

การพัฒนาโปรแกรมการสร้างลายเส้นและจำลองแรงสัมผัสของการวาดภาพ
ด้วยสีเทียนโดยใช้อุปกรณ์ตอบสนองต่อแรง

Development of a rendering and force-reflective simulation program for
wax crayon drawing using a force feedback device



รศ.ดร.นพพร โขติศักดิ์



H002284

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 2 ปีการศึกษา 2547
คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ชื่อหัวข้อเรื่อง	การพัฒนา โปรแกรมการสร้างลายเส้นและจำลองแรงสัมผัสของการวาดภาพด้วยสีเทียน โดยใช้อุปกรณ์ตอบสนองต่อแรง
นักศึกษา	นายไพรัตน์ ปะตาทะโก
อาจารย์ที่ปรึกษา	รศ.ดร.นพพร โชติกกำธร
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2547

บทคัดย่อ

โครงการพัฒนาระบบงานนี้เป็นการศึกษาและพัฒนาอัลกอริทึมสำหรับการสร้างลายเส้นและจำลองแรงสัมผัสของการวาดภาพด้วยสีเทียน โดยใช้อุปกรณ์ตอบสนองต่อแรง โดยในการศึกษาและพัฒนาจะประกอบไปด้วย 2 ส่วนคือ การจำลองแรงสัมผัสของการวาด และการสร้างลายเส้นสีเทียน การจำลองแรงสัมผัสของการวาดซึ่งจะพิจารณาถึงคุณสมบัติทางกายภาพต่างๆ ของกระดาษและแท่งสีเทียน โดยจะขึ้นอยู่กับหลักการทางฟิสิกส์ และในส่วนของ การสร้างลายเส้นสีเทียนจะเป็นการจำลองทางกราฟิกของปฏิสัมพันธ์ระหว่างแท่งสีเทียนกับพื้นผิวกระดาษ

การพัฒนาและทดสอบใช้ร่วมกับอุปกรณ์จำลองแรงสัมผัส PHANTOM รวมถึง HDAPI ในการเชื่อมต่อระหว่างอุปกรณ์กับผู้พัฒนา และใช้ Microsoft Visual C++ 6.0 พัฒนาบนระบบปฏิบัติการ Microsoft Windows XP

Title	Development of a force-reflective simulation program for wax crayon drawing using a force feedback device
Student	Mr.Pairat Patatako
Advisor	Assoc.Prof.Dr. Nopporn Chotikakamthorn
Level of Study	Master of Science in Information Technology
Major	Information Science
Academic Year	2004

ABSTRACT

The objective of this research project is to develop an algorithm for wax crayon drawing and force-reflecting simulation using a force feedback device. This project is divided into two major parts. In the first one, force-reflecting simulation of wax crayon drawing is presented. In this part, some physical properties of both crayon and paper will be described. The second part is focused on graphic rendering of wax crayon drawing which present simulation of the interaction between wax crayon and paper.

In this project, the force feedback device PHANToM has been used for development and. All tasks of this project were developed and tested on Microsoft Windows XP by HDAPI to interface with Microsoft Visual C++

กิตติกรรมประกาศ

ขอกราบขอบพระคุณ รศ. ดร. นพพร โชติกกำทร อาจารย์ปรึกษาโครงการ ที่กรุณาให้ความรู้ คำปรึกษา และคำแนะนำต่างๆ ในการพัฒนาโครงการนี้ให้สำเร็จลุล่วงไปด้วยดี

ขอกราบขอบพระคุณคณาจารย์ทุกท่านในคณะเทคโนโลยีสารสนเทศที่กรุณาถ่ายทอดความรู้ และเทคนิคต่างๆ ในการศึกษา ตลอดระยะเวลาที่ศึกษาในสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ขอขอบคุณพี่ๆ และเพื่อนๆ ในห้องปฏิบัติการวิจัยสื่อประสมและระบบเสมือน และคณะเทคโนโลยีสารสนเทศทุกคนที่คอยให้คำปรึกษา คำแนะนำ และกำลังใจในยามที่มีปัญหา

สุดท้ายขอกราบขอบพระคุณคุณพ่อ คุณแม่ และบุคคลในครอบครัว ที่ให้กำลังใจและให้การอุปการะมาโดยตลอด ขอกราบขอบพระคุณเป็นอย่างสูง

คุณความดีและประโยชน์ที่ได้รับจากโครงการนี้ ผู้เขียนขอบแต่ผู้มีพระคุณทุกท่าน

ไพรัตน์ ปะตาทะโก

25 มีนาคม 2548

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญภาพ.....	VII
บทที่	
1. บทนำ.....	1
1.1 โครงการพัฒนาระบบงาน.....	1
1.2 แนวทางการพัฒนา.....	2
1.3 วัตถุประสงค์.....	3
1.4 ขอบเขตของโครงการ.....	3
1.5 สภาพแวดล้อมในการดำเนินโครงการ.....	3
1.6 ประโยชน์ที่คาดว่าจะได้รับ.....	4
2. เทคโนโลยีเพื่อการติดต่อกับผู้ใช้.....	5
2.1 อุปกรณ์ตอบสนองต่อแรง.....	5
2.2 3DTouch HDAPI.....	6
2.3 Open Graphic Library (OpenGL).....	7
3. การจำลองแรงต้านในการวาดภาพด้วยสีเขียน.....	13
3.1 แรงต้านที่เกิดจากการวาดภาพ.....	13
3.2 แรงต้านการเคลื่อนที่.....	13
3.3 แรงที่เกิดจากการจำลองลักษณะของกระดาษ.....	14
4. การสร้างลายเส้นสีเขียน.....	18
4.1 การจำลองทางกายภาพ.....	18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

บทที่	หน้า
4.2 แรงเสียดทาน.....	18
4.3 แรงยึดหยุ่น.....	20
4.4 การออกแบบพื้นผิวกระดานและสี่เหลี่ยม.....	23
4.5 การออกแบบกระบวนการวาดสี่เหลี่ยม.....	25
5. การออกแบบโปรแกรมและอัลกอริทึม.....	30
5.1 การออกแบบ โปรแกรมและอัลกอริทึม.....	30
5.2 การออกแบบ โปรแกรม.....	31
5.3 อัลกอริทึมสำหรับการจำลองแรงสัมผัส.....	34
5.4 อัลกอริทึมสำหรับการสร้างลายเส้นสี่เหลี่ยม.....	36
5.5 การออกแบบ user-interface.....	39
6. ผลการพัฒนาและทดสอบ โปรแกรม.....	41
6.1 สรุปผลการประเมินความพึงพอใจ.....	43
7. สรุปผลการดำเนิน โครงการ.....	48
7.1 สรุปการดำเนิน โครงการ.....	48
7.2 ข้อจำกัดและปัญหาของ โครงการ.....	48
7.3 ข้อเสนอแนะ.....	48
บรรณานุกรม.....	49
ภาคผนวก.....	50
ก พารามิเตอร์สำหรับการสร้างแบบจำลอง.....	51
ข พารามิเตอร์สำหรับกำหนดคุณสมบัติสีแสง.....	52
ค การติดตั้ง โปรแกรม.....	54
ง คู่มือการใช้งาน โปรแกรม.....	57
ประวัติผู้เขียน.....	59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 ชนิดของข้อมูลที่ใช้ใน OpenGL.....	9
5.1 ค่าตำแหน่งของพิกเซลรอบข้าง.....	37
6.1 ตัวอย่างสี่เหลี่ยมและคุณสมบัติทางออปติคัล.....	42
6.2 ผลการประเมินความพึงพอใจต่อความนุ่มของกระดาษ.....	45
6.3 ผลการประเมินความพึงพอใจต่อความฝืดของกระดาษ.....	46
6.4 ผลการประเมินความพึงพอใจต่อความหนืดของกระดาษ.....	47



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

ภาพที่	หน้า
2.1 อุปกรณ์ Personal Haptic Interface Mechanism (PHANToM).....	5
2.2 รูปแบบโครงสร้างการตั้งชื่อของ OpenGL.....	8
2.3 ตัวอย่างการสร้างรูปสามเหลี่ยมใน OpenGL.....	10
2.4 รูปทรงพื้นฐานที่ใช้ใน OpenGL.....	10
2.5 ขั้นตอนการทำ texture mapping.....	11
2.6 แสดงผลการทำ texture mapping.....	12
3.1 การทำ texture mapping โดย height map มีขนาดตั้งแต่(0, 0) ถึง (width, height).....	15
3.2 การคำนวณหาทิศทางของแรงตอบสนองที่เปลี่ยนไปจากค่า local gradient	16
3.3 แบบจำลองการเคลื่อน HCP และบนพื้นผิวกระดาษเสมือน.....	16
3.4 ลักษณะของพื้นผิวภายหลังจากที่ทำการแมปพื้นผิว.....	17
4.1 แ่งวัตถุถูกแรงดึงขนาดเท่ากัน ทิศทางตรงข้ามกระทำกระจายบนพื้นที่หน้าตัด A ในแนวตั้งฉาก	20
4.2 แ่งวัตถุถูกแรงดึงขนาดเท่ากัน ทิศทางตรงข้ามกระทำกระจายบนพื้นที่หน้าตัด A' ในแนวใดๆ กับแ่งวัตถุ.....	20
4.3 แรงขนาด F ดึงปลายทั้งสองด้านของแ่งวัตถุยาว L ทำให้ยืดออก ΔL	22
4.4 ตัวอย่างลักษณะเซลล์ของพื้นผิวกระดาษ.....	24
4.5 ไฮท์แมป 2 มิติ.....	25
4.6 ชั้นสีเทียบบนกระดาษ.....	25
4.7 ลักษณะสีที่เกิดจากการวาดครั้งแรก.....	26
4.8 ลักษณะสีที่เกิดจากการวาดซ้ำ ณ. จุดเดิม.....	26
4.9 พิกเซลรอบๆ ทั้ง 8 จุด.....	28
5.1 โครงสร้างของโปรแกรมทั้งหมด.....	30
5.2 การ Interpolation เพื่อให้ทิศทางมีความต่อเนื่อง.....	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
5.3 เปรียบเทียบวิธีการ Interpolation ทั้ง 3 วิธี	34
5.4 การจำลองปุ่มกดแบบกดติดปล่อยดับ	39
5.5 อธิบาย user-interface	40
6.1 โปรแกรมวาดภาพบน โมเดลสามมิติ	41
6.2 ภาพวาดโดยใช้โปรแกรมที่พัฒนา	43
6.3 ผลการประเมินความพึงพอใจต่อความนุ่มของกระดาษ	44
6.4 ผลการประเมินความพึงพอใจต่อความฝืดของกระดาษ	45
6.5 ผลการประเมินความพึงพอใจต่อความนุ่มของกระดาษ	47

บทที่ 1

บทนำ

1.1 โครงการพัฒนาระบบงาน

ปัจจุบันคอมพิวเตอร์และระบบสารสนเทศมีความก้าวหน้าทางด้านวิชาการ ไม่ว่าจะเป็นฮาร์ดแวร์ ซอฟต์แวร์ แต่ในทางตรงกันข้ามราคากลับถูกลง ซึ่งทำให้คอมพิวเตอร์เข้ามามากเป็นส่วนหนึ่งของชีวิตประจำวันของมนุษย์ในปัจจุบัน จึงมีการนำเทคโนโลยีทางด้านคอมพิวเตอร์กราฟิกมาผนวกเข้ากับอุปกรณ์ ตัวอย่าง ถุงมือ (Cyber Glove) แว่นตา (Cyber Eyes) และอุปกรณ์จำลองแรงสัมผัส (Force Feedback Device) เป็นต้น เพื่อสร้างและพัฒนางานด้านระบบเสมือนจริง (Virtual Reality) ทำให้มนุษย์สามารถติดต่อสื่อสารกับคอมพิวเตอร์ได้โดยตรงและเป็นธรรมชาติ ไม่ใช่เพียงแค่การสื่อสารผ่านการมองเห็น เสียง หรือการสัมผัสในรูปแบบของเมาส์ คีย์บอร์ด จอยสติค แต่สามารถเข้ามาควบคุมและสื่อสารผ่านอุปกรณ์โดยตรง ซึ่งอุปกรณ์จะมีการตอบสนองกลับมาที่ผู้ใช้ผ่านการสัมผัสหรือแรง

สำหรับอุตสาหกรรมหลายๆ ด้าน เริ่มมีการพัฒนาและตื่นตัวกับการนำวิทยาการทางด้านระบบเสมือนจริงเข้ามาใช้ช่วยในระบบงานด้านการออกแบบ การผลิต และการฝึกหัด เพื่อลดต้นทุนและเวลามากยิ่งขึ้น ตัวอย่างเช่น นำอุปกรณ์ที่ตอบสนองด้วยแรงต้านมาช่วยในระบบการออกแบบลดทลายบนภาชนะ ออกแบบรองเท้า หรือแม้กระทั่งระบบฝึกหัดการผ่าตัดผ่านเครือข่าย เป็นต้น

จากแนวความคิดข้างต้น ทำให้เกิดแนวความคิดในการพัฒนาโครงการระบบจำลองสำหรับการวาดภาพระยะด้วยสเตอริโอผ่านอุปกรณ์จำลองแรงสัมผัส ซึ่งในโครงการนี้จะกล่าวถึงส่วนของการสร้างลายเส้นสเตอริโอโดยใช้วิธีการจำลองทางกายภาพ ประกอบด้วยการสร้างแบบจำลองแท่งสเตอริโอ กระดาษ และลักษณะของสเตอริโอจะจำลองบนกระดาษ สำหรับรายละเอียดในการออกแบบและโครงสร้างของโปรแกรมจะได้กล่าวในบทต่อไป

การวาดภาพด้วยสเตอริโอโดยปกติแล้วจะประกอบไปด้วยกระดาษ และสเตอริโอ ซึ่งการฝึกฝนในการวาดภาพนั้นต้องใช้สเตอริโอและกระดาษเป็นจำนวนมาก ถ้าต้องการฝึกฝนมากต้องสิ้นเปลืองอุปกรณ์มากขึ้นไปด้วย ต่อมาได้มีการพัฒนาการวาดภาพโดยใช้เครื่องคอมพิวเตอร์ ผ่านอุปกรณ์ต่างๆ ซึ่งได้สร้างความสะดวกมากยิ่งขึ้น แต่ยังคงติดขัดในเรื่องของความสมจริงในการวาดภาพ นั่นคือการขาดแรงสัมผัสหรือความรู้สึกถึงการสัมผัสกับแท่งสเตอริโอหรือกระดาษ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น เมื่อผู้ยืมได้ให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จึงได้มีการพัฒนาการ โปรแกรมประยุกต์ในการจำลองการวาดภาพด้วยสีเทียนด้วยอุปกรณ์
จำลองแรงสัมผัส ซึ่งอุปกรณ์นี้จะสามารถให้ผู้ใช้สามารถที่จะรู้สึกถึงแรงสัมผัสต่อแท่งสีเทียน และ
การสัมผัสกับพื้นผิวของกระดาษ รวมถึงแรงต้านในการวาดภาพซึ่งเกิดจากสัมประสิทธิ์แรงเสียด
ทานของกระดาษ และของสีเทียน ในการพัฒนาโปรแกรมนั้นจะประกอบไปด้วย 2 ส่วนหลัก
กล่าวคือส่วนของกราฟิก ใช้ในการแสดงรูปภาพของกระดาษและลายเส้นของการวาดภาพภายหลัง
จากการวาดภาพ และอีกส่วนหนึ่งคือการแสดงผลแรงสัมผัส ซึ่งแรงสัมผัสที่จำลองขึ้นมาจะทำให้
ผู้ใช้รู้สึกถึงความขรุขระของพื้นผิวและแรงต้านในการวาดภาพ ซึ่งมีด้วยกัน 2 แรงคือ แรงต้านการ
เคลื่อนที่เป็นแรงที่จำลองการต้านการเคลื่อนที่ โดยจำลองความหนืดของกระดาษและความหนืด
ของสีเทียนภายหลังที่ได้วาดภาพแล้ว ส่วนอีกแรงหนึ่งเป็นแรงที่จำลองลักษณะความขรุขระ
ของพื้นผิว โดยจะทำการอ่านค่าความสว่างของรูปภาพของกระดาษจำลอง มาจำลองเป็นลักษณะ
ของความขรุขระของพื้นผิวของกระดาษ

ผลที่ได้หลังจากที่พัฒนาคือการที่มีระบบในการจำลองการวาดภาพโดยใช้สีเทียนโดยใช้
อุปกรณ์จำลองแรงสัมผัส ซึ่งสามารถที่จะฝึกฝนการวาดภาพ โดยมีความสมจริงมากกว่าการวาด
ภาพในสภาพแวดล้อมเสมือนโดยทั่วไป และเป็นการใช้เครื่องมือจำลองการสัมผัสให้แพร่หลาย
มากยิ่งขึ้น

1.2 แนวทางการพัฒนา

ในการพัฒนาโครงการนี้จำเป็นต้องมีขั้นตอนในการพัฒนา ซึ่งประกอบไปด้วย

1. ศึกษาทฤษฎีเกี่ยวกับการจำลองทางกายภาพ (Physical Simulation) เพื่อนำไปสร้างเป็น
ระบบการจำลองให้มีความเสมือนจริงวิเคราะห์การทำงานของโครงการ
2. ศึกษาอัลกอริทึมในการสร้างแบบจำลองการวาดเส้นสีเทียนพัฒนาโปรแกรม
3. วิเคราะห์และออกแบบ โครงสร้างข้อมูล
4. ออกแบบการทำงานของทั้งโครงการ
5. พัฒนาโปรแกรมที่ใช้ในโครงการ
6. ทดสอบการทำงานของทั้งโครงการ
7. บันทึกผลการดำเนินโครงการ
8. สรุปผลการดำเนินโครงการ
9. เขียนรายงานประกอบการทำโครงการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 วัตถุประสงค์

1. จำลองระบบการวาดภาพด้วยสีเทียนโดยใช้อุปกรณ์จำลองการสัมผัส เพื่อนำไปเป็นพื้นฐานสำหรับการพัฒนาระบบงานทางด้านการออกแบบที่ใหญ่ขึ้น
2. พัฒนาการจำลองแรงสัมผัสของพื้นผิวกระดาษและการวาดภาพให้มีความเสมือนจริง
3. มีเครื่องมือในการฝึกฝนการวาดภาพด้วยสีเทียน
4. สามารถนำเอาอุปกรณ์จำลองการสัมผัสมาใช้ได้หลากหลายมากยิ่งขึ้น
5. เป็นทางเลือกอีกทางหนึ่งในการฝึกฝนการวาดภาพ
6. เพื่อศึกษาและพัฒนากการทำงาน โดยใช้อุปกรณ์จำลองการสัมผัส

1.4 ขอบเขตของโครงการ

ในการพัฒนาโครงการมีขอบเขตในการทำงานดังนี้

1. พัฒนาโดยใช้ Microsoft Windows XP
2. จำเป็นต้องใช้อุปกรณ์ร่วม ในการทำงานจำพวก Force Feedback Device คือ PHANToM Premium 1.0 with Encoder Finger stylus

1.5 สภาพแวดล้อมในการดำเนินโครงการ

สภาพแวดล้อมหมายถึงรวมถึงอุปกรณ์ ระบบปฏิบัติการ ที่ใช้ในการพัฒนา และการดำเนินโครงการ ซึ่งประกอบไปด้วย

1. Microsoft Windows XP
2. Microsoft Visual C++ 6.0
3. อุปกรณ์ที่ให้ความรู้สึกได้โดยแรงสัมผัส PHANToM Premium 1.0 with Encoder Finger stylus
4. Software library 3DTouch สำหรับติดต่อกับอุปกรณ์ PHANToM
5. OpenGL สำหรับแสดงภาพกราฟิกสามมิติ

1.6 ประโยชน์ที่คาดว่าจะได้รับ

1. มีโปรแกรมการวาดภาพด้วยสีเทียน โดยใช้อุปกรณ์จำลองการสัมผัสซึ่งสามารถจำลองแรงสัมผัสของพื้นผิวกระดาษและการวาดภาพให้มีความเสมือนจริง
2. มีอุปกรณ์หรือระบบเพื่อนำไปประยุกต์ใช้กับงานทางด้านการผลิต การออกแบบ และการฝึกหัด ด้วยระบบเสมือนจริงผ่านอุปกรณ์ที่ตอบสนองด้วยแรงต้าน
3. มีความรู้ความเข้าใจในอุปกรณ์จำลองการสัมผัสเพิ่มมากขึ้น ซึ่งสามารถนำไปประยุกต์ใช้ในงานอื่นๆ ได้
4. เป็นแนวทางในการพัฒนา โปรแกรมประยุกต์อื่นๆ เกี่ยวกับอุปกรณ์จำลองการสัมผัส
5. ทำให้เครื่องมือจำลองการสัมผัสใช้อย่างแพร่หลายมากยิ่งขึ้น



บทที่ 2

เทคโนโลยีเพื่อการติดต่อกับผู้ใช้

ส่วนติดต่อกับผู้ใช้ (User Interface) ที่ใช้ในการศึกษานี้ประกอบไปด้วย GUI (Graphic User Interface) และ HUI (Haptic User Interface) โดยส่วนแรกนั้นสามารถสร้างขึ้นมาได้โดยใช้ OpenGL ซึ่งเป็น Software Development Kit สำหรับการสร้างภาพกราฟิกทั่วไป ส่วนหลังนั้นเกิดจากการใช้อุปกรณ์ตอบสนองต่อแรงทำงานร่วมกับ 3DTouch ซึ่งเป็น Software Development Kit สำหรับใช้ติดต่อกับอุปกรณ์ตอบสนองต่อแรงโดยเฉพาะ

ทฤษฎีพื้นฐานที่จำเป็นต้องใช้ในการดำเนินโครงการนั้น จะประกอบไปด้วย คุณลักษณะของ อุปกรณ์ตอบสนองต่อแรง โลบารี 3DTouch และ OpenGL

2.1 อุปกรณ์ตอบสนองต่อแรง

อุปกรณ์ที่ใช้ในการดำเนินโครงการคือ Personal Haptic Interface Mechanism (PHANToM) ซึ่งเป็นอุปกรณ์ที่ต้องติดตั้งเป็น โຕะ ออกแบบมาเพื่อจำลองการสร้างแรงด้านให้กับผู้ใช้ ดังแสดงในรูปที่ 2.1



รูปที่ 2.1 อุปกรณ์ Personal Haptic Interface Mechanism (PHANToM)

ส่วนที่สำคัญที่สุดก็คือส่วนเชื่อมต่อกับผู้ใช้ ซึ่งส่วนปลายจะมีปลอกสวมนิ้ว ซึ่งสามารถหมุนได้รอบทิศทาง ส่วนแขนก็สามารถขยับได้ อุปกรณ์ชนิดนี้มีค่าของการอิสระเป็น 6 การทำงานจะจำลองการสัมผัสกับวัตถุที่ปลอกหุ้มปลายนิ้ว ส่วนการส่งแรงให้กับสถานะแวดล้อมเสมือนจะเกิดที่ปลอกสวมนิ้วเช่นกัน แต่จะไม่มีการหมุน ตัวกระตุ้นจะติดตั้งให้สมดุล นั่นคือว่า อุปกรณ์ชนิดนี้จะมีความสมดุลอยู่ในตัวอยู่แล้ว จะไม่มีการชดเชยแรงโน้มถ่วงในขณะที่มอเตอร์กำลังหมุน อุปกรณ์จะมีขนาด $8 \times 17 \times 25$ เซนติเมตร และสร้างกำลังด้วยมอเตอร์กระแสตรงจำนวน 3 ตัวและตรวจถอดสัญญาณแบบแสง 1 ตัวซึ่งติดตั้งอยู่บนคานของตัวกระตุ้น ตัวส่งผ่านสัญญาณจะใช้สายเคเบิลพร้อมด้วยลูกรอก และทำให้กลไกการทำงานง่ายขึ้น โดยการยึดมอเตอร์ 2 ตัว กับสายเคเบิลเส้นเดียว PHANTOM จะมีความเฉื่อยประมาณ 0.1 กิโลกรัม ความเสียดทานสถิตอยู่ที่ 0.1 นิวตัน ส่งแรงด้านได้สูงสุดอยู่ที่ 10 นิวตัน การควบคุมนั้นจะควบคุมผ่านคอมพิวเตอร์ ในการส่งสัญญาณนั้น เมื่อมีแรงซึ่งกระทำโดยผู้ใช้ ระบบจะตอบสนองต่อเหตุการณ์ที่ผู้ใช้กระทำนั้น ๆ โดยการสร้างแรงต้านกลับมายังผู้ใช้

2.2 3DTouch HDAPI

Haptic Device Application Programming Interface (HDAPI) เป็นส่วนที่ใช้ติดต่อสื่อสารกับอุปกรณ์ตอบสนองต่อแรง ซึ่งหมายความว่าเราสามารถเขียนโปรแกรมควบคุมอุปกรณ์ตอบสนองต่อแรงนี้ได้โดยใช้ HDAPI ข้อดีของการใช้ HDAPI แทนที่ GHOST SDK ก็คือสามารถที่จะควบคุมในระดับล่างได้ซึ่งก็หมายความว่าสามารถที่จะควบคุมอุปกรณ์เซอร์โวมอเตอร์ได้ การใช้ HDAPI ช่วยให้ส่งแรงไปยังอุปกรณ์ได้ง่ายและมีความยืดหยุ่น อีกทั้งยังสามารถจัดการกับข้อผิดพลาดที่อาจเกิดขึ้นในระหว่างการส่งแรงช่วยให้การทำงานกับอุปกรณ์ตอบสนองต่อแรงเป็นไปอย่างปลอดภัย

HDAPI ประกอบไปด้วย 2 โมดูลคือ ส่วนของอุปกรณ์ซึ่งจะทำหน้าที่จัดการในเรื่องของการติดต่อสื่อสารกับอุปกรณ์ตอบสนองต่อแรง และในส่วนตัวจัดการการทำงานซึ่งจะดูแลในส่วนของการทำงาน เช่นการส่งแรง เป็นต้น

รูปแบบการใช้งาน HDAPI ทั่วๆ ไปนั้นเริ่มจาก

1. กำหนดค่าเริ่มต้นให้กับตัวแปรต่างๆ
2. สร้างฟังก์ชันที่เป็น Callback function สำหรับเป็นตัวจัดการเกี่ยวกับอุปกรณ์ ช่วยให้สามารถที่จะรู้ตำแหน่ง ความเร็ว หรือค่าอื่นๆ ของอุปกรณ์ได้ นอกจากนั้นยังสามารถส่งแรงไปยังตัวอุปกรณ์ได้
3. ส่งคำสั่งไปยังอุปกรณ์ให้เริ่มทำงาน สามารถเริ่มส่งแรงไปยังอุปกรณ์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เริ่มการทำงานของ Callback function
5. จบการทำงานของ Callback function

2.3 Open Graphic Library (OpenGL)

OpenGL เป็นซอฟต์แวร์ที่ใช้ในการติดต่อกับอุปกรณ์ทางการแสดงผลรูปภาพทั้งในส่วนจอภาพและการ์ดแสดงผล จะประกอบไปด้วยโพธิ์เซอร์และฟังก์ชัน เพื่อใช้ในการโปรแกรมเพื่อสร้างกราฟิกที่มีคุณภาพสูง โดยเฉพาะกับการสร้างกราฟิกแบบ 3 มิติ ซึ่ง OpenGL นั้นเริ่มแนะนำในปี พ.ศ.2534 โดยบริษัท Silicon Graphics Incorporated (SGI)

2.3.1 คุณสมบัติเด่นของ OpenGL

1. เป็นมาตรฐานในด้านกราฟิก กล่าวคือ รายละเอียดของ OpenGL ถูกกำหนดและจัดการโดยสมาคมอิสระ Architecture Review Board (ARB)
2. มีเสถียรภาพ เนื่องจากมีการพัฒนาขึ้นมาอย่างยาวนาน รวมถึงได้มีการทดสอบและเป็นที่ยอมรับของนักพัฒนากราฟิกอย่างแพร่หลาย
3. มีความน่าเชื่อถือและไม่ขึ้นกับแพลตฟอร์ม OpenGL นั้นไม่คำนึงถึงระบบปฏิบัติการ แอปพลิเคชันที่ใช้ OpenGL สามารถทำงานได้ตั้งแต่คอมพิวเตอร์เล็ก เวิร์คสเตชัน จนถึงซูเปอร์คอมพิวเตอร์
4. ความง่ายในการใช้งาน OpenGL มีคำสั่งต่างๆมากมายซึ่งได้ถูกจัดเป็น โครงสร้างอย่างดี ทำให้มีความง่ายในการใช้งาน

2.3.2 สถาปัตยกรรมของ OpenGL

OpenGL ถูกออกแบบในลักษณะของสถานะ (state) ที่คอยควบคุมการแสดงผลภาพ กล่าวคือ . เมื่อกำหนดรูปทรงที่จะวาดแล้วผลลัพธ์ที่ได้นั้นขึ้นอยู่กับที่กำหนดสถานะก่อนหน้านั้นอย่างไร เช่นการกำหนดสี รูปแบบ รูปทรง แสงเงา เมื่อรวมกันเข้าจะเกิดการแสดงผลที่สวยงามขึ้น

2.3.3 การตั้งชื่อของฟังก์ชันของ OpenGL

OpenGL นั้นมีแบบแผนในการตั้งชื่อเพื่อให้ผู้พัฒนาสามารถทราบได้ว่ามาจากฟังก์ชันไหน โลบารี่อะไร รวมถึงจำนวนอาร์กิวเมนต์เป็นชนิดอะไร ซึ่งรูปแบบ โครงสร้างมีดังนี้

<Library Type><Root command><Number of arguments><Data Type>

Library Type	บอกว่าใช้ไลบรารีไหน
Root Command	เป็นรากของคำสั่ง
Number of arguments	จำนวนของพารามิเตอร์
Data Type	ชนิดของตัวแปร

รูปที่ 2.2 รูปแบบโครงสร้างการตั้งชื่อของ OpenGL

จากรูปที่ 2.5 ตัวอย่างในการตั้งชื่อ เช่น glVertex3f รากของคำสั่งคือ Vertex ส่วนขึ้นต้นด้วย gl ก็มาจากไลบรารีของ gl ส่วน 3f หมายความว่ามีการ์คิวเมนต์ 3 ตัวเป็นชนิด float

2.3.4 ชนิดข้อมูลของ OpenGL

เพื่อที่จะสามารถนำเอาโปรแกรมที่พัฒนาด้วย OpenGL ไปใช้ในระบบปฏิบัติการใดๆ ก็ได้ นั้น จึงต้องใช้ชนิดของข้อมูลซึ่งกำหนดขึ้นเอง โดยชนิดข้อมูลเหล่านี้จะคล้ายกับชนิดข้อมูลที่มีอยู่ในภาษา C ซึ่งในแต่ละ Compiler จะกำหนดขนาดของหน่วยความจำและการใช้งานที่แตกต่างกัน ดังนั้นการใช้งานชนิดข้อมูลที่กำหนดโดย OpenGL นั้นจะเป็นการป้องกันการเปลี่ยนแปลงชนิดข้อมูลได้ด้วย ชนิดข้อมูลที่ OpenGL กำหนดขึ้นมาใหม่นั้นจะขึ้นต้นด้วย GL ตามด้วยชนิดของข้อมูล ดังมีรายละเอียดดังนี้

ตารางที่ 2.1 ชนิดของข้อมูลที่ใช้ใน OpenGL

Suffix	Data Type	Typical C type	OpenGL Type Name
B	8-bit integer	Signed char	GLbyte
S	16-bit integer	Short	GLshort
I	32-bit integer	Int or long	GLint ,GLsizei
F	32-bit floating point	Float	GLfloat , GLclampf
D	64-bit floating point	Double	GLdouble , GLclampd
Ub	8-bit unsigned number	Unsigned char	GLubyte , GLboolean
Us	16-bit unsigned number	Unsigned short	GLushort
Ui	32-bit unsigned number	Unsigned int or unsigned long	GLuint , GLenum , GLbitfield

2.3.5 รูปทรงพื้นฐานของ OpenGL

รูปทรงพื้นฐานของ OpenGL นั้น สร้างได้จากการกำหนดจุดยอด และลำดับการเรียงกันของจุดยอด ซึ่งกำหนดโดยฟังก์ชัน `glVertex3f ()` การกำหนดเซตของจุดยอด เพื่อกำหนดเป็นรูปทรงต่างๆ นั้น โดยจะเริ่มจากการกำหนดประเภทของรูปทรงที่ต้องการ ซึ่งเป็นส่วนหนึ่งของพารามิเตอร์ `glBegin ()` ตามด้วยชุดของจุดยอด และสุดท้ายคือ `glEnd ()`

ตัวอย่างการสร้างรูปทรงนั้น จะแสดงการสร้างรูปทรงสามเหลี่ยมซึ่งจะมีจุดยอดทั้งหมด 3 จุด ซึ่งต้องใช้พารามิเตอร์ 3 ตัวเป็นประเภท float รวมทั้งสิ้น 3 เพื่อสร้างรูปสามเหลี่ยม แสดงได้ดังนี้

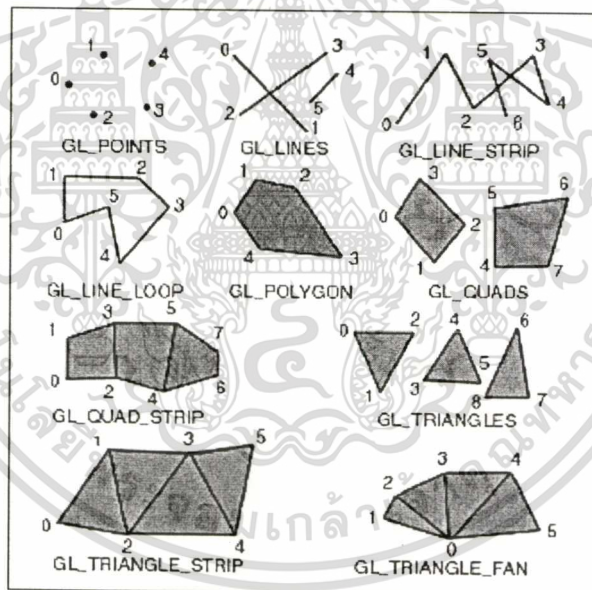
```

Begin( GL_TRIANGLES);
Vertex3f(1.0,1.0,1.0);
Vertex3f(1.0,-1.0,1.0);
Vertex3f(1.0,1.0,-1.0);
End();

```

รูปที่ 2.3 ตัวอย่างการสร้างรูปสามเหลี่ยมใน OpenGL

รูปทรงพื้นฐานต่างๆ ที่ใช้ใน OpenGL แสดงดังในรูปที่ 2.7



รูปที่ 2.4 รูปทรงพื้นฐานที่ใช้ใน OpenGL

2.3.6 การกำหนดสถานะ

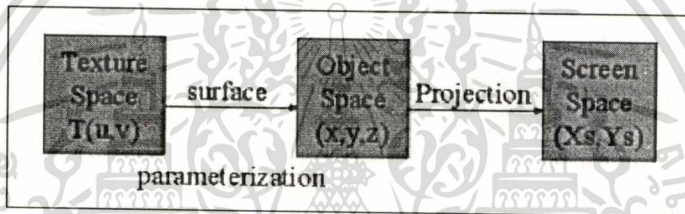
การกำหนดสถานะนั้นใช้เพื่อในการเรียกใช้งานส่วนเพิ่มเติมต่างๆ โดยจะมีการยอมให้แสดงผลหรือไม่ยอมให้แสดงผล ฟังก์ชันที่ใช้ในการยอมให้มีการแสดงผลคือ `glEnable ()` ส่วนฟังก์ชันที่ไม่ยอมให้มีการแสดงผลนั้นคือ `glDisable()` ตัวอย่างเช่น `glEnable(GL_LIGHTING)` หมายถึง ยอมให้มีการแสดงผลของการใช้แสง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

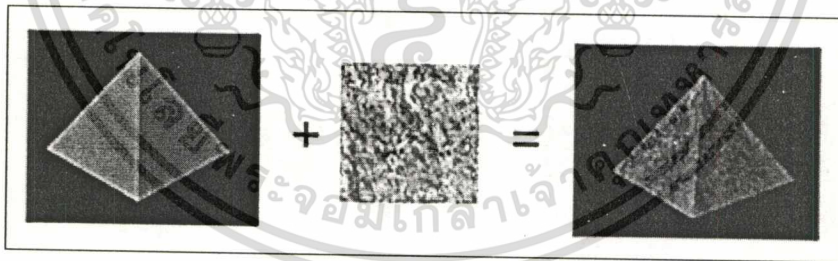
ส่วนการกำหนดสถานะนั้นจะเรียกใช้ตามฟังก์ชันนั้น เช่นการใส่สีให้กับวัตถุ จะใช้ `glColor3f(...)`

2.3.7 การเพิ่มรายละเอียดให้กับโมเดล

ในการที่จะเพิ่มความสมจริงให้กับโมเดลนั้น โดยทั่วไปจะใช้เทคนิคซึ่งเรียกว่า texture mapping ซึ่งจะทำให้โมเดลนั้นมีความสวยงามและความซับซ้อน โดยที่ไม่ได้เพิ่มความซับซ้อนของโพลีกอนหรือวัตถุพื้นฐานที่ประกอบกันเป็นโมเดลนั้นๆ เลย การทำ texture mapping นั้นจะช่วยให้เราสามารถวาดภาพบนโมเดลได้ และคุณสมบัติของภาพที่วาดนั้นไม่สามารถที่จะทำให้เกิดขึ้นได้ โดยใช้หลักการทางเรขาคณิต คุณสมบัติของพื้นผิว และการกำหนดคุณสมบัติของแสง



รูปที่ 2.5 ขั้นตอนการทำ texture mapping



รูปที่ 2.6 แสดงผลการทำ texture mapping

ในส่วนของการวาดภาพนั้นจะใช้การทำ texture mapping โดยที่เมื่อมีการเรียกหรือต้องการให้วาดภาพ ฟังก์ชันที่ทำหน้าที่ในส่วนนี้จะรับเอาค่าตำแหน่งของโพรบซึ่งก็คือตำแหน่งที่จะวาดจะนำไปสู่การคำนวณว่าจะไปอ็อปเทคส่วนไหนของภาพที่จะนำมาทำ texture mapping ส่วนค่าสีที่จะนำไปอ็อปเทคในรูปภาพนั้น ในโปรแกรมจะใช้ค่าสีของ pixel ที่ใช้เป็น sub texture ฟังก์ชันใน OpenGL ที่ทำหน้าที่สำหรับการอ็อปเทค sub texture คือ

```
glTexSubImage2D(GL_TEXTURE_2D, level, xoffset,
                yoffset, width, height, format, type, pixels);
```

โดยที่ pixels นั้นได้มาจากการคำนวณค่าสีของกระดาด ณ ตำแหน่งที่ถูกวาดทับด้วยสีเขียน ฟังก์ชันนี้จะไม่ต้อง redefine texture ใหญ่ทั้งหมดอีกครั้ง ดังนั้นจึงมีประสิทธิภาพมากขึ้น แต่ก็ไม่ควรที่จะนำฟังก์ชันนี้ไปใช้แทนที่รูปภาพทั้งหมด



บทที่ 3

การจำลองแรงต้านในการวาดภาพด้วยสีเทียน

2.4 แรงต้านที่เกิดจากการวาดภาพ

แรงต้านที่เกิดขึ้นหรือที่จะต้องจำลองขึ้นมาในขณะที่วาดภาพนั้นประกอบไปด้วย 2 แรง คือแรงต้านการเคลื่อนที่และแรงซึ่งจำลองลักษณะของกระดาษ แรงต้านการเคลื่อนที่นั้นยังแบ่งเป็น 2 แรงคือ แรงเสียดทานระหว่างสีเทียนกับกระดาษและแรงหนืดที่เกิดจากสีเทียน ซึ่งจะได้อธิบายรายละเอียดต่อไป

3.2 แรงต้านการเคลื่อนที่

แรงต้านการเคลื่อนที่ที่สอดคล้องกับจำนวนสีเทียนที่ติดลงบนกระดาษ เมื่อเริ่มแรกของการวาดภาพกระดาษซึ่งไม่มีการติดลงบนกระดาษเป็นการจำลองสัมประสิทธิ์แรงเสียดทานของกระดาษ และเมื่อทำการวาดภาพซ้ำในจุดเดิมขีดถือเอาว่า ได้มีสีเทียนติดอยู่ที่พื้นผิวของกระดาษแล้ว ให้ทำการจำลองแรงสัมประสิทธิ์แรงเสียดทานของสีเทียนเข้ามาแทนที่

$$\vec{F}_F = \mu \vec{F}_N \quad (3.1)$$

เมื่อ

\vec{F}_F	คือ	แรงต้านที่แสดงให้กับผู้ใช้
\vec{F}_N	คือ	ทิศทางปกติของพื้นผิว
μ	คือ	สัมประสิทธิ์ของแรงต้านบนกระดาษ

ด้วยการใช้ค่าความสูงของพื้นผิว จะใช้ค่าความสูงในการกำหนดความสูงของกระดาษ ในการต้านการเคลื่อนที่ของแท่งสีเทียน และการคำนวณแรงต้านการเคลื่อนที่สำหรับกระดาษ ค่าของ μ ขึ้นอยู่กับเงื่อนไขว่าแท่งสีกระทำกับกระดาษที่ว่างเปล่าหรือกับกระดาษที่มีสีอยู่แล้ว

ในการคำนวณแรงเสียดทานนั้น จะต้องคิดทั้งสองค่าคือ แรงเสียดทานสถิตและแรงเสียดทานจลน์ โดยการเคลื่อนที่ของ HCP จะมีสองสถานะได้แก่ สถานะที่หยุดนิ่งซึ่งจะใช้ค่าแรงเสียดทานสถิต เมื่อไหร่ที่แรงในแนวเดียวกับพื้นผิว (tangent force) มีค่ามากกว่าแรงเสียดทานสถิต HCP จะเปลี่ยนสถานะเป็นเคลื่อนที่ซึ่งจะต้องนำค่าแรงเสียดทานจลน์มาคิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

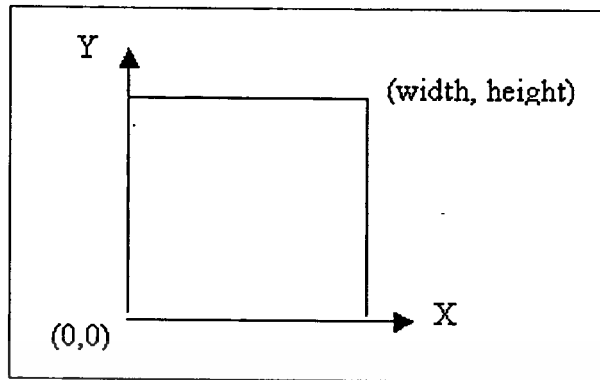
การคำนวณค่าแรงในแนวเดียวกับพื้นผิว (tangent force) คำนวณคล้ายกับแรงในแนวตั้งฉากกับพื้นผิว (normal force) โดยในโปรแกรมนี้ได้ใช้ระยะห่างระหว่างจุดแรกที่ HCP แตะพื้นกับจุดที่ลากออกไปจากจุดเดิม ซึ่งก็คือค่าตำแหน่งของสถานะก่อนหน้าและค่าตำแหน่งของสถานะปัจจุบัน โดยทั้งสองค่านี้ได้จากการสอบถามจาก callback function จากนั้นจึงนำระยะห่างนี้ไปคำนวณแรง tangent force เช่นเดียวกับแรง normal force ระยะห่างก็คือ penetration distance หรือระยะที่ HCP ทะลุเข้าไปในพื้นผิวนั่นเอง

การคำนวณค่าแรงเสียดทานนั้นแบ่งออกเป็น 2 สถานะคือ แรงเสียดทานสถิตและแรงเสียดทานจลน์ อัลกอริทึมที่ใช้คำนวณนั้นจะพัฒนาตามโมเดลของ Karnopp Friction ซึ่งมีข้อดีคือสามารถจำลองแรงเสียดทานได้อย่างถูกต้องและมีการคำนวณน้อย แต่ก็มีข้อเสียคือ ไม่สามารถจำลองแรงเสียดทานสถิตได้สมบูรณ์ เช่น พื้นผิวที่มีความยืดหยุ่นและมีวัตถุเคลื่อนที่อยู่บนพื้นผิวเมื่ออยู่ในสถานะแรงเสียดทานสถิต พื้นผิวควรจะยอมให้วัตถุนั้นเคลื่อนที่ได้ แต่อัลกอริทึมไม่สามารถจำลองในส่วนนี้ได้ นอกจากนี้ยังมีข้อเสียอีกคือทำให้เกิดการ oscillation ขณะที่เปลี่ยนสถานะจากแรงเสียดทานสถิตเป็นแรงเสียดทานจลน์หรือในทางกลับกันด้วย

นอกจากแรงเสียดทานสถิตกับแรงเสียดทานจลน์แล้ว Karnopp Friction ยังสามารถจำลองแรงหน่วงที่ต้านการเคลื่อนที่ damping force ซึ่งมีทิศทางตรงข้ามกับความเร็วของการเคลื่อนที่ ได้อีกด้วย

3.3 แรงที่เกิดจากการจำลองลักษณะของกระดาด

ในการออกแบบกระดาดหรือพื้นผิวที่จะใช้สำหรับวาดนั้น จะทำให้กระดาดนั้นมีความขรุขระซึ่งสามารถทำได้โดยการสร้างลักษณะของพื้นผิวให้มีความสูง โดยการทำให้ลักษณะของพื้นผิวปกติเปลี่ยนแปลงซึ่งสามารถทำให้พื้นผิวเรียบกลายเป็นพื้นผิวที่ขรุขระได้ โดยใช้หลักการคือทำให้ทิศทางของแรงที่ตอบสนองนั้นเปลี่ยนแปลงไปจากเดิมซึ่งเรียบและมีทิศทางเดียวกัน วิธีนี้ได้ดัดแปลงมาจากวิธีที่ใช้สำหรับการทำ bump-mapping ทั่วๆ ไป ขั้นตอนการทำงานจะเริ่มจาก การคำนวณค่า local gradient ของภาพที่นำมาใช้สำหรับการทำ texture mapping เมื่อมีการเคลื่อนที่ HCP และแตะกับพื้นผิว จะต้องคำนวณทิศทางของแรงที่จะตอบสนองโดยประมาณได้จากจุดรอบๆ ทั้งสี่จุด



รูปที่ 3.1 การทำ texture mapping โดย height map มีขนาดตั้งแต่ (0, 0) ถึง (width, height)

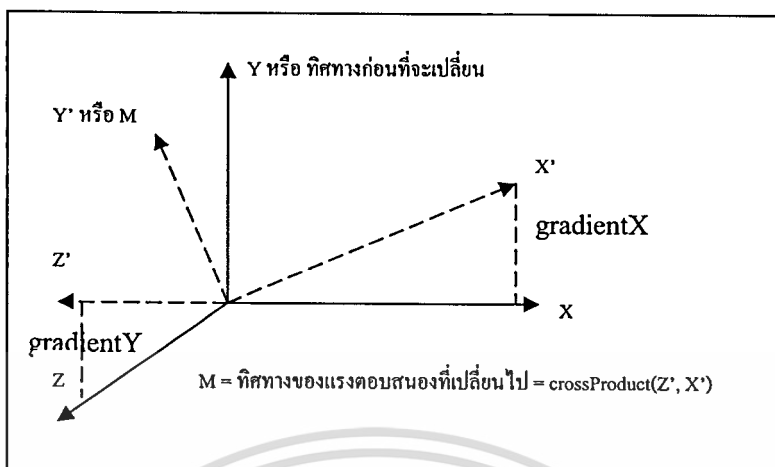
จากโปรแกรมที่ได้พัฒนานั้น ทิศทางของแรงก่อนที่จะทำ texture mapping นั่นก็คือทิศทางตามแกน y (0, 1, 0) ซึ่งขั้นตอนต่อไปจะคำนวณหาทิศทางที่เปลี่ยนไปได้โดยอาศัยค่า local gradient จาก height map หรือภาพโดยกำหนดให้

$$\begin{aligned}\alpha &= Z' = (0, \text{gradient}X, 1) \\ \beta &= X' = (1, \text{gradient}Y, 0)\end{aligned}\quad (3.2)$$

α และ β เป็นเวกเตอร์ที่กำหนดขึ้นมา ดังนั้นทิศทางของแรงที่เปลี่ยนไปจะหาได้จากสมการดังต่อไปนี้

$$\vec{M} = Y' = \text{crossproduct}(\alpha, \beta) \quad (3.3)$$

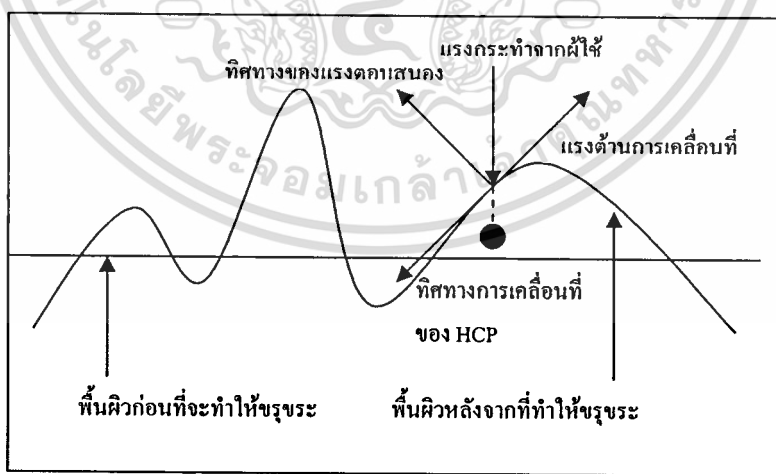
โดยที่ \vec{M} จะต้องถูก normalize ก่อนนำไปใช้คำนวณหาแรง



รูปที่ 3.2 การคำนวณหาทิศทางของแรงตอบสนองที่เปลี่ยนไปจากค่า local gradient

จากสมการที่ 3.2 เมื่อค่า local gradient เป็น 0 ซึ่งหมายความว่าจุดใน height map นั้นมีค่าของสีที่เท่ากันจะทำให้ α และ β เป็น (0, 0, 1) และ (1, 0, 0) ตามลำดับ ผลของ cross product จะได้เป็น (0, 1, 0) นั่นคือจะได้ทิศทางเป็น Y ตามรูปที่ 3.2

แต่ถ้าค่าของ local gradient ไม่เป็น 0 ทิศทางของแรงก็จะเปลี่ยนไปโดยจะเบี่ยงเบนออกจากแกน Y ปกติ

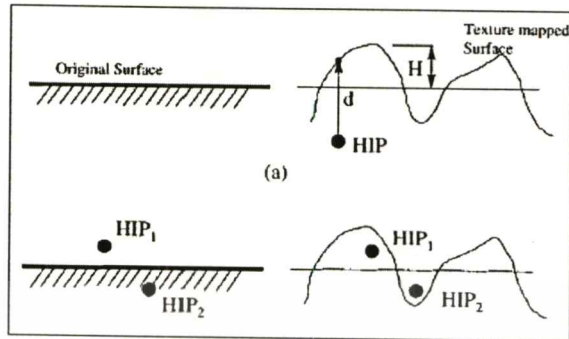


รูปที่ 3.3 แบบจำลองการเคลื่อน HCP และบนพื้นผิวระคายเคือง

เมื่อได้ลักษณะของพื้นผิวแล้วนั้นต่อไปจะทำการคำนวณหาแรงที่จะทำการตอบสนอง

ให้กับผู้ใช้งานดังสมการที่ 3.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 ลักษณะของพื้นผิวภายหลังจากที่ทำการแมปพื้นผิว

เมื่อ

$$\vec{F} = d \vec{M}$$

(3.4)

\vec{F} คือ แรงที่จะตอบสนองให้กับผู้ใช้
 \vec{M} คือ พื้นผิวภายหลังจากการที่ทำการ texture mapping
 d คือ ความลึกของการทะลุเข้าไปในพื้นผิว

บทที่ 4

การสร้างลายเส้นสี่เหลี่ยม

โครงการนี้เป็นระบบงานสำหรับการสร้างลายเส้นสี่เหลี่ยมโดยใช้วิธีการจำลองทางกายภาพ เพื่อสนับสนุนการทำงานในส่วนของการเรนเดอร์ภาพลายเส้นสี่เหลี่ยมบนอุปกรณ์ตอบสนองด้วยแรงต้าน ในการพัฒนาโครงการจำเป็นต้องเข้าใจทฤษฎีและหลักการพื้นฐานทฤษฎีทางด้านฟิสิกส์ เพื่อนำไปใช้สำหรับการจำลองทางกายภาพให้การทำงานมีความเหมือนจริงมากที่สุด ซึ่งจะกล่าวต่อไป

4.1 แบบจำลองทางกายภาพ

แบบจำลองทางกายภาพ (Physical Simulation) คือการนำเอาหลักการหรือกฎทางฟิสิกส์มาประยุกต์ใช้สร้างเป็นแบบจำลอง เพื่อให้การสร้างโมเดลเสมือนจริง ซึ่งในโครงการนี้มีการนำทฤษฎีทางด้านกลศาสตร์มาประยุกต์ใช้ ได้แก่ แรงเสียดทาน และแรงยืดหยุ่น จะได้กล่าวต่อไป

แรง (Force) เป็นการกระทำของวัตถุหนึ่งต่ออีวัตถุหนึ่ง กำหนดขึ้นทั้งขนาดและทิศทางตามแนวแรงที่กระทำ ซึ่งก็คือแรงเป็นปริมาณเวกเตอร์ ขนาดแรงแทนด้วยหน่วยนิวตัน (Newton)

แรง 1 นิวตัน หมายถึง แรงที่ดันหรือผลักมวล 1 กิโลกรัมไปในทิศทางของแรงนั้นให้เคลื่อนที่ไปโดยมีความเร่ง 1 เมตรต่อวินาที² (m/sec^2)

จากนิยามขนาดของแรงที่ได้กล่าวไปข้างต้น จะต้องเป็นไปตามการเคลื่อนที่ของนิวตัน

4.2 แรงเสียดทาน

แรงเสียดทาน (Friction Force) เป็นแรงที่เกิดขึ้นระหว่างผิวสัมผัสของวัตถุสองชิ้นที่กำลังเคลื่อนที่ผ่านกัน เป็นแรงที่มีทิศอยู่ในแนวขนานกับผิวสัมผัสและตรงข้ามกับการเคลื่อนที่ สำหรับแรงเสียดทานที่เกิดจากการไถล (Sliding Friction) ประเภทนี้แบ่งออกได้เป็น 2 แบบ ดังนี้

4.2.1 แรงเสียดทานสถิต

แรงเสียดทานสถิต (Static Friction, F_s) เป็นแรงเสียดทานที่เกิดขึ้นก่อนที่วัตถุจะมีการเคลื่อนที่ออกจากจุดเริ่มต้น โดยจะเกิดขึ้นเมื่อมีแรงภายนอกมากระทำต่อวัตถุตั้งแต่เริ่มหยุดนิ่ง แรงภายนอกนี้จะมีขนาดเพิ่มขึ้นเรื่อยๆ ซึ่งแรงเสียดทานจะมีขนาดเพิ่มขึ้นด้วย จนกระทั่งวัตถุเริ่มขยับเคลื่อนที่ออก ขณะนี้เองที่แรงภายนอกและแรงเสียดทานมีค่าสูงสุด โดยค่าสูงสุดของแรงเสียดทานสถิตสามารถคำนวณได้จากสมการที่ 4.1 ดังนี้

$$F_s = \mu_s F_c \quad (4.1)$$

โดยที่

μ_s แทนสัมประสิทธิ์ความเสียดทานสถิต (Coefficient of Static Friction)

F_c แทนแรงกดในแนวตั้งฉากของวัตถุบนพื้นผิวที่พิจารณา

4.2.2 แรงเสียดทานจลน์

แรงเสียดทานจลน์ (Kinetic Friction) เป็นแรงที่เกิดขึ้นขณะวัตถุกำลังเคลื่อนที่ แรงนี้จะมีค่าน้อยกว่าค่าของแรงเสียดทานสถิต สามารถคำนวณได้จากสมการที่ 4.2 ดังนี้

$$F_k = \mu_k F_c \quad (4.2)$$

โดยที่

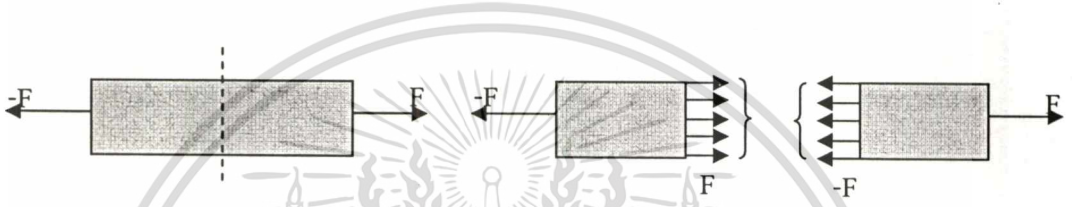
μ_k แทนสัมประสิทธิ์ความเสียดทานจลน์ (Coefficient of Kinetic Friction)

F_c แทนแรงกดในแนวตั้งฉากของวัตถุบนพื้นผิวที่พิจารณา

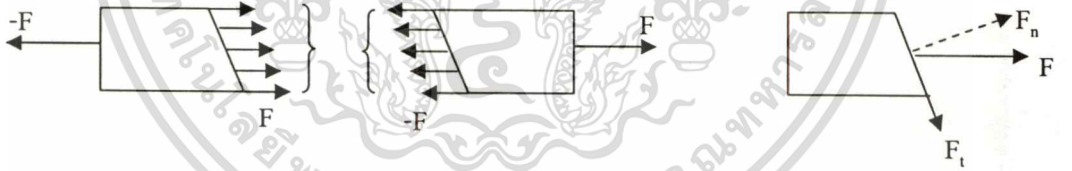
เมื่อค่าสัมประสิทธิ์ความเสียดทานเป็นค่าคงที่ ขึ้นอยู่กับชนิดของผิวสัมผัส ซึ่งค่าของ μ_s มีค่ามากกว่า μ_k

4.3 แรงยืดหยุ่น

แรงยืดหยุ่น (Elastic Force) เป็นแรงภายในตัววัตถุ มักจะเกิดขึ้นเมื่อมีแรงภายนอกมากระทำต่อวัตถุนั้นก่อน วัตถุที่มีความยืดหยุ่นสมบูรณ์จะกลับคืนสู่สภาพเดิมได้ภายหลังเลิกแรงกระทำ ซึ่งวัตถุนั้นอาจจะมีรูปร่างหรือขนาดเปลี่ยนไปจากเดิมมากน้อยขึ้นอยู่กับขนาดของแรงที่กระทำและสมบัติของวัตถุ ตัวอย่างเช่น แรงยืดหยุ่นของสปริง และแรงยืดหยุ่นในแท่งของแข็ง เป็นต้น ในโครงการนี้จะศึกษาในเรื่องของแรงยืดหยุ่นที่เกิดขึ้นในแท่ง



รูปที่ 4.1 แท่งวัตถุถูกแรงดึงขนาดเท่ากัน ทิศทางตรงข้ามกระทำกระจายบนพื้นที่หน้าตัด A ในแนวตั้งฉาก



รูปที่ 4.2 แท่งวัตถุถูกแรงดึงขนาดเท่ากัน ทิศทางตรงข้ามกระทำกระจายบนพื้นที่หน้าตัด A' ในแนวใดๆ กับแท่งวัตถุ

4.3.1 ความเค้น

ความเค้น (Stress) คือ อัตราส่วนของขนาดของแรงกระทำต่อพื้นที่หน้าตัด แบ่งลักษณะของความเค้นออกเป็น 2 ประเภท ได้แก่ ความเค้นตั้งฉาก หรือความเค้นดึง และความเค้นเฉือนหรือความเค้นสัมผัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความเค้นดึง (Tensile Stress) หรือความเค้นตั้งฉาก (Normal Stress) พิจารณาส່วนของแรงกระทำที่รอยตัดที่กระจายไปทั่วพื้นที่หน้าตัด A ในแนวตั้งฉาก แสดงดังรูปที่ 4.1 สามารถแทนด้วยสมการที่ 4.3 ดังนี้

$$S_n = \frac{F}{A} \quad (4.3)$$

โดยที่

- S_n แทนความเค้น
 F แทนแรงขนานกับพื้นที่หน้าตัด A
 A แทนพื้นที่หน้าตัด

ความเค้นเฉือน (Shearing Stress) หรือความเค้นสัมผัส (Tangential Stress) พิจารณาในส่วนของแท่งวัตถุตามแนวใดๆ กับความยาวแท่ง แสดงดังรูปที่ 4.2 แรง F ที่กระทำต่อ A' จะแยกออกเป็นแรงในแนวตั้งฉากและแรงขนานกับพื้นที่หน้าตัด A' คือ F_n และ F_t สามารถเขียนแทนด้วยสมการที่ 4.4 และ 4.5 ตามลำดับ ดังนี้

$$S_n = \frac{F_n}{A'} \quad (4.4)$$

และ

$$S_t = \frac{F_t}{A'} \quad (4.5)$$

โดยที่

S_n แทนความเค้นตามแนวตั้งฉากกับพื้นที่หน้าตัด A'

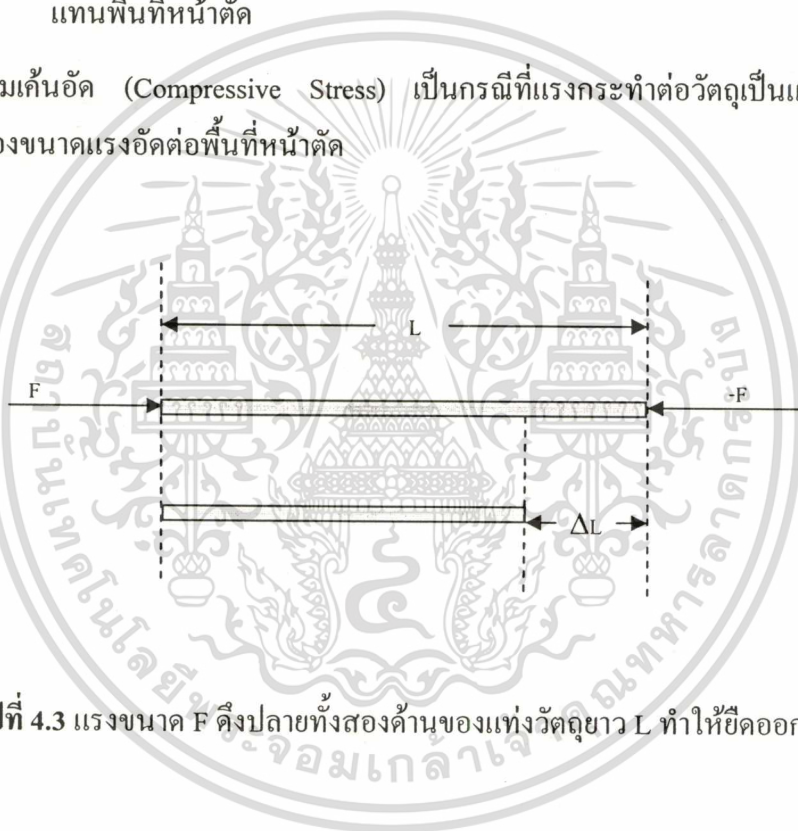
S_t แทนความเค้นตามแนวขนานกับพื้นที่หน้าตัด A'

F_n แทนแรงตามแนวตั้งฉากกับพื้นที่หน้าตัด A'

F_t แทนแรงตามแนวขนานกับพื้นที่หน้าตัด A'

A' แทนพื้นที่หน้าตัด

ความเค้นอัด (Compressive Stress) เป็นกรณีที่แรงกระทำต่อวัตถุเป็นแรงอัด ซึ่งเป็นอัตราส่วนของขนาดแรงอัดต่อพื้นที่หน้าตัด



รูปที่ 4.3 แรงขนาด F ดึงปลายทั้งสองด้านของแท่งวัตถุยาว L ทำให้ยืดออก ΔL

4.3.2 ความเครียด

ความเครียด (Strain) คือ อัตราส่วนระหว่างขนาดหรือรูปร่างที่เปลี่ยนไปกับขนาดหรือรูปร่างเดิม เมื่อมีแรงกระทำต่อวัตถุแล้วมีผลทำให้รูปร่างหรือขนาดของวัตถุนั้นเปลี่ยนไป แสดงดังรูปที่ 4.3 สามารถเขียนได้ดังสมการที่ 4.6

$$\phi_t = \frac{\Delta L}{L} \quad (4.6)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่

ϕ แทนความเครียดดึง

ΔL แทนความยาวของแท่งวัตถุที่เปลี่ยนแปลงไป

L แทนความยาวของแท่งวัตถุ

ความเครียดอัด (Compressive Strain) เป็นกรณีที่แรงกระทำเป็นแรงอัดจะให้ความเครียดอัดคือความยาวที่ลดลงต่อความยาวเดิม

4.3.3 โมดูลัสของความยืดหยุ่น

โมดูลัสของความยืดหยุ่น (Modulus of Elasticity) คือ อัตราส่วนระหว่างความเค้นต่อความเครียด โมดูลัสของวัตถุใดๆ มีค่าคงที่เสมอภายในขอบเขตยืดหยุ่น ทั้งนี้จะขึ้นอยู่กับชนิดของวัสดุด้วย ดังนั้นจากกฎของฮุก (Hooke's Law) ทำให้ทราบว่า ภายในขอบเขตยืดหยุ่นความเค้นเป็นปฏิภาคโดยตรงกับความเครียด ซึ่งในโครงการนี้จะพิจารณาเฉพาะค่าโมดูลัสของยัง ดังนี้

โมดูลัสของยัง (Young's Modulus) คืออัตราส่วนของความเค้นดึงหรืออัดต่อความเครียดดึงหรืออัด ดังสมการที่ 4.7 ดังนี้

$$Y = \frac{F_n / A}{\Delta L / L} \quad (4.7)$$

โดยที่

Y แทนค่าโมดูลัสของยัง

4.4 การออกแบบพื้นผิวกระดาดและสี่เหลี่ยม

4.4.1 โครงสร้างพื้นผิวกระดาด

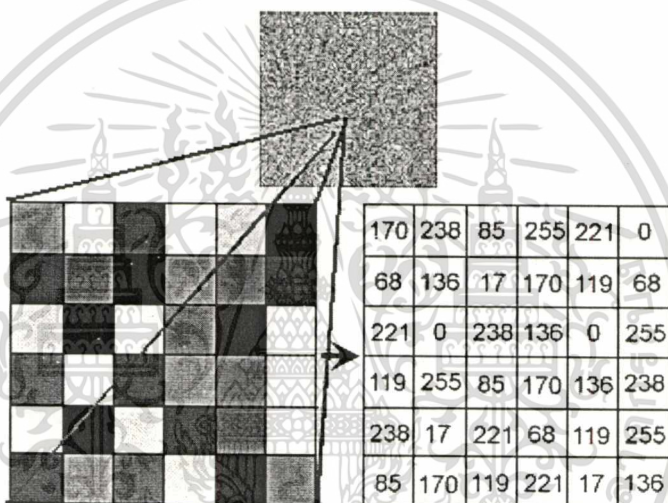
พื้นผิวกระดาดจะออกแบบด้วยไฮท์แมป 2 มิติ (2D Height Map) คือพื้นผิวจะมีลักษณะเป็นตาราง แต่ละตารางจะเรียกกันว่าเซลล์ (Cell) แสดงดังรูปที่ 4.3 และนำค่าความสว่างของแสงใน

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของโรงเรียนเพื่อการศึกษาเท่านั้น เมื่อผู้ยืมหนังสือเรียบร้อยแล้วต้องคืนค่า

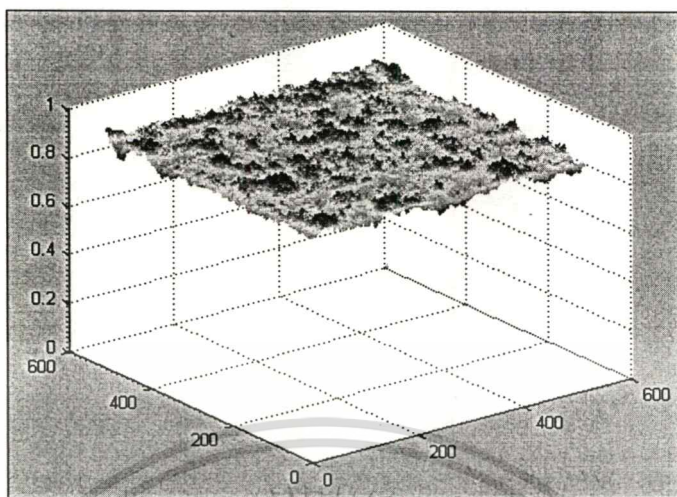
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ละตำแหน่งมาใช้เป็นค่าความสูงของพื้นผิวกระดาษ หรือค่าในแนวแกน Y แสดงดังรูปที่ 4.4 ในแต่ละสดมภ์ (Column) ของเซลล์จะเก็บปริมาณสีที่ชนเป็นชั้น (Layers) ในแต่ละสดมภ์จะตั้งฉากกับระนาบของกระดาษ แสดงดังรูปที่ 4.5 ในแต่ละชั้นสีที่อยู่บนกระดาษจะมีคุณสมบัติ ดังนี้

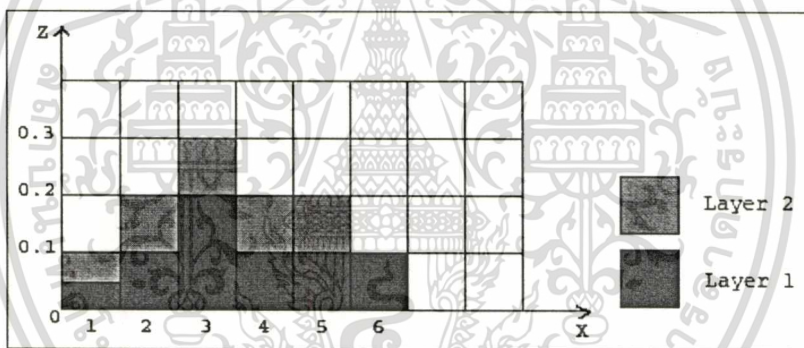
1. ความสูง
2. สี
3. ความสามารถในการทะลุผ่านของแสง (Transmittance)
4. ความสามารถในการสะท้อนของแสง (Scattering)



รูปที่ 4.4 ตัวอย่างลักษณะเซลล์ของพื้นผิวกระดาษ



รูปที่ 4.5 ไฮท์แมป 2 มิติ



รูปที่ 4.6 ชั้นสีเทียนบนกระดาษ

4.4.2 โครงสร้างแท่งสีเทียน

ลักษณะ โครงสร้างของแท่งสีเทียนที่ใช้ในการวาด มีคุณสมบัติ ดังนี้

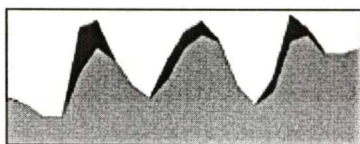
1. กำหนดพื้นที่หน้าตัดเป็นสี่เหลี่ยมจัตุรัส
2. กำหนดความสูงของแต่ละจุดบนพื้นที่หน้าตัด
3. มีค่าความสูง ในช่วง 0.0 – 15.0

4.4.3 โครงสร้างสีเทียนบนกระดาษ

ลักษณะปริมาณสีที่ติดอยู่บนพื้นผิวกระดาษมี 2 แบบ ดังนี้

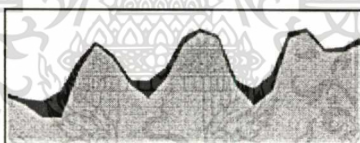
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ดินบนส่วนโค้งบนกระดาศ ลักษณะแบบนี้จะเกิดขึ้นเมื่อบริเวณ ณ. จุดนั้นมีการวาดเป็นครั้งแรกจึงยังทำให้ไม่มีปริมาณสีซึมตกลงไปภายในส่วนโค้งที่เว้าลงไปบนพื้นผิวกระดาศ แสดงดังรูปที่ 4.7



รูปที่ 4.7 ลักษณะสีที่เกิดจากการวาดครั้งแรก

2. ดินบนส่วนโค้งเว้าลงในกระดาศ ลักษณะแบบนี้จะเกิดในกรณีที่บริเวณ ณ. จุดนั้นเคยถูกวาดและมีปริมาณสีติดอยู่ก่อนหน้านั้นแล้ว ทำให้สีที่เคยติดอยู่บนส่วนโค้งบนซึมตกลงมาในส่วนล่าง แสดงดังรูปที่ 4.8



รูปที่ 4.8 ลักษณะสีที่เกิดจากการวาดซ้ำ ณ. จุดเดิม

4.5 การออกแบบกระบวนการวาดสีเทียน

การออกแบบ จะมีการนำเอาทฤษฎี การทดสอบการวาดภาพลายเส้นจากสีเทียนบนกระดาศจริง ทำให้ทราบถึงลักษณะกระบวนการทำงาน มีขั้นตอนการสร้างแบบจำลองได้ดังนี้

ในการสร้างแบบจำลองลายเส้นสีเทียน จะพิจารณาจุดเริ่มต้น P_i ที่อยู่ในพื้นที่ของแท่งสีเทียนและจุดสิ้นสุด P_j มีขั้นตอนในการทำวาด ดังนี้

1. พิจารณาแต่ละจุด P_j ที่เป็นตำแหน่งต่อไปของแต่ละจุด P_i ตามทิศทางการวาด
2. คำนวณแรงทั้งหมดที่เกิดขึ้นบนพื้นผิวแท่งสีเทียน
3. ทำการเกลี่ยสีลงไปในเว้าของพื้นผิวกระดาศ
4. เพิ่มปริมาณสีที่ได้จากการวาดและปรับขนาดของสีเทียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.1 การคำนวณแรงและการปรับขนาดของแท่งสี่เหลี่ยม

เมื่อมีการวาดเส้นด้วยสี่เหลี่ยม ทำให้แท่งสี่เหลี่ยมหดตัวสั้นลงเนื่องจากปริมาณสี่เหลี่ยมใช้ไปและไปติดอยู่บนกระดาษ ซึ่งปริมาณของสี่เหลี่ยมขึ้นอยู่กับผลต่างระหว่างความสูงของมาซซ์คของแท่งสี่เหลี่ยมและความสูงของกระดาษ ณ ตำแหน่งนั้น

โดยจะต้องทำการปรับขนาดความสูงของแท่งสี่เหลี่ยมๆ ครั้งที่มีการเคลื่อนที่ ซึ่งในการเคลื่อนที่และการจำลองการทำงานขนาดของแรงจะไม่คงที่ หรือขนาดของแรงจะเปลี่ยนไปตามแรงกด ดังนั้นแบบจำลองการปรับขนาดของแท่งสี่เหลี่ยม สามารถสร้างได้จากสมการที่ 4.7 ซึ่งเมื่อค่า L มีค่าเข้าใกล้ค่าคงที่ ทำให้สามารถลดรูปของสมการ แสดงดังสมการที่ 4.8

$$F = \lambda A \Delta L \quad (4.8)$$

เมื่อ

F	แทนแรงกด
λ	ค่าคงที่
A	แทนพื้นที่หน้าตัดของแท่งสี่เหลี่ยม
ΔL	แทนขนาดของแท่งสี่เหลี่ยมที่เปลี่ยนแปลงไป

4.5.2 การคำนวณปริมาณสี่เหลี่ยมที่ติดลงบนกระดาษ

เป็นวิธีการหาปริมาณสี่เหลี่ยมที่ถูกปรับปริมาณสี่เหลี่ยมจากแท่งสี่เหลี่ยมไปยังกระดาษว่ามีปริมาณและความสูงเท่าไร ในการสร้างแบบจำลองจะนำแบบจำลองของแรงเสียดทานมาพิจารณา แสดงดังสมการที่ 4.9 ในการพัฒนาอัลกอริทึมจะสนใจเฉพาะแนวแรงในแกนตั้งฉากกับพื้นผิวกระดาษเท่านั้น

$$\vec{F}_F = \mu \vec{F}_N = \mu \vec{N} \frac{\vec{N} \cdot \vec{F}_c}{\|\vec{N}\| \|\vec{F}_c\|} \quad (4.9)$$

เมื่อ

\vec{F}_c	แรงกดของสี่เหลี่ยม
\vec{F}_F	แรงเสียดทาน
\vec{F}_N	แรงกดของสี่เหลี่ยมในแนวตั้งฉากกับพื้นผิว

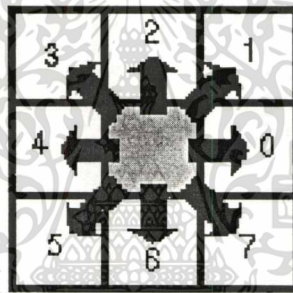
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

N แรงในแนวตั้งฉากกับพื้นผิว

μ สัมประสิทธิ์แรงเสียดทานของพื้นผิวกระดาษ

4.5.3 แบบจำลองการเคลื่อนที่

การทำสี่เหลี่ยมมีลักษณะคล้ายกับการระบายสีน้ำ คือในระหว่างที่แท่งสีเคลื่อนที่ไปบนกระดาษสีที่เขียนที่ติดอยู่ทั้งเก่าและใหม่จะถูกเคลือบหรือซึมลงในบริเวณรอบๆ ในการจำลองการซึมของสี จะนำมาใช้หรือตารางสี่เหลี่ยมขนาด 3×3 เรียกว่า 8-เนบอร์ (8-neighbors) แสดงดังรูปที่ 4.9 เป็นตัวควบคุมตำแหน่งและการกระจายตัวของสีออกไปบริเวณรอบๆ โดยกำหนดให้ตำแหน่งปัจจุบันหรือเซลล์ปัจจุบันเป็นจุดศูนย์กลางของมาซค์ และถูกกำหนดให้มีตำแหน่งเป็น S ,



รูปที่ 4.9 พิกเซลรอบๆ ทั้ง 8 จุด

ในการสร้างมาซค์จะพิจารณาค่าความสูงของกระดาษ รวมทั้งสีเขียน และทิศทาง การเคลื่อนที่ของแท่งสีที่สัมพันธ์กับจุด S สามารถสร้างมาซค์ได้จากสมการที่ 4.10

$$S_{xy} = \frac{1}{\|(x, y)\|} (\alpha \Delta z + \beta \cos((x, y), \vec{V}_C)) \quad (4.10)$$

เมื่อ

S เป็นเมตริกซ์ขนาด 3×3

α ปัจจัยการเคลื่อนตัวของสี

β ปัจจัยทิศทางการซึมตัวของสี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่า β นั้นจะมีค่าตั้งแต่ -1 ถึง 1 พิจารณาตามทิศทางของการวาด ถ้าพิกเซลรอบข้างนั้นเป็นจุดต่อไปของจุดศูนย์กลางของมาซค์ ค่า β ก็จะเป็น 1 แต่ถ้าเป็นจุดที่อยู่ตรงข้ามกับจุดนี้ค่า β ก็จะเป็น -1 เพราะมีทิศทางสวนกับทิศทางของการวาด หรือพูดอีกอย่างได้ว่าพิกเซลรอบข้างนั้นทำมุมกับทิศทางของการวาดเป็นมุม 180 องศา

ส่วนค่า Δz นั้นเป็นความแตกต่างของผลรวมความสูงของกระดาศและปริมาณสีที่ติดอยู่ ณ จุดศูนย์กลางของมาซค์กับพิกเซลรอบข้างใดๆ

ทั้งสองค่านี้จะถูกนำไปคำนวณหาปริมาณสีที่จะเปลี่ยนไปยังพิกเซลรอบข้างต่อไป

4.5.4 การสร้างภาพจากแบบจำลองสีเทียน

ในส่วนนี้จะเป็นการนำแบบจำลองของสีเทียนที่ได้กล่าวไปแล้วข้างต้น มาสร้างเป็นภาพถ่ายเส้น โดยการนำแบบจำลองของคูเบลคา-มังก์ (Kubelka-Munk:KM) อย่างง่ายมาใช้

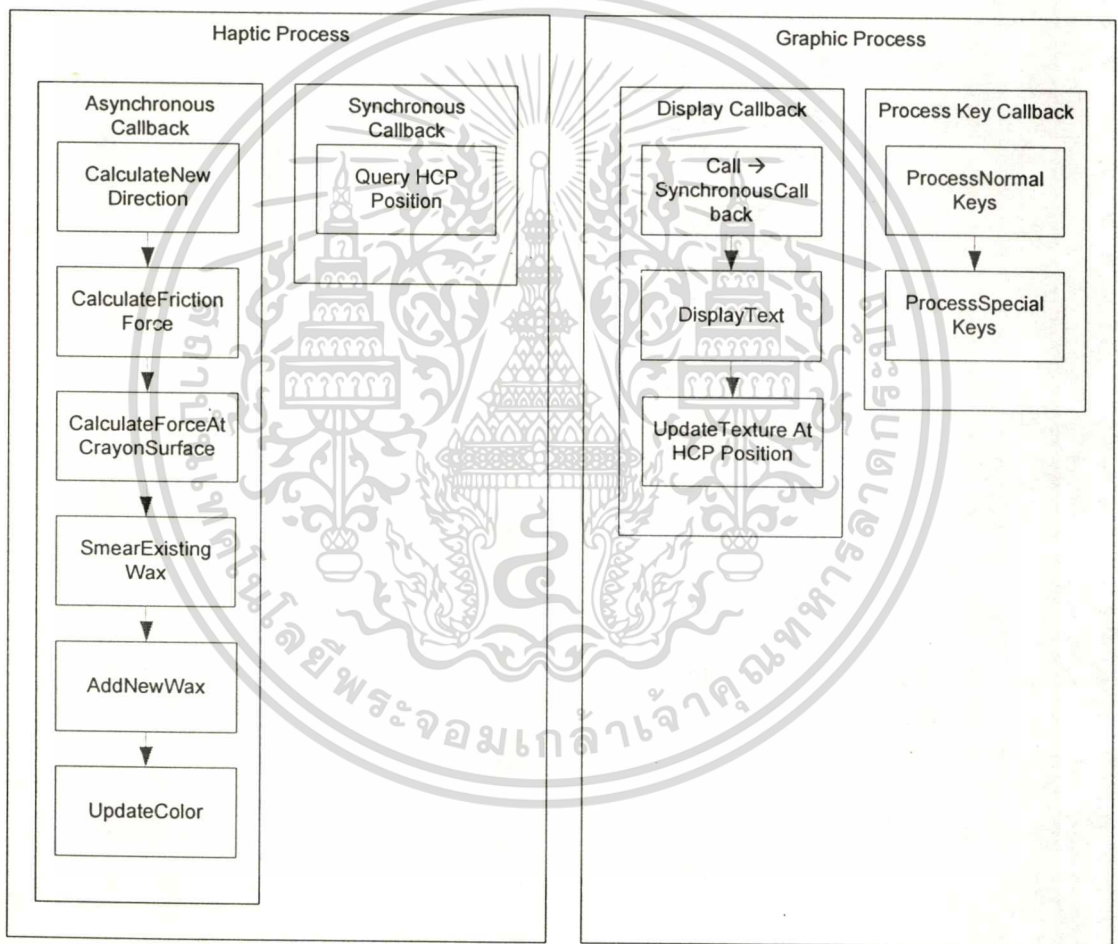
แบบจำลอง KM ใช้ในการประมาณค่าของการทะลุผ่านแสง (Transmittance) และการสะท้อน (Scattering) คุณสมบัติของสีเหล่านี้จะขึ้นอยู่กับสีชั้นบนเปรียบเทียบกับสีของพื้นหลัง สำหรับการสร้างแบบจำลอง

บทที่ 5

การออกแบบโปรแกรมและอัลกอริทึม

5.1 การออกแบบโปรแกรมและอัลกอริทึม

โปรแกรมทั้งหมดของโครงการนั้นพัฒนาโดยใช้เครื่องมือ Microsoft Visual C++ โครงสร้างของโปรแกรมทั้งหมดแสดงดังรูปที่ 5.1



รูปที่ 5.1 โครงสร้างของโปรแกรมทั้งหมด

โปรแกรมจะแบ่งการทำงานออกเป็น 2 ส่วน ส่วนแรกคือ Haptic Process จะทำหน้าที่เกี่ยวกับการจำลองแรงที่เกิดจากการวาดภาพสีเทียน ส่วนที่สองคือ Graphic Process ทำหน้าที่เกี่ยวกับการแสดงผลภาพที่ได้จากการวาดสีเทียน การทำงานของโปรแกรมจะเริ่มจากการคำนวณค่า เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เริ่มต้นต้นของตัวแปรต่างๆ และเข้าสู่ Haptic Process โดยการเรียก Asynchronous Callback ซึ่งภายในประกอบไปด้วยฟังก์ชันที่จำเป็นสำหรับการคำนวณแรงส่งไปยังอุปกรณ์ Haptic และคำนวณภาพสีเทียบเพื่อส่งไปแสดงผลที่ Graphic Process โปรแกรมจะทำงานซ้ำในส่วนนี้จนกว่าจะส่งให้จบการทำงานของ Asynchronous Callback

5.1.1 Haptic Process

ประกอบไปด้วยฟังก์ชันที่จำเป็นสำหรับการคำนวณแรงและภาพจากการจำลองการวาดสีเทียบ แบ่งออกเป็น 2 ส่วนคือ Asynchronous Callback ซึ่งเป็นส่วนที่ประกอบไปด้วยฟังก์ชันที่คำนวณทุกอย่างใน Haptic Process ส่วน Synchronous Callback นั้นจะทำหน้าที่สอบถามข้อมูลตำแหน่งของ HCP ข้อมูลที่ได้จะนำไปใช้สำหรับการแสดงผลของ Graphic Process โดยจะถูกเรียกทุกครั้งที่ Graphic Process ทำงาน และเมื่อได้ข้อมูลก็จะส่งค่ากลับและจบการทำงาน

5.1.2 Graphic Process

ประกอบไปด้วยฟังก์ชันที่เกี่ยวกับการแสดงผลทั้งข้อความและภาพสีเทียบ นอกจากนั้นยังทำหน้าที่ประมวลผลเกี่ยวกับการกดคีย์บอร์ด การทำงานจะมีการเรียกฟังก์ชันที่อยู่ใน Haptic Process คือ Synchronous Callback เพื่อสอบถามข้อมูลตำแหน่งของ HCP และนำไปใช้เกี่ยวกับการคำนวณและแสดงผลภาพสีเทียบต่อไป

5.2 การออกแบบโปรแกรม

ในส่วนนี้จะพูดถึงการออกแบบโปรแกรม โดยประกอบไปด้วยการออกแบบกระดาด แผงสีเทียบ และสีเทียบที่ติดบนกระดาด

5.2.1 การออกแบบกระดาด

กระดาดที่ใช้ในการจำลองการวาดภาพสีเทียบนั้นจะสร้างจากโปรแกรม Adobe Photoshop โดยการเพิ่มจุดรบกวนบนกระดาดสีขาวหรือโดยการใส่ texture ที่มีลักษณะของกระดาด จากนั้นจึงกรองให้กระดาดมีความเรียบ ซึ่งถ้ากระดาดมีความขรุขระมากไป ก็จะเกิดการสั่นในขณะที่วาด

เมื่อได้กระดาดที่มีความขรุขระซึ่งก็คือมีค่าสีของแต่ละพิกเซลที่ไม่เท่ากัน ก็จะนำกระดาดนี้ไปใช้ในโปรแกรมต่อไป โดยกระดาดจะมีผลต่อทิศทางและขนาดของแรงตอบสนอง ทิศทางของแรงนั้นได้อธิบายในบทที่ 3 ส่วนผลของกระดาดที่มีต่อขนาดของแรงนั้น เป็นเพราะว่าในการ

จำนวนแรงของ Haptic จะคำนวณได้จาก ระยะที่ HCP ทะลุเข้าไปในพื้นที่กระดาด เนื่องจากกระดาด มีความสูงไม่เท่ากันจึงทำให้ขนาดของแรงตอบสนองไม่เท่ากันด้วย

5.2.2 การออกแบบแท่งสี่เหลี่ยม

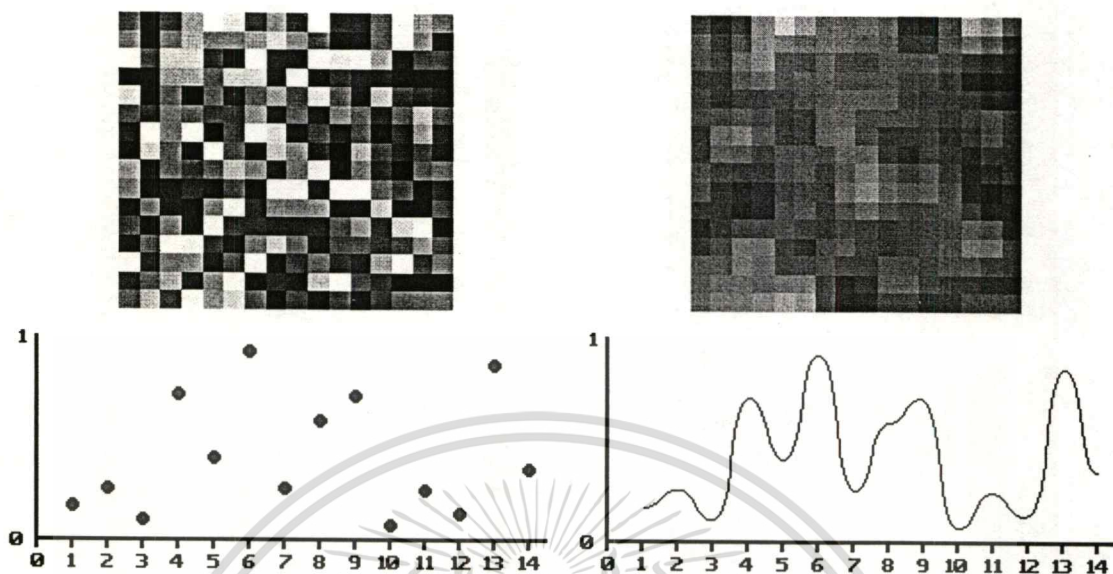
แท่งสี่เหลี่ยมที่ใช้จะมีขนาดจำกัดค่าหนึ่ง ในขณะที่วาล์วสี่เหลี่ยมจะหลุดออกจากแท่งสี่เหลี่ยม เป็นปริมาณเท่ากับปริมาณที่ติดอยู่บนกระดาด นั่นคือเมื่อเพิ่มสี่เหลี่ยมบนกระดาดแล้วจะต้องไปปรับ ความสูงของแท่งสี่เหลี่ยมด้วย

5.2.3 ปริมาณสี่เหลี่ยมมีผลต่อการคำนวณแรงและลักษณะลายเส้นสี่เหลี่ยม

การจำลองสี่เหลี่ยมที่ติดอยู่บนกระดาดนั้น ปริมาณสี่เหลี่ยมจะมีผลต่อแรงลัพธ์ที่จะส่งไปยัง อุปกรณ์และมีผลต่อลักษณะลายเส้นที่เขียนด้วย สำหรับการคำนวณแรงนั้น ปริมาณสี่เหลี่ยมจะมีผล ต่อขนาดของแรงตอบสนองและแรงเสียดทาน ส่วนผลต่อลายเส้นนั้นจะมีผลตอนที่ปรับค่าสีโดย ความสูงของสี่เหลี่ยมที่ติดอยู่บนกระดาดนั้นจะมีผลต่อการคำนวณค่าสีตามอัลกอริทึมของ Kubelka-munk นอกจากนี้ยังมีผลทำให้ปริมาณสี่เหลี่ยมที่จะเพิ่มบนกระดาดน้อยลงเมื่อสี่เหลี่ยมที่ติดบน กระดาดตรงจุดนั้นมีอยู่มาก

5.2.4 การปรับทิศทางการตอบสนองของแรง

เนื่องจากทิศทางการตอบสนองของแรงที่คำนวณจากกระดาดที่จำลองนั้นไม่มีความ ต่อเนื่อง ซึ่งจะทำให้เกิดการสั่นของอุปกรณ์ในระหว่างการจำลอง ดังนั้นจึงต้องมีกระบวนการที่ทำให้ ทิศทางการตอบสนองของแรงนั้นมีความต่อเนื่องด้วย ซึ่งสามารถทำได้โดยการใช้กระบวนการ ทำ Interpolation ซึ่งมีอยู่หลายวิธีโดยใน โปรแกรมที่พัฒนานั้นได้เลือกใช้ Bilinear Interpolation ด้วยเหตุผลที่ วิธีนี้ให้ผลที่ยอมรับได้ ทิศทางของแรงตอบสนองนั้นมีความต่อเนื่องมากขึ้น แต่ก็ยังไม่ดีนักเพราะก็ยังมี การสั่นอยู่บ้าง แต่สามารถแก้ปัญหาได้โดยการลดค่าความแข็งของกระดาดลง

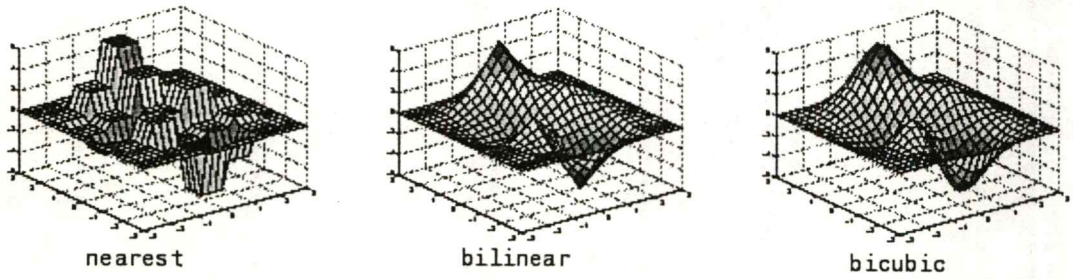


รูปที่ 5.2 การ Interpolation เพื่อให้ทิศทางมีความต่อเนื่อง

การเปรียบเทียบและเหตุผลที่เลือกใช้ Bilinear Interpolation แทนที่จะใช้ Bicubic Interpolation ซึ่งให้ผลที่มีความต่อเนื่องมากกว่านั้น เนื่องจากว่าความขรุขระที่เกิดจากการจำลองกระดานนั้น ขึ้นอยู่กับความไม่ต่อเนื่องของพื้นและขนาดของความแตกต่างของความสูง แนวทางการจำลองคายนั้นมี 2 แบบ คือ

1. ทำให้พื้นมีความแตกต่างกันอย่างต่อเนื่อง การจำลองก็จะรู้สึกว่ามันเรียบมาก ซึ่งอาจจะไม่ตรงกับที่ต้องการ และเพื่อที่จะให้รู้สึกถึงความขรุขระเพิ่มขึ้นจะต้องปรับค่าที่เหลือซึ่งก็คือ ค่าความแตกต่างของความสูงของกระดาน โดยปรับให้มีค่ามากขึ้น
2. ทำให้พื้นมีความแตกต่างที่ต่อเนื่องน้อยกว่า แต่จะลดความแตกต่างของความสูงของพื้นกระดานลง ก็จะทำได้ผลที่คล้ายๆ กัน

การเปรียบเทียบระหว่าง Bilinear Interpolation และ Bicubic Interpolation นั้น Bilinear จะให้ผลที่มีความต่อเนื่องน้อยกว่า แต่จะคำนวณน้อยกว่า ซึ่งถ้าดูจากแนวทางการจำลองกระดานสามารถที่จะเลือกใช้ Bilinear Interpolation แล้วกำหนดให้กระดานมีความแตกต่างของความสูงไม่มากเกินไป พร้อมทั้งลดค่าความแข็งของกระดานลง ก็จะได้ผลที่น่าพอใจและสามารถคำนวณแรงได้ถูกต้องครบถ้วนด้วย (กรณีที่มีการคำนวณมีมากเกินไปจะทำให้การคำนวณแรงมีความผิดพลาดได้)



รูปที่ 5.3 เปรียบเทียบวิธีการ Interpolation ทั้ง 3 วิธี

5.3 อัลกอริทึมสำหรับการจำลองแรงสั่นผัด

1. CalculateNewDirection เป็นฟังก์ชันที่สร้างทิศทางของแรงตอบสนองของพื้นผิวที่ถูกทำให้เบี่ยงเบนออกไป perturbedDirection จากนั้นจึงนำค่านี้ไปคำนวณโดยใช้เป็นทิศทางของแรงที่ตอบสนองหรือแรงจากการจำลองพื้นผิวกระดาศ

```

...// CalculateNewDirection
getImage(imageName) /*เป็นการโหลดและเก็บค่าสีจากภาพที่ใช้สำหรับแทนพื้นผิวของ
กระดาศ*/
convertToGray(image) /*เปลี่ยนค่าสีให้เป็น gray scale*/
calculateGradient(grayImage) /*คำนวณหาค่า local gradient*/
calculatePerturbedDirection(gradientImage) /*คำนวณทิศทางจากค่า local gradient*/
...

```

```

..// calculatePerturbedDirection
w = image.width
h = image.height
for each i, j
    alpha.set(0.0, gradientX[(j * w) + i], 1.0)
    beta.set(1.0, gradientY[(j * w) + i], 0.0)
    MVector = crossProduct(alpha, beta)
    crossProductMag[(j * w) + i] = MVector.magnitude
    MVector.normalize()
    perturbedDirection[(j * w) + i] = MVector/*ใส่ค่าให้ทั้งแกน x y z*/
...

```

2. CalculateFrictionForce เป็นฟังก์ชันสำหรับการคำนวณแรงเสียดทานที่เกิดจากการวาด ซึ่งสามารถจำลองได้ทั้งแรงเสียดทานสถิตและแรงเสียดทานจลน์ นอกจากนั้นยังจำลองแรงหน่วง การเคลื่อนที่ ซึ่งมีทิศทางตรงข้ามการเคลื่อนที่ และมีค่าขึ้นกับความเร็วที่เคลื่อนที่ ค่าแรงเสียดทาน ที่ได้คือ *frictionForce* โดยจะนำไปรวมกับแรงที่เกิดจากการจำลองพื้นผิวกระดาษ หรือแรงที่ทำให้ รู้สึกว่าพื้นผิวขรุขระนั่นเอง

```

...// CalculateFrictionForce
if(state == stuck)
    friction =- frictionStiffness*tangentDistance*direction
    if(abs(friction) > staticFriction)
        state = sliding
        friction = -direction*dynamicFriction
    frictionForce = friction
else
    if(abs(velocity.magnitude < velocityThreshold))
        state = stuck
        stuckPosition = position +
            (staticFrictionD/frictionStiffness)*frictionFDirection
        frictionF = -frictionStiffness*(position - stuckPosition)
    else
        frictionF = -dynamicFrictionD*velocityDirection
    frictionForce = frictionF
    frictionFDirection = friction.normalize
...

```

5.4 อัลกอริทึมสำหรับการสร้างลายเส้นสีเทียน

1. CalculateForceAtCrayonSurface เป็นฟังก์ชันสำหรับคำนวณแรงที่เกิดบนพื้นผิวของแท่งสีเทียน โดยที่ crFactor เป็นค่าความยืดหยุ่นของสีเทียน

```

...// CalculateForceAtCrayonSurface
for each i
    crayon.stressedLength[i] = crayon.maskPaper[i] + crayon.waxHeight[i] -
        (position[1] - crayon.maskCell[i])
    if(crayon.stressedLength[i] > 0)
        stressedForce += (crayon.stressedLength[i] * crFactor)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. SmearExistingWax เป็นฟังก์ชันที่ใช้เกลี่ยสีไปยังพิกเซลรอบข้างทั้งแปดจุด โดยปริมาณสีที่เกลี่ยไปยังแต่ละพิกเซลนั้นจะขึ้นอยู่กับมุมระหว่างตำแหน่งพิกเซลนั้นกับทิศทางของการวาดภาพ นอกจากนี้ยังขึ้นอยู่กับความแตกต่างของความสูงระหว่างพิกเซลตรงกลางกับพิกเซลรอบข้างแต่ละจุด

angle = มุมระหว่างตำแหน่งพิกเซลนั้นกับทิศทางของการวาดภาพ

directionalFactor = $\cos(\text{angle})$

กำหนดให้ U เป็นยูนิตเวกเตอร์ที่แทนตำแหน่ง i, j ของพิกเซล

ตารางที่ 5.1 ค่าตำแหน่งของพิกเซลรอบข้าง

-1, -1	0, -1	1, -1
-1, 0	0, 0	1, 0
-1, 1	0, 1	1, 1

กำหนดให้ V เป็นยูนิตเวกเตอร์ที่แทนทิศทางของการเคลื่อนที่

$U \cdot V = U.\text{magnitude} * V.\text{magnitude} * \cos(\text{มุมระหว่างUV})$

$U \cdot V = U.x * V.x + U.y * V.y$

$\cos(\text{มุมระหว่างUV}) = (i * V.x + j * V.y) / (V.\text{magnitude} * \text{pow}((i*i + j*j), 0.5))$

เมื่อได้ค่า directionalFactor แล้วจึงหาค่า heightDifFactor ซึ่งเป็นค่าความแตกต่างของความสูงระหว่างพิกเซลตรงกลางกับพิกเซลรอบข้าง

ปริมาณสีที่เกลี่ยไปยังพิกเซลรอบข้างแต่ละจุด คำนวณได้จาก

$\text{smearFactor} = \max(0, \text{directionalFactor} + \text{heightDifFactor})$

$\text{smearedWax} = (\text{smearFactor} * \text{centerPixelWax}) / \text{sumOfsmearedWax}$

3. AddNewWax เป็นฟังก์ชันที่คำนวณปริมาณสีที่ติดอยู่บนกระดาษจากการวาด ปัจจุบันที่มีผลต่อปริมาณสีที่ติดบนกระดาษนั้นได้แก่ ค่าของแรงที่กดลงบนกระดาษ ปริมาณสีที่ติดอยู่ที่ตำแหน่งนั้น ความชันระหว่างจุดที่กดกับจุดต่อไปตามทิศทางการวาดสีเทียน

ปริมาณสีเทียนที่ติดอยู่บนกระดาษจะนำไปคำนวณหาค่าน้ำหนักของค่าสัมประสิทธิ์การวาดของกระดาษกับสัมประสิทธิ์การวาดของสีเทียน ซึ่งตามความจริงแล้วเมื่อจุดนั้นเคยถูกวาดด้วยสีเทียนแล้วจะทำให้ปริมาณสีเทียนที่จะเพิ่มจากการวาดทับนั้นลดน้อยลง ดังนั้นค่าสัมประสิทธิ์ของสีเทียนควรจะมีค่าน้อยกว่าของกระดาษ

```

..AddNewWax

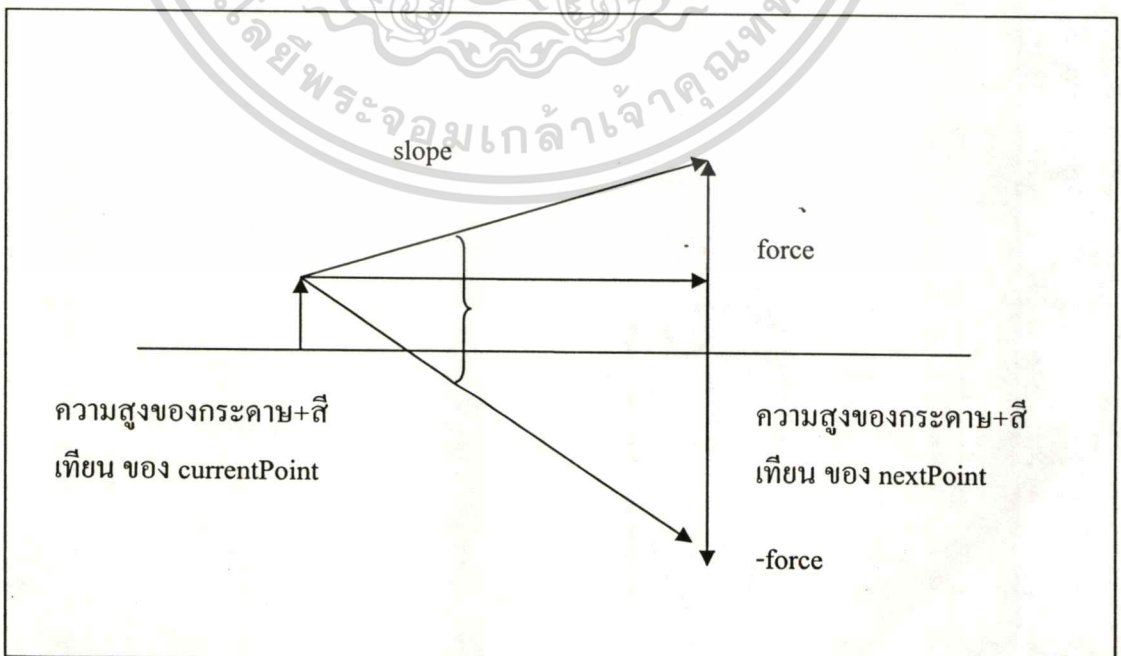
findNextPoint(tanVelocity)/*ค่า tan ของความเร็วหรือทิศทางการวาดในแนวระนาบ x
z*/

slope = calculateSlope(stuckPoint, nextPoint)

force = calculateForce()

drawFactor = calculateDrawFactor()

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. UpdateColor เป็นฟังก์ชันที่คำนวณค่าสีที่เกิดจากการวาดภาพสีเทียมน พัฒนาตามโมเดลของ Kubelka-Munk Theory โดยใช้หลักการง่ายๆ

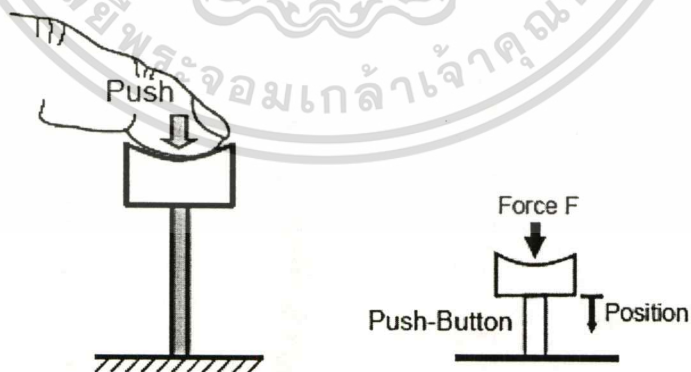
```

...// UpdateColor
//wow เป็น factor ที่ใช้ปรับผลของความสูงที่มีต่อค่าสี
wax.transmitC = pow((currentColor.transmit * currentColor), waxHeight*wow) * wax.color;
wax.scatterC = 1 - pow((1 - currentColor), (currentColor.scatter * waxHeight*wow))
wax.color = wax.transmitC + wax.scatterC
//wax.color นั้นจะต้องทำทั้ง R G B แยกกัน

```

5.5 การออกแบบ user-interface

user-interface สำหรับรับข้อมูลจากผู้ใช้ที่มีอยู่ใน โปรแกรมประกอบไปด้วย 2 ส่วนคือ ส่วนแรกเป็นการติดต่อกับผู้ใช้ผ่านคีย์บอร์ด และส่วนที่สองเป็น user-interface ที่สร้างโดยการใช้อุปกรณ์ haptic ซึ่งผู้ใช้งานจะรู้สึกถึงการมีอยู่ของ user-interface ด้วย



รูปที่ 5.4 การจำลองปุ่มกดแบบกดติดปล่อยดับ

การออกแบบ user-interface ในส่วนของการกดคีย์นั้นจะใช้สำหรับการปรับเปลี่ยนมุมมอง

ให้เหมาะสำหรับการวาด โดยการกดคีย์ตามคำอธิบายบนจอแสดงผล

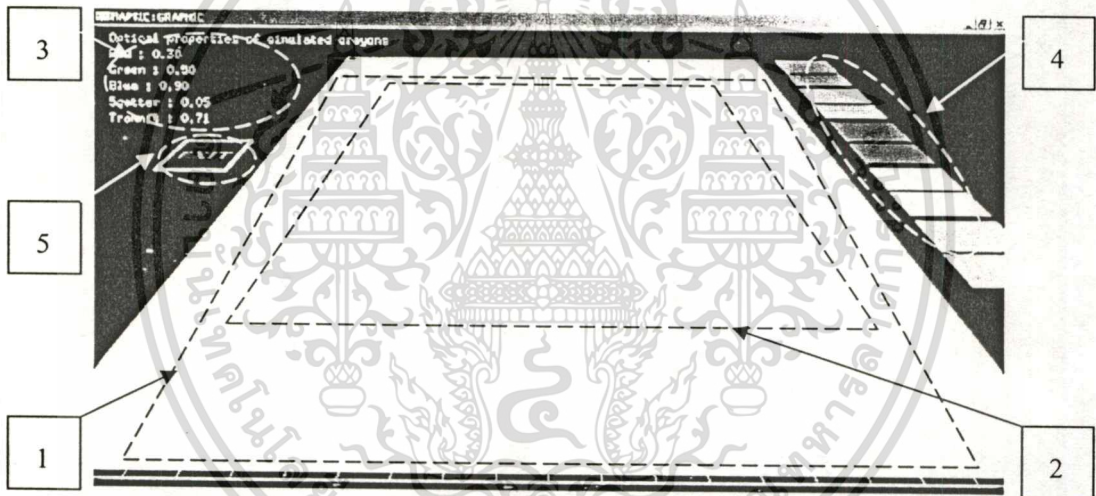
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบ user-interface ในส่วนของปุ่มกดนั้นจะกำหนดให้เป็นแบบ กดติดปล่อยดับ เพราะว่าปุ่มนี้จะนำไปใช้สำหรับให้ผู้ใช้งานปรับเปลี่ยนค่าสีของแท่งสีเทียน เมื่อผู้ใช้กดแท่งสีเทียนบนปุ่มนี้จะรู้สึกเหมือนกับว่ากำลังกดปุ่มของจริงคือจะมีแรงต้านดันมือออกจากปุ่ม เมื่อกดปุ่มค้างไว้ค่าพารามิเตอร์ที่ปุ่มนั้นเป็นตัวกำหนดก็จะเปลี่ยนไปซึ่งอาจจะมีค่าเพิ่มขึ้นหรือลดลง สังเกตได้จากค่าที่แสดงผลบนจอภาพ

คีย์ที่ใช้ในโปรแกรมมีดังนี้

1. ลูกศรเลื่อนขึ้น ลง ซ้าย ขวา จะใช้สำหรับปรับมุมมองของการวาด
2. ปุ่ม Z จะใช้สำหรับปรับมุมมองของการวาด โดยกดเพียงครั้งเดียว ขึ้นตอนภายในประกอบไปด้วยการกดปุ่มลูกศรเพื่อปรับให้เหมาะสม



รูปที่ 5.5 อธิบาย user-interface

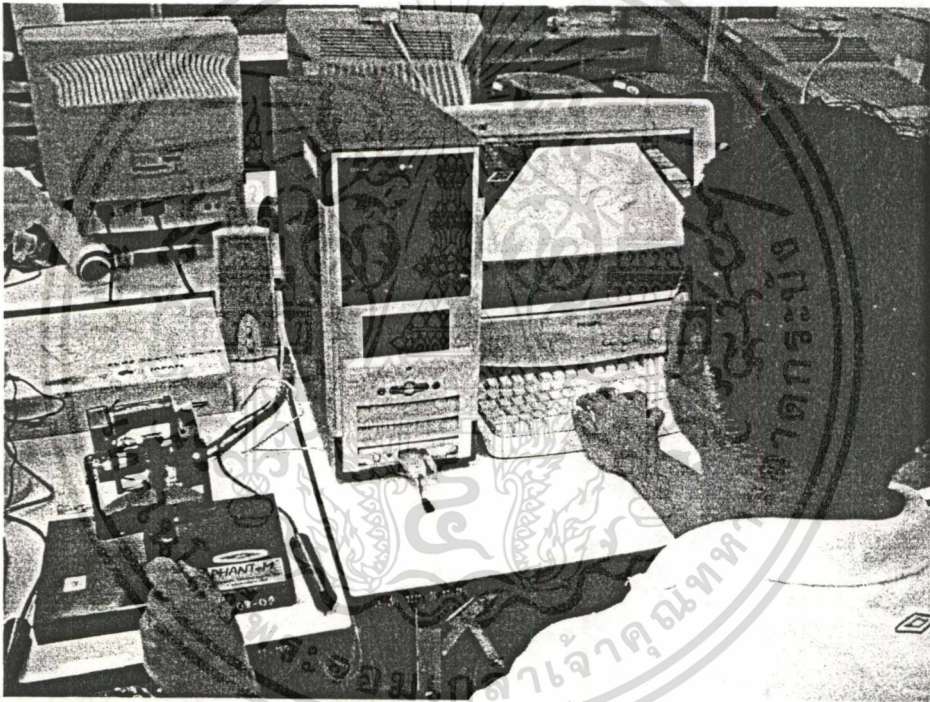
อธิบาย user-interface ที่มีอยู่ในโปรแกรม

1. พื้นที่สำหรับการวาดสีเทียน
2. ร่างต้นแบบก่อนการวาดสีเทียน
3. พื้นที่แสดงผลค่าองค์ประกอบต่างๆ ของสี
4. ปุ่มที่กดด้วยอุปกรณ์ haptic ใช้สำหรับเปลี่ยนค่าองค์ประกอบของสี
5. ปุ่มที่กดด้วยอุปกรณ์ haptic ใช้สำหรับออกจากโปรแกรม

บทที่ 6

ผลการพัฒนาและทดสอบโปรแกรม








จากการดำเนินโครงการผลที่ได้คือ โปรแกรมจำลองสัมผัสของการวาดภาพด้วยสีเทียนซึ่งจะเป็นโปรแกรมจำลองการทำงานของการวาดภาพ



รูปที่ 6.1 โปรแกรมวาดภาพบนโมเดลสามมิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 6.1 ตัวอย่างสีเทียนและคุณสมบัติทางออปติคัล

Crayon	Color	R	G	B	T	S
	Yellow	0.950	0.550	0.200	0.605	0.042
	Blue	0.300	0.500	0.900	0.715	0.054
	Red	0.950	0.450	0.450	0.065	0.051
	Green	0.350	0.800	0.350	0.385	0.096
	Periwinkle	0.700	0.700	0.900	0.385	0.210
	Sea green	0.600	0.900	0.650	0.330	0.150
	Orchid	0.850	0.400	0.840	0.660	0.090

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.2 ภาพวาดโดยใช้โปรแกรมที่พัฒนา

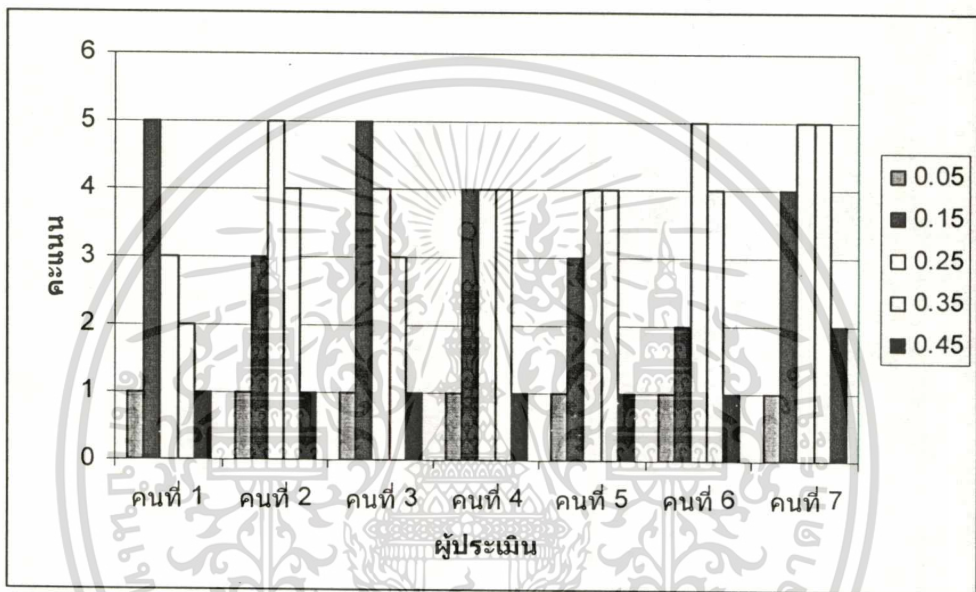
6.1 สรุปผลการประเมินความพึงพอใจ

เป็นการประเมินจากการเปรียบเทียบระหว่างความรู้สึกของการวาดสีเทียนที่จำลองด้วยโปรแกรมกับความรู้สึกเมื่อวาดด้วยสีเทียนจริง โดยการประเมินนั้นจะเปลี่ยนค่าตัวแปรบางตัวที่มีผลชัดเจนต่อความรู้สึกเมื่อวาดสีเทียน ซึ่งผลการประเมินที่ได้รับจากผู้ประเมินทั้งหมด 7 คน แสดงดังตารางที่ 6.2, 6.3 และ 6.4

ตัวแปรที่มีผลชัดเจนต่อความรู้สึกของการวาดสีเทียนนั้นประกอบไปด้วย ค่าความนุ่มของกระดาษ ค่าแรงเสียดทานของกระดาษ ค่าสัมประสิทธิ์ความหนืดของสีเทียน โดยจะมีผลต่อความรู้สึก นุ่ม ความฝืด และความหนืด ขณะที่วาดสีเทียน ในการประเมินจะแยกประเมินความพึงพอใจที่มีต่อความรู้สึกทั้งสาม

6.1.1 การประเมินความพึงพอใจต่อการวาดสี่เหลี่ยมด้วยโปรแกรมเปรียบเทียบกับกรวดด้วยสี่เหลี่ยมจริง ในด้านความนุ่มของกระดาษ

ตัวแปรที่มีผลชัดเจนต่อความนุ่มของกระดาษคือ planeStiffness การประเมินจะเปลี่ยนค่าตัวแปรนี้ทั้งหมด 5 ระดับ ผลที่ได้จากการประเมิน เป็นคะแนนที่มี 5 ระดับดังนี้
1 – ต่ำมาก, 2 – ต่ำ, 3 - ปานกลาง, 4 – สูง, 5 – สูงมาก



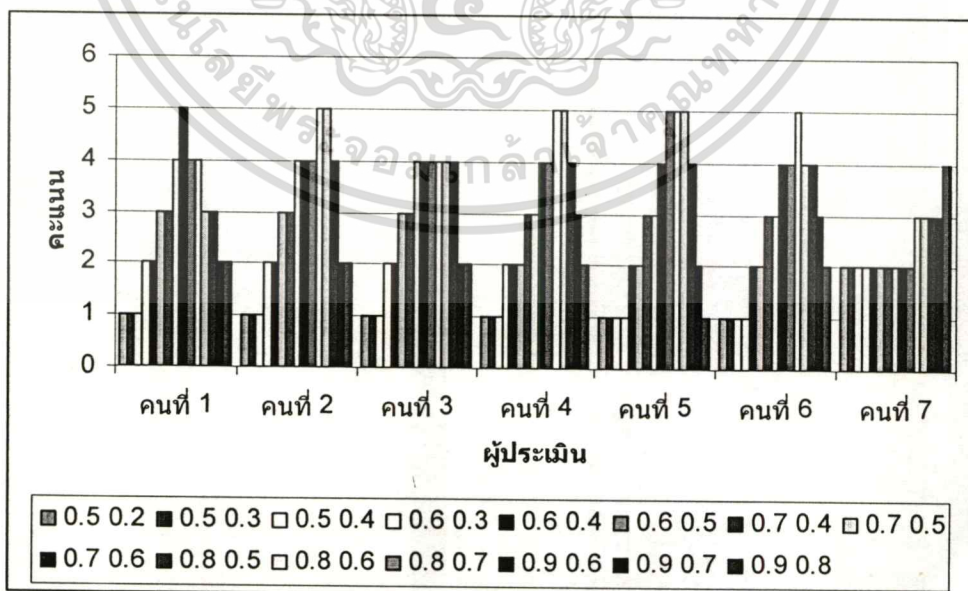
รูปที่ 6.3 ผลการประเมินความพึงพอใจต่อความนุ่มของกระดาษ

ตารางที่ 6.2 ผลการประเมินความพึงพอใจต่อการวาดสี่เหลี่ยมด้วยโปรแกรมเปรียบเทียบกับการวาดด้วยสี่เหลี่ยมจริง ในด้านความนุ่มของกระดาษ

planeStiffness	คะแนนเฉลี่ย
0.05	1.00
0.15	3.71
0.25	4.28
0.35	3.71
0.45	1.14

6.1.2 การประเมินความพึงพอใจต่อการวาดสี่เหลี่ยมด้วยโปรแกรมเปรียบเทียบกับการวาดด้วยสี่เหลี่ยมจริง ในด้านความฝืดของกระดาษ

ตัวแปรที่มีผลชัดเจนต่อความฝืดของกระดาษคือ staticCoefficient, dynamicCoefficient การประเมินจะเปลี่ยนค่าตัวแปรทั้งสองค่านี้ โดยหลักการแล้วค่า staticCoefficient จะมีค่ามากกว่า dynamicCoefficient ผลที่ได้จากการประเมิน เป็นคะแนนที่มี 5 ระดับดังนี้
1 - ต่ำมาก, 2 - ต่ำ, 3 - ปานกลาง, 4 - สูง, 5 - สูงมาก



รูปที่ 6.4 ผลการประเมินความพึงพอใจต่อความฝืด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

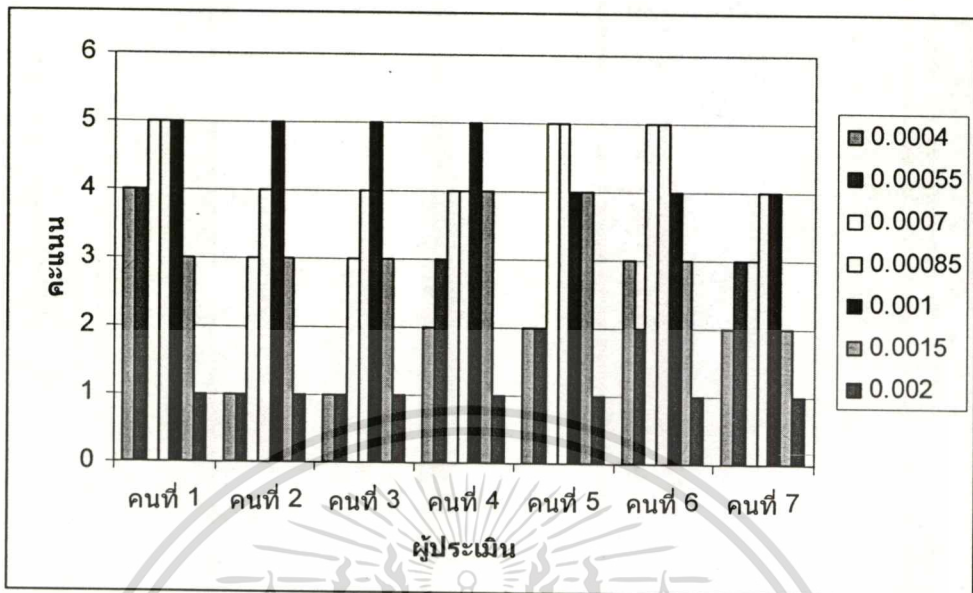
ตารางที่ 6.3 ผลการประเมินความพึงพอใจต่อการวาดสี่เหลี่ยมด้วยโปรแกรมเปรียบเทียบกับการวาดด้วยสี่เหลี่ยมจริง ในด้านความฝืดของกระดาษ

staticCoefficient	dynamicCoefficient	คะแนนเฉลี่ย
0.5	0.2	1.14
0.5	0.3	1.14
0.5	0.4	1.14
0.6	0.3	1.71
0.6	0.4	2.00
0.6	0.5	2.42
0.7	0.4	2.85
0.7	0.5	3.28
0.7	0.6	3.85
0.8	0.5	3.85
0.8	0.6	4.42
0.8	0.7	4.14
0.9	0.6	3.71
0.9	0.7	2.42
0.9	0.8	2.14

6.1.3 การประเมินความพึงพอใจต่อการวาดสี่เหลี่ยมด้วยโปรแกรมเปรียบเทียบกับการวาดด้วยสี่เหลี่ยมจริง ในด้านความหนืดของกระดาษ

ตัวแปรที่มีผลชัดเจนต่อความหนืดของกระดาษคือ $kDamping$ การประเมินจะเปลี่ยนค่าตัวแปรนี้ทั้งหมด 5 ระดับ ผลที่ได้จากการประเมิน เป็นคะแนนที่มี 5 ระดับดังนี้

1 – ต่ำมาก, 2 – ต่ำ, 3 - ปานกลาง, 4 – สูง, 5 – สูงมาก



รูปที่ 6.5 ผลการประเมินความพึงพอใจต่อความหนืด

ตารางที่ 6.4 ผลการประเมินความพึงพอใจต่อการวาดสี่เหลี่ยมด้วยโปรแกรมเปรียบเทียบกับการวาดด้วยสี่เหลี่ยมจริง ในด้านความหนืดของกระดาษ

kDamping	คะแนนเฉลี่ย
0.0004	2.14
0.00055	2.28
0.0007	4.00
0.00085	4.42
0.001	4.57
0.0015	3.14
0.002	1.00

บทที่ 7

สรุปผลการดำเนินโครงการ

ผลการดำเนินโครงการสรุปเป็นสรุปผลการดำเนินโครงการ ข้อจำกัดของโครงการ รวมถึงข้อเสนอแนะในการพัฒนาระบบต่อไป

7.1 สรุปการดำเนินโครงการ

จากการศึกษาและพัฒนาโปรแกรมวาดภาพสีเทียนโดยใช้อุปกรณ์ตอบสนองต่อแรง โดยได้ศึกษาจากงานวิจัยที่เกี่ยวข้อง ซึ่งบางงานวิจัยก็ไม่ได้นำอุปกรณ์นี้มาใช้ ความรู้ที่ได้จากการศึกษา งานวิจัยที่ผ่านมาและจากหลักการที่ได้นำเสนอในบทที่ผ่านมา ช่วยให้เราสามารถพัฒนาโปรแกรมจำลองการวาดภาพสีเทียน โดยใช้อุปกรณ์ตอบสนองต่อแรงได้

7.2 ข้อจำกัดและปัญหาของโครงการ

การดำเนินโครงการครั้งนี้ยังมีข้อจำกัดอยู่หลายประการ คือ

1. สีเทียนที่ได้ทดลองยังมีน้อยไป
2. มุมของการเอียงแท่งสีเทียนจะไม่มีผลต่อแรงและสายเส้นที่จะจำลอง
3. ส่วนติดต่อกับผู้ใช้ยังไม่เหมาะสมและสวยงามมากพอ
4. การเขียนโค้ดไม่ดีพอเนื่องจากเป็นโปรแกรมที่ใช้สำหรับนำเสนอ โปรแกรมใช้หน่วยความจำเปลือง

7.3 ข้อเสนอแนะ

1. ในการดำเนินการต่อ ส่วนติดต่อกับผู้ใช้ต้องทำให้ดีกว่านี้
2. เพิ่มสีเทียนเข้าไปในโปรแกรมอีก
3. พัฒนาโปรแกรมมีเป็นระบบกว่านี้โดยใช้การออกแบบเชิงวัตถุ

บรรณานุกรม

- ขวัญเรือน โสอุบล. 2545. “การพัฒนาโปรแกรมจำลองแรงสัมผัสของการวาดภาพด้วยสีเทียนโดยใช้อุปกรณ์จำลองการสัมผัส.” โครงการพัฒนาระบบงานวิทยาศาสตร์มหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.
- ธีรเดช โอทกานนท์. 2545. “การพัฒนาโปรแกรมจำลองแรงสัมผัสของการวาดภาพด้วยสีเทียนโดยใช้อุปกรณ์จำลองการสัมผัส.” โครงการพัฒนาระบบงานวิทยาศาสตร์มหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.
- อุราเคน กิจพยัคฆ์. 2545. “การพัฒนา Collaborative Network Game โดยใช้อุปกรณ์ตอบสนองด้วยแรงต้าน.” โครงการพัฒนาระบบงานวิทยาศาสตร์มหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.
- Ho,C.H. et al. 1999. “Efficient point-based rendering techniques for haptic display of Virtual objects.” Presence: 477-491.
- Minsky, M. 1995. “Computational Haptics: The sandpaper system for synthesizing texture for a force-feedback display.” PhD thesis, Ph.D. Dissertation, Program in Media Arts and Sciences, MIT.
- Rudolf, D. et al. 2003. “Simulating Wax Crayon.” In Proceedings of the 11th Pacific Conference on Computer Graphics and Applications (PG’03). IEEE Computer Society.
- Sensible Technologies. 1998. **GHOST SDK Programmer’s Guide Version 4.0.** Cambridge: Sensible Technologies.
- S.E. Salcudean and T. Vlaar. 1997. “On the Emulation of Stiff Walls and Static Friction with a Magnetically Levitated Input-Output Device.” ASME Journal of Dynamics, 119:127–132.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก
พารามิเตอร์สำหรับการสร้างแบบจำลอง

1. พารามิเตอร์ที่ใช้สำหรับการสร้างแบบจำลอง

ค่าคงที่ที่ใช้เป็นตัวแปรและกำหนดค่าในการพัฒนา

ตารางที่ ก.1 พารามิเตอร์คงที่

ตัวแปร	คำอธิบาย	ค่า
Paper Friction	สัมประสิทธิ์แรงเสียดทานของ wax สำหรับลายเส้น	0.15
Wax Friction	สัมประสิทธิ์แรงเสียดทานของกระดาษ สำหรับลายเส้น	0.75
crFactor	ความต้านทานแรงกดของ wax	2.0
Max Contact	ความสูงที่เปลี่ยนแปลงไป	0.0001
Plane Stiffness	ความแข็งของกระดาษ	0.25
Wax Stiffness	ความแข็งของสีเทียน	0.2
Friction Stiffness	ความแข็งของกระดาษสำหรับคำนวณแรงเสียดทาน	0.15
Static Coefficient	สัมประสิทธิ์แรงเสียดทานสถิต	0.8
Dynamic Coefficient	สัมประสิทธิ์แรงเสียดทานจลน์	0.7
VThreshold	Threshold ของความเร็วการวาดสีเทียน	45
kDamping	สัมประสิทธิ์แรงหน่วงของการวาดสีเทียน	0.001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

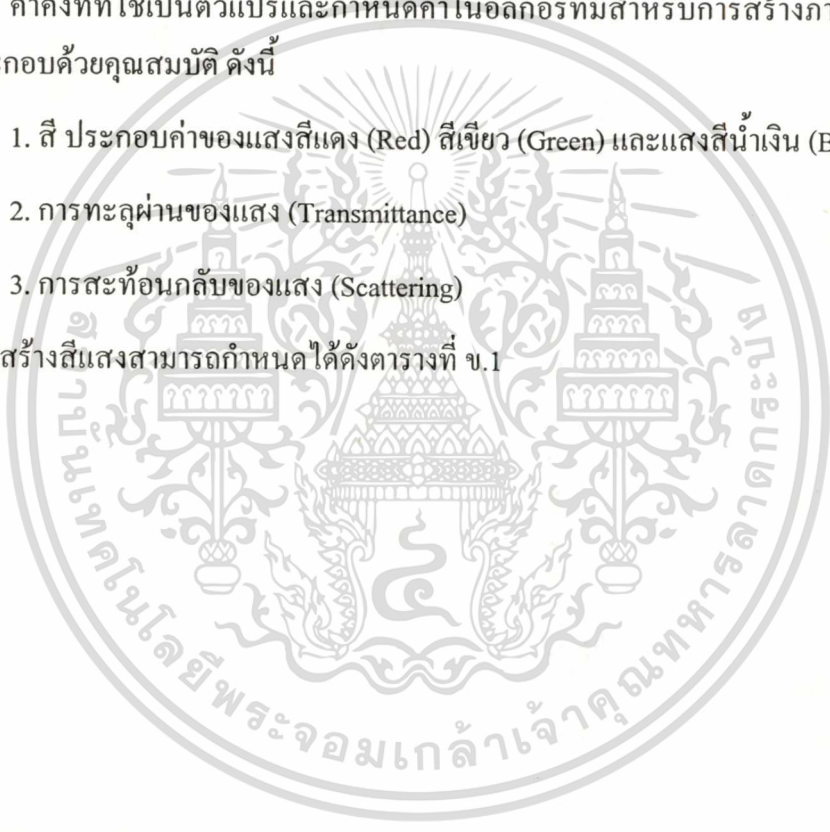
พารามิเตอร์สำหรับกำหนดคุณสมบัติสีแสง

พารามิเตอร์สำหรับการกำหนดคุณสมบัติของสีแสง

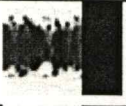











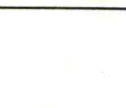
ค่าคงที่ที่ใช้เป็นตัวแปรและกำหนดค่าในอัลกอริทึมสำหรับการสร้างภาพ (Rendering) จะประกอบด้วยคุณสมบัติ ดังนี้

1. สี ประกอบค่าของแสงสีแดง (Red) สีเขียว (Green) และแสงสีน้ำเงิน (Blue)
2. การทะลุผ่านของแสง (Transmittance)
3. การสะท้อนกลับของแสง (Scattering)

ในการสร้างสีแสงสามารถกำหนดได้ดังตารางที่ ข.1



ตารางที่ ข.1 คุณสมบัติแสง

Crayon	Colour	R	G	B	T	S
	red	0.95	0.45	0.45	0.605	0.051
	orange	0.999	0.55	0.2	0.605	0.042
	yellow	0.95	0.9	0.2	0.715	0.111
	green	0.35	0.8	0.35	0.385	0.096
	blue	0.3	0.5	0.9	0.715	0.054
	purple	0.65	0.4	0.9	0.385	0.06
	brown	0.8	0.55	0.5	0.3025	0.15
	black	0.26	0.25	0.245	0.11	0.24
	grey	0.42	0.4	0.39	0.44	0.51
	white	0.8	0.8	0.79	0.55	0.33
	periwinkle	0.7	0.7	0.9	0.385	0.21
	sea green	0.6	0.9	0.65	0.33	0.15
	orchid	0.85	0.4	0.84	0.66	0.09

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

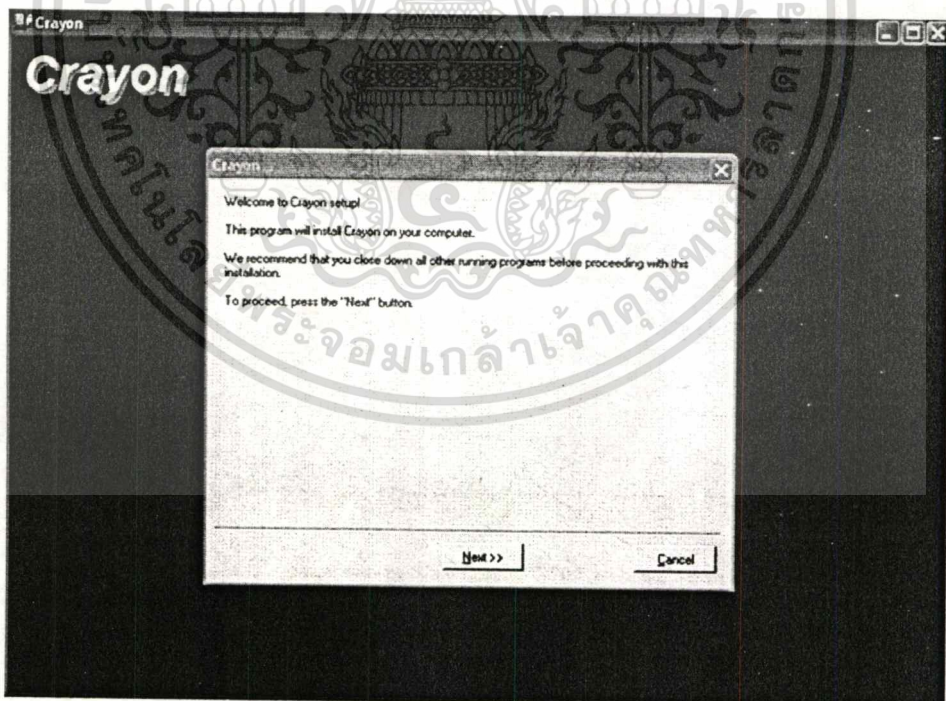
การติดตั้งโปรแกรม Crayon

ความต้องการระบบขั้นต่ำ

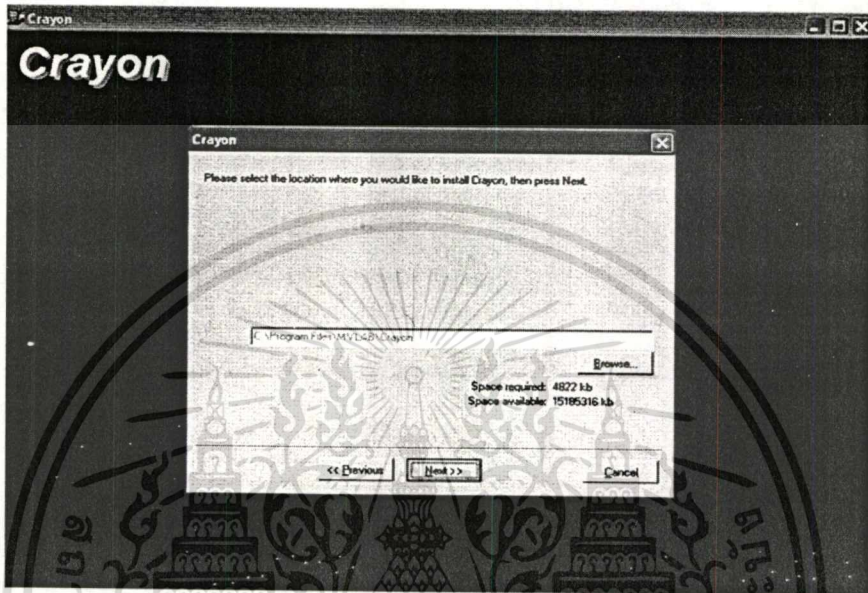
1. Pentium PC 1.6 GHz
2. Windows XP with Service Pack 2
3. อุปกรณ์ PHANToM
4. PHANToM Device Driver Version 4.2 และ HDAPI

ขั้นตอนการติดตั้งใช้งาน

1. เปิดโปรแกรม CrayonSetup.exe เลือก Next

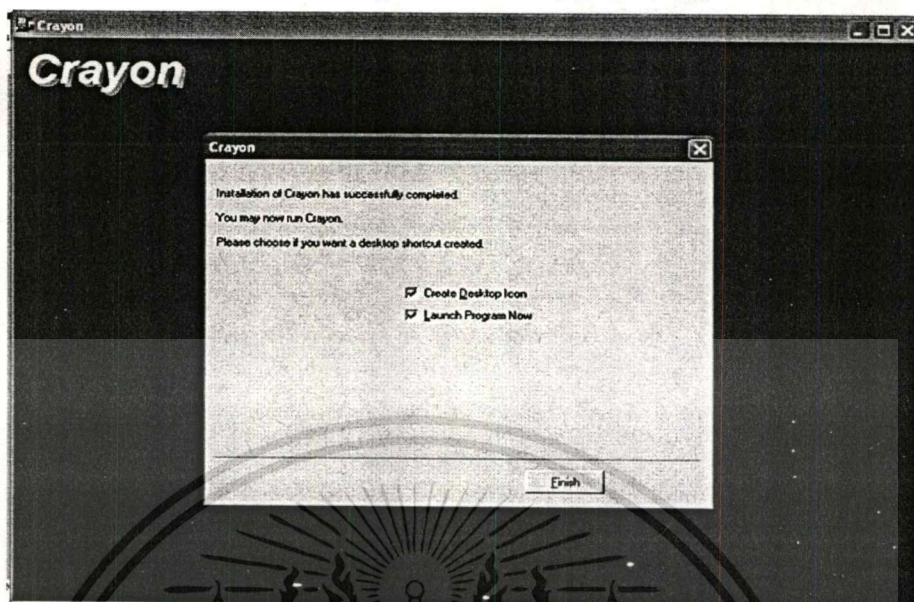


- เลือก Location ที่จะทำการติดตั้ง แล้วกด Next



- รอนจนการติดตั้งสิ้นสุด ถ้าต้องการสร้าง Desktop Icon เลือกเครื่องหมายถูกที่ ช่อง Create Desktop Icon หรือถ้าต้องการเรียกโปรแกรมทำงานเลือกเครื่องหมายถูกที่ Launch Program Now สุดท้ายกดปุ่ม Finish

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



4. การเรียกใช้งาน โปรแกรมคือ Start->Programs->Crayon หรือที่ Desktop Icon ถ้าสร้างไว้

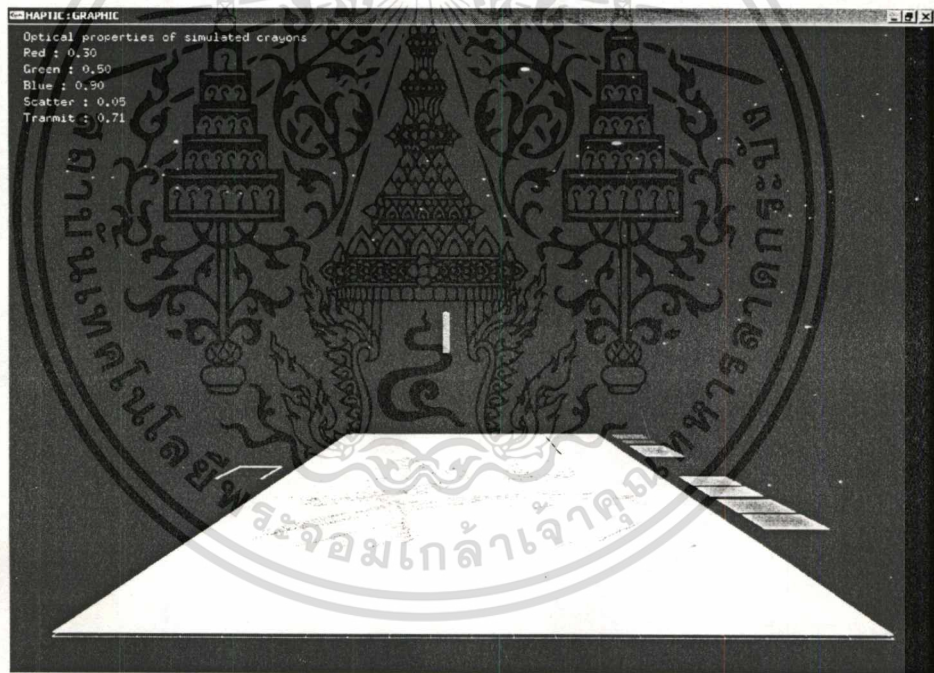
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง

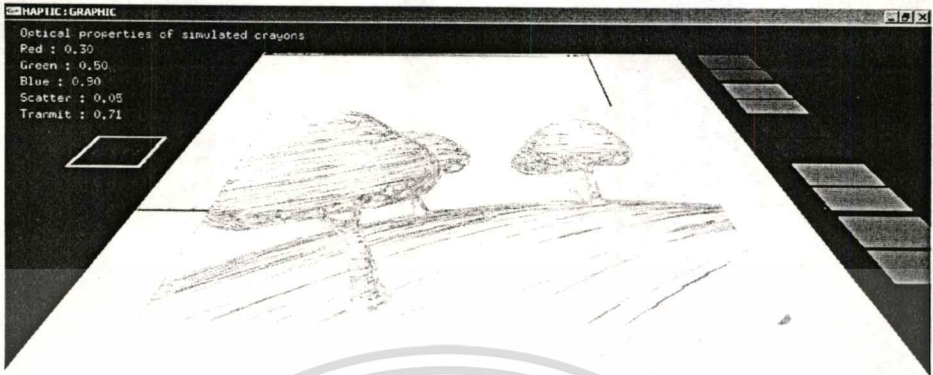
คู่มือการใช้งานโปรแกรม Crayon

การใช้งานโปรแกรม

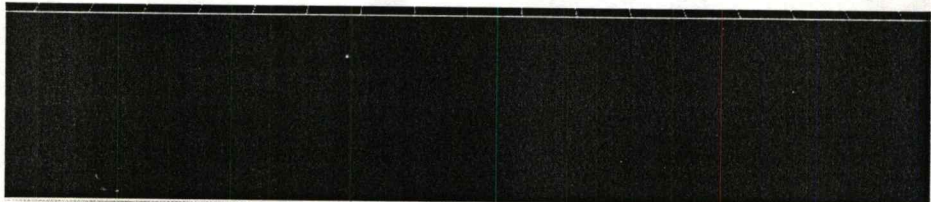
1. ให้ทำการ reset ตำแหน่งของ PHANToM ไปยังตำแหน่งที่อยู่เหนือตำแหน่ง natural เล็กน้อย
2. เปิดโปรแกรม Crayon.exe ที่ start->programs->Crayon หรือที่ Desktop Icon และทำการเปิดสวิตช์อุปกรณ์ PHANToM รวมถึงกดปุ่มการทำงานของอุปกรณ์ จะปรากฏหน้าจอ ดังภาพ



3. เมื่อเข้าสู่หน้าจอการทำงานของโปรแกรมแล้วให้กดปุ่ม Z เพื่อปรับมุมมองการวาดใหม่



4. ลากแท่งดินสอ ไปสัมผัสกับกระดาษเมื่อต้องการวาดรูป ทำการวาดภาพตามต้องการ



5. ถ้าต้องการออกจากโปรแกรม กดปุ่ม Exit หรือปิดที่หน้าต่างของ Windows

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

