

การติดตามการเคลื่อนที่ของกล้องด้วยภาพความลึกและภาพขาวดำ

นิรมล เรืองพยุงค์ศักดิ์¹

บทคัดย่อ

การติดตามการเคลื่อนที่ด้วยภาพ เป็นหัวข้อกำลังได้รับความสนใจทางด้านหุ่นยนต์เคลื่อนที่ เทคนิคการหาตำแหน่งด้วยภาพ สามารถทำการประมวลผลภาพที่ได้จากกล้องชนิดต่างๆ เช่น กล้อง 2 มิติ กล้อง 3 มิติ กล้องสเตอริโอ หรือใช้เซนเซอร์ที่ให้ข้อมูลความลึกระหว่างหุ่นยนต์และวัตถุ เช่น เลเซอร์สแกนเนอร์ ในงานวิจัยนี้ใช้ข้อมูลภาพที่ได้จากกล้อง PMD ซึ่งเป็นกล้อง TOF วัดเวลาในการเคลื่อนที่ของการสะท้อนกลับ ซึ่งให้ข้อมูล 2 ชนิด ได้แก่ ระยะทางระหว่างกล้องกับวัตถุ หรือภาพความลึก และภาพขาวดำ การติดตามการเคลื่อนที่ โดยใช้ข้อมูลที่ได้จากกล้อง สามารถทำได้ด้วยการซ้อนทับภาพโดยการกำหนดตำแหน่งภาพในเฟรมที่ต่อเนื่องกัน โดยกล้องถูกติดตั้งบนระบบกล้องที่เลื่อนในแนวเส้นตรง เพื่อเป็นแนวอ้างอิงเปรียบเทียบกับผลลัพธ์ที่คำนวณได้ การจับภาพในขณะที่กล้องเคลื่อนที่ แล้วนำภาพเหล่านั้นมาประมวลผล จะได้ผลลัพธ์เป็นเส้นทางการ

เคลื่อนที่ของกล้อง ในงานวิจัยนี้ได้ทำการเปรียบเทียบวิธีที่มีคุณลักษณะต่างๆ กัน ได้แก่ วิธีที่ใช้เพียงภาพขาวดำร่วมกับภาพความลึก วิธีที่ใช้เพียงภาพความลึก วิธีที่ใช้หรือไม่ใช้การตรวจจับขอบของวัตถุในภาพ การวนรอบ อัลกอริทึม Iterative Closest Point และอัลกอริทึม k-d Tree บทความนี้นำเสนอและเปรียบเทียบวิธีที่ผสมผสานคุณลักษณะที่ต่างกัน 4 วิธี อัลกอริทึมได้ถูกเขียนขึ้นและทดสอบด้วยข้อมูลจากกล้อง TOF เพียงลำพัง โดยปราศจากการใช้กล้องชนิดอื่นร่วม ซึ่งแสดงให้เห็นศักยภาพของกล้อง ผลลัพธ์ที่ได้จากวิธีที่ต่างกันและชุดข้อมูลที่ต่างกันได้ถูกเปรียบเทียบค่าความผิดพลาดในแนวแกน x, y และ z

คำสำคัญ: การติดตามการเคลื่อนที่ของกล้อง ภาพความลึก กล้องที่วัดระยะทางด้วยการสะท้อนกลับของแสง การซ้อนทับภาพโดยการกำหนดจุดซ้อนทับ

¹ อาจารย์ ภาควิชาวิศวกรรมการผลิต คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ โทรศัพท์ 02-555-2000 ต่อ 8222 อีเมล: nirramonr@kmutnb.ac.th



Ego Motion Tracking Based Depth and Intensity Image

Niramon Ruangpayoongsak¹

Abstract

Vision-based motion tracking is an interesting topic in mobile robotics. The vision-based positioning technique processes the images captured by various types of camera including a CCD camera, 3D camera, Stereo camera or TOF sensor. In the investigation, image data are obtained from PMD, a time-of-flight (TOF) camera, hence yielding two outputs with distances between the camera and the object, or depth and intensity images. Tracking motion using the camera data can be performed by image registration between consecutive frames. The camera is mounted on a translation platform as a reference for comparison of calculated output. While the camera moves, consecutive image frames are captured. These images are processed, yielding the trajectory formation. This study presents the comparison among

egomotion tracking methods with different features such as the one utilizing both intensity and depth image approach, sole depth image, method with or without edge detection, iteration loop, Iterative Closest Point and k-d Tree algorithm. Four proposed methods combine these features in different ways. Algorithms were written and tested by using only data acquired from a single TOF camera, reflecting the camera potential. Outputs of different methods and different data sets were compared for errors along the x, y and z axes respectively.

Keywords: EgoMotionTracking, DepthImage, Timeof Flight (TOF) Camera, Image Registration, Iterative Closest Point

¹ Lecturer, Department of Production Engineering, Faculty of Engineering, King Mongkut's University of Technology North Bangkok, Tel. 0-2555-2000 Ext. 8222, E-mail: nirramonr@kmutnb.ac.th

1. บทนำ

การติดตามการเคลื่อนที่ของหุ่นยนต์ เป็นการหาตำแหน่งของหุ่นยนต์ในเวลาต่อเนื่องกัน ทำให้ได้เส้นทางการเคลื่อนที่ซึ่งนำไปใช้ในการพัฒนาการวางแผนที่ และการทำงานของหุ่นยนต์แบบอัตโนมัติ เช่น การสำรวจบริเวณที่เกิดภัยพิบัติ มีสารเคมี หรือแก๊สพิษที่เป็นอันตรายต่อชีวิต เพื่อค้นหาและระบุตำแหน่งของผู้ประสบภัยเป็นต้น ด้วยเทคโนโลยีที่ก้าวไปอย่างรวดเร็ว ทำให้สามารถติดตามการเคลื่อนที่ ด้วยเทคนิคที่หลากหลาย ขึ้นอยู่กับชนิดของกล้องที่ใช้ เช่น กล้องซีซีดี กล้องสเตอริโอ กล้องรอบทิศทาง กล้อง 3 มิติ เป็นต้น อีกทั้งยังมีงานวิจัยที่ทำการผนวกภาพจากกล้องต่างชนิดเข้าด้วยกัน เพื่อคำนวณหาการเคลื่อนที่ที่มีความแม่นยำ ซึ่งอาจนำมาใช้ทดแทนการติดตามตำแหน่งด้วยเซนเซอร์ชนิดอื่นๆ

ในปัจจุบัน กล้อง TOF หรือกล้องที่วัดระยะทางด้วยเวลาการสะท้อนกลับ ได้รับความสนใจเป็นอย่างมาก เนื่องจากกล้องชนิดนี้ให้ภาพความลึก ซึ่งสามารถนำมาประยุกต์ใช้ในงานต่างๆ [1] เช่น การจับการเคลื่อนไหวของร่างกายมนุษย์ การแยกแยะวัตถุในภาพ การสร้างโมเดล 3 มิติของวัตถุหรือสถานที่ การสร้างแผนที่ 3 มิติ การหลบลสิ่งกีดขวางของหุ่นยนต์เคลื่อนที่ การตรวจจับวัตถุและแยกแยะชนิดของวัตถุ การหาตำแหน่งของวัตถุ การสร้างภาพพื้นผิวของวัตถุ การหยาบวัตถุที่รู้จัก เป็นต้น หนึ่งในกล้อง TOF ที่มีขายในท้องตลาด [2] คือ กล้อง PMD (Photonic Mixer Device) [3]

สำหรับการติดตามการเคลื่อนที่ของหุ่นยนต์ด้วยกล้อง PMD แม้ว่าจะมีงานวิจัยปรากฏอยู่บ้าง แต่งานเหล่านั้นได้มีการใช้กล้องชนิดอื่นร่วมด้วย ดังนั้นการใช้เพียงกล้อง PMD เพื่อติดตามการเคลื่อนที่ ยังคงเป็นคำถามที่น่าสนใจและเป็นทางเลือกใหม่ในการใช้ข้อมูลภาพแทนข้อมูลเซนเซอร์ ซึ่งมีศักยภาพเป็นที่ได้รับความสนใจอย่างสูง แต่เนื่องจากการประมวลผลข้อมูลภาพนั้นสามารถทำได้หลายวิธี และข้อมูลภาพยังมีสัญญาณรบกวนอยู่มากพอสมควร จึงจำเป็นต้องทำการศึกษาและทดสอบผลลัพธ์ของวิธีต่างๆ รวมถึงการพัฒนาวิธีขึ้นมาใหม่ และเปรียบเทียบกับวิธีที่มีอยู่แล้ว เพื่อศึกษาความเป็นไปได้ในการแก้ปัญหา

และเพื่อเป็นองค์ความรู้ในการดำเนินวิจัยต่อไปในอนาคต

บทความนี้นำเสนอวิธีการใช้ข้อมูลภาพความลึกและภาพขาวดำจากกล้อง PMD เพื่อการติดตามการเคลื่อนที่ของกล้องในพิกัด 3 มิติ และแสดงผลจากการติดตามการเคลื่อนที่ ด้วยวิธีที่ต่างๆ กัน ด้วยชุดข้อมูลที่ต่างกัน เพื่อหาวิธีที่เหมาะสมที่สุดในการติดตามการเคลื่อนที่โดยใช้เพียงข้อมูลจากกล้อง PMD

2. งานวิจัยและอัลกอริทึมที่เกี่ยวข้อง

2.1 งานวิจัยที่เกี่ยวข้อง

กล้อง TOF ได้ถูกนำมาใช้ควบคู่กับการพัฒนาอัลกอริทึมในการประมวลผลภาพ เพื่อการหาตำแหน่งของหุ่นยนต์และสร้างแผนที่ ซึ่งให้ผลลัพธ์ที่ต่างกันไป

2.1.1 งานวิจัยที่ใช้กล้อง PMD ร่วมกับกล้องชนิดอื่นโดยใช้วิธีการซ้อนทับหรือผนวกภาพ

Kuhnert et al. [4] ประยุกต์ใช้กล้อง PMD ร่วมกับกล้องสเตอริโอ เพื่อสร้างแผนที่ความลึกด้วยการสอบเทียบกล้องสเตอริโอ จับคู่ภาพ กรองสัญญาณภาพ และผนวกภาพสเตอริโอและภาพความลึกเข้าด้วยกัน ในบริเวณที่ความลึกในภาพไม่ต่อเนื่อง กล้อง PMD แม้ว่าจะให้ภาพวงไวดแต่จะให้ภาพความลึกที่ไม่แม่นยำเนื่องจากความละเอียดสูงของภาพต่ำ ในขณะที่ภาพสเตอริโอมีความละเอียดสูงกว่า และให้ส่วนที่เป็นขอบของวัตถุในภาพได้อย่างชัดเจน แผนที่ความลึกที่ได้จะขัดแย้งขัดต่อของกันและกันระหว่างภาพสเตอริโอกับภาพจากกล้อง PMD ได้เป็นแผนที่ความลึกที่แม่นยำ

Prusak et al. [5] ติดตั้งกล้อง PMD ร่วมกับกล้องซีซีดีที่ถ่ายภาพรอบทิศทางบนหุ่นยนต์เคลื่อนที่เพื่อใช้ในการหลบลสิ่งกีดขวางและสร้างแผนที่ 3 มิติรอบทิศทาง ด้วยวิธีการหาตำแหน่งหุ่นยนต์ SfM (Structure-from-Motion) และ Model Tracking ตำแหน่งของหุ่นยนต์ที่ได้ถูกนำมาใช้เพื่อสร้างแผนที่ ด้วยภาพที่ได้จากกล้อง PMD และวิธี Trimmed Iterative Closest Point เมื่อหุ่นยนต์เคลื่อนที่ตามเส้นทางที่กำหนดให้ ผลลัพธ์จากการสร้างแผนที่แสดงการสะสมของภาพที่เป็นแผนที่ 3 มิติที่สร้างขึ้นจากจุดเริ่มต้น

Huhle et al. [6] สร้างแผนที่ 3 มิติด้วยการซ้อนทับภาพสีที่เป็นจุด ภาพจากกล้อง PMD เป็นภาพความลึกเมื่อนำมาผนวกกับภาพจากกล้องสีซีดี ด้วยวิธี NDT (Normal Distributions Transform) ซึ่งเป็นการใช้ความน่าจะเป็นกับโมเดลของการกระจายของจุดสี นำไปสู่การซ้อนทับภาพโดยใช้สี อัลกอริทึมได้ถูกทดสอบบนหุ่นยนต์เคลื่อนที่และทำการสร้างภาพ 3 มิติ โดยอ้างอิงตำแหน่งของหุ่นยนต์จากเซนเซอร์วัดระยะทาง ผลลัพธ์ที่ได้คือภาพที่เป็นจุดสีในพิกัด 3 มิติ

2.1.2 งานวิจัยหาตำแหน่งของหุ่นยนต์ด้วยกล้อง TOF ร่วมกับกล้องชนิดอื่น

Netramai et al. [7] คำนวณเส้นทางการเคลื่อนที่ของหุ่นยนต์ ด้วยการใช้ข้อมูลที่ได้จากเซนเซอร์ 3 มิติ ได้แก่ กล้อง PMD กล้องสเตอริโอ และการผนวกกล้องทั้งสอง ในขณะที่หุ่นยนต์เคลื่อนที่ได้ทำการถ่ายภาพจุดสังเกตแบบประดิษฐ์ ผลจากการใช้กล้องสเตอริโอเพียงอย่างเดียวมีความเด่นในการหาทิศทางเนื่องจากกล้องมีความละเอียดสูง ในขณะที่กล้อง PMD ให้ผลดีในการหาการย้ายที่ และการผนวกข้อมูลจากกล้องทั้งสองเข้าด้วยกันทำให้เสริมข้อเด่นของกล้องทั้งสองเข้าด้วยกัน

May et al. [8] ประยุกต์ใช้กล้อง SwissRange SR3000 เพื่อสร้างแผนที่ 3 มิติพร้อมทั้งหาตำแหน่งของแขนกล KUKA KR 16 โดยกล้องถูกติดตั้งบนแขนกลและทำการถ่ายภาพมาร์คเกอร์ที่ใช้อ้างอิงตำแหน่ง การสอบเทียบกล้อง 3 มิติร่วมกับกล้องสีซีดี และกรองข้อมูลสามารถพัฒนาความถูกต้องแม่นยำและรวดเร็ว ร่วมกับการใช้อัลกอริทึม Iterative Closest Point เพื่อซ้อนทับภาพโดยกำหนดตำแหน่ง และปรับปรุงความแม่นยำด้วยการสอบเทียบด้วยตนเอง ซึ่งได้ผลลัพธ์แม่นยำกว่าการสอบเทียบด้วยผู้ผลิต

2.1.3 งานวิจัยที่ใช้เพียงกล้อง TOF หาตำแหน่งของหุ่นยนต์

Ye at al. [9] พัฒนารูปแบบการหาตำแหน่งและทิศทางการเคลื่อนที่ของหุ่นยนต์เพื่อใช้แทนเซนเซอร์วัดระยะทางและทิศทาง ด้วยภาพจากกล้อง SwissRange SR4000 ซึ่งประกอบด้วยภาพความลึกและภาพขาวดำที่มีความ

ละเอียดภาพเท่ากัน ทำการหาการเคลื่อนที่ของกล้องด้วยการจับคู่จุดในภาพขาวดำ ด้วยอัลกอริทึม SIFT (Scale-Invariant Feature Transform) จากภาพขาวดำ แล้วคำนวณหาการเคลื่อนที่ด้วยวิธี SVD (Singular Value Decomposition) เพื่อซ้อนทับภาพความลึกและคำนวณหาตำแหน่งการเคลื่อนที่ ผลลัพธ์ในการหาตำแหน่งของการเคลื่อนที่ในแนวแกน y มีความแม่นยำและการทำซ้ำที่ดี แต่ยังคงต้องพัฒนาต่อไปในแกนอื่นๆ

2.2 อัลกอริทึมที่เกี่ยวข้อง

2.2.1 Iterative Closest Point (ICP)

ICP [10] เป็นอัลกอริทึมที่ใช้สำหรับการซ้อนทับภาพ 3 มิติด้วยการกำหนดจุดซึ่งได้รับความนิยมใช้กันอย่างแพร่หลาย เพื่อวางเรียงภาพโมเดล 3 มิติ และยังสามารถใช้เพื่อซ้อนทับภาพ 3 มิติด้วยซึ่งจะได้ผลลัพธ์ของการเคลื่อนที่ของวัตถุในพิกัด 3 มิติ

กำหนดให้เซตของจุดในภาพโมเดล A และภาพข้อมูล B มีเซตของคู่จุดที่สอดคล้องกันเป็น (a, b) โดยที่การเคลื่อนที่ (R, t) ทำให้เกิดการซ้อนทับภาพความลึกตัวแปรการเคลื่อนที่ประกอบด้วยเมตริกซ์การหมุน R และเมตริกซ์การย้าย t ในปริภูมิ 3 มิติ และค่าความผิดพลาดที่เกิดจากการซ้อนทับ คำนวณได้จากสมการ

$$e = \frac{1}{N} \sum_{i=1}^N \|a_i - Rb_i - t\|^2 \quad (1)$$

ICP ใช้การวนรอบเพื่ออัปเดตค่าตัวแปร ภาพข้อมูลและคำนวณค่าความผิดพลาด การวนรอบจะกระทำไปจนกว่าค่าความผิดพลาดจะไม่ลดลงอีกต่อไป

2.2.2 Picky Iterative Closest Point (Picky ICP)

ICP ได้ถูกพัฒนาต่อโดย ZinBer [11] ให้มีประสิทธิภาพสูงขึ้นด้วยการเพิ่มเกณฑ์ต่างๆ ได้แก่

- การกำหนดจำนวนจุดและตำแหน่งตายตัวของจุด เพื่อควบคุมระยะเวลาที่ใช้ในการคำนวณ
- การหาจุดที่ใกล้ที่สุดด้วยอัลกอริทึม k-d Tree
- การปฏิเสธคู่ของจุดที่มีระยะห่างเกินกว่าค่าที่ตั้งไว้และคู่ของจุดที่ซ้ำ

- การตรวจจับการเคลื่อนที่เพื่อยุติการวนรอบ โดยอนุญาตให้ค่าความผิดพลาดสูงว่าในรอบก่อนได้ชั่วคราว จำนวนรอบสูงสุดคือ 30 รอบ และสามารถหยุดวนรอบได้ เมื่อค่าความผิดพลาดไม่มีการเปลี่ยนแปลงอีกต่อไป

2.2.3 อัลกอริทึมการค้นหาด้วย k-D Tree

k-D Tree [12] ย่อมาจาก k Dimensional Tree เป็นโครงสร้างข้อมูลแบ่งแยกในปริภูมิสำหรับจัดการกับจุดต่างๆ ในปริภูมิจำนวน k มิติ อัลกอริทึมนี้ เป็นโครงสร้างข้อมูลที่ประยุกต์ใช้งาน เช่น การค้นหาในบริเวณที่ใกล้ที่สุด การค้นหาช่วง เป็นต้น k-D Tree เป็นกรณีพิเศษของไบนารีเสิร์ช (Binary Search)

สำหรับการซ่อนทับภาพ k-D Tree ถูกนำมาใช้เป็นอัลกอริทึมค้นหา เพื่อหาจุดที่สอดคล้องกันมากที่สุดระหว่างภาพโมเดลกับภาพข้อมูล ซึ่งผลลัพธ์ที่ได้นำไปคำนวณหาการเคลื่อนที่ในพิกัด 3 มิติต่อไป

2.2.4 การตรวจจับขอบของวัตถุในภาพ

การตรวจจับขอบของวัตถุในภาพขาวดำ ในงานวิจัยนี้ได้เลือกใช้วิธี Canny [13] เนื่องจากวิธีนี้สามารถลดสัญญาณรบกวนในภาพได้อย่างมีประสิทธิภาพ จากตำแหน่งพิกเซลของเส้นขอบที่ได้ ถูกนำมาสร้างเป็นชุดของจุดในพิกัด 3 มิติเพื่อใช้ในการซ่อนทับภาพต่อไป

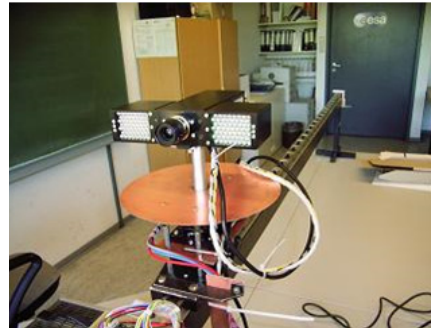
3. อุปกรณ์การทดลองและวิธีการวิจัย

3.1 ระบบกล้อง

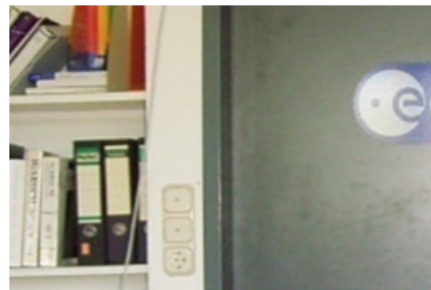
ระบบกล้องประกอบด้วยกล้อง PMD ซึ่งมีความละเอียดภาพ 3072 จุด และมีความเร็วในการถ่ายภาพ 30 เฟรมต่อวินาที และแท่นควบคุมการเคลื่อนที่ของกล้องในแนวเส้นตรง ดังแสดงในรูปที่ 1

3.2 วิธีการทดลอง

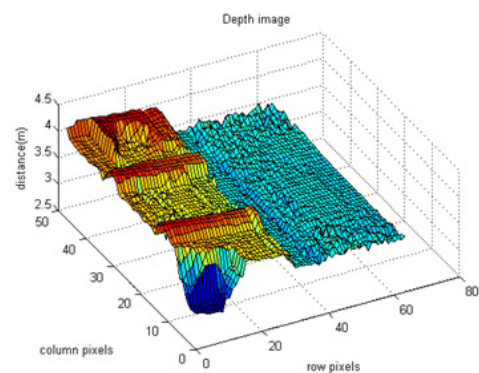
เคลื่อนที่กล้องในแนวเส้นตรงเป็นระยะทาง 180 เซนติเมตร และจับภาพทุกๆ 5 เซนติเมตร ซึ่งแนวการเคลื่อนที่ของกล้องนี้ ถูกใช้เป็นพิกัดอ้างอิง 3 มิติเพื่อคำนวณหาค่าความผิดพลาดของวิธีติดตามการเคลื่อนที่ที่ประมวลผลจากภาพ ดังที่จะอธิบายในหัวข้อ 3.4



รูปที่ 1 ระบบกล้องที่ใช้ในการทดลอง



(ก)



(ข)

รูปที่ 2 ภาพที่ถ่ายบริเวณเดียวกันด้วยกล้องซีซีดี (ก) และภาพความลึกที่ได้จากกล้อง PMD (ข)

3.3 ภาพที่ได้จากกล้อง

ภาพถ่ายที่ได้จากกล้อง PMD ในแต่ละเฟรมจะประกอบด้วย 2 ส่วน คือ ภาพความลึก และภาพขาวดำ ซึ่งมีความละเอียดภาพเท่ากัน

3.3.1 ภาพความลึก

แต่ละจุดของภาพความลึกคือ ระยะทางระหว่าง

กล้องถึงวัตถุที่สะท้อนคลื่นแสงกลับมา ภาพความลึกที่ได้จากภาพของฉากในห้องทำงาน ประกอบด้วยชั้นวางของแฟ้มต่างๆ รวมไปถึงประตูห้อง ดังแสดงในรูปที่ 2 ภาพความลึกจะไม่ปรากฏความแตกต่างของสีสันของวัตถุในภาพแต่จะแสดงเฉพาะความลึกของวัตถุที่ต่างกัน ที่ความลึกเท่ากันจะแสดงด้วยสีเดียวกัน เช่น บานประตูจะเป็นสีฟ้า ส่วนชั้นวางของที่มีแฟ้มต่างๆ จะเป็นสีส้มปนเหลือง เนื่องจากความลึกที่ต่างกัน นอกจากนี้ภาพความลึกยังสามารถแสดงเป็นภาพกลุ่มของจุดได้ด้วย ดังแสดงในรูปที่ 5

3.3.2 ภาพขาวดำ

กล้อง PMD ให้ภาพขาวดำ (Intensity Image) ซึ่งแต่ละจุดในภาพเป็นค่าเกรย์ (Gray Value) เมื่อนำค่าเกรย์มาแสดงจะได้ภาพขาวดำ ซึ่งเป็นภาพ 2 มิติ ดังรูปที่ 3 ภาพขาวดำจะมีลักษณะคล้ายภาพสีคือจะปรากฏความแตกต่างสีสันของวัตถุแต่จะไม่ปรากฏความลึกของวัตถุในภาพ

3.4 วิธีติดตามการเคลื่อนที่

การติดตามการเคลื่อนที่ด้วยการซ้อนทับภาพ 3 มิติระหว่างเฟรมที่ต่อเนื่องกัน โดยภาพที่อยู่ในเฟรมก่อนหน้าจะเรียกว่า ภาพโมเดล ส่วนภาพที่อยู่ในเฟรมปัจจุบันเรียกว่า ภาพข้อมูล ในการกำหนดตำแหน่งซ้อนทับ สามารถทำได้โดยใช้ภาพขาวดำ หรือภาพความลึก ส่วนการคำนวณหาพิิกัดการเคลื่อนที่ จะคำนวณจากภาพความลึกหรือภาพ 3 มิติ โดยวิธีที่ทำการศึกษาในงานวิจัยนี้มี 4 วิธี ดังแสดงในตารางที่ 1

ตารางที่ 1 เปรียบเทียบความแตกต่างของวิธีที่นำเสนอ

คุณลักษณะ/วิธี	Picky	IIPicky	DCFT	IIL
ภาพความลึก	ใช่	ใช่	ใช่	ใช่
ภาพขาวดำ	ไม่ใช่	ใช่	ไม่ใช่	ใช่
การวนรอบ	ใช่	ใช่	ไม่ใช่	ใช่
การตรวจจับขอบ	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
k-d Tree	ใช่	ใช่	ใช่	ไม่ใช่



รูปที่ 3 ภาพขาวดำที่ถ่ายด้วยกล้อง PMD

3.4.1 วิธี Picky

วิธี Picky เป็นวิธีที่ใช้เฉพาะข้อมูลภาพความลึกเท่านั้น ซึ่งตามกระบวนการมีการใช้อัลกอริทึม k-d Tree และการวนรอบร่วมด้วยดังที่ได้อธิบายไว้ในหัวข้อ 2.2.2 วิธีนี้จะต่างจาก Picky ICP เล็กน้อย กล่าวคือ วิธีนี้จะไม่ใช้การกำหนดจำนวนจุดและตำแหน่งตายตัวของจุด

3.4.2 วิธี IIPicky (Intensity Initialized Picky ICP)

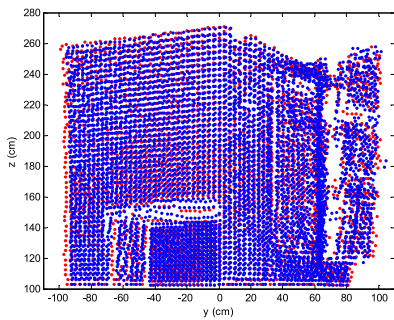
วิธีนี้มีจุดเด่นคือใช้ทั้งข้อมูลภาพความลึกและภาพขาวดำและในขณะเดียวกัน ก็ยังคงความเป็น Picky ICP ไว้โดยให้มีการกำหนดจำนวนจุดและตำแหน่งตายตัวของจุด ด้วยการตรวจจับขอบของวัตถุในภาพขาวดำเพื่อกำหนดจุดเริ่มต้น จากนั้นจึงเข้าสู่กระบวนการของ Picky ICP

3.4.3 วิธี DCFT (Direct Calculation for Tracking)

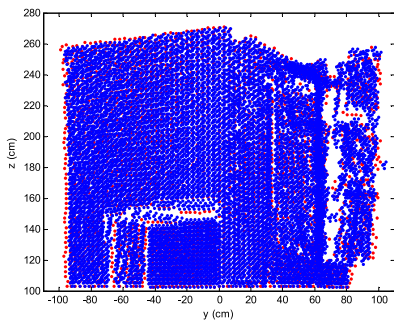
วิธี DCFT เป็นวิธีที่เร็วที่สุดในการหาตำแหน่งเคลื่อนที่ด้วยข้อมูลภาพความลึกเพียงอย่างเดียว เริ่มจากการกำหนดจุดซ้อนทับ โดยใช้อัลกอริทึม k-d Tree ค้นหาจุดที่สอดคล้องกันมากที่สุด แล้วคำนวณหาพิิกัดเคลื่อนที่ทันที โดยไม่ต้องทำการวนรอบ

3.4.4 วิธี IIL (Intensity Initialized Loop)

จุดเด่นของวิธีนี้คือไม่ใช้อัลกอริทึม k-d Tree และการตรวจจับขอบในภาพขาวดำ วิธีนี้เริ่มจากการใช้ภาพขาวดำเพื่อหาจุดที่สอดคล้องกัน โดยจับคู่ค่าเกรย์ระหว่างภาพโมเดลและภาพข้อมูลโดยไม่สนใจขอบของวัตถุในภาพ แล้วจึงทำการคำนวณหาตำแหน่งเคลื่อนที่จากข้อมูลความลึก ทำการวนรอบจนกว่าค่าความผิดพลาดจะไม่เปลี่ยนแปลงต่อไป



(ก)



(ข)

รูปที่ 4 ภาพก่อน (ก) และหลัง (ข) การซ้อนทับภาพ
ความลึกซึ่งแสดงด้วยกลุ่มของจุด (Point Cloud)

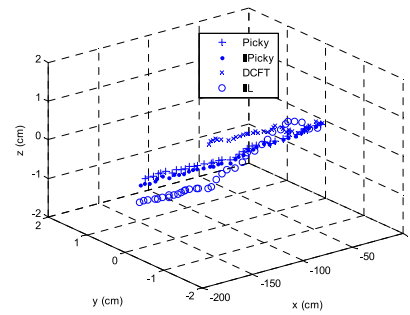
4. ผลการทดลอง

อัลกอริทึมได้ถูกเขียนขึ้นและทดสอบบนโปรแกรม MATLAB การทดลองแบ่งเป็น 2 ส่วนคือ การเปรียบเทียบผลลัพธ์ที่ได้จากวิธีที่นำเสนอ และการทดลองเปรียบเทียบผลลัพธ์ด้วยชุดข้อมูลที่ต่างกัน

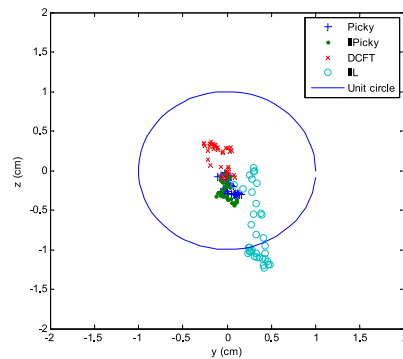
ตัวอย่างการซ้อนทับภาพ ระหว่างการเคลื่อนที่ในแนวเส้นตรงแสดงไว้ในรูปที่ 4 ภาพกลุ่มของจุด (Point Cloud) ลักษณะภาพก่อนและหลังการซ้อนทับ แสดงให้เห็นความหนาแน่นของจุดในภาพหลังการซ้อนทับที่เพิ่มขึ้นแต่ไม่เกิดการเหลื่อมของเฟรม เมื่อทำการซ้อนทับภาพหลายๆเฟรมต่อกัน จะได้เกิดการเคลื่อนที่หรือเส้นทางการเคลื่อนที่ดังที่จะกล่าวถึงในหัวข้อถัดไป

4.1 ผลการเปรียบเทียบผลลัพธ์ด้วยวิธีที่นำเสนอ

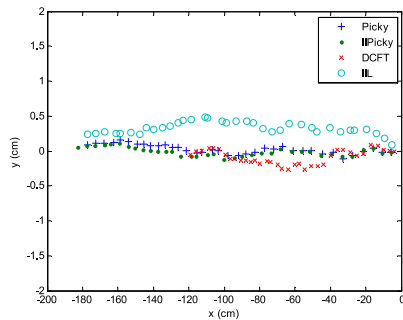
เมื่อกล้องเคลื่อนที่เป็นเส้นตรงในแนวแกน x เป็น



(ก)



(ข)



(ค)

รูปที่ 5 ผลลัพธ์ของการติดตามตำแหน่งด้วยวิธีที่นำเสนอ

ระยะทาง 180 เซนติเมตร และถ่ายภาพทุกๆ 5 เซนติเมตร จะได้ข้อมูลภาพ 36 เฟรม เมื่อทำการซ้อนทับภาพด้วยเทคนิคที่ต่างกัน จะได้ผลลัพธ์ดังแสดงในรูปที่ 5 ซึ่งประกอบด้วยเส้นทางการเคลื่อนที่ในพิกัด 3 มิติ (บน) เส้นทางการเคลื่อนที่บนระนาบ yz (กลาง) และเส้นทางการเคลื่อนที่บนระนาบ xy (ล่าง)

เส้นทางการเคลื่อนที่ในพิกัด 3 มิติ รูปที่ 5 (ก)

แสดงระยะทางจริงที่กล้องเคลื่อนที่จากจุดเริ่มต้นจนถึงจุดปลายขนานกับแกน x

รูปที่ 5 (ข) แสดงให้เห็นว่ามีเพียงวิธี IIL เท่านั้นที่มีค่าความผิดพลาดในแนวแกน z มากกว่า 1 เซนติเมตรเมื่อพิจารณาระยะทางในแนวแกน x รูปที่ 5 (ค) ผลลัพธ์ที่ได้จากวิธี IIPicky มีค่าความผิดพลาดใกล้เคียงกับวิธี Picky และ IIL ในขณะที่วิธี DFCT ให้ผลลัพธ์ที่คลาดเคลื่อนมากที่สุด

เมื่อเปรียบเทียบโดยคุณลักษณะ วิธี DCFT แม้ว่าจะใช้ข้อมูลความลึกเช่นเดียวกับวิธี Picky แต่ขาดการวนรอบ จึงทำให้ได้ผลลัพธ์ที่มีค่าความผิดพลาดสูงกว่าเมื่อเปรียบเทียบวิธีที่ใช้ภาพความลึกร่วมกับภาพขาวดำ ได้แก่ IIPicky และ IIL ให้ผลลัพธ์ที่ใกล้เคียงกันแต่วิธี IIL ใช้เวลาในการคำนวณนานกว่า เมื่อเปรียบเทียบระหว่างวิธีที่ใช้อัลกอริทึมค้นหา k -d Tree กับวิธีไม่ใช้อัลกอริทึมนี้พบว่าวิธีที่ใช้ k -d Tree จะให้ผลลัพธ์ที่แม่นยำกว่า

ตารางที่ 2 แสดงค่าต่ำสุด ค่าสูงสุด ค่าเฉลี่ย และค่าความเบี่ยงเบนของค่าความผิดพลาดในแต่ละแกน และระยะเวลาที่ใช้ในการคำนวณของแต่ละวิธี ซึ่งกราฟได้แสดงไว้แล้วในรูปที่ 5 เมื่อมองภาพรวมในทั้ง 3 แกนวิธี IIPicky มีค่าความผิดพลาดใกล้เคียงกับวิธี Picky ส่วนวิธี IIL แม้จะให้ผลลัพธ์ในแนวแกน x ที่ดีที่สุด แต่ในขณะเดียวกันก็มีค่าความผิดพลาดในแนวแกน z สูงที่สุดด้วย

ตารางที่ 2 ค่าความผิดพลาดของแต่ละวิธี ซึ่งประมวลได้จากข้อมูลชุดที่ 1

Methods	abs	min	max	avg	std
Picky 8.69 s	e_x	0.49	8.46	4.17	2.65
	e_y	0.00	0.16	0.06	0.04
	e_z	0.03	0.34	0.22	0.09
IIPicky 8.23 s	e_x	0.92	11.07	6.20	3.15
	e_y	0.00	0.13	0.05	0.03
	e_z	0.02	0.44	0.26	0.12
DCFT 6.27 s	e_x	1.19	61.08	29.12	18.30
	e_y	0.01	0.26	0.11	0.08
	e_z	0.02	0.36	0.21	0.12
IIL 28.93 s	e_x	0.19	4.41	1.94	1.32
	e_y	0.08	0.48	0.33	0.09
	e_z	0.01	1.23	0.75	0.42

ตารางที่ 3 ค่าความผิดพลาดของแต่ละวิธี ซึ่งประมวลได้จากข้อมูลชุดที่ 2

Methods	abs	min	max	avg	std
Picky	e_x	-6.95	1.16	-2.93	2.81
	e_y	0.00	1.25	0.43	0.40
	e_z	0.02	0.36	0.27	0.11
IIPicky	e_x	-7.01	0.99	-3.00	2.82
	e_y	0.00	1.24	0.42	0.40
	e_z	0.02	0.37	0.27	0.10
IIL	e_x	-1.51	6.01	0.53	1.91
	e_y	0.00	0.33	0.15	0.10
	e_z	0.02	1.26	0.74	0.46

ตารางที่ 4 ค่าความผิดพลาดของแต่ละวิธี ซึ่งประมวลได้จากข้อมูลชุดที่ 3

Methods	abs	min	max	avg	std
Picky	e_x	-6.72	4.39	-1.68	3.49
	e_y	0.00	0.20	0.08	0.06
	e_z	0.01	0.21	0.10	0.06
IIPicky	e_x	-9.77	2.20	-5.03	3.17
	e_y	0.00	0.10	0.04	0.03
	e_z	0.00	0.27	0.13	0.07
IIL	e_x	-1.50	6.92	2.24	2.37
	e_y	0.04	0.52	0.34	0.10
	e_z	0.03	1.06	0.62	0.37

วิธีที่ใช้เวลาน้อยที่สุดในการประมวลผลคือ วิธี DCFT ส่วนวิธีที่ใช้เวลามากที่สุดคือ วิธี IIL เนื่องจากไม่ได้ใช้การตรวจจับขอบของวัตถุในภาพ ทำให้ใช้เวลาในการจับคู่จุดมาก ส่วนวิธี IIPicky ใช้เวลาน้อยกว่าวิธี Picky เล็กน้อย ในขณะที่ได้ผลลัพธ์ใกล้เคียงกัน

4.2 ผลการเปรียบเทียบผลลัพธ์ที่ได้จากชุดข้อมูลที่ต่างกัน

การทดสอบกับชุดข้อมูลที่ต่างกันซึ่งได้จากการถ่ายภาพในบริเวณที่ต่างกัน เพื่อเปรียบเทียบผลกระทบของความต่างของข้อมูล ที่มีต่อผลลัพธ์ที่ได้จากวิธีต่างๆ นั้น ตารางที่ 3 และ 4 แสดงค่าความผิดพลาดที่คำนวณได้จากข้อมูลชุดที่ 2 และ 3 ซึ่งเปรียบเทียบเฉพาะวิธีที่มีประสิทธิภาพใกล้เคียงกัน ได้แก่ Picky IIPicky และ IIL สำหรับทั้ง 3 แกน ชุดข้อมูลที่ต่างกันแสดงให้เห็นว่า

แต่ละวิธีได้ผลลัพธ์ที่มีค่าความผิดพลาดในแต่ละแกนต่างกัน รวมไปถึงค่าต่ำสุด ค่าสูงสุด ค่าเฉลี่ยและค่าความเบี่ยงเบนของค่าความผิดพลาดก็ต่างกันไปด้วย

จากข้อมูลทั้ง 3 ชุด พบว่าวิธี IIL ให้ผลลัพธ์ที่ดีในแนวแกน x เนื่องจากมีค่าความเบี่ยงเบนของค่าความผิดพลาดต่ำที่สุด และค่าเฉลี่ยของค่าความผิดพลาดค่อนข้างต่ำ ในขณะที่วิธี IIPicky มีค่าเฉลี่ยของค่าความผิดพลาดสูงที่สุด

สำหรับในแนวแกน y วิธี IIPicky มีค่าความผิดพลาดต่ำที่สุดและดีกว่าวิธี Picky เล็กน้อย ในขณะที่ IIL ให้ผลลัพธ์ที่ต่ำกว่าวิธีอื่นในข้อมูลบางชุด

พบว่าวิธี IIL มีค่าความผิดพลาดสูงที่สุดในข้อมูลทั้ง 3 ชุด เฉพาะในแนวแกน z

นอกจากนี้ เมื่อพิจารณาเฉพาะพิกัดปลายทาง (พิกัดสุดท้าย) ของข้อมูลแต่ละชุดและแต่ละวิธี จะสามารถพิจารณาผลรวมของค่าความผิดพลาดของ จากผลลัพธ์ในตารางที่ 5 ดังนี้ ทั้ง 3 วิธีมีค่าพิกัดที่จุดปลายในแกน x, y และ z แตกต่างกันไม่เกิน 6.88, 0.41 และ 0.93 เซนติเมตร ตามลำดับ

จุดปลายที่ได้จากวิธี Picky และ IIPicky มีพิกัดใกล้เคียงกันมากในแนวแกน y และ z ผลต่างของพิกัดเพียง 0.19 และ 0.12 เซนติเมตร ตามลำดับ

ตารางที่ 5 ค่าพิกัดที่จุดปลายของแต่ละวิธีในแต่ละชุดข้อมูล

Scene	Axis	Picky	IIPicky	IIL
1	x	-177.28	-182.66	-177.01
	y	0.08	0.05	0.24
	z	-0.29	-0.41	-0.98
2	x	-178.84	-179.01	-173.99
	y	1.25	1.24	0.33
	z	0.33	0.33	1.26
3	x	-176.88	-183.76	-177.33
	y	0.13	-0.06	0.35
	z	-0.16	-0.16	-0.79

5. อภิปรายผลและสรุป

งานวิจัยนี้นำเสนอและเปรียบเทียบวิธี ในการติดตามการเคลื่อนที่โดยใช้ทั้งข้อมูลภาพความลึก และภาพ

ขาวดำที่ได้จากกล้อง TOF เพียงลำพัง โดยได้นำเสนอวิธี Picky IIPicky DCFT และ IIL ซึ่งแต่ละวิธีมีความแตกต่างกันทางคุณลักษณะ

ในงานวิจัยนี้ ใช้ภาพที่มีความละเอียดต่ำกว่าภาพที่ได้จากกล้อง SR4000 ในงานวิจัยของ Ye at al. [9] ซึ่งทำการประมวลผลภาพโดยใช้กล้อง TOF เพียงตัวเดียว โดยใช้ภาพขาวดำและภาพความลึกร่วมกันเพื่อหาตำแหน่งของหุ่นยนต์ ผลลัพธ์ที่ได้สอดคล้องกันคือพบว่า มีความแม่นยำในบางแกนเท่านั้น

นอกจากนี้จากผลการทดลองพบว่า วิธีที่ใช้การวนรอบ ให้ผลลัพธ์ที่แม่นยำกว่าวิธีที่ไม่ใช้การวนรอบ การใช้ อัลกอริทึม k-d Tree ร่วมกับภาพความลึก หรือใช้ภาพทั้งสองแบบ ให้ผลลัพธ์ที่ใกล้เคียงกันมาก แต่หากเพิ่มจากตรวจจับขอบในภาพขาวดำ จะทำให้เวลาในการประมวลผลลดลงประมาณ 5 เปอร์เซ็นต์

การใช้ทั้งภาพขาวดำและภาพความลึก ร่วมกับการตรวจจับขอบวัตถุในภาพและใช้อัลกอริทึม k-d Tree ช่วยลดเวลาการประมวลผลได้ประมาณ 3.5 เท่า ในขณะที่ผลลัพธ์ที่ได้เทียบเท่าหรือแม่นยำขึ้น

เมื่อทดสอบด้วยชุดข้อมูลที่ต่างกัน พบว่า ข้อมูลที่เปลี่ยนแปลงไป แม้จะทำให้ช่วงของค่าความผิดพลาด และค่าความผิดพลาดในแต่ละแกนแตกต่างกันไป แต่ค่าความผิดพลาดในแนวแกน x จะมีขนาดใหญ่กว่าในแกนอื่นๆ เสมอ แม้ว่าวิธี DFCT จะใช้เวลาในการประมวลผลสั้นที่สุด แต่ก็ให้ค่าความผิดพลาดสูงที่สุดด้วย ส่วนวิธี IIL ให้ค่าความผิดพลาดที่ต่ำในแนวแกน x แต่หากมองภาพโดยรวมของผลลัพธ์ทั้ง 3 แกน ค่าพิกัดที่จุดปลายแล้ว วิธี Picky และ IIPicky มีประสิทธิภาพที่เหนือกว่าวิธี IIL ทั้งผลลัพธ์และระยะเวลาในการประมวลผล และแม้ว่า IIPicky กับ Picky จะมีประสิทธิภาพที่ใกล้เคียงกันมาก แต่หากต้องการประหยัดระยะเวลาในการคำนวณแล้ว IIPicky เป็นตัวเลือกที่ดีที่สุด

6. กิตติกรรมประกาศ

ระบบกล้องที่ใช้ในงานวิจัยนี้ เป็นส่วนหนึ่งของโครงการวิจัยภายใต้การดูแลของ Prof.Dr.-Ing. Rudolf Schwarte ผู้คิดค้นและประดิษฐ์กล้อง PMD และก่อตั้ง



บริษัท PMD Technologies GmbH นอกจากนี้ผู้วิจัย
ขอขอบคุณ Steven Michael ผู้พัฒนาอัลกอริทึม k-D Tree
ด้วย MATLAB

เอกสารอ้างอิง

- [1] S. Foix, G. Aleny`a, and C. Torras, "Exploitation of Time-of-Flight (ToF) Cameras," Institut de Rob`otica i Inform`atica Industrial (IRI), Barcelona, Spain, IRI-TR-10-07, 2007.
- [2] S. Foix, G. Aleny`a, and C. Torras, "Lock-in Time-of-Flight (ToF) Cameras: A Survey," *IEEE Sensors J.*, vol. 11, no. 3, Mar. 2011.
- [3] R. Schwarte, H. Heinol, B. Buxbaum, T. Ringbeck, Z. Xu, and K. Hartmann, "Principles of 3-D Imaging Techniques," *Computer Vision and Applications*, vol. 1, Sensors and Imaging, Academic Press, pp. 374-393, 1999.
- [4] K. D. Kuhnert and M. Stommel, "Fusion of stereo-camera and PMD-camera data for realtime suited precise 3D environment reconstruction," In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1-12, pp. 4780-4785, Beijing, Oct. 2006.
- [5] A. Prusak, O. Melnychuk, H. Roth, I. Schiller, and R. Koch, "Pose estimation and map building with a time-of-flight camera for robot navigation," *International Journal on Int. Syst. Technology Applications*, vol. 5(3-4), pp. 355-364, 2008.
- [6] B. Huhle, M. Magnusson, W. Strasser, and A. J. Lilienthal, "Registration of colored 3D point clouds with a kernel-based extension to the normal distributions transform," In *Proc. IEEE International Conference on Robotics and Automation*, vol. 1-9, pp. 4025-4030, Pasadena, May 2008.
- [7] C. Netramai, M. Oleksandr, C. Joochim, and H. Roth, "Motion estimation of a mobile robot using different types of 3D sensors," In *Proc. 4th International Conference on Autonomic and Autonomous Systems*, pp. 148-153, Gosier, Mar. 2008.
- [8] S. May, D. Droschel, D. Holz, S. Fuchs, E. Malis, A. Nu`chter, J. Hertzberg, "Three-Dimensional Mapping with Time-of-Flight Cameras," *Journal of Field Robotics*, vol. 26(11-12), pp. 934-965, 2009.
- [9] C. Ye, M. Bruch, "A Visual Odometry Method Based on the SwissRanger SR4000," *Unmanned Systems Technology XII, Proc. of SPIE*, vol. 7692, 76921I, 2010.
- [10] P. J. Besl and N.D. McKay, "A Method for Registration of 3-D Shapes," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, 1992.
- [11] T. Zinßer, J. Schmidt, H. Niemann, "Refined ICP Algorithm for Robust 3-D Correspondence Estimation," *IEEE international Conference on Image Processing*, 2003, pp. 695-96.
- [12] A. Moore, "Efficient Memory based Learning for Robot Control," Ph.D. dissertation, Computer Laboratory, University of Cambridge, Cambridge, UK, 1990.
- [13] J. Canny, "A Computational Approach to Edge Detection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, 1986, pp. 679-698.