

การเปรียบเทียบวิธีการหาปลายนิ้วและร่องนิ้วบนพื้นฐานของการประมวลผลภาพ

A Comparative of the Methods to Find Fingertips and Valleys Based on Images Processing

อภิวัฒน์ สวัสดิรัตน์ และ จักรี ศรีนนท์ฉัตร
ห้องปฏิบัติการและวิจัยทางด้านการประมวลผลสัญญาณ
ภาควิชาวิศวกรรมอิเล็กทรอนิกส์และโทรคมนาคม คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี
E-mails: apiwat.sa@rmuti.ac.th, jakkree.s@en.rmutt.ac.th

บทคัดย่อ

งานวิจัยทางด้านระบบวัดโครงสร้างมือด้วยการประมวลผลภาพส่วนใหญ่มุ่งเน้นไปที่ความแม่นยำในการระบุตัวตน โดยอาศัยคุณลักษณะของนิ้ว งานวิจัยนี้จึงมุ่งเน้นในการเปรียบเทียบวิธีการหาปลายนิ้วและร่องนิ้วจากเส้นรอบรูปมือเพื่อนำไปประยุกต์ใช้ในระบบสมองกลฝังตัวซึ่งมีทรัพยากรจำกัด โดยทดสอบด้วยการสร้างโปรแกรมที่พัฒนาด้วยภาษา C++ และใช้ gcc4 เป็นคอมไพเลอร์ร่วมกับฟังก์ชันการประมวลผลภาพ OpenCV 2.1 ทำการประมวลผลบน Samsung S3C2440A ARM920T ผลการทดลองพบว่าวิธีวัดรัศมีจากเส้นรอบรูปมือมีข้อดีในด้านความยืดหยุ่นสูงสามารถรับข้อมูลที่มีขนาดแตกต่างกันมากๆ ได้ ด้านวิธีการวัดหาความโค้งจากเส้นรอบรูปมือมีข้อดีที่ความเร็วในการประมวลผล ส่วนวิธีหากรอบด้านนอกและความยาวจากเส้นรอบรูปมือเป็นวิธีที่นำข้อดีของสองวิธีข้างต้นมารวมกัน จึงเป็นข้อดีและเหมาะสมที่จะนำมาใช้ในระบบสมองกลฝังตัว

คำสำคัญ

ระบบวัดโครงสร้างมือ เส้นรอบรูปมือ การหาปลายนิ้ว

Abstract

Most of hand geometry system research using image processing is focused to provide the accuracy identify user which used the characteristic of finger. This research focuses to comparative the methods of the detection finger tips and valleys from the hand contours for apply to embedded system which it is a resource limited. In the experiment, the program is developed by using C++ language and compiled with gcc4 compiler and OpenCV 2.1. This experiment is also operated on

Samsung S3C2440A ARM920TS. The primary results show the advantage of the radius distance from the hand contour is provided the optimize solution for the large scale different of the hand data. The advantage of the curvature of the hand contour is provided the fastest computation processing. The hull convexity defect of the hand contour is taken the advantage of the both methods. It is an optimize method to apply for the embedded system.

Keywords

hand geometry system, hand contour, fingertips detection

1. บทนำ

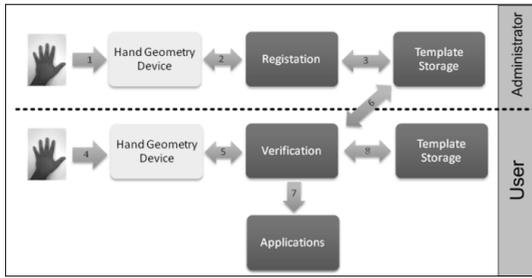
งานวิจัยด้านระบบวัดโครงสร้างมือ (Hand Geometry System) ส่วนใหญ่มุ่งเน้นไปที่การระบุตัวตน (Identification) ที่ต้องการความแม่นยำสูง จึงใช้ทรัพยากรในการประมวลผลมาก [1-4] อีกทั้งระบบที่มีวางจำหน่ายก็มีราคาแพง เนื่องจากมีเพียงไม่กี่บริษัทที่พัฒนาระบบในเชิงพาณิชย์ ผู้วิจัยจึงได้มีแนวคิดที่จะพัฒนาต้นแบบระบบวัดโครงสร้างมือเพื่อการยืนยันเอกลักษณ์ (Verification) ด้วยเทคโนโลยีสมองกลฝังตัวในแบบระบบเปิด เพื่อเป็นอีกทางเลือกนอกเหนือจากระบบในเชิงพาณิชย์ โดยงานวิจัยนี้จะมุ่งเน้นไปที่การเปรียบเทียบเทคนิคการหาปลายนิ้วและร่องนิ้วจากเส้นรอบรูปมือ (Hand Contour) แบบต่างๆ ในด้านความเร็วในการประมวลผล พร้อมทั้งหาปัจจัยที่มีผลกระทบกับความถูกต้องของพิกัดปลายนิ้วและร่องนิ้วในแต่ละวิธีการ ซึ่งการหาปลายนิ้วและร่องนิ้วนี้เป็นกระบวนการสำคัญส่วนหนึ่งในอุปกรณ์วัดโครงสร้างมือ (Hand Geometry Device) โดยผลลัพธ์จากกระบวนการนี้จะนำไปหาความกว้างและความยาวของนิ้วมือแต่ละนิ้วต่อไป จากการค้นหาข้อมูลงานวิจัยด้านนี้พบว่าวิธีที่นิยมใช้อยู่สองวิธีคือ วิธีวัดรัศมีจากเส้นรอบรูปมือ ดังในงานวิจัย [5-10] กับวิธีหาความโค้งจากเส้นรอบรูปมือ ดังในงานวิจัย [3][11-

13] พร้อมทั้งปรับปรุงวิธีหากรอบด้านนอกและความเว้าจากเส้นรอบรูปมือของงานวิจัย [3][14] ด้านระบบสัญญาณมือ (Hand Gesture System) ให้สามารถระบุปลายนิ้วและร่องนิ้วได้อีกรูปแบบหนึ่ง

2. ทฤษฎีที่เกี่ยวข้อง

2.1 ระบบวัดโครงสร้างมือเพื่อการยืนยันเอกลักษณ์

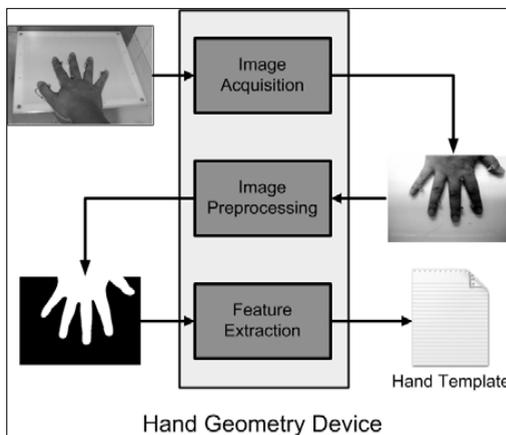
ระบบวัดโครงสร้างมือเพื่อการยืนยันเอกลักษณ์มีองค์ประกอบและกระบวนการโดยทั่วไปดังภาพที่ 1 ซึ่งได้ข้อมูลมาจาก [15] เริ่มจากกระบวนการ {1} นำมือเข้าอุปกรณ์วัดโครงสร้างมือ {2} ทำการประมวลผลหาลักษณะเด่นเพื่อสร้างแม่แบบพร้อมขึ้นทะเบียนผู้ใช้ {3} นำแม่แบบบันทึกลงบนหน่วยเก็บข้อมูลบนเครือข่ายกลาง หรือบนสื่อบันทึกข้อมูลอื่นๆ เช่น สมาร์ทการ์ด {4} นำมือเข้าอุปกรณ์วัดโครงสร้างมือเพื่อใช้งานระบบ {5} ประมวลผลหาลักษณะเด่นเพื่อสร้างแบบ {6} เปรียบเทียบแบบที่ได้กับแม่แบบจากหน่วยเก็บข้อมูล {7} นำผลการเปรียบเทียบส่งต่อไปให้ประโยชน์ต่อไป {8} บันทึกข้อมูลที่จำเป็นในด้านความปลอดภัยเพื่อนำไปตรวจสอบ



ภาพที่ 1 องค์ประกอบและกระบวนการระบบวัด
โครงสร้างมือ

2.2 อุปกรณ์วัดโครงสร้างมือ

อุปกรณ์วัดโครงสร้างมือทำหน้าที่รับภาพมือของผู้ใช้และหาลักษณะเด่นออกมา เพื่อส่งต่อไปยังกระบวนการยืนยันเอกลักษณ์มีขั้นตอนการทำงานหลักแบ่งเป็นสามส่วนคือ การเตรียมภาพ (Image Acquisition) การประมวลผลภาพเบื้องต้น (Image Preprocessing) และการหาลักษณะเด่น (Feature Extraction) ดังภาพที่ 2



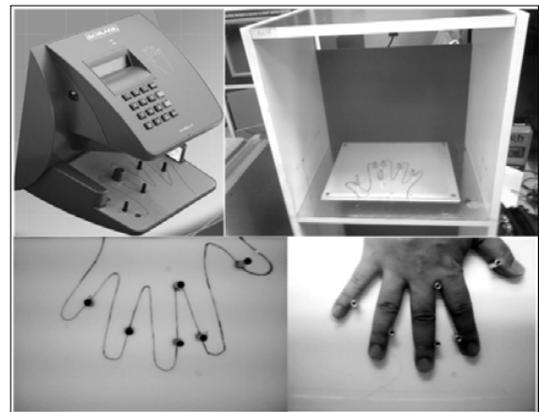
ภาพที่ 2 ขั้นตอนการทำงานหลักของอุปกรณ์วัด
โครงสร้างมือ

3. ขั้นตอนวิธีการดำเนินการวิจัย

3.1 การเตรียมภาพ

ในขั้นตอนการเตรียมภาพจากการศึกษางานวิจัยที่ผ่านมาได้เลือกใช้อุปกรณ์ในการรับภาพเป็นสอง

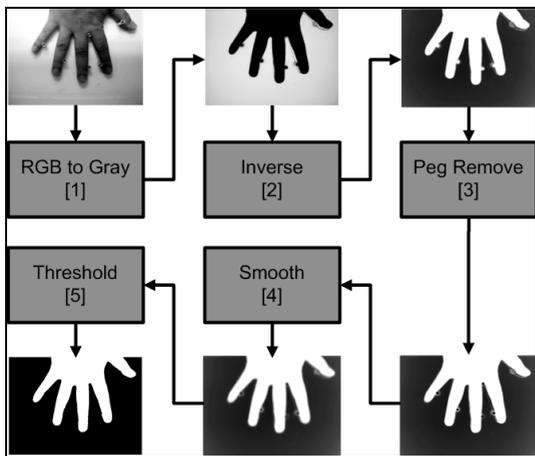
ประเภทคือ ประเภทที่หนึ่งใช้กล้อง CCD ในการรับภาพ เช่น [3], [5-8] และ [12] ทั้งนี้เพื่อเป็นการลดขนาดของอุปกรณ์วัดโครงสร้างมือ จึงลดระยะห่างระหว่างกล้องและแผ่นวางฝ่ามือ จึงทำให้เห็นภาพมือในส่วน นิ้วก้อย นิ้วนาง นิ้วกลาง และนิ้วชี้เป็นหลัก พร้อมทั้งมีหมุดกำกับฝ่ามือ เนื่องจากข้อจำกัดของกล้องซึ่งใช้เลนส์ในการรับภาพซึ่งมีความโค้ง ทำให้ตำแหน่งของมือมีผลต่อสัดส่วนของภาพที่ได้ซึ่งต่างจากงานวิจัยประเภทที่สองที่ใช้เครื่องสแกนเอกสารในการรับภาพเช่น [3], [6] และ [7] จะใช้ภาพที่มีส่วนมือนิ่งตั้งแต่ข้อมือขึ้นไป และภาพที่ได้จะได้สัดส่วนที่เท่ากันไม่ว่าจะอยู่ตำแหน่งใดในแผ่นรับภาพ โดยในงานวิจัยนี้ได้ใช้กล้องเว็บแคม (Web Camera) บันทึกภาพมือขวาจากตู้ควบคุมแสงสว่าง ซึ่งมีหมุด 6 หมุดช่วยบังคับมือให้อยู่ในตำแหน่งที่ต้องการ ภาพที่ได้มีขนาด 640 x 480 พิกเซล มีความชัดเจนของนิ้วมือทั้ง 4 นิ้วคือ นิ้วก้อย นิ้วนาง นิ้วกลาง และนิ้วชี้ คล้ายกับผลิตภัณฑ์เชิงพาณิชย์ดังภาพที่ 3



ภาพที่ 3 (ซ้ายบน) ผลิตภัณฑ์ Schlage HandKey II (ขวาบน) ตู้ควบคุมแสง (ซ้ายล่าง) แผ่นรองมือพร้อมหมุด (ขวาล่าง) ตัวอย่างภาพมือที่ได้

3.2 การประมวลผลภาพเบื้องต้น

เนื่องจากภาพที่ได้จากระบบการก่อนหน้าเป็นภาพสี RGB ต้องนำมาแปลงเป็นระดับสีเทา (RGB-to-Gray) พร้อมทั้งกลับสีระดับสีเทา (Inverse) เพื่อส่งต่อไปยังการลบหมุด (Peg Remove) เพื่อลบภาพหมุดทั้ง 6 ที่ใช้บังคับตำแหน่งมือ โดยขบวนการนี้ผู้ทดลองได้นำภาพจากขั้นตอนการกลับสีระดับสีเทามาหาตำแหน่งจุดศูนย์กลางและรัศมีของแต่ละหมุด จากนั้นใช้ฟังก์ชันวาดภาพวงกลมในส่วนของกราฟิกไลบรารีวาดทับลงในตำแหน่งหมุดที่ได้ จากนั้นกำหนดตำแหน่งและขนาดรัศมีเป็นค่าคงที่ในโปรแกรมเพื่อใช้กับภาพอื่นต่อไป ส่วนภาพที่ได้จากการลบหมุดแล้วจะนำมาปรับเรียบ (Smooth) ต่อไป ซึ่งขบวนการปรับเรียบนี้จะเกลี่ยพิกเซลขนาดเล็กที่ไม่ต้องการที่กระจายอยู่ในภาพให้กลืนไปกับพิกเซลใกล้เคียง และกระบวนการสุดท้ายคือแปลงภาพเป็นขาวดำ (Threshold) ดังภาพที่ 4

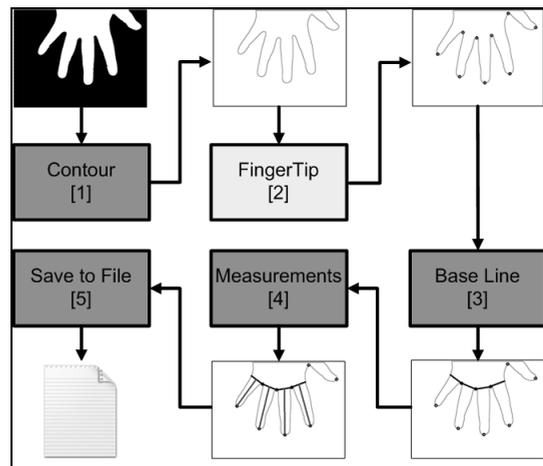


ภาพที่ 4 ขั้นตอนการทำงานย่อยของการภาพประมวลผลเบื้องต้น

3.3 การหาลักษณะเด่น

ในขั้นตอนนี้จะนำภาพขาวดำที่ได้จากระบบการประมวลผลเบื้องต้นเพื่อให้ได้ลักษณะเด่นของมือในแต่ละบุคคลออกมาในรูปแบบของชุดข้อมูลเพื่อ

ประหยัดพื้นที่ในการการบันทึกและความสะดวกในการนำไปเปรียบเทียบ เช่น ความกว้างและความยาวของแต่ละนิ้ว อัตราส่วนระหว่างนิ้วกับฝ่ามือและความหนาของมือ เป็นต้น ซึ่งจะแตกต่างกันไปในแต่ละงานวิจัยนั้นๆว่าต้องการความถูกต้องแม่นยำหรือความผิดพลาดต้งไว้ในระดับใด โดยในงานวิจัยนี้จะนำไปประยุกต์ใช้ในการยืนยันเอกลักษณ์บุคคลด้วยเทคโนโลยีสมองกลฝังตัวในแบบระบบเปิด จึงแบ่งขั้นตอนย่อยของการหาลักษณะเด่นออกเป็นห้าขั้น ดังนี้ การหาเส้นรอบรูปมือ (Contour) การหาปลายนิ้วและร่องนิ้วมือ (Finger Tip) การหาเส้นฐานของนิ้ว (Base Line) การหาความยาวของนิ้ว (Measurements) และการสร้างแม่แบบ (Template) ดังภาพที่ 5

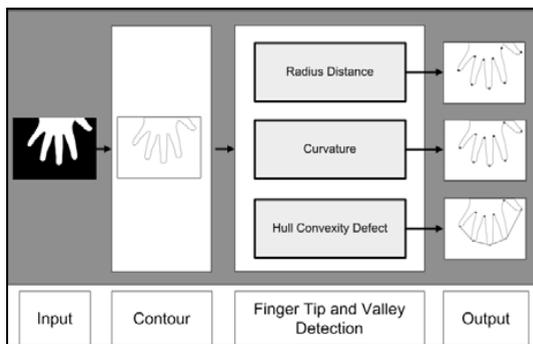


ภาพที่ 5.ขั้นตอนการทำงานย่อยของการหาลักษณะเด่น

3.4 วิธีการหาปลายนิ้วและร่องนิ้วจากเส้นรอบรูปมือ

จากที่ได้เกริ่นนำในบทนำแล้วว่างานวิจัยนี้จะเน้นไปที่การเปรียบเทียบวิธีการหาปลายนิ้วและร่องนิ้วจากเส้นรอบรูปมือในสามวิธีที่นิยมใช้คือ วิธีรัศมี (Radius Distance) จากเส้นรอบรูปมือ วิธีหาความโค้ง (Curvature) จากเส้นรอบรูปมือ และวิธีหากรอบ

ด้านนอกและความเว้าจากเส้นรอบรูปมือ (Hull Convexity Defects) จึงได้ขั้นตอนการทดลองเพื่อเปรียบเทียบหาผลลัพธ์ของทั้งสามวิธีดังภาพที่ 6 ทั้งสามวิธีจะใช้ข้อมูลเส้นรอบรูปมือเดียวกันเพื่อนำไปหาปลายนิ้วและร่องนิ้ว และแสดงผลลัพธ์ออกมาเป็นภาพที่ระบุจุดปลายนิ้วและร่องนิ้ว ทั้งนี้เพื่อความสะดวกในการตรวจสอบความถูกต้องของตำแหน่งปลายนิ้วและร่องนิ้วด้วยบุคคล จากนั้นเปลี่ยนอินพุตไปเรื่อยๆ พร้อมทั้งสังเกตและบันทึกผลลัพธ์ ทำไปจนกระทั่งได้ข้อมูลเพียงพอ จากนั้นนำผลลัพธ์มาวิเคราะห์หาบทสรุปของการทดลอง



ภาพที่ 6 ขั้นตอนการทดลองเพื่อเปรียบเทียบหาผลลัพธ์ของทั้งสามวิธี

3.5 วิธีวัดรัศมีจากเส้นรอบรูปมือ

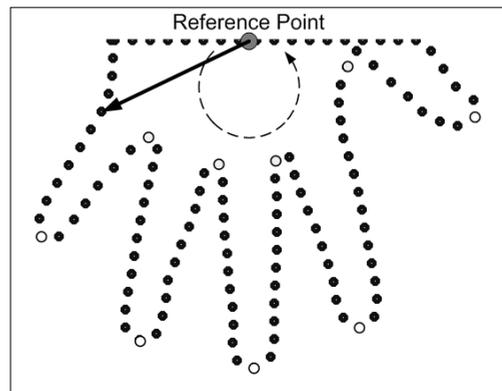
จากข้อมูลเส้นรอบรูปมือที่ได้นำมาหาจุดอ้างอิงหลัก (reference point) โดยในการทดลองนี้ใช้จุด $X_{ref} = X$ of Maximum $Y, Y_{ref} = 0$ ในระบบพิกัดฉาก (Cartesian coordinates) เนื่องจากภาพอินพุตเป็นแบบมีหมุดกำกับทำให้ค่าสูงสุดของแกน Y จะเป็นปลายนิ้วกลาง และแกน X ของจุดนี้จะเป็นตำแหน่งแบ่งครึ่งฝ่ามือ เมื่อได้จุดอ้างอิงหลักแล้วให้แปลงข้อมูลเส้นรอบรูปมือจากระบบพิกัดฉากเป็นระบบพิกัดเชิงขั้ว (Polar coordinates) ซึ่งประกอบไปด้วยค่ารัศมีและค่ามุม ดังสมการที่ (1) และ (2) จุดที่มีค่ารัศมีสูงที่สุดก่อนที่ค่ารัศมีจะลดลงจะเป็นตัวบ่งชี้ว่าจุดนั้นเป็น

ปลายนิ้ว ส่วนจุดที่มีค่ารัศมีต่ำที่สุดก่อนที่ค่าจะเพิ่มขึ้นจะเป็นตัวบ่งชี้ว่าจุดนั้นเป็นร่องนิ้ว ผลลัพธ์ที่ได้ดังภาพที่ 7

$$radius = \sqrt{|X_i - X_{ref}|^2 + |Y_i - Y_{ref}|^2} \quad (1)$$

$$\theta = \tan^{-1} \left(\frac{Y_i - Y_{ref}}{X_i - X_{ref}} \right) \quad (2)$$

เมื่อ i คือ ตำแหน่งลำดับของเส้นรอบรูปมือ
 ref คือ จุดอ้างอิงหลัก



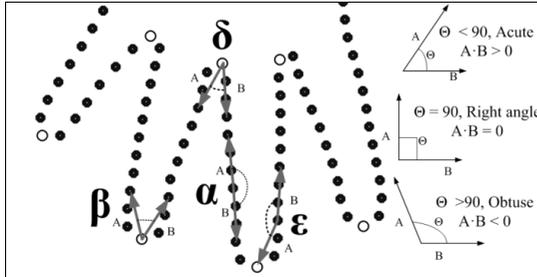
ภาพที่ 7 ความสัมพันธ์ระหว่างเส้นรอบรูปมือและตำแหน่งปลาย/ร่องนิ้ว

3.6 วิธีหาความโค้งจากเส้นรอบรูปมือ

หากกำหนดให้เส้นรอบรูปมือแทนด้วย C ที่มีจำนวนจุดทั้งหมด n จุด และแต่ละจุดแทนด้วย i และ k คือลำดับของจุดที่ห่างจากจุด i สามารถหาค่าความโค้งได้โดยการ dot product ระหว่างสองเวกเตอร์ คือ $[C_i, C_{i-k}]$ และ $[C_i, C_{i+k}]$ หรือเวกเตอร์ A และ B ตามลำดับดังสมการที่ (3) และ (4) จากภาพที่ 8 เป็นตัวอย่างการหาความโค้งของจุดดัชนี ซึ่งกำหนดให้ค่า $k = 3$ จะได้จุดปลายนิ้วและจุดร่องนิ้วคือจุด β และ δ ซึ่งมีค่าเข้าใกล้ 90 องศาที่สุด ก่อนจะห่างออกไปส่วนจุด α และ ϵ จะไม่ถูกระบุ

$$A \perp B = |A| |B| \cos \theta \tag{3}$$

$$\theta = \cos^{-1} \frac{A \perp B}{|A||B|} \tag{4}$$

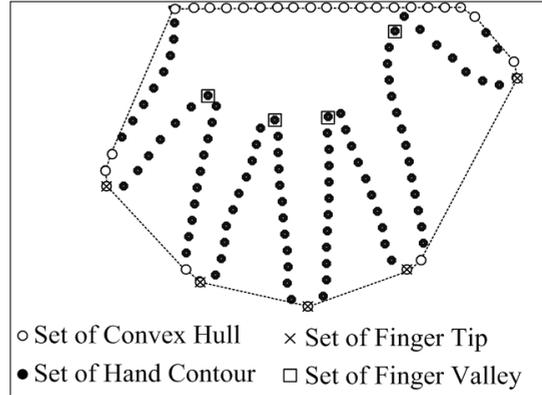


ภาพที่ 8 ความสัมพันธ์ระหว่างความโค้งดัชนี และปลาย/ร่องนิ้ว

3.7 วิธีหากรอบด้านนอกและความเว้าจากเส้นรอบรูปมือ

จากข้อมูลเส้นรอบรูปมือที่ได้ขั้นตอนแรกให้นำมาหากรอบด้านนอก หรือ Convex Hull ซึ่ง Convex Hull ของเส้นรอบรูปมือ คือ เซตย่อยที่เล็กที่สุดที่สามารถครอบคลุมทุกจุดของเส้นรอบรูปมือ โดยให้เซตของเส้นรอบรูปมือคือทุกๆจุดบนรูปและ Convex Hull ของเส้นรอบรูปมือคือเซตของจุดวงกลมสีขาว โดยมีเส้นประแสดงให้ดูว่าเซตของจุดสีขาวสามารถครอบคลุมทุกจุดบนเส้นรอบรูปมือ ขั้นตอนที่สองให้นำเซตของ Convex Hull มาหาจุดระบุตำแหน่งร่องนิ้ว โดยจุดนี้คือจุดที่แทรกอยู่ระหว่างจุด Convex Hull ที่มีลำดับในเซตติดกันในเซตของ Convex Hull ซึ่งจุดนี้เมื่อคำนวณหาระยะทางจากจุด Convex Hull ตั้งต้นผ่านจุดที่ต้องการระบุค่าและสิ้นสุดที่จุด Convex Hull ปิดท้ายแล้วได้ความยาวมากที่สุด จากสมการที่ (5) (6) (7) ผลที่ได้แสดงด้วยรูปสี่เหลี่ยมจากกระบวนการข้างต้นสามารถระบุตำแหน่งปลายนิ้วก็ง่ายและนิ้วนางได้คือจุด Convex Hull ตั้งต้น ส่วนตำแหน่งปลายนิ้วชี้และนิ้วโป้งคือจุด Convex Hull ปิดท้าย จากนั้นหาจุดระบุตำแหน่งปลายนิ้วกลางซึ่ง

ต้องอยู่กึ่งกลางระหว่างจุด Convex Hull ปิดท้ายนิ้วนาง และจุด Convex Hull ตั้งต้นนิ้วชี้ ดังแสดงด้วยรูปกากบาท ดังตัวอย่างในภาพที่ 9



ภาพที่ 9 ความสัมพันธ์ระหว่างเส้นรอบรูปมือ, กรอบด้านนอก, ตำแหน่งปลายนิ้วและตำแหน่งร่องนิ้ว

$$length_{st} = \sqrt{|CX_{st} - X_i|^2 + |CY_{st} - Y_i|^2} \tag{5}$$

$$length_{end} = \sqrt{|CX_{end} - X_i|^2 + |CY_{end} - Y_i|^2} \tag{6}$$

$$length = length_{st} + length_{end} \tag{7}$$

เมื่อ C คือ เซตของ Convex Hull

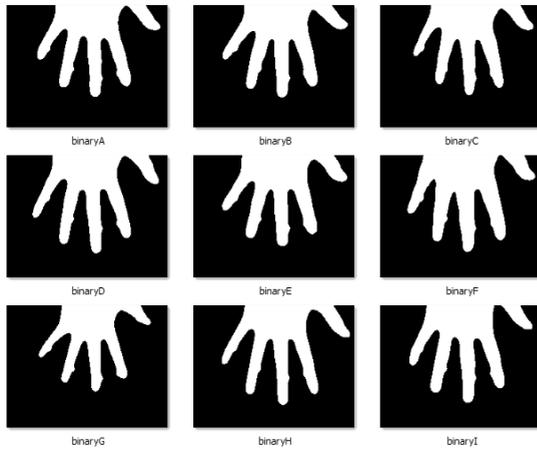
i คือ ตำแหน่งลำดับของเส้นรอบรูปมือ

st คือ จุดตั้งต้น ; end คือ จุดปิดท้าย

4. ผลการดำเนินการ

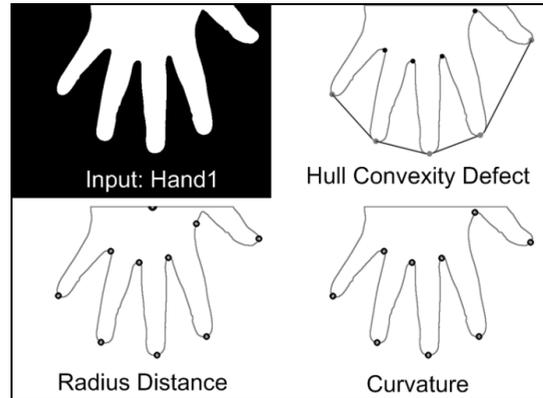
จากหลักการที่กล่าวมาข้างต้นทั้งสามวิธีนำมาเขียนเป็นโปรแกรมโดยกำหนดให้รับข้อมูลจากภาพมือแบบขาวดำ ขนาด 640x480 พิกเซล ที่ลบหมุดกำกับนิ้วมือแล้ว และผลลัพธ์ของโปรแกรมให้แสดงภาพรูปมือพร้อมแสดงจุดปลายนิ้วและร่องนิ้ว ระบบถูกพัฒนาด้วยภาษา C++ ใช้คอมไพเลอร์ gcc4.4.1

ร่วมกับฟังก์ชันการประมวลผลภาพของ OpenCV 2.1 บนเครื่องคอมพิวเตอร์ที่ติดตั้งระบบปฏิบัติการ Ubuntu 10.04 Desktop มีหน่วยประมวลผลกลาง Intel Pentium Dual ความเร็ว 3 GHz หน่วยความจำหลักขนาด 4 GB แบบ DDR2 เพื่อให้ได้โปรแกรมต้นแบบของทั้ง 3 วิธี นำภาพถ่ายมือที่ได้จากนักศึกษาจำนวน 30 ตัวอย่าง ดังภาพที่ 10



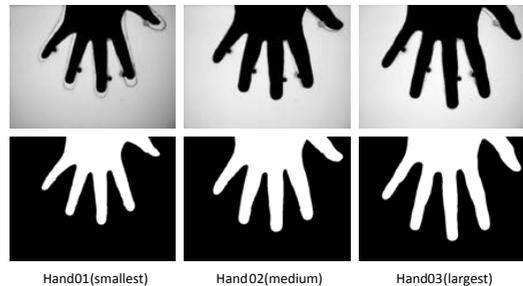
ภาพที่ 10 รูปมือบางส่วนที่ใช้ในการทดสอบ

เพื่อทดสอบโปรแกรมต้นแบบ โดยโปรแกรมต้องให้ผลลัพธ์ที่ถูกต้องครบทั้ง 30 ตัวอย่าง พบว่าในวิธีวัดรัศมีจากเส้นรอบรูปมือในกระบวนการหาจุดอ้างอิงได้ค่า X_{ref} อยู่ระหว่าง 322 ถึง 336, $Y_{ref} = 0$ ส่วนวิธีหาความโค้งจากเส้นรอบรูปมือได้กำหนดค่าในกลุ่มทดสอบนี้คือใช้ค่า $k = 35$ และ $A \cdot B$ อยู่ระหว่าง 550 ถึง -550 และวิธีหากรอบด้านนอกและความเว้าจากเส้นรอบรูปมือได้ระบุค่าความยาวของนิ้วที่สั้นที่สุดในกลุ่มทดสอบนี้ใช้ค่า 60 พิกเซล หรือประมาณ 20 มิลลิเมตร ซึ่งตัวอย่างภาพที่ได้แสดงในภาพที่ 11



ภาพที่ 11 ตัวอย่างภาพผลลัพธ์จากทั้งสามวิธี

จากนั้นเลือกภาพมือที่มีขนาดเส้นรอบรูปมือที่สั้นที่สุด, ขนาดปานกลางและขนาดยาวที่สุดจากชุดข้อมูลตัวอย่างดังภาพที่ 12 นำมาหาเวลาในการประมวลผลหาปลายนิ้วโดยเฉลี่ยจากเส้นรอบรูปมือทั้งสามขนาด



ภาพที่ 12 รูปมือที่ใช้ในการทดสอบทั้งสามขนาด

เมื่อได้เวลาในการประมวลผลจากเครื่องคอมพิวเตอร์ข้างต้นแล้ว นำมาคอมไพล์ด้วย GCC Toolchain (gcc 4.6.2, libc 2.14.1) เพื่อนำไปใช้กับระบบปฏิบัติการ Free Linux BSP for FriendlyARM's Mini2440 ซึ่งใช้หน่วยประมวลผลกลาง Samsung S3C2440A ARM9 ความเร็ว 400 MHz โดยใช้หน่วยความจำหลักแบบ SDRAM ขนาด 64 MB และบน FriendlyARM's Tiny6410 ซึ่งใช้หน่วย

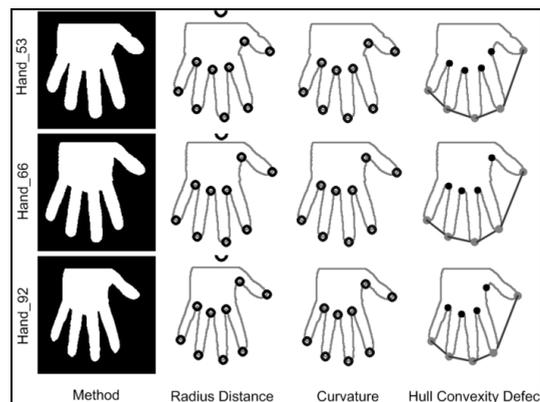
ประมวลผลกลาง Samsung S3C6410A ARM11 ความเร็ว 533 MHz โดยใช้หน่วยความจำหลักแบบ SDRAM ขนาด 64 MB เวลาที่ใช้ในการประมวลผลหาปลายนิ้วโดยเฉลี่ยจากเส้นรอบมือทั้งสามขนาด ได้ผลดังตารางที่ 1 ตารางที่ 1 เวลาที่ใช้ในการประมวลผลบนหน่วยประมวลผลต่างๆ

	Contour pixels	Radius Distance	Curvature	Hull Convexity Defects
CPU: Intel Pentium Dual Core E5700 Speed: 3 GHz		RAM: 4GB		
Hand01 (smallest)	2150	0.9 msec.	0.5 msec.	0.7 msec.
Hand02 (medium)	2450	0.9 msec.	0.5 msec.	0.7 msec.
Hand03 (largest)	2800	0.9 msec.	0.5 msec.	0.7 msec.
CPU: ARM9 Samsung S3C2440 Speed: 400 MHz		RAM: 64MB		
Hand01 (smallest)	2150	18 msec.	9.4 msec.	12 msec.
Hand02 (medium)	2450	20 msec.	9.6 msec.	13 msec.
Hand03 (largest)	2800	21 msec.	10 msec.	15 msec.
CPU: ARM11 Samsung S3C6410 Speed: 533 MHz		RAM: 256MB		
Hand01 (smallest)	2150	11.0 msec.	6.4 msec.	8 msec.
Hand02 (medium)	2450	12.0 msec.	6.5 msec.	9 msec.
Hand03 (largest)	2800	12.5 msec.	6.7 msec.	10 msec.

ทดสอบโปรแกรมเพิ่มเติมอีก100 ตัวอย่างด้วยชุดข้อมูลรูปร่างมือ (Hand Shape Database) ของ Prof. BÜLENT SANKUR จาก Bogazici University ผู้เขียนงานวิจัย [4], [9-10] โดยเป็นภาพมือแบบขาวดำ ขนาด 200x200 พิกเซล โดยโปรแกรมต้องให้ผลลัพธ์ที่ถูกต้องทั้งหมด พบว่าในวิธีวัดรัศมีจากเส้นรอบรูปมือไม่ต้องปรับค่าใดๆ ส่วนวิธีหาความโค้งจากเส้นรอบรูปมือได้กำหนดค่าในกลุ่มทดสอบนี้คือใช้ค่า $k = 15$ และ $A \cdot B$ อยู่ระหว่าง 100 ถึง -100 และวิธีหากรอบด้านนอกและความเว้าจากเส้นรอบรูปมือได้ระบุค่าความยาวของนิ้วที่สั้นที่สุดในกลุ่มทดสอบนี้ใช้ค่า 20 พิกเซล ซึ่งตัวอย่างภาพที่ได้แสดงในภาพที่ 13

จากการวิเคราะห์หาปัจจัยที่มีผลกระทบต่อวิธีการพบว่าในวิธีวัดรัศมีจากเส้นรอบรูปมือนั้นไม่ต้องการการระบุค่าเบื้องต้นใดๆของนิ้วมือ เนื่องจากมีกระบวนการหาจุดอ้างอิงหลักช่วยในส่วนนี้แล้ว ส่วนวิธีหาความโค้งจากเส้นรอบรูปมือต้องกำหนดค่า k ซึ่งต้อง

มีระยะมากกว่าความกว้างของนิ้ว แต่ต้องไม่เกินความยาวนิ้วซึ่งวิธีนี้หากขนาดมือมีความแตกต่างกันมากจะเกิดข้อผิดพลาดในการระบุตำแหน่ง และวิธีหากรอบด้านนอกและความเว้าจากเส้นรอบรูปมือ ต้องระบุค่าความยาวของนิ้วที่สั้นที่สุด เพื่อป้องกันการระบุตำแหน่งนิ้วก้อยกับขอบเว้าอื่นๆของเส้นรอบรูปมือผิด



ภาพที่ 13 ตัวอย่างภาพผลลัพธ์จากชุดข้อมูลรูปร่างมือ Bogazici University

5. สรุปผลการทดลอง

จากการนำเสนอการเปรียบเทียบเทคนิคการหาปลายนิ้วและร่องนิ้วจากเส้นรอบรูปมือนี้ได้ข้อสรุปว่าวิธีวัดรัศมีจากเส้นรอบรูปมือนี้อาจมีข้อดีในด้านความยืดหยุ่นสูงสามารถรับข้อมูลที่มีขนาดแตกต่างกันมากๆได้ ด้านวิธีการหาความโค้งจากเส้นรอบรูปมือนี้อาจมีข้อดีที่ความเร็วในการประมวลผล ส่วนวิธีหาขอบด้านนอกและความเว้าจากเส้นรอบรูปมือนี้อาจเป็นวิธีที่น่าเชื่อถือของสองวิธีข้างต้นมารวมกัน จึงเป็นอีกทางเลือกในการนำมาประยุกต์ใช้ในระบบวัดโครงสร้างมือ เพื่อการยืนยันเอกลักษณ์ด้วยเทคโนโลยีสมองกลฝังตัวในแบบระบบเปิด

เอกสารอ้างอิง

- [1] N. Duta, "A survey of biometric technology based on hand shape," *Pattern Recognition*, vol. 42, pp. 2797-2806, 2009.
- [2] A. K. Singh, A. K. Agrawal, and C. B. Pal, "Hand geometry verification system: A review," in *Ultra Modern Telecommunications & Workshops, 2009. ICUMT '09. International Conference on*, 2009, pp. 1-7.
- [3] G. Amayeh, G. Bebis, and M. Hussain, "A Comparative Study of Hand Recognition Systems," in *Emerging Techniques and Challenges for Hand-Based Biometrics (ETCHB), 2010 International Workshop on*, 2010, pp. 1-6.
- [4] E. Yörük, H. Dutağacı, and B. Sankur, "Hand biometrics," *Image and Vision Computing*, vol. 24, pp. 483-497, 2006.
- [5] J. M. Guo, Y. F. Liu, C. Mei-Hui, W. Chia-Chu, and L. Thanh-Nam, "Contact-free hand geometry identification system," in *Image Processing (ICIP), 2011 18th IEEE International Conference on*, 2011, pp. 3185-3188.
- [6] M. A. Ferrer, J. Fabregas, M. Faundez, J. B. Alonso, and C. Travieso, "Hand geometry identification system performance," in *Security Technology, 2009. 43rd Annual 2009 International Carnahan Conference on*, 2009, pp. 167-171.
- [7] M. A. Ferrer, A. Morales, C. M. Travieso, and J. B. Alonso, "Influence of the pegs number and distribution on a biometric device based on hand geometry," in *Security Technology, 2008. ICCST 2008. 42nd Annual IEEE International Carnahan Conference on*, 2008, pp. 221-225.
- [8] J. Xiaoqian, X. Wanhong, L. Sweeney, L. Yiheng, R. Gross, and D. Yurovsky, "New Directions in Contact Free Hand Recognition," in *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, 2007, pp. II - 389-II - 392.

- [9] E. Yoruk, E. Konukoglu, B. Sankur, and J. Darbon, "Shape-based hand recognition," *Image Processing, IEEE Transactions on*, vol. 15, pp. 1803-1815, 2006.
- [10] E. Yoruk, E. Konukoglu, B. Sankur, and J. Darbon, "Person authentication based on hand shape," in *Signal Processing and Communications Applications Conference*, 2004. *Proceedings of the IEEE 12th*, 2004, pp. 152-155.
- [11] G. Park and S. Kim, "Hand Biometric Recognition Based on Fused Hand Geometry and Vascular Patterns," *Sensors*, vol. 13, pp. 2895-2910, 2013.
- [12] X. Wei, X. Changsheng, and O. Sim Heng, "Peg-free Human Hand Shape Analysis and Recognition," in *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on*, 2005, pp. 77-80.
- [13] G. Boreki and A. Zimmer, "Hand geometry: a new approach for feature extraction," in *Automatic Identification Advanced Technologies, 2005. Fourth IEEE Workshop on*, 2005, pp. 149-154.
- [14] G. Tofighi, S. A. Monadjemi, and N. Ghasem-Aghaee, "Rapid hand posture recognition using Adaptive Histogram Template of Skin and hand edge contour," in *Machine Vision and Image Processing (MVIP), 2010 6th Iranian*, 2010, pp. 1-5.
- [15] S. Liu and M. Silverman, "A practical guide to biometric security technology," *IT Professional*, vol. 3, pp. 27-32, 2001.