

บทที่ 3

การหาขอบเขตภาพและการแปลงภาพ

การหาขอบภาพ (Edge Detection) เป็นเทคนิคหนึ่งในการประมวลผลภาพ (Image Processing) ที่มีจุดประสงค์เพื่อทำการหาขอบเขตในภาพ คือการทำให้ขอบของภาพนั้นมีความเด่นชัดขึ้นมา เพื่อที่จะทำการหาขอบเขตของภาพต่าง ๆ ได้ โดยขอบเขตภาพที่เกิดความเด่นชัดขึ้นมานั้นมาจากความแตกต่างความเข้มของแสงจากจุดภาพหนึ่งไปอีกจุดภาพหนึ่งที่มีความต่อเนื่องกัน โดยขอบภาพจะเด่นชัดหรือไม่ขึ้นอยู่กับความเข้มของแสงระหว่างจุดภาพ และในการหาขอบภาพที่ถูกต้องสมบูรณ์นั้นไม่ใช่เรื่องง่าย โดยเฉพาะอย่างยิ่งในการหาขอบภาพที่มีคุณภาพต่ำหรือมีความเข้มของแสงไม่สม่ำเสมอทั่วทั้งภาพหรือมีความแตกต่างระหว่างพื้นหน้ากับพื้นหลังที่มีค่าน้อย มีลายเส้นจำนวนมากที่มีความใกล้เคียงระหว่างจุดภาพที่มีความละเอียดขีดติดกัน ก็จะทำให้การหาขอบภาพให้ได้ภาพที่สมบูรณ์นั้นยากมากขึ้น

3.1 การหาขอบเขตภาพ (Edge Detection)

การหาขอบเขตภาพ คือ การตรวจสอบว่าเส้นขอบลากผ่านหรือใกล้เคียงกับจุดใด โดยวัดจากการเปลี่ยนแปลงของความเข้มในตำแหน่งที่ใกล้เคียงกับจุดดังกล่าว ซึ่งวิธีการหาขอบเขตภาพนั้นมีด้วยกันหลายวิธีสามารถแบ่งได้เป็น 2 กลุ่มหลักคือ เกรเดียนต์เมทรูด และ ลาปลาเซียนเมทรูด สำหรับงานวิจัยนี้ได้เลือกใช้เกรเดียนต์เมทรูดซึ่งจะหาขอบโดยการหาจุดต่ำสุดและจุดสูงสุดในรูปของอนุพันธ์อันดับหนึ่งของภาพซึ่งวิธีนี้สามารถที่จะหาขอบภาพได้ด้วยกันหลายวิธี เช่น โซเบล, โรเบิร์ต, พรวิท, แคนนี่ เป็นต้น ส่วนอีกวิธีหนึ่งจะเป็น ลาปลาเซียนเมทรูด จะเป็นการหาขอบเขตภาพโดยการใช้อนุพันธ์อันดับ 2 โดยที่จะใช้จุดที่มีค่า y เป็น 0 (Zerocrossing) และ Laplacian of Guassian (Log)

สำหรับการหาขอบเขตภาพโดยเลือกใช้เทคนิคโซเบลนั้น เพราะจากรูปภาพที่ได้มานั้นแต่ละภาพมีลักษณะเฉพาะรูป จึงไม่จำเป็นที่จะต้องหารายละเอียดของภาพให้ครบ แต่จุดสำคัญจะต้องสามารถหาขอบเขตภาพได้ผลลัพธ์ในระดับดี ดังนั้นจึงได้เลือกใช้เทคนิคโซเบล สำหรับการหาขอบเขตภาพในงานวิจัยนี้

การหาขอบเขตภาพโดยใช้อนุพันธ์อันดับที่หนึ่งเป็นการแปลงเกรเดียนต์ แบบไม่ต่อเนื่องบนข้อมูลภาพเชิงตัวเลขเนื่องจากการหาขอบภาพเป็นการประมวลผลแบบไม่ต่อเนื่อง ดังนั้นจึงต้องใช้อนุพันธ์ย่อยแบบไม่ต่อเนื่องตามทิศทางที่ตั้งฉากกับแกน x และแกน y กำหนดค่าได้ตามสมการ

$$\nabla_x g(x,y) = g(x,y) - g(x-1,y)$$

และ

$$\nabla_y g(x,y) = g(x,y) - g(x,y-1) \quad \text{สมการ 3.1}$$

คำนวณโดยการประมาณค่าของเกรเดียนต์ $g(x,y)$ กำหนดค่าได้จาก

$$|\nabla g(x,y)| = \sqrt{(\nabla_x g(x,y))^2 + (\nabla_y g(x,y))^2} \quad \text{สมการ 3.2}$$

เพื่อให้่ายต่อการคำนวณกำหนดให้ประมาณค่าขนาดของเกรเดียนต์ตามทิศทางที่ตั้งฉากกับแกน x และแกน y รวมกันคือ

$$|\nabla g(x,y)| = |(\nabla_x g(x,y))| + |(\nabla_y g(x,y))| \quad \text{สมการ 3.3}$$

การค้นหาลักษณะขอบภาพโดยใช้อ่อนพวนอันดับหนึ่ง เป็นวิธีการของการหาจุดต่ำสุดและจุดสูงสุดขององค์ประกอบภาพ วิธีการหาขอบภาพโดยใช้โซเบล เขียนเป็นสมการได้

$$|\nabla g(x,y)| = \left(\begin{array}{l} |g(x,y)+2g(x,y+1)+g(x,y+2)-g(x+2,y) \\ +2g(x+2,y+1)+g(x+2,y+2) \\ g(x,y)+2g(x+1,y)+g(x+2,y)-g(x+2,y) \\ +2g(x+1,y+2)+g(x+2,y+2)| \end{array} + \right) \quad \text{สมการ 3.4}$$

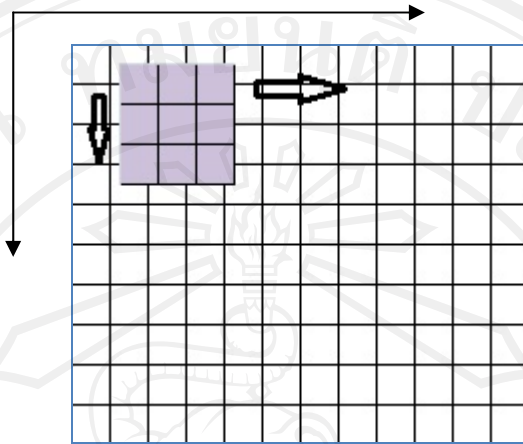
จากสมการ กำหนดขนาดของ Masks เท่ากับเมตริกซ์ขนาด 3×3 และมีค่าเท่ากับ

$$E_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad \text{และ} \quad E_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

เป็นการหาขอบภาพโดยใช้เทมเพลตขนาด 3×3 สองเทมเพลต โดยเทมเพลตแรกจะใช้หาค่าความแตกต่างในแนวนอน และเทมเพลตสองจะหาค่าความแตกต่างในแนวตั้ง

ผลที่ได้เมื่อนำค่าคงที่ดังกล่าวไปทำการปรับค่าความสว่างบนภาพ โดยการนำค่าของตัวเลขที่กำหนดไว้ในแผ่นกรองไปคูณกับค่าความสว่างของจุดภาพในแต่ละจุดภาพ โดยการคูณที่เกิดขึ้นจะถูกวนจากมุมซ้ายบนไปทางขวาจนครบแถวแรกจากนั้นก็เริ่มใหม่ในแถวที่ 2 ทาง

ด้านซ้ายเช่นเดิมดังรูป 3.1 จนครบทั้งภาพจะทำให้แต่ละจุดภาพแสดงรายละเอียดขอบของวัตถุที่ไม่เป็นเส้นตรงได้อย่างชัดเจน



รูป 3.1 แสดงลักษณะการเคลื่อนที่ของแผ่นกรองในการคำนวณ

การเลือกใช้ฟังก์ชันค้นหาขอบภาพของโปรแกรม MATLAB ที่เลือกใช้นั้น 3 วิธีซึ่งเป็นการค้นหาขอบภาพโดยการใช้อุปกรณ์อันดับหนึ่ง ได้แก่ โรเบิร์ต, พรีวิท, แคนนี่ ดังแสดงได้ในตาราง 3.1

ตาราง 3.1 แสดงฟังก์ชันการค้นหาขอบภาพของโปรแกรม Matlab

วิธีการของฟังก์ชัน	ลักษณะของคำสั่งที่ใช้
1. Robert Operator	$BW = \text{edge}(I, 'roberts', \text{thresh})$
2. Prewitt Operator	$BW = \text{edge}(I, 'prewitt', \text{thresh})$
3. Canny Operator	$BW = \text{edge}(I, 'canny', \text{thresh})$

ดังนั้น ฟังก์ชันของโปรแกรม Matlab มีวิธีการกำหนดค่าดังสมการ

$$BW = \text{Edge}(\text{Image}, 'Method', \text{Parameters})$$

กำหนดให้

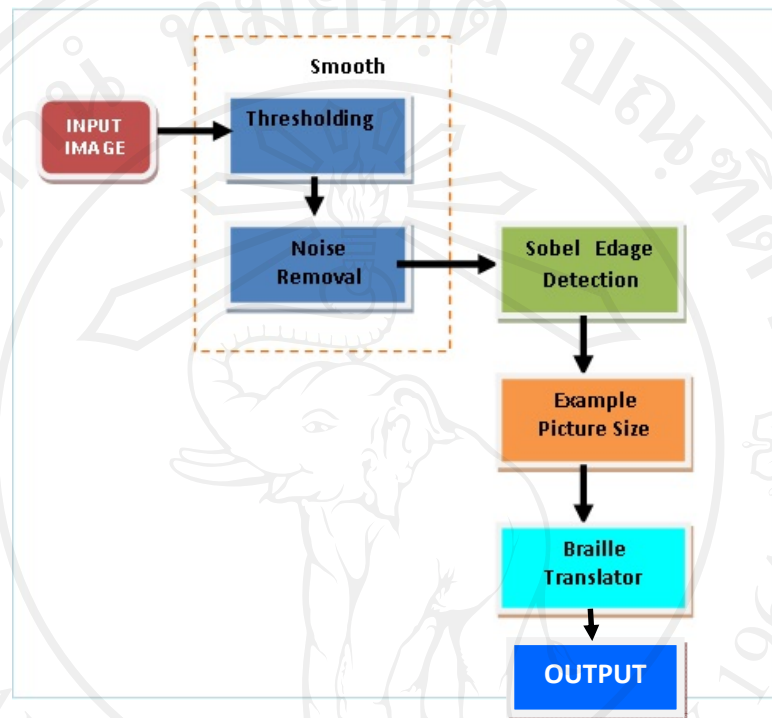
BW คือ ผลลัพธ์ภาพที่มีการค้นหาขอบภาพ (Black&White)

Method คือ Robert , Prewitt และ Canny

Parameter คือ ค่า Threshold ที่ใช้ในแต่ละวิธีการ

3.2 ขั้นตอนในการแปลงภาพกราฟิกเป็นอักษรเบรลล์

ขั้นตอนวิธีในการแปลงภาพกราฟิกเป็นอักษรเบรลล์ สามารถที่จะทำการแยกขั้นตอนต่าง ๆ อธิบายได้ดังรูป 3.2



รูป 3.2 แสดงโครงสร้างการทำงานของการทำงานของการแปลงภาพกราฟิกให้เป็นภาพอักษรเบรลล์

3.2.1 การรับภาพเข้าระบบ

สำหรับส่วนรับภาพเข้าระบบ ทำหน้าที่ในการอ่านข้อมูลหรือทำการโหลดข้อมูลรูปภาพซึ่งจะเป็นไฟล์นามสกุล JPG เป็นหลัก และระบบสามารถที่จะรองรับไฟล์รูปภาพที่มีนามสกุลเป็น BRF ได้อีกแบบ โดยภาพที่มีประสิทธิภาพในการใช้งานนั้นควรจะเป็นภาพที่มีขอบที่เด่นชัดเจน ยิ่งมีความคมชัดมากเท่าไรการหาขอบเขตภาพก็จะสามารถที่จะได้ขอบภาพออกมาได้ชัดเจนมากยิ่งขึ้น

3.2.2 การแปลงภาพสีให้เป็นภาพสีขาวดำ (Threshold)

การแปลงข้อมูลภาพที่มีความเข้มหลายระดับ ให้เป็นภาพที่มีระดับความเข้ม 2 ระดับต่อ 1 จุดภาพ เพื่อทำการหาขอบที่ให้ค่าความคมชัดและแน่นอน โดยการหาขอบจะต้องทำการแปลงภาพต้นฉบับให้มีค่าเป็นเพียง 0 หรือ 1 หรือที่เรียกว่าเป็นการทำให้ภาพมีลักษณะเป็น ขาว – ดำ ในการแปลงภาพให้เป็นภาพขาว – ดำ นั้นไม่จำเป็นที่จะต้องทำเฉพาะภาพที่เป็นภาพสีเท่านั้น

ภาพที่เป็นภาพขาว – ดำ ก็ต้องผ่านขั้นตอนนี้เช่นกัน ซึ่งในการแปลงภาพนั้นจะพบว่าในส่วนที่มีการแสดงผลของสีที่มีลักษณะเป็นสีเข้ม เมื่อทำการแปลงออกมาภาพที่มีสีเข้มก็จะแสดงผลเปลี่ยนเป็นสีดำยิ่งภาพมีความเข้มภาพ ภาพที่ได้ออกมาก็จะมีสีดำมากตามไปด้วย และในทางตรงกันข้ามภาพที่มีสีอ่อนก็จะได้ภาพออกมาเป็นสีเทา หรือออกเป็นสีขาวทั่วไป ในการกำหนดค่าการแปลงภาพสีเป็นขาว-ดำ จะทำการกำหนดขึ้นมา 2 ค่า คือ high threshold(T1) และ low threshold(T2) โดยจุดภาพที่มีค่ามากกว่า T1 จะถูกปรับเป็น 1 ซึ่งแสดงว่าเป็นจุดภาพที่มีค่าเป็นขอบหรือที่มีสีเข้ม (สีดำ) นั่นเอง แต่ถ้าน้อยกว่า T2 ก็จะถูกรับเป็น 0 โดยภาพที่ได้ผ่านกระบวนการนี้จะสามารถเห็นขอบที่ชัดเจนมากขึ้น กระบวนการทำให้ความเข้มของแม่สีในภาพมีระดับเดียวกัน คือในจุดภาพหนึ่งจะประกอบไปด้วย ค่าสี R G B จะเห็นได้ว่ามีถึง 3 ค่าใน 1 จุดภาพการทำให้แม่สี R G B ทั้ง 3 ค่ามีค่าเท่ากัน ใช้สมการดังนี้

$$R_R = \frac{(R_s + G_s + B_s)}{3} \quad R_R = ((0.299 \times R_s) + (0.587 \times G_s) + (0.114 \times B_s))$$

สมการ 3.5

$$G_R = \frac{(R_s + G_s + B_s)}{3} \quad \text{หรือ } G_R = ((0.299 \times R_s) + (0.587 \times G_s) + (0.114 \times B_s))$$

สมการ 3.6

$$B_R = \frac{(R_s + G_s + B_s)}{3} \quad B_R = ((0.299 \times R_s) + (0.587 \times G_s) + (0.114 \times B_s))$$

สมการ 3.7

เมื่อ R_R หมายถึง ค่าเอาต์พุต Pixel สีแดง

G_R หมายถึง ค่าเอาต์พุต Pixel สีเขียว

B_R หมายถึง ค่าเอาต์พุต Pixel สีน้ำเงิน

R_s หมายถึง ค่าอินพุต Pixel สีน้ำเงิน

G_s หมายถึง ค่าอินพุต Pixel สีน้ำเงิน

B_s หมายถึง ค่าอินพุต Pixel สีน้ำเงิน

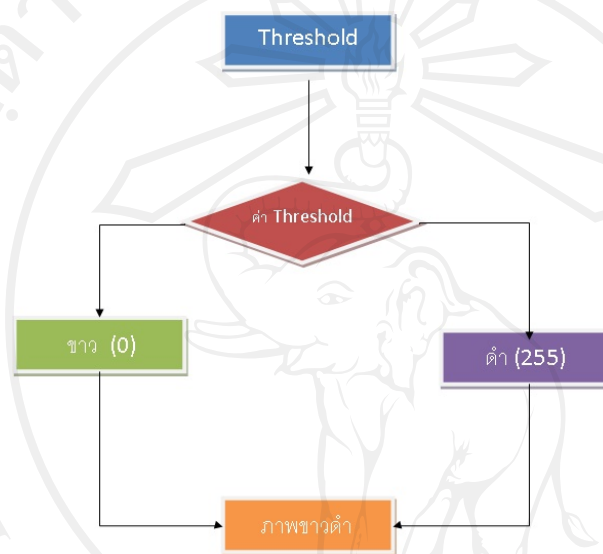
จากนั้นทำการหาค่าของการทำ Threshold เพื่อหาขอบที่มีความเด่นชัดมากขึ้น สามารถกำหนดได้ดังนี้ เมื่อผ่านทดสอบภาพที่ได้จะมีระดับของสีอยู่ 2 ค่า คือ 0 กับ 1 นั่นคือ สีขาวและดำ เรียกว่า เป็นภาพไบนารี (Binary Image) แสดงได้ดังรูป 3.3

$$g(x,y) = \begin{cases} 0, & f(x,y) < \text{Threshold} \\ 1, & f(x,y) \geq \text{Threshold} \end{cases}$$

เมื่อ $g(x,y)$ = ระดับสีเทาที่จุด (x,y) ใดๆ ของภาพ

$f(x,y)$ = เป็นค่าความสว่างของจุด (x,y)

Threshold = ระดับสีเทาที่ใช้อ้างอิงการแปลงข้อมูลซึ่งดูได้จากผลการทดลอง



รูป 3.3 แสดงการทำ Threshold

3.2.3 ขั้นตอนของการปรับภาพให้เรียบ

ขั้นตอนนี้ เป็นการแปลงภาพที่ได้จากการทำภาพขาว - ดำ เพื่อที่จะทำการกำจัดสิ่งรบกวน (Noise) ออกจากภาพ โดยการกำจัดสิ่งรบกวนนั้นจะเป็นการช่วยให้ระบบสามารถที่จะทำการจับขอบภาพได้คมชัดมากยิ่งขึ้น โดยอาศัยเทคนิคใช้ค่ามัธยฐาน (Median filtering) ซึ่งในเทคนิคนี้จะเป็นการหาสิ่งรบกวนที่เกิดขึ้นในภาพด้วยการกรอง (Filter) โดยมีรูปแบบการคำนวณโดยการนำเอาความเข้มแสงของจุดที่ตรงกันในภาพต่าง ๆ มาเรียงลำดับ (Sort) จากนั้นไปหามากจากนั้นจะเลือกค่าที่อยู่ตรงกลางไปใช้ หากจำนวนภาพทั้งหมดเป็นจำนวนคู่ จะนำค่าทั้งสองที่อยู่ตรงกลางมาหาค่าเฉลี่ย และเมื่อผ่านกระบวนการออกมาสิ่งรบกวนก็จะถูกกำจัดออกไป ซึ่งอาจจะไม่สมบูรณ์แต่ก็สามารถที่จะลดจำนวนของสิ่งรบกวนออกไปให้เหลือน้อยลงไปได้ ซึ่งวิธีนี้จะใช้เวลาในการคำนวณสูง แต่ข้อดีคือภาพที่ได้ออกมามีความคมชัด

ตัวอย่าง

ภาพที่ 1	ภาพที่ 2	ภาพที่ 3	ผลลัพธ์
1 2 1 3	2 3 4 3	3 2 1 4	2 2 1 3
4 2 2 1	5 3 4 1	2 1 4 0	4 2 4 1
0 1 1 3	3 2 4 2	1 4 2 0	1 2 2 2
2 2 1 1	1 3 1 2	2 4 0 2	2 3 1 2

3.2.4 ขั้นตอนการหาขอบเขตของภาพ

ในการหาขอบเขตของภาพที่ได้ซึ่งในขั้นตอนนี้จะเป็นการนำเทคนิคการหาขอบแบบโซเบลเข้ามาใช้งาน ดังนั้นภาพที่ผ่านกระบวนการนี้จะได้ขอบที่ชัดเจนมากยิ่งขึ้นแต่ในขณะเดียวกัน ระบบก็สามารถที่จะจับขอบของเส้นต่าง ๆ ที่ปรากฏในรูปที่เป็นส่วนที่เราไม่ได้ต้องการให้ด้วยเช่นกัน ดังที่ได้กล่าวไว้ในหัวข้อข้างต้น

3.2.5 ส่วนของการปรับขนาดภาพตัวอย่าง

ทำหน้าที่ ในการแสดงขนาดตัวอย่างรูปภาพเพื่อเป็นการตรวจเช็คก่อนที่จะมีการส่งภาพนั้นไปสู่กระบวนการสุดท้ายนั่นคือ กระบวนการการแปลงภาพเป็นอักษรเบรลล์นั่นเอง ซึ่งจุดนี้ผู้ใช้งานสามารถที่จะทำการดูภาพได้ตามขนาดที่ได้กำหนดไว้ให้ เช่น 50% ,150% หรือ 200% เพื่อทำการเปรียบเทียบกับภาพต้นฉบับถึงความถูกต้องของการหาขอบภาพที่ได้

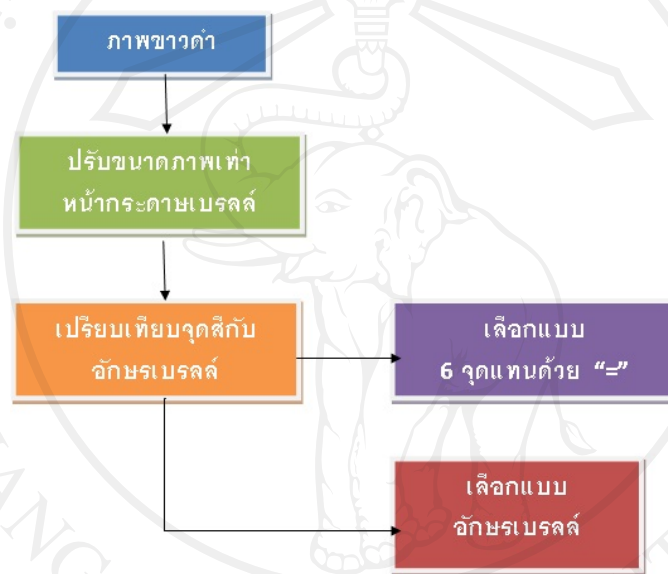
3.2.6 ส่วนของการแปลงเป็นอักษรเบรลล์

ทำหน้าที่ในการแปลงภาพที่ได้จากกระบวนการต่าง ๆ เพื่อทำการแมปขอบเขตภาพที่ได้ให้เป็นภาพกราฟิกอักษรเบรลล์ ซึ่งจะมีด้วยกัน 2 ลักษณะคือ ภาพที่แปลงได้ในลักษณะที่เป็น 6 จุด และภาพที่เป็นตามลักษณะอักษรเบรลล์ซึ่งประสิทธิภาพที่ได้ก็จะมีแตกต่างกันออกไปแล้วแต่ภาพที่นำมาทดลอง โดยภาพที่นำมาทดลองก็จะมีกรอบคำสีขาวดำอยู่ในช่วงสีที่กำหนดไว้เท่าไร โดยมากแล้วภาพขาวดำจะกำหนดระดับของสีอยู่ที่ 0 – 255 อาทิเช่น ภาพรูปหลายเหลี่ยมเมื่อทำการแปลงภาพออกมาแล้วจะได้ค่าความถี่อยู่ที่ 48 ซึ่งจะทำให้ภาพที่ได้ถึงจะมีความคมชัด แต่ในขณะเดียวกันระบบ ก็สามารถอำนวยความสะดวกผู้ใช้งานโดยสามารถที่จะทำการปรับเพิ่ม – ลด ระดับสีได้ตั้งแต่ 0 – 255 ตามความเหมาะสมที่ผู้ใช้งานต้องการแสดงได้ ดังนี้ ในการแปลงออกเป็นภาพอักษรเบรลล์นั้น ได้มีการแสดงภาพในตำแหน่งที่ภาพเบรลล์มีความใกล้เคียงกับภาพต้นฉบับ ซึ่งแต่ละภาพที่ได้ออกมาจะไม่มีค่าความเฉลี่ยที่เท่ากันแต่ขึ้นอยู่กับความละเอียด ความคมชัดของภาพต้นฉบับ การแมประหว่างจุดภาพที่หาได้กับกลุ่มอักษรเบรลล์นั้น ได้มีการสร้างกลุ่มอักษรเบรลล์ไว้รองรับจำนวน 63 ตัวซึ่งจะประกอบด้วย กลุ่มของตัวอักษรภาษาอังกฤษ a – z กลุ่มตัวเลข 0 – 9 และกลุ่มของสัญลักษณ์พิเศษ เช่น เครื่องหมาย ‘ , “ (,) เป็นต้น ซึ่งแต่ละ

ตัวอักษรที่สร้างขึ้นไว้รองรับนั้น ก็จะเป็นตัวอักษรที่เป็นมาตรฐานที่ใช้ในระบบการเรียนการสอนของนักเรียนตาบอด ซึ่งจะทำให้การสัมผัสสามารถที่จะบอกได้ว่ากลุ่มเซลล์นั้นคือตัวอักษรใดบ้างเช่นกัน อาทิเช่น ตัวอักษร $a \leq 50$ จะมีลักษณะเขียนเป็นสัญลักษณ์อักษรเบรลล์ได้เป็น



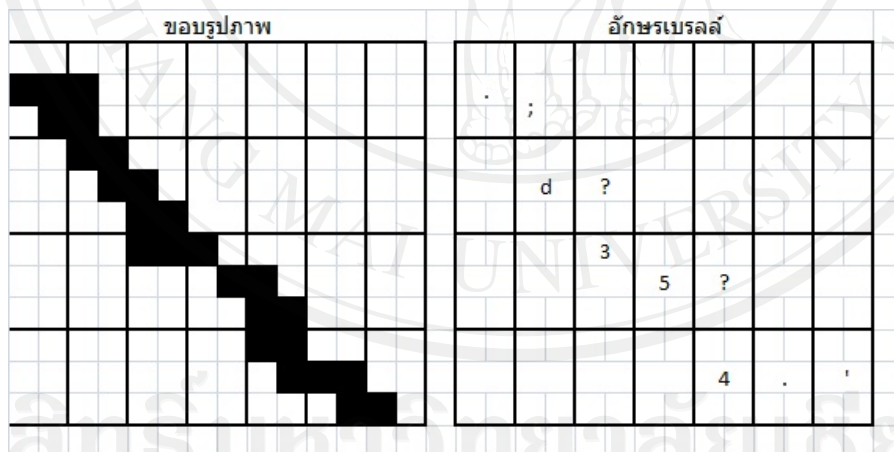
โดยขั้นตอนแสดงได้ดังรูป 3.4



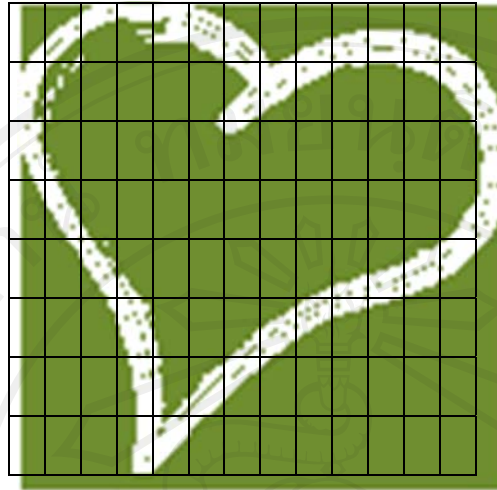
รูป 3.4 แสดงขั้นตอนการแม็บอักษรเบรลล์

โดยขั้นตอน ในการแม็บจะเริ่มจากการรับภาพที่มีการปรับภาพที่ได้เป็นสีขาวดำที่ผ่านการหาขอบเขตภาพมาแล้วนั้นทำให้ได้ความถี่ของสีที่ 0 – 255 ซึ่งเป็นค่าช่วงสีขาว ถึง ดำ และเมื่อได้ค่าความถี่สีในแต่ละจุด Pixels แล้ว ก็จะมีการนำวัดปรับขนาดให้เท่ากับจำนวนของอักษรเบรลล์ที่แสดงใน 1 หน้ากระดาษ ซึ่งอักษรเบรลล์ใน 1 หน้ากระดาษจะประกอบไปด้วยจำนวนเซลล์ทั้งหมด 40 เซลล์หรือ 40 ตัวอักษรต่อ 1 บรรทัด 1 เซลล์จะประกอบไปด้วยจุด 6 ตำแหน่ง และมีจำนวนบรรทัดทั้งหมด 25 บรรทัดต่อ 1 หน้ากระดาษ ในส่วนนี้เราสามารถที่จะทำการกำหนดขนาดของรูปได้ว่าต้องการให้ภาพที่ออกมานั้นมีขนาดเท่าใด โดยตัวระบบได้จะกำหนดไว้ที่ 40 : 25 ใน 1 หน้ากระดาษไว้ให้ ซึ่งจะได้ภาพที่มีความสมบูรณ์ แต่จะมีข้อเสียเนื่องจากวิธีนี้เป็นวิธีที่สิ้นเปลืองกระดาษ เพราะจะได้รูปเพียงรูปเดียวต่อกระดาษ 1 แผ่น

เมื่อทำการกำหนดขนาดของภาพกับกระดาษได้แล้วตัวระบบก็จะทำการเปรียบเทียบจุดภาพในภาพโดยการแบ่งให้ภาพเป็น Grid ที่มีขนาด 80 x 75 ตามขนาดของกระดาษเบรลล์ ดังนั้นจะได้ Grid ทั้งหมด 6000 แล้วทำการวิเคราะห์ต่อไปว่า โดยจะดู Grid ที่เส้นขอบลากผ่านนั้นมีลักษณะอย่างไรซึ่งจะทำการดูครั้งละ 6 ตำแหน่งตามลักษณะของอักษรเบรลล์ แล้วดูว่ามีส่วนที่คาดว่าจะเป็นตัวอักษรนั้นอยู่หรือไม่ ถ้ามีก็ให้กลุ่มของ Grid นั้นมีค่าเป็นจริง ให้เป็นสีดำ ส่วนที่ไม่ใช่ให้เป็นสีขาว เมื่อวิเคราะห์จนครบทั้ง 6000 ตำแหน่งแล้ว ก็จะได้องค์ประกอบของจุดภาพ ในภาพ เมื่อทำการเปรียบเทียบแล้วว่าตรงกับอักษรนั้น ๆ จะแสดงภาพจุดอักษรนั้นออกมาเป็นภาพเอาต์พุตต่อไปซึ่งในตัวระบบสามารถที่จะทำการเลือกได้ว่าจะให้แสดงภาพในลักษณะใด คือ ภาพที่เป็นจุดเต็ม 6 จุดที่มีค่าเท่ากับ (=) โดยการแสดงในลักษณะนี้จะทำให้ภาพที่ได้ออกมามีมิติมากขึ้นและจะทำให้การสัมผัสสามารถที่จะเข้าถึงได้ง่าย และในลักษณะของอักษรเบรลล์นั้นในการแสดงผลจะให้ความสมบูรณ์ของภาพเป็นอีกรูปแบบหนึ่งซึ่งจำเป็นต้องอาศัยความชำนาญในการสัมผัสมากพอสมควรในการบอกว่ารูปที่ได้ออกมามีลักษณะเช่นไร ซึ่งในการแสดงผลเช่นนี้จะมี ความเหมาะสมกับนักเรียนที่โตแล้วหรือผู้ที่มีความชำนาญถึงจะให้ผลการสัมผัสการรับรู้ที่ดี ลักษณะของการแม็บบขอบเขตภาพที่ได้กับอักษรเบรลล์ ดังรูป 3.5



รูป 3.5 แสดงการแม็บบขอบเขตภาพกับอักษรเบรลล์



```

%qc      _ta&z=a_      A-----'
%&ta'    ac,y,taa A(aqay=xy=====.'
=b       a,y=a===qfaaa ccd=y=a
la       %y=,tf      daaa
l        A,t&ja      a==v
q        aca        a(=
_        a,t
?a/;     ==
,ya      a,y,t
,ya      %&,yf
a,ya"    A%=af|
,yr'     A%!&,tqa
gya      _((,tq=,tgc
? =a'    _,(t?==ghfa
,y=v     A%tvqfc
a=y'     Aaa,tga
,yar     %,y,tia
a=a      ay=ia
== aa,tqa
=q%aqc
A=a=f









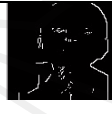











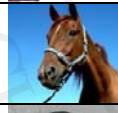















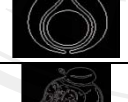


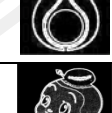
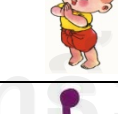














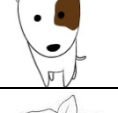



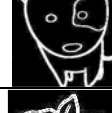
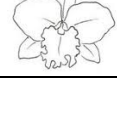




```

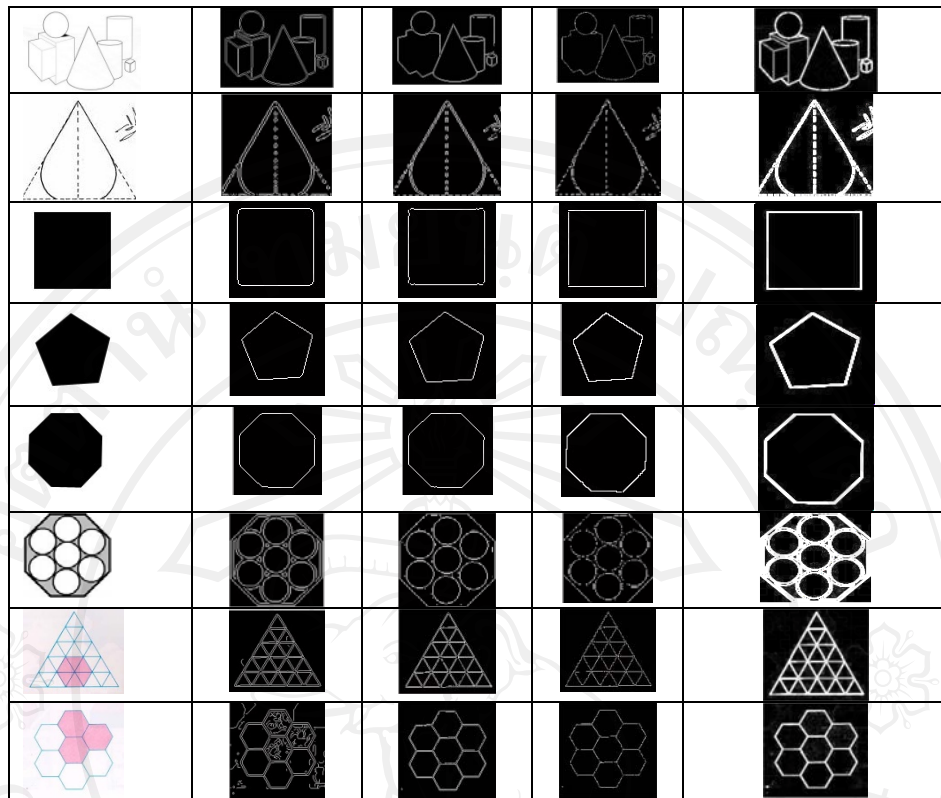
รูป 3.6 แสดงตัวอย่างการแม็บบอกภาพกับอักษรเบรลล์

ตารางแสดงการเปรียบเทียบการแปลงภาพ ด้วยวิธีการใช้โปรแกรม MatLab โดยการใช้เทคนิคโรเบิร์ต, พรีวิท และ แคนนี่ และการใช้โปรแกรมการแปลงภาพอักษรเบรลล์ ที่ได้จัดทำขึ้น โดยใช้เทคนิคโซเบลโดยผลของการทดลองออกมาได้ดังตาราง 3.2

Copyright © by Chiang Mai University
All rights reserved

ตาราง 3.2 แสดงตารางเปรียบเทียบการแปลงภาพที่ได้จากเทคนิคทาง MatLab และ โปรแกรมการแปลงภาพอักษรเบรลล์

ภาพต้นฉบับ	โปรแกรม MatLab			โปรแกรมแปลงภาพด้วยวิธี Sobel
	Canny	Prewitt	Roberts	
				
				
				
				
				
				
				
				
				
				
				
				
				



ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
 Copyright© by Chiang Mai University
 All rights reserved