การสร้างฉากเสมือนโดยการวิเคราะห์ภาพจากกล้องเดี่ยว

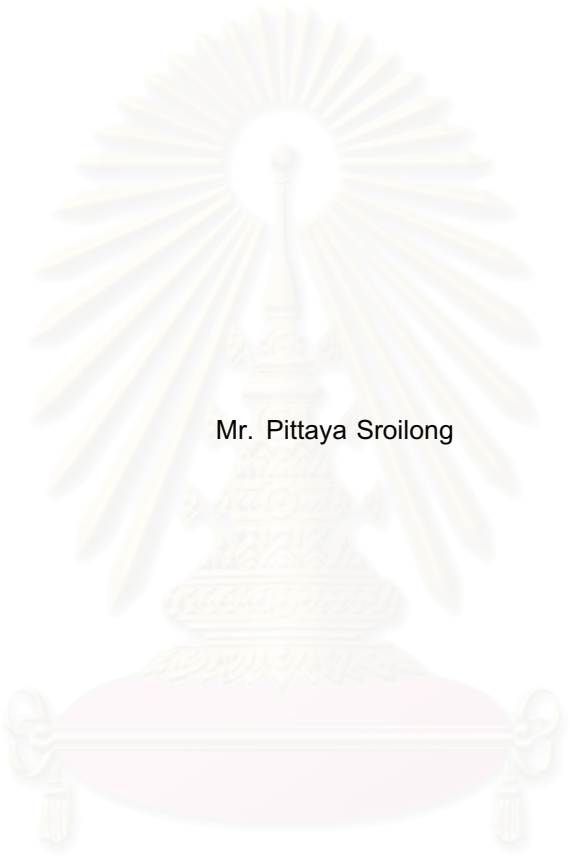
นายพิทยา สร้อยหลง

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2550
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

CONSTRUCTION OF A VIRTUAL SET BY IMAGE ANALYSIS FROM A SINGLE CAMERA

Mr. Pittaya Sroilong



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Computer Science
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2007

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การสร้างฉากเสมือนโดยการวิเคราะห์ภาพจากกล้องเดี่ยว
โดย	นายพิทยา สร้อยหลง
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์
อาจารย์ที่ปรึกษา	อาจารย์ ดร.อรรถวิทย์ สุดแสง

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร. ดิเรก ลาวัณย์ศิริ)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ)

..... อาจารย์ที่ปรึกษา
(อาจารย์ ดร.อรรถวิทย์ สุดแสง)

..... กรรมการ
(อาจารย์ ดร.เศรษฐา ปานงาม)

..... กรรมการ
(อาจารย์ ดร.วิษณุ โคตรจรัส)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

พิทยา สร้อยหลง : การสร้างฉากเสมือนโดยการวิเคราะห์ภาพจากกล้องเดี่ยว (CONSTRUCTION OF A VIRTUAL SET BY IMAGE ANALYSIS FROM A SINGLE CAMERA). อาจารย์ที่ปรึกษา : อ. ดร. อรรถวิทย์ สุดแสง, 59 หน้า.

งานวิจัยนี้มีแนวคิดเริ่มต้นที่ต้องการสร้างระบบฉากเสมือนที่ทำงานได้โดยอาศัยการวิเคราะห์ภาพที่ถ่ายได้เพียงอย่างเดียว สามารถใช้งานได้บนเครื่องคอมพิวเตอร์ส่วนบุคคลทั่วไป ไม่จำเป็นต้องซื้อฮาร์ดแวร์เพิ่มเติม และมีต้นทุนในการผลิตที่ต่ำ

ในการทำงานของระบบฉากเสมือนจะอาศัยการวิเคราะห์ภาพที่ถ่ายได้ เพื่อคำนวณตำแหน่งและแนวการเคลื่อนไหวของตัวกล้อง โดยฉากที่ใช้จะมีลวดลายเฉพาะเพื่อให้สามารถคำนวณตำแหน่งของกล้องได้ แล้วจึงนำมาสร้างโมเดลสามมิติที่เหมาะสมกับมุมมองของกล้อง เพื่อทำเป็นฉากเสมือนให้กับวัตถุที่ถ่าย ระบบที่สร้างขึ้นสามารถประมวลผลภาพวิดีโอที่ความละเอียด 352 x 240 พิกเซลได้ที่ความเร็วไม่ต่ำกว่า 25 เฟรมต่อวินาที และมีความแม่นยำในการคำนวณตำแหน่ง สามารถสร้างฉากสามมิติได้อย่างถูกต้องสมจริง

ระบบที่สร้างขึ้นสามารถนำไปประยุกต์ใช้ได้ทั้งในงานภาพยนตร์ โฆษณา หรือรายการโทรทัศน์ ที่ต้องการความรวดเร็วในการทำงานและความประหยัด

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา .....วิศวกรรมคอมพิวเตอร์ ..... ลายมือชื่อนิสิต พิทยา สร้อยหลง .....
สาขาวิชา.....วิทยาศาสตร์คอมพิวเตอร์ ..... ลายมือชื่ออาจารย์ที่ปรึกษา .....
ปีการศึกษา .....2550.....

##4771435821 : MAJOR COMPUTER SCIENCE
KEYWORDS : VIRTUAL SET/ CAMERA CALIBRATION/ COMPUTER VISION/ MATTING

PITTAYA SROILONG : CONSTRUCTION OF A VIRTUAL SET BY IMAGE ANALYSIS FROM A SINGLE CAMERA. THESIS ADVISOR : ATTAWITH SUDSANG, Ph.D., 59 pp.

This thesis proposes a construction of a virtual set using information retrieved from a single camera. The system is able to operate on a personal computer, with no special hardware required.

The system analyzes video stream from camera to obtain the camera's rotation and translation by extracting information from the pattern on the background. The 3-dimensional scene is rendered corresponding to the retrieved rotation and translation information. The system is able to operates at the resolution 352 x 240 pixel at speed 25 frames per second and the rendered scene is correctly displayed according to the real camera.

The constructed system can be adopted in various fields such as movie industry, advertisement or television works that need correctness on a low budget.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

Department .....Computer Engineering..... Student's signature .....
Field of study .....Computer Science..... Advisor's signature .....
Academic year .....2007.....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี จากความช่วยเหลือของ อ.ดร.อรรถวิทย์ สุดแสง อาจารย์ที่ปรึกษาวิทยานิพนธ์ ซึ่งอาจารย์ได้ให้ความช่วยเหลือเป็นอย่างมากทั้งในการให้คำแนะนำต่างๆ ตลอดจนการกำกับดูแลการทำวิทยานิพนธ์เป็นอย่างดี คอยถามไถ่ ติดตามความคืบหน้า และกระตุ้นให้วิทยานิพนธ์นี้สามารถสำเร็จลงได้

ขอ ขอบคุณ ผศ. ดร. วิวัฒน์ วัฒนา วุฒิ ประธานกรรมการ สอบ วิทยานิพนธ์ และ กรรมการ สอบ วิทยานิพนธ์ อ.ดร.วิษณุ โคตรจรัส และ อ.ดร.เศรษฐา ปานงาม ที่ได้กรุณาสละเวลาอันมีค่าในการช่วยตรวจสอบวิทยานิพนธ์ฉบับนี้

ขอบคุณเพื่อนฝูง รุ่นพี่ รุ่นน้อง ทุกคนในห้องปฏิบัติการที่ช่วยเหลือในด้านต่างๆ ทั้งเรื่องที่เกี่ยวข้องและไม่เกี่ยวข้องกับวิทยานิพนธ์ ขอบคุณแด้ที่ช่วยชี้แนะในหลายๆ เรื่อง ขอบคุณพีมสำหรับอัลกอริทึมและความรู้ทางคณิตศาสตร์ ขอบคุณป๊อกสำหรับคำปรึกษาในเรื่องระเบียบต่างๆ ขอบคุณกอล์ฟสำหรับความรู้ในฝั่งที่ไม่เคยสัมผัส และขอบคุณทุกคนในห้องปฏิบัติการที่มีส่วนในการผลักดันให้วิทยานิพนธ์นี้สำเร็จลงได้

สุดท้ายนี้ขอขอบคุณบิดา มารดา และญาติพี่น้องที่ให้การสนับสนุนทั้งทางด้านกำลังใจและทุนทรัพย์ตั้งแต่วัยเด็กจนกระทั่งมาถึงระดับอุดมศึกษา

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

หน้า

บทคัดย่อภาษาไทย . . . . . ง
บทคัดย่อภาษาอังกฤษ . . . . . จ
กิตติกรรมประกาศ . . . . . ฉ
สารบัญ . . . . . ช
สารบัญตาราง . . . . . ฌ
สารบัญรูปภาพ . . . . . ญ
บทที่

1 บทนำ **1**
1.1 ความเป็นมาและความสำคัญของปัญหา 1
1.2 วัตถุประสงค์ของการวิจัย 2
1.3 ขอบเขตของการวิจัย 2
1.4 วิธีดำเนินการวิจัย 3

2 เอกสารและงานวิจัยที่เกี่ยวข้อง **4**
2.1 งานวิจัยที่เกี่ยวข้อง 4
2.1.1 วิธีการตรวจจับการเปลี่ยนแปลงตำแหน่งและแนวการวางตัวของกล้อง 4
2.1.2 การลบภาพฉากหลัง 5
2.1.3 การสร้างฉากสามมิติ 6
2.2 ทฤษฎีที่เกี่ยวข้อง 7
2.2.1 ทฤษฎีด้านคอมพิวเตอร์กราฟิก 7
2.2.1.1 สีปฐมภูมิ 7
2.2.1.2 ระบบสี 8
2.2.1.3 การจัดเก็บภาพกราฟิกแบบสองมิติ 10
2.2.1.4 การจัดเก็บภาพกราฟิกแบบสามมิติ 11
2.2.1.5 ทฤษฎีเกี่ยวกับเรขาคณิตในสามมิติ 14
2.2.1.6 กล้องในโลกสามมิติ 15
2.2.1.7 โมเดลกล้องแบบรูเข็ม 16
2.2.1.8 ภาพพื้นผิว 17
2.2.1.9 กลไกการสร้างภาพสามมิติ 18
2.2.2 ทฤษฎีที่เกี่ยวข้องกับกระบวนการทางภาพยนตร์ 19
2.2.2.1 การแยกภาพวัตถุออกจากฉากหลัง 19
2.2.2.2 การกำจัดการเปื้อนของสี 23
2.2.2.3 การรวมภาพตัวแสดงและภาพฉาก 23

3 การออกแบบและวิธีในการพัฒนา **25**
3.1 การตรวจสอบหาตำแหน่งของกล้อง 25

หน้า

3.2 การลบภาพฉากหลัง 32
3.3 การสร้างฉากสามมิติ 33
3.4 การรวมภาพตัวแสดงและภาพฉาก 34
3.5 การออกแบบและพัฒนาระบบ 35

4 การทดลองและผลการทดลอง 36
4.1 ระบบที่ใช้ในการทดลอง 36
4.2 ผลการตรวจสอบหาตำแหน่งของกล้อง 36
4.2.1 การทดสอบกับภาพเคลื่อนไหวสามมิติที่สร้างขึ้นจากคอมพิวเตอร์ 36
4.2.2 การทดสอบกับภาพเคลื่อนไหวที่ได้จากการถ่ายทำด้วยกล้องจริง 37
4.3 ผลการลบภาพฉากหลังออกจากภาพที่ถ่ายได้ 38
4.4 ผลการสร้างฉากสามมิติ 41
4.5 ผลการรวมภาพที่ถ่ายได้เข้ากับฉากสามมิติที่สร้างขึ้น 42
4.6 ผลการทดลองโดยสรุป 43

5 สรุปการวิจัย 44
5.1 ข้อเสนอแนะ 44
5.2 งานวิจัยในอนาคต 45
รายการอ้างอิง 46
ประวัติผู้เขียนวิทยานิพนธ์ 48

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญตาราง

ตารางที่ หน้า
4.1 ความเร็วในการสร้างฉากสามมิติที่ความละเอียดต่างๆ 42
4.2 สรุปเวลาที่ใช้ในการทำงานขั้นตอนต่างๆ 43

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญรูปภาพ

รูปที่ หน้า

1.1 ตัวอย่างการใช้งานฉากสีฟ้า 1
2.1 แสดงการแบ่งพื้นที่ และการสร้างรูปทรงแปดเหลี่ยมตามวิธีของ [6] 5
2.2 แสดงการแยกภาพวัตถุออกจากฉากหลังตามวิธีของ [7] 6
2.3 แสดงจุดบนปริภูมิสีแบบ YCbCr ในการคำนวณตามวิธีของ [8] 6
2.4 แสดงการตอบสนองของประสาทตามนุษย์ที่ความถี่ต่างๆ 8
2.5 แสดงระบบสี RGB บนแกนสามมิติ 8
2.6 กรวยแสดงระบบสีแบบ HSV 9
2.7 เปรียบเทียบภาพกราฟิกแบบเวคเตอร์และราสเตอร์ 10
2.8 ตัวอย่างภาพกราฟิกสองมิติ 11
2.9 ตัวอย่างภาพกราฟิกสามมิติ 12
2.10 ภาพเปรียบเทียบระบบแกนมือซ้าย (ซ้าย) และระบบแกนมือขวา (ขวา) 12
2.11 แสดงพิกัด (x, y, z) ของลูกบาศก์ขนาด 1 หน่วย 12
2.12 แสดง จุด เส้น และรูปหลายเหลี่ยม 13
2.13 แสดงการสร้างโมเดลสามมิติขึ้นจากรูปหลายเหลี่ยมจำนวนหลายๆ ชิ้น 13
2.14 ภาพแสดงความละเอียดของโมเดลที่มีจำนวนรูปหลายเหลี่ยมต่างกัน 13
2.15 การย้ายตำแหน่งในพิกัดสามมิติ 14
2.16 การหมุนในพิกัดสามมิติ 15
2.17 การเปลี่ยนขนาดในพิกัดสามมิติ 15
2.18 แสดงส่วนต่างๆ ของกล้องรูเข็ม 16
2.19 รูปแสดงความสัมพันธ์ระหว่างจุดในโลกสามมิติและจุดที่ปรากฏบนระนาบรับภาพ 16
2.20 แสดงการใช้ภาพพื้นผิวในการสร้างความสมจริงให้กับโมเดลสามมิติ 18
2.21 เปรียบเทียบภาพที่ได้จากการประมวลผลแบบทันกาล (ซ้าย) และแบบเป็นชุด (ขวา) 19
2.22 แสดงระดับการทำงานของชุดคำสั่ง 19
2.23 ภาพต้นฉบับ (ซ้าย) และภาพโปร่งใส (ขวา) 19
2.24 ภาพต้นฉบับก่อนผ่านกระบวนการแยกภาพ 20
2.25 การแยกภาพฉากหลังออกโดยใช้ขอบเขตความสว่างเดียว 20
2.26 การแยกภาพฉากหลังออกโดยใช้ขอบเขตความสว่างมากกว่าหนึ่งจุด 21
2.27 แสดงระยะห่างระหว่างจุดสองจุดในปริภูมิสีแบบ RGB 21
2.28 ตัวอย่างภาพที่แยกฉากหลังออกมาโดยใช้การพิจารณาความต่างของสี 22
2.29 ภาพที่ยังไม่ผ่านการกำจัดการเปื้อน 23
2.30 ภาพที่ผ่านการกำจัดการเปื้อนแล้ว 24
3.1 แผนภาพแสดงการทำงานโดยรวมของระบบ 26
3.2 การคำนวณหาพารามิเตอร์ภายนอกโดยกำหนดจุดอ้างอิง 4 จุดบนฉากหลัง 27
3.3 การใช้ฉากหลังที่มีลวดลายเป็นตารางสี่เหลี่ยม 28
3.4 แสดงลายตารางที่ใช้เป็นฉากหลัง 28

รูปที่ หน้า

3.5 แสดงปัญหาของการใช้ลายตารางเพียงอย่างเดียว 29
3.6 แสดงการใช้จุดเพื่อสร้างความแตกต่างระหว่างตารางสี่เหลี่ยม 29
3.7 ภาพตารางต้นฉบับ 30
3.8 การใช้ค่าขีดแบ่งเดียวทั้งภาพ 30
3.9 การใช้ค่าขีดแบ่งแบบปรับตัวได้ 30
3.10 ภาพที่ใช้ขีดแบ่งแบบปรับตัวได้ และผ่านการเพิ่มขนาดจุด 31
3.11 แสดงการหาอัตราส่วนไขว้ 31
3.12 ตัวอย่างการหาอัตราส่วนไขว้จากรูปที่ได้จากกล้อง 31
3.13 ความสัมพันธ์ระหว่างความสว่าง (แกนนอน) และความแตกต่างของสี (แกนตั้ง) 33
3.14 ภาพที่ได้ในขั้นตอนแรก (ซ้าย) และภาพหลังจากการปรับอัตราส่วน (ขวา) 34
3.15 คลาสไดอะแกรมของระบบฉากเสมือน 35
4.1 การสร้างฉากสามมิติโดยใช้โปรแกรม Blender 37
4.2 ภาพต้นฉบับ 37
4.3 ภาพผลลัพท์ที่มีการสร้างฉากหลังขึ้นตามข้อมูลพารามิเตอร์ของกล้อง 38
4.4 ภาพเปรียบเทียบฉากจริงและฉากที่สร้างขึ้น 38
4.5 ภาพต้นฉบับ 39
4.6 ภาพผลลัพท์ที่มีการสร้างฉากหลังขึ้นตามข้อมูลพารามิเตอร์ของกล้อง 39
4.7 ภาพเปรียบเทียบฉากจริงและฉากที่สร้างขึ้น 39
4.8 กราฟทั่วไปของฟังก์ชันซิกมอยด์ 40
4.9 กราฟของฟังก์ชันสำหรับคำนวณค่าความโปร่งใส 40
4.10 การใช้ค่า amount, fuzziness ที่ 2.0, 0.5 (ซ้าย) และ 5.0, 3.0 (ขวา) 41
4.11 การปิดและเปิดการทำงานส่วนปรับความสมดุลสีขาวของกล้อง 41
4.12 ภาพฉาก Return to the Castle จากเกม Quake3 42
4.13 เปรียบเทียบภาพก่อนและหลังการกำจัดการเปื้อนของสี 43

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เทคโนโลยีในการถ่ายทำภาพเคลื่อนไหว เช่น ภาพยนตร์ หรือรายการโทรทัศน์ ตั้งแต่อดีตจนถึงปัจจุบันมีการพัฒนาอย่างต่อเนื่อง จากการสร้างฉากจำลองในสมัยก่อน หรือการถ่ายทำในสถานที่จริง ในปัจจุบันที่คอมพิวเตอร์มีสมรรถนะและความสามารถมากขึ้น งานเหล่านี้สามารถถูกจำลองได้ด้วยการสร้างฉากขึ้นในคอมพิวเตอร์ แล้วนำไปรวมกับภาพถ่ายเพื่อให้ได้ภาพเสมือนว่าถ่ายทำในสถานที่จริง

เทคโนโลยีของฉากเสมือน (Virtual Set) เป็นเทคนิคการประมวลผลภาพที่มีการพัฒนามาจากต้นแบบการใช้งานฉากสีฟ้า (Bluescreen) ในเบื้องต้นคือการถ่ายภาพของวัตถุบนฉากหลังสีเดียวซึ่งโดยปกตินิยมใช้ฉากสีฟ้า แล้วนำภาพที่ถ่ายได้มาประมวลผล เพื่อตัดส่วนที่เป็นฉากหลังสีฟ้าออก และนำไปซ้อนทับกับภาพฉากหลังที่เตรียมไว้ล่วงหน้า เพื่อให้เกิดเป็นภาพที่ดูเหมือนกับวัตถุอยู่ในสถานที่เดียวกับภาพฉากที่เตรียมไว้แล้ว ตัวอย่างการใช้งานฉากสีฟ้าที่เห็นได้ในชีวิตประจำวันคือ รายการพยากรณ์อากาศ ซึ่งมีผู้ทำหน้าที่พยากรณ์อากาศยืนอยู่ด้านหน้าของแผนที่แสดงสภาพอากาศที่เปลี่ยนแปลงไปได้ตามที่กำหนด วิธีการถ่ายทำคือ ถ่ายภาพของผู้พยากรณ์อากาศที่ยืนอยู่ด้านหน้าของฉากหลัง แล้วจึงลบส่วนที่เป็นฉากหลังออกเพื่อนำไปซ้อนทับลงบนภาพแผนที่ซึ่งเตรียมไว้แล้ว ดังตัวอย่างในรูปที่ 1.1 [1]

รูปที่ 1.1: ตัวอย่างการใช้งานฉากสีฟ้า

เทคนิคการใช้งานฉากสีฟ้านี้ได้รับความนิยมอย่างแพร่หลายในวงการโทรทัศน์และในอุตสาหกรรมภาพยนตร์ เนื่องจากสามารถลดต้นทุนในการสร้างฉาก หรือการหาสถานที่ถ่ายทำที่ตรงตามความต้องการและสามารถสร้างฉากที่ไม่สามารถสร้างได้ในทางปฏิบัติ เช่นในภาพยนตร์เรื่อง The Matrix [2], Sin City หรือ 300

เทคนิคการใช้งานฉากสีฟ้านั้นยังมีข้อจำกัดอยู่บางประการ คือภาพฉากหลังที่นำมาซ้อนจะต้องมีมุมกล้องที่สอดคล้องกับภาพของตัวแสดง ดังนั้นจะต้องเตรียมภาพเหล่านี้ไว้ก่อนที่จะนำมาใช้งาน ทำให้รูปแบบของฉากมีได้จำกัดเท่าที่ได้มีการเตรียมไว้ก่อนเท่านั้น และถ้าฉากหลังเหล่านี้เป็นภาพนิ่ง จะไม่

สามารถเคลื่อนไหวได้ตามมุมกล้องที่เปลี่ยนแปลงไป หรือในกรณีที่มีการเตรียมฉากหลังแบบที่เป็นภาพเคลื่อนไหวไว้แล้ว การถ่ายภาพวัตถุเบื้องหน้าก็จำเป็นจะต้องทำให้สอดคล้องกับจังหวะการเคลื่อนไหวของฉากหลัง เช่นการใช้กล้องที่ยึดติดกับเครนที่ควบคุมการเคลื่อนไหวด้วยคอมพิวเตอร์ มิฉะนั้นแล้ววัตถุเบื้องหน้าและฉากหลังจะเคลื่อนที่ไม่ประสานกัน ทำให้ขาดความสมจริง

จากข้อจำกัดนี้ ทำให้เกิดแนวคิดในการสร้างฉากเสมือนแบบสามมิติ ที่สามารถเปลี่ยนแปลงมุมกล้อง หรือการเคลื่อนไหวของกล้องได้อย่างอิสระ และนำข้อมูลการเคลื่อนไหวและมุมมองของกล้องเหล่านี้ มาคำนวณสร้างฉากหลังแบบสามมิติที่สอดคล้องกับกล้องได้ ทำให้ผลลัพธ์ที่ได้ดูสมจริงยิ่งขึ้น แต่อย่างไรก็ตาม การใช้งานฉากเสมือนแบบสามมิติดังกล่าว ยังมีข้อจำกัดอยู่ด้วยค่าใช้จ่ายที่สูง จำเป็นต้องพึ่งพาอุปกรณ์เฉพาะทาง เช่น เครื่องมือสำหรับติดตามการเคลื่อนไหวของกล้อง ฉากแบบพิเศษ หรืออุปกรณ์ประมวลผลภาพที่เป็นฮาร์ดแวร์เฉพาะ ขึ้นอยู่กับผู้ผลิตแต่ละราย

จากปัญหาดังกล่าว ประกอบกับในปัจจุบัน คอมพิวเตอร์ส่วนบุคคลมีสมรรถนะที่สูงขึ้น สามารถประมวลผลและทำงานที่ซับซ้อนได้ในเวลาอันรวดเร็ว รวมทั้งยังสร้างภาพสามมิติความละเอียดสูงได้มีคุณภาพดีกว่าในสมัยก่อนมาก ทำให้เกิดแนวคิดของงานวิจัยนี้ ในการสร้างฉากเสมือนแบบสามมิติขึ้นมาจากเครื่องคอมพิวเตอร์ส่วนบุคคลทั่วไป โดยเลือกใช้อุปกรณ์ต่อพ่วงพื้นฐานที่สามารถใช้งานกับเครื่องคอมพิวเตอร์ส่วนบุคคลได้ เพื่อมีเป้าหมายในการสร้างระบบฉากเสมือนแบบสามมิติที่มีคุณภาพดี มีค่าใช้จ่ายในการสร้างต่ำ และสามารถนำไปใช้งานในวงการอุตสาหกรรมได้จริง

นอกจากเหตุผลที่กล่าวมาแล้ว ในโลกยุคปัจจุบัน สื่อพลเมือง (Citizen media) มีการเติบโตขึ้นอย่างสูง สาเหตุเนื่องจากการเจริญเติบโตของอินเตอร์เนทและค่าใช้จ่ายในการผลิตผลงานที่ต่ำลง ทำให้ผู้ใช้ทั่วไปสามารถสร้างผลงานภาพเคลื่อนไหวได้โดยมีค่าใช้จ่ายต่ำลงมาก เมื่อเทียบกับในสมัยก่อน โดยที่คุณภาพของผลงานในแง่ของความละเอียดนั้น ยังไม่สูงเท่างานที่ทำเพื่อเชิงพาณิชย์ ระบบฉากเสมือนที่สร้างขึ้นนี้จึงเหมาะสมกับการนำไปใช้งานในระดับของผู้ใช้ทั่วไปที่มีความต้องการเพิ่มความน่าสนใจให้กับผลงานภาพเคลื่อนไหวของตนเองด้วย

1.2 วัตถุประสงค์ของการวิจัย

งานวิจัยนี้มีวัตถุประสงค์หลักคือการสร้างระบบฉากเสมือนแบบสามมิติที่สามารถแสดงฉากผลลัพธ์เปลี่ยนแปลงได้ตามการเคลื่อนที่ของกล้อง โดยส่วนประกอบย่อยต่างๆ ของระบบฉากเสมือนสามมิตินี้ เช่น ระบบสร้างภาพสามมิติ ระบบซ้อนภาพและตัดภาพฉากหลัง หรือระบบสำหรับหาเวคเตอร์การย้ายตำแหน่ง (Translation vector) และเมตริกซ์การหมุน (Rotation matrix) สามารถนำไปใช้ในงานอื่นที่เกี่ยวข้องกันได้ ระบบฉากเสมือนที่สร้างขึ้นนี้สามารถทำงานแบบเวลาจริงได้ที่ความละเอียดปานกลาง เทียบเท่าคุณภาพของวิดีโอแบบ MPEG-1

1.3 ขอบเขตของการวิจัย

ในการทำงานวิจัยนี้ ได้กำหนดของเขตของงานวิจัยไว้เป็นข้อๆ ดังนี้

- คุณภาพของภาพผลลัพธ์ - ระบบที่สร้างขึ้นจะสามารถใช้งานได้ที่ความละเอียดตามมาตรฐาน MPEG-1 คือที่ 352x240 จุด จำนวน 30เฟรมต่อวินาที

- การตรวจจับการเคลื่อนไหวของกล้อง - ขอบเขตการหาเวคเตอร์การย้ายตำแหน่งและเมตริกซ์การหมุน จะจำกัดอยู่เฉพาะบนแกน x, y, z และไม่มีการซูมภาพเข้าออก
- ระยะความคมชัดของภาพ - การถ่ายภาพโดยใช้รูรับแสงที่มีขนาดกว้าง จะทำให้ภาพบางส่วนมีความเลือนรางและไม่ชัดเจน ซึ่งภาพที่ใช้ในงานวิจัยนี้อย่างน้อยต้องมีฉากหลังที่คมชัด จึงจะสามารถตรวจสอบได้
- ความละเอียดของฉากสามมิติ - ฉากสามมิติที่นำมาซ้อน จะถูกสร้างขึ้นแบบทันกาล ซึ่งในการสร้างฉากสามมิติด้วยวิธีนี้ต้องการความสามารถในการประมวลผลสูง ทำให้ระบบยังมีข้อจำกัดในเรื่องของความซับซ้อนของรูปหลายเหลี่ยมในฉากอยู่ ไม่สามารถสร้างให้ละเอียดสมจริงได้
- ลักษณะของแสงเงาในฉาก - สำหรับในงานวิจัยนี้จะไม่สนใจเรื่องการเรนเดอร์แสงเงาที่เกิดจากแหล่งกำเนิดแสงที่เคลื่อนไหวได้ เนื่องจากไม่สามารถหาอุปกรณ์สำหรับตรวจจับตำแหน่งทิศทางของแสงได้ และเพื่อลดภาระในการประมวลผลของระบบทั้งส่วนตรวจจับภาพและส่วนสร้างภาพสามมิติ

1.4 วิธีดำเนินการวิจัย

ขั้นตอนการวิจัยที่ได้วางแผนไว้มีดังนี้

- แบ่งงานวิจัยออกเป็นส่วนย่อยๆ แล้วศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้องกับวิทยานิพนธ์ในแต่ละส่วนย่อย
- เลือกทฤษฎีหรืออัลกอริทึมที่เหมาะสม นำมาใช้ทดสอบสร้างฉากเสมือนต้นแบบ เพื่อใช้สำหรับทดสอบความถูกต้องของแนวคิดว่าสามารถนำมาใช้ปฏิบัติได้จริง โดยในขั้นต้น จะใช้อุปกรณ์พื้นฐานที่ไม่จำเป็นต้องลงทุนสูง เช่น กล้องถ่ายภาพเคลื่อนไหวสำหรับเวบ (Webcam) ความละเอียดต่ำ, สร้างฉากหลังสีเขียวโดยใช้แผ่นพลาสติกประกอบกับกระดาษสีที่ตัดเป็นรูปสี่เหลี่ยม ติดเข้าด้วยกันเป็นลวดลายตารางตามที่ต้องการ
- ทดสอบกับอัลกอริทึมแบบต่างๆ รวมทั้งปรับเปลี่ยนรูปแบบของฉาก เช่น ขนาดของจุด หรือขนาดของตาราง เพื่อหาวิธีการที่มีประสิทธิภาพดี
- สร้างฉากเสมือนสำหรับใช้ทดสอบจริง โดยใช้วัสดุอุปกรณ์ที่มีคุณภาพสูงขึ้น เช่น ใช้กล้องที่มีความละเอียดสูงขึ้น ฉากหลังขนาดใหญ่ขึ้น และทดสอบในสภาพแสงที่มีการควบคุมที่ดี
- ทดสอบระบบกับฉากที่สร้างขึ้น และวัดประสิทธิภาพในการทำงานของระบบในด้านต่างๆ เช่น ความถูกต้องของการตรวจจับการเคลื่อนไหว, ความเร็วในการทำงาน
- สรุปผลการทำงานของระบบ

บทที่ 2

เอกสารและงานวิจัยที่เกี่ยวข้อง

2.1 งานวิจัยที่เกี่ยวข้อง

การทำงานของระบบฉากเสมือนนั้น ประกอบด้วยขั้นตอนที่จำเป็น ได้แก่ การหาแนวการวางตัวและตำแหน่งของกล้อง, การลบฉากหลังส่วนที่ไม่ต้องการออก, การนำภาพที่ถ่ายได้มารวมเข้ากับภาพของฉากหลังที่เตรียมไว้ โดยระบบฉากเสมือนที่สร้างขึ้นในงานวิจัยนี้ประกอบด้วยส่วนย่อยๆ หลายส่วนซึ่งแต่ละส่วนมีผู้เสนองานวิจัยที่เกี่ยวข้องไว้ดังนี้

2.1.1 วิธีการตรวจจับการเปลี่ยนแปลงตำแหน่งและแนวการวางตัวของกล้อง

การทำงานของระบบฉากเสมือนในงานวิจัยนี้ ผู้ที่ใช้งานระบบสามารถถ่ายภาพโดยเปลี่ยนแปลงมุมกล้องได้อย่างอิสระ จากนั้นระบบจะสร้างฉากหลังที่สอดคล้องกับมุมมองของการถ่ายภาพให้เอง ซึ่งการจะทำเช่นนั้นได้ จำเป็นต้องรู้ลักษณะการวางตัวของกล้องรวมทั้งตำแหน่งของกล้องเทียบกับฉากหลังด้วย โดยทั้งสองส่วนนี้สามารถอธิบายในเชิงคณิตศาสตร์ได้ คือ

- เมตริกซ์การหมุน (Rotation matrix) เป็นเมตริกซ์ขนาด 3x3 ที่บอกแนวการวางตัวของกล้อง ในแนวแกน x, y, z ของโลกจริง
- เวคเตอร์การย้ายตำแหน่ง (Translation vector) เป็นเวคเตอร์สามมิติ บอกตำแหน่งของกล้องในแนวแกน x, y, z ของโลกจริง

แนวคิดในการหาเมตริกซ์การหมุนและเวคเตอร์การเลื่อนตำแหน่ง สามารถแบ่งออกเป็นกลุ่มตามวิธีการที่ใช้ได้ 2 กลุ่มใหญ่ๆ คือ การวัดแนวการวางตัวและตำแหน่งของกล้องโดยวัดที่ตัวกล้องโดยตรงกับการหาค่าจากการประมวลผลภาพที่กล้องถ่ายมาได้ ในวิธีแรก สิ่งที่จำเป็นต้องมีคืออุปกรณ์ที่สามารถใช้ตรวจสอบตำแหน่งของกล้องทางกายภาพได้ (tracker) อาจเป็นอุปกรณ์ประเภทตัวรับรู้ (sensor) แบบแสง หรือตัวรับรู้แบบคลื่นวิทยุ หรือเป็นอุปกรณ์ที่ติดตั้งเข้ากับชุดกล้องที่ใช้ในการถ่ายทำเพื่อตรวจสอบการเคลื่อนไหวทางกายภาพโดยตรง [3] โดยวิธีการตรวจสอบแบบนี้ จะเป็นการวัดระยะการเคลื่อนที่ตามแนวแกนต่างๆ รวมทั้งมุมในการหมุนและการวางตัวของกล้อง จากตัวกล้องจริงๆ ข้อดีของวิธีนี้คือ ได้ข้อมูลที่ต้องการโดยตรง ไม่จำเป็นต้องผ่านการประมวลผลอย่างอื่นอีก ซึ่งความแม่นยำเที่ยงตรงก็จะขึ้นกับประสิทธิภาพของอุปกรณ์ที่ใช้ ข้อเสียคือ จำเป็นต้องใช้อุปกรณ์เฉพาะทางในการวัด ซึ่งอุปกรณ์ที่มีความแม่นยำสูงก็จะมีราคาสูงตามไปด้วย และนอกจากนี้ระบบจะได้รับข้อมูลของภาพเคลื่อนไหวที่ถ่ายได้ กับข้อมูลเกี่ยวกับตำแหน่งของกล้อง แยกกันเป็นคนละส่วนที่ไม่เกี่ยวข้องกัน เนื่องจากได้มาจากอุปกรณ์คนละชิ้นกัน (ภาพเคลื่อนไหวได้จากกล้อง, ข้อมูลตำแหน่งของกล้องได้จากอุปกรณ์ตรวจจับ) ดังนั้นก่อนจะนำไปใช้งานจะต้องมีการประสานเวลา (synchronize) ของข้อมูลทั้งสองส่วนให้สอดคล้องกันด้วย

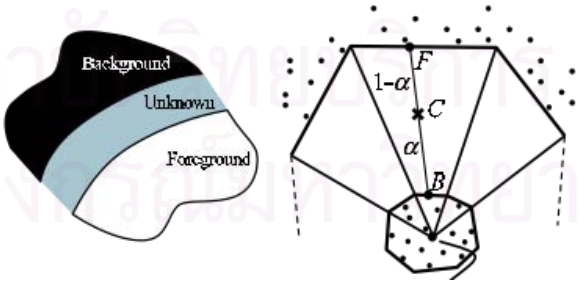
อีกวิธีหนึ่งในการหาตำแหน่งและแนวการวางตัวของกล้อง ทำได้โดยประมวลผลจากภาพที่กล้องถ่ายได้ นำมาคำนวณย้อนกลับโดยพิจารณาจากภาพที่มีการเปลี่ยนแปลงไปในแต่ละเฟรม [4] วิธีนี้มีข้อดี

คือ ไม่มีปัญหาเรื่องความไม่ตรงกันของภาพเคลื่อนไหวกับข้อมูลของกล้อง เพราะว่าภาพเคลื่อนไหว 1 ภาพ เมื่อนำมาประมวลผล จะได้ข้อมูลเมตริกซ์การหมุนและเวคเตอร์การย้ายตำแหน่งของภาพนั้นๆ โดยอัตโนมัติ หากภาพเคลื่อนไหวที่ถ่ายมามีอยู่ 25 เฟรมใน 1 วินาที ข้อมูลของกล้องที่คำนวณได้ก็จะตรงกับภาพเคลื่อนไหวทั้ง 25 เฟรมนั้นๆ ไม่มีความจำเป็นต้องปรับแต่งให้ตรงกันอย่างวิธีการตรวจจับตำแหน่งทางกายภาพ ส่วนข้อเสียของวิธีนี้คือ เราไม่สามารถระบุข้อมูลของกล้องจากภาพที่ถ่ายได้โดยตรง จำเป็นจะต้องมีจุดอ้างอิงในภาพที่ถ่ายได้ เพื่อเป็นข้อมูลในการหาเมตริกซ์การหมุนและเวคเตอร์การย้ายตำแหน่งและวิธีนี้จะต้องใช้ทรัพยากรเพิ่มเติมในการคำนวณและประมวลผลภาพเพื่อหาข้อมูลเหล่านี้ด้วย

2.1.2 การลบภาพฉากหลัง

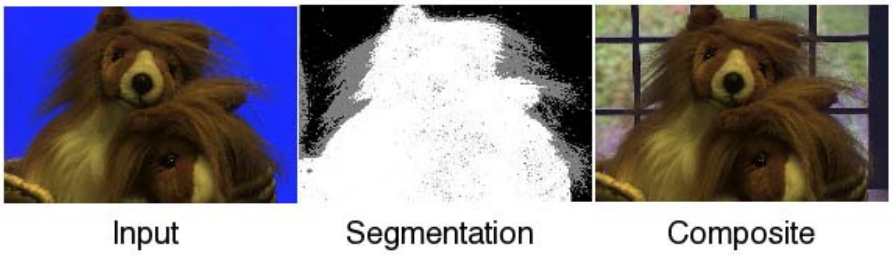
แนวคิดในการเปลี่ยนแปลงแก้ไขภาพเคลื่อนไหวโดยใช้คอมพิวเตอร์ได้รับการเสนอโดย Pedro Vlahos [5] โดยมีแนวคิดการจัดเก็บภาพเคลื่อนไหวที่ถ่ายได้จากกล้องลงในหน่วยความจำคอมพิวเตอร์และสามารถแก้ไขค่าของแต่ละจุดเพื่อเปลี่ยนแปลงรูปภาพต้นฉบับ ซึ่งถูกนำไปใช้กันอย่างแพร่หลายในวงการภาพยนตร์

ขั้นตอนจำเป็นขั้นตอนหนึ่งในการแก้ไขภาพเคลื่อนไหวคือ การแยกภาพของวัตถุเบื้องหน้าที่สนใจออกจากฉากหลัง และนำภาพที่ได้รวมเข้ากับภาพที่เตรียมไว้ก่อนหน้า ซึ่งปัญหาหลักสองประการที่ต้องให้ความสนใจในการลบภาพฉากหลัง คือ การลบรอยหยักบริเวณขอบเพื่อให้กลืนเข้ากับภาพที่นำมารวมเข้าด้วยกัน และปัญหาการจัดการกับสีที่มีความใกล้เคียงกัน ซึ่งมีผู้เสนอวิธีแก้ไขปัญหาไว้หลากหลายวิธี เช่น การคำนวณจากรูปทรงแปดเหลี่ยมในปริภูมิสีแบบ RGB [6] ใช้การแบ่งพื้นที่ของภาพออกเป็น 3 บริเวณ ได้แก่ส่วนที่เป็นฉากหลัง วัตถุเบื้องหน้า และบริเวณที่ยังไม่รู้ จากนั้นจึงสร้างรูปทรงแปดเหลี่ยมสองอันซ้อนกันขึ้นในปริภูมิสีแบบ RGB โดยรูปทรงแปดเหลี่ยมอันแรก เป็นรูปทรงแปดเหลี่ยมที่เล็กที่สุด ที่สามารถบรรจุจุดสีทั้งหมดที่สุ่มมาจากบริเวณที่เป็นฉากหลัง รูปทรงแปดเหลี่ยมอีกอันหนึ่ง สร้างให้ครอบรูปทรงแปดเหลี่ยมอนแรกและมีขนาดใหญ่ที่สุดที่ไม่มีจุดสีที่สุ่มมาได้จากบริเวณที่เป็นวัตถุเบื้องหน้า บริเวณที่เป็นช่องว่างระหว่างรูปทรงแปดเหลี่ยมอันนอกและอันใน จะใช้ในการคำนวณการลบฉากหลังออกในภาพบริเวณที่ยังไม่รู้ว่าเป็นฉากหลังหรือวัตถุเบื้องหน้า ดังรูป 2.1



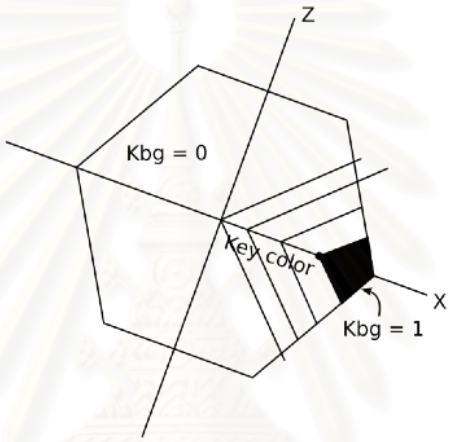
รูปที่ 2.1: แสดงการแบ่งพื้นที่ และการสร้างรูปทรงแปดเหลี่ยมตามวิธีของ [6]

นอกจากวิธีข้างต้น [7] เสนอการคำนวณโดยอาศัยข้อมูลภาพจากจุดที่อยู่ใกล้เคียง แล้ววิเคราะห์โดยใช้ความน่าจะเป็นแบบของเบย์ (Bayesian matting) ทำให้สามารถแยกภาพของวัตถุออกจากฉากหลังได้แม่นยำดังในรูป 2.2 และไม่จำเป็นต้องใช้ฉากหลังเป็นสีเดียว



รูปที่ 2.2: แสดงการแยกภาพวัตถุออกจากฉากหลังตามวิธีของ [7]

Ashikhmin [8] ใช้การวิเคราะห์สีหลักบนปริภูมิสีแบบ YCbCr โดยคำนวณสัดส่วนระยะห่างจากจุดสีแต่ละจุดไปถึงจุดสีที่กำหนดให้เป็นสีหลักทีละจุดอย่างอิสระต่อกัน ดังรูป 2.3



รูปที่ 2.3: แสดงจุดบนปริภูมิสีแบบ YCbCr ในการคำนวณตามวิธีของ [8]

จากงานวิจัยของ C.H. Kim [9] ชี้ให้เห็นว่าค่าความสว่าง ณ จุดใดๆ ในปริภูมิสีแบบ HSI มีผลต่อการแยกส่วนสีผิวมนุษย์ออกจากส่วนอื่นๆ ของภาพ โดยการทดลองแสดงถึงความกระจายตัวของสีผิวมนุษย์ที่ความสว่างต่างๆ กัน

2.1.3 การสร้างฉากสามมิติ

ในงานวิจัยของ [10] เสนอแนวทางการสร้างฉากสามมิติ 2 รูปแบบ วิธีแรกจะเป็นการสร้างฉากขึ้นมาจากภาพถ่ายทัศนียภาพแบบกว้าง (panorama) โดยใช้การแก้ค่าความบิดเบี้ยวของภาพ เพื่อสร้างเป็นมุมมองที่ถูกต้องซึ่งวิธีนี้สามารถสร้างภาพจำลองได้เฉพาะในแนวนอน ส่วนวิธีที่สอง เป็นการสร้างฉากขึ้นจากแบบจำลองสามมิติ แล้วประมวลผลสร้างเป็นภาพฉากสามมิติขึ้นมา ด้วยความสามารถในการประมวลผลในขณะนั้น ทั้งสองวิธีสามารถทำงานได้แบบทันกาล โดยวิธีแรกใช้คอมพิวเตอร์ในการประมวลผลเครื่องเดียว แต่วิธีที่สองใช้คอมพิวเตอร์หลายเครื่องช่วยกันทำงาน

แนวคิดของการวิจัยสำหรับในงานวิจัยนี้ ระบบฉากเสมือนที่สร้างขึ้นมีจุดมุ่งหมายเพื่อใช้ผลิตผลงานสำหรับใช้ในวงการโทรทัศน์ ซึ่งแนวคิดหลักจะไม่แตกต่างกับระบบฉากเสมือนของเดิมมากนัก แต่จะมีส่วนการจับภาพที่สามารถเปลี่ยนแปลงมุมกล้องได้อย่างอิสระ และการสร้างฉากที่เป็นฉากแบบสามมิติที่สร้าง

ขึ้นโดยคอมพิวเตอร์ในระดับเวลาจริง โดยสามารถแยกการทำงานออกเป็นขั้นตอนต่างๆ ได้ดังนี้

1. ถ่ายภาพวัตถุที่ต้องการโดยมีฉากหลังเป็นฉากที่เตรียมไว้ล่วงหน้า
2. หาเวคเตอร์การย้ายตำแหน่ง และเมตริกซ์การหมุนจากภาพที่ได้ในขั้นตอนที่หนึ่ง
3. ลบฉากหลังส่วนที่ไม่ต้องการในภาพที่ได้จากขั้นตอนแรกออก
4. นำข้อมูลของเวคเตอร์การย้ายตำแหน่ง และเมตริกซ์การหมุน ที่ได้จากในขั้นตอนที่สอง สร้างเป็นฉากสามมิติโดยกำหนดมุมมองของการเรนเดอร์ให้สอดคล้องกับเวคเตอร์และเมตริกซ์ทั้งสองค่าที่ได้มา
5. นำภาพที่ลบฉากหลังออกแล้วในขั้นตอนที่สาม ซ้อนทับลงบนฉากที่เรนเดอร์ขึ้นมาในขั้นตอนที่สี่ ได้เป็นผลลัพธ์สุดท้าย

2.2 ทฤษฎีที่เกี่ยวข้อง

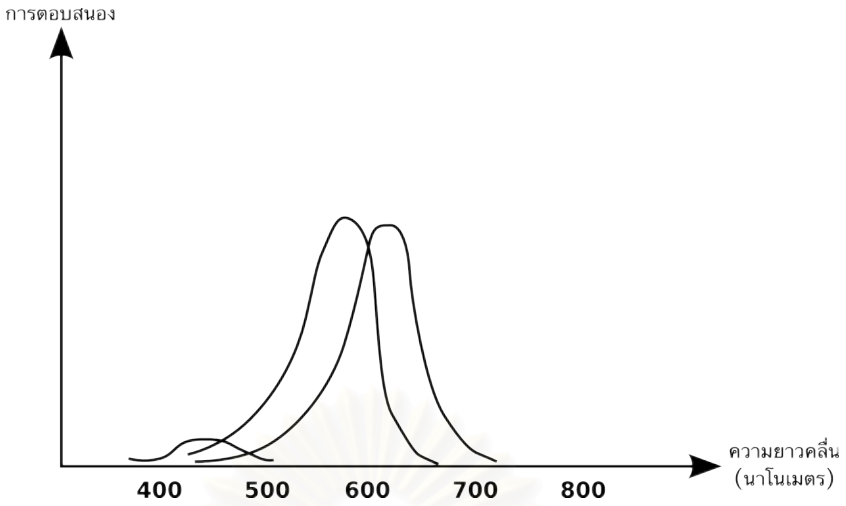
2.2.1 ทฤษฎีด้านคอมพิวเตอร์กราฟิก

ธรรมชาติการมองเห็นของมนุษย์นั้น เกิดจากการที่มีแสงตกกระทบวัตถุแล้วสะท้อนเข้าตา โดยที่ในดวงตาจะมีประสาทตอบสนองต่อความเข้มของสีต่างๆ แล้วส่งต่อข้อมูลเหล่านี้ให้สมองแปลความหมายต่อไป สำหรับในระบบคอมพิวเตอร์มีการทำงานที่แตกต่างออกไป แต่ก็เลียนแบบการมองเห็นของมนุษย์ โดยการนำภาพเข้าสู่ระบบคอมพิวเตอร์นั้นสามารถทำได้หลายวิธี เช่น กล้องดิจิทัล, กล้องวิดีโอ, สแกนเนอร์เป็นต้น เทียบกับดวงตาของมนุษย์ที่มีการทำงานในแบบอนาล็อกแล้ว อุปกรณ์คอมพิวเตอร์เหล่านี้จะใช้ CCD (Charge-Coupled Device) ในการรับภาพ ซึ่งจะให้ผลการวัดค่าของแสงออกมาเป็นปริมาณทางดิจิทัล ซึ่งคอมพิวเตอร์สามารถนำไปประมวลผลต่อได้

สำหรับภาพเคลื่อนไหวหรือภาพวิดีโอนั้นเกิดจากการฉายภาพนิ่งหลายๆ ภาพต่อเนื่องด้วยความเร็วสูงจนตาของมนุษย์มองเห็นเป็นภาพเคลื่อนไหวต่อเนื่อง ซึ่งโดยปกติแล้วความเร็วขั้นต่ำในการแสดงภาพต่อเนื่องที่ตาของมนุษย์ไม่ทันสังเกตนั้นอยู่ที่ประมาณ 24 ภาพต่อวินาที ดังนั้นในงานวิจัยนี้ จำเป็นต้องประมวลผลภาพที่ได้รับมาในแต่ละภาพให้เสร็จอย่างรวดเร็วอย่างน้อยที่ 24 ภาพต่อวินาที ผู้ใช้จึงจะไม่รู้สึกว่าภาพที่เห็นมีอาการกระตุกหรือสะดุด

2.2.1.1 สีปฐมภูมิ

ประสาทตาของมนุษย์ปกติจะตอบสนองต่อคลื่นที่ความถี่ 400 นาโนเมตรถึง 700 นาโนเมตร [11] โดยที่ในดวงตาของมนุษย์จะมีเซลประสาทที่ตอบสนองต่อคลื่นความถี่ต่างๆ กันอยู่ 3 ช่วงความถี่ คือที่ความถี่ของสีน้ำเงิน สีเขียว และสีแดง ดังรูปที่ 2.4 ซึ่งเมื่อสมองนำไปแปลความหมายแล้วจะได้ออกมาเป็นภาพอย่างที่เราเห็นกัน



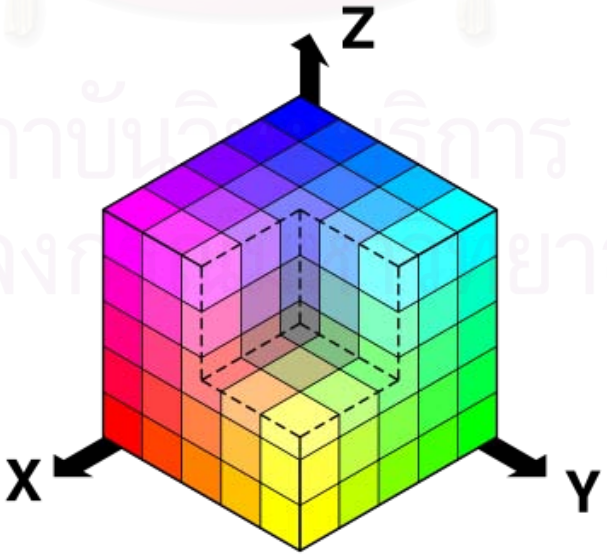
รูปที่ 2.4: แสดงการตอบสนองของประสาทตามนุษย์ที่ความถี่ต่างๆ

2.2.1.2 ระบบสี

ในทางคอมพิวเตอร์ การอ้างอิงถึงสีใดสีหนึ่ง สามารถทำได้หลายวิธี โดยปกติจะกำหนดเป็นกลุ่มของตัวเลข 3 หรือ 4 ตัว เพื่อใช้แทนแต่ละส่วนประกอบในการอธิบายสีนั้นๆ โดยระบบสีที่นิยมใช้กันก็เช่น RGB, CMY, HSV

ระบบสีแบบ RGB

ระบบสีแบบ RGB ประกอบขึ้นจากส่วนประกอบพื้นฐานสามส่วนคือ สีแดง (Red) สีเขียว (Green) และสีน้ำเงิน (Blue) โดยการระบุสีในระบบสีแบบ RGB นี้จะระบุด้วยค่าทั้งสามค่านี้ในสัดส่วนที่ต่างๆ กันเพื่อสร้างเป็นสีที่ต้องการ เราสามารถจำลองภาพของระบบสี RGB นี้ได้ด้วยการกำหนดลูกบาศก์ขนาด 1x1x1 และแทนที่แกน x, y, z ด้วยค่าสี R, G, B จะได้ลูกบาศก์ของสีดังในรูปที่ 2.5



รูปที่ 2.5: แสดงระบบสี RGB บนแกนสามมิติ

ระบบสีแบบ RGB นี้จัดเป็นระบบสีแบบ additive เนื่องจากที่จุด (0, 0, 0) หมายถึงสีดำ เมื่อแต่ละสีมีค่าเพิ่มขึ้นก็จะได้ค่าสีที่สว่างมากขึ้น จนกระทั่งที่จุด (1, 1, 1) ซึ่งหมายถึงสีขาว

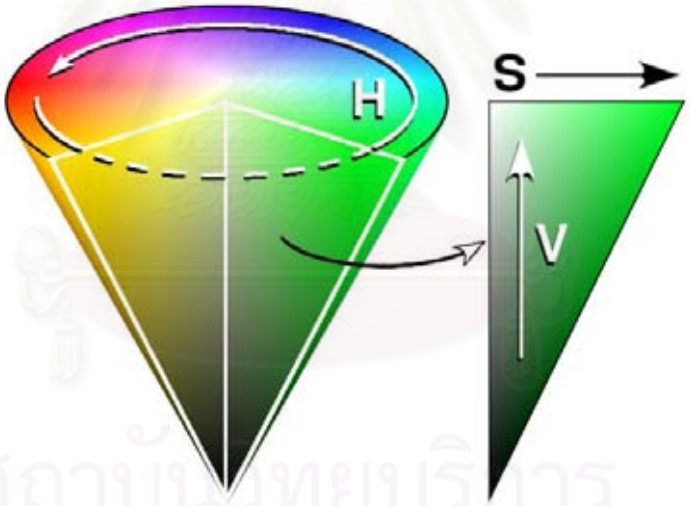
ระบบสีแบบ CMY

ระบบสีแบบ CMY ประกอบด้วยส่วนประกอบย่อยของสามสี ได้แก่ สีฟ้า (Cyan) สีม่วงแดง (Magenta) และสีเหลือง (Yellow) ซึ่งทั้งสามสีเป็นสีหลักของระบบงานพิมพ์ หมึกสีฟ้าจะดูดกลืนสีแดง หมึกสีม่วงแดงจะดูดกลืนสีเขียว และหมึกสีเหลืองจะดูดกลืนสีน้ำเงิน ดังนั้นสีที่มองเห็นจะเป็นผลจากการสะท้อนสีที่ไม่ถูกดูดกลืนออกมา ด้วยคุณสมบัติแบบนี้ เราจึงจัดระบบสีแบบ CMY ว่าเป็นระบบสีแบบ subtractive

โดยพื้นฐานแล้ว ที่ค่า CMY = (0, 0, 0) จะหมายความว่า ไม่มีการดูดกลืนสีใดๆ เลย ดังนั้นแสงที่ตกกระทบจะสะท้อนไปทุกช่วงความถี่ ทำให้เห็นเป็นสีขาว และที่ (1, 1, 1) หมายถึงทุกสีจะถูกดูดกลืนทั้งหมดโดยไม่มีการสะท้อนสีใดออกไปเลย ทำให้มองเห็นเป็นสีดำ

ระบบสีแบบ HSV

โดยที่จริงแล้ว ระบบสีแบบ HSV นั้นเป็การนำระบบสีแบบ RGB มาผ่านกระบวนการเปลี่ยนรูปอย่างไม่เป็นเชิงเส้น (nonlinear transformation) เพราะการระบุค่าสีแบบแยก RGB นั้นทำความเข้าใจได้ยาก ในระบบสีแบบ HSV จึงใช้การระบุค่าแบบ Hue, Saturation และ Value แทน



รูปที่ 2.6: กรวยแสดงระบบสีแบบ HSV

โดยจากในรูปที่ 2.6 ค่าของ hue จะเป็นค่าที่กำหนดว่าเป็นสีอะไร ซึ่งค่าที่ 0 จะหมายถึงสีแดง และวนไปแบบทวนเข็มนาฬิกา ค่าที่ π จะหมายถึงสีฟ้า และกลับมาจบที่สีแดงอีกครั้งที่ค่า 2π ส่วนค่าของ saturation คือความอิ่มตัวของสี แทนด้วยระยะห่างจากจุดศูนย์กลางของวงกลม โดยสีที่มีค่าความอิ่มตัวน้อยจะมีสีใกล้เคียงกับสีดำและสีเทา สีที่มีค่าความอิ่มตัวมากจะมีความสดและฉูดฉาด (vivid) สูง และสุดท้ายค่า value คือค่าความสว่างของสีนั้นๆ บางครั้งก็เรียกว่า brightness ทำให้ย่อได้เป็น HSB

2.2.1.3 การจัดเก็บภาพกราฟิกแบบสองมิติ

ภาพกราฟิกที่ใช้ในงานคอมพิวเตอร์ สามารถแบ่งตามลักษณะการจัดเก็บได้เป็น 2 ประเภทใหญ่ๆ คือ ภาพแบบเวคเตอร์ (vector) [12] และภาพแบบราสเตอร์ (raster) [13] ซึ่งภาพแบบเวคเตอร์จะมีการจัดเก็บรูปภาพในลักษณะของสมการทางคณิตศาสตร์ แทนสี เส้นตรง เส้นโค้ง ทั้งหมดด้วยสมการ ส่วนภาพแบบราสเตอร์มีการจัดเก็บคือแบ่งภาพที่ต้องการจัดเก็บออกเป็นตาราง แล้วสุ่มตัวอย่าง (sampling) ค่าภายในตารางแต่ละช่องและจัดเก็บโดยที่ตาราง 1 ช่องจะแทนที่ด้วยจุด 1 จุด

รูปที่ 2.7: เปรียบเทียบภาพกราฟิกแบบเวคเตอร์และราสเตอร์

ในรูปที่ 2.7 เป็นการเปรียบเทียบภาพกราฟิกแบบเวคเตอร์ (B) และภาพกราฟิกแบบราสเตอร์ (C) โดยที่ภาพ A เป็นภาพต้นฉบับ จะเห็นได้ว่าเมื่อมีการขยายภาพ ภาพแบบเวคเตอร์ยังคงความคมชัดเอาไว้ได้ เนื่องจากเป็นภาพที่ประกอบขึ้นจากสมการทางคณิตศาสตร์ แต่ภาพแบบราสเตอร์จะดูแตกออกเป็นจุดให้เห็นได้ชัดเจน เนื่องจากมีการจัดเก็บเรียงไปแบบจุดต่อจุด

โดยปกติแล้วภาพที่ถ่ายได้จากกล้องดิจิทัลหรือกล้องวิดีโอ จะให้ภาพออกมาเป็นกราฟิกแบบราสเตอร์ ซึ่งประกอบด้วยจุดซึ่งเป็นหน่วยย่อยที่เล็กที่สุดของภาพประเภทนี้ ซึ่งเราเรียกว่าพิกเซล (pixel: picture element)

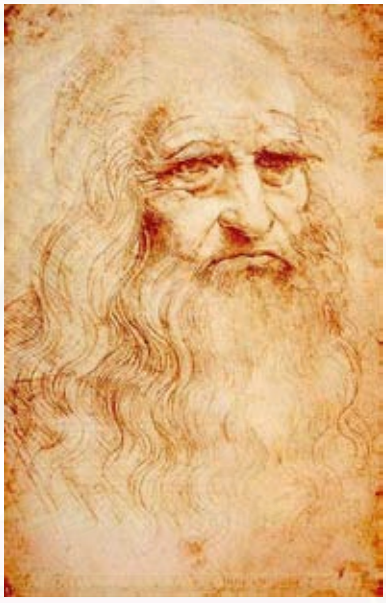
ในการจัดเก็บข้อมูล ภาพราสเตอร์ทั้งภาพจะถูกแทนด้วยจุดตามแนวกว้างและแนวยาว โดยแต่ละจุดจะมีค่าสีตามระบบสีที่ใช้งาน เช่น ภาพที่ใช้ระบบสีแบบ RGB แต่ละจุดจะแทนที่ด้วยระดับของสีทั้งสามสี หรือถ้าเป็นภาพแบบระดับสีเทา แต่ละจุดจะแทนที่ด้วยระดับของสีเดียว โดยการจัดเก็บอาจจัดเก็บเรียงแต่ละพิกเซลจากมุมใดมุมหนึ่ง เรียงกันไปตามแนวนอนหรือแนวตั้งก็ได้ ขึ้นอยู่กับรูปแบบ (format) ของภาพ

คุณภาพของภาพแบบราสเตอร์แบบที่ไม่มีการบีบอัดจะขึ้นอยู่กับ 2 ปัจจัยคือ ขนาดของภาพและจำนวนบิตสี ขนาดของภาพคือจำนวนพิกเซลทั้งหมดที่ประกอบขึ้นมาเป็นภาพ เรานิยมบอกขนาดของภาพในหน่วยของความกว้างและความยาวของจำนวนพิกเซล เช่น ภาพขนาด 800x600 พิกเซล หมายถึงภาพที่มีจำนวนจุดในแนวนอน 800 พิกเซล และมีจำนวนจุดในแนวตั้ง 600 พิกเซล ดังนั้น ภาพที่มีความละเอียดสูง จะมีคุณภาพสูงกว่าภาพที่มีความละเอียดต่ำ ส่วนจำนวนบิตสีคือข้อมูลที่บอกว่าในแต่ละพิกเซล สามารถแสดงสีที่ต่างกันได้กี่ระดับ เช่น ภาพระดับเทา 8 bit หมายถึงในแต่ละพิกเซล สามารถแสดงระดับของสีเทาที่แตกต่างกันได้ 256 สี หรือภาพ RGB 24 bit หมายถึงในแต่ละพิกเซล จะใช้เนื้อที่ 8 bit ในการเก็บแต่ละสี ทำให้ใน 1 พิกเซล แต่ละสีสามารถแสดงความแตกต่างได้ 256 ระดับ นั่นคือภาพ RGB แบบ 24 bit

สามารถแสดงสีได้สูงสุด 256x256x256 หรือประมาณ 16.7 ล้านสี ซึ่งครอบคลุมสีทั้งหมดที่ตาของมนุษย์สามารถแยกแยะออกได้ และเพียงพอสำหรับการแสดงผลทั่วไป

2.2.1.4 การจัดเก็บภาพกราฟิกแบบสามมิติ

การแสดงผลภาพกราฟิกของคอมพิวเตอร์ ปกติจะทำโดยการแสดงผลผ่านจอภาพซึ่งแสดงผลได้เพียงภาพสองมิติ นั่นก็คือภาพกราฟิกสามมิติ ที่จริงแล้วก็เป็นภาพประเภทเดียวกับภาพกราฟิกสองมิติ ทว่าสิ่งที่แตกต่างกันคือในกระบวนการสร้างภาพกราฟิกสองมิติ จะมีแกนที่เกี่ยวข้องเพียงสองแกนคือแกน x และแกน y เท่านั้น ทำให้ภาพมีเพียงข้อมูลของความกว้างและความสูง แบนราบ ขาดมิติของแสงเงา ถ้าต้องการจะสร้างเป็นภาพเคลื่อนไหว จะต้องสร้างภาพใหม่ขึ้นสำหรับทุกๆ เฟรม รูปตัวอย่างที่ได้จากภาพกราฟิกสองมิติ แสดงดังรูป 2.8



รูปที่ 2.8: ตัวอย่างภาพกราฟิกสองมิติ

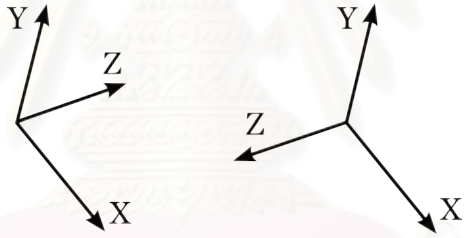
สำหรับการสร้างภาพกราฟิกแบบสามมิตินั้น ในกระบวนการสร้างจะเกี่ยวข้องกับความตื้น-ลึกด้วย โดยจะทำงานบนแกนทั้งสามแกนคือแกน x, y และ z ซึ่งการสร้างภาพกราฟิกสามมิติแบบนี้จะต้องใช้โปรแกรมที่มีความสามารถในการสร้างโมเดลสามมิติได้ ตัวอย่างเช่น 3D Studio, Blender เป็นต้น แต่จอภาพคอมพิวเตอร์ไม่สามารถแสดงความลึกของโมเดลสามมิติได้ จึงจำเป็นต้องมีกระบวนการในการแปลงโมเดลสามมิติให้กลายเป็นภาพสองมิติ เพื่อสามารถแสดงผลบนจอภาพได้ โดยกระบวนการนี้เรียกว่าการเรนเดอร์ (render)

กระบวนการสร้างภาพกราฟิกสามมิติมีข้อดีที่เหนือกว่าภาพกราฟิกสองมิติข้อหนึ่งคือ ความสามารถในการเปลี่ยนแปลงมุมมองหรือมุมกล้องที่ทำต่อวัตถุได้อย่างอิสระ ทำให้ภาพที่เรนเดอร์ออกมาได้มีลักษณะที่สมจริง มีความลึกและทิศทางของแสงเงาที่ถูกต้องตามความเป็นจริง และยังทำให้การสร้างภาพเคลื่อนไหวสามารถทำได้ง่ายกว่าภาพกราฟิกแบบสองมิติ โดยในกระบวนการสร้างภาพกราฟิกสามมิติ ผู้ใช้งานเพียงแค่เปลี่ยนมุมกล้องหรือย้ายตำแหน่งวัตถุตามต้องการ แล้วสั่งให้โปรแกรมเรนเดอร์ใหม่ ก็จะได้ภาพที่ต้องการทันที ต่างจากภาพกราฟิกสองมิติที่ถ้ามีการเปลี่ยนแปลงมุมมองต่อวัตถุ ก็จะต้องวาดภาพ

วัตถุนั้นขึ้นมาใหม่ทุกครั้งโดยไม่สามารถนำเอาข้อมูลจากภาพเดิมมาแก้ไขได้ ตัวอย่างภาพกราฟิกสามมิติแสดงดังรูป 2.9

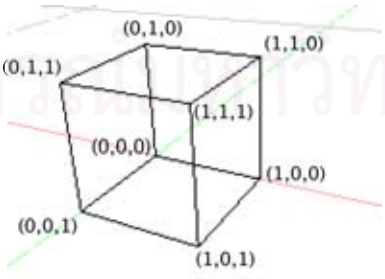
รูปที่ 2.9: ตัวอย่างภาพกราฟิกสามมิติ

ในการสร้างภาพกราฟิกสองมิติ เรากำหนดจุดต่างๆ ให้อยู่บนปริภูมิสองมิติของแกน x และแกน y แต่ในกระบวนการสร้างภาพสามมิติ เรากำหนดให้จุดต่างๆ อยู่บนปริภูมิสามมิติ ที่ประกอบด้วยแกน x, y และ z โดยระบบที่ใช้งานมีอยู่ 2 ระบบ คือ ระบบแกนมือขวา และระบบแกนมือซ้าย ความแตกต่างของทั้งสองระบบดูได้จากรูปที่ 2.10



รูปที่ 2.10: ภาพเปรียบเทียบระบบแกนมือซ้าย (ซ้าย) และระบบแกนมือขวา (ขวา)

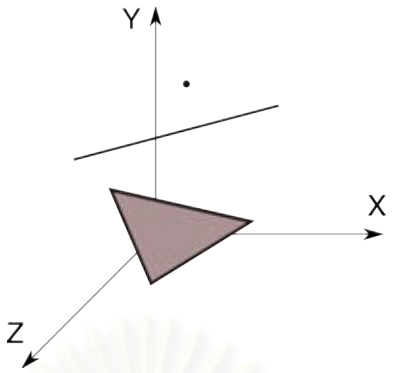
เราสามารถอ้างอิงถึงพิกัดต่างๆ ในปริภูมิสามมิตินี้โดยใช้ตัวเลขชุดจำนวน 3 ตัว ซึ่งแทนค่าบนแกน x, y และ z ตามลำดับ ยกตัวอย่างพิกัดของจุดมุมของลูกบาศก์ขนาด 1 หน่วย แสดงได้ดังรูป 2.11



รูปที่ 2.11: แสดงพิกัด (x, y, z) ของลูกบาศก์ขนาด 1 หน่วย

เมื่อจุดสองจุดถูกเชื่อมเข้าด้วยกัน ก็จะเกิดเป็นเส้นตรง และถ้าจุดตั้งแต่สามจุดขึ้นไปถูกเชื่อมเข้าด้วยกันเป็นพื้นผิวปิด จะเกิดเป็นรูปหลายเหลี่ยม (polygon) ซึ่งจุดมุมแต่ละจุดของรูปหลายเหลี่ยมจะเรียก

ว่าจุดยอด (vertex) ลักษณะของจุด เส้น และรูปหลายเหลี่ยมแสดงดังรูป 2.12



รูปที่ 2.12: แสดง จุด เส้น และรูปหลายเหลี่ยม

รูปหลายเหลี่ยมพื้นฐานที่นิยมใช้เป็นส่วนประกอบในการสร้างโมเดลสามมิติคือรูปหลายเหลี่ยมรูปสามเหลี่ยม เนื่องจากมีประสิทธิภาพในการคำนวณสูงที่สุด โมเดลแต่ละชิ้นจะเกิดจากการนำเอารูปหลายเหลี่ยมหลายๆ ชิ้นมาประกอบกันจนเกิดเป็นรูปร่างที่ต้องการ ดังตัวอย่างในรูป 2.13 (ภาพจาก [14])

รูปที่ 2.13: แสดงการสร้างโมเดลสามมิติขึ้นจากรูปหลายเหลี่ยมจำนวนหลายๆ ชิ้น

โมเดลสามมิติที่ประกอบด้วยรูปหลายเหลี่ยมจำนวนมาก ก็จะดูละเอียดกว่าโมเดลที่ประกอบขึ้นจากรูปหลายเหลี่ยมจำนวนน้อยกว่า ทำให้ภาพที่เรนเดอร์ออกมามีความสมจริง แต่ยิ่งมีจำนวนของรูปหลายเหลี่ยมมาก ก็จะยิ่งเสียเวลาในการเรนเดอร์นานขึ้น ภาพเปรียบเทียบระหว่างโมเดลแบบละเอียดและแบบหยาบแสดงดังในรูป 2.14

รูปที่ 2.14: ภาพแสดงความละเอียดของโมเดลที่มีจำนวนรูปหลายเหลี่ยมต่างกัน

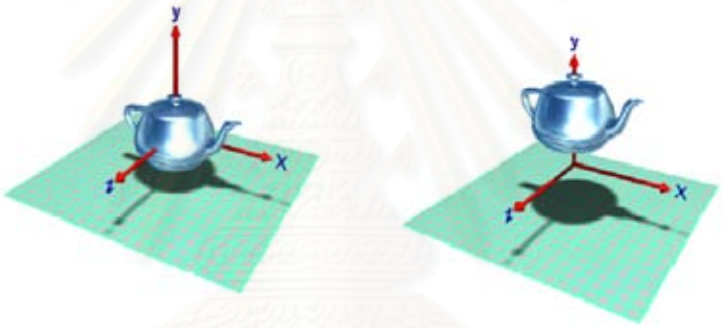
2.2.1.5 ทฤษฎีเกี่ยวกับเรขาคณิตในสามมิติ

เมื่อเรามีโมเดลสามมิติและวิธีการระบุตำแหน่งของจุดต่างๆ ในโลกสามมิติแล้ว เราสามารถใช้การคำนวณทางคณิตศาสตร์เพื่อหมุน, ย้ายตำแหน่ง, ปรับขนาด ของจุดใดๆ ที่อยู่ในโลกสามมิติได้ [11] ดังนี้

การย้ายตำแหน่ง (Translation)

เราใช้เมตริกซ์เป็นเครื่องมือในการคำนวณการย้ายตำแหน่ง (translation) โดยสมมุติว่าวัตถุอยู่ที่พิกัด (x_1, y_1, z_1) อยากให้เคลื่อนที่ไปในแกนสามมิติตามเวคเตอร์ (x_0, y_0, z_0) จะคำนวณพิกัดใหม่ได้เป็นที่ (x_2, y_2, z_2) ตามสมการ 2.1 และแสดงได้ดังรูป 2.15

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & x_0 \\ 0 & 1 & 0 & y_0 \\ 0 & 0 & 1 & z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} \tag{2.1}$$



รูปที่ 2.15: การย้ายตำแหน่งในพิกัดสามมิติ

การหมุน (Rotation)

การหมุนทำได้โดยนำเมตริกซ์การหมุน (rotation matrix) มาคูณ สมมุติว่าพิกัดเริ่มต้นของเราคือที่ (x_1, y_1, z_1) ทำการหมุนไปเป็นมุม θ รอบแกน X จะใช้เมตริกซ์การหมุนดังในสมการ 2.2

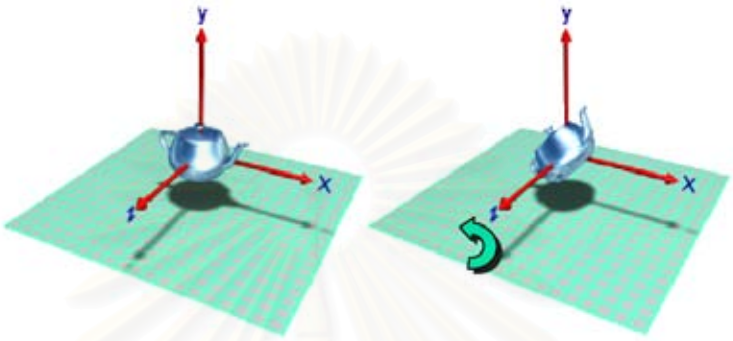
$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & cos\theta & -sin\theta & 0 \\ 0 & sin\theta & cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} \tag{2.2}$$

การหมุนเป็นมุม θ รอบแกน Y ใช้การคูณด้วยเมตริกซ์การหมุนดังในสมการ 2.3

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} = \begin{bmatrix} cos\theta & 0 & sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -sin\theta & 0 & cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} \tag{2.3}$$

การหมุนเป็นมุม θ รอบแกน Z ใช้การคูณด้วยเมตริกซ์การหมุนดังสมการ 2.4 แสดงได้ดังรูป 2.16

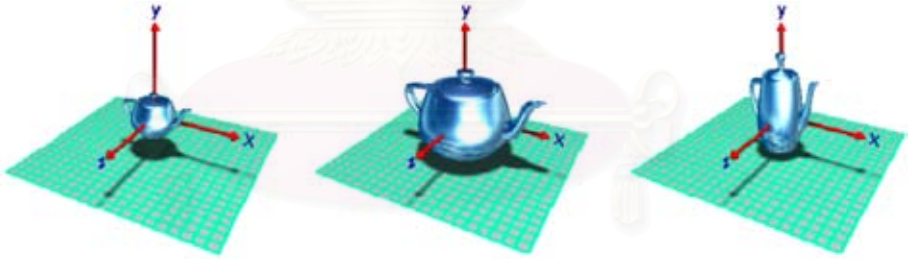
$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta & 0 & 0 \\ sin\theta & cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} \quad (2.4)$$



รูปที่ 2.16: การหมุนในพิกัดสามมิติ

การเปลี่ยนขนาด (Scaling)

การเปลี่ยนขนาดทำได้โดยการคูณเมตริกซ์เช่นกันโดยเราสามารถกำหนดอัตราส่วนของการขยายขนาดในแต่ละแนวแกนทั้งสามได้ ดังแสดงในรูป 2.17



รูปที่ 2.17: การเปลี่ยนขนาดในพิกัดสามมิติ

2.2.1.6 กล้องในโลกสามมิติ

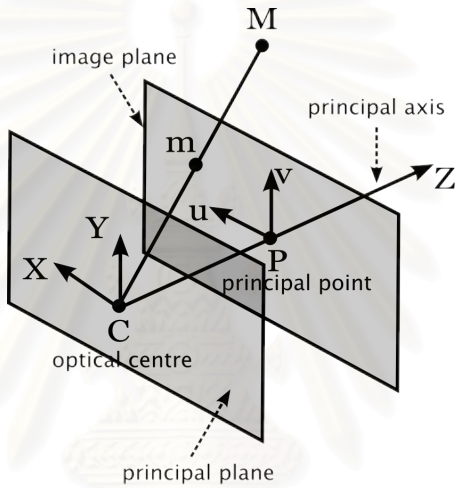
ในกระบวนการสร้างภาพสามมิติ การจะกำหนดว่าส่วนใดของฉากและโมเดลจะถูกแสดงบนจอภาพจะกำหนดผ่านตำแหน่งและมุมของกล้องในโลกสามมิติ ซึ่งถ้าเปรียบเทียบแล้วก็จะมีหลักการทำงานที่ใกล้เคียงกับกล้องที่มีอยู่ในโลกจริง คือสิ่งที่กล้องจับอยู่ คือสิ่งที่จะถูกแสดงบนจอภาพ ในกรณีที่เป็นกล้องในโลกสามมิติ หรือเป็นสิ่งที่จะถูกบันทึกลงบนฟิล์ม ในกรณีที่เป็นกล้องในโลกจริง กล้องสามารถปรับเปลี่ยนตำแหน่งการจัดวางและทิศทางการหมุนให้เป็นไปตามที่ต้องการได้ ทำให้กระบวนการสร้างภาพสามมิติ สามารถสร้างภาพขึ้นจากมุมมองใดๆ ก็ได้โดยอิสระ

กล้องในโลกจริงจะมีความยาวโฟกัสของเลนส์ที่ใช้ถ่าย โดยมีหน่วยวัดเป็นมิลลิเมตร ถ้าเป็นเลนส์ที่

ถ่ายภาพมุมกว้างจะมีความยาวโฟกัสต่ำ ถ้าเป็นเลนส์สำหรับถ่ายภาพระยะไกลจะมีความยาวโฟกัสสูงและมีมุมรับภาพที่แคบ ส่วนกล้องในโลกสามมิติก็มีคุณสมบัติที่คล้ายกัน โดยกำหนดสิ่งที่เรียกว่ามุมรับภาพ (FOV: Field of view) ขึ้นมาใช้แทนหน่วยของความยาวโฟกัส

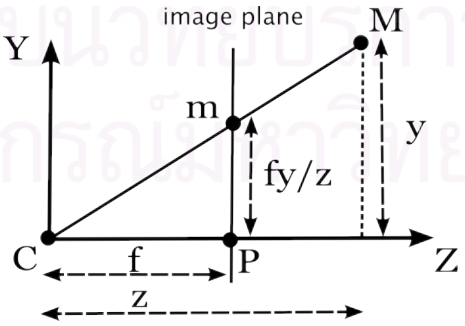
2.2.1.7 โมเดลกล้องแบบรูเข็ม

ในงานวิจัยนี้เลือกใช้โมเดลของกล้องรูเข็ม (Pinhole camera) [11] ในการคำนวณทางสามมิติ โดยกล้องรูเข็มมีลักษณะดังแสดงในรูป 2.18 มีส่วนประกอบต่างๆ คือ จุด C เป็นตำแหน่งของจุดศูนย์กลางของกล้อง (Optical center) ระยะห่างระหว่างจุดศูนย์กลางของกล้องจนถึงระนาบรับภาพ (Image plane) เรียกว่าระยะโฟกัส (Focal length) จุดศูนย์กลางของกล้องอยู่บนระนาบหนึ่งซึ่งขนานกับระนาบรับภาพ เรียกระนาบนี้ว่าระนาบหลัก (Principal plane)



รูปที่ 2.18: แสดงส่วนต่างๆ ของกล้องรูเข็ม

ความสัมพันธ์ระหว่างจุดในโลกสามมิติกับจุดที่ปรากฏบนระนาบรับภาพแสดงได้ดังรูปที่ 2.19 จุด M ในโลกสามมิติ จะปรากฏเป็นจุด m บนระนาบรับภาพ โดยที่จุด M, m และ C จะอยู่บนเส้นตรงเดียวกัน



รูปที่ 2.19: รูปแสดงความสัมพันธ์ระหว่างจุดในโลกสามมิติและจุดที่ปรากฏบนระนาบรับภาพ

ความสัมพันธ์ระหว่างจุดที่ปรากฏบนระนาบรับภาพและจุดในโลกสามมิติสามารถเขียนได้ตาม

สมการ

$$m = A[Rt]M \tag{2.5}$$

โดยที่

R คือเมตริกซ์การหมุน

t คือเวคเตอร์การย้ายตำแหน่ง

A คือพารามิเตอร์ภายใน (intrinsic parameter) ของกล้อง

m คือตำแหน่งของจุดบนระนาบรับภาพ

M คือตำแหน่งของจุดในโลกสามมิติ

เราเรียก R และ t ว่าเป็นพารามิเตอร์ภายนอก (extrinsic parameter) ซึ่งจะมีค่าเปลี่ยนแปลงไปตามตำแหน่งและแนวการวางตัวของกล้อง แต่พารามิเตอร์ภายในของกล้องแต่ละตัวจะมีค่าคงที่เสมอ

พารามิเตอร์ภายในของกล้องแต่ละตัวแสดงค่าได้ในรูปของเมตริกซ์

$$A = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2.6}$$

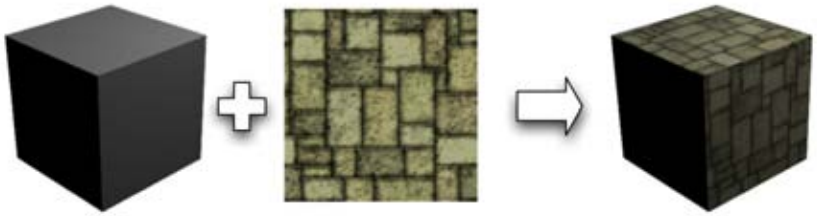
(f_x, f_y) คือความยาวโฟกัส (focal length)

(c_x, c_y) คือพิกัดของจุดกึ่งกลางภาพ (principal point)

เราสามารถใช้ความสัมพันธ์ข้างต้นในการคำนวณพิกัดในโลกจริง (x, y, z) ได้โดยการพิจารณาจากพิกัดของจุดบนระนาบรับภาพ (x, y) แต่ในการแปลงจุดพิกัดสองมิติให้อยู่ในพิกัดสามมิตินั้นใช้จุดเดียวไม่เพียงพอ เพราะการแปลงโดยใช้จุดสองมิติเพียงจุดเดียวจะได้ผลลัพธ์แค่เพียงแนวการฉายของจุดในระนาบสามมิติ ดังนั้นจึงจำเป็นต้องใช้จุดมากกว่าหนึ่งจุดที่มีรูปแบบการเรียงตัวที่แน่นอน จึงจะเพียงพอสำหรับการคำนวณ

2.2.1.8 ภาพพื้นผิว

ในการสร้างฉากและโมเดลสามมิตินั้น เราสามารถเพิ่มความสวยงามและสมจริงได้โดยการใช้ภาพพื้นผิว เพื่อแสดงลวดลาย จำลองลักษณะของพื้นผิวให้มีความสมจริงยิ่งขึ้น โดยขั้นตอนในการปะภาพพื้นผิวลงบนวัตถุในโลกสามมิติแสดงได้ดังรูป 2.20 โดยขั้นตอนการทำจะต้องใช้ภาพกราฟิกสองมิติ เช่น ภาพตระกูล JPG, TIF หรือ BMP กำหนดตำแหน่งของภาพสองมิติเข้ากับพิกัดในโลกสามมิติ และปะภาพพื้นผิวนั้นเข้ากับรูปหลายเหลี่ยมที่ต้องการจนครบทุกหน้าที่ต้องการ



รูปที่ 2.20: แสดงการใช้ภาพพื้นผิวในการสร้างความสมจริงให้กับโมเดลสามมิติ

2.2.1.9 กลไกการสร้างภาพสามมิติ

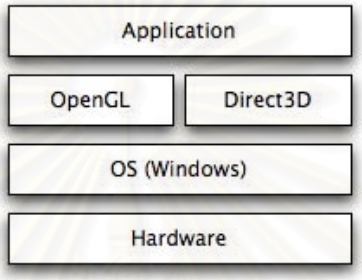
กระบวนการสร้างภาพกราฟิกสามมิตินั้น สามารถสร้างได้ทั้งในแบบทันกาล (real-time) และแบบเป็นชุด (batch) โดยในแบบทันกาลนั้นสามารถสร้างภาพสามมิติขึ้นมาตอบสนองได้ทันกับการเปลี่ยนแปลงของฉาก มุมมอง วัตถุ ส่วนในการคำนวณแบบเป็นชุดนั้นเป็นการสร้างภาพสามมิติโดยกำหนดมุมกล้องและตำแหน่งของวัตถุไว้ตั้งแต่แรก ไม่สามารถเปลี่ยนแปลงในทันทีได้ หากต้องการเปลี่ยนแปลงจะต้องเริ่มกระบวนการเรนเดอร์ใหม่

กระบวนการสร้างภาพกราฟิกสามมิติแบบทันกาลมีข้อดีคือสามารถสร้างภาพออกมาได้ทันต่อการเปลี่ยนแปลง จึงเหมาะสำหรับการใช้งานที่ต้องตอบสนองต่อการใช้งานของผู้ใช้ในทันที เช่น เกมสามมิติ แต่วิธีนี้มีจุดด้อยคือ ความเร็วในการแสดงผล, ความละเอียดของฉาก และคุณภาพของภาพผลลัพธ์จะขึ้นอยู่กับพลังการประมวลผลของเครื่อง ถ้าหากใช้เครื่องที่มีความสามารถในการประมวลผลต่ำ จะไม่สามารถสร้างภาพเคลื่อนไหวที่ความเร็วที่มองแล้วลื่นไหลได้ ภาพที่เห็นจะดูกระตุก ขัดกับความรู้สึก ส่วนกระบวนการสร้างภาพแบบเป็นชุดมีการกำหนดจังหวะเวลาการเปลี่ยนแปลงของโมเดล ฉาก หรือมุมกล้องเอาไว้ตั้งแต่ต้นแล้ว จากนั้นจึงมาคำนวณให้ออกเป็นภาพสามมิติ วิธีนี้ตอบสนองต่อปฏิสัมพันธ์ของผู้ใช้ไม่ได้ แต่ว่าสามารถใช้เวลาในการคำนวณภาพกราฟิกแต่ละภาพได้โดยไม่จำกัด ดังนั้นเราจึงสามารถออกแบบโมเดลสามมิติให้มีความซับซ้อนอย่างไรก็ได้ เพื่อผลลัพธ์สุดท้ายจะได้ออกมาดูดี แต่การสร้างโมเดลหรือฉากที่มีความซับซ้อนของรูปหลายเหลี่ยมสูง มีการเปลี่ยนแปลงของแสงเงา ก็จะใช้เวลาในการคำนวณมากตามไปด้วย ตัวอย่างของงานที่เหมาะกับการประมวลผลลักษณะนี้ เช่น ภาพยนตร์อนิเมชันต่างๆ

ตัวอย่างของภาพที่ได้จากการประมวลผลทั้งสองแบบแสดงดังรูป 2.21 จะสังเกตได้ว่าภาพที่ได้จากการประมวลผลแบบทันกาลมีรายละเอียดของโมเดล แสงเงา และความสมจริงน้อยกว่าภาพที่ได้จากการประมวลผลแบบเป็นชุด

การสร้างภาพสามมิติแบบทันกาลสามารถสร้างได้อย่างรวดเร็วโดยอาศัยประสิทธิภาพของฮาร์ดแวร์ เช่น การ์ดกราฟิกสามมิติ ซึ่งมีความสามารถในการประมวลผลแบบขนานและการประมวลผลเฉพาะทางสำหรับกราฟิกสามมิติ ทำให้ได้ความเร็วสูงกว่าการประมวลผลจากหน่วยประมวลผลกลางมาก การเขียนโปรแกรมควบคุมการทำงานของฮาร์ดแวร์เหล่านี้ทำได้โดยผ่านทางชุดคำสั่งสามมิติ ซึ่งชุดคำสั่งสามมิตินี้จะควบคุมการทำงานผ่านทางระบบปฏิบัติการและไดรเวอร์อีกต่อหนึ่ง ดังแสดงในรูป 2.22 สำหรับบนแพลตฟอร์มที่เป็นไมโครซอฟท์วินโดวส์ มีชุดคำสั่งให้เลือกใช้ 2 ชุด คือ OpenGL และ Direct3D

รูปที่ 2.21: เปรียบเทียบภาพที่ได้จากการประมวลผลแบบทันกาล (ซ้าย) และแบบเป็นชุด (ขวา)



รูปที่ 2.22: แสดงระดับการทำงานของชุดคำสั่ง

นอกจากชุดคำสั่งพื้นฐานที่ใช้ในการสร้างภาพสามมิติแล้ว ยังมีการสร้างชุดคำสั่งระดับสูงขึ้นไปอีก เพื่อให้การใช้งานชุดคำสั่งพื้นฐานอย่าง OpenGL และ Direct3D ทำได้ง่ายขึ้น ซึ่งนอกจากชุดคำสั่งสำหรับแสดงผลแล้ว อาจมีชุดคำสั่งสำหรับคำนวณด้านอื่น เช่น คำนวณการเคลื่อนที่ทางฟิสิกส์, ระบบเสียง หรือปัญญาประดิษฐ์ด้วย ชุดคำสั่งเหล่านี้มีทั้งแบบที่จำหน่ายในเชิงพาณิชย์และแบบใช้งานได้โดยไม่เสียค่าใช้จ่าย ตัวอย่างของชุดคำสั่งเหล่านี้ เช่น Java 3D, Irrlicht engine, Quake engine, Source engine

2.2.2 ทฤษฎีที่เกี่ยวข้องกับกระบวนการทางภาพยนตร์

2.2.2.1 การแยกภาพวัตถุออกจากฉากหลัง

การแยกภาพวัตถุออกจากฉากหลัง (Matting) คือการแยกภาพฉากที่ต้องการออกมาจากภาพหลัก โดยภาพที่แยกออกมาได้ (matte) จะบอกค่าความโปร่งใสในแต่ส่วนของภาพ ดังในรูป 2.23 ส่วนสีขาวหมายถึงส่วนที่โปร่งใส และส่วนสีดำแสดงส่วนที่ทึบแสง

รูปที่ 2.23: ภาพต้นฉบับ (ซ้าย) และภาพโปร่งใส (ขวา)

วิธีการแยกภาพฉากหลังและภาพวัตถุออกจากกันสามารถทำได้หลายแบบ ในที่นี้จะยกตัวอย่างของวิธีการพิจารณาจากความสว่าง (Luma key matting), การพิจารณาจากค่าสี (Chroma key matting) และการพิจารณาจากความแตกต่างของสี (Color difference matting) [15]

การแยกภาพโดยพิจารณาจากความสว่าง (Luma key matting)

วิธีการแยกภาพฉากหลังและภาพวัตถุออกจากกันโดยใช้ค่าความสว่างของภาพต้นฉบับ มีขั้นตอนในการประมวลผลคือ แปลงภาพต้นฉบับให้อยู่ในรูปแบบของภาพระดับสีเทาตามความสว่างของแต่ละจุด จากนั้นจึงกำหนดขอบเขตของความสว่างว่าค่าเท่าใด จึงจะมีความโปร่งใสสูงสุดและต่ำสุด

การแปลงภาพต้นฉบับให้เป็นภาพระดับสีเทาตามความสว่าง สามารถทำได้โดยการเฉลี่ยค่าสีของสีแดง สีเขียว และสีน้ำเงิน อย่างละเท่าๆ กันก็ได้ แต่ในความเป็นจริงแล้วประสาทตาของคนเรามีการตอบสนองต่อสีทั้งสามสี แตกต่างกัน ดังนั้นค่าความสว่าง (Luminance) อาจคำนวณได้จากสมการ [15]

$$Luminance = 0.29R + 0.59G + 0.12B \tag{2.7}$$

การกำหนดขอบเขตความสว่างของภาพระดับสีเทา สามารถกำหนดให้จุดที่มีความสว่างมากกว่าหรือน้อยกว่าขอบเขตที่กำหนดมีค่าความโปร่งใสได้ตามกำหนด โดยอาจกำหนดเป็นฟังก์ชันในการแปลงได้ ดังรูป 2.24 และ 2.25

รูปที่ 2.24: ภาพต้นฉบับก่อนผ่านกระบวนการแยกภาพ

รูปที่ 2.25: การแยกภาพฉากหลังออกโดยใช้ขอบเขตความสว่างเดียว

จากรูปเป็นการกำหนดฟังก์ชันการแปลงเป็นแบบมีขอบเขตเดียว (hard clip) จะทำให้ภาพที่ได้มีขอบเป็นรอยหยักเพราะภาพผลลัพธ์ที่ได้จะมีแค่ส่วนที่เป็นสีดำและสีขาวเท่านั้น ในกรณีที่ต้องการภาพที่มี

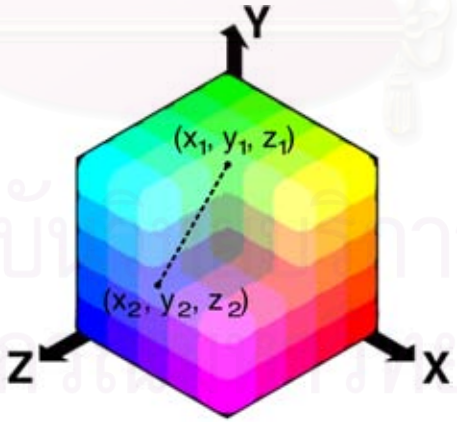
ผลลัพธ์บริเวณขอบเรียบเนียนกว่า ในขั้นตอนการแปลงจะใช้ขอบเขตมากกว่าหนึ่งจุด (soft clip) เนื่องจากจุดที่มีค่าความสว่างอยู่ระหว่างสองขอบเขตจะถูกแปลงเป็นระดับสีเทาที่ความเข้มต่างๆ กันดังในรูป 2.26

รูปที่ 2.26: การแยกภาพฉากหลังออกโดยใช้ขอบเขตความสว่างมากกว่าหนึ่งจุด

การแยกภาพโดยพิจารณาจากค่าสี (Chroma key matting)

การแยกภาพฉากหลังและภาพวัตถุออกจากกันโดยใช้ค่าความสว่างเหมาะสมสำหรับการแยกภาพที่ตัววัตถุและตัวฉากมีความสว่างแตกต่างกันในระดับหนึ่ง ถ้าหากวัตถุและฉากมีความสว่างไม่แตกต่างกันมาก จะทำให้ผลลัพธ์ของภาพที่ได้ออกมาดูไม่ดีนัก ข้อจำกัดนี้แก้ไขได้ด้วยการพิจารณาจากสีของวัตถุและฉากในขั้นตอนการแยกภาพฉากหลังออกจากวัตถุ

ขั้นตอนในการแยกภาพโดยพิจารณาจากสี ทำได้โดยการกำหนดสีหลักที่จะใช้เป็นตัวกำหนดความโปร่งใสสูงสุดหรือความทึบสูงสุดเสียก่อน จุดที่กำหนดในปริภูมิสีแบบ RGB สามารถพล็อตเป็นจุดในแกนสามมิติได้ จากนั้นเราสามารถกำหนดฟังก์ชันที่คำนวณความโปร่งใสของจุดสีใดๆ ในปริภูมิสีแบบ RGB สามมิตินี้ได้ โดยคำนวณจากระยะห่างระหว่างจุดนั้นๆ กับจุดที่ใช้เป็นสีหลัก ดังในรูป 2.27



รูปที่ 2.27: แสดงระยะห่างระหว่างจุดสองจุดในปริภูมิสีแบบ RGB

เราสามารถหาระยะทางระหว่างจุดสองจุดในปริภูมิสามมิติได้โดยสมการ 2.8

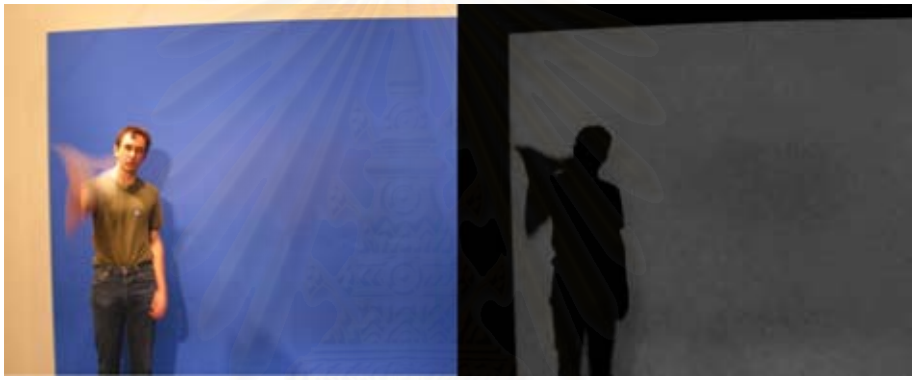
$$Distance = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \quad (2.8)$$

ในกรณีนี้ เราพิจารณาหาค่าความโปร่งใสจากปริภูมิสีของ RGB ดังนั้นจะได้ค่าของความโปร่งใสดังสมการ 2.9

$$Gray\ value = \sqrt{(R_1 - R_2)^2 + (G_1 - G_2)^2 + (B_1 - B_2)^2} \tag{2.9}$$

เราสามารถปรับอัตราส่วนเพื่อกำหนดให้จุดที่เป็นสีหลักมีค่าความโปร่งใสเป็น 0.0 และจุดที่ห่างจากสีหลักที่สุดมีค่าเป็น 1.0 และจุดที่มีระยะห่างจากสีหลักน้อยกว่า จะมีค่าความโปร่งใสไล่เรียงไปตามระยะห่าง

การแยกภาพโดยพิจารณาจากความต่างของสี (Color different matting)
การแยกภาพวัตถุออกจากฉากหลังโดยใช้การพิจารณาจากค่าสี สามารถแยกภาพฉากหลังออกจากวัตถุได้ค่อนข้างแม่นยำ แต่ยังมีปัญหาในเรื่องของบริเวณขอบภาพที่แข็งเป็นรอยหยัก จึงมีความเหมาะสมกับการแยกภาพคร่าวๆ ก่อนที่จะจัดการกับภาพบริเวณขอบด้วยวิธีอื่นที่เหมาะสมกว่า ตัวอย่างผลลัพธ์การแยกฉากหลังโดยพิจารณาความต่างของสี แสดงดังรูป 2.28



รูปที่ 2.28: ตัวอย่างภาพที่แยกฉากหลังออกมาโดยใช้การพิจารณาความต่างของสี

การแยกภาพแบบการคิดจากค่าความแตกต่างของสี เป็นวิธีการที่เหมาะสมกับการใช้ร่วมกับฉากสีฟ้า และสภาพแวดล้อมในการถ่ายทำที่สามารถควบคุมได้ โดยมีแนวคิดที่ว่าในฉากหลังที่เราจัดไว้มีค่าความแตกต่างของสีระหว่างสีเขียวกับสีแดงสูงมาก หรือความแตกต่างระหว่างสีเขียวกับสีน้ำเงินสูงมากในกรณีใช้ฉากหลังสีเขียว แต่สีของตัววัตถุจะมีความแตกต่างระหว่างสีเขียวกับสีแดง หรือสีเขียวกับสีน้ำเงินน้อยกว่า ดังนั้น เราสามารถใช้ความแตกต่างตรงจุดนี้มาช่วยในการแยกภาพตัววัตถุออกจากฉากหลังได้

วิธีการคำนวนหาค่าความโปร่งใสโดยใช้ความต่างของสีอย่างง่ายคือ

$$Raw\ matte = G - max(R, B) \tag{2.10}$$

หรือถ้าใช้ฉากหลังสีฟ้า ก็จะเป็น

$$Raw\ matte = B - max(R, G) \tag{2.11}$$

2.2.2.2 การกำจัดการเปื้อนของสี

ภาพที่ได้จากการถ่ายทำจริงจะมีปัญหาเรื่องการเปื้อนของสี (Color spilling) คือสีของฉากสะท้อนไปกระทบบนตัวของวัตถุเบื้องหน้า ทำให้บริเวณขอบวัตถุหรือตัวแสดงมีสีของฉากเข้าไปปะปนด้วย ทำให้ในขั้นตอนการรวมภาพวัตถุเบื้องหน้ากับฉากสามมิติที่สร้างขึ้นจำเป็นต้องลบเอาส่วนที่มีสีเปื้อนแบบนี้ออก

วิธีกำจัดการเปื้อนของสีมีให้เลือกใช้หลายวิธี เช่น การจำกัดสีเขียวด้วยสีแดง (green limited by red), การจำกัดสีเขียวด้วยสีน้ำเงิน (green limited by blue) ซึ่งจะเป็นการปรับลดค่าสีเขียวของจุดที่จะกำจัดการเปื้อนให้มีค่าไม่เกินค่าสีแดงหรือสีน้ำเงิน ใช้สำหรับในกรณีที่ฉากหลังเป็นสีเขียว ถ้าฉากหลังที่ใช้เป็นสีน้ำเงิน วิธีการกำจัดการเปื้อนจะใช้การจำกัดสีน้ำเงินแทนที่จะเป็นสีเขียว สำหรับในงานวิจัยนี้ใช้ฉากหลังสีเขียวจึงเลือกใช้วิธีแบบจำกัดสีเขียวด้วยค่าเฉลี่ยของสีแดงและสีน้ำเงิน (green limited by average of red and blue) ซึ่งจะปรับค่าสีของจุดที่ถูกกำจัดการเปื้อนเป็นตามสมการ 2.12

$$despilled\,green = if\ G > avg(R,B)\ then\ avg(R,B) : else\ G \tag{2.12}$$

หมายความว่า ถ้าค่าของสีเขียวมากกว่าค่าเฉลี่ยของค่าสีแดงกับสีน้ำเงิน ให้ใช้ค่าเฉลี่ยของสีแดงและสีน้ำเงินเป็นสีเขียวแทน นอกจากนั้นให้คงค่าสีเขียวไว้ตามเดิม ตัวอย่างผลการกำจัดการเปื้อนโดยใช้วิธีจำกัดสีเขียวด้วยค่าเฉลี่ยของสีแดงและสีน้ำเงิน แสดงในรูป 2.29 และ 2.30

รูปที่ 2.29: ภาพที่ยังไม่ผ่านการกำจัดการเปื้อน

2.2.2.3 การรวมภาพตัวแสดงและภาพฉาก

ในการนำภาพของตัวแสดงและภาพของฉากมารวมกันเป็นภาพผลลัพธ์สุดท้าย สามารถทำได้ตามสมการประกอบ [16] ดังในสมการ 2.13

$$C_o = \alpha C_f + (1-\alpha)C_b \tag{2.13}$$

C_o คือ สีของจุดที่เป็นผลลัพธ์

รูปที่ 2.30: ภาพที่ผ่านการกำจัดการเปื้อนแล้ว

C_f คือ สีของจุดที่เป็นภาพเบื้องหน้า

C_b คือ สีของจุดที่เป็นภาพฉาก

α คือ ค่าความโปร่งใส

ในการรวมภาพตัวแสดงและภาพฉากในแต่ละจุดจุะพิจารณาจากค่าความโปร่งใส α โดยถ้าค่าความโปร่งใสมีค่าเข้าใกล้ 1 นั่นคือภาพเบื้องหน้าจะมีความทึบและบดบังภาพของฉากที่อยู่ด้านหลัง ในทางกลับกันบริเวณที่มีค่าความโปร่งใสใกล้ 0 ภาพเบื้องหน้าจะโปร่งใส ฉากหลังจะดูชัดเจน

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 3

การออกแบบและวิธีในการพัฒนา

จากแนวคิดในงานวิจัยที่กล่าวมา สามารถนำมากำหนดเป็นรายละเอียดและความสามารถของระบบได้ดังนี้

1. ระบบที่สร้างขึ้นสามารถตรวจสอบหาพารามิเตอร์ภายนอกของกล้องขณะเคลื่อนที่ได้
2. ระบบสามารถลบฉากหลังที่ไม่ต้องการออกได้
3. ระบบสามารถสร้างฉากสามมิติขึ้นจากโมเดลที่กำหนดไว้ให้สอดคล้องกับมุมกล้องได้
4. ระบบสามารถนำภาพเคลื่อนไหวที่ถ่ายได้จากกล้องรวมเข้ากับภาพฉากหลังที่สร้างขึ้นได้ผลลัพธ์ออกมาเป็นภาพเคลื่อนไหวที่มีความสมจริง

จากมุมมองในภาพรวม ระบบที่สร้างขึ้นมีขั้นตอนการทำงานตามลำดับคือ ผู้ใช้จะจัดสถานที่สำหรับถ่ายทำขึ้น โดยให้มีฉากหลังเป็นฉากสีเขียว ที่มีลวดลายสำหรับตรวจจับตามที่กำหนด เพื่อใช้ในการหาพารามิเตอร์ภายนอกของกล้อง จากนั้นดำเนินการถ่ายทำตามปกติ โดยให้มีวัตถุหรือตัวแสดงอยู่ที่ตำแหน่งระหว่างกล้องกับฉาก และขณะถ่ายทำต้องถ่ายภาพให้ติดส่วนใดส่วนหนึ่งของลวดลายบนฉากเสมอ ซึ่งภาพเคลื่อนไหวที่ได้จากการถ่ายด้วยกล้องนี้ จะถูกนำไปประมวลผลต่อเพื่อหาพารามิเตอร์ภายนอกของกล้องโดยโปรแกรมที่ทำงานอยู่บนเครื่องคอมพิวเตอร์ที่ต่ออยู่กับกล้อง เมื่อได้พารามิเตอร์ภายนอกของกล้องซึ่งประกอบด้วยเมตริกซ์การหมุนและเวคเตอร์การเคลื่อนที่แล้ว ระบบจะนำค่าที่ได้มาจำลองมุมกล้องและสร้างฉากสามมิติ (จากโมเดลสามมิติที่เตรียมไว้แล้ว) ให้สอดคล้องกับมุมกล้องของจริงออกมาได้ หลังจากนั้นระบบจะนำภาพที่ได้ทั้งสองส่วน คือภาพที่ถ่ายได้จากกล้อง และภาพฉากสามมิติที่สร้างขึ้นมาประกอบกัน โดยลบเอาฉากหลังที่เป็นสีเขียวออกจากภาพที่ถ่ายได้จากกล้อง แล้วแทนที่ด้วยภาพฉากสามมิติ ได้เป็นผลลัพธ์สุดท้าย โดยกระบวนการทำงาน สามารถแสดงได้ดังรูป 3.1

3.1 การตรวจสอบหาตำแหน่งของกล้อง

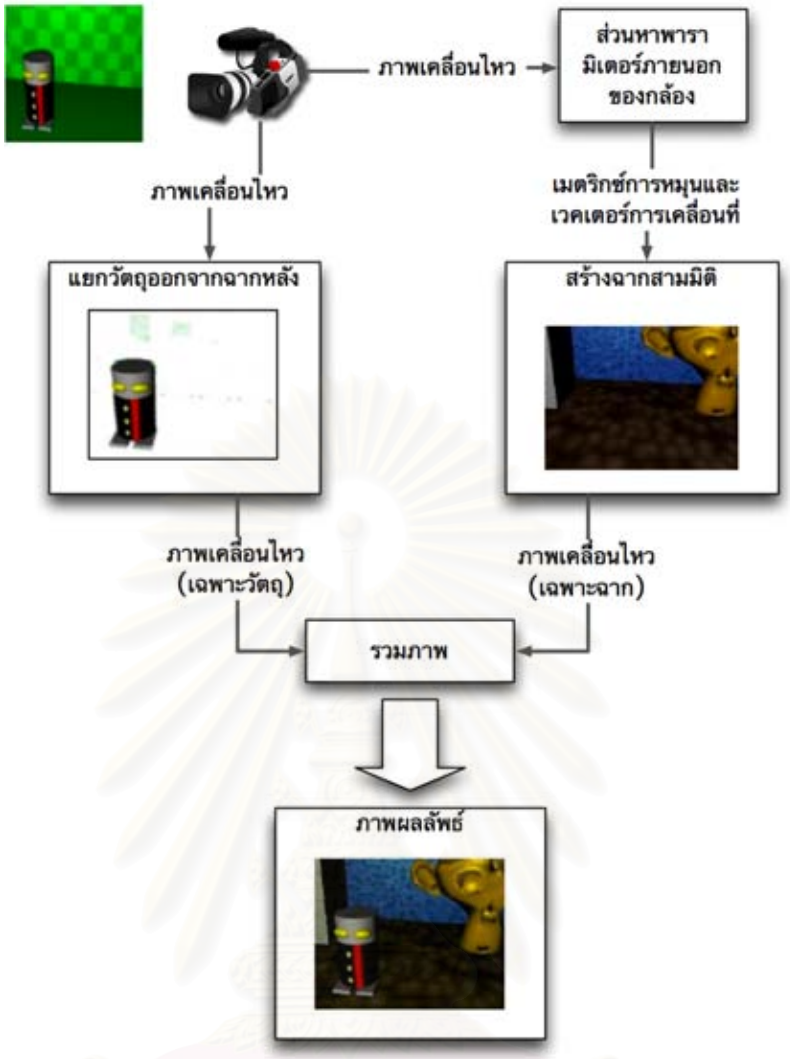
ในงานวิจัยนี้มุ่งสร้างระบบฉากเสมือนสามมิติที่ใช้อุปกรณ์เชื่อมต่อมาตรฐานทั่วไป ไม่ใช้ อุปกรณ์ที่เป็น ฮาร์ดแวร์พิเศษหรือเฉพาะทาง ดังนั้น จึงเลือกใช้วิธีการวิเคราะห์ภาพที่ถ่ายได้ ในการหาข้อมูลบอกมุมมองของกล้องโดย มีแนวคิดในการคำนวณดังนี้

ถ้าพิจารณาโดยใช้แบบจำลองของกล้องรูเข็ม (Pinhole camera) ปกติ จะได้ความสัมพันธ์ดังสมการ 3.1

$$m = A[Rt]M \tag{3.1}$$

โดยที่

R คือ เมตริกซ์การหมุน



รูปที่ 3.1: แผนภาพแสดงการทำงานโดยรวมของระบบ

t คือ เวคเตอร์การย้ายตำแหน่ง

A คือ เมตริกซ์ที่แสดงค่าพารามิเตอร์ภายใน (Intrinsic parameter) ของกล้องนั้นๆ

m คือตำแหน่งของจุดบนภาพที่ได้จากกล้อง

M คือตำแหน่งจริงของจุดในโลกสามมิติ

จากสมการ สิ่งที่เราต้องการหาคือค่าของเมตริกซ์การหมุน R และเวคเตอร์การย้ายตำแหน่ง t ดังนั้น ถ้าสามารถรู้ค่าของ m, M และ A ได้ จะสามารถแก้สมการหาเมตริกซ์การหมุน และเวคเตอร์การย้ายตำแหน่งได้ สำหรับค่า m สามารถรู้ได้จากตำแหน่งของจุดบนภาพที่ได้จากกล้อง, ค่า M ถ้าหากกำหนดกรอบอ้างอิงในโลกจริงไว้แล้ว จะสามารถหาค่า M ใดๆ ของจุดที่สนใจได้เสมอด้วยการวัดระยะจากวัตถุจริง และค่า A หรือพารามิเตอร์ภายในของกล้องนั้น เป็นค่าคงที่เฉพาะตัวสำหรับกล้องแต่ละตัว ซึ่งมีวิธีการหาได้โดยวิธีการเทียบมาตรฐานกล้อง (Camera calibration)

วิธีการเทียบมาตรฐานกล้องที่เลือกใช้ คือการตรวจสอบจากลายตารางหมากรุกตามขั้นตอนวิธีของ [17] ซึ่งจะทำการปรับตั้งก่อนการนำกล้องไปใช้งานเพียงครั้งเดียว ไม่จำเป็นต้องปรับตั้งทุกครั้งในการใช้งาน เนื่องจากพารามิเตอร์ภายในของกล้องเป็นค่าคงที่ ไม่มีการเปลี่ยนแปลง แต่ค่าพารามิเตอร์ภายในของกล้องนั้น ประกอบด้วย ทางยาวโฟกัสของเลนส์, จุดกึ่งกลางของภาพ, ขนาดของจุดที่ได้ และ ความบิดเบี้ยวของเลนส์ ซึ่งเป็นข้อจำกัดที่ทำให้เมื่อมีการเปลี่ยนกล้อง จะต้องทำการเทียบมาตรฐานกล้องใหม่ทุกครั้ง เพื่อให้ได้ค่าพารามิเตอร์ภายในที่ถูกต้องเสมอ นอกจากนี้ยังต้องกำหนดระยะโฟกัสตายตัวสำหรับกล้องประเภทที่สามารถซูมได้ด้วย เป็นเพราะว่าในการซูมเข้าออก จะทำให้ค่าความยาวโฟกัสของกล้องเปลี่ยนไป ส่งผลให้ค่าพารามิเตอร์ภายในเปลี่ยนไปด้วย

ในการใช้งานฉากสีฟ้าโดยทั่วไป ฉากหลังสามารถเลือกใช้สีอะไรก็ได้ ขึ้นอยู่กับตัวของวัตถุด้านหน้าที่สนใจว่ามี ส่วนประกอบของสีใดน้อย จะเลือกใช้สีนั้นเป็นสีของฉากหลัง ตัวอย่างเช่น สีผิวคนมีส่วนประกอบของสีเขียวและสีน้ำเงิน น้อย ดังนั้นถ้าวัตถุที่ต้องการจะถ่ายเป็นคน จะใช้ฉากหลังเป็นสีเขียวหรือน้ำเงินก็ได้ โดยพิจารณาจากวัตถุอื่นเป็นส่วน ประกอบ เช่น สีของเสื้อผ้าเครื่องแต่งกาย เป็นต้น สำหรับในงานวิจัยนี้มีการสร้างฉากจำลองขนาดเล็กเป็นต้นแบบ โดยใช้ ฉากหลังเป็นฉากสีเขียว

เพื่อการหาเมตริกซ์การหมุน จำเป็นต้องใส่ข้อมูลบางอย่างที่สนใจลงไปบนฉากหลัง เพื่อนำมาใช้ในการคำนวณ เช่นใน [18] ใช้วิธีการกำหนดจุดอ้างอิง 4 จุด ดังในรูป 3.2 ซึ่งเพียงพอต่อการคำนวณหาพารามิเตอร์ภายนอกของกล้อง

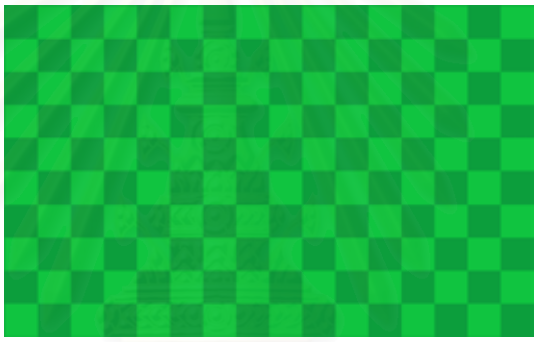
รูปที่ 3.2: การคำนวณหาพารามิเตอร์ภายนอกโดยกำหนดจุดอ้างอิง 4 จุดบนฉากหลัง

วิธีข้างต้นมีข้อจำกัดในเรื่องของจุดอ้างอิง ในกรณีที่มีวัตถุหรือตัวแสดงอยู่ในตำแหน่งที่บังจุดอ้างอิง จะทำให้ขาดข้อมูลในการคำนวณหาตำแหน่งที่ถูกต้อง จึงมีการใช้ฉากแบบที่มีจุดอ้างอิงจำนวนมากบนฉากหลัง ทำให้ยังสามารถคำนวณหาตำแหน่งที่ถูกต้องได้ถึงแม้จะมีจุดอ้างอิงบางส่วนถูกบัง เช่นใน [19] ซึ่งเป็นอุปกรณ์เชิงพาณิชย์ ดังในรูป 3.3

การใช้ลวดลายฉากหลังตามวิธีข้างต้น อาศัยการคำนวณพารามิเตอร์ภายนอกจากการลากเส้นผ่านจุดอ้างอิงที่อยู่ในแนวเดียวกันแล้วหาจุดตัด (Vanishing point) ซึ่งถ้าหากจุดตัดที่คำนวณได้มีการคลาดเคลื่อน จะทำให้พารามิเตอร์ภายนอกที่ได้มีค่าผิดเพี้ยนไปโดยเฉพาะการหมุนกล้องในแนวแกน X และแกน Y

รูปที่ 3.3: การใช้ฉากหลังที่มีลวดลายเป็นตารางสี่เหลี่ยม

ในงานวิจัยนี้เลือกใช้วิธีการใส่ข้อมูลสำหรับการคำนวณลงบนฉากที่ถ่ายทำ โดยเลือกใช้ลวดลายบนฉากเป็นลักษณะตารางหมากรุกที่มีสีเขียวเข้มสลับกับสีเขียวอ่อน ตามรูป 3.4



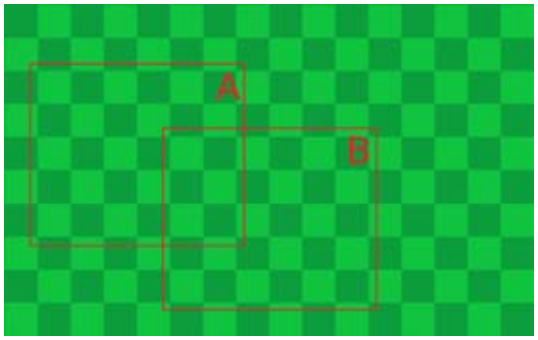
รูปที่ 3.4: แสดงลายตารางที่ใช้เป็นฉากหลัง

ด้วยวิธีการนี้ จะสามารถหาค่าของเวคเตอร์การย้ายตำแหน่งและเมตริกซ์การหมุนของกล้องในแต่ละภาพนิ่งของภาพเคลื่อนไหวได้ โดยให้จุดมุมแต่ละจุดของตารางสี่เหลี่ยมเป็นจุดที่สนใจ เทียบกับตำแหน่งจริงของตารางสี่เหลี่ยมในโลกสามมิติ

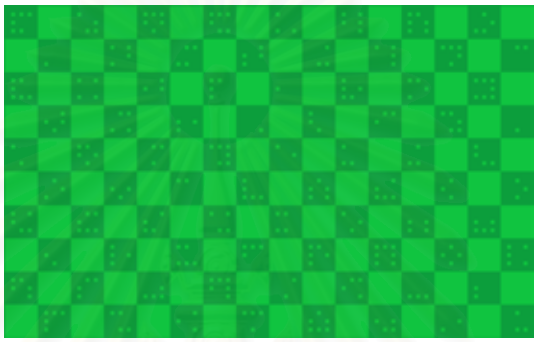
แต่การใช้วิธีนี้ อาจจะคำนวณค่าของเวคเตอร์การย้ายตำแหน่งไม่ได้ในกรณีที่กล้องจับภาพแค่เพียงบางส่วนของฉาก เช่นกรณีในรูปที่ 3.5 ถ้าภาพที่กล้องจับได้เป็นขอบเขตในพื้นที่ A เทียบกับขอบเขตของพื้นที่ B ซึ่งเป็นคนละส่วนกัน แต่ในการประมวลผล ถ้าหากมีข้อมูลเพียงเท่านี้ ไม่สามารถแยกความแตกต่างระหว่างสองพื้นที่นี้ได้

ดังนั้นเพื่อให้เกิดความแตกต่างของแต่ละตารางสี่เหลี่ยม ในงานวิจัยนี้จะใช้วิธีการเพิ่ม จุด เพื่อใช้ในการแยกความแตกต่างของแต่ละสี่เหลี่ยมออกจากกัน โดยสี่เหลี่ยมแต่ละช่องจะมีจำนวนจุดได้ 0-8 จุด ตามรูปที่ 3.6

เมื่อเทียบเคียงกับสี่เหลี่ยมที่อยู่ติดกันแล้ว ทำให้สามารถแยกแยะตำแหน่งของสี่เหลี่ยมแต่ละอันจากภาพได้



รูปที่ 3.5: แสดงปัญหาของการใช้ลายตารางเพียงอย่างเดียว



รูปที่ 3.6: แสดงการใช้จุดเพื่อสร้างความแตกต่างระหว่างตารางสี่เหลี่ยม

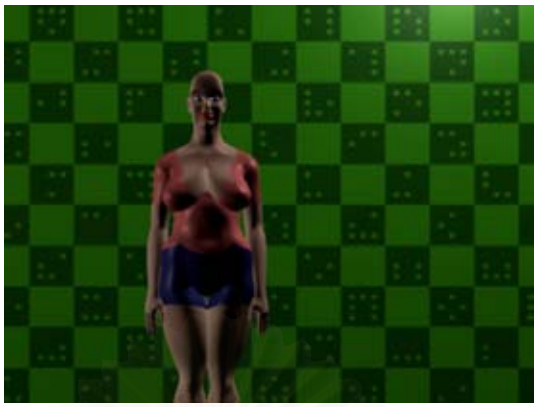
ในการเทียบตำแหน่งจุดของภาพที่ถ่ายได้จากกล้องกับตำแหน่งของจุดในสามมิติจริงๆ นั้น จะใช้ขั้นตอนวิธีชุดหนึ่งในการตรวจสอบหาตำแหน่งของจุดที่สนใจในภาพที่ถ่ายมาได้ โดยขั้นตอนวิธีนี้จะทำงานกับภาพที่เป็นขาวดำเท่านั้น จึงจำเป็นต้องเปลี่ยนจากภาพสีเป็นภาพขาวดำเสียก่อน โดยใช้การกำหนดค่าขีดแบ่ง (Threshold) [20] เพื่อเปลี่ยนให้ภาพมีเพียงสีขาวและดำเท่านั้น โดยใช้วิธีตามสมการ

$$g(x,y) = \begin{matrix} 1\ if\ f(x,y) > T \\ 0\ if\ f(x,y) \leq T \end{matrix} \tag{3.2}$$

โดยที่ค่าขีดแบ่ง T เป็นค่าคงที่ที่กำหนดขึ้น

แต่ปัญหาหนึ่งในการกำหนดค่าขีดแบ่ง เพื่อแปลงตารางให้เป็นรูปขาวดำคือ ฉากที่ใช้ในการทำงานเป็นสีเขียวกับเขียวอ่อน การกำหนดค่าขีดแบ่งโดยใช้ค่าเดียวกันทั่วทั้งภาพทำได้ลำบาก เพราะสีทั้งสองมีความใกล้เคียงกัน ถ้าเทียบความแตกต่างตรงตำแหน่งที่แสงตกกระทบสี่เหลี่ยมสีเข้ม อาจจะมีโอกาสที่สี่เหลี่ยมสีเข้มมีสีสว่างกว่าสี่เหลี่ยมสีอ่อนที่ได้รับแสงน้อย ซึ่งถ้าใช้วิธีกำหนดค่าขีดแบ่ง และใช้ค่าเดียวกันทั่วทั้งภาพ จะทำให้มีบางสี่เหลี่ยมหายไป หรือทำให้บริเวณบางมุมของสี่เหลี่ยมไม่แยกสีขาวดำชัดเจน จึงพิจารณาใช้วิธีทำ ขีดแบ่งแบบปรับตัวได้ (Adaptive threshold) ในการเปลี่ยนจากภาพตารางสีเป็นตารางขาวดำ ซึ่งใช้หลักการเปรียบเทียบความสว่างของจุดที่สนใจกับจุดอื่นโดยรอบในระยะรัศมีที่กำหนด จะให้ภาพที่มีการแบ่งสีขาวและดำได้ชัดเจนกว่า ดังรูปที่ 3.7, รูปที่ 3.8 และ รูปที่ 3.9 และเพื่อให้จุดภายในตารางมีความชัดเจน สามารถระบุตำแหน่งได้แม่นยำยิ่งขึ้น จะใช้วิธีการ ขยายส่วนที่เป็นสีดำ (dilation) หรือลดส่วนที่เป็นสีดำ (erosion) เพื่อเพิ่มความชัดเจนให้จุด ดังรูปที่ 3.10 แล้วแต่กรณีว่าลายของฉากที่ใช้เป็นจุด

สีเข้มบนสี่เหลี่ยมสีอ่อน หรือว่าจุดสีอ่อนบนสี่เหลี่ยมสีเข้ม ซึ่งจะให้ผลตรงข้ามกัน



รูปที่ 3.7: ภาพตารางต้นฉบับ

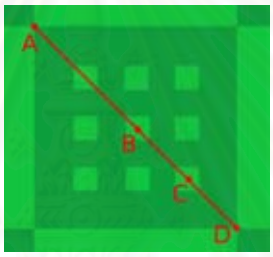
รูปที่ 3.8: การใช้ค่าขีดแบ่งเดียวทั้งภาพ

รูปที่ 3.9: การใช้ค่าขีดแบ่งแบบปรับตัวได้

จากขั้นตอนที่ผ่านมา เมื่อได้ภาพขาวดำดังรูปที่ 3.10 แล้ว สามารถนำมาหาตำแหน่งของจุดที่ได้จากกล้องได้ โดยบริเวณจุดมุมของแต่ละสี่เหลี่ยม สามารถตรวจสอบได้โดยการหาเส้นเค้าโครง (contour) ของรูปทั้งหมด และเลือกพิจารณาเฉพาะเส้นเค้าโครงที่มีลักษณะประมาณได้ใกล้เคียงกับรูปสี่เหลี่ยม และมีขนาดไม่เล็กจนเกินไป

รูปที่ 3.10: ภาพที่ใช้ขีดแบ่งแบบปรับตัวได้ และผ่านการเพิ่มขนาดจุด

สำหรับการตรวจสอบตำแหน่งของจุดที่อยู่ในตารางสี่เหลี่ยม จะใช้การคำนวณทางคณิตศาสตร์ด้วยวิธีอัตราส่วนไขว้ (Cross ratio) [21] ซึ่งมีหลักการคือ



รูปที่ 3.11: แสดงการหาอัตราส่วนไขว้

จากในรูปที่ 3.11 เป็นรูปของตำแหน่งจุดบนตารางที่มีโอกาสเป็นไปได้ พิจารณาจุด A, B, C และ D โดย A และ D เป็นจุดมุมของรูปสี่เหลี่ยม B เป็นจุดกึ่งกลางที่เส้นแทยงมุมทั้งสองของสี่เหลี่ยมตัดกัน และ C เป็นจุดจุดหนึ่งที่อยู่บนเส้นแทยงมุม AD

ภาพสี่เหลี่ยมที่ถ่ายได้จากกล้อง จะมีลักษณะต่างออกไป ไม่ได้มีลักษณะเป็นสี่เหลี่ยมผืนผ้าปกติ เนื่องจากเป็นภาพที่ฉายลงบนระนาบสองมิติ ทำให้เห็นเป็นสัดส่วนอันเกิดจากมุมมองในสามมิติ ตัวอย่างดังในรูปที่ 3.12

รูปที่ 3.12: ตัวอย่างการหาอัตราส่วนไขว้จากรูปที่ได้จากกล้อง

จาก ความ รู้เรื่อง เรขาคณิต เชิง ภาพ ฉาย (Projective geometry) ค่า ของ อัตราส่วน ไขว้ จะ ไม่

เปลี่ยนแปลง จึงสา มาถสรุปได้ว่า รูปทั้งสองรูปนี้ มีค่าของอัตราส่วนไขว้เท่ากัน และมีความสัมพันธ์ระหว่างระยะทางของจุด A, B, C และ D ดังนี้

อัตราส่วนไขว้ (ABCD) ของจุดสี่จุด A, B, C, D หาได้จาก

$$(ABCD) = CA/CB : DA/DB \tag{3.3}$$

ในการใช้งานจริง การหาอัตราส่วนไขว้ของตำแหน่งจุดบนฉากหลัง สามารถทำได้ด้วยวิธีการวัดจากฉากจริง ดังนั้น ถ้ารู้ค่าอัตราส่วนไขว้, ตำแหน่งของจุด A, B และ D จะสามารถหาตำแหน่งของจุด C ได้ด้วยการแก้สมการเชิงเส้นปกติ

จากข้อมูลตำแหน่งของจุดที่ตรวจสอบได้ เปรียบเทียบกับจุดที่มีในฉากจริง ทำให้สามารถคำนวณหาเมตริกซ์การหมุนและเวคเตอร์การย้ายตำแหน่ง ได้ตามสมการ $m = A[Rt]M$

3.2 การลบภาพฉากหลัง

การพิจารณาเลือกวิธีที่จะนำมาใช้กับงานวิจัยนี้ นอกจากจะดูจากคุณภาพของผลลัพธ์ที่ได้ ยังต้องคำนึงถึงความเหมาะสมด้านความเร็วในการทำงานด้วย เนื่องจากเวลาส่วนใหญ่ในการประมวลผลจะถูกใช้ไปในขั้นตอนของการวิเคราะห์ภาพเพื่อหาตำแหน่งของกล้อง ดังนั้นการเลือกใช้วิธีการลบภาพฉากหลังที่ทำงานได้รวดเร็ว จะทำให้ระบบโดยรวมทำงานได้เร็วขึ้นด้วย โดยหลังจากศึกษางานวิจัยและทฤษฎีที่เกี่ยวข้องแล้วพบว่าวิธีการสร้างรูปทรงแปดเหลี่ยม [6] มีข้อดีคือให้คุณภาพของผลลัพธ์ที่ดี แต่มีความซับซ้อนในการคำนวณ ทำให้ไม่สามารถประมวลผลได้ทันบนสภาพแวดล้อมที่มี เช่นเดียวกับวิธีวิเคราะห์ภาพด้วยความน่าจะเป็นแบบเบย์ [7] ที่ให้ภาพที่มีคุณภาพดีที่สุด แต่ใช้เวลาในการคำนวณมากที่สุด ทำให้ไม่สามารถนำมาใช้ได้ ส่วนการคำนวณภายใต้ปริภูมิสีแบบ YCbCr ตามวิธีของ [8] มีการคำนวณไม่ซับซ้อน สามารถประมวลผลแบบขนานได้ และให้ผลลัพธ์ที่ดี แต่จะต้องเสียเวลาในการแปลงภาพจากต้นฉบับที่เป็น RGB ให้เป็นภาพแบบ YCbCr จึงจะนำไปคำนวณได้ เมื่อคำนวณเสร็จต้องแปลงกลับเป็นภาพแบบ RGB อีกครั้งเพื่อนำไปประมวลผลขั้นตอนอื่นต่อ ทำให้วิธีนี้ไม่เหมาะสมจะนำมาใช้ในงานวิจัยนี้

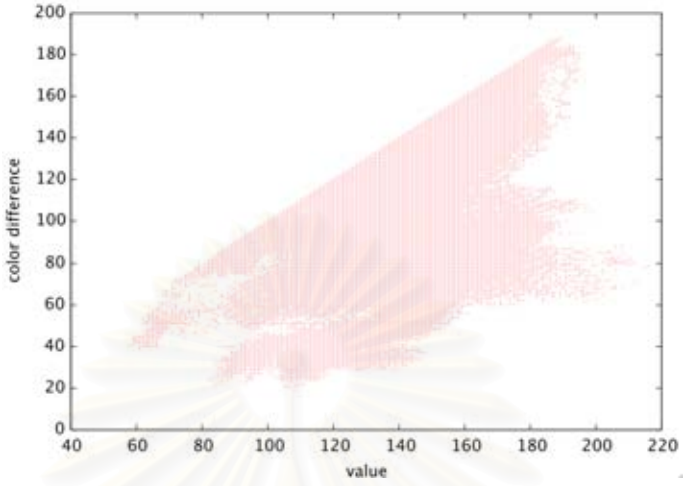
สำหรับงานวิจัยนี้เลือกใช้วิธีลบภาพฉากหลังโดยใช้การพิจารณาจากความต่างของสีซึ่งสามารถทำงานได้บนระบบสีแบบ RGB ไม่จำเป็นต้องผ่านขั้นตอนการแปลงไปมาระหว่างปริภูมิสี และมีความรวดเร็วในการทำงาน โดยมีหลักการทำงานคือ ส่วนที่มีค่าความแตกต่างระหว่างสีเขียวกับสีแดงมากๆ จะลบออกเยอะ ส่วนที่มีความต่างน้อยๆ ก็จะไม่ลบออก ตามสมการอย่างง่าย

$$raw\ matte = G - R \tag{3.4}$$

แต่ในบางกรณี เช่นภาพที่มีสีน้ำเงินบนพื้นสีเขียว วิธีนี้จะใช้งานได้ไม่ดีนักเนื่องจากไม่ได้มีการนำค่าของสีน้ำเงินมาคิดด้วย ดังนั้น ถ้าต้องการให้สามารถแยกสีน้ำเงินออกจากฉากสีเขียวได้ดียิ่งขึ้น จะปรับสมการได้เป็น

$$raw\ matte = G - max(R, B) \tag{3.5}$$

ในงานวิจัยนี้ได้ศึกษาความสัมพันธ์ของความสว่างในแต่ละจุดกับค่าความแตกต่างของสี โดยการวัดค่าจากจุดในภาพถ่ายของฉากสีเขียวจำนวน 565,060 จุด แล้วนำมาสร้างกราฟเพื่อหาความสัมพันธ์ได้ดังในรูป 3.13



รูปที่ 3.13: ความสัมพันธ์ระหว่างความสว่าง (แกนนอน) และความแตกต่างของสี (แกนตั้ง)

จะเห็นได้ว่าจุดที่มีความสว่างสูง จะมีค่าความแตกต่างของสีได้สูง และในทางกลับกันจุดที่มีความสว่างต่ำ ก็จะมีความแตกต่างของสีต่ำไปด้วย เนื่องจากค่าความสว่างในระบบสีแบบ HSV นั้นคำนวณได้จาก ค่าที่สูงที่สุดของสีแดง เขียว และน้ำเงิน ดังนั้นที่ค่าความสว่างต่ำย่อมจะได้ค่าความแตกต่างของสีต่ำไปด้วย ดังนั้นการกำหนดขอบเขตในการลบส่วนพื้นหลังของภาพโดยใช้ขีดแบ่งของความแตกต่างของค่าสีเพียงค่าเดียวสำหรับทุกจุด จะมีประสิทธิภาพไม่ดีเท่ากับการปรับเปลี่ยนระดับการลบภาพพื้นหลังตามค่าของความสว่าง โดยในงานวิจัยนี้กำหนดให้ความสว่างของจุดนั้นๆ มีผลต่อขีดแบ่งในการกำหนดขอบเขตความแตกต่างของค่าสีด้วย

ภาพที่แยกออกมาจากการคำนวณในครั้งแรกจะได้ออกมาเป็นภาพระดับสีเทา ที่มีความแตกต่างระหว่างความสว่างและความมืดไม่ชัดเจนนัก ดังนั้นเราจะต้องทำการปรับค่าอีกครั้งหนึ่งโดยเทียบอัตราส่วนให้ส่วนที่มืดที่สุดมีค่าเป็น 0 และให้ส่วนที่สว่างที่สุดมีค่าเป็น 1 เพื่อกำหนดขอบเขตค่าความโปร่งใส ดังในรูป 3.14 ภาพด้านซ้ายกำหนดให้จุดที่มีความต่างของสีน้อยแสดงเป็นสีเทาเข้มและจุดที่มีความต่างของสีมากแสดงเป็นสีเทาอ่อน โดยมีน้ำหนักต่างกันไปตามค่าความต่างของสี ส่วนในภาพทางขวามือเป็นการนำภาพทางซ้ายมาปรับระดับสี กำหนดจุดมืดที่สุดและสว่างที่สุด และคำนวณค่าของจุดที่อยู่ระหว่างมืดที่สุดและสว่างที่สุดใหม่ตามสัดส่วน

3.3 การสร้างฉากสามมิติ

งานวิจัยนี้เลือกใช้การคำนวณตำแหน่งของจุดในสามมิติ เปลี่ยนให้เป็นตำแหน่งจุดในสองมิติเพื่อแสดงผลบนระนาบจอภาพ ทำได้โดยการคำนวณทางเรขาคณิต ซึ่งมีชุดคำสั่งประมวลผลสามมิติเพื่อใช้ในการคำนวณ สามารถสร้างภาพ ย้ายตำแหน่ง หมุน ได้อย่างมีประสิทธิภาพ โดยผู้พัฒนาไม่จำเป็นต้องสนใจกับการคำนวณที่ซับซ้อน และชุดคำสั่งประเภทนี้สามารถใช้ประโยชน์จากฮาร์ดแวร์ที่มีส่วนประมวลผลด้านสามมิติ เพื่อเพิ่มประสิทธิภาพในการทำงานได้อีกด้วย

รูปที่ 3.14: ภาพที่ได้ในขั้นตอนแรก (ซ้าย) และภาพหลังจากการปรับอัตราส่วน (ขวา)

ในปัจจุบันมีชุดคำสั่งสำหรับประมวลผลสามมิติที่นิยมใช้กันบนคอมพิวเตอร์ส่วนบุคคลอยู่ ได้แก่ ไดเรกทรีดี (Direct3D) [22] สำหรับใช้งานบนระบบปฏิบัติการไมโครซอฟท์วินโดวส์เท่านั้น และโอเพนจีแอล (OpenGL) [23] ที่สามารถใช้งานได้กับหลายแพลตฟอร์มกว่า แต่เนื่องจากชุดคำสั่งทั้งสองนี้เป็นเพียงแค่ชุดคำสั่งพื้นฐานเท่านั้น เพื่อความสะดวกในการใช้งานจึงมีผู้พัฒนาชุดคำสั่งที่ทำงานอยู่บนชุดคำสั่งทั้งสองแบบข้างต้นอีกระดับหนึ่งด้วย

ในงานวิจัยนี้เลือกใช้ Irrlicht engine [24] ซึ่งเป็นไลบรารีชุดคำสั่งแบบสามมติติที่ใช้สัญญาอนุญาตแบบโอเพ่นซอร์ส เขียนขึ้นด้วยภาษา C++ โดยผู้เริ่มต้นพัฒนาคือ Nikolaus Gebhardt มีความสามารถในการทำงานได้หลายแพลตฟอร์ม ทั้งไมโครซอฟท์วินโดวส์, ลินุกส์ และแมคโอเอส โดยสามารถใช้งานระบบการสร้างภาพได้ทั้ง OpenGL, Direct3D หรือใช้ซอฟท์แวร์จำลองการทำงานก็ได้

3.4 การรวมภาพตัวแสดงและภาพฉาก

ในขั้นตอนการคำนวณหาตำแหน่งและแนวการวางตัวของกล้องของงานวิจัยนี้ใช้การเก็บรวบรวมภาพที่ถ่ายได้จากกล้องจำนวนหนึ่งก่อนแล้วจึงคำนวณและหาค่าเฉลี่ยของแต่ละเฟรมเพื่อให้ได้ภาพการเคลื่อนไหวที่ราบรื่น ไม่แกว่งหรือสะดุดจากการคำนวณที่อาจมีการผิดพลาดในบางเฟรม ดังนั้นค่าของตำแหน่งและแนวการวางตัวของกล้องจะถูกหน่วงไปเป็นจำนวนเฟรมตามที่กำหนดในการคำนวณ ทำให้ระบบต้องมีการหน่วงภาพเบื้องหน้าที่จะนำไปรวมเข้ากับภาพฉากด้วย

สมมุติว่าในการทำงานมีการหน่วงภาพเป็นจำนวน 16 เฟรม เมื่อได้ค่าของตำแหน่งและแนวการวางตัวของกล้องมาแล้ว ระบบจะสร้างภาพฉากหลังแบบ 3 มิติขึ้นจากข้อมูลที่ได้นี้ แต่ภาพเบื้องหน้าที่จะนำมารวม จะเป็นภาพของ 16 เฟรมก่อนหน้านี้ มิฉะนั้นแล้วภาพการเคลื่อนไหวของตัวแสดงและฉากหลังจะดูไม่สอดคล้องกัน

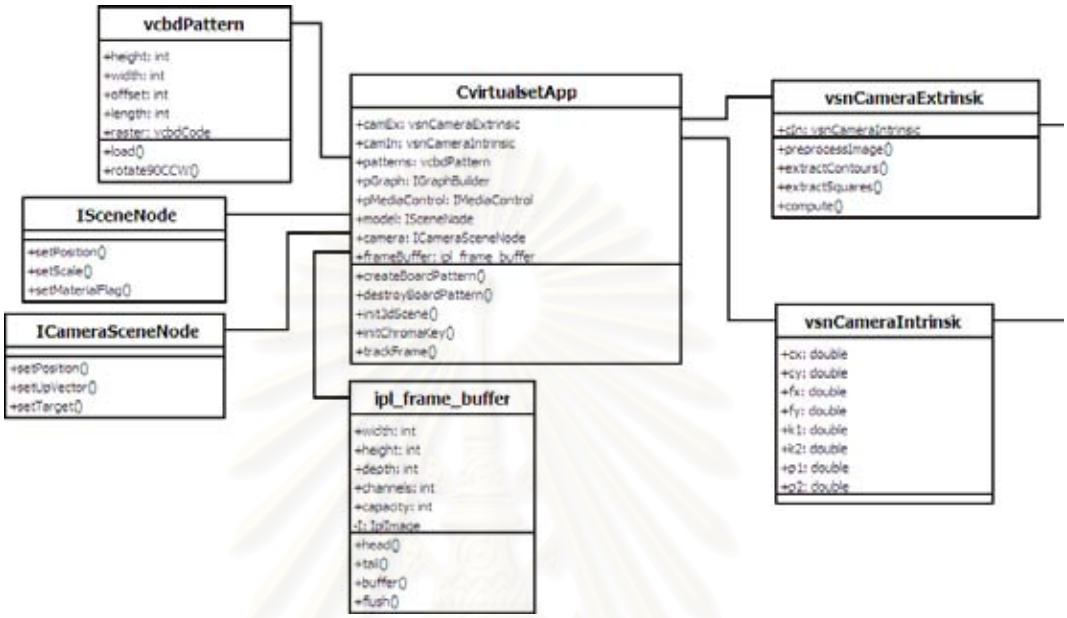
การรวมภาพโดยการคำนวณจากค่าความโปร่งใสตามสมการ 2.13 สามารถเขียนได้ใหม่เป็นตามสมการ 3.6

$$C_o = C_b + \alpha(C_f - C_b) \tag{3.6}$$

เมื่อต้องคำนึงถึงว่าการคำนวณนี้จะต้องกระทำกับทุกจุดในภาพ การจัดรูปสมการใหม่ดังในสมการ 3.6 สามารถลดจำนวนครั้งในการคูณลงได้ถึง 50 เปอร์เซนต์

3.5 การออกแบบและพัฒนาระบบ

ในการพัฒนาระบบฉากเสมือนในงานวิจัยนี้เลือกใช้ภาษา C++ บนระบบปฏิบัติการไมโครซอฟท์วินโดวส์เป็นเครื่องมือในการพัฒนา และแบ่งโครงสร้างของระบบออกเป็นส่วนๆ โดยมีความสัมพันธ์กันดังคลาสไดอะแกรม (Class diagram) ที่แสดงในรูป 3.15



รูปที่ 3.15: คลาสไดอะแกรมของระบบฉากเสมือน

- **CVirtualsetApp** เป็นส่วนประกอบหลักที่ทำหน้าที่ควบคุมการทำงานของระบบโดยรวม โดยจะเป็นตัวจัดการรับข้อมูลภาพเคลื่อนไหว การพักข้อมูลเพื่อรอการส่งให้ส่วนประมวลผล และนำภาพที่ประมวลผลเสร็จแล้วแสดงผ่านทางจอภาพ
- **ipl_frame_buffer** เป็นส่วนพักข้อมูลภาพที่ยังไม่ได้ประมวลผล โดยระบบต้องการภาพจำนวนหนึ่งก่อน จึงจะสามารถนำไปคำนวณหาพารามิเตอร์ภายนอกได้ ถ้ามีการคำนวณจากภาพจำนวนมากขึ้นจะได้ค่าพารามิเตอร์ที่ต่อเนื่องกันมากขึ้น ทำให้การเคลื่อนไหวในภาพผลลัพธ์มีความราบรื่นสวยงาม
- **vscCameraIntrinsic** เป็นส่วนที่จัดเก็บและคำนวณค่าพารามิเตอร์ภายในของกล้อง
- **vsnCameraExtrinsic** เป็นส่วนคำนวณค่าพารามิเตอร์ภายนอกซึ่งจะถูกเรียกใช้งานทุกเฟรมที่ประมวลผล
- **vcbdPattern** เป็นส่วนจัดเก็บข้อมูลลวดลายตารางที่อยู่บนฉากหลัง เพื่อนำมาเปรียบเทียบหาพารามิเตอร์ภายนอก
- **ISceneNode** เป็นคลาสของวัตถุสามมิติที่สร้างขึ้นเพื่อนำไปใช้เป็นฉาก
- **ICameraSceneNode** เป็นคลาสตัวแทนมุมมองของกล้องในโลกสามมิติ สามารถปรับเปลี่ยนมุมมองและตำแหน่งได้ตามค่าพารามิเตอร์ที่คำนวณได้

บทที่ 4

การทดลองและผลการทดลอง

เนื้อหาในบทนี้จะกล่าวถึงการทดลองใช้งานระบบฉากเสมือนที่สร้างขึ้น รวมทั้งผลที่ได้รับจากการทดลอง โดยการทดลองจะแบ่งออกเป็นหัวข้อต่างๆ ทำการทดลองที่ 2 ระดับความละเอียด คือที่ระดับ 352x240 พิกเซล และ 640x480 พิกเซล หลังจากนั้นมีการวิเคราะห์ผลการทดลองสำหรับในแต่ละหัวข้อ ส่วนสุดท้ายเป็นการสรุปผลการทดลองทั้งหมด

4.1 ระบบที่ใช้ในการทดลอง

ระบบที่ใช้ในการทดลองเป็นคอมพิวเตอร์ส่วนบุคคล มีคุณสมบัติดังนี้

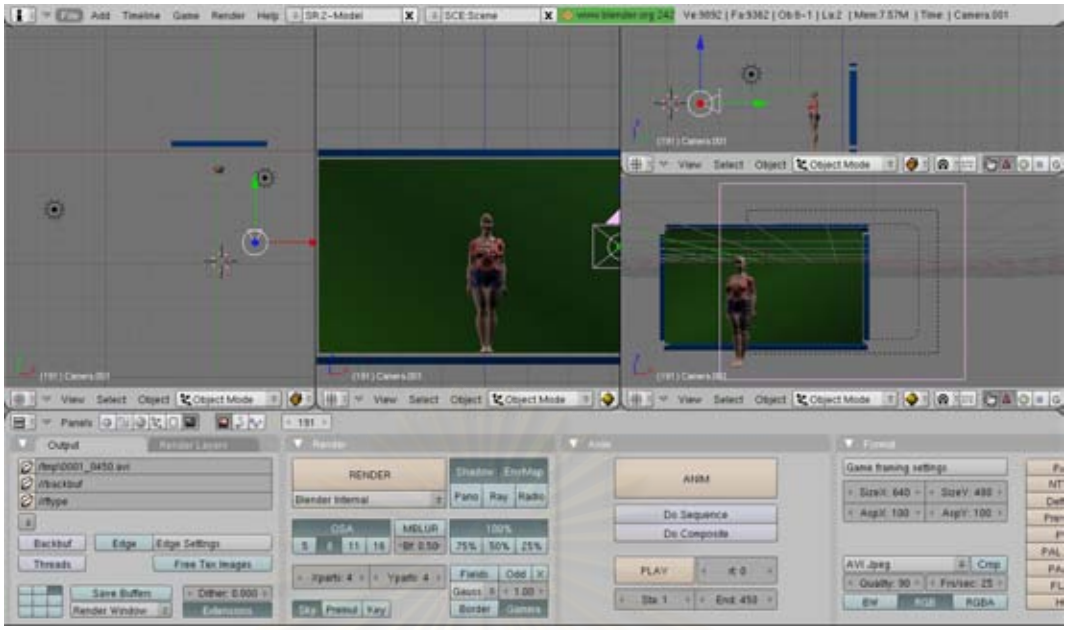
- ใช้หน่วยประมวลผล Intel Core 2 Duo ความเร็ว 1.86 GHz
- หน่วยความจำ 2 GB
- การ์ดแสดงผล nVidia GeForce 7300GT มีหน่วยความจำการแสดงผล 256 MB
- กล้องวิดีโอแบบ 3CCD ถ่ายทำที่อัตราส่วน 4:3 ต่อเชื่อมกับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรมแบบ IEEE 1394 (Firewire)

4.2 ผลการตรวจสอบหาตำแหน่งของกล้อง

เนื่องจากผลลัพธ์ที่ได้จากการตรวจสอบหาตำแหน่งของกล้อง จะได้ออกมาเป็นพารามิเตอร์ภายนอก ซึ่ง ประกอบ ด้วย เมตริกซ์ การ หมุน และ เวคเตอร์ การ เคลื่อนที่ ซึ่ง ตัวเลข เหล่านี้ ไม่ สามารถ นำมา แปลความหมาย เป็น ภาพ ผลลัพธ์ได้ โดย ตรง ดังนั้น การ ตรวจสอบ ผลลัพธ์ จะ ใช้ การ สังเกต ภาพ ผลลัพธ์ ด้วย ตาเปล่าเปรียบเทียบกับภาพจริง โดยสร้างโมเดลสามมิติจำลองให้มีลักษณะเหมือนกับฉากของจริง แล้วจึงนำภาพเคลื่อนไหวต้นฉบับมาสังเกตเปรียบเทียบกับฉากสามมิติที่สร้างขึ้น โดยในการทดลองได้แยกการทดสอบหาตำแหน่งของกล้องออกเป็นสองส่วนคือ การทดสอบกับภาพเคลื่อนไหวสามมิติที่สร้างขึ้นจากคอมพิวเตอร์ และอีกส่วนจะทดสอบกับภาพเคลื่อนไหวที่ได้จากการถ่ายโดยใช้กล้องวิดีโอของจริง

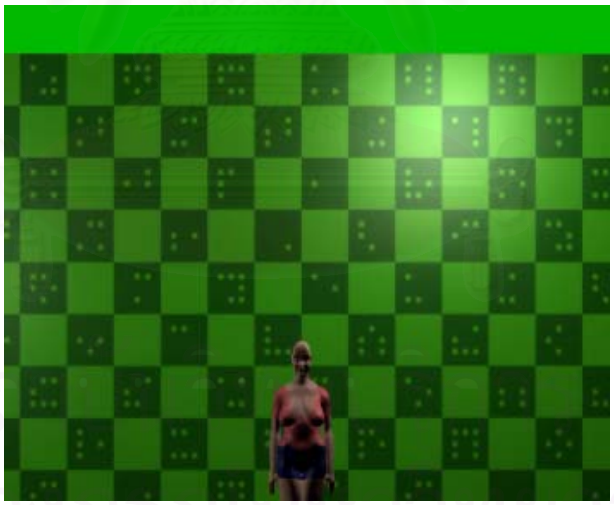
4.2.1 การทดสอบกับภาพเคลื่อนไหวสามมิติที่สร้างขึ้นจากคอมพิวเตอร์

การ ทดลอง ส่วน นี้ มี ขึ้น เพื่อ เป็น การ ตรวจสอบ ความ ถูกต้อง ของ การ ทำงาน ของ ระบบ ภายใต้สภาวะแวดล้อม ที่สามารถควบคุมได้ โดยเราสามารถควบคุมได้ทั้งตำแหน่งของกล้องเมื่อเทียบกับฉาก (เวคเตอร์การเคลื่อนที่) และแนวการวางตัวของกล้อง (เมตริกซ์การหมุน) สามารถวัดค่าออกมาได้เป็นตัวเลขที่แม่นยำ ลดปัจจัยความไม่แน่นอนอื่นๆ ที่จะมีผลกระทบให้ผลการทดลองคลาดเคลื่อน โดยในงานวิจัยนี้สร้างภาพเคลื่อนไหวสามมิติขึ้นด้วยโปรแกรม Blender (รูป 4.1) ซึ่งเป็นโปรแกรมสร้างภาพสามมิติที่มีสัญญาอนุญาตแบบโอเพ่นซอร์ส สามารถนำมาใช้งานได้โดยไม่เสียค่าใช้จ่าย เมื่อสร้างฉากจำลองวัตถุและมุมกล้องในโปรแกรมแล้วจึงเรนเดอร์ออกมาเพื่อให้ได้ภาพเคลื่อนไหว ที่จะนำไปทดสอบกับส่วนตรวจสอบหาตำแหน่งของกล้องอีกครั้งหนึ่ง



รูปที่ 4.1: การสร้างฉากสามมิติโดยใช้โปรแกรม Blender

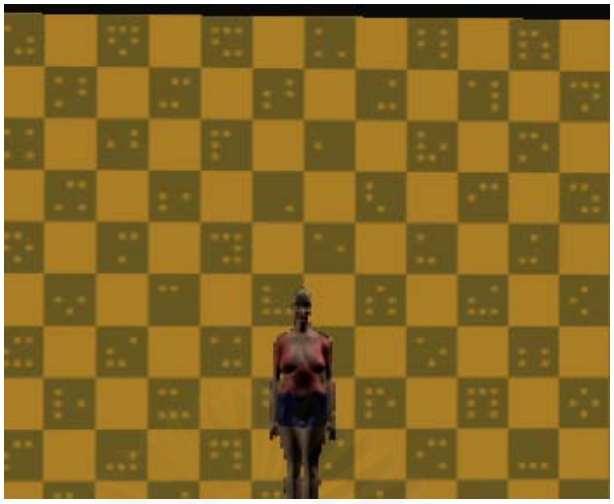
ผลการทดลองกับภาพเคลื่อนไหวที่สร้างขึ้นจากคอมพิวเตอร์ โดยการเปรียบเทียบภาพฉากที่สร้างขึ้นกับภาพฉากจริง พบว่าตำแหน่งและแนวการวางตัวของกล้องที่คำนวณได้มีความสอดคล้องกับภาพเคลื่อนไหวที่สร้างขึ้นไว้ก่อน แสดงได้ดังภาพ 4.2, 4.3 และ 4.4



รูปที่ 4.2: ภาพต้นฉบับ

4.2.2 การทดสอบกับภาพเคลื่อนไหวที่ได้จากการถ่ายทำด้วยกล้องจริง

หลังจากทดสอบระบบที่สร้างขึ้นในสภาวะแวดล้อมที่ควบคุมได้โดยใช้ภาพเคลื่อนไหวที่สร้างขึ้นเองแล้ว ในขั้นตอนถัดมาจะเป็นการทดสอบระบบกับภาพเคลื่อนไหวที่ถ่ายจากกล้องจริงๆ ซึ่งการทดสอบส่วนนี้จะมีปัจจัยภายนอกอื่นๆ เช่น ความแม่นยำในการปรับตั้งพารามิเตอร์ภายใน ความไม่สม่ำเสมอของแสง หรือจุดรบกวนภายในภาพที่ถ่ายมาได้ ซึ่งล้วนแต่มีส่วนในการทำให้ผลลัพธ์ที่ได้คลาดเคลื่อนออกไปจาก



รูปที่ 4.3: ภาพผลลัพท์ที่มีการสร้างฉากหลังขึ้นตามข้อมูลพารามิเตอร์ของกล้อง

รูปที่ 4.4: ภาพเปรียบเทียบฉากจริงและฉากที่สร้างขึ้น

ความเป็นจริง

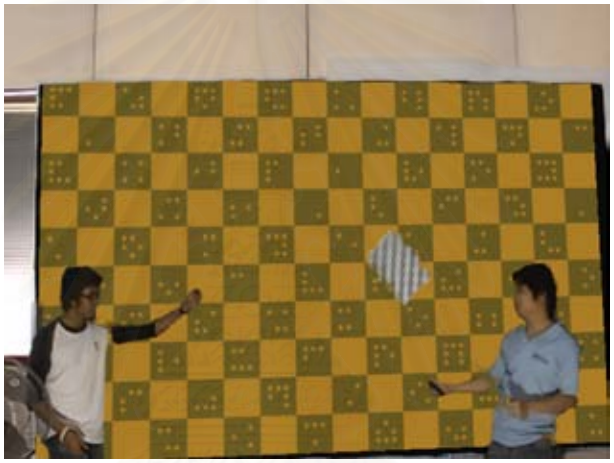
จากการทดลองกับภาพที่ถ่ายได้จากฉากจริงได้ผลลัพธ์ดังในรูป 4.5, 4.6 และ 4.7 โดยตำแหน่งและแนวการวางตัวของกล้องที่ได้ มีความสอดคล้องกับภาพเคลื่อนไหวที่ถ่ายทำได้จริง

4.3 ผลการลบภาพฉากหลังออกจากภาพที่ถ่ายได้

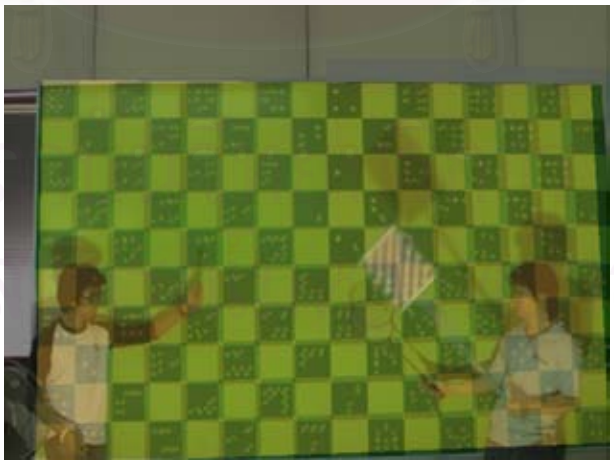
ในขั้นตอนการลบภาพฉากหลัง ระบบจะใช้ตัวแปรสองตัวคือ amount และ fuzziness ในการคำนวณระดับของการลบภาพฉากหลัง ซึ่ง amount และ fuzziness เป็นค่าที่ผู้ใช้สามารถปรับได้ ผ่านทางหน้าจอการใช้งานระบบ เพื่อให้เหมาะสมกับลักษณะของฉากและสภาพแสงในการถ่ายทำ นอกจากนี้ปัจจัยอื่นที่มีผลต่อการลบภาพฉากหลังคือค่าความสว่าง (value) ของแต่ละจุดสีในระบบปริภูมิสีแบบ HSV ซึ่งระบบจะคำนวณให้โดยอาศัยข้อมูลทางสถิติที่รวบรวมไว้ก่อนแล้ว

ระดับของการลบภาพฉากหลังหรือระดับความโปร่งใสของภาพ จะถูกกำหนดโดยค่า alpha ซึ่งจะมี

รูปที่ 4.5: ภาพต้นฉบับ



รูปที่ 4.6: ภาพผลลัพท์ที่มีการสร้างฉากหลังขึ้นตามข้อมูลพารามิเตอร์ของกล้อง

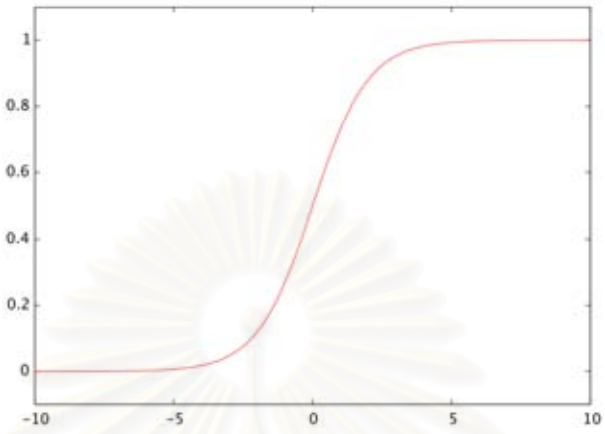


รูปที่ 4.7: ภาพเปรียบเทียบฉากจริงและฉากที่สร้างขึ้น

ค่าได้ตั้งแต่ 0-1 โดยจุดที่มีค่า alpha = 0 หมายถึงจุดนั้นๆ จะไม่ถูกลบออก (เป็นจุดทึบ) จุดที่มีค่า alpha = 1 หมายความว่าจุดนั้นจะถูกลบออก (เป็นจุดใส) และแทนที่ด้วยภาพฉากหลัง สำหรับในการคำนวณ ค่าระดับความโปร่งใส ถูกกำหนดโดยใช้สมการของฟังก์ชันซิกมอยด์ (Sigmoid function) ซึ่งมีรูปทั่วไปตาม

สมการ 4.1 และมีลักษณะของกราฟดังในรูป 4.8

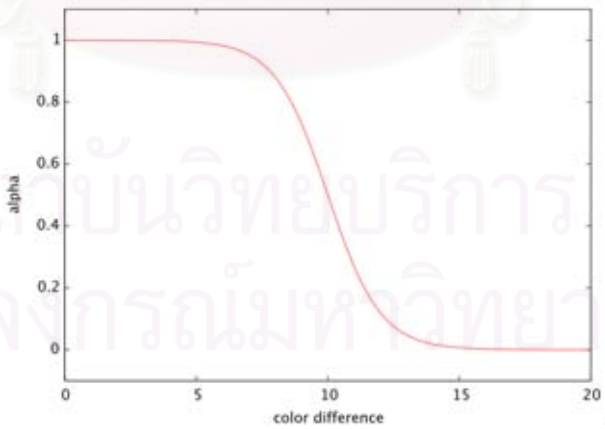
$$P(t) = \frac{1}{1 + e^{-t}} \tag{4.1}$$



รูปที่ 4.8: กราฟทั่วไปของฟังก์ชันซิกมอยด์

ในการปรับค่าความโปร่งใส ผู้ใช้สามารถปรับการเลื่อนตำแหน่งของกราฟในแนวแกน x ได้ โดยปรับที่ค่าของ amount และสามารถปรับค่าความชันของกราฟได้โดยปรับที่ค่าของ fuzziness ค่าตัวแปรที่ผู้ใช้สามารถปรับได้ทั้งสองนี้ เมื่อนำมาเพิ่มเติมในฟังก์ชันซิกมอยด์จะได้เป็นสมการสำหรับคำนวณค่าความโปร่งใสตามสมการ 4.2 ซึ่งมีกราฟลักษณะตามรูป 4.9

$$alpha = \frac{1}{1 + e^{fuzziness \times (colordiff - amount)}} \tag{4.2}$$

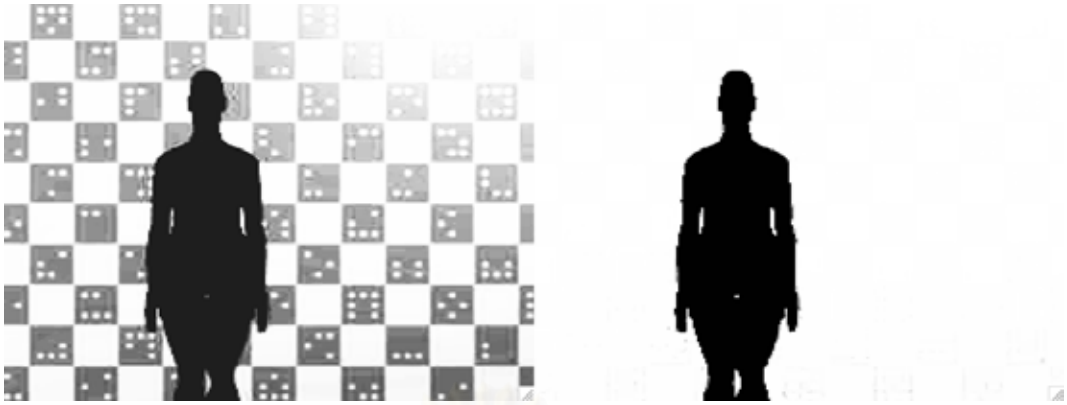


รูปที่ 4.9: กราฟของฟังก์ชันสำหรับคำนวณค่าความโปร่งใส

การกำหนดค่า amount และ fuzziness ที่แตกต่างกันจะให้ผลลัพธ์ในการลบภาพฉากหลังแตกต่างกันด้วย ดังรูป 4.10

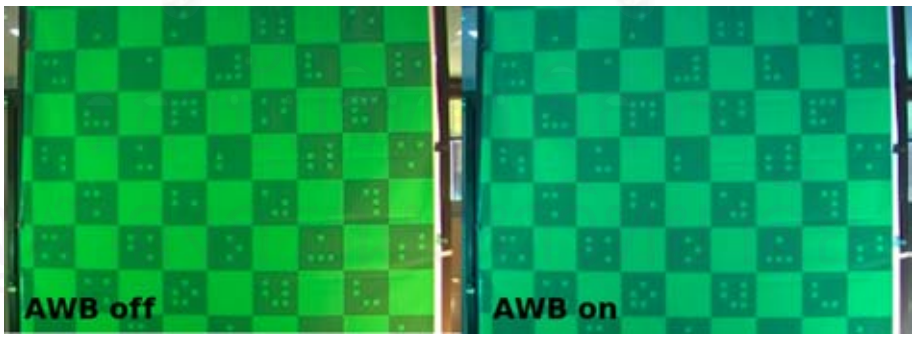
จากการทดลองถ่ายทำจริงพบว่ามีปัจจัยหลายอย่างที่ส่งผลกระทบต่อความสามารถในการลบภาพฉากหลัง โดยปัจจัยหลักประกอบด้วย

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.10: การใช้ค่า amount, fuzziness ที่ 2.0, 0.5 (ซ้าย) และ 5.0, 3.0 (ขวา)

- **สีของฉากหลัง** โดยฉากหลังที่ใช้สีเขียวซึ่งมีค่าความแตกต่างของสีระหว่างสีเขียวและสีอื่นมากๆ จะสามารถลบออกได้ง่ายโดยไม่กระทบกับส่วนอื่นของภาพ ฉากที่สร้างขึ้นด้วยโมเดลในคอมพิวเตอร์มีระบบสีแบบ RGB ซึ่งสามารถกำหนดให้มีความต่างของสีได้สะดวก แต่ในการสร้างฉากจริงจะต้องใช้การพิมพ์สีเขียวลงบนวัสดุจริง ซึ่งในการพิมพ์จะใช้ระบบสีแบบ CMYK ซึ่งในปริภูมิสีดังกล่าว สีเขียวจะมีความแตกต่างจากสีอื่นน้อยกว่าในปริภูมิสีแบบ RGB
- **วัสดุของฉากหลัง** การใช้ฉากหลังเป็นวัสดุที่ไม่สะท้อนแสง เช่น กระดาษพิมพ์สีเขียวด้วยเครื่องพิมพ์แบบฉีดหมึก จะให้ผลลัพธ์ที่ดีกว่า เนื่องจากฉากที่สะท้อนแสง เช่น ฉากที่ทำจากไวนีลจะสะท้อนแสงสีขาวออกมาซึ่งจะรบกวนกระบวนการลบภาพและการตรวจจับลวดลาย
- **การปรับสมดุลของสีขาว** กล้องวิดีโอที่ใช้ในการถ่ายทำจะต้องปิดการทำงานส่วนปรับความสมดุลสีขาวอัตโนมัติ (Auto white balance) เนื่องจากความสามารถดังกล่าวจะพยายามปรับค่าสีของภาพที่ถ่ายให้มีความสมดุลของสีขาว เมื่อถ่ายทำกับฉากที่มีสีเขียวแบบที่ใช้ในงานวิจัยนี้ ตัวกล้องจะพยายามปรับให้ภาพเกิดสมดุล ภาพฉากที่ได้จะออกมามีสีอมฟ้าดังในรูป 4.11 ซึ่งจะทำให้ผลการทดลองไม่ถูกต้อง



รูปที่ 4.11: การปิดและเปิดการทำงานส่วนปรับความสมดุลสีขาวของกล้อง

4.4 ผลการสร้างฉากสามมิติ

ในระบบที่สร้างขึ้นนี้ ต้องการโมเดลสามมิติซึ่งจะนำมาใช้เป็นฉากหลังสำหรับตัวแสดง โดยผู้ใช้สามารถเลือกใช้โมเดลของฉากสามมิติที่สร้างขึ้นโดยใช้โปรแกรมสำหรับสร้างโมเดลสามมิติทั่วไปได้

สำหรับในการทดลองนี้จะใช้ฉากสามมิติตัวอย่าง "Return to the Castle" จากเกม Quake3 ซึ่งเป็นฉากที่มีจำนวนรูปหลายเหลี่ยม 7753 ชิ้น ซึ่งเป็นความซับซ้อนในระดับที่ฮาร์ดแวร์ปัจจุบันสามารถสร้างและแสดงผลได้ในเวลาที่รวดเร็ว แต่ก็ยังดูแล้วเป็นฉากที่มีความสวยงามเพียงพอเมื่อเทียบกับฉากสามมิติที่สร้างขึ้นแบบทันกาลที่ใช้กับรายการโทรทัศน์ในปัจจุบัน รูปตัวอย่างของฉากแสดงดังในรูป 4.12

รูปที่ 4.12: ภาพฉาก Return to the Castle จากเกม Quake3

ผลการทดลองสร้างเฉพาะภาพสามมิติของฉากเพียงอย่างเดียวที่ความละเอียดต่างๆ ที่การแสดงผล 32 บิตสี ได้ความเร็วในการทำงานดังตาราง 4.1

ความละเอียด (พิกเซล)	ความเร็ว (เฟรมต่อวินาที)
352 x 240	300
640 x 480	260

ตารางที่ 4.1: ความเร็วในการสร้างฉากสามมิติที่ความละเอียดต่างๆ

จากผลการทดลองพบว่า การใช้โมเดลสามมิติที่มีความซับซ้อนของรูปหลายเหลี่ยมระดับประมาณ 7000 ชิ้น ระบบยังมีความเร็วเพียงพอที่จะสร้างภาพสามมิติขึ้นมาได้โดยไม่มีอาการกระตุก แต่อย่างไรก็ตาม การสร้างภาพสามมิติเป็นเพียงขั้นตอนหนึ่งของกระบวนการทั้งหมดเท่านั้น ระบบยังต้องประมวลผลส่วนของการหาพารามิเตอร์ภายนอกของกล้องและการลบภาพฉากหลังด้วย ซึ่งเทียบกันแล้วการสร้างภาพสามมิติจากโมเดลนี้ควรจะเป็นขั้นตอนที่ใช้เวลาน้อยที่สุด

4.5 ผลการรวมภาพที่ถ่ายได้เข้ากับฉากสามมิติที่สร้างขึ้น

การรวมภาพเข้าด้วยกันทำตามสมการ 3.6 และใช้วิธีกำจัดการเปื้อนของสี (despilling) โดยใช้วิธีจำกัดสีเขียวด้วยค่าเฉลี่ยของสีแดงและสีน้ำเงิน (green limited by average of red and blue) ผลลัพธ์ของการรวมภาพแสดงได้ดังในรูป 4.13

รูปที่ 4.13: เปรียบเทียบภาพก่อนและหลังการกำจัดการเปื้อนของสี

4.6 ผลการทดลองโดยสรุป

จากการทดลองของระบบพบว่า ระบบมีความเร็วในการทำงานอยู่ในระดับที่เพียงพอในการใช้งานทั่วไป คือมีความเร็ว 25 เฟรมต่อวินาทีที่ความละเอียด 352x240 โดยเมื่อแยกเวลาการทำงานของขั้นตอนการทำงานแต่ละส่วนออก จะได้ผลดังตาราง 4.2

ความละเอียด (พิกเซล)	การหาพารามิเตอร์ภายนอก (มิลลิวินาที)	การลบภาพฉากหลังและรวมภาพ (มิลลิวินาที)	ความเร็ว (เฟรมต่อวินาที)
352 x 240	21.418	1.338	29.117
640 x 480	48.776	3.829	17.329

ตารางที่ 4.2: สรุปเวลาที่ใช้ในการทำงานขั้นตอนต่างๆ

โดยสรุปแล้ว ระบบที่สร้างขึ้นสามารถตรวจสอบตำแหน่งของกล้องได้อย่างถูกต้องในระดับที่สายตายอมรับได้ มีความสามารถในการลบภาพฉากหลังที่ไม่ต้องการออกและรวมภาพตัวแสดงเข้ากับภาพฉากสามมิติที่สร้างขึ้นได้อย่างถูกต้อง โดยมีความเร็วในการทำงานอยู่ที่ประมาณ 25 เฟรมต่อวินาที ซึ่งอยู่ในระดับที่ยอมรับได้และเพียงพอสำหรับการใช้งานทั่วไป

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 5

สรุปการวิจัย

แนวคิดเริ่มต้นของงานวิจัยนี้ มาจากความต้องการสร้างฉากเสมือนจากอุปกรณ์คอมพิวเตอร์ที่มีอยู่ทั่วไป โดยคาดหวังคุณภาพของผลลัพธ์ที่ได้ให้มีความละเอียดในระดับหนึ่งที่พอเพียงสำหรับการใช้งานทั่วๆ ไป ซึ่งจะทำให้เกิดประโยชน์อย่างมาก โดยเฉพาะในยุคที่สื่อพลเมืองเติบโตอย่างในปัจจุบัน ผู้ใช้ทั่วไปตามบ้านสามารถผลิต และเผยแพร่สื่อผ่านทางอินเตอร์เน็ตได้อย่างอิสระ โดยมีต้นทุนที่ต่ำมากเมื่อเทียบกับสื่อในยุคก่อนหน้านี้ ด้วยเหตุนี้ ระบบที่สร้างขึ้นสามารถเอื้อประโยชน์ให้กับผู้ใช้ทั่วไป ไม่มีความจำเป็นต้องลงทุนในการซื้อฮาร์ดแวร์ราคาแพง ก็สามารถสร้างสรรค์ผลงานภาพเคลื่อนไหวที่มีคุณภาพพอเพียงได้ด้วยตนเอง

ระบบที่สร้างขึ้นในงานวิจัยนี้ สามารถใช้งานได้โดยให้ผู้ใช้ถ่ายภาพเคลื่อนไหวของตัวแสดงที่อยู่ด้านหน้าของฉากที่มีรูปแบบตรวจจับ (pattern) แล้วนำภาพที่ถ่ายได้มาคำนวณร่วมกับพารามิเตอร์ภายในที่มีการปรับตั้งไว้แล้วเพื่อหาค่าพารามิเตอร์ภายนอกของกล้องที่ใช้ถ่าย คือ เวคเตอร์การเคลื่อนที่ที่มีข้อมูลพิกัด (x, y, z) ของตำแหน่งของฉากเทียบกับกล้อง และเมตริกซ์การหมุน ซึ่งมีข้อมูลลักษณะแนวการวางตัวในสามมิติของตัวกล้อง ที่ทำให้รู้ได้ว่ากล้องที่ใช้ถ่าย มีการหันไปยังส่วนใดของฉาก จากข้อมูลเหล่านี้ ระบบสามารถนำไปสร้างฉากสามมิติที่เหมาะสมกับมุมมองที่ได้จากกล้อง และเมื่อนำมาซ้อนกับภาพเคลื่อนไหวของตัวแสดง จะได้ผลลัพธ์สุดท้ายออกมาเป็นภาพเคลื่อนไหว ที่ดูเหมือนตัวแสดงยืนอยู่ในฉากสามมิติที่เราสร้างขึ้นจริงๆ

ผลจากการทดลองใช้งานระบบพบว่า การสร้างฉากเสมือนโดยการใช้วิธีวิเคราะห์ภาพจากกล้องเดี่ยวสามารถสร้างผลลัพธ์เป็นฉากเสมือนที่มีการรวมภาพที่ถ่ายทำจากกล้องวิดีโอเข้ากับฉากสามมิติที่สร้างขึ้นได้อย่างถูกต้อง เพียงพอสำหรับการใช้งานที่ไม่ต้องการความละเอียดสูงมากนัก มีความแม่นยำอยู่ในระดับที่ดี นั่นคือในการทดลองเปรียบเทียบภาพเคลื่อนไหวที่ถ่ายได้จริง กับภาพผลลัพธ์ พบว่าการเคลื่อนไหวของฉากที่สร้างขึ้นนั้น เทียบได้ใกล้เคียงกับการเคลื่อนไหวของภาพที่ถ่ายได้จริงโดยไม่รู้สึกถึงความผิดปกติในการเคลื่อนไหว ผลลัพธ์ของการแทนที่ภาพฉากหลังก็มีคุณภาพที่ดี สามารถลบฉากหลังที่เป็นสีเขียวออกได้หมด และสามารถปรับแก้โทนสีของตัวแสดงที่ถูกแสงสะท้อนจากฉากหลังได้ ทำให้บริเวณขอบของการแทนที่ดูเรียบ ไม่เป็นรอยหยัก (jitter) โดยผู้ใช้สามารถเลือกปรับระดับของการแทนที่ให้เหมาะสมได้เอง แต่อย่างไรก็ตาม ยังมีปัญหาอื่นๆ ที่เกิดขึ้นในการใช้งาน เช่น การลบภาพฉากหลังยังทำได้ไม่ดีนักในสภาพแสงที่ไม่เพียงพอ หรือฉากสามมิติที่สร้างขึ้นยังไม่ละเอียดสมจริงเพียงพอ แต่ในอนาคตเมื่อเทคโนโลยีพัฒนาขึ้น น่าจะทำให้ระบบสามารถทำงานได้ดีขึ้น เช่น กล้องวิดีโอที่สามารถถ่ายภาพที่สภาพแสงน้อยมีจุดรบกวนน้อยลง ความเร็วของหน่วยประมวลผลที่มากขึ้น จะยิ่งที่ให้ความเร็วในการคำนวณหาพารามิเตอร์ต่างๆ สูงขึ้น และสามารถสร้างฉากสามมิติที่มีความซับซ้อนได้อย่างรวดเร็ว

5.1 ข้อเสนอแนะ

จากข้อผิดพลาดต่างๆ ที่พบในการทดลองใช้งานระบบนี้ ผู้พัฒนาจึงอยากจะมีข้อเสนอแนะเพื่อให้ระบบสามารถทำงานให้ผลลัพธ์ที่ดีขึ้น ได้แก่

1. ปัญหาเรื่องความแม่นยำในการคำนวณค่าของพารามิเตอร์ภายนอก เนื่องจากความคลาดเคลื่อนในตำแหน่งของจุดที่ตรวจจับได้ สามารถส่งผลให้การคำนวณพิกัดในโลกจริงผิดพลาดไปได้มาก ปัญหานี้อาจแก้ไขได้โดยเพิ่มความละเอียดของกล้องที่ใช้ถ่ายทำ ซึ่งจะทำให้ความคลาดเคลื่อนของการวิเคราะห์จุดในภาพลดลง แต่ก็จะเพิ่มภาระในการประมวลผลให้มากขึ้น ซึ่งถ้ามีหน่วยประมวลผลที่มีความสามารถเพียงพอก็จะสามารถใช้วิธีการนี้ได้ หรืออีกวิธีอาจใช้ฉากหลังที่มีลวดลายถี่ขึ้น ทำให้จำนวนของจุดที่ตรวจพบมีมากขึ้น ซึ่งเมื่อนำมาเฉลี่ยแล้วน่าจะได้ผลลัพธ์ที่มีความถูกต้องมากขึ้น แต่วิธีนี้ก็จะต้องใช้ความสามารถของหน่วยประมวลผลเพิ่มขึ้นเช่นกัน
2. ปัญหาเรื่องวัตถุสีเขียวบนตัวแสดง ถ้าหากว่าตัวแสดงที่อยู่หน้าฉากสวมเสื้อผ้าสีเขียว หรือมีวัตถุประกอบฉากที่เป็นสีเขียว ส่วนนี้จะถูกลบและแทนที่ด้วยฉากสามมิติที่สร้างขึ้น ซึ่งจะทำให้ผลลัพธ์ออกมาไม่ถูกต้อง ปัญหานี้เป็นข้อจำกัดของการถ่ายภาพบนฉากสีเดียวแบบนี้อยู่แล้ว มีวิธีแก้ไขได้โดยการเปลี่ยนไปใช้ฉากหลังสีฟ้าแทน และแก้ไขขั้นตอนวิธีในการลบฉากหลังและปรับแก้โทนสี ให้เหมาะสมกับฉากหลังสีฟ้า

5.2 งานวิจัยในอนาคต

งานวิจัยชิ้นนี้สามารถนำไปปรับปรุงและพัฒนาต่อยอดได้อีก เช่น พัฒนาให้มีความแม่นยำของการตรวจจับการเคลื่อนไหวที่ดีขึ้น โดยใช้อัลกอริทึมที่มีอยู่แล้วปรับแก้ตัวกรองผลลัพธ์ให้มีความราบรื่นขึ้น หรืออาจพิจารณาใช้อัลกอริทึมอื่นก็ได้

ในแง่ของการสร้างฉากสามมิติ เมื่อคอมพิวเตอร์มีความเร็วในการประมวลผลมากขึ้น สามารถเพิ่มความละเอียดของฉากสามมิติ ใช้เทกซ์เจอร์ที่มีคุณภาพสูงขึ้น หรือใส่ลูกเล่นพิเศษ เช่น ระยะชัดลึก-ชัดตื้นได้ เพื่อเพิ่มความสมจริงของฉาก อาจมีการเพิ่มความสามารถในการใส่แสงเงาให้ถูกต้องตามสภาพแสงจริงด้วย แต่ในกรณีนี้เราไม่สามารถตรวจจับตำแหน่งของแหล่งกำเนิดแสงได้จากภาพวิดีโอ จึงต้องใช้วิธีอื่นในการตรวจจับ เช่น มีเซนเซอร์ที่ใช้ในการวัดความเข้มของแสงจำนวนหนึ่งติดอยู่ที่ฉาก แล้วนำข้อมูลที่ได้มาคำนวณและสร้างตำแหน่งของแหล่งกำเนิดแสงที่ถูกต้องในฉากสามมิติ

ในแง่ของผู้ใช้ทั่วไปที่ไม่มีความต้องการใช้งานแบบทันกาล สามารถปรับปรุงระบบให้ทำงานแบบทีละส่วนได้ เมื่อไม่มีข้อจำกัดทางด้านเวลาแล้วผู้ใช้สามารถเลือกใช้วิธีการตรวจจับพารามิเตอร์ที่แม่นยำกว่าได้ หรืออาจใช้วิธีลบภาพฉากหลังด้วยการใช้ความน่าจะเป็นของเบย์ซึ่งใช้เวลานานกว่าวิธีคำนวณจากความแตกต่างของสีตามที่ใช้ในงานวิจัยนี้ แต่ผลลัพธ์ที่ได้จะมีความแม่นยำมากกว่า ส่วนของการสร้างฉากสามมิตินั้นก็จะสามารถสร้างฉากที่สมจริง มีความซับซ้อนของรูปหลายเหลี่ยมในระดับสูงได้

รายการอ้างอิง

[1] Bluescreen. Available from http://en.wikipedia.org/wiki/Bluescreen, 2006.

[2] J. Oreck. The matrix revisited [dvd]. Warner Home Video, 2001.

[3] R. Siegwart and I. Nourbakhsh. Introduction to Autonomous Mobile Robots. The MIT Press, 2004.

[4] O. Faugeras, Q. Luong, and S. Maybank. Camera Self-Calibration: Theory and Experiments. European Conference on Computer Vision (1992): 321--334.

[5] P. Vlahos. Image Modification of Motion Pictures, 1973. US Patent 3,772,465.

[6] Y. Mishima. Soft edge chroma-key generation based upon hexoctahedral color space, 1994. US Patent 5,355,174.

[7] Y. Y. Chuang, B. Curless, D. H. Salesin, and R. Szeliski. A bayesian approach to digital matting. In Proceedings of IEEE CVPR 2001, volume 2, pp. 264--271. IEEE Computer Society, December 2001.

[8] M. Ashikhmin. High quality chroma key. Available from http://www.cs.utah.edu/ michael/chroma/, 1997.

[9] C. Kim, B. You, and H. Kim. Color Segmentation Robust to Brightness Variations by Using B-spline Curve Modeling. Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on (2006): 4874--4879.

[10] J. Selan. Merging Live Video with Synthetic Imagery. PhD thesis, Cornell University, Jan, 2003.

[11] L. Shapiro and G. Stockman. Computer vision. Prentice Hall Upper Saddle River, NJ, 2001.

[12] Vector graphics. Available from http://en.wikipedia.org/wiki/Vector_graphics, 2007.

[13] Raster graphics. Available from http://en.wikipedia.org/wiki/Raster_graphics, 2007.

[14] Polygon mesh. Available from http://en.wikipedia.org/wiki/Polygon_mesh, 2006.

[15] S. Write. Digital Composition for Film and Video. Focal Press, 2002.

[16] T. Porter and T. Duff. Compositing digital images. In SIGGRAPH '84: Proceedings of the 11th annual conference on Computer graphics and interactive techniques, pp. 253--259, New York, NY, USA. ACM Press, 1984.

[17] Z. Zhang. A flexible new technique for camera calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 11(2000): 1330-1334.

[18] K. Kansy, T. Berlage, G. Schmitgen, and P. Wisskirchen. Real-Time Integration of Synthetic Computer Graphics into Live Video Scenes. Proceedings of the Conference on the Interface of Real and Virtual Worlds (1995): 93--101.

[19] A. Steinberg, Z. Livshits, I. Wilf, M. Nissim, M. Tamir, A. Sharir, and D. Aufhauser. Method and apparatus for determining the position of a TV camera for use in a virtual studio, October 16 2001. US Patent 6,304,298.

[20] Intel Corporation. Open Source Computer Vision Reference Manual, 2001.

[21] J. Cederberg. A Course in Modern Geometries. Springer, 2 edition, 1995.

[22] Technical resources for directx. Available from http://www.microsoft.com/windows/directx/techinfo, 2007.

[23] M. Woo, J. Neider, and T. Davis. OpenGL programming guide: the official guide to learning OpenGL version 1.1. (1997).

[24] N. Gebhardt. Irrlicht 3d engine. Available from http://irrlicht.sourceforge.net, 2007.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ประวัติผู้เขียนวิทยานิพนธ์

นายพิทยา สร้อยหลง เกิดเมื่อวันที่ 27 มีนาคม พ.ศ.2523 ที่จังหวัดราชบุรี สำเร็จการศึกษาระดับปริญญาวิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ จากภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2544 หลังจากนั้นได้เข้ามาศึกษาต่อในหลักสูตรวิทยาศาสตร์มหาบัณฑิต สาขาวิทยาการคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2547 ปัจจุบันทำงานอยู่ในตำแหน่งนักวิจัยและพัฒนา บริษัทโมโนเทคโนโลยี จำกัด

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย