

วิธีการที่มีประสิทธิภาพในการสร้างรหัสสำหรับเครื่องกัดชิ้นงาน ด้วยคอมพิวเตอร์วิชัน

An effective technique to generate the code for engraving machine by computer vision

นายสุเทพ อาษา¹

รศ.ดร.ทวีพล ชี้อัสตัย^{1*} และ รศ.สุพรรณ กุลพาณิชย์

สาขาวิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ถนนฉลองกรุง เขตลาดกระบัง กรุงเทพมหานคร 10520

*Email: suthepasa@gmail.com, 0814944903,

*Email: taweepol.suesut@gmail.com, *Email: kgsuphan@kmitl.ac.th

บทคัดย่อ

งานวิจัยนี้นำเสนอวิธีการอย่างง่ายและมีประสิทธิภาพสำหรับสร้างรหัสคำสั่งที่ใช้ได้กับเครื่องกัด เครื่องพลอตเตอร์ และเครื่องหยอดกาวซึ่งมีลักษณะเป็นเส้นทางเดิน ในงานวิจัยนี้ได้ใช้การสร้างรหัสคำสั่งสำหรับเครื่องกัดชิ้นงานเป็นกรณีศึกษา โดยใช้คอมพิวเตอร์วิชันและการประมวลผลภาพดิจิทัลจากภาพถ่ายของวัตถุต้นแบบ เนื่องจากการสร้างรหัสคำสั่งวิธีเดิมอาศัยการสร้างจากโปรแกรม CAD/CAM เพื่อเป็นต้นแบบ วิธีการที่นำเสนอนี้จัดเป็นวิศวกรรมย้อนรอยสำหรับใช้งานกับการสร้างเส้นทางเดินจากภาพสองมิติ โดยค้นหาขอบภาพของชิ้นงานแล้วใช้วิธีการสเก็ลตัน (Skeleton) เพื่อลดขนาดของขอบให้เหลือขอบพิกเซลเดียว ซึ่งเป็นทางเดินของเครื่องจักรอัตโนมัติอย่างง่าย จากนั้นพิจารณาตัดจุดที่ไม่ต้องการออกจากรูปภาพของชิ้นงานรวมทั้งทำการแปลงพิกัดจริงด้วยวิธีโฮโมกราฟี (Homography transform) เพื่อแก้ไขความคลาดเคลื่อนจากมุมมองภาพและใช้วิธีเชนโค้ด (chain code) ในการแปลงเส้นทางเดินของเครื่องจักรแบบอัตโนมัติ ซึ่งการสร้างรหัสทางเดินด้วยวิธีเชนโค้ดจากการทดลองพบว่าวิธีนี้มีความรวดเร็วและแม่นยำมากขึ้นเมื่อเทียบกับการป้อนพิกัดแบบเรียนรู้ตำแหน่ง (Teaching point) และวิธีเลือกตำแหน่งบนภาพสองมิติ

คำหลัก: CAD/CAM; สเก็ลตัน; โฮโมกราฟี; เชนโค้ด; ภาพสองมิติ

Abstract

This paper presents a simple and effective method to generate the machine tool path code for applying to the milling, the plotter and the glue coating machines. The plotter coding for the engraving machine using computer vision and image processing was used as a case study in the paper. Generating command code using CAD/CAM software for prototype is the traditional method. Our method is one of the reverse engineering technique to create routing of tool path from two-dimensional image. The skeleton image processing technique and edge finding were applied to sharpen images for the basic machine path. The homography transform was used to transform the image plane to the real world plane in order to correct the perspective distortion. The chain code was employed to formulate the machine path from the image coordinates automatically. The experimental results showed that this method is faster and more accurate when compared to the teach point method.

Keywords: CAD / CAM; Skeleton; Homography; Chain code; Images two-dimensional.

AMM-167

1. บทนำ

ในปัจจุบันนี้อุตสาหกรรมการผลิตได้เจริญก้าวหน้าไปอย่างรวดเร็ว การผลิตชิ้นงานเพื่อใช้ในอุตสาหกรรม ต้องเป็นชิ้นงานที่มีคุณภาพดี มีขนาดและสัดส่วนที่ถูกต้องตามกำหนดไว้ ส่วนใหญ่ในกระบวนการประกอบเครื่องยนต์จะใช้วิธีแปลงรหัสทางเดินด้วยวิธีเรียนรู้ตำแหน่งแบบจุด[1][7] ซึ่งในอุตสาหกรรมเครื่องพลาตเตอร์ เครื่องกัด และเครื่องหยอดกาวจะทำงานด้วยการป้อนรหัสทางเดินแบบเรียนรู้ตำแหน่ง หรือแบบแปลงรหัสทางเดินจากโปรแกรมออโตแคด (AutoCAD) จากการปฏิบัติงานพบว่าการป้อนรหัสทางเดินแบบเรียนรู้ตำแหน่ง จะใช้เวลาในการป้อนรหัสทางเดินค่อนข้างนาน เมื่อรหัสทางเดินของเครื่องพลาตเตอร์ เครื่องกัด เครื่องหยอดกาวมีความซับซ้อน จึงจำเป็นต้องมีการพัฒนาเพื่อให้ได้ชิ้นงานที่มีคุณภาพ มีความแม่นยำและรวดเร็วขึ้น

ในงานวิจัยนี้จึงได้นำเสนอวิธีการที่มีประสิทธิภาพในการสร้างรหัสสำหรับเครื่องกัดชิ้นงานด้วยคอมพิวเตอร์วิชัน เพื่อสร้างรหัสทางเดินอย่างง่ายและย่อระยะเวลาในกระบวนการสร้างรหัสทางเดินโดยใช้คอมพิวเตอร์วิชันและการประมวลผลภาพดิจิทัลจากภาพถ่ายของวัตถุต้นแบบ วิธีการที่นำเสนอนี้จัดเป็นวิศวกรรมย้อนรอยสำหรับใช้งานกับการสร้างเส้นทางเดินจากภาพสองมิติ โดยค้นหาขอบภาพของชิ้นงาน[4] แล้วใช้วิธีสเกิร์ตัน(Skeleton) เพื่อลดขนาดของขอบให้เหลือขอบพิกเซลเดียว[5] ซึ่งจะเป็นทางเดินของเครื่องจักรอัตโนมัติอย่างง่าย จากนั้นพิจารณาตัดจุดที่ไม่ต้องการออกจากรูปภาพของชิ้นงาน รวมทั้งทำการแปลงพิกัดจริงด้วยวิธีโฮโมกราฟฟี (Homography transform) เพื่อแก้ไขความคลาดเคลื่อนจากมุมมองภาพ[2][3][6] และใช้วิธีเชนโค้ด (chain code) ในการแปลงเส้นทางเดิน[7]ของเครื่องจักรแบบอัตโนมัติ ซึ่งการสร้างรหัสทางเดินด้วยวิธีเชนโค้ดจากการทดลองพบว่าวิธีนี้มีความรวดเร็วและแม่นยำมากขึ้นเมื่อเทียบกับการป้อนพิกัด

แบบเรียนรู้ตำแหน่ง (Teaching point) และวิธีเลือกตำแหน่งบนภาพสองมิติ [7]

2. หลักการของโซเบล (Sobel)

การหาขอบภาพจะกระทำกับภาพระดับสีเทา (Gray level image) ที่มีค่าระดับสีเทาอยู่ระหว่าง 0-255 ซึ่งในที่นี้จะเป็นการหาขอบของวัตถุเพื่อลดจำนวนข้อมูลให้น้อยลงโดยยังคงเอกลักษณ์ของวัตถุนั้นๆ การหาขอบด้วยวิธีโซเบล (Sobel) [4] เป็นการหาขอบภาพโดยใช้หน้าต่างขนาด 3x3 วางทับกับข้อมูลภาพสองมิติ $f(x,y)$ ที่ตำแหน่ง (x,y) แล้วทำการคอนโวลูชัน (Convolution) คำนวณในหน้าต่างกับข้อมูลภาพ ซึ่งการหาขอบภาพด้วยวิธีโซเบล (Sobel) ซึ่งจะทำการหาขอบภาพทั้งในแนวนอนและแนวตั้ง ดังนั้นจึงมีหน้าต่างขนาด 3x3 จำนวน 2 หน้าต่างดังแสดงในตารางที่ 1 และตารางที่ 2 ดังนี้

-1	-2	-1
0	0	0
1	2	1

ตารางที่ 1 หน้าต่างสำหรับหาขอบภาพแนวนอน

1	0	-1
2	0	-2
1	0	-1

ตารางที่ 2 หน้าต่างสำหรับหาขอบภาพแนวตั้ง

เนื่องจากวิธีโซเบล(Sobel) จะหาขอบภาพทั้งในแนวนอนและแนวตั้ง ดังนั้นในตอนแรกจะทำการวางหน้าต่างในแนวนอนทับลงไปบนด้านซ้ายมือของข้อมูลภาพ $f(x,y)$ จากนั้นทำการคอนโวลูชัน (Convolution) หน้าหนักที่อยู่ในหน้าต่างกับข้อมูลภาพสองมิติผลลัพธ์ที่ได้จะเก็บไว้ที่ภาพเอาท์พุท $Gx(x,y)$ เมื่อได้ผลลัพธ์แล้วหน้าต่างจะเลื่อนไปทางขวามือหนึ่งพิกเซลและทำการคอนโวลูชัน (Convolution) เพื่อให้ได้ผลลัพธ์และเก็บภาพเอาท์พุท $Gx(x,y)$ การดำเนินการจะเป็นเช่นนี้ไปเรื่อยๆ จากซ้ายไปขวาและ

AMM-167

จากบนลงล่างจนหมดทั้งภาพ ซึ่งผลที่ได้คือขอบภาพในแนวนอน $Gx(x,y)$ ในทำนองเดียวกันจะใช้หน้าต่างในแนวตั้งทำการคอนโวลูชัน (Convolution) กับข้อมูลภาพ $f(x,y)$ จนกระทั่งครบทั้งภาพ จะได้ขอบภาพในแนวตั้ง $Gy(x,y)$ เมื่อได้ขอบภาพในแนวนอน $Gx(x,y)$ และแนวตั้ง $Gy(x,y)$ แล้วจะทำการหาขอบภาพทั้งหมดในรูปภาพด้วยการคำนวณหาขนาดของเกรเดียนท์ (∇f) โดยให้สมการที่ (1) และ (2)

$$\nabla f = \sqrt{Gx^2 + Gy^2} \quad (1)$$

$$\nabla f \approx |Gx| + |Gy| \quad (2)$$

อย่างไรก็ตามเนื่องจากขนาดของเกรเดียนท์ (∇f) ที่คำนวณได้อาจมีค่ามากกว่า 255 ดังนั้นจึงจำเป็นที่จะต้องทำการสเกล (Scaling) อยู่ในช่วง 0 – 255 ก่อนจากนั้นจึงเลือกค่า Threshold เพื่อแสดงขอบภาพเป็นสีขาวและสีดำหรือภาพสองระดับ (Binary images)

3. หลักการ Skeletons detection

การหาแกนกลางภาพเป็นการหา [5] แกนกลางของวัตถุซึ่งในที่นี้เส้นขอบให้บางลงจนเหลือแต่แกนกลางที่มีความหนาเพียง 1 พิกเซลเพื่อเป็นการลดขนาดของข้อมูลของเส้นขอบให้เหลือแต่ส่วนที่สำคัญเท่านั้นภาพที่ต้องการหาแกนกลางนั้นจะต้องเป็นภาพสองระดับ (Binary Image) นั่นคือมีค่าเป็น "0" และ "1" โดยที่ในส่วนที่เป็นพื้นที่ของเส้นมีค่าเป็น "1" และ "0" คือส่วนที่เป็นพื้นฉากหลัง วิธีที่การหาแกนกลางนั้นประกอบด้วยสองขั้นตอนซึ่งในแต่ละขั้นตอนจะทำการลบพิกเซลแต่ละพิกเซลของเส้นขอบโดยใช้หน้าต่าง 3x3 ดังแสดงในตารางที่ 3 วางทาบลงไปจุดภาพ โดยการตรวจหาพิกเซลที่มีค่าเป็น "1" เมื่อตรวจพบแล้วจึงเริ่มขั้นตอนที่ 1 และ 2

P9	P2	P3
P8	P1	P4
P7	P6	P5

ตารางที่ 3 หน้าต่างขนาด 3x3 ซึ่ง p1 คือขอบของวัตถุ

ขั้นตอนในการหาแกนกลางมีดังนี้

ขั้นตอนที่ 1 เป็นการลบจุดภาพบริเวณขอบวัตถุทางด้านขวามือและด้านล่าง โดยวางหน้าต่าง 3x3 ทาบไปบนจุดภาพที่มีค่าเป็น "1" และกำหนดให้เป็น p1 จากนั้นตรวจสอบเงื่อนไขดังต่อไปนี้

- $2 \leq N(p1) \leq 6$
- $S(p1) = 1$
- $p2.p4.p6 = 0$
- $p4.p6.p8 = 0$

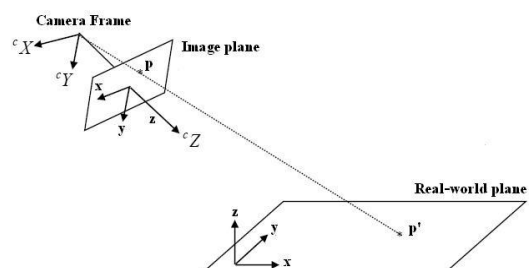
ขั้นตอนที่ 2 เป็นการลบจุดภาพบริเวณขอบวัตถุทางด้านซ้ายมือและด้านบน โดยมีเงื่อนไขดังต่อไปนี้

- $2 \leq N(p1) \leq 6$
- $S(p1) = 1$
- $p2.p4.p8 = 0$
- $p2.p6.p8 = 0$

4. หลักการ Homography Transform

4.1 Homography Transform

ระบบการวัดบนระนาบเป็นวิธีการวัดในลักษณะเรขาคณิตบนระนาบ นำมาใช้เพื่อแปลงระยะทางในพิกัดบนรูปภาพเป็นค่าระยะทางจริงโดยเทียบกับพิกัดอ้างอิง [2][3][6] ในระบบสองมิติ Homography Matrix จะเป็นตัวแปลงระหว่าง Homogeneous Point และ Homogeneous Coordinate



รูปที่ 1 แสดงความสัมพันธ์ระหว่างจุดบนพิกัด (P) และจุดบนพิกัดจริง (P')

AMM-167

จากความสัมพันธ์ระหว่างระนาบในระบบพิกัดรูปภาพ (p) และจุดบนพิกัดจริง (p') สามารถเขียนได้เป็น

$$p' = Hp \quad (3)$$

เมื่อ p' และ p เป็น Homogeneous Coordinate ที่ร่วมกันของตำแหน่ง P (พิกัดรูปภาพ) และ p' (พิกัดจริง) ซึ่งมี H เป็น Homography Matrix

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix} \quad (4)$$

โดยในสมการที่ (4) จะเพิ่มจุดสมมติ w และ w' ขึ้นมา โดยให้มีค่าเท่ากับ 1 แล้วคูณ P และ H เมื่อ $\frac{x'}{w'}$ และ $\frac{y'}{w'}$ โดยที่ค่า $w = w' = 1$ จัดรูปแบบสมการใหม่ได้เป็น

$$x' = \frac{h_1x + h_2y + h_3}{h_7x + h_8y + h_9}, \quad y' = \frac{h_4x + h_5y + h_6}{h_7x + h_8y + h_9} \quad (5)$$

ซึ่งสามารถจัดสมการใหม่ให้อยู่ในรูปแบบ Homogeneous ได้เป็นสมการที่ (6) โดยที่ตัวแปร 9 ตัวภายใน H สามารถที่จะแปลความหมายเป็นอัตราการขยายหรือตัวแปลงค่าจุดระหว่างพิกัด ซึ่งในการหาค่า Homography Matrix จำเป็นจะต้องทราบค่าจุดบนพิกัดอย่างน้อย 4 จุด [4]

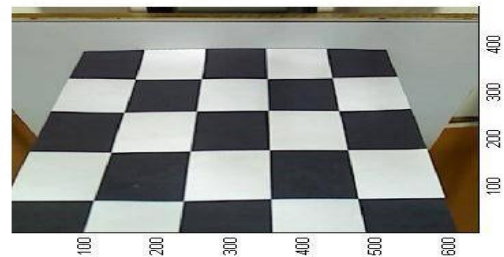
$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & x'_1x_1 & x'_1y_1 & x'_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_4 & y_4 & 1 & 0 & 0 & 0 & x'_4x_4 & x'_4y_4 & x'_4 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & y'_1x_1 & y'_1y_1 & y'_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & x_4 & y_4 & 1 & y'_4x_4 & y'_4y_4 & y'_4 \\ h_1 & h_2 & h_3 & 0 & 0 & 0 & 0 & 0 & 0 \\ h_4 & h_5 & h_6 & 0 & 0 & 0 & 0 & 0 & 0 \\ h_7 & h_8 & h_9 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (6)$$

Homography Matrix สามารถที่จะหาได้โดยการใช้ระบบสมการเชิงเส้น Singular value decomposition (SVD) เป็นวิธีการประมาณค่าแบบกำลังสองน้อยสุด (Least Squares; LS) ซึ่งเมื่อทราบค่าของ H ก็จะสามารถที่จะนำไปคำนวณหาค่าจริง

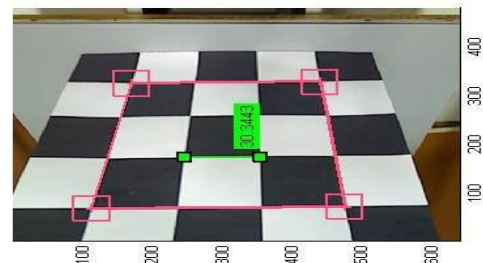
ของตำแหน่งบนรูปภาพโดยการคูณกันระหว่าง H และ P ดังสมการที่ (3)

4.2 การสอบเทียบ

ในการสอบเทียบจะใช้ตารางที่ทราบขนาดเพื่อจะเป็นตัวกำหนดตำแหน่ง 4 จุดของพิกัดรูปภาพเทียบกับขนาดจริงในพิกัดจริง เพื่อที่จะนำไปคำนวณหาค่า Homography Matrix (H) ซึ่งเป็นค่าแฟกเตอร์ที่นำไปแปลงค่าจากข้อมูลรูปภาพเป็นค่าในพิกัดจริง



รูปที่ 2 ตารางขนาด 30x30 mm. ที่ใช้ในการสอบเทียบ โดยทำการกำหนดจุดแต่ละจุดและนำค่าที่ได้ไปคำนวณตามสมการที่ (4) โดยการใช้ฟังก์ชัน SVD ที่เป็นการประมาณค่าแบบกำลังสองน้อยสุด (Least Squares; LS) ในโปรแกรม Matlab เป็นตัวช่วยในการคำนวณหาค่า Homography Matrix (H) ซึ่งผลลัพธ์ที่ได้จะอยู่ในรูปแบบเมทริกซ์ขนาด 3X3 และสามารถนำค่า H ไปใช้ในสมการที่ (3)



รูปที่ 3 ขั้นตอนและผลลัพธ์ที่ได้จากการสอบเทียบ

4.3 วัตถุตัวอย่างที่ใช้ในการทดสอบ



AMM-167

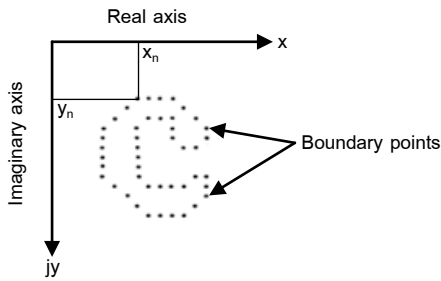
4 0
5 7
6

รูปที่ 4 แสดงวิธีการเก็บภาพตัวอย่าง

รูปที่ 6 ทิศทาง 8 ทิศทางรอบจุด p1 ที่พิจารณา

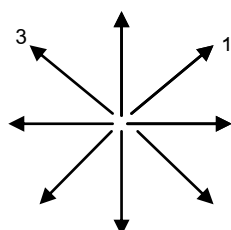
5. หลักการของ Chain Code

การหาตำแหน่งพิกัดคือการค้นหาและเก็บพิกัดของจุดภาพ [7] ที่อยู่บนขอบในแนวแกน x และ y แล้วจัดเก็บในรูปแบบของตัวเลขเชิงซ้อน

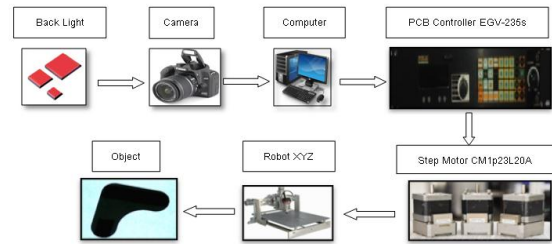


รูปที่ 5 แสดงในรูปแบบตัวเลขเชิงซ้อน จุด (x_n, y_n)

จากรูปที่ 5 แสดงพิกัดของเส้นขอบที่มีจำนวน N จุด ในแนวแกน x, y โดยกำหนดให้จุดเริ่มต้นอยู่ที่จุด (x_n, y_n) ในการหาพิกัดของเส้นขอบของภาพต้องผ่านการหาเส้นขอบภาพมาแล้วและถูกทำให้บางลงเหลือความหนาขอบเพียง 1 พิกเซล จากนั้นทำการตรวจหาจุดภาพที่มีค่าเป็น "1" โดยโปรแกรมจะเคลื่อนที่จากซ้ายไปขวาและจากบนลงล่าง เมื่อพบจุดพิกเซลที่มีค่าเป็น "1" จะวางหน้าต่าง 3×3 ทาบที่จุดนั้น แล้วพิจารณาจุดข้างเคียงที่อยู่ล้อมรอบจุด $P1$ ในลักษณะ 8 ทิศทางตามเข็มนาฬิกา ดังรูปที่ 6 ถ้าตรวจพบจุดภาพถัดไปมีค่าเป็น "1" จะทำการย้ายจุด $P1$ ไปที่จุดนั้นแล้วตรวจหาจุดข้างเคียง 8 ทิศทางอีกเมื่อพบจุดที่มีค่าเป็น "1" จะเก็บพิกัดในแนวแกน x และ y การดำเนินการจะกระทำไปเรื่อยๆ จนกระทั่งสามารถเก็บพิกัดของจุดภาพที่อยู่บนขอบภาพได้ทั้งหมดสุดท้ายจะได้พิกัด x และ y ของตำแหน่งพิกัดจุดทั้งหมดที่มีค่าเป็น "1"



6. ส่วนประกอบของระบบ



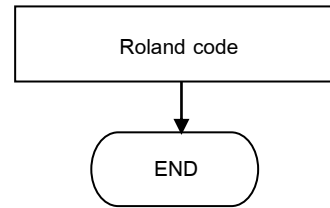
รูปที่ 7 ส่วนประกอบของเครื่องกัด PCB

ส่วนประกอบของของระบบในงานวิจัยนี้ใช้แหล่งกำเนิดแสงรุ่น LP-200 หลอดฟลูออเรสเซนต์ 178 มม. x 127 มม. และส่วนของการรับภาพจะใช้กล้อง EOS Canon 1000D เลนส์ EFS18-55 mm คอมพิวเตอร์ Lenovo T500 Intel core2duo CPU T9600 2.8 GHz Ram 4 GB Win7 32 bit Controller Robot รุ่น EGV-235s Motor Stepping รุ่น CM1P23L20A พื้นที่ของการเคลื่อนที่ของหุ่นยนต์ 540 มม. x 540 มม. ในการกำหนดเส้นทางเดินของเครื่องกัดแบบอัตโนมัติทำงานด้วยการเขียนแบบในโปรแกรมออตแคด เปรียบเทียบการเคลื่อนที่จากการแปลงรหัสแบบ Roland Printer MDX-650 (RML-1) ด้วยโปรแกรม VCam Version 3.0 ผ่านพอร์ตการสื่อสารอนุกรมโดยการแปลงจาก CAD ซึ่งจะมีความถูกต้องมากที่สุดแต่ในการปฏิบัติงานจริงพบว่าในบางครั้งจะไม่สามารถรู้ขนาดของวัตถุตัวอย่างได้ด้วยการวัดด้วยเครื่องมือวัด เช่น ชิ้นส่วนเครื่องจักร ปลายไทย บนไม้ฝาเฟอร์รา

ในงานวิจัยนี้ได้นำคอมพิวเตอร์วิชั่นมาประยุกต์ใช้งานเพื่อสร้างรหัสทางเดินของเครื่องกัด

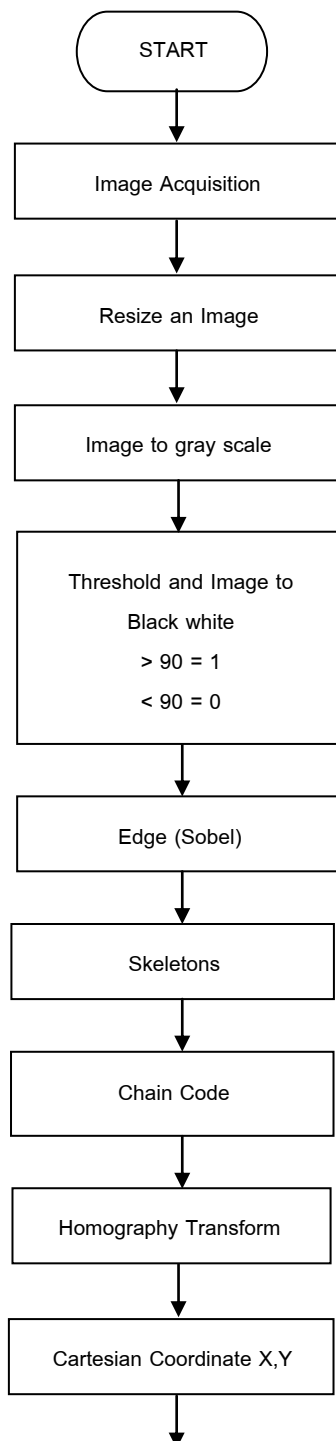
AMM-167

แบบอัตโนมัติ ได้จากการถ่ายภาพด้วยกล้อง EOS Canon 1000D หนึ่งความเร็วในการถ่ายภาพ 1/40 และเปิดรูรับแสง F4.0 เพื่อแก้ปัญหาการบิดเบี้ยวของภาพ หลังจากที่ได้ภาพที่สมบูรณ์มาแล้วจะทำการประมวลผลภาพด้วยโปรแกรม Matlab 7.0.4 และแปลงเป็นรหัสทางเดินของ เครื่องกัด



รูปที่ 8 แสดงการประมวลผลสร้างทางเดินเครื่องกัด

7. กระบวนการเก็บตัวอย่างภาพ



8. การกำหนดและปรับปรุงรหัสทางเดินของเครื่อง

กระบวนการกำหนดและปรับปรุงทางเดินของเครื่องกัดเครื่องพล็อตเตอร์ และเครื่องหยอดกาว เริ่มต้นจากการใช้กล้อง EOS Canon 1000D Lens EFS 18-55 mm ปรับรูรับแสงให้กว้างสุด (F4.0) ความเร็วในการถ่ายภาพ 1/40 ความไวแสง ISO 100 ถ่ายภาพวัตถุตัวอย่างที่มุมกล้องแตกต่างกันเพื่อเปรียบเทียบความแม่นยำและเที่ยงตรงของการแปลงรหัสทางเดินแต่ละครั้ง ดังนั้นในงานวิจัยนี้จึงได้ทำการถ่ายภาพตัวอย่างที่มุม 90, 60 และ 45 องศา หลังจากที่ได้ภาพถ่ายที่มุมต่างๆมาครบทั้ง 3 ภาพแล้วจะทำการลดขนาดภาพเพื่อลดสัญญาณรบกวน ในงานวิจัยนี้จะลดขนาดภาพ 80 เปอร์เซ็นต์ของความละเอียดกล้องสูงสุดและใช้โปรแกรม MatLab 7.0.4 ในการแปลงภาพทั้ง 3 ภาพเป็นภาพเทา เพื่อลดขนาดของข้อมูลในการประมวลผล ทำการปรับ Threshold เพื่อแปลงเป็นภาพขาวดำ (Binary) ต่อจากนั้นให้หาขอบภาพโดยใช้หลักการของโซเบล (Sobel) แต่ผลของขอบภาพที่ได้จะยังคงมีความหนาแน่นมากกว่า 1 พิกเซล และใช้หลักการของสกีริตัน (skeleton) เพื่อให้ได้ขอบภาพ 1 พิกเซล กระบวนการต่อมาจึงแปลงภาพให้เป็นขนาดจริง โดยใช้หลักการของ Homography Transform และแปลงเป็นพิกัด (x,y) โดยใช้หลักการของเซนโค้ด หลังจากที่ได้พิกัด (x,y) จะบ่งบอกรหัสทางเดินของเครื่องกัด เครื่องพล็อตเตอร์และเครื่องหยอดกาวและแปลงเป็นรหัส Roland Printer ดังตารางที่ 5

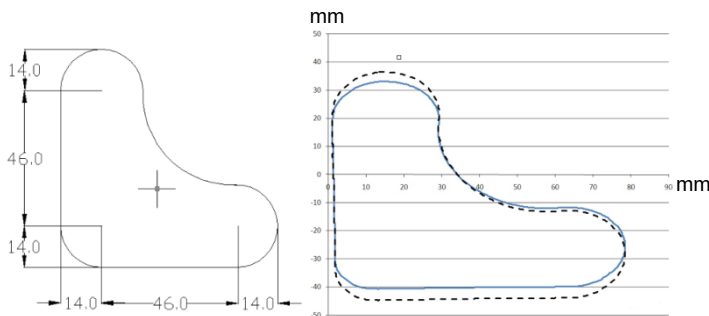
^	PD	X1	,	Y1	;
^	PD	X2	,	Y2	;
^	PD	X3	,	Y3	;

AMM-167

^	PD	X4	,	Y4	;
.
.
^	PDn	Xn	,	Yn	;

ตารางที่ 5 ตารางแสดงการแปลงรหัสโรแลนด์ปริ้นเตอร์

8.1 การเปรียบเทียบระหว่างการป้อนพิกัดแบบเรียนรู้อัตโนมัติกับการแปลงพิกัดแบบอัตโนมัติ



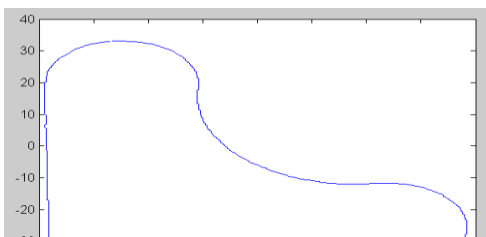
รูปที่ 9 แสดงขนาดวัตถุตัวอย่าง (ด้านซ้าย) และการเปรียบเทียบระหว่างการแปลงพิกัดแบบเลือกจุดแสดงด้วยเส้นประ กับการแปลงพิกัดแบบอัตโนมัติแสดงด้วยเส้นทึบ (ด้านขวา)

8.2 การประมวลผลภาพและการปรับปรุงลายเส้นที่มุมถ่ายภาพ 90,60 และ 45 องศา

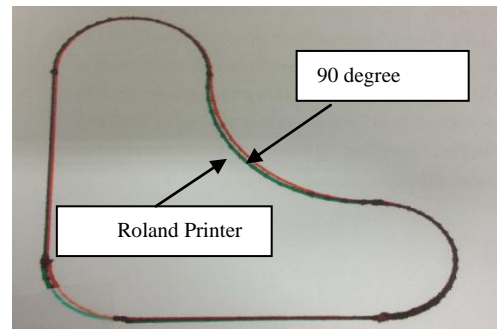
นำภาพ RGB ที่ได้มาประมวลผลภาพบนโปรแกรม MATLAB 7.0.4 เพื่อแปลงเป็นภาพเทา และแปลงเป็นภาพขาวดำตามลำดับ จากนั้น พิจารณาภาพเพื่อลบพื้นที่บริเวณกลุ่มพิกเซลสีขาวออก และหาเส้นขอบภาพของวัตถุตัวอย่างโดยใช้วิธีโซเบล (Sobel) และทำให้ขอบภาพบางเหลือ 1 พิกเซลโดยใช้หลักการสเกิร์ตตัน (Skeleton) แล้วทำการหาตำแหน่งพิกเซลโดยใช้หลักการเชนโค้ด (Chain code)

8.2.1 มุมกล้อง 90° นำพิกัดที่ได้จากเชนโค้ดมาคูณกับสมการที่ 7 เพื่อแปลงเป็นพิกัดจริง แสดงดังรูปที่ 10

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} 0.0136 & -0.0000 & -0.1903 \\ 0.0000 & -0.0138 & 0.9811 \\ -0.0000 & -0.0000 & 0.0276 \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix} \quad (7)$$



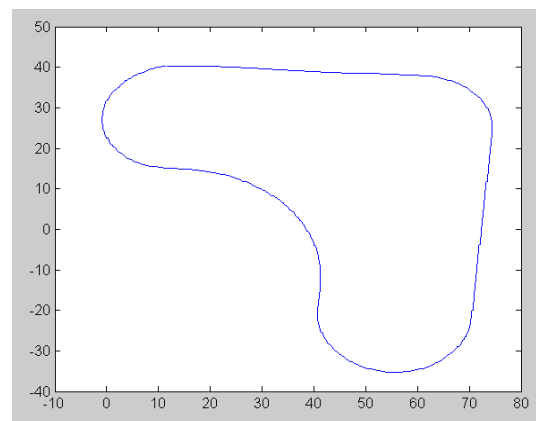
รูปที่ 10 แสดงภาพหาโฮโมกราฟฟิทรานฟอร์ม



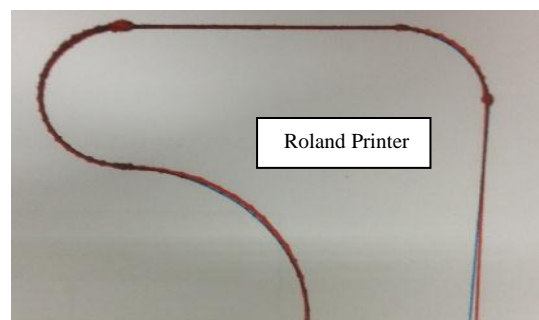
รูปที่ 11 แสดงภาพรหัสทางเดินจากโปรแกรมอัตโนมัติ เติบกับ รหัสทางเดินจากการแปลงภาพ

8.2.2 มุมกล้อง 60° นำพิกัดที่ได้จากเชนโค้ดมาคูณกับสมการที่ 8 เพื่อแปลงเป็นพิกัดจริง แสดงดังรูปที่ 12

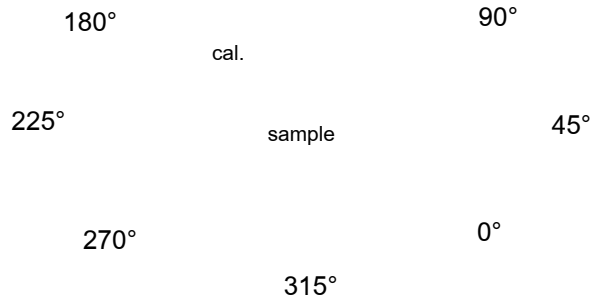
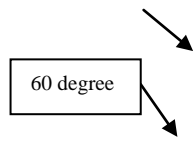
$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} 0.0056 & -0.0001 & -0.0982 \\ -0.0002 & -0.0056 & 0.9948 \\ 0.0000 & 0.0000 & 0.0266 \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix} \quad (8)$$



รูปที่ 12 แสดงภาพหาโฮโมกราฟฟิทรานฟอร์ม

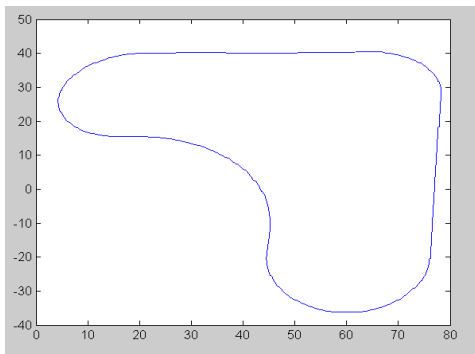


AMM-167

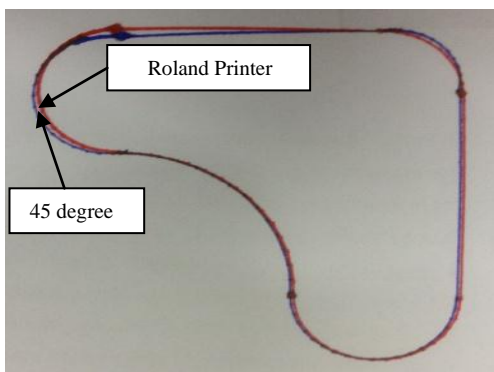


รูปที่ 13 แสดงภาพรหัสทางเดินจากโปรแกรมอัตโนมัติ
แคด เทียบกับ รหัสทางเดินจากการแปลงภาพ
8.2.3 มุมกลิ้ง 45° นำพิกัดที่ได้จากเซนโค้ดมาคูณ
กับสมการที่ 9 เพื่อแปลงเป็นพิกัดจริง แสดงดังรูปที่ 14

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} 0.0050 & 0.0012 & -0.2655 \\ 0.0001 & -0.0056 & 0.9637 \\ 0.0000 & 0.0000 & 0.0257 \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix} \quad (9)$$

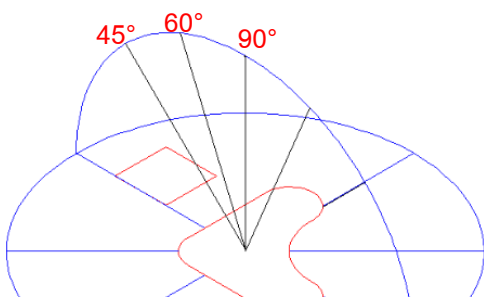


รูปที่ 14 แสดงภาพหาโฮโมกราฟฟิทรานฟอร์ม



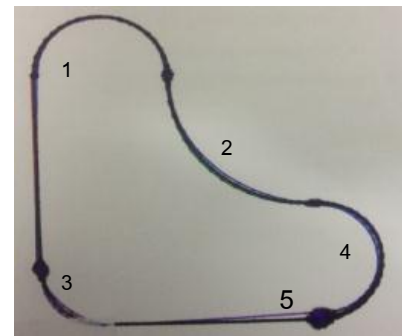
รูปที่ 15 แสดงภาพรหัสทางเดินของเครื่องจักร
จากโปรแกรมอัตโนมัติแคดเทียบกับรหัสทางเดินจากการ
แปลงภาพ

8.3 เปรียบเทียบแปลงรหัสด้วยวิธีโรแลนด์ปริ้น
เตอร์ (Roland Printer)



รูปที่ 16 การพล็อตที่มุมกลิ้ง 90,60 และ 45 องศา
เปรียบเทียบกับตัวอย่าง

หลังจากทำการพล็อตจุดด้วยโปรแกรม ออโต
แคด (AutoCAD) และวิธีรหัสโรแลนด์ (Roland Code)
พบว่ามีความคลาดเคลื่อนประมาณ ±1.5 มม. โดย
ส่วนใหญ่จะพบบริเวณส่วนโค้งของขอบภาพ แสดงดัง
รูปที่ 17



รูปที่ 17 ส่วนของภาพที่มีความคลาดเคลื่อนทั้ง 5 จุด

มุม ถ่ายภาพ (องศา)	ค่าความคลาดเคลื่อน (มม.)					ค่า ความคลาด เคลื่อนเฉลี่ย (มม.)
	จุด ที่ 1	จุด ที่ 2	จุด ที่ 3	จุด ที่ 4	จุด ที่ 5	
90°	0.3	0.5	0.3	0.3	0.3	0.34
60°	0.4	0.6	0.4	1	0.3	0.54
45°	0.5	1	0.7	0.4	2	0.94

ตารางที่ 6 แสดงค่าความคลาดเคลื่อนของแต่ละองศา

9. การประมวลผลภาพและการปรับปรุงลายเส้น
รูปร่างต่าง ๆ

หลังจากทำการแปลงพิกัดภาพจะได้ผลดัง
ตารางที่ 6 ซึ่งแสดงค่าความคลาดเคลื่อนของชิ้นงานแต่
ละองศา พบว่าเมื่อปรับมุมกลิ้งที่ 45 องศา จะให้ค่า

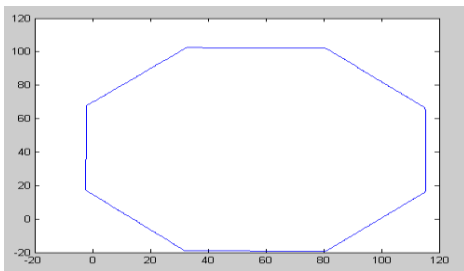
AMM-167

ความแม่นยำน้อยที่สุด และที่มุม 90 องศาให้ค่าความแม่นยำมากที่สุด โดยมีค่าความคลาดเคลื่อนเฉลี่ย 0.94, 0.34 มม. ตามลำดับ ทั้งนี้ผู้วิจัยได้ทำการศึกษาการแปลงตำแหน่งพิกัดเพิ่มเติม โดยใช้ชิ้นงานที่มีรูปร่างที่หลากหลายมากขึ้นอีก 2 รูปร่าง ได้แก่ รูปร่างแปดเหลี่ยม และรูปร่างอิสระ โดยมีผลการประมวลผลภาพดังต่อไปนี้

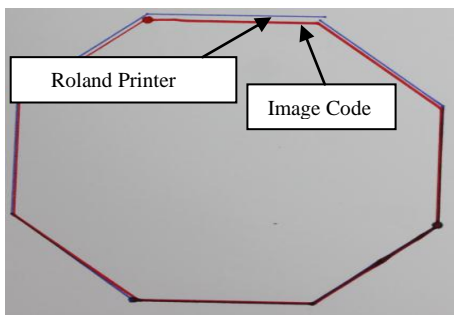
9.1 การประมวลผลภาพและการปรับปรุงลายเส้นรูปร่างแปดเหลี่ยม และรูปร่างอิสระ

9.1.1 จากรูปร่างแปดเหลี่ยมนำพิกัดที่ได้จากเซนโค้ดมาคูณกับสมการที่ 10 เพื่อแปลงเป็นพิกัดจริง แสดงดังรูปที่ 18

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} 0.0025 & 0.0000 & -0.1049 \\ 0.0000 & -0.0026 & 0.9944 \\ 0.0000 & 0.0000 & 0.0089 \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix} \quad (10)$$



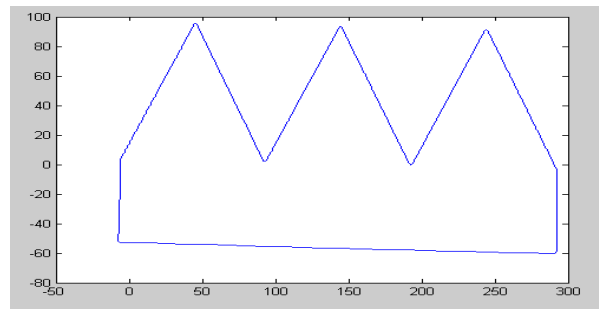
รูปที่ 18 แสดงภาพหา Homography Transform



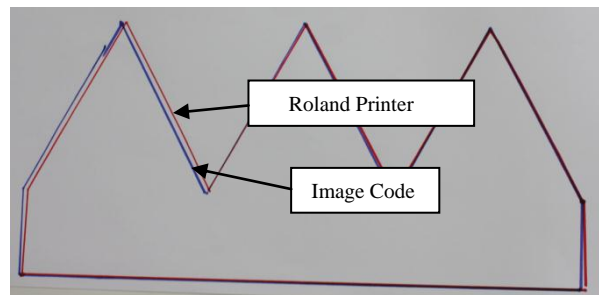
รูปที่ 19 แสดงภาพรหัสทางเดินจากโปรแกรมอัตโนมัติ เทียบกับ รหัสทางเดินจากการแปลงภาพ

9.1.2 จากรูปร่างอิสระนำพิกัดที่ได้จากเซนโค้ดมาคูณกับสมการที่ 11 เพื่อแปลงเป็นพิกัดจริง แสดงดังรูปที่ 20

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} 0.0032 & -0.0000 & -0.1140 \\ 0.0001 & -0.0032 & 0.9634 \\ 0.0000 & -0.0000 & 0.0089 \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix} \quad (11)$$



รูปที่ 20 แสดงภาพหาโฮโมกราฟฟีทรานฟอร์ม



รูปที่ 21 แสดงภาพรหัสทางเดินจากโปรแกรมอัตโนมัติ เทียบกับ รหัสทางเดินจากการแปลงภาพ

11. สรุป

งานวิจัยนี้นำเสนอวิธีการอย่างง่ายและมีประสิทธิภาพสำหรับสร้างรหัสคำสั่งที่ใช้ได้กับเครื่องกัดเครื่องพล็อตเตอร์ และเครื่องหยอดกาว ซึ่งมีลักษณะเป็นเส้นทางเดิน โดยใช้คอมพิวเตอร์วิชันและการประมวลผลภาพดิจิทัลจากภาพถ่ายของวัตถุต้นแบบวิธีการที่จะนำเสนอนี้จัดเป็นวิศวกรรมย้อนรอยสำหรับใช้งานกับการสร้างเส้นทางเดินจากภาพสองมิติ โดยการค้นหาขอบภาพของชิ้นงานแล้วใช้วิธีการสเกิร์ตัน (Skeleton) เพื่อลดขนาดของขอบภาพให้เหลือขอบพิกเซลเดียว ซึ่งจะเป็นทางเดินของเครื่องพล็อตเตอร์อัตโนมัติอย่างง่าย จากนั้นแปลงพิกัดด้วยวิธี โฮโมกราฟฟีทรานฟอร์ม (Homography Transform) เพื่อแก้ไขมุมมองของขอบภาพแบบค้นหาตำแหน่งในการเปรียบเทียบพิกัดทั้ง 4 จุด หลังจากนั้นใช้วิธีการเซนโค้ด (Chain Code) ในการแปลงเส้นทางเดินของเครื่องจักรแบบอัตโนมัติ จากการทดลองพบว่าที่มุมกล้อง 90 องศา จะให้ความแม่นยำและเที่ยงตรงมาก

AMM-167

ที่สุด และเมื่อเทียบกับวิธีป้อนพิกัดแบบเรียนรู้ตำแหน่ง (Teach Point) และวิธีเลือกตำแหน่งบนภาพ 2 มิติ[6] ยังให้ผลที่รวดเร็วกว่าและแม่นยำมากที่สุด หลังจากนั้นทำการทดลองแบบปรับเปลี่ยนรูปร่างของวัตถุตัวอย่างที่มีรูปร่างแปดเหลี่ยม และรูปร่างอิสระจากการทดลองในการแปลงรหัสทางเดินสำหรับเครื่องกัด เครื่องหยอดกาวและเครื่องพอลอตเตอร์ด้วยคอมพิวเตอร์วิชันจะได้พิกัดตำแหน่งมีค่าใกล้เคียงกับการแปลงรหัสทางเดินที่แปลงจากโปรแกรมอัตโนมัติ

12. เอกสารอ้างอิง

- [1] Research on Intelligent Glue-coating Robot Based on Visual ServoPhysics Procedia, Volume 24, Part C, 2012, Pages2165-2171Zhang Yang, Yi An, Yanbin Sun, Jingchun Zhang
- [2] I. Ardpairin and T. Suesut, 2012. "Inspection depth of grooves on tire using 3D computer vision system." Proceedings of 50th Kasetsart University Annual Conference: Architecture and Engineering
- [3] Suesut, T. and Inthuset.S, 2009. "Fish Size Measurement by Computer Vision using Light Sectioning." International Conference on Innovations in Agricultural Food and Renewable Energy Productions for Mankind.
- [4] Min Max Threshold Range (MMTR) based approach in palmpoint authentication by sobel code method Original Research Article Procedia Computer Science,Volume 2,2010, Pages149-158 Jyoti Malik, G. Sainarayanan, Ratna Dahiya
- [5] Using skeleton-based tracking to increase the reliabilityof optical motioncapture Human Movement Science, Volume 20, Issue 3, June 2001, Pages 313-341Lorna Herda, Pascal Fua, Ralf Plänklers, Ronan Boulic, Daniel Thalmann
- [6] การสร้างรหัส G-Code ของเส้นทางการหยอดปะเก็นเหลวสำหรับเครื่องยนต์ด้วยคอมพิวเตอร์วิชัน, การประชุมวิชาการเครือข่ายวิศวกรรมเครื่องกลแห่ง

ประเทศไทย ครั้งที่ 27, วันที่ 16-18 ตุลาคม 2556, วิชาญ พัฒนาธาดากุล,สุเทพ อาษา,รศ.ดร.ทวีพล ชื้อสัตย์,รศ.สุพรรณ กุลพาณิชย์

[7] Feature extraction from vein images using spatial information and chain codes original Research Article Information Security Technical Report, Volume 17,Issues 1-2,February 2012,Pages 26-35 Anika Pflug, Daniel Hartung,Christoph Bussch