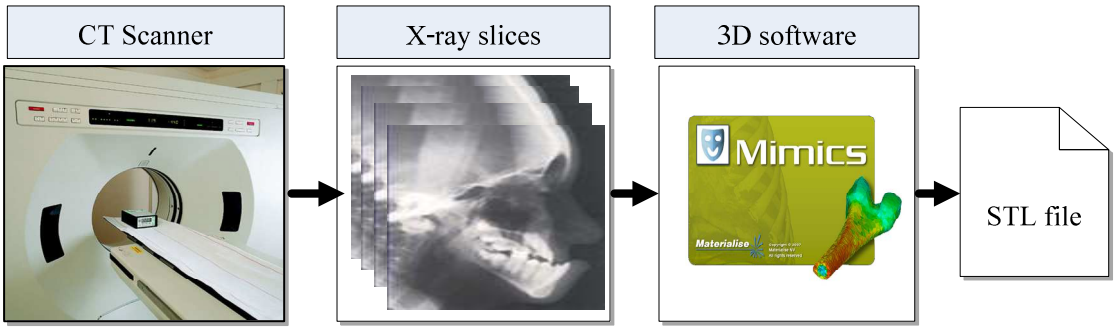
3.1 ระเบียบวิธีวิจัย

แนวทางการวิจัยของงานวิจัยนี้จะประกอบด้วยการออกแบบและพัฒนาขั้นตอนวิธีในการพัฒนาซอฟแวร์ทางการแพทย์ เพื่อช่วยงานด้านทันตกรรมจัดฟัน โดยมีกระบวนการดังนี้

1) ศึกษาค้นคว้ากระบวนการประยุกต์ใช้เทคนิควิธีการประมวลผลภาพ เพื่อช่วยงานด้านทันตกรรม โดยเทคนิควิธีที่เลือกใช้ในการวิจัยนี้ประกอบด้วย
   a) การลดทอนรายละเอียดของวัตถุ (Mesh Decimation)
   b) การตัดพื้นผิวของวัตถุ (Surface Cutting)
   c) การสร้างแบบจำลองมวลสปริง (Mass-spring Model)
2) ศึกษาค้นคว้าและรวบรวมงานวิจัยที่เกี่ยวข้องกับการลดทอนรายละเอียดของวัตถุ การตัดพื้นผิวของวัตถุ และการสร้างแบบจำลองมวลสปริง เพื่อเลือกขั้นตอนวิธีที่เหมาะสมสำหรับแต่ละกระบวนการ
3) ออกแบบโครงสร้างข้อมูล ที่เหมาะสมกับขั้นตอนวิธีต่าง ๆ
4) ออกแบบและพัฒนาขั้นตอนวิธีการลดทอนรายละเอียดของวัตถุ การตัดพื้นผิว และการสร้างแบบจำลองมวลสปริง
5) รวมขั้นตอนวิธีทั้งหมด และทดสอบประสิทธิภาพของการจำลองการจัดฟัน
6) วิเคราะห์และสรุปผลการวิจัย

3.2 ข้อมูลที่ใช้ในการวิจัย

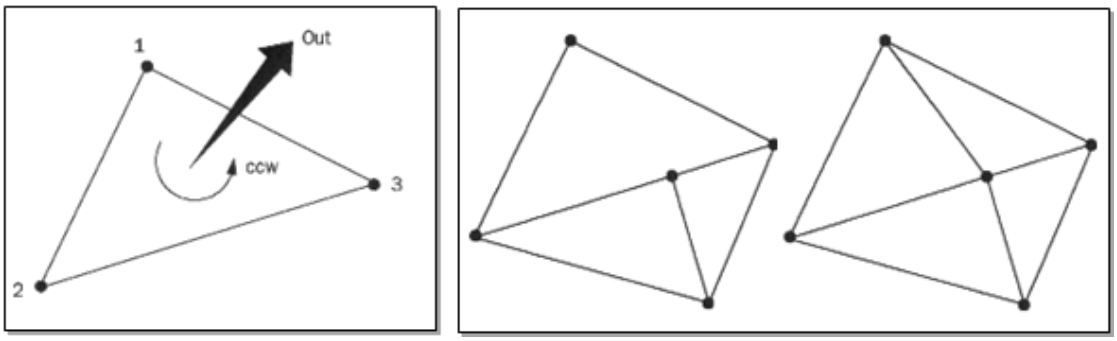
ข้อมูลทดสอบที่ใช้ในงานวิจัยนี้ได้รับการสนับสนุนจากศูนย์เทคโนโลยีทางทันตกรรมขั้นสูง (ADTEC) สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ (สวทช.) ประกอบด้วยข้อมูลของแบบจำลองขากรรไกรบนและขากรรไกรล่าง ซึ่งข้อมูลดังกล่าวอยู่ในรูปแบบของแฟ้มข้อมูล STL (Stereolithographic) binary กระบวนการได้มาของแฟ้มข้อมูล STL (Ma, Lin, and Chua, 2001) เริ่มต้นจากการนำภาพสไลด์เอกซ์เรย์ (DICOM) ซึ่งได้มาจากเครื่องซีทีแสกนเนอร์ (Computed Tomography Scanner) เข้าสู่โปรแกรมประมวลผลภาพสามมิติ จากนั้นโปรแกรมจะสร้างแฟ้มข้อมูล STL ออกมา แสดงดังรูปที่ 3.1



รูปที่ 3.1 แสดงกระบวนการสร้างแฟ้มข้อมูล STL

3.2.1 โครงสร้างแฟ้มข้อมูล STL

แฟ้มข้อมูล STL ถูกใช้เพื่ออธิบายรูปร่างลักษณะของวัตถุในสามมิติ ประกอบด้วยลิสต์ข้อมูลของสามเหลี่ยม (facet) สามเหลี่ยมแต่ละรูปจะประกอบด้วยจุด 3 จุด (vertex) และเวกเตอร์ทิศทาง (unit normal) การกำหนดทิศทางของสามเหลี่ยม (facet orientation) ซึ่งสามเหลี่ยมทั้งหมดต้องมีทิศทางแบบเดียวกัน สามารถกำหนดได้ 2 แบบคือ เข้า และออก การเรียงลำดับของจุดมุมของสามเหลี่ยมต้องมีลักษณะหมุนทวนเข็มนาฬิกา (counterclockwise) ตามกฎมือขวา (right-hand rule) ดังรูปที่ 3.2(a) สามเหลี่ยมแต่ละรูปต้องมีจุดที่ใช้ร่วมกันเพียง 2 จุดเท่านั้นตามกฎการเชื่อมโยงจุด (Vertex-to-vertex rule) ดังรูปที่ 3.2(b) รูปซ้ายมือ แฟ้มข้อมูล STL สามารถแบ่งออกได้เป็น 2 ประเภท คือ ASCII และ Binary สำหรับงานวิจัยนี้ใช้แฟ้มข้อมูลประเภท Binary เนื่องจากแฟ้มข้อมูลมีขนาดเล็กและสามารถอ่านและบันทึกข้อมูลได้อย่างรวดเร็ว เพราะไม่จำเป็นต้องเข้ารหัสสำหรับการบันทึกข้อมูล



(a) (b)

รูปที่ 3.2 แสดงทิศทางของสามเหลี่ยม (a) และกฎการเชื่อมโยงจุด (b)
ดูรายละเอียดในเนื้อหา (Burns, 1989)

รูปแบบของแฟ้มข้อมูล STL (ASCII) ประกอบด้วย ข้อมูลเวกเตอร์ทิศทาง และจุดมุมของสามเหลี่ยม มีดังนี้

solid *name*

facet normal *ni nj nk*

outer loop

Vertex *v1x v1y v1z*

Vertex *v2x v2y v2z*

Vertex *v3x v3y v3z*

endloop

endfacet

endsolid *name*

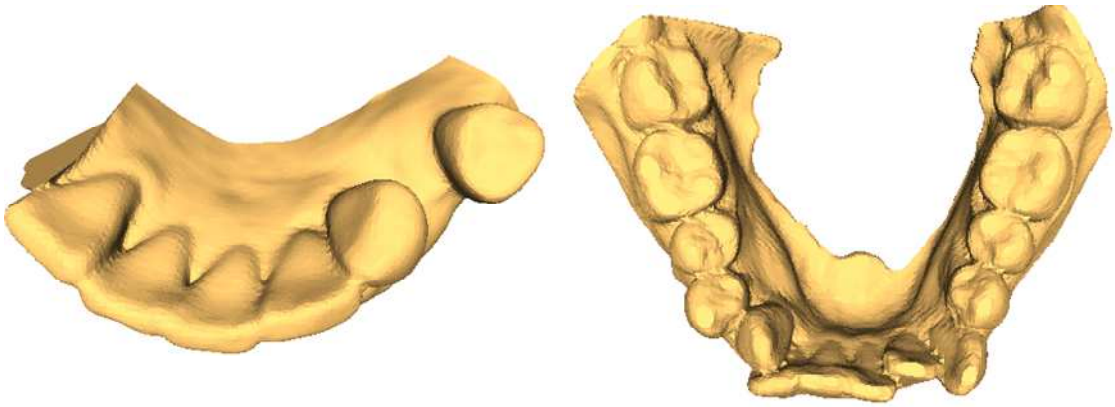
รูปแบบของแฟ้มข้อมูล STL (Binary) ประกอบด้วย ส่วนหัวของแฟ้มข้อมูล จำนวนของสามเหลี่ยมทั้งหมด เวกเตอร์ทิศทาง (normal) และจุดพิกัดบน 3 มิติของจุดมุมของสามเหลี่ยม โดยส่วนหัวของแฟ้มข้อมูล (header file) เป็นข้อมูลชนิด ASCII มีขนาด 80 ไบต์ ส่วนจำนวนของสามเหลี่ยม ซึ่งเป็นข้อมูลชนิดจำนวนเต็มบวก มีขนาด 4 ไบต์ สำหรับเวกเตอร์ทิศทาง เป็นตัวเลขทศนิยมมีขนาด 12 ไบต์ และจุดมุมของสามเหลี่ยมทั้งหมด เป็นตัวเลขทศนิยม โดยมีขนาดทั้งหมด 36 ไบต์ ซึ่งแต่ละจุดมีขนาด 12 ไบต์ แสดงดังตารางที่ 3.1

ตารางที่ 3.1 รูปแบบแฟ้มข้อมูล STL (Binary)

Bytes	Data type	Description
80	ASCII	Header. No data significance.
4	unsigned long integer	Number of facet in file
4	float	*i* for normal
4	float	*j*
4	float	*k*
4	float	*x* for vertex 1
4	float	*y*
4	float	*z*
4	float	*x* for vertex 2
4	float	*y*
4	float	*z*
4	float	*x* for vertex 3
4	float	*y*
4	float	*z*
2	unsigned integer	Attribute byte count.

3.2.2 ข้อมูลแบบจำลองขากรรไกรบนและขากรรไกรล่าง

ข้อมูลแบบจำลองบางส่วนของขากรรไกรบน (Maxilla model) ประกอบด้วยสามเหลี่ยมทั้งหมด 56,880 รูป ดังรูปที่ 3.3 (ขวา) ซึ่งมีฟันทั้งหมด 6 ซี่ ลักษณะของปัญหาของฟันคือ ฟันห่าง และข้อมูลแบบจำลองบางส่วนของขากรรไกรล่าง (Mandible model) ประกอบด้วยสามเหลี่ยมทั้งหมด 29,108 รูป ดังรูปที่ 3.3 (ซ้าย) มีจำนวนฟันทั้งหมด 14 ซี่ ฟันบ้างซี่มีลักษณะซ้อนทับกัน



รูปที่ 3.3 แบบจำลองของขาไกรกรรบน (ขวา) และแบบจำลองของขาไกรกรรล่าง (ซ้าย)

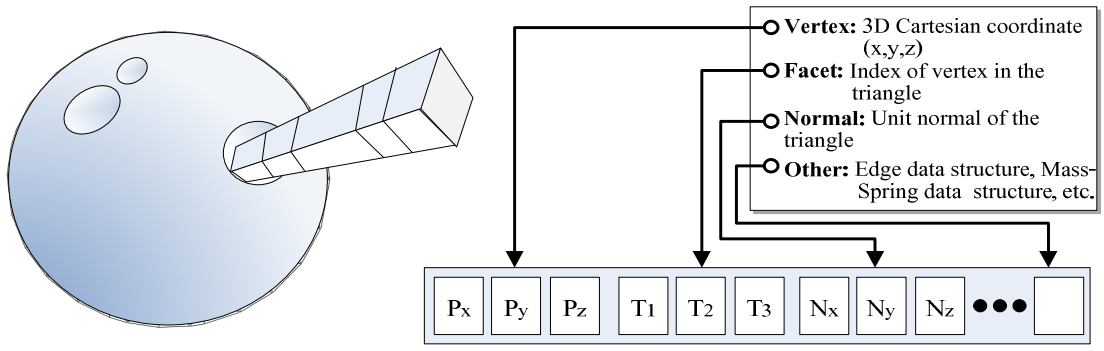
เมื่อทราบรูปแบบและโครงสร้างของข้อมูลที่ใช้สำหรับจำลองการจัดฟัน ซึ่งรูปแบบของข้อมูลหรือแบบจำลองฟันที่ใช้ในงานวิจัยนี้เป็นโครงข่ายแบบสามเหลี่ยม (Triangular mesh) กระบวนการเริ่มต้นก่อนเข้าสู่กระบวนการจำลองการจัดฟัน คือ การออกแบบโครงสร้างข้อมูล เพื่อใช้สำหรับการพัฒนาในขั้นตอนวิธีต่าง ๆ ซึ่งจะกล่าวถึงในหัวข้อถัดไป

3.3 ขั้นตอนวิธีการจำลองการจัดฟัน

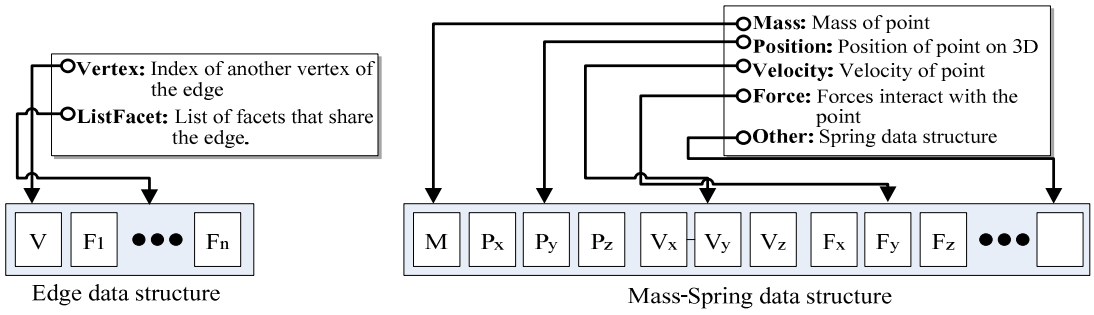
3.3.1 โครงสร้างข้อมูล

โครงข่ายของแบบจำลองฟันที่ใช้ในงานวิจัยนี้เป็นโครงข่ายแบบสามเหลี่ยม (Triangular mesh) ซึ่งประกอบด้วย สามเหลี่ยม (facet) จุด (vertex) และเวกเตอร์ทิศทางของสามเหลี่ยม (normal) สามารถออกแบบโครงสร้างข้อมูล ได้เป็น 2 ส่วน คือ โครงสร้างข้อมูลของจุด และโครงสร้างข้อมูลของสามเหลี่ยม โดยโครงสร้างข้อมูล ประกอบด้วย จุดพิกัดบนระบบ 3 มิติ (Px, Py, Pz) ซึ่งเป็นข้อมูลตัวเลขทศนิยม (float) ส่วนโครงสร้างข้อมูลของสามเหลี่ยม จะประกอบไปด้วย ดัชนี (index) ของจุดมุมของสามเหลี่ยม (T1, T2, T3) และเวกเตอร์ทิศทางของสามเหลี่ยม (Nx, Ny, Nz) แสดงดังรูปที่ 3.4 นอกจากโครงสร้างข้อมูลดังที่ได้กล่าวมาแล้วนั้น ยังมีโครงสร้างข้อมูลอื่น ๆ อีก เช่น โครงสร้างข้อมูลของด้านสามเหลี่ยม (Edge) สำหรับการสร้างกราฟของด้านสามเหลี่ยม (EdgeGraph) เพื่อค้นหาจุดบนขอบของโครงข่าย (Boundary vertex) และโครงสร้างข้อมูลของแบบจำลองมวลสปริง (Mass-Spring) สำหรับการจำลองการเปลี่ยนแปลงรูปร่างของเหงือกในขั้นตอนการจัดเรียงฟัน

โครงสร้างข้อมูลของด้านสามเหลี่ยม มีองค์ประกอบดังนี้ ดัชนีของจุดอื่นที่เป็นด้านของสามเหลี่ยม (V) ลิสต์ของสามเหลี่ยมที่ใช้ด้านร่วมกัน (F1, F2) ดังรูปที่ 3.5 (ซ้าย) จำนวนของสามเหลี่ยมมีค่ามากที่สุด 2 รูปเท่านั้น ถ้าจำนวนของสามเหลี่ยมมีค่าเท่ากับ 1 แสดงว่าด้านสามเหลี่ยมนี้เป็นขอบของโครงข่าย ส่วนโครงสร้างข้อมูลของแบบจำลองมวลสปริงนั้น สามารถแบ่งออกได้เป็น 2 ส่วน คือ โครงสร้างข้อมูลของจุดมวล (MassPoint) และโครงสร้างข้อมูลของสปริง (Spring) แสดงดังรูปที่ 3.5 (ขวา) องค์ประกอบของโครงสร้างข้อมูลของจุดมวล ได้แก่ มวล (M) พิกัดตำแหน่ง (Px, Py, Pz) ความเร็ว (Vx, Vy, Vz) และแรงทั้งหมดที่กระทำ (Fx, Fy, Fz) ส่วนโครงสร้างข้อมูลของสปริงจะคล้ายกับโครงสร้างข้อมูลของด้านสามเหลี่ยม เพราะสปริงเป็นด้านของสามเหลี่ยมนั่นเอง ซึ่งมีองค์ประกอบเพิ่มด้วย คือ ค่าคงที่สปริง (stiffness) และความยาวปกติของสปริง (rest length)



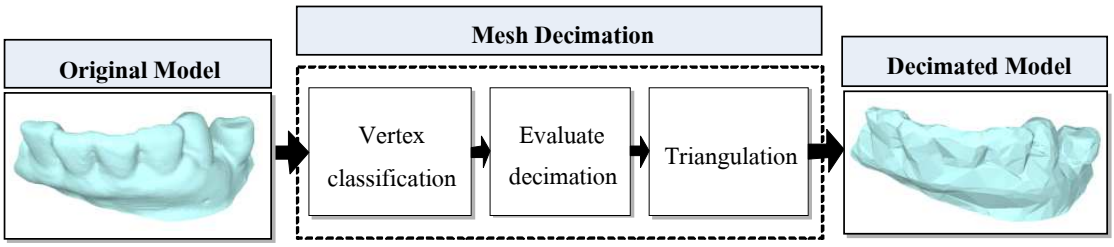
รูปที่ 3.4 แสดงโครงสร้างข้อมูลของโครงข่ายแบบสามเหลี่ยม



รูปที่ 3.5 แสดงโครงสร้างข้อมูลของด้านสามเหลี่ยม (ซ้าย) และโครงสร้างข้อมูลของแบบจำลองมวลสปริง (ขวา)

3.3.2 เทคนิคการลดทอนรายละเอียดของวัตถุ (Mesh Decimation)

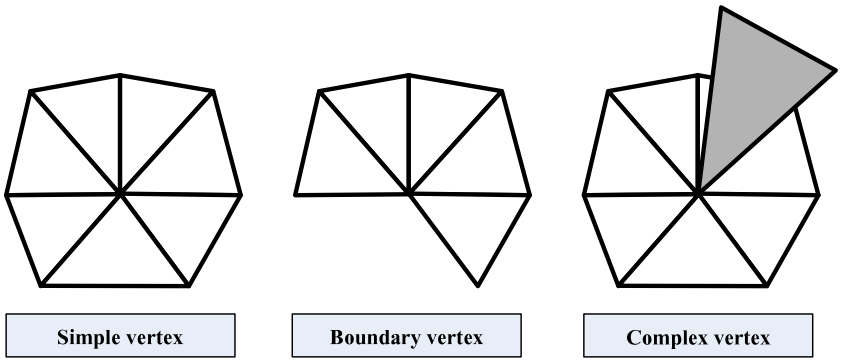
เนื่องจากข้อมูลที่ใช้ในงานวิจัยมีความละเอียดสูง เพื่อให้สิ้นเปลืองทรัพยากรและใช้เวลาประมวลผลน้อย จึงจำเป็นต้องลดทอนรายละเอียด ซึ่งเป็นกระบวนการลดจำนวนของสามเหลี่ยมภายในโครงข่าย งานวิจัยนี้ได้ประยุกต์ใช้เทคนิควิธีการลดทอนรายละเอียดของ Schroeder et al. (1992) สามารถแจกแจงขั้นตอนได้ 3 ขั้นตอน ดังรูปที่ 3.6 เริ่มต้นจากการอ่านข้อมูลของแบบจำลองฟัน จำแนกประเภทจุดบนโครงข่าย เพื่อเลือกประเภทจุดที่สามารถลดทอนรายละเอียดได้ ได้แก่ จุดอย่างง่าย (Simple vertex) และจุดขอบโครงข่าย (Boundary vertex) จากนั้นพิจารณาระยะห่างระหว่างจุดนั้นกับระนาบเฉลี่ย โดยเลือกจุดที่มีระยะห่างน้อยที่สุด เพื่อให้ไม่ทำลายลักษณะสำคัญของแบบจำลอง ผลลัพธ์ที่ได้คือโครงข่ายที่ถูกลดทอนรายละเอียดแล้ว สามารถบันทึกข้อมูลเพื่อนำไปใช้ต่อได้



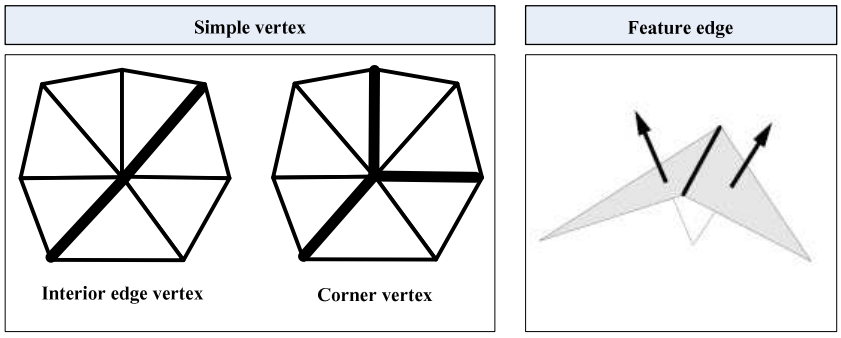
รูปที่ 3.6 แสดงกระบวนการลดทอนรายละเอียดของวัตถุ

1. การจำแนกประเภทจุดบนโครงข่าย (Vertex classification)

จำแนกประเภทของจุด (Vertex) ที่อยู่บนโครงข่ายสามเหลี่ยม สามารถแบ่งประเภทได้ 3ประเภท คือ จุดอย่างง่าย (Simple vertex), จุดเส้นขอบ (Boundary vertex) และ จุดซับซ้อน (Complex vertex) ดังรูปที่ 3.7 จุดอย่างง่าย คือ จุดที่ใช้สามเหลี่ยมสองรูปร่วมกันกับจุดข้างเคียง โดยที่จุดนั้นต้องไม่ใช่จุดที่อยู่บนเส้นขอบหรือมุม และจุดอย่างง่าย สามารถแบ่งออกเป็น 2 ประเภท คือ จุดขอบภายใน (Interior edge vertex) และ จุดมุม (Corner vertex) ดังแสดงในรูปที่ 3.8(a) หลักเกณฑ์ที่ใช้ในการแบ่งคือ มุมแหลมที่เกิดจากการกระทำของระนาบของสามเหลี่ยมสองรูป (feature edges) ดังแสดงในรูปที่ 3.8(b) โดยที่จุดขอบภายใน คือ จุดที่มี 2 feature edges ส่วนจุดมุมมี 1 หรือ 3 feature edges สำหรับจุดซับซ้อน จะมีจุดข้างเคียงที่ไม่ได้ใช้สามเหลี่ยมสองรูปร่วมกัน และจุดนั้นต้องไม่ใช่จุดที่อยู่บนเส้นขอบด้วย ส่วนจุดเส้นขอบ คือ จุดที่อยู่บนเส้นขอบของโครงข่าย ซึ่งจะมีจุดข้างเคียงที่ไม่ได้ใช้สามเหลี่ยมสองรูปร่วมกัน ซึ่งประเภทของจุดที่ไม่สามารถดึงออกจากโครงข่ายได้ คือ จุดมุม และจุดซับซ้อน



รูปที่ 3.7 ประเภทจุดบนโครงข่ายสามเหลี่ยม

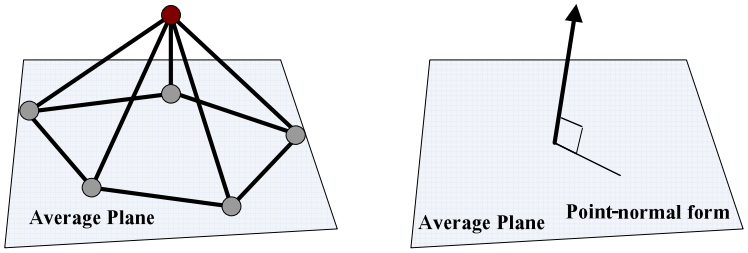


(a) (b)

รูปที่ 3.8 ประเภทจุดอย่างง่าย (Simple vertex) (a) และFeature edge (b) ดูรายละเอียดในเนื้อหา

2. การพิจารณาจุดที่เหมาะสมกับการลดทอนรายละเอียด (Evaluate decimation)

เกณฑ์การพิจารณาเลือกจุดที่สามารถดึงออกจากโครงข่ายได้ พิจารณาจากจุดที่ไม่ใช่จุดมุมและไม่ใช่จุดซับซ้อน และพิจารณาระยะห่างระหว่างจุดนั้นกับระนาบเฉลี่ย โดยที่ระนาบเฉลี่ยจะอยู่ในรูปแบบของจุดและเวกเตอร์ตั้งฉาก (point-normal form) (Hill, 2001) ดังแสดงในรูปที่ 3.9



รูปที่ 3.9 ระนาบเฉลี่ยในรูปแบบจุดและเวกเตอร์ตั้งฉาก

ขั้นตอนในการพิจารณาเลือกจุดที่จะถูกดึงออก มีดังต่อไปนี้

a) หาจำนวนจุดที่จะถูกดึงออกโดยพิจารณาจากเปอร์เซ็นต์การลดทอนรายละเอียด โดยผู้ใช้จะกำหนดเปอร์เซ็นต์การลดทอนรายละเอียด การคำนวณหาจำนวนจุดที่จะถูกดึงออกจากโครงข่าย (D) สามารถคำนวณได้จากผลคูณของเปอร์เซ็นต์การลดทอนรายละเอียด (d) กับจำนวนจุดทั้งหมดบนโครงข่าย (V) ในกรณีที่ผลคูณดังกล่าวมีค่ามากกว่าหรือเท่ากับผลต่างของจำนวนจุดทั้งหมดกับจำนวนจุดมุมทั้งหมดบนโครงข่าย (C) จำนวนจุดที่ถูกลดทอนรายละเอียดจะมีค่าเท่ากับจำนวนจุดทั้งหมดลบด้วยจำนวนจุดมุมทั้งหมด ดังสมการที่ 3.1

$$D = \begin{cases} d*V & \text{if } d*V < (V-C) \text{ else} \\ V-C \end{cases} \tag{3.1}$$

เมื่อ D คือ จำนวนของจุดที่จะถูกดึงออกจากโครงข่าย

d คือ เปอร์เซ็นต์การลดทอนรายละเอียด

V คือ จำนวนจุดทั้งหมดบนโครงข่าย

C คือ จุดมุมทั้งหมดบนโครงข่าย

b) หาระนาบเฉลี่ย ซึ่งจะพิจารณาจากจุดที่อยู่ล้อมรอบจุดนั้น โดยใช้เวกเตอร์ตั้งฉากเฉลี่ย (**n**) และจุดเฉลี่ย (x, y, z) เป็นตัวแทนของระนาบเฉลี่ย สามารถคำนวณได้ดังสมการที่ 3.2 โดยที่เวกเตอร์ตั้งฉากเฉลี่ยต้องทำให้เป็นเวกเตอร์หนึ่งหน่วย (unit vector)

$$\bar{\mathbf{U}} = \sum_{i=0}^{i \le N} \mathbf{U}_i / N \qquad (3.2)$$

เมื่อ **U** คือ องค์ประกอบของ Cartesian coordinate **P**(x, y, z) หรือเวกเตอร์ทิศทาง **n**

N คือ จำนวนของจุดเคียงข้าง

c) คำนวณหาระยะทางจากจุดกับระนาบเฉลี่ย สามารถคำนวณจากสมการที่ 3.3

$$d = \mathbf{v} \cdot \mathbf{n} - \mathbf{p} \cdot \mathbf{n} \qquad (3.3)$$

เมื่อ **v** คือ เวกเตอร์ตำแหน่งของจุดที่พิจารณา

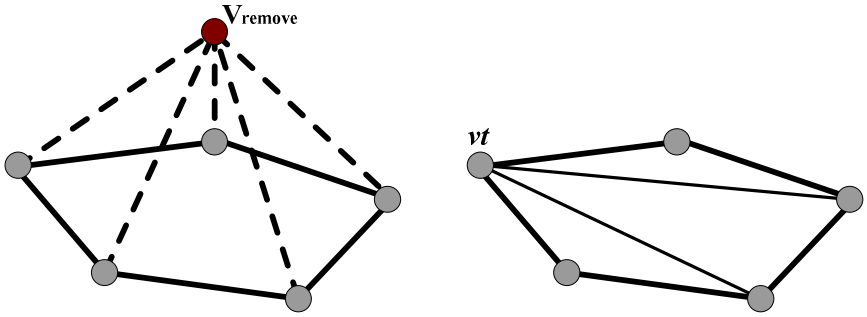
n คือ เวกเตอร์ตั้งฉากที่จุดเฉลี่ยบนระนาบ

p คือ เวกเตอร์ตำแหน่งของจุดเฉลี่ยบนระนาบ

d) เลือกจุดที่จะดึงออกจากโครงข่าย ซึ่งมีระยะทางระหว่างจุดกับระนาบเฉลี่ยน้อยที่สุด เมื่อดึงจุดออกทำให้ช่องว่าง ดังแสดงดังรูปที่ 3.10(a) จำเป็นต้องสร้างสามเหลี่ยมขึ้นมาใหม่ ทุกครั้งที่จุดถูกดึงออกจากโครงข่าย จำเป็นต้องคำนวณหาระนาบเฉลี่ยและระยะห่างใหม่

3. การสร้างสามเหลี่ยมใหม่ (Triangulation)

สร้างรูปสามเหลี่ยมขึ้นมาใหม่ในส่วนที่ถูกลดขนาด ขั้นตอนการสร้างสามเหลี่ยมเริ่มต้นจากการเลือกจุดที่รอบล้อมจุดที่ถูกดึงออกมา 1 จุด ทำการสร้างรูปสามเหลี่ยมใหม่โดยจุดนี้เป็นจุดอ้างอิง (vt) จะได้ผลลัพธ์ดังรูปที่ 3.10(b) จากรูปตัวอย่างการลดทอนรายละเอียดของวัตถุจะเห็นได้ว่าสามารถลดจำนวนสามเหลี่ยมจากเดิม 5 รูปให้เหลือเพียง 3 รูปได้

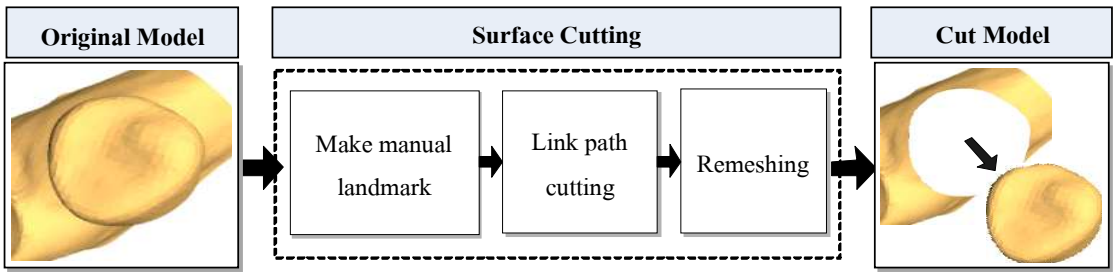


รูปที่ 3.10 แสดงการสร้างสามเหลี่ยมใหม่ (b) จากช่องว่างที่เกิดจากการดึงจุดออก (a)

งานวิจัยนี้ได้นำเสนอขั้นตอนวิธีการลดทอนรายละเอียดของวัตถุ ซึ่งเริ่มจากจำแนกประเภทของจุด เพื่อแยกประเภทจุดที่สามารถลบออกจากโครงข่ายได้ ซึ่งได้แก่ จุดขอบภายใน (Interior edge vertex) และจุดขอบของโครงข่าย (Boundary vertex) เมื่อจำแนกประเภทจุดแล้ว จากนั้นจะพิจารณาระยะห่างระหว่างจุดกับระนาบเฉลี่ย ถ้าระยะห่างดังกล่าวมีค่าน้อยที่สุดสำหรับการพิจารณาแต่ละรอบ ก็สามารถลบจุดนั้นออกจากโครงข่ายได้ สาเหตุที่เลือกระยะห่างที่มีค่าน้อย ๆ เพราะต้องการรักษาลักษณะที่สำคัญของวัตถุนั้นไว้ หลังจากขั้นตอนการลบจุดออกแล้ว จะต้องสร้างสามเหลี่ยมขึ้นใหม่ โดยสร้างได้จากการกำหนดจุดอ้างอิงมาหนึ่งจุด ขั้นตอนวิธีการลดทอนรายละเอียดนี้ไม่ได้คำนึงถึงค่าความผิดพลาดทางเรขาคณิตของวัตถุที่ถูกลดทอนรายละเอียดมากนัก เนื่องจากวัตถุประสงค์ของการลดทอนรายละเอียดสำหรับงานวิจัยนี้คือ เพื่อลดเวลาการแสดงผลและการประมวลผลเท่านั้น

3.33 เทคนิคการตัดพื้นผิว (Surface Cutting)

เป็นขั้นตอนการแยกองค์ประกอบของวัตถุออกจากกัน ซึ่งในงานวิจัยนี้จะเป็นการแยกฟันและเหงือกออกจากกัน เพราะการจำลองการจัดฟันจะจัดเรียงเฉพาะตำแหน่งของฟันเท่านั้น สามารถแจกแจงขั้นตอนได้ 3 ขั้นตอน คือ การกำหนดจุดบนพื้นผิว (Make manual landmark) การเชื่อมโยงจุดตัดที่ตัดผ่านโครงข่าย (Link path cutting) สองขั้นตอนแรกทำเพื่อกำหนดเส้นทางการตัดของฟันแต่ละซี่ ขั้นตอนสุดท้ายคือ การปรับปรุงโครงข่ายหรือการสร้างสามเหลี่ยมใหม่จากเส้นทางที่ตัดผ่าน (Remeshing) เพื่อแยกองค์ประกอบของฟันออกจากเหงือก ดังรูปที่ 3.11



รูปที่ 3.11 แสดงกระบวนการตัดพื้นผิว

1. การกำหนดจุดตัดบนพื้นผิว (Make manual landmark)

เป็นขั้นตอนการเลือกฟันซี่ที่ต้องการตัด โดยเริ่มต้นจากกำหนดจุดบนพื้นผิวของแบบจำลองฟัน เป็นกำหนดจุดรอบแบบจำลองฟัน ซึ่งผู้ใช้จำเป็นต้องคลิกเมาท์เพื่อกำหนดจุดตำแหน่งที่ต้องการผ่านทางจอแสดงผล การหาตำแหน่งของจุดที่ผู้ใช้กำหนดให้ตรงกับตำแหน่งของจุดบนแบบจำลองฟัน ซึ่งเป็นการแปลงตำแหน่งบนสองมิติ (จอแสดงผล) ไปเป็นตำแหน่งบนสามมิติ (แบบจำลองฟัน) โดยเทคนิคการเลือกจุดบนวัตถุ 3 มิติ (Picking object in 3D) (OpenGL Architecture Review Board, 2004) ซึ่งใช้ฟังก์ชัน glUnProject ของไลบารี OpenGL® เป็นฟังก์ชันที่ใช้ในการแปลงพิกัดของจอภาพไปเป็นพิกัดของ OpenGL® นั่นหมายความว่าเราสามารถรับตำแหน่งของเมาส์บนหน้าจอของ OpenGL® และใช้ในการหาค่าพิกัด (x, y, z) จากบริเวณที่เราคลิกเมาส์ได้ โดยใช้ค่าต่าง ๆ ที่เกี่ยวข้องดังนี้

a) Viewport origin and Extent - สิ่งแรกที่เราต้องการทราบ คือ viewport ปัจจุบัน ข้อมูลที่เราได้จะเป็นค่าพิกัด (x, y) เริ่มต้นของ GL viewport พร้อมกับค่าความกว้างและความสูงของ viewport ด้วย

b) The Modelview Matrix - หลังจากที่เราได้ข้อมูลของ viewport แล้ว เรายังต้องการข้อมูลของ ModelView ด้วย โดยที่ ModelView Matrix จะบอกให้เรารู้ว่าจุดต่าง ๆ บน OpenGL® ถูกแปลง (transform) ไปเป็นระบบพิกัดที่สายตาเราเห็น (eye coordinates)

c) The Projection Matrix – ข้อมูลต่อมาที่เราต้องการทราบ คือ Projection Matrix จะทำให้เราทราบว่า จุดต่าง ๆ ในระบบพิกัดที่สายตาเราเห็นถูกแปลงไปเป็น clip coordinates

d) The Windows Screen Coordinates – หลังจากที่เราทราบค่าต่าง ๆ เกือบหมดแล้ว ข้อมูลสุดท้ายที่เราต้องการทราบคือ พิกัดของหน้าจอ Windows ที่เรา

คลิกเมาส์ เมื่อเราทราบค่าพิกัดที่เมาส์คลิกแล้ว (screenX, screenY) เนื่องจากระบบพิกัดของหน้าจอ Windows จะเริ่มต้นพิกัด (0,0) ที่มุมซ้ายบนของจอภาพ ขณะที่พิกัดของ OpenGL® จะเริ่มต้นที่มุมซ้ายล่าง ดังนั้นจึงจำเป็นต้องแปลงจากระบบพิกัดบนจอภาพเป็นพิกัดของ OpenGL (winX, winY) ได้ดังนี้

```
winX = (double) screenX;
winY =  viewport [3] -(double) screenY;
```

หลังจากที่เราทราบข้อมูลที่เราต้องการหมดแล้ว ขั้นสุดท้ายก็เป็นการคำนวณหาค่าพิกัด (X, Y, Z) ที่เราต้องการ โดยใช้ฟังก์ชั่น glUnProject () ซึ่งการทำงานของฟังก์ชันนี้สามารถคำนวณได้จากสมการที่ 3.4

$$\begin{bmatrix} X \\ Y \\ Z \\ W \end{bmatrix} = INV(P \cdot M) \cdot \begin{bmatrix} \frac{2(winX - x)}{width} - 1 \\ \frac{2(winY - y)}{height} - 1 \\ 2(winZ) - 1 \\ 1 \end{bmatrix} \tag{3.4}$$

เมื่อ INV คือ ฟังก์ชันเมตริกผกผัน (inverse matrix)

P คือ projection matrix

M คือ modelview matrix

$width$ คือ ความกว้างของ viewport

$height$ คือ ความสูงของ viewport

เมื่อกำหนดจุดรอบแบบจำลองฟันแล้ว จำเป็นต้องสร้างเส้นทางการตัด ซึ่งเส้นทางการตัดนั้นเป็นเส้นโค้ง โดยใช้เทคนิค natural cubic spline (Hill, 2001) กำหนดให้จุดที่กำหนด (landmark) ล้อมรอบแบบจำลองฟันเป็นจุดควบคุม (control point) ของเส้นโค้ง สามารถคำนวณหาจุดบนเส้นโค้งระหว่างจุดควบคุมได้จากสมการที่ 3.5 และสมการที่ 3.6 เพื่อคำนวณหาพิกัดของจุดบนเส้นโค้ง โดยสมการที่ 3.5 เป็นเมตริกซ์สมการของ natural cubic spline โดยที่กำหนดให้ f_0'' และ

f_n'' มีค่าเท่ากับศูนย์ ใช้สำหรับคำนวณหาค่าอนุพันธ์อันดับสองของฟังก์ชันเส้นโค้ง สำหรับการคำนวณหาพิกัดของเส้นโค้งระหว่างจุดควบคุมดังสมการที่ 3.6

$$\begin{bmatrix} 4 & 1 & & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 \\ & 1 & \ddots & 1 & \\ 0 & 0 & 1 & 4 & 1 \\ 0 & 0 & & 1 & 4 \end{bmatrix} \begin{bmatrix} f_1'' \\ f_2'' \\ \vdots \\ f_{n-2}'' \\ f_{n-1}'' \end{bmatrix} = 6 \begin{bmatrix} f_2 - 2f_1 + f_0 \\ f_3 - 2f_2 + f_1 \\ \vdots \\ f_{n-1} - 2f_{n-2} + f_{n-3} \\ f_n - 2f_{n-1} + f_{n-2} \end{bmatrix} \tag{3.5}$$

เมื่อ f คือ ฟังก์ชันของเส้นโค้ง

f'' คือ อนุพันธ์อันดับสองของฟังก์ชันของเส้นโค้ง

n คือ จำนวนจุดบนเส้นโค้งในแต่ละช่วงระหว่างจุดควบคุม

$$f = Af_j + Bf_{j+1} + Cf_j'' + Df_{j+1}'' \tag{3.6}$$

$$A = x_{j+1} - x$$

$$B = 1 - A$$

$$C = (A^3 - A)/6$$

$$D = (B^3 - B)/6$$

ขั้นตอนการคำนวณหาจุดบนเส้นโค้งระหว่างจุดควบคุม ทำให้เกิดปัญหาสำหรับการสร้างเส้นทางการตัดพื้นผิว คือ จุดบนเส้นโค้งหรือจุด landmark ไม่ได้อยู่บนพื้นผิวทั้งหมด ดังแสดงในรูปที่ 3.13(a) สามารถแก้ปัญหาดังกล่าว ได้ตามขั้นตอนดังนี้

a) ฉายจุด landmark ทั้งหมดลงบนพื้นผิวของวัตถุ โดยใช้วิธีการของ Lagrange (Thomas et al., 2006) ซึ่งเป็นการแปลงระบบพิกัด 3 มิติของจุด landmark แต่ละจุดไปเป็นระบบพิกัด 2 มิติ นิยามสมการดังสมการที่ 3.3 ซึ่งคำนวณได้จากจุดของสามเหลี่ยม (x1,y1,z1), (x2,y2,z2) และ (x3,y3,z3) จากนั้นคำนวณหาจุดพิกัดที่ฉายจุดลงบนระนาบได้ดังสมการที่ 3.8

$$Ax_3 + By_3 + Cz_3 + D = 0 \tag{3.3}$$

$$A = y1(z2 - z3) + y2(z3 - z1) + y3(z1 - z2)$$
$$B = z1(x2 - x3) + z2(x3 - x1) + z3(x1 - x2)$$
$$C = x1(y2 - y3) + x2(y3 - y1) + x3(y1 - y2)$$
$$D = -[x1(y2z3 - y3z2) + x2(y3z1 - y1z3) + x3(y1z2 - y2z1)]$$

$$\begin{cases} \lambda = -\dfrac{D + Ax_3 + By_3 + Cz_3}{A^2 + B^2 + C^2} \\ x_2 = \lambda x_3 \\ y_2 = \lambda y_3 \end{cases} \tag{3.4}$$

เมื่อ x_3, y_3, z_3 คือ จุดพิกัดบน 3 มิติ

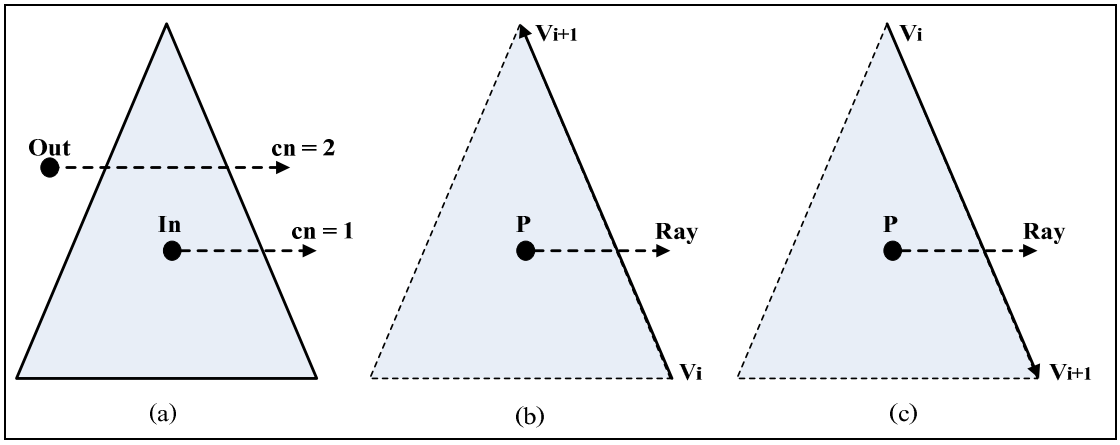
x_2, y_2 คือ จุดพิกัดบน 2 มิติ

λ คือ ตัวดำเนินการของลากรอง (Lagrange multiplier)

b) พิจารณาว่าจุด landmark ที่ถูกฉายนั้นอยู่ภายในสามเหลี่ยมของโครงข่ายหรือไม่ สามารถพิจารณาได้จากเทคนิควิธี Crossing test ของทฤษฎีเส้นโค้งของจอร์แดน (Foley et al., 1996) การทดสอบดังกล่าวทดสอบจากเส้นตรงในแนวนอนจากจุดทดสอบไปยังด้านของสามเหลี่ยม พิจารณาจำนวนด้านของสามเหลี่ยมที่เส้นตรงตัดผ่าน (Crossing number, cn) ดังรูปที่ 3.12(a) โดยที่เส้นตรงดังกล่าวสามารถตัดผ่านได้ทั้งด้านที่ชี้ขึ้น (upward edge) และด้านที่ชี้ลง (downward edge) ดังรูปที่ 3.12(b) และ 3.12(c) โดยรายละเอียดของขั้นตอนวิธี Crossing test มีดังนี้

```
input  : Point P and Point of polygon V[]
output : Point is inside when "crossing number" is odd;
         otherwise, when it is even, point is outside
begin
cn := 0;  // initialization the crossing number counter
// loop through all edges of the polygon
for each edge E := (V[i],V[i+1]) in polygon then
   if (E crosses upward edge || E crosses downward edge) then
```

```
            if (P.x < x_intersect of E[i] with y=P.y)  then
                ++cn;   // a valid crossing to the right of P.x
            end if
        end if
    end for
    return cn%2;    //if 0 is outside and 1 is inside
    end
```



รูปที่ 3.12 เทคนิค crossing test (a), upward edge (b) และ downward edge (c)
ดูรายละเอียดในเนื้อหา

c) ถ้าจุด landmark นั้นไม่อยู่ภายในสามเหลี่ยมใด ๆ เลย แสดงว่าจุดนั้นอยู่บนด้านของรูปสามเหลี่ยม ซึ่งกรณีดังกล่าวต้องพิจารณาบนระบบพิกัด 3 มิติ แล้ววิเคราะห์ว่าจุด landmark นั้นอยู่ใกล้กับด้านของรูปสามเหลี่ยมใดมากที่สุด โดยเริ่มต้นจากการพิจารณาว่าเวกเตอร์ที่ฉายของจุดทดสอบนั้นอยู่บนด้านของสามเหลี่ยมหรือไม่ สามารถคำนวณได้สมการที่ 3.5 โดยใช้จุดของสามเหลี่ยม (x1,y1,z1), (x2,y2,z2) และ (x3,y3,z3) ในการคำนวณ หลักเกณฑ์พิจารณาจุดที่อยู่บนด้านเหลี่ยม สามารถพิจารณาจากค่าอัตราส่วนเฉลี่ยของเวกเตอร์ (u) ซึ่งถ้าค่าอัตราส่วนเฉลี่ยนั้นมีค่าอยู่ในช่วง 0 ถึง 1 แสดงว่าเวกเตอร์ฉายของจุดทดสอบนั้นอยู่บนด้านของสามเหลี่ยม และสามารถคำนวณหาจุดที่เวกเตอร์ฉายลงบนด้านของสามเหลี่ยมได้จากสามการที่ 3.6 จากนั้นคำนวณหาค่า

ระยะทางระหว่างจุดทดสอบนั้นกับจุดที่อยู่บนด้านของสามเหลี่ยม เลือกด้านของสามเหลี่ยมที่มีค่าระยะทางดังกล่าวน้อยที่สุด

$$u = \frac{(x_3 - x_1)(x_2 - x_1) + (y_3 - y_1)(y_2 - y_1) + (z_3 - z_1)(z_2 - z_1)}{\left\|\mathbf{P}_2 - \mathbf{P}_1\right\|^2} \tag{3.5}$$

$$\mathbf{r} = \mathbf{r}_1 + u(\mathbf{r}_2 - \mathbf{r}_1) \tag{3.6}$$

เมื่อ u คือ อัตราส่วนเฉลี่ยของเวกเตอร์

$\mathbf{P}$ คือ พิกัดจุดของด้านสามเหลี่ยม

$\mathbf{r}$ คือ องค์ประกอบของ Cartesian coordinate (x, y, z)

d) กรณีที่จุด landmark นั้นอยู่ภายในสามเหลี่ยม ให้แปลงจุดที่ฉายบนระบบพิกัด 2 มิติ เป็นจุดบนระบบพิกัด 3 มิติ โดยใช้วิธี Barycentric (Bradley, 2007) ดังสมการที่ 3.7 ส่วนกรณีที่จุด landmark นั้นอยู่บนขอบของรูปสามเหลี่ยมก็เปลี่ยนพิกัดของจุด landmark เดิม เป็นจุดพิกัดที่อยู่บนขอบของรูปเหลี่ยมนั้น จากรูปที่ 3.13(a) พบว่ามีจุด landmark บางส่วนที่อยู่ใต้พื้นผิว และผลลัพธ์ที่ได้จากการแก้ไขปัญหาจุด landmark ไม่อยู่บนพื้นผิวของวัตถุ แสดงได้ดังรูปที่ 3.13(b) ซึ่งจะเห็นได้ว่าจุด landmark ทั้งหมดจะอยู่บนพื้นผิวของแบบจำลองฟัน และสาเหตุที่เกิดช่องว่างระหว่างจุดนั้นเกิดการรวมจุดกันไปยังจุดมุมของสามเหลี่ยม เนื่องจากว่าจุด landmark เดิมที่ไม่ได้อยู่บนพื้นผิวอยู่ใกล้กับจุดมุมของสามเหลี่ยม

$$w_i = \frac{a_i}{A}, \sum_i w_i = 1$$

$$\mathbf{r} = w_1\mathbf{r}_1 + w_2\mathbf{r}_2 + w_3\mathbf{r}_3$$

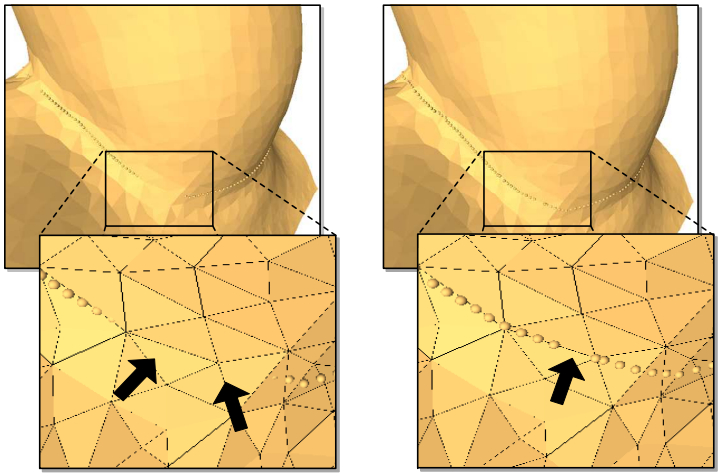
(3.7)

เมื่อ w คือ ค่าเฉลี่ยถ่วงน้ำหนักของพื้นที่สามเหลี่ยมภายในแต่ละรูป

a คือ พื้นที่ของสามเหลี่ยมภายในแต่ละรูป

A คือ พื้นที่ของสามเหลี่ยมที่จุดกำหนดอยู่ภายใน

$\mathbf{r}$ คือ องค์ประกอบของ Cartesian coordinate (x, y, z)



(a) (b)

รูปที่ 3.13 แสดงจุด Landmark ที่ไม่อยู่บนพื้นผิว (a) และผลลัพธ์ (b)

ขั้นตอนการกำหนดจุดตัดบนพื้นผิว สามารถแบ่งออกเป็น 2 กรณี คือ กรณีฟันซี่ที่จะตัดไม่ได้ติดกับฟันซี่ที่ถูกตัด และกรณีที่ฟันซี่ต้องการจะตัดอยู่ติดกับฟันซี่ที่ถูกตัดไปแล้ว สำหรับกรณีที่ฟันซี่ต้องการจะตัดอยู่ติดกับฟันซี่ที่ถูกตัดไปแล้ว ต้องกำหนดจุดตัดที่แตกต่างกับกรณีฟันซี่ที่จะตัดไม่ได้ติดกับฟันซี่ที่ถูกตัด คือ เริ่มจากหาจุดขอบของฟันซี่ที่ถูกตัดแล้ว จากนั้นกำหนดจุดเริ่มต้นและจุดสิ้นสุด ซึ่งต้องเป็นจุดที่อยู่บนขอบของฟันซี่ที่ถูกตัด ดังรูปที่ 3.14 ขั้นตอนการค้นหาจุด landmark จากจุดเริ่มต้นไปยังจุดสุดท้ายสามารถดำเนินการตามขั้นตอนวิธีที่ได้กล่าวข้างต้น สำหรับจุด landmark ที่อยู่ระหว่างจุดสุดท้าย และจุดเริ่มต้น ซึ่งเป็นจุดที่อยู่บนขอบของฟันซี่ที่อยู่ติดกัน สามารถค้นหาด้วยขั้นตอนวิธีของ Dijkstra (Collins and McMillan, 1992) ซึ่งมีรายละเอียดดังรหัสเทียมตามที่ได้แสดงไว้ โดยที่ *source* คือ จุด landmark เริ่มต้น และค่าระยะทางที่ใกล้ที่สุดระหว่างจุดเริ่มต้นไปยังจุดอื่น ๆ บนโครงข่ายจะถูกเก็บไว้ที่ *previous*[]

```
function  Dijkstra (EdgeGraph, source)
  begin
      for each vertex V in EdgeGraph then
      dist[V] := infinity            //unknown distance from source to v
      previous[V] := undefined
      end for
      dist[source] := 0              //distance from source to source
      Q := copy (EdgeGraph)  //all nodes in edge graph are
```

```
    while Q is not empty then  //unoptimized – thus are in Q
      U := extract_min (Q)    //remove and return best vertex from nodes
      for each neighbor V of U then
          if dist[U] + length (U,V) < dist[V] then
                dist[V] := dist[U] + length (U,V)
                previous[V] := U
          end if
      end for
    end while
  return previous[]
end
```

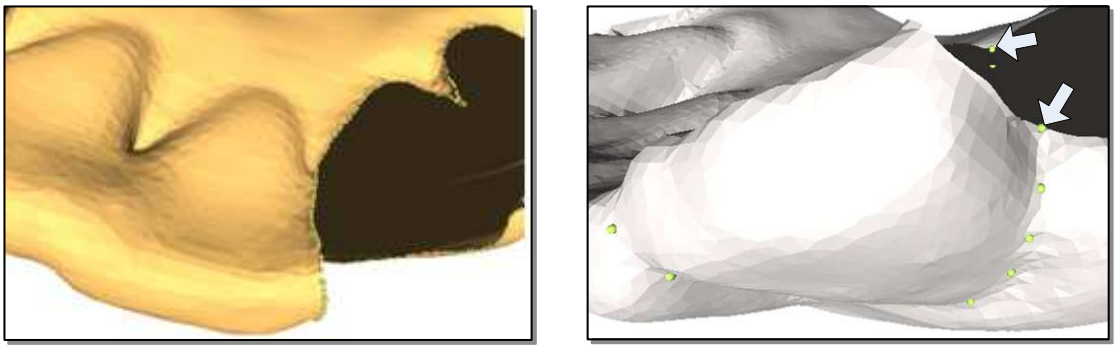
ขั้นตอนถัดไปคือ การค้นหาจุด (*S*) ที่อยู่ระหว่างจุด landmark เริ่มต้น (*source*) และสุดท้าย (*target*) ซึ่งทำให้ระยะทางระหว่างจุดสั้นที่สุด ซึ่งมีรายละเอียดดังนี้

```
S := empty sequence
U := target
while define previous[U] then
    U := previous[U]   //insert U at beginning of S
end while
```

ส่วนขั้นตอนสร้างกราฟของด้านสามเหลี่ยม ซึ่งใช้โครงสร้างข้อมูลประเภทลิสต์ในการสร้างกราฟ ลิสต์จะเก็บข้อมูลด้านสามเหลี่ยม (Edge) การเริ่มต้นจากค้นหาด้านต่าง ๆ ของสามเหลี่ยมแต่ล่ะรูปภายในโครงข่าย ซึ่งมีรายละเอียดดังนี้

```
function   EdgeGraph (FaceList)
begin
  list<Edge*> *m_aedgeGraph := new
  list<Edge*>[nvertices]  //initial list of edge
   for each face F in FaceList then
      for each edge E := (V[i],V[i+1]) in F then
```

```
        if m_aedgeGraph[V[i]] has not V[i+1] in their list then
                Edge *pnew0, *pnew1;
                pnew0->appendEdge(V[i+1], F)    //add factets to this node
                pnew1->appendEdge(V[i], F)      //add factets to its pair too
                m_aedgeGraph[V[i]].push_back(pnew0)     //append edge
                m_aedgeGraph[V[i+1]].push_back(pnew1)
        else //otherwise (this edge's already exist)
                E->m_lstfacets.pus_back(F)          //add factets to this node
              If its pair of edge E(E2) has not been added facet F then
                E2->m_lstfacets.pus_back(F)         //add factets to its pair too
             end if
        end if
      end for
   end for
end
```



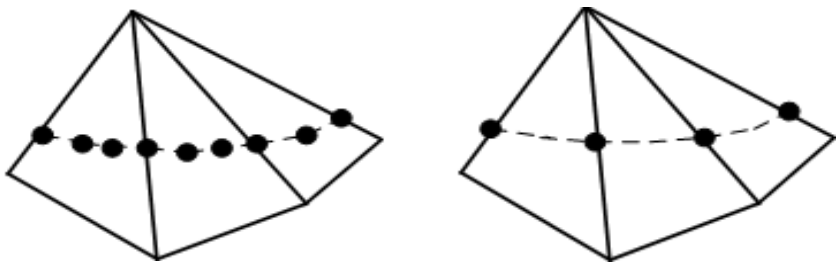
(a) (b)

รูปที่ 3.14 แสดงจุดขอบของฟันที่ตัดแล้ว (a) และการกำหนดจุดตัดของฟัน
กรณีที่ฟันมีโครงสร้างที่ฟันมีโครงสร้างติดกัน (b)

เมื่อกำหนดจุดตัดรอบฟันซี่ที่ต้องการตัดแล้ว ซึ่งจุดดังกล่าวต้องอยู่บนพื้นผิว และสามารถจำแนกได้ว่า จุดแต่ละจุดอยู่ภายในสามเหลี่ยมหรืออยู่บนด้านสามเหลี่ยมรูปใดบนโครงข่าย ขั้นตอนถัดไปคือ การกำหนดเชื่อมโยงจุดกำหนดขอบเขตของการตัด

2. การเชื่อมโยงจุดกำหนดขอบเขตการตัด (Cutting path linking)

เนื่องจากจุดที่ได้จากการกำหนดจุดของเส้นโค้งนั้นมีจำนวนมากเกินความจำเป็น ดังนั้นขั้นตอนวิธีการเชื่อมจุดจะเป็นการลดจำนวนจุดที่ไม่จำเป็นออกไป โดยเลือกเฉพาะจุดที่อยู่บนขอบของสามเหลี่ยมที่เส้นกำหนดขอบเขตการตัดลากผ่าน จากรูปที่ 3.15(a) จะพบว่าในรูปสามเหลี่ยมแต่ละรูปจะมีจุด landmark อยู่ภายในหลายจุด เพื่อความสะดวกในขั้นตอนการตัดจำเป็นต้องลดจำนวนจุดภายในรูปสามเหลี่ยมแต่ละรูปให้เหลือเฉพาะจุดที่ตัดผ่านเส้นขอบของรูปสามเหลี่ยมเท่านั้น ดังแสดงในรูปที่ 3.15(b)



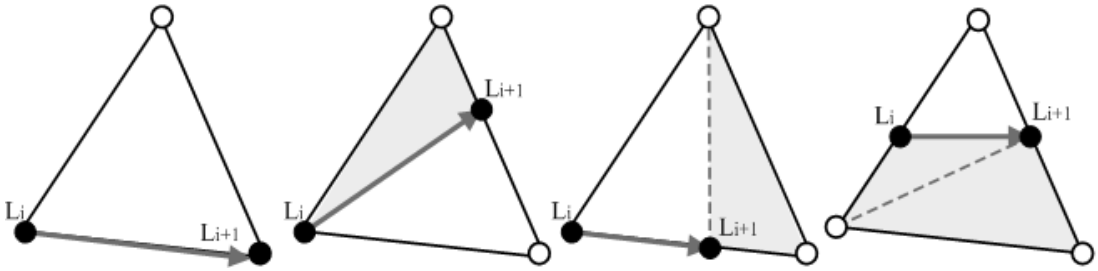
(a) (b)

รูปที่ 3.15 แสดงจุดกำหนดขอบเขตการตัด (a) และการเชื่อมโยงจุดกำหนดขอบเขตการตัด (b)

หลังจากตัดจุดที่อยู่ภายในสามเหลี่ยมออกไป เหลือเพียงจุด landmark ที่เป็นจุดมุมสามเหลี่ยม และจุดบนด้านสามเหลี่ยม ซึ่งเป็นขั้นตอนการสร้างเส้นทางการตัด เพื่อให้กระบวนการตัดพื้นผิวเสร็จสิ้น จำเป็นต้องสร้างสามเหลี่ยมใหม่ ซึ่งเป็นกระบวนการปรับปรุงโครงสร้างโครงข่าย

3. การปรับปรุงโครงสร้างโครงข่าย (Remeshing)

เป็นกระบวนการสร้างสามเหลี่ยมขึ้นใหม่จากเส้นทางการตัด การสร้างสามเหลี่ยมใหม่จะเกิดขึ้นเมื่อจุด landmark นั้นตัดผ่านด้านสามเหลี่ยมอย่างน้อยหนึ่งด้าน และจุด landmark ทั้งสองของสามเหลี่ยมนั้นต้องไม่ใช่จุดมุมของสามเหลี่ยม รูปแบบของการปรับปรุงโครงสร้างของโครงข่าย สามารถแบ่งออกได้เป็น 2 กรณี คือ กรณีที่มีจุด landmark เป็นจุดมุมสามเหลี่ยม (Li) และเป็นจุดบนด้านสามเหลี่ยม (Li+1) ต้องสร้างสามเหลี่ยม (แรเงา) ใหม่ 1 รูป (+1 facet) และเพิ่มจุดบนโครงข่ายอีก 1 จุด (+1 vertex) ดังรูปที่ 3.16(b) และ 3.16(c) ส่วนกรณีที่มีจุด landmark ทั้งสองอยู่บนด้านสามเหลี่ยม ซึ่งไม่ใช่ด้านเดียวกัน ต้องสร้างสามเหลี่ยม (แรเงา) ใหม่ 2 รูป (+2 facets) และเพิ่มจุด (vertex) บนโครงข่ายอีก 2 จุด (+2 vertices) ดังรูปที่ 3.16(d)



รูปที่ 3.16 แสดงเส้นทางการตัดลากผ่านด้านสามเหลี่ยม ดูได้จากรายละเอียดในเนื้อหา

หลังจากปรับปรุงโครงสร้างของโครงข่ายแล้ว ขั้นตอนต่อไป คือ การแยกแบบจำลองฟันซี่ที่ตัดออกจากเหงือก ด้วยเทคนิควิธีการขยายพื้นที่ (Region growing) (Castleman, 1996) เริ่มต้นจากการกำหนดจุดเริ่มต้นของการค้นหาจุดภายในของแบบจำลองฟัน (Vstart) แล้วขยายไปยังจุดรอบข้าง แล้วจะมีกระบวนการทำซ้ำ โดยตรวจสอบเงื่อนไขว่าจุดปัจจุบันใช่จุดที่เป็นขอบเขตการตัด (marker vertices) หรือไม่ ถ้าใช่ก็หยุดการค้นหาในรอบนั้น รายละเอียดของเทคนิควิธีการขยายพื้นที่ (Region growing) แสดงได้ดังนี้

Region growing algorithm.

input : Mesh *M*, start seed vertex *Vstart*, and marker vertices.
output : Segmented part of the mesh, the array label contains elements that are labeled.

Begin
Initialize heap *S*; *label* [*Vstart*]; PushHeap (*Vstart*);
while *not* Empty (*S*) **do**
  v_i := PopHeap (*S*);
  if v_i *is not marker vertices* **then**
    Grow (v_i **,** *S*, *label*);
  end if
end while
end

Procedure: Grow (*vertex, heap, label*) เป็นฟังก์ชั่นสำหรับค้นหาจุดที่อยู่รอบข้าง

input : Vertex v_i , an array *label*, and a heap *S*
output : Populate heap *S* with neighbors of v_i .

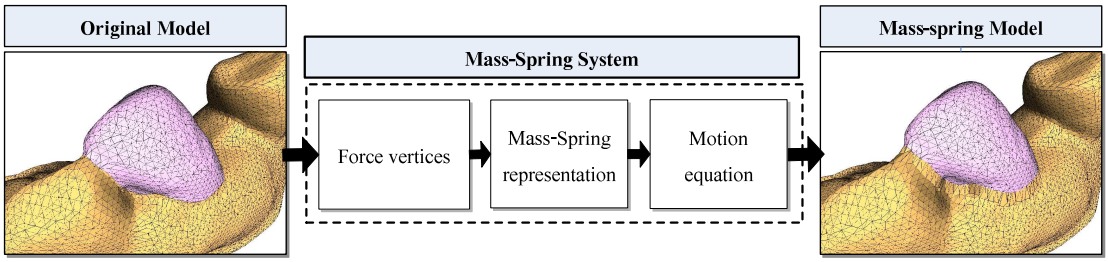
Begin
for each v_j *in umbrella neighborhood of* v_i **do**
  if *not lable* [v_j] **then**
    lable [v_j];
    PushHeap (*S,* v_j);
  end if
end for
end

ขั้นตอนการตัดพื้นผิว เป็นขั้นตอนการแยกแบบจำลองฟันออกจากเหงือก เพื่อใช้สำหรับกระบวนการจัดเรียงฟัน โดยเริ่มต้นจากกำหนดขอบเขตของการตัดรอบแบบจำลองฟัน ซึ่งจุดที่กำหนดจะเป็นจุดควมคุม (knot or control point) ของเส้นโค้ง (natural cubic spline curve) แล้วคำนวณหาจุดบนเส้นโค้ง จากนั้นเลือกเฉพาะจุด landmark ที่เป็นจุดมุมของสามเหลี่ยม หรือจุดที่อยู่บนด้านของสามเหลี่ยม เมื่อทราบจุดที่เป็นขอบเขตการตัดแล้ว ก็ดำเนินการสร้างสามเหลี่ยมใหม่ กรณีที่สามารถสร้างสามเหลี่ยมใหม่ได้ คือ กรณีที่จุด landmark ทั้งสองจุดของสามเหลี่ยมนั้น ไม่ได้เป็นจุดที่เป็นด้านของสามเหลี่ยม เราสามารถแยกจุดส่วนที่เป็นแบบจำลองฟันได้ด้วยวิธีการขยายพื้นที่ (Region growing) เมื่อแยกแบบจำลองฟันออกจากเหงือกแล้ว ขั้นตอนถัดไป คือ วิเคราะห์การเปลี่ยนแปลงรูปร่างของเหงือก ระหว่างขั้นตอนการจัดเรียงฟัน โดยใช้เทคนิคการสร้างแบบจำลองมวลสปริง

3.3.4 เทคนิคการสร้างแบบจำลองมวลสปริง (Mass-Spring Model)

แบบจำลองมวลสปริงถูกสร้างขึ้นเพื่อวิเคราะห์การเปลี่ยนแปลงรูปร่างของวัตถุ ซึ่งสำหรับงานวิจัยนี้ ใช้สำหรับวิเคราะห์การเปลี่ยนแปลงรูปร่างของเหงือก ขณะจัดเรียงฟัน เทคนิคการสร้างแบบจำลองมวลสปริง แสดงดังรูปที่ 3.17 เริ่มจากค้นหาจุดเชื่อมต่อระหว่างแบบจำลองฟันที่ตัด

และเหงือก ซึ่งเป็นจุดที่กำหนดแรงกระทำ (Force vertices) ขั้นตอนถัดไป คือ การกำหนดค่าต่าง ๆ ของแบบจำลองมวลสปริง (Mass-spring representation) ประกอบด้วย มวลสปริง ค่าคงที่สปริง ค่าคงที่การหน่วง ความยาวเริ่มต้น และตำแหน่งพิกัดของจุดมวล ขั้นตอนสุดท้าย เป็นกระบวนคำนวณหาตำแหน่งใหม่ของจุดบนแบบจำลองเหงือก โดยการแก้สมการการเคลื่อนที่ (Motion equation)



รูปที่ 3.17 แสดงกระบวนการสร้างแบบจำลองมวลสปริง

1. การกำหนดจุดกำเนิดแรง (Force vertices)

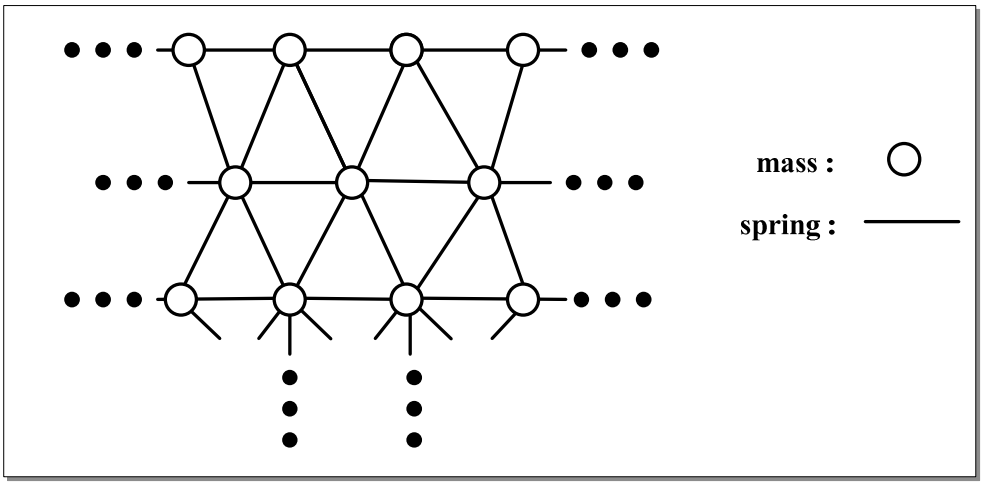
เป็นกระบวนการค้นหาจุดที่เชื่อมระหว่างแบบจำลองฟันที่ต้องการจัดเรียงกับเหงือก โดยเริ่มต้นจากกระบวนค้นหาจุดขอบของแบบจำลองฟันและเหงือก ถ้าจุดขอบของแบบจำลองทั้งสองเป็นจุดเดียวกัน แสดงว่าจุดดังกล่าวเป็นจุดเชื่อม และถูกกำหนดให้เป็นจุดที่มีแรงกระทำในขั้นตอนการจัดเรียงฟัน เนื่องจากจุดนี้จะเลื่อนตำแหน่งเมื่อมีการจัดเรียงฟัน สำหรับกระบวนการค้นหาจุดขอบของแบบจำลอง สามารถพิจารณาได้จากจำนวนสามเหลี่ยมที่ใช้ด้านร่วมกัน ถ้ามีจำนวนสามเหลี่ยมที่ใช้ด้านร่วมกัน 1 รูป แสดงว่าจุดของด้านนั้นเป็นจุดขอบของแบบจำลอง ดังรูปที่ 3.18

รูปที่ 3.18 แสดงจุดขอบของแบบจำลองฟันและเหงือก

เมื่อทราบจุดที่เป็นจุดกำเนิดแรง ซึ่งก็คือ จุดที่เชื่อมต่อระหว่างแบบจำลองฟันที่จัดเรียงกับเหงือก เนื่องจากจุดดังกล่าวจะทราบพิกัดตำแหน่งที่เปลี่ยนแปลงไปจากพิกัดเดิม ทำให้สามารถคำนวณแรงกระทำจากจุดดังกล่าวนี้ได้ ขั้นตอนถัดไป คือ การกำหนดค่าตัวแปรต่าง ๆ ให้กับแบบจำลองมวลสปริง สำหรับคำนวณหาแรงที่กระทำกับจุดภายในของแบบจำลองเหงือก

2. การสร้างตัวแทนแบบจำลองมวลสปริง (Mass-Spring representation)

ขั้นตอนนี้ เป็นการกำหนดตัวแทนของแบบจำลองมวลสปริง โดยกำหนดให้จุดบนโครงข่ายเป็นจุดมวลของสปริง (mass) และด้านของสามเหลี่ยมแต่ละด้านเป็นตัวแทนของสปริง (spring) ดังรูปที่ 3.19 ซึ่งค่าต่าง ๆ ที่เกี่ยวข้องกับการสร้างแบบจำลองมวลสปริง ได้แก่ มวลสปริง (mass) ค่าคงที่สปริง (stiffness) ค่าคงที่การหน่วง (damping coefficient) ความยาวเริ่มต้น (rest length) และตำแหน่งพิกัดปัจจุบันของจุดมวล สามารถกำหนดได้ดังนี้



รูปที่ 3.19 แสดงการกำหนดตัวแทนของแบบจำลองมวลสปริง

a) มวลสปริง (m) คือ จุดบนโครงข่ายของแบบจำลองเหงือก สามารถกำหนดค่ามวลสปริงได้ 3 วิธี คือ กำหนดด้วยค่าคงที่ (constant) ค่าความหนาแน่นของพื้นที่สามเหลี่ยมรอบข้าง และค่าความยาวรวมของด้านรอบข้าง ดังสมการที่ 3.8

$$m_i = \begin{cases} \text{constant} \\ \dfrac{\sum_{j}^{j \le N} A_j}{3} \\ \sum_{j}^{j \le N} l_{ij} \end{cases} \tag{3.8}$$

เมื่อ N คือ จำนวนของสามเหลี่ยมรอบข้างหรือจำนวนของด้านรอบข้าง

A คือ พื้นที่ของสามเหลี่ยมรอบข้าง

l คือ ความยาวของด้านรอบข้าง

b) ค่าคงที่สปริง (k) สามารถกำหนดได้ 3 วิธี คือ กำหนดด้วยค่าคงที่ (constant) ค่าสัดส่วนของความยาวสปริง และอัตราส่วนของพื้นที่สามเหลี่ยมรอบข้างต่อความยาว ดังสมการที่ 3.9

$$k_i = \begin{cases} \text{constant} \\ \dfrac{\sum_{j}^{j \le N} A_j}{l^2} \\ \dfrac{1}{l} \end{cases} \tag{3.9}$$

c) ค่าคงที่การหน่วง (c) กำหนดได้ 2 วิธี คือ กำหนดด้วยค่าคงที่ (constant) และอัตราส่วนของมวลรวมของสปริงต่อความยาว ดังสมการที่ 3.10

$$c_i = \begin{cases} \text{constant} \\ \dfrac{2\sqrt{kM}}{l} \end{cases} \tag{3.10}$$

เมื่อ k คือ ค่าคงที่สปริง

M คือ มวลรวมของสปริง ($m1 + m2$)

l คือ ความยาวปกติของสปริง (rest length)

d) ค่าความยาวปกติของสปริง (rest length) คือ ความยาวของด้านของสามเหลี่ยม ส่วนค่าตำแหน่งพิกัดของจุดมวลสปริง กำหนดได้จากอ้างอิงพิกัดของจุดบนโครงข่าย

การกำหนดค่าต่าง ๆ งานวิจัยนี้จะกำหนดให้เป็นค่าคงที่ค่าใดค่าหนึ่งเท่านั้น เนื่องจากว่าแบบจำลองที่ใช้ในการทดสอบไม่ได้มีรายละเอียดที่ซับซ้อน ทำให้คุณสมบัติของสปริงและมวลสปริงในการสร้างแบบจำลองมวลสปริงมีความใกล้เคียงกัน เพื่อความสะดวกต่อการวิเคราะห์การเปลี่ยนแปลงรูปร่างของแบบจำลองของเหงือก และช่วยลดเวลาในการประมวลผล

ตัวแทนของแบบจำลองมวลสปริง คือ จุดบนโครงข่าย (vertex) และด้านของสามเหลี่ยม (edge) ซึ่งจุดบนโครงข่ายเป็นตัวแทนของมวลสปริง และด้านสามเหลี่ยมเป็นตัวแทนของสปริง ค่าต่าง ๆ ที่เกี่ยวข้องกับการสร้างแบบจำลองมวลสปริง สามารถกำหนดได้ดังที่กล่าวมาแล้ว จะเห็นได้ว่าสามารถกำหนดค่าให้กับตัวแปรต่าง ๆ ได้หลายวิธี เพื่อเปรียบเทียบผลลัพธ์ที่ได้ในกระบวนทดสอบ เมื่อแบบจำลองฟันมีการเปลี่ยนแปลงตำแหน่ง จำเป็นต้องคำนวณหาจุดบนโครงข่ายของแบบจำลองเหงือก เพื่อวิเคราะห์การเปลี่ยนแปลงรูปร่าง

3. การสร้างและแก้สมการการเคลื่อนที่ (Motion equation)

เป็นขั้นตอนการหาจุดพิกัดใหม่ของแบบจำลองเหงือก เมื่อมีการจัดเรียงฟัน เพื่อวิเคราะห์การเปลี่ยนแปลงรูปร่างของเหงือก สามารถคำนวณด้วยสมการการเคลื่อนที่ ตามกฎการเคลื่อนที่ข้อที่สองของนิวตัน ดังสมการที่ 3.11 เริ่มต้นจากคำนวณหาแรงที่กระทำกับจุดมวล จากนั้นจึงแก้สมการการเคลื่อนที่ด้วยวิธีการวิเคราะห์เชิงตัวเลข (numerical analysis)

$$\begin{cases} \boldsymbol{F}_i = m\boldsymbol{a}_i \\ \boldsymbol{a}_i = \dfrac{\partial \boldsymbol{v}_i}{\partial t} = \boldsymbol{v}' = \boldsymbol{x}'' = \dfrac{\boldsymbol{F}_i}{m} \\ \boldsymbol{v}_i = \dfrac{\partial \boldsymbol{x}}{\partial t} = \boldsymbol{x}' \end{cases} \tag{3.11}$$

เมื่อ $\boldsymbol{F}$ คือ แรงที่กระทำกับจุดมวล

m คือ มวลของจุดมวล

$\boldsymbol{a}$ คือ ความเร่งของจุดมวล

$\boldsymbol{v}$ คือ ความเร็วของจุดมวล

$\boldsymbol{x}$ คือ ตำแหน่งบนพิกัด 3 มิติของจุดมวล

แรงที่กระทำกับจุดมวล สามารถแบ่งออกได้เป็น 2 ประเภท คือ แรงภายใน และแรงภายนอก สามารถคำนวณหาแรงต่าง ๆ ได้ดังนี้

a) แรงภายใน (internal force, F_{int}) คือ แรงที่เกิดจากแรงกระทำของสปริงรอบข้างจุดมวลนั้น สามารถคำนวณได้จากผลรวมของแรงกระทำของสปริงต่าง ๆ ดังสมการที่ 3.12

$$\boldsymbol{F}_{\text{int}} = \sum_{j \in R} k_j \left(\boldsymbol{L}_j - \boldsymbol{l}_j^0 \frac{\boldsymbol{L}_j}{\|\boldsymbol{L}_j\|} \right) \tag{3.12}$$

เมื่อ k คือ ค่าคงที่ของสปริง

$\boldsymbol{L}$ คือ ความยาวของสปริงที่ถูกยืดหรือหด

$\boldsymbol{l}^0$ คือ ความยาวเริ่มต้นของสปริง

R คือ สปริงที่อยู่รอบข้างจุดมวล

b) แรงภายนอก (external force, F_{ext}) คือ แรงที่เกิดจากแรงการหน่วง (damping force, F_{dis}) ของสปริงรอบข้างของจุดมวล คำนวณได้ผลคูณของค่าคงที่การหน่วง (damping coefficient, c) ของสปริง กับความเร็วของจุดมวล ดังสมการที่ 3.13

$$\boldsymbol{F}_{ext} = \boldsymbol{F}_{dis} = -\sum_{j \in R} c\boldsymbol{v}_i \tag{3.13}$$

เมื่อ $\boldsymbol{c}$ คือ ค่าคงที่การหน่วงของสปริง

$\boldsymbol{v}$ คือ ความเร็วของจุดมวล

จากสมการที่ 3.11 – 3.13 สามารถสร้างสมการการเคลื่อนที่ของจุดมวล ณ เวลาใด ๆ สำหรับแบบจำลองมวลสปริง ได้ดังนี้

$$\begin{cases} m\dfrac{\partial^2 \boldsymbol{x}(t)}{\partial t^2} = \boldsymbol{F}_{\text{int}}(t) + \boldsymbol{F}_{ext}(t) \\ \boldsymbol{a}(t) = \dfrac{\boldsymbol{F}_{\text{int}}(t) + \boldsymbol{F}_{ext}(t)}{m} = \dfrac{\boldsymbol{F}_{\text{int}}(t) - c\boldsymbol{v}(t)}{m} \\ \boldsymbol{v}(t) = \dfrac{\partial \boldsymbol{x}(t)}{\partial t} \end{cases} \tag{3.14}$$

การแก้สมการการเคลื่อนที่ของจุดมวล เพื่อคำนวณหาตำแหน่งพิกัดใหม่ของจุดมวล สามารถแก้สมการเพื่อหาผลเฉลยสมการการเคลื่อนที่ ซึ่งเป็นสมการเชิงอนุพันธ์ ด้วยวิธีแบบชัดแจ้ง (Explicit method) สำหรับงานวิจัยนี้เลือกใช้ระเบียบวิธีของออยเลอร์ (Euler's method) และวิธีรุงเก-คุตตาอันดับสี่ (4^{th} order Runge-Kutta method) เพื่อเปรียบเทียบประสิทธิภาพของวิธีทั้งสอง ซึ่งมีข้อดีข้อเสียแตกต่างกัน ซึ่งวิธีของออยเลอร์จะคำนวณหาอนุพันธ์เพียงครั้งเดียว แต่ค่าผิดพลาดมาก ส่วนวิธีรุงเก-คุตตาอันดับสี่มีค่าผิดพลาดน้อยกว่าวิธีการของออยเลอร์ แต่ต้องคำนวณหาอนุพันธ์หลายครั้ง

a) วิธีของออยเลอร์ (Euler's method) ใช้เทคนิคการประมาณค่านอกช่วงเวลาเพื่อหาผลเฉลยสมการเชิงอนุพันธ์ ขั้นตอนการประมาณค่า เริ่มต้นจากกำหนดความเร็ว $v(t_0)$ และพิกัดตำแหน่งเริ่มต้น $x(t_0)$ ดังสมการที่ 3.15 แล้วคำนวณหาค่าอนุพันธ์ที่เวลาเริ่มต้น (t_0) ดังสมการที่ 3.16 สุดท้ายคำนวณหาผลเฉลยของสมการด้วยการประมาณค่า ณ เวลาถัดไป (t_0+h) โดยใช้ค่าอนุพันธ์ที่คำนวณได้ที่เวลาเริ่มต้น ดังสมการที่ 3.18 และสมการที่ 3.19

กำหนดค่าเริ่มต้น, $\boldsymbol{x}(t_0) = \boldsymbol{x}_0 \qquad \boldsymbol{v}(t_0) = \boldsymbol{v}_0$ (3.15)

คำนวณหาค่าอนุพันธ์ ณ เวลา t_0, $\boldsymbol{x}'(t_0) \qquad \boldsymbol{v}'(t_0)$ (3.16)

สมมุติค่าอนุพันธ์ ณ เวลา t, $\boldsymbol{x}'(t) = \boldsymbol{x}'(t_0) \qquad \boldsymbol{v}'(t) = \boldsymbol{v}'(t_0)$ (3.17)

ประมาณค่าพิกัดตำแหน่งของจุดมวล ณ เวลา $t_o \rightarrow t_o + h$,

$$\boldsymbol{x}(t_o + h) = \boldsymbol{x}(t_o) + h\boldsymbol{x}'(t_o) = \boldsymbol{x}(t_o) + h\boldsymbol{v}(t_o) \tag{3.18}$$

ประมาณค่าความเร็วของจุดมวล ณ เวลา $t_o \rightarrow t_o + h$,

$$\boldsymbol{v}(t_o + h) = \boldsymbol{v}(t_o) + h\boldsymbol{v}'(t_o) = \boldsymbol{v}(t_o) + h\frac{\boldsymbol{F}(t_o) - c\boldsymbol{v}(t_o)}{m} \tag{3.19}$$

b) วิธีรุงเก-คุตตาอันดับสี่ (4^{th} order Runge-Kutta method) ใช้เทคนิคการประมาณค่านอกช่วงเวลาเพื่อหาผลเฉลยสมการเชิงอนุพันธ์เช่นเดียวกับวิธีของออยเลอร์ สำหรับฟังก์ชันความเร่งจะถูกกำหนดในรูปแบบของพิกัดตำแหน่งและความเร็ว ดังสมการที่ 3.20 การประมาณค่าพิกัดตำแหน่ง และความเร็วของจุดมวล วิธีรุงเก-คุตตาอันดับสี่ คือ การคำนวณหาอนุพันธ์ทั้งหมดสี่ครั้งในช่วงเวลา t_o, $t_o + h/2$, $t_o + h/2$ และ $t_o + h$ ดังสมการที่ 3.21 สมการที่ 3.24 สมการที่ 3.27 และสมการที่ 3.30 ตามลำดับ โดยตำแหน่งของจุดมวล ณ เวลาใด ๆ สามารถคำนวณได้จากตำแหน่งของจุดมวล และความเร็ว ณ เวลาก่อนหน้า ขณะที่ความเร็วมวลคำนวณได้จากความเร็ว และความเร่ง ณ เวลาก่อนหน้า เช่น ถ้าต้องการหาตำแหน่ง และความเร็วของจุดมวล ณ เวลา $t_o + h/2$ ก็คำนวณได้จากตำแหน่ง ความเร็ว และความเร่ง ณ เวลา t_o ดังสมการที่ 3.22 และสมการที่ 3.23 เป็นต้น สุดท้ายคำนวณหาผลเฉลยของสมการด้วยการประมาณค่าตำแหน่งของจุดมวล และความเร็ว ณ เวลาถัดไป (t_o+h) โดยใช้ค่าอนุพันธ์ที่คำนวณได้ที่เวลาเริ่มต้น ดังสมการที่ 3.30 และสมการที่ 3.31

กำหนดฟังก์ชันความเร่ง ณ เวลาใด ๆ, $\boldsymbol{a}(\boldsymbol{x}(t), \boldsymbol{v}(t)) = \dfrac{\boldsymbol{F}(t) - c\boldsymbol{v}(t)}{m}$ (3.20)

คำนวณหาค่าอนุพันธ์ ณ เวลา t_o, $\boldsymbol{k}_1 = \boldsymbol{x}'(t_o) = \boldsymbol{v}(t_o)$ (3.21)

พิกัดตำแหน่งของจุดมวล ณ เวลา $t_o \rightarrow t_o + h/2$,

$$\boldsymbol{x}(t_0 + h/2) = \boldsymbol{x}(t_0) + \frac{h}{2}\boldsymbol{v}(t_0) \tag{3.22}$$

ความเร็วของจุดมวล ณ เวลา $t_0 \rightarrow t_0 + h/2$,

$$\boldsymbol{l}_1 = \boldsymbol{a}(\boldsymbol{x}(t), \boldsymbol{v}(t))$$

$$\boldsymbol{v}(t_0 + h/2) = \boldsymbol{v}(t_0) + \frac{h}{2}\boldsymbol{l}_1 \tag{3.23}$$

คำนวณหาค่าอนุพันธ์ ณ เวลา $t_0 \rightarrow t_0 + h/2$,

$$\boldsymbol{k}_2 = \boldsymbol{x}'(t_0 + h/2) = \boldsymbol{v}(t_0 + h/2) = \boldsymbol{v}(t_0) + \frac{h}{2}\boldsymbol{l}_1 \tag{3.24}$$

พิกัดตำแหน่งของจุดมวล ณ เวลา $t_0 \rightarrow t_0 + h/2$,

$$\boldsymbol{x}(t_0 + h/2) = \boldsymbol{x}(t_0) + \frac{h}{2}\boldsymbol{v}(t_0) \tag{3.25}$$

ความเร็วของจุดมวล ณ เวลา $t_0 \rightarrow t_0 + h/2$,

$$\boldsymbol{l}_2 = \boldsymbol{a}(\boldsymbol{x}(t_0) + \boldsymbol{k}_1 \frac{h}{2}, \boldsymbol{k}_2)$$

$$\boldsymbol{v}(t_0 + h/2) = \boldsymbol{v}(t_0) + \frac{h}{2}\boldsymbol{l}_2 \tag{3.26}$$

คำนวณหาค่าอนุพันธ์ ณ เวลา $t_0 \rightarrow t_0 + h/2$,

$$\boldsymbol{k}_3 = \boldsymbol{x}'(t_0 + h/2) = \boldsymbol{v}(t_0 + h/2) = \boldsymbol{v}(t_0) + \frac{h}{2}\boldsymbol{l}_2 \tag{3.27}$$

พิกัดตำแหน่งของจุดมวล ณ เวลา $t_0 \rightarrow t_0 + h$,

$$\boldsymbol{x}(t_0 + h) = \boldsymbol{x}(t_0) + h\boldsymbol{v}(t_0) \tag{3.28}$$

ความเร็วของจุดมวล ณ เวลา $t_0 \rightarrow t_0 + h$,

$$\boldsymbol{l}_3 = \boldsymbol{a}(\boldsymbol{x}(t_0) + \boldsymbol{k}_2 \frac{h}{2}, \boldsymbol{k}_3)$$

$$\boldsymbol{v}(t_0 + h) = \boldsymbol{v}(t_0) + h\boldsymbol{l}_3 \tag{3.29}$$

คำนวณหาค่าอนุพันธ์ ณ เวลา $t_0 \rightarrow t_0 + h$,

$$\boldsymbol{k}_4 = \boldsymbol{x}'(t_0 + h) = \boldsymbol{v}(t_0 + h) = \boldsymbol{v}(t_0) + h\boldsymbol{l}_3$$
$$\boldsymbol{l}_4 = \boldsymbol{a}(\boldsymbol{x}(t_0) + \boldsymbol{k}_3 h, \boldsymbol{k}_4) \qquad (3.30)$$

ประมาณค่าพิกัดตำแหน่งของจุดมวล ณ เวลา $t_0 \rightarrow t_0 + h$,

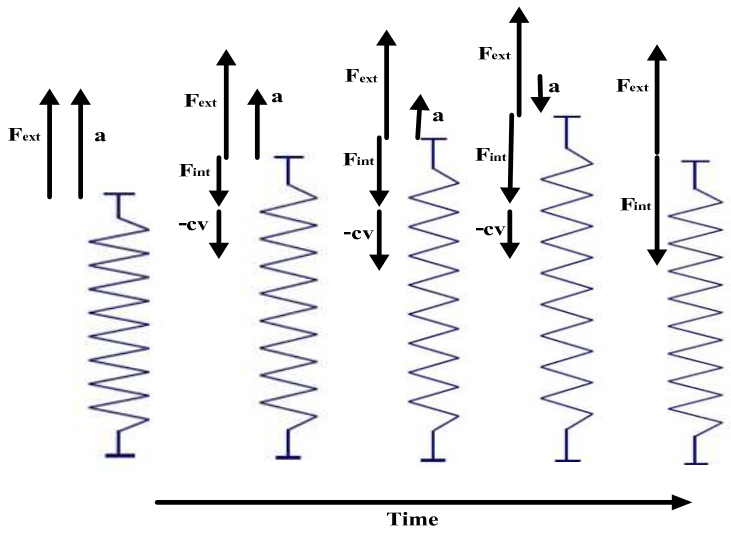
$$\boldsymbol{x}(t_0 + h) = \boldsymbol{x}(t_0) + \frac{h}{6}(\boldsymbol{k}_1 + 2\boldsymbol{k}_2 + 2\boldsymbol{k}_3 + \boldsymbol{k}_4) \qquad (3.31)$$

ประมาณค่าความเร็วของจุดมวล ณ เวลา $t_0 \rightarrow t_0 + h$,

$$\boldsymbol{v}(t_0 + h) = \boldsymbol{v}(t_0) + \frac{h}{6}(\boldsymbol{l}_1 + 2\boldsymbol{l}_2 + 2\boldsymbol{l}_3 + \boldsymbol{l}_4)$$

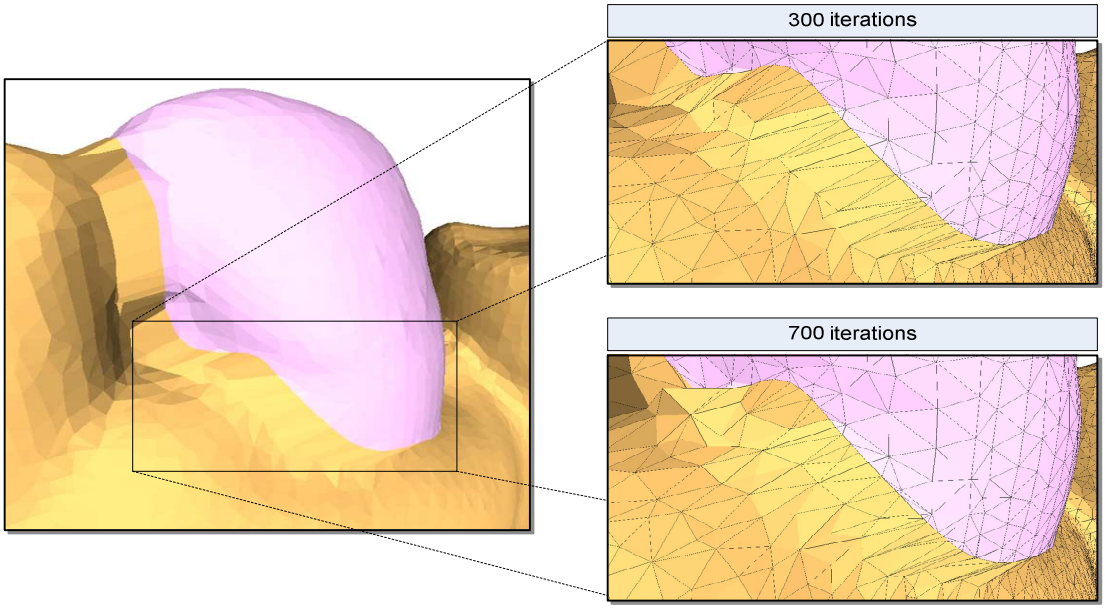
(3.32)

โดยคำนวณหาตำแหน่งพิกัดใหม่ของจุดมวล ไปเรื่อย ๆ จนกว่าระบบจะเข้าสู่สภาวะสมดุลย์ คือ พลังงานหรือแรงภายนอกที่กระทำกับจุดมวลสปริงมีค่าเท่ากับพลังงานหรือแรงภายในที่กระทำกับจุดมวลสปริงนั้น จากรูปที่ 3.20 จะเห็นได้ว่าเมื่อเวลาผ่านไปเรื่อย ๆ ค่าแรงภายในของมวลสปริง (F_{int}) จะมีค่าเพิ่มขึ้น และความเร่งก็จะมีค่าลดลงด้วย จนกระทั่งแรงภายในมีค่าเท่ากับแรงภายนอกระบบก็จะอยู่ในสภาวะสมดุลย์ กล่าวคือ ตำแหน่งของมวลสปริงจะไม่มีเปลี่ยนอีกจนกว่าจะมีแรงภายนอกมากระทำอีกครั้ง



รูปที่ 3.20 แสดงการเข้าสู่สภาวะสมดุลย์ของระบบมวลสปริง

ตัวอย่างผลลัพธ์การจำลองเปลี่ยนแปลงรูปร่างของเหงือก ในแต่ละรอบของการคำนวณหาค่าตำแหน่งจุดมวลซึ่งใช้วิธีรุงเก-คุตตาอันดับสี่ จะเห็นได้ว่าจุดมวลของเหงือกจะมีการเปลี่ยนแปลงตำแหน่งไปยังตำแหน่งใหม่ที่เหมาะสม ดังรูปที่ 3.21



รูปที่ 3.21 แสดงการจำลองการเปลี่ยนแปลงรูปร่างของเหงือก

3.4 สรุปวิธีดำเนินการวิจัย

ขั้นตอนดำเนินการวิจัยเริ่มต้นจากขั้นตอนการลดทอนรายละเอียดของวัตถุ นำข้อมูลที่ได้จากการลดทอนรายละเอียดไปทดสอบกับขั้นตอนวิธีอื่น ๆ ได้แก่ ขั้นขอนการตัดพื้นผิว และขั้นตอนการจำลองการเปลี่ยนแปลงรูปร่างของเหงือก เพื่อเปรียบเทียบประสิทธิภาพของขั้นตอนวิธีต่าง ๆ และเพื่อค้นหาการลดทอนรายละเอียดของวัตถุที่เหมาะสมกับกระบวนการจำลองการจัดฟัน ซึ่งวิธีการทดลองและผลลัพธ์ที่ได้จากการทดลองจะกล่าวถึงในบทถัดไป

บทที่ 4
การทดสอบและอภิปรายผล

การจำลองการรักษาทางทันตกรรมจัดฟัน ผู้วิจัยได้พัฒนาโปรแกรมต้นแบบด้วย Borland C++ Builder® และ OpenGL® ซึ่งโปรแกรมจะแสดงผลในรูปแบบ 3 มิติ การทดสอบประสิทธิภาพของขั้นตอนวิธีจำลองการจัดฟัน จะทดสอบกับข้อมูลฟัน 2 ชุด คือ ขากรรไกรล่าง (Mandible) และขาไกรกรรบน (Maxilla) ซึ่งข้อมูลฟันแต่ละชุดจะมีลักษณะที่แตกต่างกัน คือ ข้อมูลแบบจำลองขากรรไกรล่าง ปัญหาฟันมีลักษณะเป็นฟันห่าง ส่วนข้อมูลแบบจำลองขากรรไกรบน ฟันมีลักษณะซ้อนกัน การทดสอบนั้นจะทดสอบประสิทธิภาพของแต่ละขั้นตอนวิธีที่ได้พัฒนาขึ้น เริ่มต้นจากทดสอบการลดทอนรายละเอียดของวัตถุ นำแบบจำลองฟันที่ถูกลดทอนรายละเอียดไปทดสอบการตัดพื้นผิว และทดสอบการเปลี่ยนแปลงรูปร่างของแบบจำลองเหงือก เปรียบเทียบผลการทดสอบประสิทธิภาพของแต่ละขั้นตอนวิธี และสรุปผลการทดสอบ การทดสอบทั้งหมดประมวลผลบนเครื่องคอมพิวเตอร์ Pentium® IV ความเร็ว 3.2GHz (Hyper Threading) หน่วยความจำหลัก 512 MB การ์ดแสดงผล RADEON X300 SE 128 MB

4.1 ผลการทดสอบการลดทอนรายละเอียดของวัตถุ (Mesh Decimation)

ทดสอบกับข้อมูลแบบจำลองฟัน 2 ชุด คือ แบบจำลองฟันล่าง และแบบจำลองฟันบน ซึ่งมีจำนวนจุดบนโครงข่ายทั้งหมด 14,556 จุด และ 28,444 จุด ตามลำดับ โดยทดสอบลดทอนรายละเอียดแตกต่างกัน คือ ตั้งแต่ 10% ถึง 90% การลดทอนรายละเอียด การทดสอบประสิทธิภาพของเทคนิควิธีการลดทอนรายละเอียด พิจารณาจากข้อมูลต่อไปนี้ คือ จำนวนจุดที่ลบออกจากโครงข่าย จำนวนจุดและสามเหลี่ยมที่เหลือ เวลาประมวลผลสำหรับการลดทอนรายละเอียด เวลาแสดงผล เวลาประมวลผลสำหรับขั้นตอนวิธีการสร้างกราฟด้านสามเหลี่ยม (Edge graph) และขนาดของแฟ้มข้อมูล

ตารางที่ 4.1 และตารางที่ 4.2 แสดงจำนวนจุดที่ลบออกจากโครงข่าย จำนวนจุดและสามเหลี่ยมที่เหลืออยู่หลังจากการลดทอนรายละเอียดแล้ว และขนาดของแฟ้มข้อมูลของแบบจำลองฟันล่าง (mandible) และฟันบน (maxilla) ตามเปอร์เซ็นต์การลดทอนรายละเอียดต่าง ๆ จะเห็นได้ว่าเมื่อเปอร์เซ็นต์การลดทอนรายละเอียดเพิ่มมากขึ้นจะทำให้จำนวนจุดที่ลบออกจากโครงข่ายมากขึ้น ส่วนจำนวนจุดและสามเหลี่ยมที่เหลือจะน้อยลง และขนาดของแฟ้มข้อมูลก็จะน้อยลงด้วย

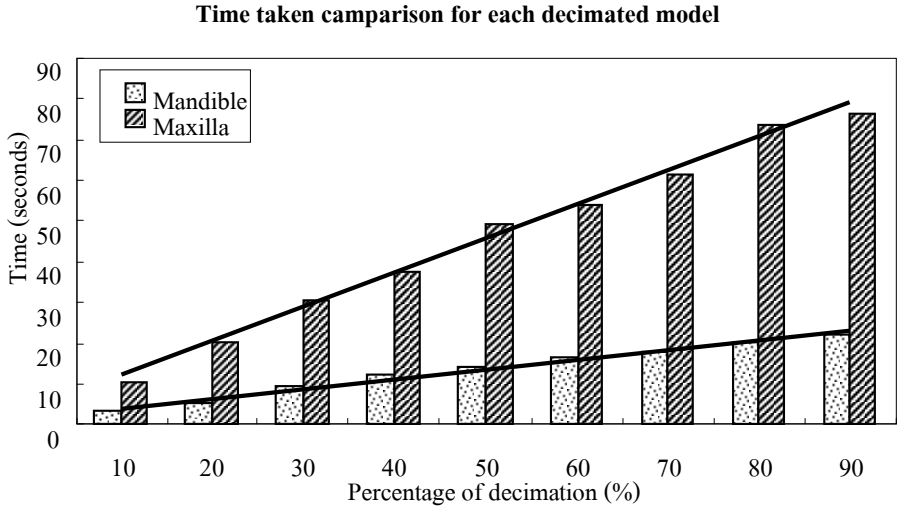
ตารางที่ 4.1 แสดงผลการทดสอบขั้นตอนวิธีลดทอนรายละเอียดกับแบบจำลองขากรรไกรล่าง

ลดทอน รายละเอียด (%)	จำนวนจุดที่ ลบออก	จำนวนจุดที่ เหลือ	จำนวนสามเหลี่ยม ที่เหลือ	ขนาดของแฟ้มข้อมูล (กิโลไบต์)
original	0	14556	29108	1934
10	1455	13101	26198	1280
20	2911	11645	23286	1138
30	4366	10190	20376	996
40	5822	8734	17464	853
50	7278	7278	14552	711
60	8733	5823	11642	569
70	10189	4367	8730	427
80	11644	2912	5820	285
90	13100	1456	2908	143

ตารางที่ 4.2 แสดงผลการทดสอบขั้นตอนวิธีลดทอนรายละเอียดกับแบบจำลองขากรรไกรบน

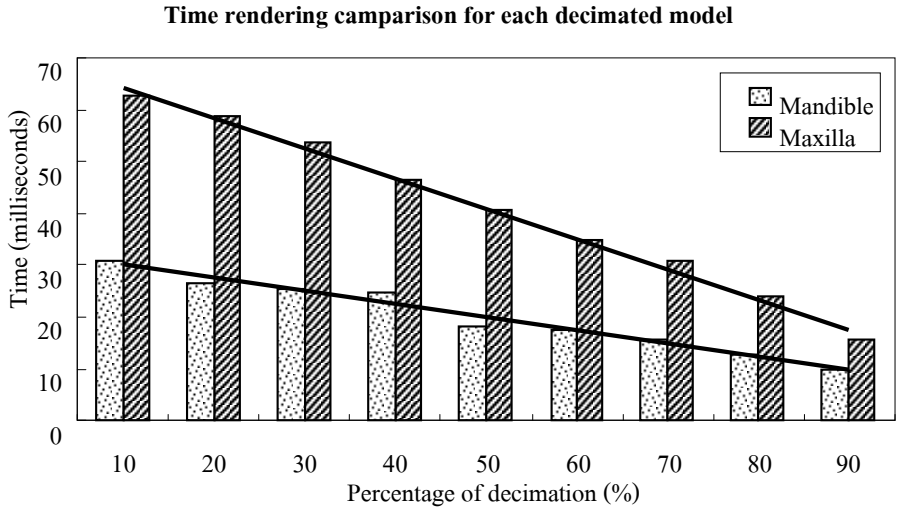
ลดทอน รายละเอียด (%)	จำนวนจุดที่ ลบออก	จำนวนจุดที่ เหลือ	จำนวนสามเหลี่ยม ที่เหลือ	ขนาดของแฟ้มข้อมูล (กิโลไบต์)
original	0	28444	56880	3778
10	2844	25600	51192	2500
20	5688	22756	45504	2222
30	8533	19911	39814	1945
40	11377	17067	34126	1667
50	14222	14222	28436	1389
60	17066	11378	22748	1111
70	19910	8534	17058	833
80	22755	5689	11374	556
90	25599	2845	5686	287

รูปที่ 4.1 แสดงการเปรียบเทียบเวลาประมวลผลสำหรับการลดทอนรายละเอียดของแบบจำลองขากรรไกรล่างและบน จะเห็นได้ว่า เวลาที่ใช้ในการลดทอนรายละเอียดจะเพิ่มขึ้นตามเปอร์เซ็นต์การลดทอนรายละเอียด เนื่องจากว่าจำนวนจุดที่ลบออกจากโครงข่ายจะมากขึ้นตามจำนวนเปอร์เซ็นต์การลดทอนรายละเอียดที่เพิ่มขึ้น



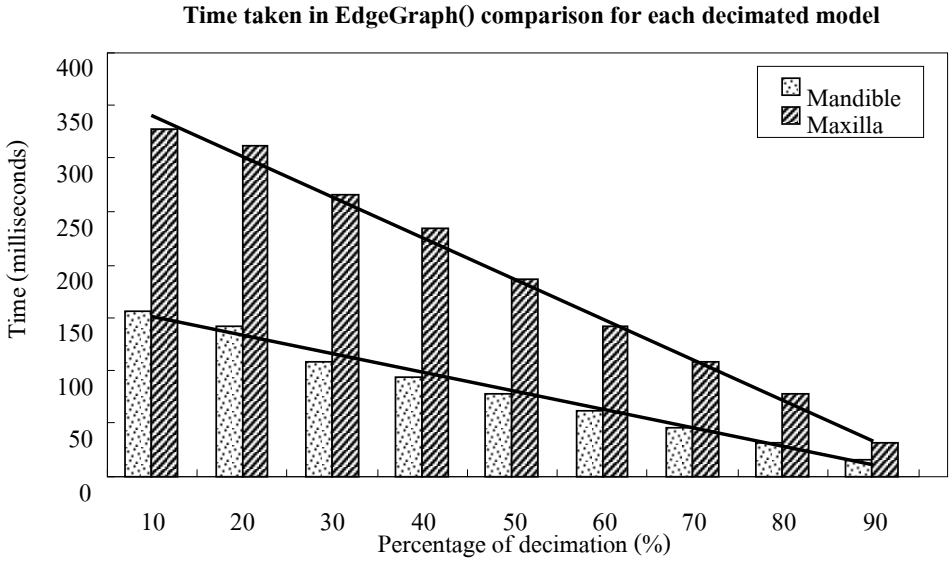
รูปที่ 4.1 แสดงเวลา (วินาที) ที่ใช้ในการลดทอนรายละเอียดของแบบจำลองขากรรไกรล่างและบน

รูปที่ 4.2 แสดงเวลาที่ใช้ในการแสดงผลใน 3 มิติ แบบแบนราบ (flat shading) ของแบบจำลองขากรรไกรล่างและบน จะพบว่า เมื่อมีการลดทอนรายละเอียดมากขึ้น แบบจำลองที่ถูกลดทอนรายละเอียดแล้ว จะใช้เวลาในการแสดงผลน้อยลงด้วย



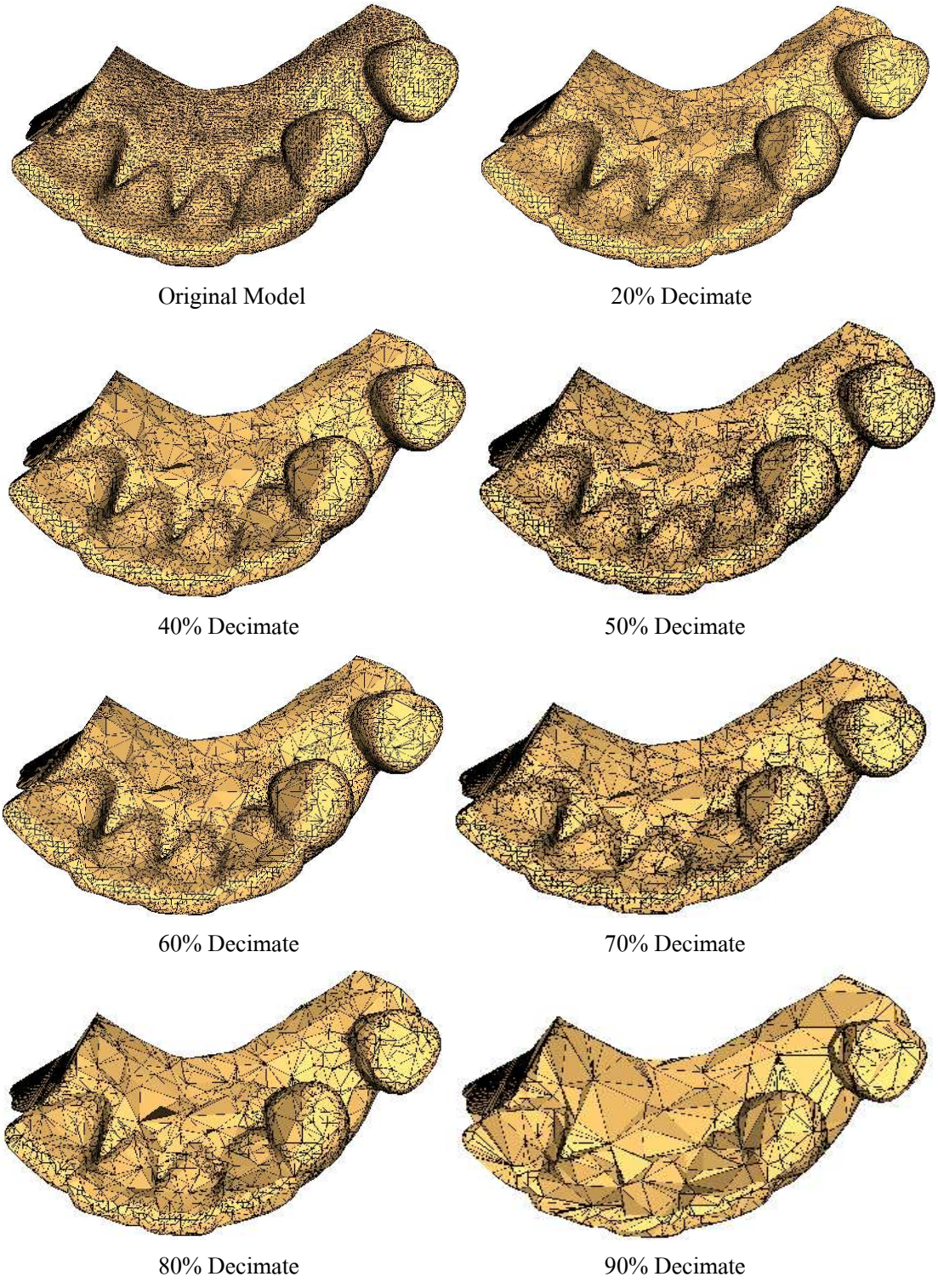
รูปที่ 4.2 แสดงเวลา (มิลลิวินาที) ที่ใช้ในการแสดงผลแบบแบนราบ (flat shading) ของแบบจำลองขากรรไกรล่างและบน

จากรูปที่ 4.3 แสดงเวลาประมวลผลสำหรับขั้นตอนวิธีการสร้างกราฟด้านสามเหลี่ยม (Edge graph) พบว่า เวลาที่ใช้ประมวลผลนั้นจะลดลงตามเปอร์เซ็นต์การลดทอนรายละเอียดที่มากขึ้นด้วย

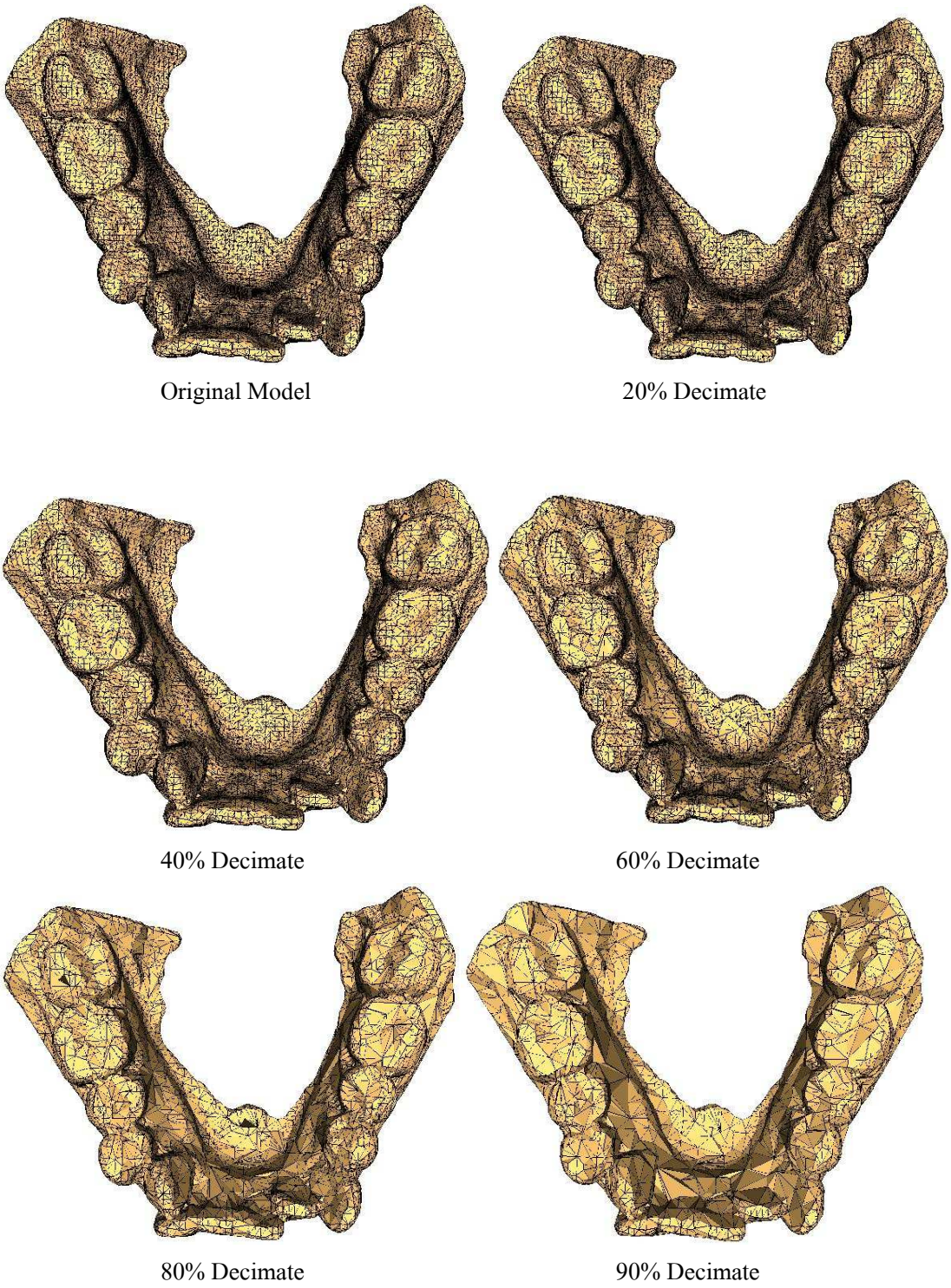


รูปที่ 4.3 แสดงเวลา (มิลลิวินาที) ที่ใช้ประมวลผลสำหรับขั้นตอนวิธีการสร้างกราฟด้านสามเหลี่ยม

รูปที่ 4.4 และรูปที่ 4.5 แสดงแบบจำลองของขากรรไกรล่างและบนที่ได้จากการลดทอนรายละเอียดระดับต่าง ๆ จะเห็นได้ว่าแบบจำลองที่ถูกลดทอนรายละเอียดแล้วจะมีจำนวนสามเหลี่ยมน้อยลงตามระดับของการลดทอนรายละเอียด และแบบจำลองที่ถูกลดทอนรายละเอียดแล้วยังคงรักษาลักษณะทางกายภาพที่สำคัญไว้คือ สามารถรักษาโครงสร้างของฟันแต่ละซี่ได้ แต่จะมีการสูญเสียโครงสร้างฟันมากในกรณีที่กระทำการลดทอนรายละเอียดที่เปอร์เซ็นต์สูงเช่น 80% ถึง 90% การลดทอนรายละเอียด ซึ่งสังเกตุได้จากรูปแสดงเปอร์เซ็นต์การลดทอนรายละเอียดที่ 90% ของแบบจำลองของขากรรไกรล่าง พบว่าโครงสร้างฟันสูญเสียไป 3 ซี่ ดังนั้นเปอร์เซ็นต์การลดทอนรายละเอียดที่เหมาะสมและสามารถเพิ่มประสิทธิภาพในการแสดงผลและประมวลผลคือ อยู่ในช่วงระหว่าง 50% ถึง 60%



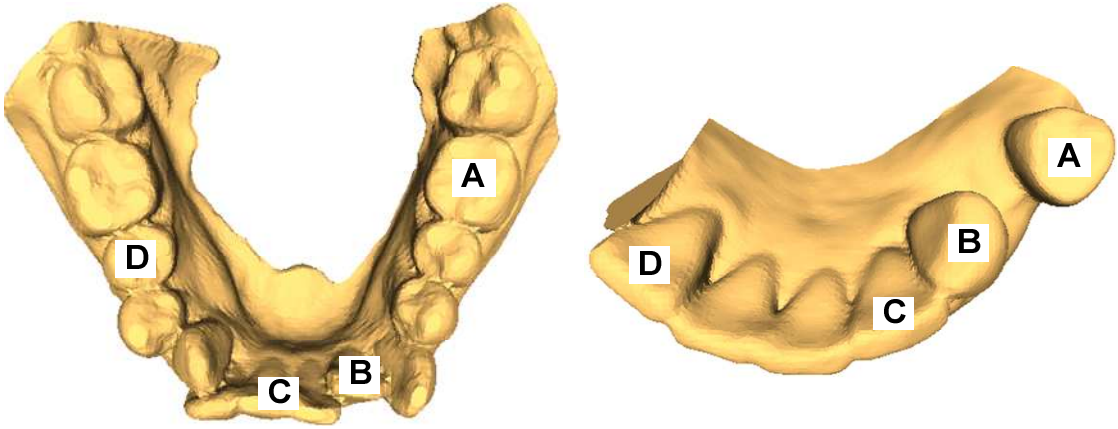
รูปที่ 4.4 เปรียบเทียบแบบจำลองขากรรไกรล่างที่ถูกลดทอนรายละเอียดระดับต่าง ๆ



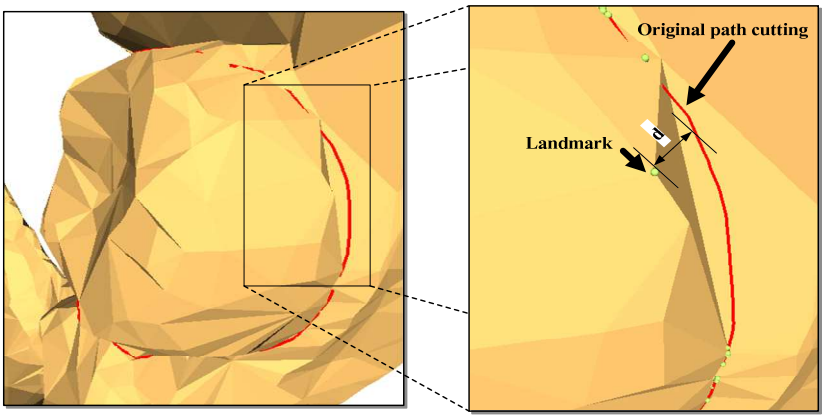
รูปที่ 4.5 เปรียบเทียบแบบจำลองขากรรไกรบนที่ถูกลดทอนรายละเอียดระดับต่าง ๆ

4.2 ผลการทดสอบการตัดพื้นผิวของวัตถุ (Surface cutting)

จากขั้นตอนการลดทอนรายละเอียดของวัตถุ นำแบบจำลองที่ได้มากระทำการทดสอบการตัดพื้นผิว โดยเลือกฟัน 4 ซี่ของแบบจำลองขากรรไกรล่างและบน แสดงดังรูปที่ 4.6 ประสิทธิภาพของขั้นตอนวิธีนี้พิจารณาจากค่าความผิดพลาดของการตัดพื้นผิวที่เกิดขึ้น ซึ่งวัดได้จากระยะห่างเฉลี่ยของจุด landmark บนพื้นผิวของแบบจำลองที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดที่ได้จากแบบจำลองเดิม ดังรูปที่ 4.7

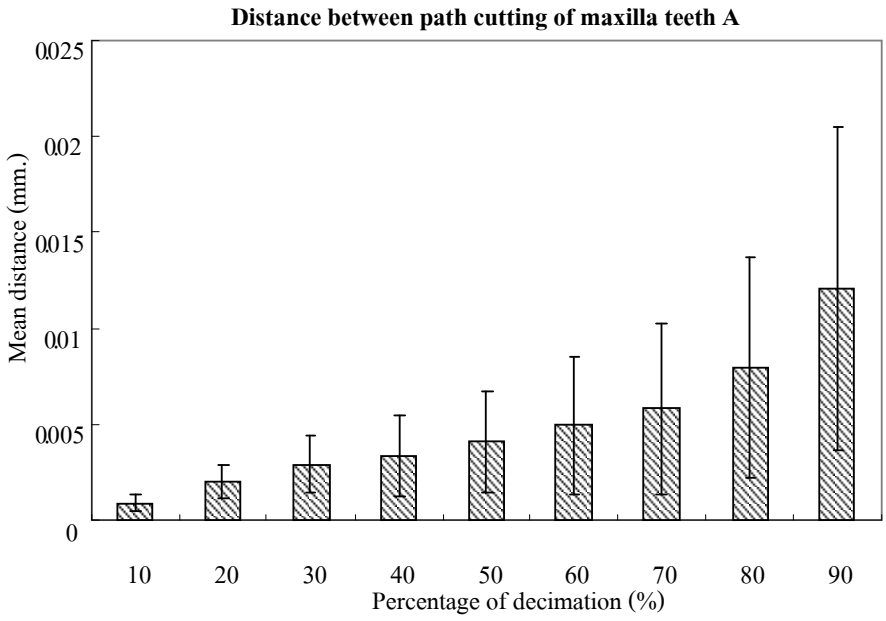


รูปที่ 4.6 ฟันแต่ละซี่ของแบบจำลองขากรรไกรล่าง (ขวา) และบน (ซ้าย) ที่ใช้ทดสอบการตัดพื้นผิว

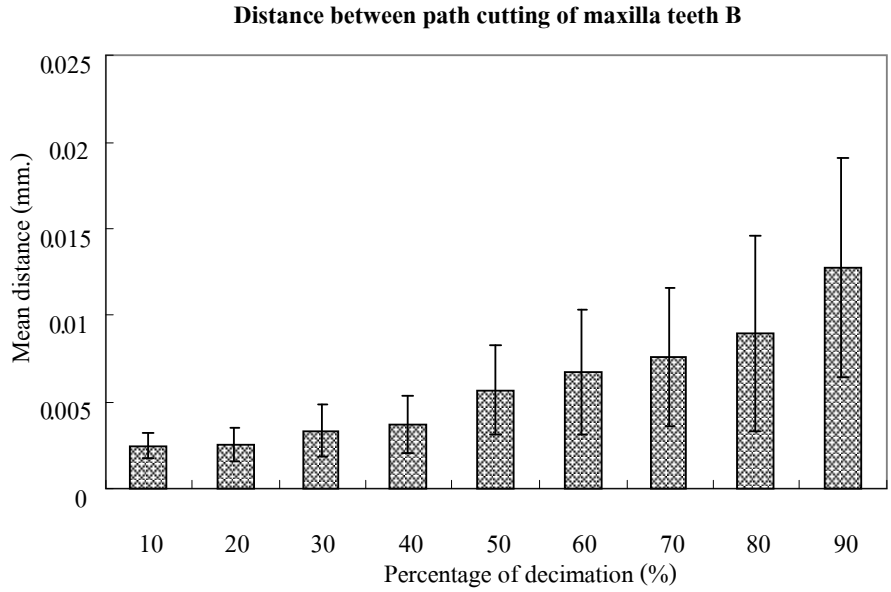


รูปที่ 4.7 แสดงระยะห่างระหว่างจุด landmark บนพื้นผิวของแบบจำลองที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม

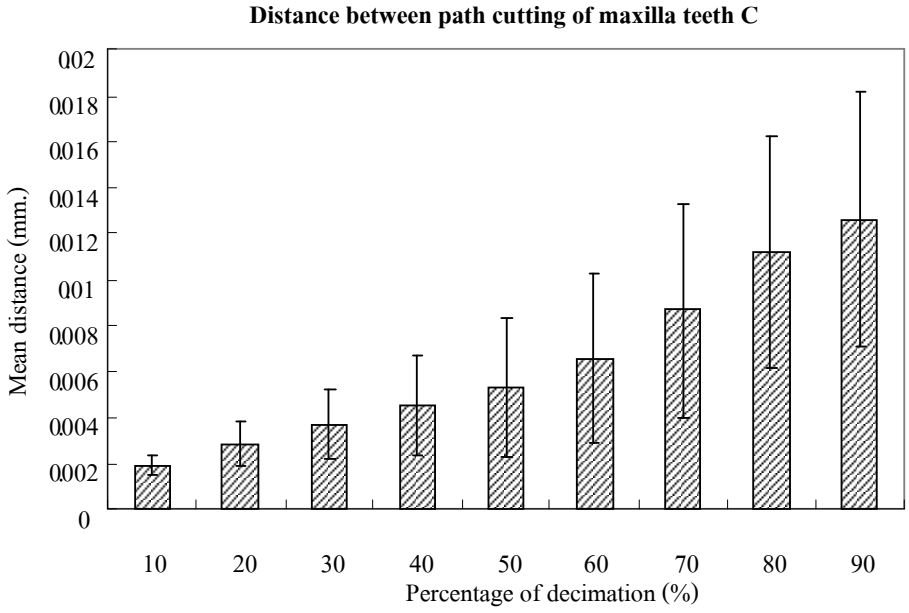
จากรูปที่ 4.8–4.11 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันแต่ละซี่ของแบบจำลองขากรรไกรบนที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม และจากรูปที่ 4.12–4.15 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันแต่ละซี่ของแบบจำลองขากรรไกรบนที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม จะพบว่าค่าระยะห่างดังกล่าวจะเพิ่มขึ้นตามเปอร์เซ็นต์การลดทอนรายละเอียด กรณีที่ระยะห่างมีค่าแตกต่างกันมากเมื่อเทียบกับเปอร์เซ็นต์การลดทอนรายละเอียดที่ใกล้เคียงกันเช่น ดังรูปที่ 4.12 ค่าระยะห่างระหว่าง 70% กับ 80% สาเหตุเกิดจากโครงสร้างฟัน สังเกตุว่าฟัน A มีลักษณะที่มีโคนฟันล้อมรอบไม่ได้ติดกับฟันซี่อื่น ทำให้การลดทอนรายละเอียดบริเวณโคนฟันจะมีมากกว่า เมื่อลดทอนรายละเอียดที่ 80% จะแตกต่างจาก 70% มาก



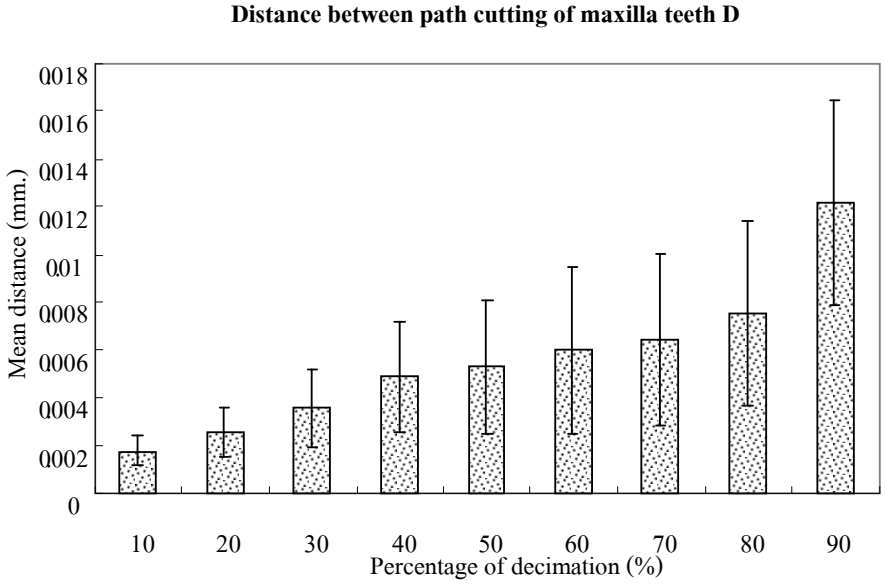
รูปที่ 4.8 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันซี่ A ของแบบจำลองขากรรไกรบนที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม



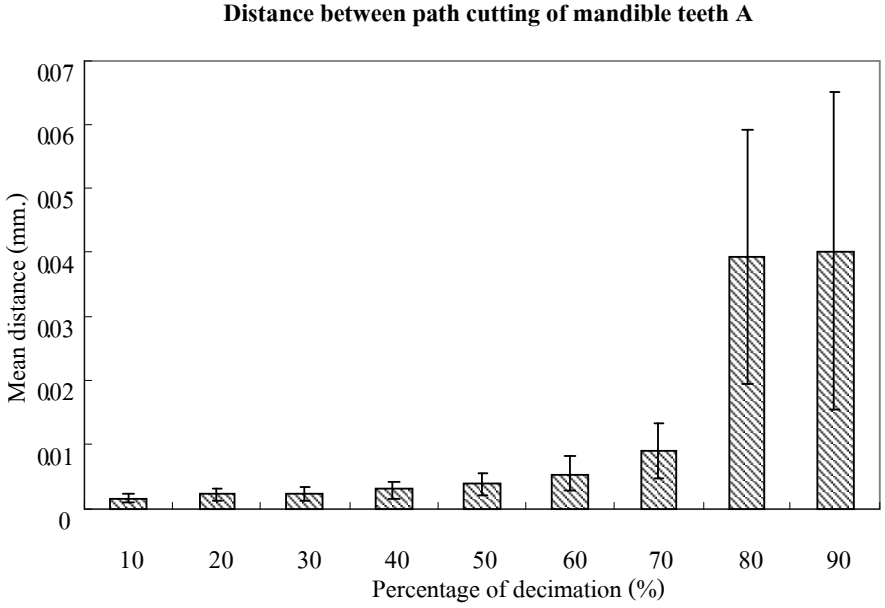
รูปที่ 4.9 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันซี่ B ของแบบจำลองขากรรไกรบนที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม



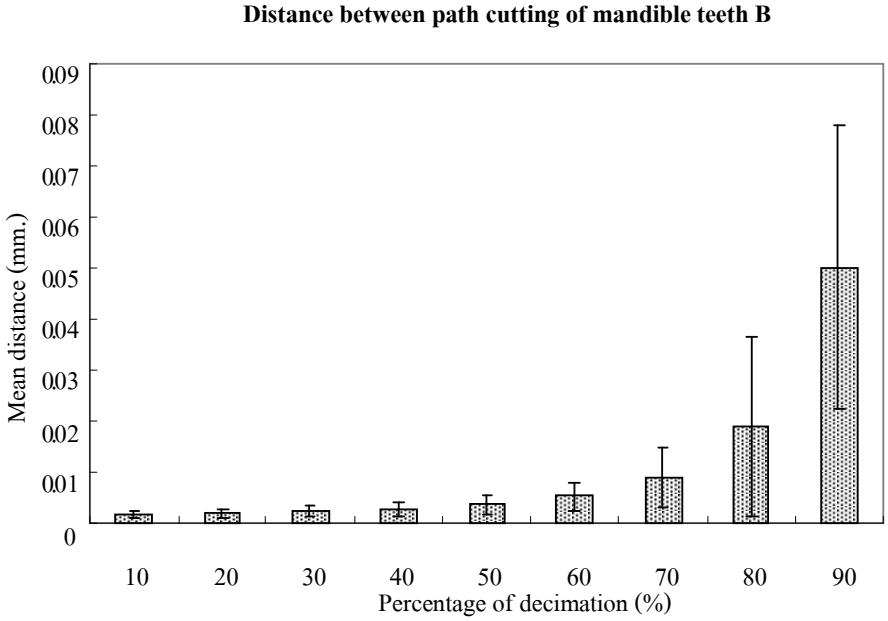
รูปที่ 4.10 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันซี่ C ของแบบจำลองขากรรไกรบนที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม



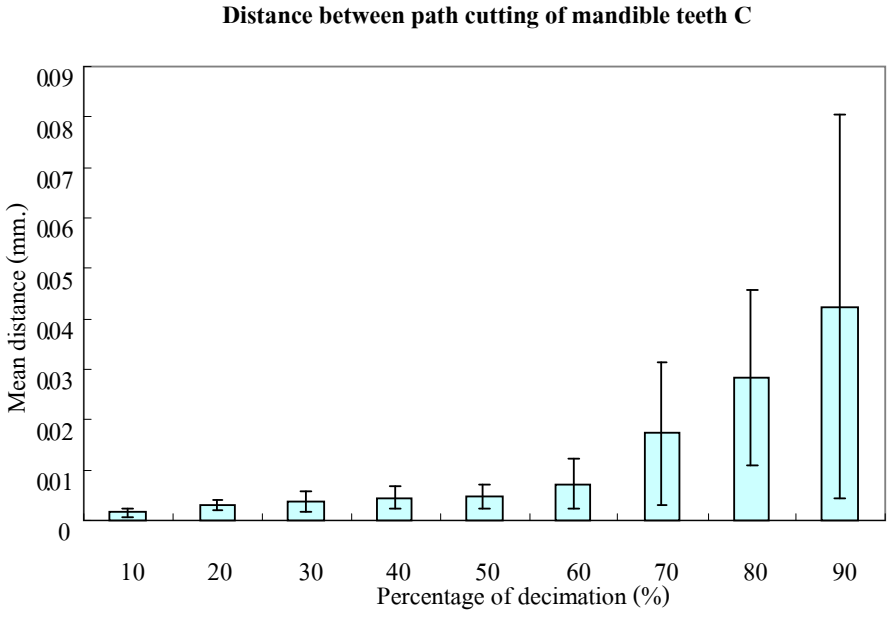
รูปที่ 4.11 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันซี่ D ของแบบจำลองขากรรไกรบนที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม



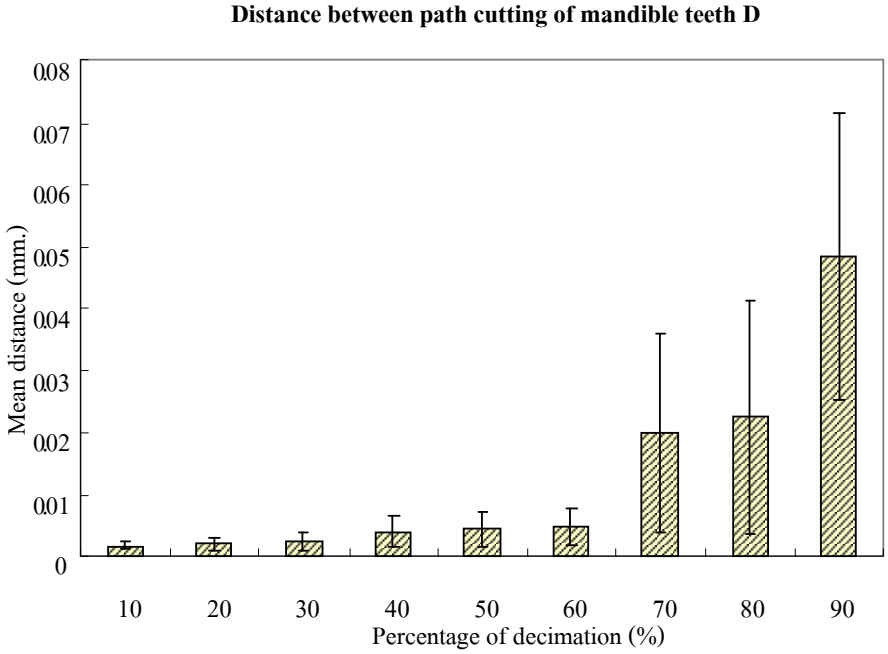
รูปที่ 4.12 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันซี่ A ของแบบจำลองขากรรไกรล่างที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม



รูปที่ 4.13 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันซี่ B ของแบบจำลองขากรรไกรล่างที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม

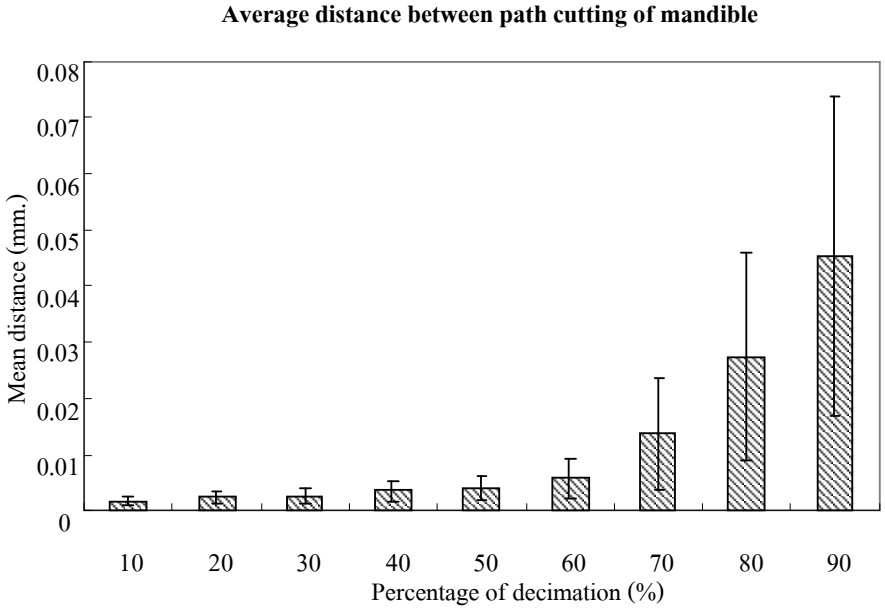


รูปที่ 4.14 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันซี่ C ของแบบจำลองขากรรไกรล่างที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม

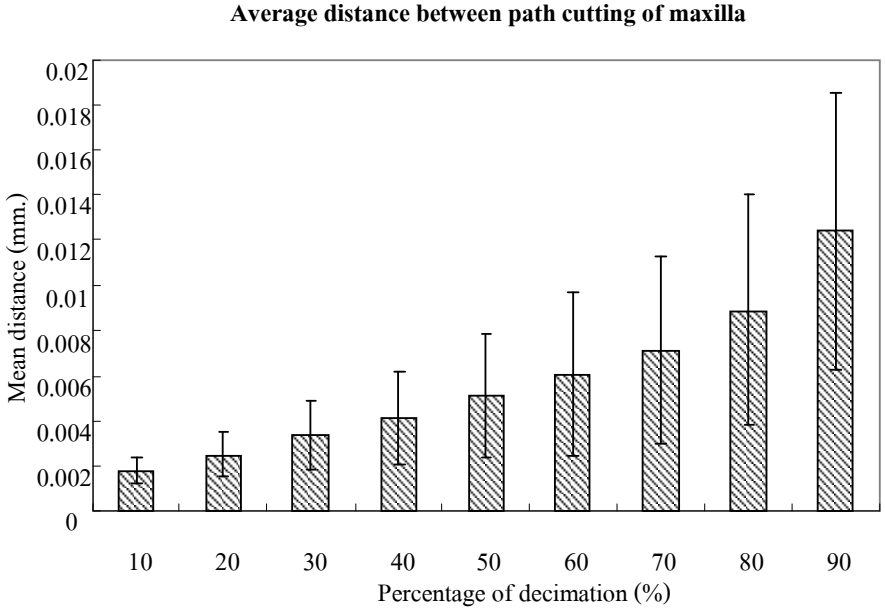


รูปที่ 4.15 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันซี่ D ของแบบจำลองขากรรไกรล่างที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม

รูปที่ 4.16 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันทั้งหมดที่ใช้ในการทดสอบการตัดพื้นผิวของแบบจำลองขากรรไกรล่างที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม และรูปที่ 4.17 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันทั้งหมดที่ใช้ในการทดสอบการตัดพื้นผิวของแบบจำลองขากรรไกรบนที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม จะเห็นได้ว่าแบบจำลองขากรรไกรล่างระยะห่างดังกล่าวจะค่อย ๆ เพิ่มขึ้นเพียงเล็กน้อยในช่วงเปอร์เซ็นต์การลดทอนรายละเอียดที่ 10% ถึง 60% แล้วจึงเพิ่มมากขึ้นในช่วงถัดไป ขณะที่แบบจำลองขากรรไกรบนระยะห่างนั้นจะเพิ่มขึ้นอย่างสม่ำเสมอ



รูปที่ 4.17 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันทั้ง 4 ซี่ของแบบจำลองขากรรไกรล่างที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม



รูปที่ 4.18 แสดงระยะห่างเฉลี่ยระหว่างจุด landmark บนพื้นผิวของฟันทั้ง 4 ซี่ของแบบจำลองขากรรไกรบนที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิม

4.3 ผลการทดสอบการสร้างแบบจำลองมวลสปริง (Mass-spring model)

เป็นการทดสอบการเปลี่ยนแปลงของแบบจำลองเหงือกเมื่อมีการจัดฟัน ซึ่งกระบวนการนี้ใช้ขั้นตอนวิธีการสร้างแบบจำลองเชิงมวลสปริง กระบวนการดังกล่าวจำเป็นต้องมีการแก้สมการการเคลื่อนที่ของมวลปริงหรือจุดบนโครงข่ายของแบบจำลองเหงือก ในการทดสอบจะใช้วิธีการของออยเลอร์ และรุงเก-คุตตาอันดับสี่ เพื่อเปรียบเทียบประสิทธิภาพของวิธีการทั้งสอง ซึ่งพิจารณาได้จากเวลาในการประมวลผลและความผิดพลาดที่เกิดขึ้น โดยที่เวลาที่วัดนั้นจะรวมถึง เวลาในการคำนวณหาแรงที่กระทำ เวลาที่ใช้แก้สมการการเคลื่อนที่ และเวลาที่ใช้แสดงผล

ตารางที่ 4.3 และตารางที่ 4.4 แสดงการทดสอบกับแบบจำลองขากรรไกรล่างด้วยวิธีของออยเลอร์ และวิธีรุงเก-คุตตาอันดับสี่ ในแบบจำลองที่มีรายละเอียดต่าง ๆ เวลาที่ใช้เลื่อนระดับ (time step) มีค่าเท่ากับ 0.00001 วินาที จากการทดสอบพบว่าเวลาที่ใช้ในการประมวลผลของแบบจำลองที่มีรายละเอียดต่ำจะน้อยกว่าเวลาของแบบจำลองที่มีรายละเอียดสูง อีกทั้งจะพบว่าเวลาของวิธีของออยเลอร์จะน้อยกว่าวิธีรุงเก-คุตตาอันดับสี่

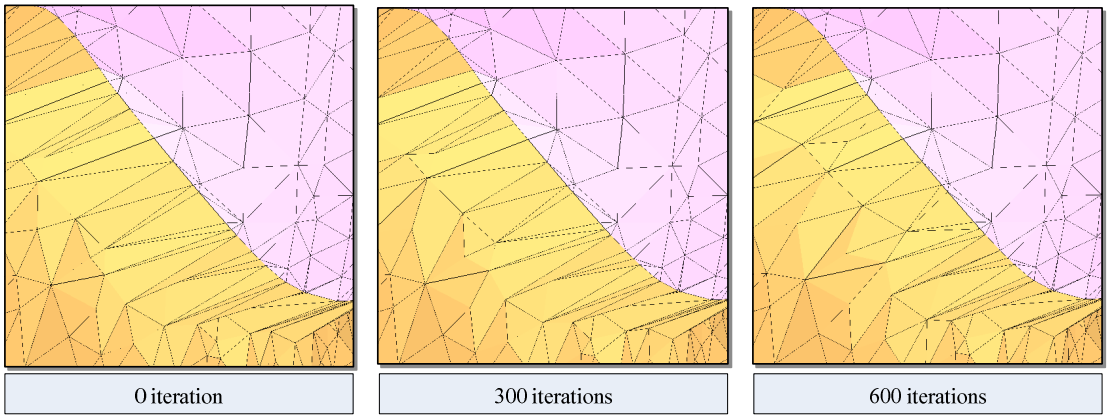
ตารางที่ 4.3 แสดงผลการทดสอบการสร้างแบบจำลองมวลสปริงด้วยวิธีออยเลอร์

Number of Triangles	Time for 100 frames (ms)	Time per frame (ms)	Frames per second
29108	38.27	0.38	2613
26198	36.62	0.37	2730
23286	31.10	0.31	3215
14552	21.11	0.21	4737
11642	19.80	0.19	5050
5820	13.61	0.14	7348

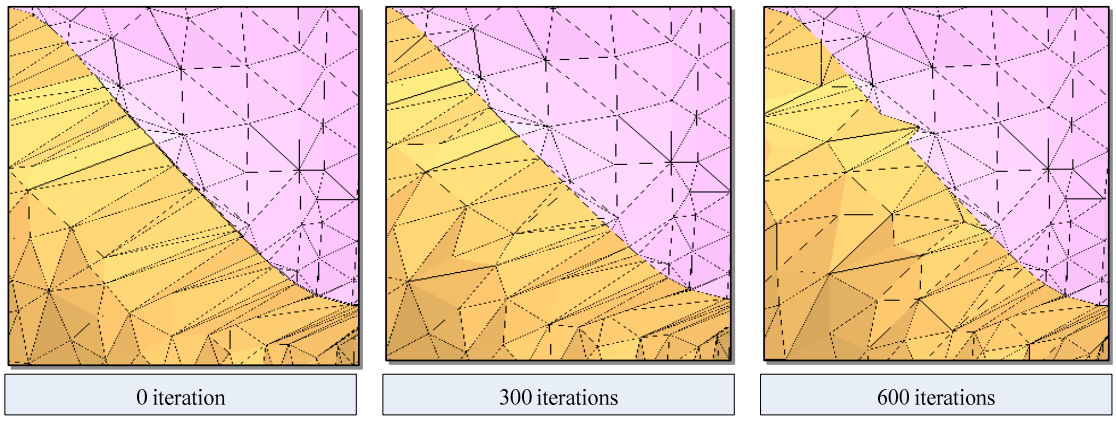
ตารางที่ 4.4 แสดงผลการทดสอบการสร้างแบบจำลองมวลสปริงด้วยวิธีรุงเก-คุตตาอันดับสี่

Number of Triangles	Time for 100 frames (ms)	Time per frame (ms)	Frames per second
29108	55.65	0.56	1796
26198	52.51	0.53	1904
23286	45.23	0.45	2210
14552	29.45	0.29	3395
11642	26.09	0.26	3832
5820	16.38	0.16	6101

รูปที่ 4.19 และรูปที่ 4.20 จำลองทดสอบการเปลี่ยนแปลงรูปร่างของแบบจำลองเหงือกซึ่งใช้วิธีออยเลอร์ และวิธีรุงเก-คุตตาอันดับสี่ ในรอบต่าง ๆ จะเห็นเมื่อจำนวนรอบของการทำงานเพิ่มขึ้นแบบจำลองของเหงือกก็จะเลื่อนตำแหน่งไปยังตำแหน่งสมดุล สังเกตว่าแบบจำลองเหงือกที่ใช้วิธีวิธีรุงเก-คุตตาอันดับสี่ จะมีลักษณะที่เหมือนจริงกว่าแบบจำลองที่ได้จากวิธีออยเลอร์ โดยพิจารณาจากลักษณะของสามเหลี่ยมที่มีขนาดใกล้เคียงกัน



รูปที่ 4.19 แสดงการเปลี่ยนแปลงรูปร่างของแบบจำลองเหงือกซึ่งใช้วิธีออยเลอร์ในแต่ละรอบ



รูปที่ 4.20 แสดงการเปลี่ยนแปลงรูปร่างของแบบจำลองเหงือกซึ่งใช้รุงเก-คุตตาอันดับสี่ในแต่ละรอบ

4.4 การอภิปรายผล

ในการทดสอบการจำลองจัดฟันในแต่ละขั้นตอนวิธีต่าง ๆ โดยเริ่มตั้งแต่การลดทอนรายละเอียดของวัตถุ เพื่อช่วยเพิ่มประสิทธิภาพในการแสดงผลและประมวลผล การตัดพื้นผิว เป็นกระบวนการแยกองค์ประกอบของฟันและเหงือกออกจากกัน และขั้นตอนสุดท้ายคือ การสร้างแบบจำลองมวลสปริง เพื่อจำลองการเปลี่ยนแปลงรูปร่างของแบบจำลองเหงือกเมื่อมีการจัดฟัน สำหรับการลดทอนรายละเอียดที่เหมาะสมคือ 50% ถึง 60% เพราะว่าถ้าลดทอนรายละเอียดน้อยไปกว่านี้ก็จะไม่ส่งผลกระทบมากนักกับเวลาที่ใช้ในการประมวลผลหรือถ้าลดทอนรายละเอียดมากไปกว่านี้ก็จะได้แบบจำลองที่สูญเสียโครงสร้างฟันมากขึ้นด้วย ส่วนการตัดพื้นผิว สาเหตุที่ค่าระยะห่างระหว่างจุด landmark บนพื้นผิวของฟันทั้งหมดที่ใช้ในการทดสอบการตัดพื้นผิวของแบบจำลองที่ถูกลดทอนรายละเอียดกับเส้นทางการตัดของแบบจำลองเดิมที่เพิ่มขึ้น เนื่องจากว่าบริเวณรอบเส้นทางการตัดมีการลดทอนรายละเอียดด้วย ทำให้สามเหลี่ยมที่อยู่บนเส้นทางการตัดเดิมหายไป ค่าผิดพลาดถือว่ายอมรับได้ในช่วงที่ไม่เกิน 60 เปอร์เซ็นต์การลดทอนรายละเอียดเพราะมีค่าน้อยมาก ในขั้นตอนการจำลองการเปลี่ยนแปลงรูปร่างของเหงือกด้วยวิธีการของออยเลอร์ และรุงเก-คุตตาอันดับสี่ ก็มีข้อดีและข้อเสียที่ต่างกันคือ วิธีออยเลอร์จะใช้เวลาประมวลน้อยกว่าวิธีรุงเก-คุตตาอันดับสี่ เนื่องจากว่าวิธีรุงเก-คุตตาอันดับสี่ ต้องแก้สมการหลายครั้งกว่า แต่วิธีรุงเก-คุตตาอันดับสี่ มีข้อดีคือ คำนวณแล้วได้ค่าผิดพลาดที่น้อยกว่า

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

การพัฒนาเครื่องมือช่วยงานทางการแพทย์ โดยเฉพาะการพัฒนาระบบภาพทางการแพทย์ภายในประเทศไทย เพื่อช่วยงานเฉพาะด้าน เช่น ด้านทันตกรรม ยังมีไม่มากนัก ซึ่งต้องข้ามกับความต้องการในกาเข้ารับการบำบัดรักษาทางทันตกรรม เพื่อแก้ไขปัญหาช่องปากและบุคคลิกภาพ ดังนั้นงานวิจัยนี้จึงมุ่งที่จะศึกษาและพัฒนาขั้นตอนวิธีเพื่อช่วยงานทางทันตกรรมจัดฟัน โดยใช้คอมพิวเตอร์เข้ามาช่วยในขั้นตอนการวิเคราะห์และวางแผนรักษาทางทันตกรรมจัดฟัน เครื่องมือนี้จะทำการแสดงผลภาพในรูปแบบสามมิติ เพื่อช่วยให้มีความเสมือนจริงกับฟันของผู้เข้ารับการรักษาและสามารถจำลอง รูปแบบการจัดเรียงของฟันแต่ละระยะว่าจะเปลี่ยนไปอย่างไรระหว่างการบำบัดรักษา นอกจากนี้ข้อมูลที่ได้จากการวิเคราะห์สามารถนำไปช่วยในการออกแบบวัสดุจัดฟันได้ ฉะนั้นวัตถุประสงค์ของงานวิจัยนี้คือ เพื่อศึกษาและพัฒนาเทคนิคการประมวลผลภาพทางการแพทย์ใน 3 มิติ เพื่อจำลองการจัดฟัน

ขั้นตอนการดำเนินงานวิจัยนี้จะแบ่งออกเป็นสามส่วน โดยส่วนแรกคือ การศึกษาข้อมูลเกี่ยวกับการจำลองการจัดฟัน เพื่อหาข้อมูลขั้นตอนวิธีที่ใช้ในการจำลองการจัดฟัน เพื่อที่จะนำมาออกแบบระบบ ส่วนที่สองคือการพัฒนา และทดสอบประสิทธิภาพของเครื่องมือ ซึ่งขั้นตอนวิธีที่เกี่ยวข้องกับกระบวนการจำลองการจัดฟัน คือ การลดทอนรายละเอียดของวัตถุ การตัดพื้นผิว และการสร้างแบบจำลองมวลสปริง เนื่องจากข้อมูลของแบบจำลองทางการแพทย์ส่วนใหญ่แล้วจะมีความละเอียดสูง ดังนั้นการลดทอนรายละเอียดของวัตถุจึงมีความจำเป็น เพราะส่วนแก้ปัญหาการแสดงผลและประมวลผล ส่วนขั้นตอนการตัดพื้นผิว ใช้ในกระบวนการแยกองค์ประกอบของฟันและเหงือกออกจากกัน เพื่อนำไปจัดเรียง และกระบวนการสุดท้ายคือ การสร้างแบบจำลองมวลสปริง เพื่อวิเคราะห์และจำลองการเปลี่ยนแปลงรูปร่างของเหงือกเพื่อให้การจัดฟันดูเสมือนจริงยิ่งขึ้น

5.1 สรุปผลการวิจัย

5.1.1 สรุปผลการทดสอบประสิทธิภาพของขั้นตอนวิธีการลดทอนรายละเอียดของวัตถุ

เป็นกระบวนการลดจำนวนจุด และสามเหลี่ยมบนโครงข่าย เพื่อช่วยลดเวลาในการแสดงผล และประมวลผล อีกทั้งยังช่วยลดเนื้อที่ในการจัดเก็บข้อมูลด้วย โดยประยุกต์จากงานวิจัยของ Schroeder et al. การพิจารณาจุดที่จะถูกลบออกโครงข่ายนั้นพิจารณาจากประเภทของจุด และระยะห่างระหว่างจุดนั้นกับระนาบเฉลี่ย การวัดประสิทธิภาพของการลดทอนรายละเอียด สามารถ

พิจารณาได้จากจำนวนจุด และสามเหลี่ยมที่ถูกลบออก และที่สำคัญคือ เมื่อลดทอนรายละเอียดแล้ว ต้องรักษาลักษณะทางกายภาพที่สำคัญไว้ จากการทดสอบพบว่า ขั้นตอนวิธีการลดทอนรายละเอียด สามารถลดทอนรายละเอียดของวัตถุได้จริงตามจำนวนเปอร์เซ็นต์การลดทอนรายละเอียด และยังสามารถรักษาลักษณะทางกายภาพที่สำคัญของแบบจำลองฟันไว้ได้ แต่ก็มีการสูญเสียโครงสร้างของฟันบ้างเมื่อมีการลดทอนรายละเอียดที่มากเกินไป ซึ่งจะเกิดขึ้นได้กับแบบจำลองที่มีรายละเอียดน้อย เช่น แบบจำลองขากรรไกรล่าง สามารถสรุปได้ว่าเปอร์เซ็นต์การลดทอนรายละเอียดที่เหมาะสม และมีประสิทธิภาพอยู่ในช่วง 50% ถึง 60% การลดทอนรายละเอียด เพราะสามารถรักษาลักษณะที่สำคัญของแบบจำลองนั้นไว้ได้ และยังช่วยให้การจำลองการจัดฟันมีประสิทธิภาพด้วย คือ ช่วยลดเวลาในการแสดงผล และประมวลผล รวมทั้งให้ผลการทดสอบการขั้นตอนวิธีอื่น ๆ ที่ดี เช่น ผลการทดสอบกับขั้นตอนวิธีการตัดพื้นผิวให้ค่าความผิดพลาดเป็นที่น่าพอใจ เพราะถ้าลดทอนรายละเอียดมากกว่านี้จะทำให้ค่าผิดพลาดเพิ่มขึ้นมาก

5.1.2 สรุปผลการทดสอบประสิทธิภาพของขั้นตอนวิธีการตัดพื้นผิว

การตัดพื้นผิวเป็นกระบวนการแยกองค์ประกอบของฟันและเหงือกออกจากกัน ขั้นตอนวิธีการตัดพื้นผิว เริ่มต้นจากการกำหนดเส้นทางการตัดรอบโครงสร้างฟันแต่ละซี่ ซึ่งกำหนดได้จากการกำหนดจุดลงบนพื้นผิวของแบบจำลองฟัน หลังจากนั้นก็จะทำการแยกฟันที่ตัดและเหงือกออกจากกันด้วยวิธีการแยกองค์ประกอบแบบ Region Growing การวัดประสิทธิภาพสามารถพิจารณาได้จากรอยตัดที่ได้มีความเรียบหรือไม่เมื่อมีการลดทอนรายละเอียดของแบบจำลอง โดยการวัดระยะห่างระหว่างจุด landmark บนพื้นผิวของแบบจำลองฟันที่ลดทอนรายละเอียดแล้วกับเส้นทางการตัดที่ได้จากการทดสอบกับแบบจำลองฟันที่ไม่ได้ทำการลดทอนรายละเอียด จากผลการทดสอบกับแบบจำลองฟันที่รายละเอียดต่าง ๆ พบว่า ระยะห่างดังกล่าวจะมีค่าเพิ่มขึ้นเมื่อแบบจำลองฟันมีรายละเอียดที่น้อยลง สาเหตุเกิดจากการลดทอนรายละเอียด เพราะเมื่อลบจุดออกจากโครงข่าย จำเป็นต้องมีการสร้างสามเหลี่ยมขึ้นมาใหม่ ซึ่งอาจจะเป็นสามเหลี่ยมจะจุด landmark วางตัวอยู่ภายในหรืออยู่บนด้านของสามเหลี่ยมนั้น ถ้าแบบจำลองฟันมีรายละเอียดที่ไม่ซับซ้อน เช่น แบบจำลองของขากรรไกรล่าง ระยะห่างที่ได้จะมีค่าเพิ่มขึ้นไม่มากนัก แต่ถ้าแบบจำลองฟันมีความซับซ้อนมาก เช่น แบบจำลองของขากรรไกรบน ระยะห่างที่ได้จะเพิ่มมากขึ้นตามรายละเอียดที่น้อยลงของแบบจำลองนั้น แต่ระยะห่างเฉลี่ยที่ได้จากการทดสอบกับแบบจำลองขากรรไกรล่าง และบนนั้นถือว่ามีค่าไม่มากนัก แสดงว่ารอยตัดที่ได้ก็มีความเรียบอยู่บ้าง

5.1.3 สรุปผลการทดสอบประสิทธิภาพของขั้นตอนวิธีการสร้างแบบจำลองมวลสปริง

การจำลองการเปลี่ยนแปลงของแบบจำลองเหงือกเมื่อมีการจัดเรียงฟันด้วยวิธีการสร้างแบบจำลองมวลปริง มวล คือ จุดบนโครงข่ายของแบบจำลอง และสปริง คือ ด้านของสามเหลี่ยม เริ่มต้นจากขั้นตอนการกำหนดจุดที่เกิดแรงกระทำ คือ จุดของแบบจำลองเหงือกที่ติดกับแบบจำลองฟันที่เลื่อนตำแหน่ง จากนั้นคำนวณหาตำแหน่งใหม่ของจุดบนแบบจำลองเหงือกคำนวณได้จากการแก้สมการการเคลื่อนที่ ซึ่งเป็นการแก้สมการเพื่อหาค่าอนุพันธ์ งานวิจัยนี้จะทำการเปรียบเทียบประสิทธิภาพของวิธีที่ใช้ในการแก้สมการการเคลื่อนที่ คือ วิธีออยเลอร์ และวิธีรุงเก-คุตตาอันดับสี่ จากการทดสอบพบว่า วิธีออยเลอร์จะใช้เวลาในการประมวลผลน้อยกว่าวิธีรุงเก-คุตตาอันดับสี่ แต่จะมีค่าความผิดพลาดในการคำนวณมากกว่า เพราะวิธีออยเลอร์ใช้การคำนวณหาค่าอนุพันธ์เพียงครั้งเดียว ขณะที่วิธีรุงเก-คุตตาอันดับสี่คำนวณหาค่าอนุพันธ์ถึงสี่ครั้ง สรุปได้ว่าวิธีออยเลอร์เหมาะสมสำหรับใช้ในการวิเคราะห์การเปลี่ยนแปลงรูปร่างของแบบจำลองเหงือก เพราะใช้เวลาในการประมวลผลน้อย ถึงแม้จะมีความผิดพลาดบ้างแต่ก็ไม่มากนัก

5.2 ข้อจำกัดของงานวิจัย

งานวิจัยนี้เป็นงานวิจัยเฉพาะด้าน คือ เน้นด้านการช่วยงานทางทันตกรรมจัดฟัน ดังนั้นจึงมีข้อจำกัด ดังนี้

1) สนับสนุนแฟ้มข้อมูลประเภท STL เท่านั้น คือ สามารถเปิด บันทึก แฟ้มข้อมูลในรูปแบบ STL ไบนารี
2) การกำหนดจุด landmark ต้องกำหนดตามลำดับ ไม่สามารถกำหนดสลับจุดไปมาได้
3) ช่วงเวลาที่ใช้ในการจำลองการเปลี่ยนแปลงรูปร่างของแบบจำลองเหงือกต้องมีค่าน้อย ๆ เช่น 0.00001 วินาที เป็นต้น เพราะวิธีที่ใช้แก้สมการการเคลื่อนที่เหมาะสำหรับช่วงเวลาน้อย ๆ ถ้าใช้ช่วงเวลามาก ๆ จะทำให้เกิดค่าความผิดพลาดมากขึ้นด้วย

5.3 ข้อเสนอแนะในงานวิจัยต่อไป

สำหรับงานการจำลองการจัดฟัน ประกอบด้วยขั้นตอนวิธีต่าง ๆ ที่สำคัญคือ การลดทอนรายละเอียดของวัตถุ การตัดพื้นผิว และการสร้างแบบจำลองมวลสปริง ขั้นตอนวิธีดังกล่าวสามารถนำไปใช้ประโยชน์ด้านต่าง ๆ ได้อีกมากมาย ดังนั้นเทคนิคการจำลองการจัดฟันโดยการประมวลผลภาพ 3 มิติ จึงน่าจะเป็นประโยชน์แก่นักวิจัยท่านอื่นที่สนใจที่จะพัฒนาองค์ความรู้เพื่องานการประมวลผลภาพทางการแพทย์หรือด้านอื่น ๆ โดยแนวทางวิจัยที่จะพัฒนาขั้นตอนวิธีต่าง ๆ สามารถทำได้หลายแนวทางด้วยกัน อาทิ เช่น

1) การวิจัยเพื่อเพิ่มประสิทธิภาพของขั้นตอนการลดทอนรายละเอียดของวัตถุ โดยการปรับปรุงเทคนิคการพิจารณาจุดที่ลบออกจากโครงข่ายเพื่อเพิ่มประสิทธิภาพการลดทอนรายละเอียดให้ดียิ่งขึ้น
2) การศึกษาการตัดวัตถุแบบทั้งก้อน (Volume Cutting) เพื่อนำไปใช้ประโยชน์ในการผ่าตัดศัลยกรรมต่าง ๆ
3) การศึกษาการกำหนดขอบเขตการตัด ที่สะดวกต่อการใช้งานของผู้ใช้ เช่น พิจารณาจากค่าความโค้งของแบบจำลอง (Curvature Estimation)
4) การพัฒนาขั้นตอนวิธีการสร้างแบบจำลองมวลสปริง ให้สามารถจำลองการเปลี่ยนแปลงรูปร่างของวัตถุอื่น ๆ ได้อย่างมีประสิทธิภาพ คือ เลือกใช้วิธีการแก้สมการการเคลื่อนที่ที่สามารถให้ลดจำนวนค่าผิดพลาด และใช้เวลาในการประมวลผลอย่างมีประสิทธิภาพ

รายการอ้างอิง

Ascher, U. M. and Petzold, L. R. (1998). **Computer method for ordinary differential equations and defferential-algebraic equations**. SIAM, New York.

Baraff, D. and Witkin, A. (1998). **Large steps in cloth simulation**. Proc. ACM SIGGRAPH'98, Orlando, USA.

Beier, T. and Neely, S. (1992). **Feature-based image metamorphosis**. Computer Graphics, vol. 26, pp. 35 - 42.

Bischoff, S. and Kobbelt, L. (2004). **Parameterization free active contour models**. The Visual Computer, vol. 20, pp. 217 - 228.

Boese, L. B., Clinton, J. P., Dennis, D., Golden, C. R. and Kim, B. (2008). **Digital image analysis of Zostera marina leaf injury**. Aquatic Botany, vol. 88, pp. 87 - 90.

Bradley, C. J. (2007). **The Algebra of Geometry**. Highperception Ltd., United Kingdom.

Buryns, C. and Senger, S. (2001). **Interactive cutting of 3D surface meshes**. Computer Graphics, vol. 25, no. 4, pp. 635 - 642.

Castleman, K. R. (1996). **Digital Image Processing**. Prentice Hall, New York.

Chappell, M. J., et al. (2008). **A coupled drug kinetics-cell cycle model to analyse the response of human cells to intervention by topotecan**. Computer Methods and Programs in Biomed., vol. 89, pp. 169 - 178.

Chi, Z., Ning, Z., Chengjun, L. and Guoping, W. (2004). **Marker-controlled perception-based mesh segmentation**. IEEE Proc. 3rd Int. Conf. Image Graphics, Hong Kong, China.

Collins, W. J. and McMillan, T. C. (1992). **Data Structures: An Object-Oriented Approach**. Addison-Wesley, Boston.

Debunne, G., Desburn, M., Cani, M-P. and Barr, A. H. (2001). **Dynamic real-time deformations using space and time adaptive sampling**. Proc. SIGGRAPH, Los Angeles, California USA.

Foley, J. D., Dam A. V., Feiner, S. K. and Hughes, J. F. (1996). **Computer Graphics in C**. 2nd ed., Addison-Wesley, New York.

Garland, M. and Heckbert, P. S. (1998). **Simplifying surfaces with color and texture using quadric error metrics**. IEEE Proc. Visualization, vol. 542, pp. 263 - 269.

Gopi, M. and Manocha, D. (1998). **A unified approach for simplifying polygonal and spline models**. IEEE Proc. Visualization, vol. 542, pp. 271 - 278.

Hairer, E., Lubich, C. and Wanner, G. (2006). **Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations**. 2nd ed., Springer-Verlag, Berlin.

Hayashibe, M., Suzuki, N., Hashizume, M., Konishi, K. and Hattori, A. (2006). **Robotic surgery setup simulation with the integration of inverse-kinematics computation and medical imaging**. Com. Methods Prog. Biomed., vol. 83, pp. 63 - 72.

Hill, F. S., Jr. (2001). **Computer Graphics: Using OpenGL**. 2nd ed., Prentice Hall, New Yotk.

Hoppe, H. (1998). **Progressive meshes**. Computer Graphics, vol. 30, pp. 99 - 108.

Hoppe, H., DeRose, T., Duchamp, T., MaDonald, J. and Stuetzle, W. (1993). **Mesh optimization**. Computer Graphics, vol. 27, pp. 19 - 26.

HuyViet, H. Q., Kamada, T. and Tanaka, H. T. (2006). **An adaptive 3D surface mesh cutting operation**. Proc. 4th Int. Conf. AMDO., Mallarca, Spain.

Ji, Z., Liu, L., Chen, Z. and Wang, G. (2006). **Easy mesh cutting**. Computer Graphics Forum, vol. 25, no. 3, pp. 283 - 291.

Kass, M., Witkin, A. and Terzopoulos, D. (1988). **Snakes: Active contour models**. Int. J. Computer Vision, vol. 1, no. 4, pp. 321 - 331.

Kempf, R. and Frazier, C. (1999). **OpenGL reference manual**. 2nd ed., Addison-Wesley, London.

Leadini, A., et al. (2006). **A new software tool for 3D motion analyses of the musculo-skeletal system**. Clinical Biomech., vol. 21, pp. 870 - 879.

Lee, D-J., Schoenberger, R., Archibald, J. and McCollum, S. (2008). **Development of a machine vision system for automatic date grading using digital reflective near-infrared imaging**. J. Food Eng., vol. 86, pp. 388 - 398.

Li, C., Wang, G., Xu, T. and Liu, Y. (2005). **Orthodontic simulation and diagnosis: an enchanced tool for dentist**. IEEE Proc. 27th Medicine Biology Conf., Shanghai, China.

Lin, L. and Ming, F. (2008). **A novel method of high resolution imaging through random perbulation wave front**. Optics Communications, vol. 281, pp. 991 - 997.

Lindenbeck, C. H., Ulmer, H. D., Lavorante, L. P. and Pflug, R. (2002). **TRICUT: a program to clip triangle meshes using the rapid and triangle libraries and the visualization toolkit**. Computer Geosci., vol. 28, pp. 841 - 850.

Lindstrom, P. and Turk, G. (1998). **Fast and memory effient polygonal simplification**. IEEE Proc. Visualization, vol. 544, pp. 279 - 286.

Luz, D., Berry, D. L. and Roos-Serote, M. (2008). **An automated method for tracking clouds in planetary atmospheres**. New Astronomy, vol. 13, pp. 224 - 232.

Ma, D., Lin, F. and Chua, C. K. (2001). **Rapid prototyping applications in medicine. Part 2: STL file generation and case studies**. Int. J. Adv. Manu. Tech., vol. 18, pp. 118 - 127.

Mah, J. and Buman, A. (2001). **Technology to create the three-dimensional patient record**. Seminars in Ortho., vol. 7, no. 4, pp. 251 - 257.

Metaaphanon, N. and Kanongchaiyos, P. (2005). **Real-time cloth simulation for garment CAD**. Proc. 3rd Int. Conf. Computer Graphics Interactive Techniques in Aust. SEA., Dunedin, New Zealand.

Nedel, L-P. and Thalmann, D. (1998). **Real time muscle deformations using mass-spring systems**. Int. Proc. Computer Graphics, Hannover, Germany.

OpenGL Architecture Review Board. (2004). **OpenGL® Reference Manual: The Official Reference Document to OpenGL.** 4th ed., Addison-Wesley, Berlin.

Pintilie, G. and McInerney, T. (2003). **Interactive cutting of the skull for craniofacial surgical planning.** Int. Proc. Biomedical Engineering, Salzburg, Austria.

Poloc, C., Bello, F., Kitney, R. I. and Darzi, A. (2002). **Online multiresolution volumetric mass spring model for real time soft tissue deformation.** Proc. MICCAI, Tokyo, Japan.

Provot, X. (1995). **Deformation constraints in a mass-spring model to describe rigid cloth behaviour.** Proc. Graphics Interface, Qu'ebec, Canada.

Rivara, M. C. (1996). **New mathematical tools and techniques for the refinementand/or improvement of unstructured triangulations.** Proc. 5th Int. Meshing Roundtable.

Rodrigues, F. M., Silva, B. W., Barbosa, G. R., Ribeiro, M. I. and Neto, E. B. (2006). **J-Ortho: an open-source orthodontic treatment simulator.** Proc. SAC., Dijon, France.

Sakas, G. (2002). **Trends in medical imaging: from 2D to 3D**. Computers & Graphics, vol. 26, pp. 577 - 587.

Schendle, S., Montgomery, K., Sorokin, A. and Lionetti, G. (2005). **A surgical simulator for planning and performing repair of cleft lips**. J. Cranio-Maxillofacial Surgery, vol. 33, pp. 223 - 228.

Schroeder, W. J., Martin, K. and Lorensen, B. (2002). **The visualization toolkit**. 3rd ed. Kitware Inc., New York.

Schroeder, W. J., Zarge, J. A. and Lorensen, W. E. (1992). **Decimation of triangle meshes**. Computer Graphics, vol. 26, no. 2, pp. 65 - 70.

Sinthanayothin, C. (2005). **Medical Imaging Aided Orthodontics**. Proc. Int. Conf Thailand Medical Imaging, Bangkok, Thailand.

Sojar, V., et al. (2004). **Liver surgery training and planning in 3D virtual space**. Int. Congress Series, vol. 1268, pp. 390 - 394.

Spencer, J., Dever, K. M., Jaarda, J. M., Roedema, H. W. and Knapp G. J. (1987). **A computer simulation of a dental articulator**.

Swanson, R. K., Alvord, C. E., Jr. and Murray, J. D. (2004). **Dynamics of a model for brain tumors reveals a small window for therapeutic intervention**. Discrete and Con. Dynamics Systems-Series B, vol. 4, no. 1, pp. 289 - 295.

Taylor, S. S., et al. (2008). **Signaling through cAMP and cAMP-dependent protein kinase: Diverse strategies for drug design**. Biochimica et Biophysica Acta, vol. 1784, pp. 16 - 26.

Terzopoulos, D., John, P., Alan, B. and Kurt, F. (1987). **Elastically deformable models**. Computer Graphics, vol. 21, no. 4, pp. 205 - 214.

Thanwadee, S. and Parun, S. (2002). **Reflections of computer use in orthodontics in Thailand**. J. TH.A.O., no. 1, pp. 1 - 4.

The Research Council of Norway. (2006). **Medical imaging for improved patient care** [On-line]. Available: http://www.ntnu.no/io/ECF_foundation.pdf

Thomas, G. B., Weir, M. D., Hass, J. and Giordano, F. R. (2006). **Thomas' Calculus**. 11th ed., Addison-Wesley, New York.

Toropova, K., Basnak, G., Twarock, R., Stockley, G. P. and Ranson, A. N. (2008). **The three-dimensional structure of genomic RNA in bacteriophage MS2: Implications for Assembly**. J. Molecule Bio., vol. 375, pp. 824 - 836.

Trunk, P., Mocnik, J., Trobec, R. and Bersak, B. (2007). **3D heart model for computer simulations in cardiac surgery**. Computers in Bio. and Med., vol. 37, pp. 1398 - 1403.

Vlaskalic, V. and Boyd, R. (2001). **Orthodontic treatment of a mildly crowded malocclusion using the Invisalign System™**. Aust. Orthod. J., vol. 17, no. 1, pp. 41 - 46.

Wang, Z. L., Teo, J. C. M., Chui, C. K. and Ong, S. H. (2005). **Computational biomechanical modelling of the lumbar spine using marching-cubes surface smoothened finite element voxel meshing**. Com. Methods Prog. Biomed., vol. 80, pp. 25 - 35.

Wong, K. C-H., Siu, T. Y-H., Heng, P-A. and Sun, H. (1998). **Interactive volume cutting**. Proc. Graphics Interface'98, Vancouver, British Columbia.

Worn, H. and Muhling, J. (2001). **Computer- and robot-based operation theatre of the future in cranio-facial surgery**. Int. Con. Series, vol. 1230, pp. 753 - 759.

Yun, L. W., et al. (2008). **Identification of different stages of diabetic retinopathy using retinal optical images**. Information Sci., vol. 178, pp. 106 - 121.

ภาคผนวก

บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่

รายชื่อบทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่

ปฐมพงศ์ พันนุรัตน์, สิทธิชัย ป้อมทอง, ปรเมศวร์ ห่อแก้ว, เสาวภาคย์ โสตถิวิรัช และจันทร์จิรา สินทนะโยธิน. 2550. **เทคนิคการแยกองค์ประกอบของภาพและการสร้างพื้นผิวเชิงตัวแปรสำหรับการวิเคราะห์รายละเอียดหลายระดับ (Image Segmentation and Conformal Surface Parameterization for Multi-Resolution Analysis).** งานประชุมวิชาการประจำปี 2550 สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ (NACS 2007), ศูนย์ประชุมอุทยานวิทยาศาสตร์ประเทศไทย จ. ปทุมธานี. 28-29 มีนาคม 2550.

ปฐมพงศ์ พันนุรัตน์, ปรเมศวร์ ห่อแก้ว และจันทร์จิรา สินทนะโยธิน. 2550. **การลดทอนรายละเอียดของวัตถุ 3 มิติเพื่อช่วยงานทันตกรรมจัดฟัน**. The 5th National Conference on Biomedical Engineering (NCBME 2007), โรงแรมทวินเทาว์เวอร์ จ. กรุงเทพมหานคร. 8 กรกฎาคม 2550**.**

Pathomphong Phannurat, Paramate Horkaew, and Chanjira Sinthanayothin. 2007. **3D Surface Cutting Technique Aided Orthodontics**. World Congress on Bioengineering (WACBE 2007), Twin Towers Hotel, Bangkok, Thailand. 9-11 July 2007.

Pathomphong Phannurat, Paramate Horkaew, and Chanjira Sinthanayothin. 2007. **3D Mesh Segmentation and Smooth Surface Extraction through Region Growing**. The International Conference on Engineering, Applied Sciences, and Technology (ICEAST 2007), Swissotel Le Concorde Hotel, Bangkok, Thailand**.** 21-23 November 2007.

ประวัติผู้เขียน

นายปฐมพงศ์ พันนุรัตน์ เกิดเมื่อวันที่ 11 กรกฎาคม พ.ศ. 2526 ที่ อ.สนม จ.สุรินทร์ สำเร็จการศึกษาระดับประถมศึกษาที่โรงเรียนบ้านโนนเปือย สำเร็จการศึกษาระดับมัธยมศึกษาที่โรงเรียนสุรวิทยาคาร จ. สุรินทร์ และเข้าศึกษาในระดับปริญญาตรีในปีการศึกษา 2545 ที่สาขาวิชาวิศวกรรมคอมพิวเตอร์ สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี จ. นครราชสีมา และสำเร็จการศึกษาเมื่อปี พ.ศ. 2548 เกียรตินิยมอันดับ 1 ระหว่างที่ศึกษาในระดับปริญญาตรีได้เข้ารับการฝึกปฏิบัติงานสหกิจศึกษา ณ ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (NECTEC) สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ (สวทช.) กระทรวงวิทยาศาสตร์ จ. ปทุมธานี ในห้อข้อโครงการที่เกี่ยวกับการประมวลผลภาพสามมิติ ทำให้เกิดแจงจูงใจในการเข้าศึกษาต่อในระดับปริญญาโท ทางด้านการประมวลภาพ จึงได้เข้าศึกษาต่อในระดับปริญญาโทสาขาวิชาวิศวกรรมคอมพิวเตอร์ สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี ในปีการศึกษา 2549 ซึ่งได้รับทุนสนับสนุนงานวิจัยจากสถาบันบัณฑิตวิทศาสตร์และเทคโนโลยีไทย (TGIST) สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ (สวทช.) ในหัวข้อโครงการวิจัย เรื่อง การปรับแก้และเปลี่ยนรูปของแบบจำลองฟันใน 3 มิติเพื่อช่วยงานทันตกรรม และได้ทุนฝึกปฏิบัติงานวิจัย ณ ศูนย์เทคโนโลยีทางทันตกรรมขั้นสูง (ADTEC) จากศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (NECTEC) ระหว่างการศึกษาได้รับความอนุเคราะห์อย่างดียิ่งจากคณาจารย์ในสาขาวิชา โดยได้รับความไว้วางใจให้เป็นผู้ช่วยสอนในสาขาวิชาวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยเทคโนโลยีสุรนารี จำนวน 2 รายวิชา คือ ปฏิบัติการดิจิตอล และปฏิบัติการคอมพิวเตอร์โปรแกรมมิง มีผลงานวิจัยตีพิมพ์เผยแพร่จำนวน 4 เรื่อง โดยเป็นผลงานที่ตีพิมพ์ระดับชาติ 2 เรื่อง และผลงานที่ตีพิมพ์ระดับนานาชาติอีก 2 เรื่อง