

หุ่นยนต์เคลื่อนที่ตามวัตถุเป้าหมายที่กำหนดโดยใช้การประมวลผลภาพ

A Designed Object Tracked by Mobile Robot using Image Processing

วณพันธ์ วิทยวุฒิ
Wanapun Waiyawut

บทคัดย่อ

ปัจจุบันมีการนำหุ่นยนต์มาใช้ในงานต่างๆ อย่างแพร่หลาย หุ่นยนต์ที่เคลื่อนที่ได้สามารถนำมาช่วยงานได้ในหลายรูปแบบ เช่นการใช้เพื่อเข้าไปในพื้นที่เสี่ยงภัย ที่มีสารเคมีเป็นพิษ เข้าไปในบริเวณที่ประสบภัย หรือบริเวณที่มีระเบิดหรือ กัมระเบิด การควบคุมการเคลื่อนที่ได้หลายแบบ บทความนี้จะนำเสนอหุ่นยนต์ที่เคลื่อนที่ได้โดยใช้ล้อ โดยเคลื่อนที่ไปตามภาพที่กำหนด โดยการใช้กล้องจับภาพเข้ามา นำมาประมวลผล แล้วเปรียบเทียบกับเป็นภาพที่กำหนดไว้หรือไม่ โดยใช้เครือข่ายแบบนิวรอล ถ้าใช่ก็จะเคลื่อนที่เข้าหา การเคลื่อนที่จะถูกกำหนดให้อยู่ในแนวราบ ถ้าภาพที่กำหนดไม่อยู่ตำแหน่งตรงกลาง หุ่นยนต์จะเคลื่อนที่ไปทางซ้ายหรือขวา เพื่อให้ภาพอยู่ในแนวกลาง ในบทความนี้จะนำเสนอเฉพาะส่วนของการประมวลผลภาพ

ABSTRACT

In present, there are widely use robot in many fields. Mobile robot can help us in several task, such as moving to toxic area, risk area or explosion area, mine. Movement control can be used in many ways. This paper proposed mobile robot that move by wheels. It move to previous defined picture. By using small camera process an images and check it that is the previous one by using neural network. If it is, robot will move toward to that picture. The movement is in flat plane. If that defined picture is not in center, robot will move to left or right depend on its position until it is on center. This paper proposed image processing only.

Keywords: Image processing, Mobile robot, neural network.

Email: yunghub@gmail.com

คำนำ

การที่หุ่นยนต์จะมองเห็นได้จะมองผ่านทางกล้องดิจิทัล และจะต้องมีความสามารถในการจดจำและจำแนกส่วนต่างๆ ของวัตถุที่อยู่ในภาพ นั่นคือการจำแนกได้ (Recognition) ของวัตถุ การจำแนกวัตถุ ที่อยู่ในภาพ เป็นงานที่มีความซับซ้อนงานหนึ่ง และมีหลายๆ ขั้นตอน ตัวอย่าง[1] เช่น ขั้นตอนการรับภาพ (Image Acquisition) ขั้นตอนเตรียมการประมวลผล (Preprocessing) ขั้นตอนการแยกลักษณะสำคัญ (Feature Extraction) การเก็บข้อมูล (Associative Storage) ฐานความรู้ (Knowledge Base) และขั้นสุดท้ายคือ การจำแนกได้ (Recognition)

ขั้นตอนการแยกลักษณะสำคัญ มีหลายๆ วิธีที่มีการนำมาใช้ เช่น Moment invariants [2] ที่ใช้กับระบบ Palmprint acquisition ที่ใช้ในการยืนยันตัวตนทางชีวภาพ (Biometrics-based personal identification) ที่อ่านภาพที่มีความละเอียดต่ำขนาด 75 จุดต่อกัน และใช้กับฐานข้อมูลขนาดเล็ก หรือจากแบบจำลอง Fuzzy and neural system ของ Gupta [3] ที่ใช้ปัญญาประดิษฐ์ (Artificial Intelligence) ช่วยในการจำแนกวัตถุ ที่อยู่ในภาพ

จากงานที่มีการวิจัยไว้แล้วมีความซับซ้อน ยากต่อการนำมาประยุกต์ใช้งาน โดยเฉพาะการให้มีการทำงานแบบเวลาจริง ในการทดลองใช้การประมวลผลบนเครื่องคอมพิวเตอร์ส่วนบุคคล

อุปกรณ์และวิธีการ

ใช้กล้อง digital ขนาดเล็กที่ติดต่อกับ เครื่องคอมพิวเตอร์ผ่านทางจุดเชื่อมต่อแบบ USB ขึ้นต้น ยังไม่ได้ติดตั้งอยู่บนหุ่นยนต์ แต่ใช้การกำหนดรูปภาพเป้าหมายแบบง่ายๆ เป็นรูปพื้นฐาน สามเหลี่ยม สีเหลี่ยม และวงกลม ตั้งระยะที่แตกต่างกัน เพื่อเก็บข้อมูลที่ใช้แทนรูปภาพในแต่ละภาพ โดยมีการประมวลผลภาพ และแยกผลของรูปภาพที่จะถูกกำหนดให้หุ่นยนต์จดจำรูปภาพแต่ละแบบ

วิธีแยกลักษณะสำคัญจะใช้วิธีอย่างง่าย ไม่ต้องใช้การประมวลผลที่ซับซ้อน โดยการแยกภาพออกมาแต่ละส่วน โดยการทดลองขึ้นต้นจะกำหนดภาพเป็นสีขาวบนพื้นสีดำ โดยกำหนดกรอบ ล้อมรอบรูปที่ต้องการ และหาอัตราส่วนของภาพสีขาวบนพื้นสีดำ ซึ่งจะทำให้ได้ค่าเฉพาะของรูปภาพแต่ละแบบ

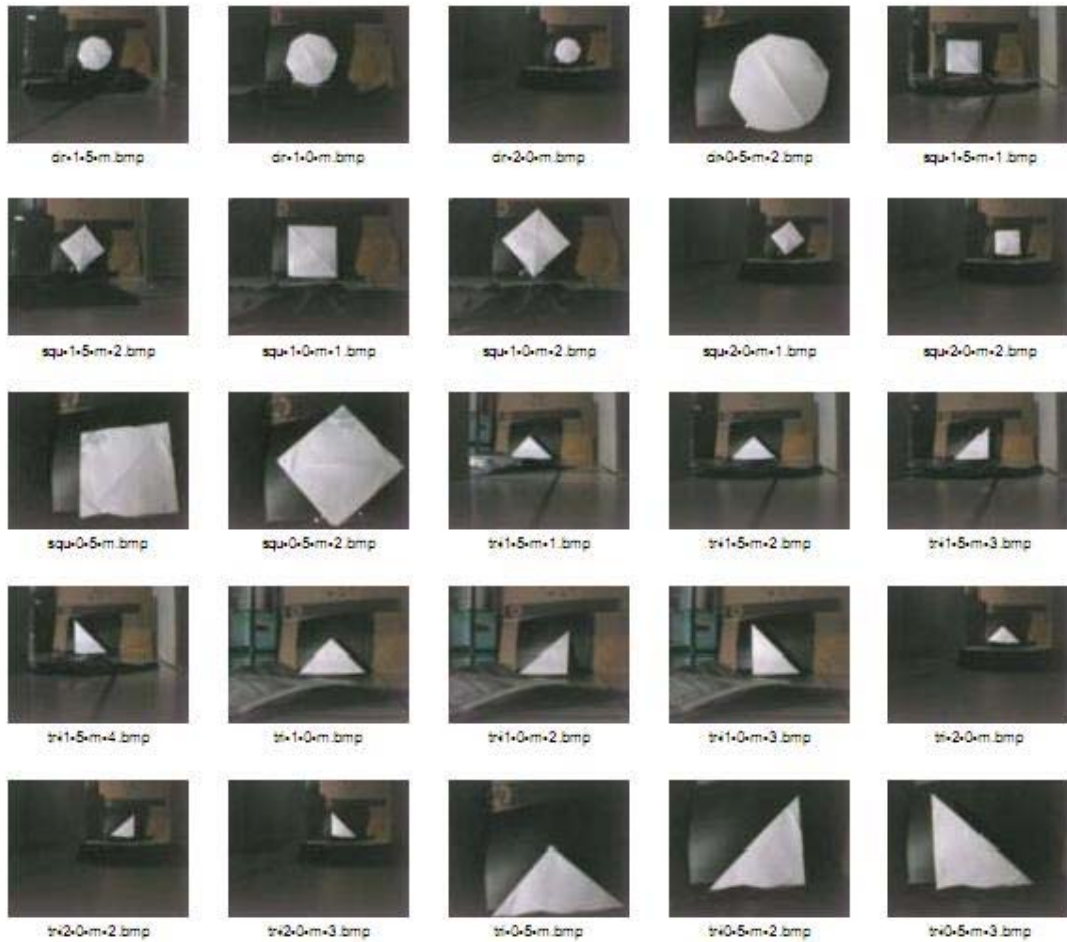
ขั้นตอนในการทดลอง

1. อ่านภาพเข้ามาแล้วเก็บข้อมูลไว้ในหน่วยความจำ (Image Acquisition)
2. ประมวลผลขั้นต้น (Preprocessing) โดยการแบ่งภาพเป็นส่วนๆ โดยการปรับสีของภาพ

มีการกำหนดเส้นกรอบที่ล้อมรูป

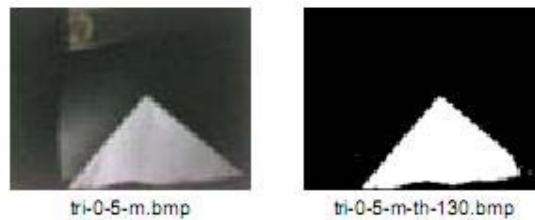
3. แยกลักษณะสำคัญ (Feature Extraction) หาอัตราส่วนของจุดสี (Pixels) ของภาพสีขาว กับ สีดำ (สีพื้น)
 4. ฐานความรู้อย่างง่าย (Simple Knowledge Base) เทียบกับข้อมูลที่มีอยู่แล้วที่ถูกเก็บไว้ในตัวแปร
 5. ขั้นสุดท้ายคือ การจำแนกได้ (Recognition) โดยดูความใกล้เคียงของค่าที่ได้ กับค่าในข้อ 4
- การเก็บข้อมูล (Associative Storage) จะมีการใช้งานร่วมกับขั้นตอนต่างๆ ที่ผ่านมา

ในการทดลองจะวางภาพไว้ที่ระยะ 0.5 เมตร, 1.0 เมตร, 1.5 เมตร, และ 2.0 เมตร รูปภาพที่ทดลองเป็นรูปพื้นฐานทางเรขาคณิต คือ สามเหลี่ยม เหลี่ยม และวงกลม โดยกำหนดให้มีระยะแตกต่างกัน และใช้พื้น (Back ground) เป็นสีดำ (แต่จะเห็นเงาและวัตถุอื่นๆ บ้าง) ตามรูปที่ 1 ด้านล่าง



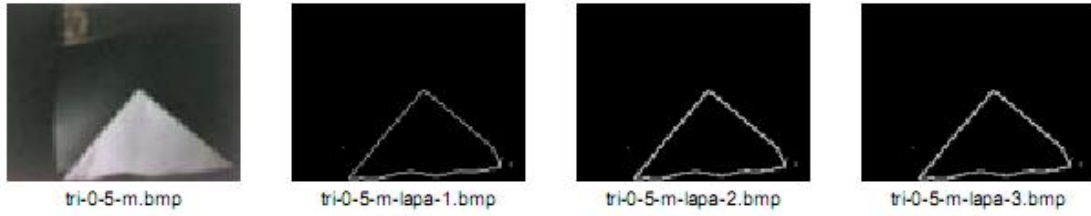
รูปที่ 1 ภาพที่ใช้ในการทดลองตามระยะต่างๆ

ขั้นตอนต่อมา นำภาพมาประมวลผลขั้นต้น (Image Preprocessing) โดยการกำหนดการกรองด้วยค่าเริ่มต้น (Threshold) เพื่อให้ภาพที่ต้องการเหลือแค่ 2 สี ผลตามภาพที่ 2

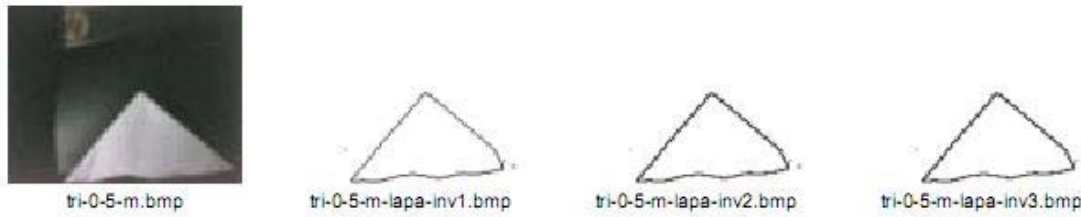


รูปที่ 2 ภาพก่อนประมวลผลและหลังจากทำค่าเริ่มต้น(Threshold) 130 (จาก 0-255) ที่ระยะ 0.5 เมตร

ต่อมาหาขอบของภาพ จากการทดลองใช้ การทำ Laplacian โดยใช้ Mask Coefficient 3 แบบตามรูปด้านล่าง



รูปที่ 3 ภาพก่อนประมวลผลและหลังจากทำ Laplaciant 3 แบบ
ต่อมานำภาพที่ได้มาทำ Invert ตามรูปด้านล่าง

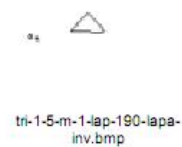
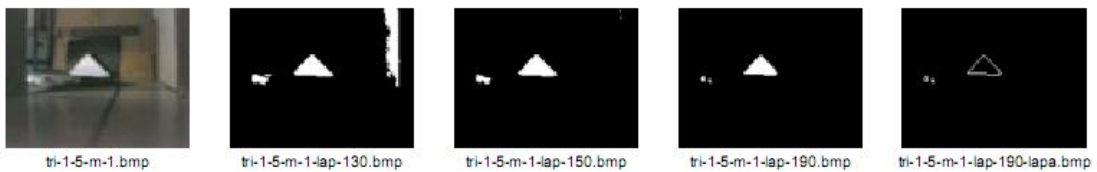


รูปที่ 4 ภาพก่อนประมวลผลและหลังจากทำ Laplaciant 3 แบบ แล้วนำมา invert ที่ระยะ
0.5 เมตร

ทดลองทำกับรูปสามเหลี่ยมที่ระยะต่างๆ ตามรูปด้านล่าง



รูปที่ 5 ภาพก่อนประมวลผลและหลังจากทำ Threshold, Laplaciant และ นำมา invert ที่
ระยะ 1.0 เมตร



รูปที่ 6 ภาพก่อนประมวลผลและหลังจากทำ Threshold 3 ค่า, Laplaciant และ นำมา
invert ที่ระยะ 1.5 เมตร

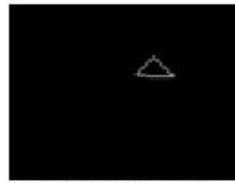
จากรูปที่ 6 ด้านบน จะทำ threshold 3 ค่า เพราะมีภาพอื่นๆ มารบกวน



tri-2-0-m.bmp



tri-2-0-m-th-130.bmp



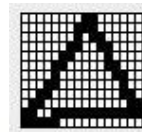
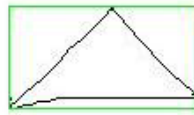
tri-2-0-m-th-130-lap.bmp



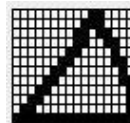
tri-2-0-m-th-130-lap-inv.bmp

รูปที่ 7 ภาพก่อนประมวลผลและหลังจากทำ Laplacian 3 แบบ แล้วนำมา invert ที่ระยะ 2.0 เมตร

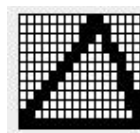
เมื่อได้ภาพจากขั้นตอน กลับภาพ (Invert) แล้วนำมาสุ่มเพื่อให้ เครือข่ายแบบนิวรอล (Neural Network) นำมาเทียบกับค่าที่เรียน (Training) ไว้แล้ว



รูปที่ 8 ตัวอย่างภาพสามเหลี่ยมนำมาจากการทำการประมวลผลเบื้องต้นแล้วนำมาสุ่ม ตัวอย่างที่ 1



รูปที่ 9 ตัวอย่างภาพสามเหลี่ยมนำมาจากการทำการประมวลผลเบื้องต้นแล้วนำมาสุ่ม ตัวอย่างที่ 2



รูปที่ 10 ตัวอย่างภาพสามเหลี่ยมนำมาจากการทำการประมวลผลเบื้องต้นแล้วนำมาสุ่ม ตัวอย่างที่ 3

ผล

รูปภาพที่ทดลองใช้มี 3 แบบ สามารถแยกความแตกต่างในแต่ละแบบได้ แต่รูปที่อยู่ในระยะไกล 2 เมตร จะแยกความแตกต่างระหว่างวงกลมกับสี่เหลี่ยมได้ไม่แน่นอน รูปที่อยู่ในระยะใกล้จะแยกได้ชัดเจนมากกว่า

ถ้ามีรูปหลายๆ แบบอยู่ในบริเวณใกล้เคียงกัน แต่ไม่ซ้อนกัน จะต้องมีการแยกรูปออกเป็นส่วนๆ ก่อน ถ้ามีภาพที่มีดีใกล้เคียงกัน จะได้ภาพหลายๆ ภาพ ก็ต้องแยกภาพออกเป็นส่วนๆ ที่จะทำการทดลองในครั้งต่อไป การหาตำแหน่งที่จะบังคับให้หุ่นยนต์ วิ่งเข้าหา ทำได้โดยหาศูนย์กลางของภาพ ที่ทดลองทำอย่างง่ายๆ โดยแบ่งภาพที่กล้องจับได้ออกเป็นสามส่วนด้านซ้าย ตรงกลาง และด้านขวา โดยหุ่นยนต์จะเคลื่อนไปทางซ้าย หรือขวา เพื่อให้กล้องที่ติดอยู่บนตัวหุ่นยนต์ถูกหมุนตามไปด้วย โดยการหาจุดศูนย์กลางของภาพ โดยจะหยุดเมื่อถึงเป้าหมายหรือไม่สามารถหาค่าที่ใกล้เคียงกับที่กำหนดไว้

วิจารณ์

ในการทดลองนี้เป็นขั้นต้น ในขั้นต่อไปจะมีการใช้รูปที่มีขอบที่ซับซ้อนขึ้น และใช้วิธีที่ซับซ้อนขึ้น เพื่อหาคุณลักษณะสำคัญของภาพ ที่เมื่อมีการเปลี่ยนแปลงด้วย การหมุน ย่อหรือขยาย แล้วยังมีคุณลักษณะสำคัญเปลี่ยนแปลงน้อย

ในส่วนของการรับภาพใช้กล้องที่มีความละเอียดได้มากขึ้น มีการแปลงภาพได้เร็ว และปรับระยะใกล้ไกลได้

ส่วนของการประมวลผลจะมีการทดลองใช้ ระบบสมองกลฝังตัว (Embedded System) มีขนาดเล็กและมีความเร็วในการประมวลผลที่มากพอโดยจะทำการทดลองในครั้งต่อไป

ส่วนสุดท้ายคือส่วนของหุ่นยนต์ที่ปรับความเร็วได้ และสัมพันธ์กับการแปลงหรือ การประมวลผลภาพ และอาจจะให้มีการควบคุมได้มากกว่าแนวราบซึ่งต้องขึ้นกับอัตรา การซูมของภาพ และความเร็วในการประมวลผล

สรุป

วิธีการที่ทดลองเป็นวิธีที่ไม่ซับซ้อนมากนัก เพื่อต้องการประมวลผลที่เร็วและ เป็นแนวทางนำไปใช้กับระบบสมองกลฝังตัวเพื่อให้มีขนาดที่ไม่ใหญ่มากนักและมีราคาไม่แพงเกินไป และได้ผลในระดับหนึ่ง และการฝึกและเรียนรู้ของเครือข่ายแบบนิวรอล

เอกสารอ้างอิง

- [1] Arun D Kulkarni, 1994. Artificial Neural Networks for Image Understanding. Van Nostrand Reinhold, New York NY 10003.
- [2] Jin Soo NOH and Kang Hyeon RHEE, Dept. of Electronic Eng., College of Elec-Info Eng., Chosun University 2005. Palmprint Identification Algorithm using Hu Invariant Moments and Otsu Binarization; Proceedings of the Fourth Annual ACIS International Conference on Computer and Information Science (ICIS'05) IEEE.
- [3] Scott E Umbaugh, 2005. Computer Imaging Digital Image Analysis and Processing; CRC Press, Boca Raton Florida 33431.
- [4] Rafael C. Gonzalez, Srichard E woods, Steven L Eddins 2004. Digital Image Processing Using MATLAB; Pearson Education, Inc.
- [5] Arun D Kulkarni, 2001. Computer Vision and Fuzzy-Neural Systems. Prentice Hall, Inc. New Jersey 07458.