

# เทคนิคการเพิ่มประสิทธิภาพของโมเดลรู้จำภาพลายมือชื่อ

นางสาวอนุสรฯ หิรัญวนากุล



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต  
สาขาวิชาวิศวกรรมคอมพิวเตอร์  
มหาวิทยาลัยเทคโนโลยีสุรนารี  
ปีการศึกษา 2557

**A TECHNIQUE TO IMPROVE IMAGE RECOGNITION  
PERFORMANCE OF HANDWRITTEN SIGNATURE**

**Anusara Hirunyanakul**



**A Thesis Submitted in Partial Fulfillment of the Requirements for the  
Degree of Master of Engineering in Computer Engineering**

**Suranaree University of Technology**

**Academic Year 2014**

## เทคนิคการเพิ่มประสิทธิภาพของโมเดลรู้จำภาพลายมือชื่อ

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้บัณฑิตวิทยาลัยรับนี้เป็นส่วนหนึ่งของการศึกษา  
ตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

คณะกรรมการสอบวิทยานิพนธ์

\_\_\_\_\_

(รศ. ดร.กิตติศักดิ์ เกิดประสพ)

ประธานกรรมการ

\_\_\_\_\_

(รศ. ดร.นิตยา เกิดประสพ)

กรรมการ (อาจารย์ที่ปรึกษาวิทยานิพนธ์)

\_\_\_\_\_

(ผศ. ดร.ปรเมศวร์ ห่อแก้ว)

กรรมการ

\_\_\_\_\_

(ศ. ดร.ชูกิจ ทิมปีจันงค์)

รองอธิการบดีฝ่ายวิชาการและนวัตกรรม

\_\_\_\_\_

(รศ. ร.อ. ดร.กนต์ธร ชำนิประศาสน์)

คณบดีสำนักวิชาวิศวกรรมศาสตร์

อนุสรณ์ หิรัญวานากุล : เทคนิคการเพิ่มประสิทธิภาพของโมเดลรู้จำภาพลายมือชื่อ  
(A TECHNIQUE TO IMPROVE IMAGE RECOGNITION PERFORMANCE OF  
HANDWRITTEN SIGNATURE) อาจารย์ที่ปรึกษา : รองศาสตราจารย์ ดร.นิตยา  
เกิดประสพ, 117 หน้า.

งานวิจัยนี้ นำเสนอเทคนิคใหม่เพื่อการเพิ่มประสิทธิภาพในการรู้จำภาพลายมือชื่อ โดย  
แหล่งข้อมูลลายมือชื่อของงานวิจัยชิ้นนี้มาจากการเก็บข้อมูลจริง จำนวนทั้งสิ้น 600 ภาพ จากบุคคล  
ทั้งหมด 30 คน โดยแต่ละลายมือชื่อจะถูกแปลงเป็นไฟล์ภาพและทำให้มีขนาด 38x144 พิกเซล  
เท่ากันทั้งหมด จากนั้นทุก ๆ พิกเซลในแต่ละภาพจะถูกแปลงค่าเป็นค่าความเข้ม จาก 0 ถึง 255 โดย  
0 คือค่าความเข้มที่มากที่สุด หรือสีดำ ส่วนค่า 255 หมายถึงสีขาว ภาพลายมือชื่อเหล่านี้จะถูกนำไป  
ฝึกการรู้จำ และทดสอบความแม่นยำโดยอัลกอริทึม ทั้ง 4 แบบ ได้แก่ ซัพพอร์ตเวกเตอร์แมชชีน  
(ด้วยเคอร์เนลฟังก์ชันเชิงเส้น พหุนาม เรเดียลเบสิส และ ซิกมอยด์) เคเนียร์เสนเบอร์  
เพอร์เซปตรอน และนาอึฟเบย์ (ด้วยฟังก์ชันการกระจายข้อมูลแบบ เกาเซียน มัลติโนเมียล และ  
เบอร์นูลลี) จากผลการทดลองแรกพบว่า ซัพพอร์ตเวกเตอร์แมชชีน ที่เคอร์เนลฟังก์ชันพหุนาม ได้ให้  
ค่าความแม่นยำที่สูงที่สุด (โดยมีค่าความแม่นยำสูงถึง 95.33%) จากนั้นผู้วิจัยได้ทดลองใช้เทคนิค  
การปรับปรุงรูปภาพเข้ามาช่วย ซึ่งได้แก่ วิธีการตรวจหาขอบภาพ ทั้ง 4 แบบ ได้แก่ โซเบล ฟรีวิต  
โรเบิร์ต และแคนนี่ รวมถึงการทำขอบภาพให้บาง เราพบว่าด้วยเทคนิคการหาขอบภาพแบบโซเบล  
ส่งผลให้อัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีน เคอร์เนลฟังก์ชันเชิงเส้น มีความแม่นยำสูงขึ้นเป็น  
96.00% (ซึ่งสูงกว่าค่าที่มากที่สุดในชุดข้อมูลดั้งเดิม) และยังทำให้อัลกอริทึมเคเนียร์เสนเบอร์ มีค่า  
ความแม่นยำสูงขึ้นอย่างมีนัยสำคัญ (จาก 84.67% เป็น 95.33% เมื่อค่า  $k=1$ ) จากนั้นผู้วิจัยได้นำเสนอ  
เทคนิคใหม่ ด้วยการเพิ่มข้อมูลที่มีค่าความเข้มสีระดับสูงเข้าไปต่อท้ายชุดข้อมูลเดิม เป็นขั้นตอนต่อ  
จากการทำขอบแบบโซเบล และด้วยวิธีการเพิ่มข้อมูลนี้เองทำให้ค่าความแม่นยำของอัลกอริทึมส่วน  
ใหญ่สูงขึ้น โดยเฉพาะอย่างยิ่งอัลกอริทึมเคเนียร์เสนเบอร์ เมื่อ  $k=1$  และ อัลกอริทึมซัพพอร์ต  
เวกเตอร์แมชชีน เคอร์เนลฟังก์ชันเชิงเส้นให้ค่าสูงถึง 98% ซึ่งเป็นค่าที่สูงที่สุดในงานวิจัยชิ้นนี้

สาขาวิชา วิศวกรรมคอมพิวเตอร์  
ปีการศึกษา 2557

ลายมือชื่อนักศึกษา \_\_\_\_\_  
ลายมือชื่ออาจารย์ที่ปรึกษา \_\_\_\_\_

ANUSARA HIRUNYAWANAKUL : A TECHNIQUE TO IMPROVE  
IMAGE RECOGNITION PERFORMANCE OF HANDWRITTEN  
SIGNATURE. THESIS ADVISOR : ASSOC. PROF. NITTAYA  
KERDPRASOP, Ph.D., 117 PP.

IMAGE RECOGNITION/HANDWRITTEN SIGNATURE/SUPPORT VECTOR  
MACHINE/K-NEAREST NEIGHBOR/NAIVE BAYES/EDGE DETECTION

In this research we propose a novel technique for the improvement of handwritten recognition. We collect totally 600 signatures from 30 people for this research. Then we transform the hand written signature images to be image file and resize each image to 144 x 38 pixels along the width and the height, respectively. Every pixel is encoded its intensity value from 0 to 255. The 0 is the highest intensity (black color) and 255 is white color. Next, we use 4 different algorithms: Support Vector Machine (with linear, polynomial, radial basis, and sigmoid kernel functions), k-Nearest Neighbor, Perceptron, and Naïve Bayes (using gaussian, multinomial, and Bernoulli distribution functions). From the experiment results, we found that SVM with polynomial kernel function showed the highest accuracy (95.33%). Then we use 4 techniques of edge detection: Sobel, Prewitt, Robert, Canny and Thinning technique. By applying Sobel edge detection technique, we found that the accuracy is improved up to 96% (higher than the highest value of original data). We also observed that the Sobel technique can improve the accuracy in k-NN with a significant level (from 84.67% to 95.33%). Moreover, we propose a novel technique to improve accuracy by appending the high intensity color data after the step of using Sobel edge detection.

And by this technique, we see accuracy of almost algorithms are improved. Especially, the accuracy of k-NN (k=1) and SVM with linear function, is 98.00%, which is the highest accuracy of this research.



School of Computer Engineering

Academic Year 2014

Student's Signature \_\_\_\_\_

Advisor's Signature \_\_\_\_\_

## กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงด้วยดี ผู้วิจัยขอกราบขอบพระคุณ บุคคล และกลุ่มบุคคลต่าง ๆ ที่ได้กรุณาให้คำปรึกษา แนะนำ ช่วยเหลืออย่างดียิ่ง ทั้งในด้านวิชาการ และด้านการดำเนินงานวิจัย ดังต่อไปนี้

รองศาสตราจารย์ ดร.นิตยา เกิดประสพ อาจารย์ที่ปรึกษาวิทยานิพนธ์ และรองศาสตราจารย์ ดร.กิตติศักดิ์ เกิดประสพ ที่ให้คำปรึกษาในการทำงานวิจัย การจัดรูปแบบ และช่วยตรวจทานความถูกต้องของวิทยานิพนธ์

คุณกัลญา พับโพธิ์ เลขานุการสาขาวิชาวิศวกรรมคอมพิวเตอร์ ที่ให้ความช่วยเหลือในการประสานงานด้านเอกสารระหว่างศึกษา

คุณวุฒิกิร เตชะเวชเจริญ คุณนันทวุฒิ กะอังกู คุณรติพร จันทร์กลั่น คุณเกตุกาญจน์ ไชยจันทร์ และนักศึกษาบัณฑิตสาขาวิชาวิศวกรรมคอมพิวเตอร์ ทุกคนที่ให้คำปรึกษา ช่วยตรวจทานความถูกต้องและช่วยเหลือด้วยดีมาโดยตลอด

ขอขอบคุณนักศึกษาชั้นปีที่ 4 คณะมนุษยศาสตร์และสังคมศาสตร์ สาขาสารสนเทศศึกษามหาวิทยาลัยบูรพา สำหรับตัวอย่างลายมือชื่อ ซึ่งใช้ในงานวิจัยนี้

นอกจากนี้ขอขอบคุณครู อาจารย์ทั้งในอดีตและปัจจุบันที่ให้ความรู้แก่ผู้วิจัยจนประสบความสำเร็จในชีวิต

ท้ายที่สุดที่จะลืมไม่ได้ ขอกราบขอบพระคุณ บิดา มารดา ที่ให้กำเนิด อบรม เลี้ยงดูด้วยความรัก และส่งเสริมการศึกษาเป็นอย่างดีโดยตลอด ทำให้ผู้วิจัยมีความรู้ ความสามารถ มีจิตใจที่เข้มแข็ง รวมทั้งเป็นกำลังใจที่ยิ่งใหญ่แก่ผู้วิจัย จนทำให้ผู้วิจัยประสบความสำเร็จในชีวิตเรื่อยมา

อนุสรุ หิรัญวนากุล

# สารบัญ

หน้า

บทคัดย่อ (ภาษาไทย).....	ก
บทคัดย่อ (ภาษาอังกฤษ).....	ข
กิตติกรรมประกาศ.....	ง
สารบัญ.....	จ
สารบัญตาราง.....	ช
สารบัญรูป.....	ฉ
<b>บทที่</b>	
<b>1 บทนำ.....</b>	<b>1</b>
1.1 ความสำคัญและที่มาของปัญหาการวิจัย.....	1
1.2 วัตถุประสงค์ของการวิจัย.....	2
1.3 ข้อยกเว้นเบื้องต้น.....	2
1.4 ขอบเขตของการวิจัย.....	3
1.5 ประโยชน์ที่ได้รับ.....	3
<b>2 ปรัชญาบรรณกรรมและงานวิจัยที่เกี่ยวข้อง.....</b>	<b>4</b>
2.1 การเรียนรู้ของเครื่อง (Machine Learning).....	4
2.2 ไบโอมेटริกซ์ (Biometrics).....	5
2.3 การรู้จำแบบ (Pattern Recognition).....	8
2.4 การทำเหมืองข้อมูลสำหรับการจำแนกประเภท (Data Mining for Classification).....	9
2.5 อัลกอริทึมเพอร์เซปตรอน (Perceptron).....	11
2.6 อัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine).....	14
2.7 อัลกอริทึมนาอิว์เบย์ (Naïve Bayes).....	17
2.8 อัลกอริทึมเคเนียร์เนสเนเบอร์ (k-Nearest Neighbors).....	20
2.9 การตรวจหาขอบภาพ และการทำภาพให้บางลง (Edge Detection and Thining).....	21
2.9.1 การตรวจหาขอบภาพ.....	21



## สารบัญ (ต่อ)

	หน้า
2.9.1.1 การตรวจหาขอบภาพแบบโซเบล.....	21
2.9.1.2 การตรวจหาขอบภาพแบบพรีวิด.....	22
2.9.1.3 การตรวจหาขอบภาพแบบโรเบิร์ต.....	22
2.9.1.4 การตรวจหาขอบภาพแบบแคนนี่.....	23
2.9.2 การทำภาพให้บางลง.....	24
2.10 โปรแกรม MATLAB.....	26
2.11 ภาษาไพธอน และเครื่องมือในการพัฒนาโปรแกรมสไปเดอร์ (Python Language and Spyder IDE).....	27
2.12 งานวิจัยที่เกี่ยวข้อง (Literature Review).....	29
<b>3 วิธีดำเนินการวิจัย.....</b>	<b>33</b>
3.1 กรอบแนวคิดของการวิจัย.....	34
3.1.1 กรอบแนวคิดขั้นตอนที่ 1: วิธีเก็บข้อมูลตัวอย่างลายมือชื่อ และการแปลงภาพลายมือชื่อให้เป็นไฟล์ภาพ.....	36
3.1.2 กรอบแนวคิดขั้นตอนที่ 2: การแปลงข้อมูลภาพลายมือชื่อ ให้เป็นข้อมูลทางตัวเลข.....	37
3.1.3 กรอบแนวคิดขั้นตอนที่ 3: การทดสอบเปรียบเทียบประสิทธิภาพ ของแต่ละอัลกอริทึม.....	39
3.1.4 กรอบแนวคิดของขั้นตอนที่ 4: การเพิ่มประสิทธิภาพการรู้จำด้วย เทคนิคการปรับปรุงรูปภาพ.....	41
3.1.5 กรอบแนวคิดของขั้นตอนที่ 5: การเพิ่มประสิทธิภาพการรู้จำด้วย เทคนิคการเพิ่มชุดข้อมูลที่มีความเข้มสูง.....	43
3.2 เครื่องมือที่ใช้ในงานวิจัย.....	46
<b>4 การทดสอบและอภิปรายผล.....</b>	<b>47</b>
4.1 ข้อมูลที่ใช้ในการทดสอบ.....	47
4.2 การทดสอบประสิทธิภาพระหว่างโมเดลการรู้จำที่สร้างขึ้น จากอัลกอริทึมทั้ง 4 แบบ.....	48

## สารบัญ (ต่อ)

หน้า

4.2.1 ผลการทดลองการเปรียบเทียบประสิทธิภาพ ระหว่างอัลกอริทึมทั้ง 4 แบบ.....	48
4.3 การทดสอบประสิทธิภาพโมเดลการเรียนรู้จำ โดยใช้ภาพถ่ายมือชื่อที่ผ่านเทคนิค การหาขอบภาพ และการทำขอบภาพให้บางลง.....	49
4.4 การทดสอบประสิทธิภาพโมเดลการเรียนรู้จำ โดยใช้ภาพถ่ายมือชื่อที่ผ่านเทคนิคการ ตรวจหาขอบภาพแบบโซเบล และใช้เทคนิคการเพิ่มชุดข้อมูลความเข้มสูง.....	50
4.4.1 การทดสอบหาความสัมพันธ์ระหว่างความแม่นยำ และเวลาที่ใช้ กับจำนวนเปอร์เซ็นต์ของชุดข้อมูลที่เพิ่มเข้าไป.....	51
4.4.2 การทดสอบประสิทธิภาพด้านความแม่นยำของการใช้เทคนิคการเพิ่มชุด ข้อมูลความเข้มสูง.....	52
4.5 อภิปรายผล.....	54
<b>5 การทดสอบและอภิปรายผล.....</b>	<b>56</b>
5.1 ขั้นตอนการดำเนินงานวิจัย.....	56
5.2 สรุปผลการวิจัย.....	57
5.3 ปัญหาและข้อเสนอแนะ.....	59
รายการอ้างอิง.....	60
ภาคผนวก	
ภาคผนวก ก. การใช้งานโปรแกรม.....	62
ภาคผนวก ข. บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างการศึกษา.....	67
ประวัติผู้เขียน.....	117

## สารบัญตาราง

ตารางที่	หน้า
2.1 ตัวอย่างข้อมูลสภาพอากาศที่มีผลต่อการออกไปเล่นกอล์ฟ.....	18
2.2 ตารางแสดงค่าความน่าจะเป็นของแต่ละแอดทริบิวต์.....	19
2.3 สรุปรูปเปรียบเทียบงานวิจัยที่เกี่ยวข้อง.....	32
4.1 แสดงค่าความแม่นยำของแต่ละอัลกอริทึม โดยใช้ภาพดั้งเดิม.....	48
4.2 แสดงค่าความแม่นยำของแต่ละอัลกอริทึม เปรียบเทียบระหว่าง 6 วิธีการ.....	50
4.3 แสดงค่าความแม่นยำของแต่ละอัลกอริทึม เมื่อเพิ่มชุดข้อมูลความเข้มสูง.....	52
4.4 แสดงค่าความแม่นยำของภาพลายมือชื่อที่ผ่านกระบวนการตรวจหาขอบแบบโซเบล และ เพิ่มชุดข้อมูลความเข้มสูง.....	53

## สารบัญรูป

รูปที่	หน้า
2.1	แผนภาพแสดงองค์ประกอบของระบบไบโอเมตริกซ์..... 6
2.2	แสดงการจัดหมวดหมู่ของไบโอเมตริกซ์..... 7
2.3	แสดงการจัดกลุ่มที่จำแนก “A” และ “B” ออกจากกัน..... 9
2.4	การรู้จำ และองค์ความรู้ที่เกี่ยวข้อง..... 9
2.5	ขั้นตอนการทำงานโดยทั่วไปของกระบวนการทำเหมืองข้อมูล..... 10
2.6	ลักษณะตัวอย่างการจำแนกประเภทที่ดี และไม่ดี..... 10
2.7	ภาพภาพใบหน้าของบุคคลเดียวกันในอิริยาบถ อารมณ์ ความสว่าง และองค์ประกอบที่ต่างกัน..... 11
2.8	ตัวอย่างการจำแนกด้วยแบบจำลองเชิงเส้น และสมการ..... 12
2.9	แสดงการเปลี่ยนแปลงของเส้นแบ่ง ในแต่ละค่าสัมประสิทธิ์การเรียนรู้..... 13
2.10	การจำแนกข้อมูลเชิงเส้น และเพอร์เซปตรอน..... 14
2.11	รูปภาพแสดงการลากเส้นเชื่อมจุดขอบของแต่ละกลุ่มตัวอย่าง..... 15
2.12	การพยายามสร้างเส้นแบ่งตรงกึ่งกลางระหว่างกลุ่มทั้งสอง โดยให้มีระยะขอบที่มากที่สุด..... 16
2.13	แสดงการใช้ตัวแปรอนุโลม ที่ยอมมองข้ามบางจุดขอบเพื่อให้ได้ Margin ที่มากที่สุด..... 16
2.14	แสดงการแบ่งข้อมูลด้วยเส้นตรงที่ต่างกัน..... 17
2.15	แสดงการจำแนกประเภทด้วยค่า $k$ ที่ต่างกัน..... 20
2.16	แสดงโซเบลฟิลเตอร์..... 22
2.17	แสดงพรีวิตฟิลเตอร์..... 22
2.18	แสดงโรเบิร์ตฟิลเตอร์..... 22
2.19	แสดงทินนิงฟิลเตอร์..... 24
2.20	เงื่อนไขของฟิลเตอร์ P1 เมื่อตำแหน่ง (ก) $p_3=0$ (ข) $p_5=0$ และ (ค) $p_1=0$ และ $p_7=0$ ..... 25
2.21	เงื่อนไขของฟิลเตอร์ P2 เมื่อตำแหน่ง (ก) $p_1=0$ (ข) $p_7=0$ และ (ค) $p_3=0$ และ $p_5=0$ ..... 26
3.1	แผนภาพแสดงขั้นตอนการดำเนินงานวิจัยทั้ง 5 ขั้นตอน..... 35
3.2	วิธีการเก็บข้อมูลลายมือชื่อ และ การแปลงข้อมูลให้อยู่ในรูปแบบไฟล์ภาพ..... 36

## สารบัญญรูป (ต่อ)

รูปที่	หน้า
3.3 ตัวอย่างภาพลายมือชื่อที่ปรับให้เป็นสีขาว ดำ และทำให้มีขนาดเท่ากัน.....	37
3.4 ตัวอย่างภาพลายมือชื่อที่อยู่ในรูปของตาราง และค่าความเข้มถูกแทนที่ด้วยตัวเลข 0 – 255.....	38
3.5 ภาพตัวอย่างแสดงการแปลงไฟล์ภาพให้เป็นข้อมูลตัวเลขตามความเข้มสีด้วยไพธอน.....	38
3.6 ขั้นตอนการแปลงไฟล์รูปภาพให้เป็นข้อมูลตัวเลข.....	39
3.7 การทดสอบเปรียบเทียบประสิทธิภาพของแต่ละอัลกอริทึม.....	40
3.8 ขั้นตอนการเพิ่มประสิทธิภาพการรู้จำด้วยเทคนิคการปรับปรุงรูปภาพ.....	42
3.9 ภาพตัวอย่างแสดงการเพิ่มข้อมูลความเข้มสูงในแต่ละลายมือชื่อ.....	43
3.10 ภาพตัวอย่างแสดงการเติมข้อมูลที่มีความเข้มสูงลงไปในแต่ละลายมือชื่อ ตามเปอร์เซ็นต์ของจำนวนข้อมูลที่มีความเข้มสูงที่สุด.....	44
3.11 ภาพแผนผังแสดงการทำงานของกรอบแนวคิดขั้นตอนที่ 5.....	45
4.1 ตัวอย่างข้อมูลภาพลายมือชื่อ.....	47
4.2 กราฟแสดงความสัมพันธ์ของจำนวน $k$ ในอัลกอริทึมเคเนียร์สเนบอร์ และความแม่นยำ... ..	49
4.3 กราฟแสดงความสัมพันธ์ระหว่างความแม่นยำ เวลาที่ใช้ในการสร้างโมเดล และเปอร์เซ็นต์ของข้อมูลความเข้มสูงที่เพิ่มเข้าไป.....	51
4.4 กราฟแสดงการเปรียบเทียบความแม่นยำระหว่างเทคนิคต่าง ๆ กับข้อมูลดั้งเดิม.....	54
5.1 แผนภาพแสดงผลการทดลองที่ดีที่สุด ในแต่ละขั้นตอนของการปรับปรุงประสิทธิภาพ.....	58

# บทที่ 1

## บทนำ

### 1.1 ความสำคัญและที่มาของปัญหาการวิจัย

เทคโนโลยีในปัจจุบันนี้ ให้ความสนใจเกี่ยวกับการตรวจสอบความถูกต้องของตัวบุคคล โดยใช้ข้อมูลทางชีวภาพ ทั้งลักษณะเฉพาะทางกายภาพ รวมไปถึงลักษณะทางพฤติกรรม มาใช้ในการตรวจสอบหรือแสดงตน เรียกว่า ไบโอมेटริกซ์ (Biometrics) ตัวอย่างของไบโอมेटริกซ์ ได้แก่ ลายนิ้วมือ ใบหน้า ม่านตา ลายมือชื่อ เป็นต้น ไบโอมेटริกซ์เหล่านี้จะถูกเก็บรวบรวมเป็นข้อมูล เพื่อนำมาเปรียบเทียบจำแนกความแตกต่างของแต่ละบุคคลนั้นออกจากกัน หรือเพื่อจดจำ รวมไปถึงการรู้จำลักษณะเฉพาะของแต่ละบุคคล

การใช้ไบโอมेटริกซ์ ในการตรวจสอบตัวบุคคล หรือระบุตัวบุคคลนั้น ทำให้เกิดความสะดวกในการที่ไม่ต้องพกเอกสารเพื่อระบุตัวบุคคล และยังสามารถลดปัญหาเรื่องการปลอมแปลงเอกสารได้อีกด้วย การรู้จำ คืองานวิจัยลักษณะหนึ่งที่มีความสำคัญ ซึ่งเป็นแขนงหนึ่งของปัญญาประดิษฐ์ (Artificial Intelligence) การรู้จำมีหลายชนิด และการรู้จำลายมือชื่อนั้นก็เป็นหนึ่งในสิ่งที่เป็นที่นิยมอย่างมาก เนื่องจากลายมือชื่อนั้นคือ อัตลักษณ์ภายนอกอย่างหนึ่ง ที่ใช้ในการระบุตัวตนของบุคคล ซึ่งนิยมใช้กันทั่วไป เนื่องจากไม่ต้องมีอุปกรณ์เฉพาะทางเพื่อมาใช้ร่วม อย่างเช่น เครื่องสแกนลายมือ หรือเครื่องสแกนม่านตา

ลายมือชื่อของแต่ละบุคคลนั้น มีลักษณะที่แตกต่างกันออกไป การปลอมแปลงหรือลอกเลียนแบบนั้น จึงเป็นไปได้ยาก งานวิจัยด้านการวิเคราะห์และรู้จำลายมือชื่อจึงได้ถูกพัฒนาขึ้น เพื่อให้สามารถวิเคราะห์ความเฉพาะตัวของลายมือชื่อของมนุษย์แต่ละคน (Human Identification) แต่ปัญหาของลายมือชื่อ ที่มีความคล้ายคลึงกันของแต่ละบุคคล หรือ ลายมือชื่อของบุคคลเดียวกัน ก็ยังคงมีความแตกต่างกัน อันเนื่องมาจากสภาวะแวดล้อมต่าง ๆ ที่ส่งผลให้การลงลายมือชื่อของบุคคลในแต่ละครั้งผิดเพี้ยนไปจากเดิมบ้าง จึงส่งผลทำให้ความถูกต้องของการระบุตัวบุคคลนั้นต่ำลงไปอีก

งานวิจัยที่เกี่ยวข้องกับการตรวจสอบลายมือชื่อ และการรู้จำลายมือชื่อนั้น ส่วนใหญ่จะมุ่งเน้นไปที่การศึกษาเปรียบเทียบ และการปรับปรุงเพื่อเพิ่มประสิทธิภาพของโมเดล ดังนั้นทางผู้วิจัยจึงได้เล็งเห็นถึงความสำคัญ ของการเพิ่มประสิทธิภาพความแม่นยำ ในการรู้จำภาพลายมือชื่อ โดยการนำเสนอเทคนิคใหม่เพิ่มประสิทธิภาพ ด้วยการทำภาพลายมือชื่อให้มีความคมชัดยิ่งขึ้น

โดยใช้เทคนิคการทำขอบภาพ ร่วมกับการเพิ่มชุดข้อมูลที่มีความเข้มสูง และเป็นข้อมูลที่มีความน่าสนใจเข้าไปต่อท้ายชุดข้อมูลเดิม

ในส่วนของฐานข้อมูลที่นำมาใช้ในงานวิจัยนั้น ทางผู้วิจัยเลือกที่จะเก็บรวบรวมตัวอย่างลายมือชื่อด้วยตนเอง จากผู้เข้าร่วมทำการวิจัย ที่เป็นนักศึกษาซึ่งมีการใช้ลายมือชื่อจริงในชีวิตประจำวัน นับได้ว่าเป็นแหล่งข้อมูลที่แสดงให้เห็นถึงการนำไปใช้งานได้จริง

## 1.2 วัตถุประสงค์ของการวิจัย

จากแนวคิดในการทำงานวิจัย ผู้วิจัยได้ตั้งวัตถุประสงค์ในการวิจัยไว้ดังนี้

- 1) เพื่อคัดเลือกอัลกอริทึมพื้นฐานที่จะใช้ในการรู้จำ ด้วยการเปรียบเทียบความแม่นยำของโมเดลการจำรู้จำลายมือชื่อที่ได้จากอัลกอริทึมทั้ง 4 แบบ ได้แก่ ซัพพอร์ตเวกเตอร์แมชชีน (ใช้เคอร์เนลฟังก์ชันสี่ชนิด คือ ฟังก์ชันเชิงเส้น ฟังก์ชันพหุนาม ฟังก์ชันเรเดียลเบสิส และฟังก์ชันซิกมอยด์) เคเนียร์เสนเบอร์ เพอร์เซปตรอน นาอ์ฟเบย์ (ใช้ฟังก์ชันการกระจายข้อมูลสามชนิด คือ เกาเซียน มัลติโนเมียล และเบอร์นูลลี)
- 2) เพื่อพัฒนาอัลกอริทึมใหม่ที่จะช่วยเพิ่มประสิทธิภาพของโมเดลการรู้จำภาพลายมือชื่อ

## 1.3 ข้อตกลงเบื้องต้น

- 1) ข้อมูลที่นำมาใช้ในงานวิจัยนี้ เกิดจากการเก็บรวบรวมข้อมูลจริง ซึ่งเก็บไว้ใน <https://sites.google.com/site/nhinganusarapesut/signature/datasets>
- 2) การทดสอบประสิทธิภาพของการจำแนกข้อมูล จะใช้วิธีการเปรียบเทียบความแม่นยำในการจำแนก และเวลาที่ใช้ในการประมวลผลข้อมูล
- 3) ขั้นตอนการแปลงข้อมูลรูปภาพให้เป็นระดับความเข้มสี ผู้วิจัยได้แบ่งระดับความเข้มสีไว้ที่ 256 ระดับ โดย 0 หมายถึงมีความเข้มสีมากที่สุด และ ไล่ลงไปเรื่อย ๆ จนถึง 255 ซึ่งหมายถึงความเข้มสีน้อยที่สุด
- 4) งานวิจัยชิ้นนี้เลือกใช้ภาษาและเครื่องมือในงานวิจัย ได้แก่ MATLAB version 7.14, Python2.7, Spyder และ Scikit Learn

#### 1.4 ขอบเขตของการวิจัย

จากการศึกษาค้นคว้าข้อมูล ผู้วิจัยได้กำหนดขอบเขตของการวิจัยไว้ดังนี้

- 1) เก็บรวบรวมข้อมูลตัวอย่างลายมือชื่อ จากผู้ใช้ลายมือชื่อในชีวิตประจำวัน จำนวน 30 คน
- 2) ปรับขนาดภาพของแต่ละลายมือชื่อให้มีขนาดเท่ากัน ในที่นี้คือ 38x144 พิกเซล
- 3) งานวิจัยชิ้นนี้ศึกษาและพัฒนาขึ้นเพื่อรู้จำลายมือชื่อแบบ Off-Line

#### 1.5 ประโยชน์ที่ได้รับ

ประโยชน์ที่เกิดขึ้นจากงานวิจัยนี้ ประกอบด้วย

- 1) ผู้ใช้สามารถเลือกใช้อัลกอริทึมพื้นฐาน สำหรับสร้างโมเดลการรู้จำลายมือชื่อได้อย่างถูกต้องเหมาะสม
- 2) อัลกอริทึมการรู้จำลายมือชื่อที่ถูกพัฒนาขึ้นมาใหม่ สามารถเพิ่มความแม่นยำในการทำนายได้
- 3) การเก็บข้อมูลเป็นการเก็บรวบรวมจากข้อมูลจริง ซึ่งแสดงให้เห็นว่าโมเดลการรู้จำลายมือชื่อนี้สามารถนำไปใช้ได้จริง รวมไปถึงฐานข้อมูลภาพลายมือชื่อนี้ สามารถนำไปใช้เป็นแหล่งอ้างอิงให้แก่งานวิจัยชิ้นอื่น ๆ ได้ในอนาคต



## บทที่ 2

### ปริทัศน์วรรณกรรมและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงปริทัศน์วรรณกรรมและงานวิจัยที่เกี่ยวข้องกับวิทยานิพนธ์นี้ โดยเนื้อหาจะประกอบด้วย การเรียนรู้ของเครื่อง (Machine Learning) ไบโอมेटริกซ์ (Biometrics) การรู้จำแบบ (Pattern Recognition) การทำเหมืองข้อมูลสำหรับการจำแนกประเภท (Data Mining for Classification) อัลกอริทึมการรู้จำแบบประเภทต่าง ๆ ได้แก่ อัลกอริทึมเพอร์เซปตรอน (Perceptron) อัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine) อัลกอริทึมนาอิวเบย์ (Naïve Bayes) อัลกอริทึมเคเน็ยเรสเนเบอร์ (k-Nearest Neighbors) การตรวจหาขอบภาพ และการทำภาพให้บางลง (Edge Detection and Thining) โปรแกรม MATLAB ภาษาไพธอน (Python Language) รวมถึงซอฟต์แวร์สไปเดอร์ (Spyder) ที่ใช้เป็นสภาพแวดล้อมในการพัฒนาโปรแกรมไพธอน และงานวิจัยที่เกี่ยวข้อง (Literature Review)

#### 2.1 การเรียนรู้ของเครื่อง (Machine Learning)

การเรียนรู้ของเครื่อง คือการพยายามทำให้เครื่องคอมพิวเตอร์ หรือเครื่องอิเล็กทรอนิกส์ มีความสามารถในการทำงานที่มีความฉลาดได้ เป็นสาขาหนึ่งของปัญญาประดิษฐ์ ที่เกี่ยวข้องกับการพัฒนาเทคนิควิธี เพื่อให้คอมพิวเตอร์สามารถเรียนรู้รูปแบบหรือโมเดลจากข้อมูลเข้า โดยเรียนรู้จากข้อมูลบางส่วน และเน้นที่วิธีการวิเคราะห์ชุดข้อมูล

เทคนิคการเรียนรู้ของเครื่อง (Mitchell, 1997) ถูกใช้เพื่อเพิ่มประสิทธิภาพในการแก้ปัญหาต่าง ๆ เช่น การสร้างให้คอมพิวเตอร์สามารถแยกแยะวัตถุ เสียง ตัวอักษร หรือจำแนกรูปภาพต่าง ๆ ได้ กระบวนการจำแนกข้อมูลจำนวนมากที่มนุษย์ต้องอาศัยระยะเวลาอย่างมากในการทำนั้น เราสามารถฝึกสอนกระบวนการดังกล่าวนี้ให้แก่เครื่องจักร หรือคอมพิวเตอร์ให้สามารถประมวลผลหรือทำแทนมนุษย์ได้

หากแบ่งประเภทเทคนิคการเรียนรู้ของเครื่องจักรตามลักษณะการใช้ข้อมูลฝึกสามารถแบ่งได้ดังนี้

### 1) การเรียนรู้แบบมีผู้สอน (Supervised Learning)

การเรียนรู้ลักษณะนี้เปรียบได้กับการเรียนรู้แบบมีครูฝึก ทำหน้าที่ให้คำตอบ จึงต้องมีการเตรียมข้อมูลฝึกที่มีผลเฉลยไว้ และหลังจากการเรียนรู้ จะได้รูปแบบที่สามารถเชื่อมโยงจากข้อมูลไปสู่ผลเฉลย ก็จะทำการทดสอบเพื่อวัดประสิทธิภาพของความรู้หรือรูปแบบที่ได้จากการเรียนรู้ ตัวอย่างอัลกอริทึมในกลุ่มนี้ ได้แก่ Perceptron, Support Vector Machine, Naïve bayes

### 2) การเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning)

การเรียนรู้ลักษณะนี้เป็นการเรียนรู้แบบไม่ต้องใช้ข้อมูลฝึกที่มีคำตอบ ไม่มีการบอกผลลัพธ์เป็นอย่างไร ไม่มีผิด ไม่มีถูก เป็นการเรียนรู้เพื่อหาความสัมพันธ์ของข้อมูล คำตอบที่ได้จากโมเดลจำพวกนี้ จึงเน้นเพื่อการอธิบายความสัมพันธ์ของสิ่งที่มีโอกาสเกิดขึ้นพร้อมกัน ตัวอย่างประสิทธิภาพของการเรียนรู้แบบนี้ จึงเป็นค่าสหพหุคูณ และค่าความเชื่อมั่น ตัวอย่างอัลกอริทึมในกลุ่มนี้ ได้แก่ Apriori

### 3) การเรียนรู้แบบกึ่งมีผู้สอน (Semi-Supervised Learning)

การเรียนรู้ลักษณะนี้เป็นการเรียนรู้ที่ผู้ใช้บอกข้อมูลบางอย่างกับเครื่องแต่จะไม่ได้บอกทั้งหมด เช่น ผู้ใช้บอกให้แบ่งกลุ่ม 3 กลุ่ม แต่ไม่บอกว่าจะต้องแบ่งอย่างไร ให้เครื่องไปคิดและประมวลผลเอง ลักษณะดังที่กล่าวมานี้ เป็นการบอกข้อมูลบางอย่างเท่านั้น ดังนั้นการเรียนรู้วิธีนี้จะซับซ้อนกว่าการเรียนรู้แบบมีผู้สอน อัลกอริทึมในกลุ่มนี้ ได้แก่ k-Means

## 2.2 ไบโอมेटริกซ์ (Biometrics)

ระบบไบโอมेटริกซ์เป็นหนึ่งในระบบการจดจำรูปแบบ (Unar et al., 2014) ซึ่งจะมีการตรวจสอบ หรือวิเคราะห์ คุณสมบัติเด่น หรือลักษณะที่แตกต่างกันของข้อมูลที่ได้มา กับคุณสมบัติของข้อมูลที่เก็บไว้ในฐานข้อมูลอยู่แล้ว เพื่อยืนยันว่าเป็นข้อมูลของบุคคลเดียวกันหรือไม่ สำหรับการไบโอมेटริกซ์นั้นจะประกอบด้วย

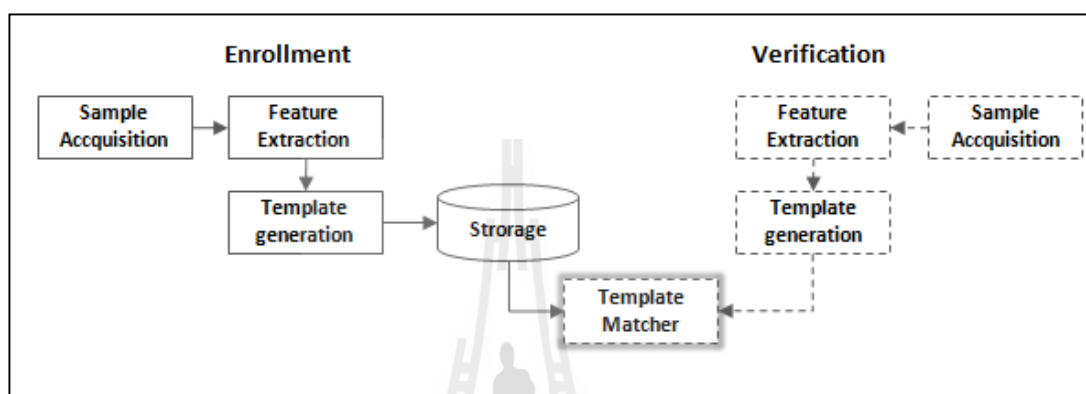
1) โมดูลการรวบรวมข้อมูล หรือภาพ (Image Acquisition Module) ในส่วนนี้คือกระบวนการได้รับภาพของลักษณะไบโอมेटริกซ์เป็นข้อมูลเข้า และส่งต่อไปยังระบบการประมวลผลต่อไป

2) โมดูลการคัดกรองคุณสมบัติ (Feature Extraction Module) จะเป็นกระบวนการประมวลผลภาพ หรือ ข้อมูลที่ได้มาแล้วนำมาคัดกรองเอาคุณสมบัติเด่น หรือ ลักษณะที่แตกต่างกันชัดเจน

3) โมดูลการจับคู่ (Matcher Module) คือการจับคู่คุณสมบัติของข้อมูล ที่ทำการตรวจสอบกับข้อมูลของสิ่งที่มีอยู่แล้วในแกลลอรี่ หรือฐานข้อมูล ตัวโมดูลการตัดสินใจที่ฝังอยู่ (Embedded

Decision Making Module) จะทำการวิเคราะห์ข้อมูล โดยการทำการยอมรับ หรือปฏิเสธนั้น จะขึ้นอยู่กับทำให้คะแนนในขณะที่ตัดสินใจ

4) โมดูลฐานข้อมูล (Database Module) ทำหน้าที่เก็บรวบรวมกลุ่มตัวอย่างที่ได้มาก่อนหน้านี้ในขั้นตอนลงทะเบียน มักจะเรียกข้อมูลไบโอเมตริกซ์ที่บันทึกไว้ว่าแม่แบบ จากรูปที่ 2.1 แสดงให้เห็นถึงองค์ประกอบของระบบไบโอเมตริกซ์ทั่วไป



รูปที่ 2.1 แผนภาพแสดงองค์ประกอบของระบบไบโอเมตริกซ์ (Unar et al., 2014)

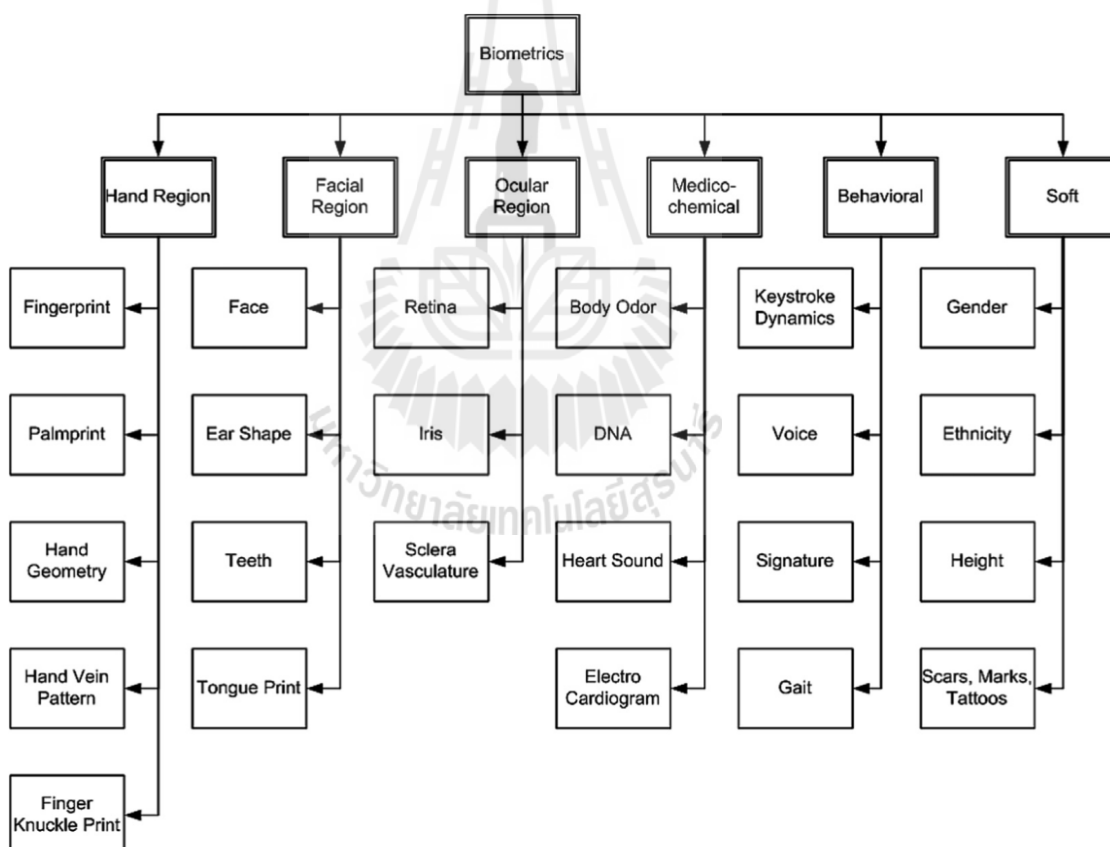
ระบบไบโอเมตริกซ์จะมีโหมดในการทำงาน โหมดใดโหมดหนึ่งดังต่อไปนี้

1) โหมดการตรวจสอบ (Verification Mode) หลังจากผู้ใช้ส่งลายมือชื่อให้กับระบบไบโอเมตริกซ์ ระบบจะทำการตรวจสอบตัวตนของแต่ละบุคคล ผ่าน PIN, Login name ฯลฯ ในการตอบสนอง ระบบจะทำการรับรู้และทำการตรวจสอบ กับข้อมูลของผู้ใช้ที่ร้องขอเข้ามา แบบ 1:1 (หนึ่งต่อหนึ่ง) โดยจะเปรียบเทียบระหว่าง ไบโอเมตริกซ์ลายมือชื่อที่ส่งเข้ามาพร้อมกับการร้องขอ และ ไบโอเมตริกซ์ลายมือชื่อของผู้ใช้งานที่ได้ทำการลงทะเบียนไว้ การใช้งานในโหมดการตรวจสอบนี้ จะประยุกต์ใช้ในการเข้าสู่ระบบคอมพิวเตอร์ ตู้ ATM การใช้งาน E-Commerce การควบคุมการเข้าถึงและการตรวจสอบการใช้งานบนอุปกรณ์มือถือ

2) โหมดระบุตัวตน (Identification Mode) ในโหมดนี้ระบบจะพยายามรับรู้ และจดจำลักษณะของผู้ใช้ โดยการเปรียบเทียบลายมือชื่อไบโอเมตริกซ์ส่งไปยังทุกลายมือชื่อที่ลงทะเบียนในฐานข้อมูลโดยเป็นปฏิบัติการแบบ 1: N (หนึ่งต่อหลายคน) เปรียบเทียบโดยไม่ต้องเรียกชื่อเอกลักษณ์เฉพาะจากผู้ใช้ (อย่างเช่น Login Name) ในความเป็นจริงการทำงานในลักษณะนี้จะป้องกันไม่ให้บุคคลใดบุคคลหนึ่งใช้ข้อมูลเฉพาะตัวในหลายการใช้งาน ซึ่งการทำงานในลักษณะนี้มักจะนำไปใช้กับ การออกบัตรหนังสือเดินทาง ใบอนุญาตขับรถ ใบอนุญาตข้ามพรมแดน และการเบิกจ่ายสวัสดิการ

3) โหมดการคัดกรอง (Screening Mode) โหมดนี้เป็นส่วนขยายของโหมดระบุตัวตน ที่ระบบไบโอเมตริกซ์มั่นใจว่า บุคคลที่ถูกระบุอยู่นั้นจะถูกตรวจสอบและผ่านการคัดกรอง โดยการทำงานจะยังคงเป็นแบบ 1: N (หนึ่งต่อหลายคน) ซึ่งจะเปรียบเทียบตลอดทั้งฐานข้อมูล ตัวอย่างการใช้งานนั้น จะพบเห็นได้บ่อยในการรักษาความปลอดภัยบริเวณท่าอากาศยาน กิจกรรมเฝ้าระวังสถานที่สาธารณะ และความปลอดภัยของประชาชนในเหตุการณ์อื่น ๆ

มนุษย์นั้นมีลักษณะหลายอย่างที่โดดเด่น ซึ่งสามารถนำมาใช้งานในการระบุและการตรวจสอบตัวบุคคลได้ เพื่อความสะดวกในการแบ่งหมวดหมู่ในการกล่าวถึงไบโอเมตริกซ์ เราสามารถแบ่งออกเป็นหมวดย่อยที่แตกต่างกันตามตำแหน่งของที่ปรากฏในร่างกายมนุษย์เช่น (1) คุณลักษณะต่าง ๆ บนมือ (2) คุณลักษณะบนใบหน้า (3) คุณลักษณะเกี่ยวกับตา และ รอบดวงตา (4) ลักษณะพฤติกรรม และ (5) และคุณลักษณะต่าง ๆ ทางชีววิทยา และ เคมี ดังที่ได้แสดงในรูปที่ 2.2



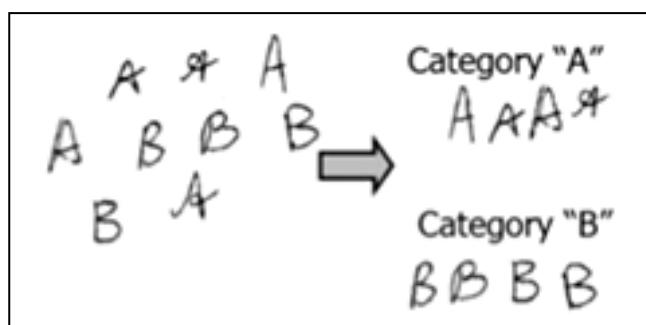
รูปที่ 2.2 แสดงการจัดหมวดหมู่ของไบโอเมตริกซ์ (Unar et al., 2014)

## 2.3 การรู้จำแบบ (Pattern Recognition)

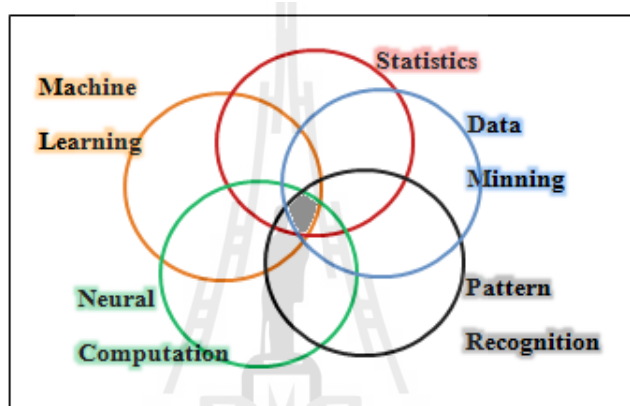
มนุษย์เรานั้นมีความสามารถในการรู้จำวัตถุสิ่งของ รวมไปถึงคุณลักษณะของสิ่งเหล่านั้น ได้จากการรับรู้ และจดจำ เนื่องจากเป็นความสามารถที่เราได้รับมาตั้งแต่กำเนิด (Dougherty, 2012) เป็นเรื่องยากมากที่เราจะนำความสามารถเหล่านี้มาวิเคราะห์และนำไปใส่ในระบบ หรือ กระบวนการใดกระบวนการหนึ่ง อย่างเช่น มันเป็นเรื่องที่ไม่ยากเลยสำหรับมนุษย์ที่จะบอกว่า “เสียงที่ได้ยินนั้นเป็นเสียงไวโอลิน”, “ลายมือที่เราเห็นนี้เป็นเลขสาม หรือ เลขแปด”, “กลิ่นนี้เป็น กลิ่นหัวหอม หรือ กลิ่นดอกกุหลาบ” หรือแม้กระทั่ง เราสามารถจดจำและแยกแยะบุคคลแต่ละ บุคคลรอบ ๆ ตัวเรา ได้จากใบหน้าโดยเพียงแค่ใช้สติตรึงตรอง แต่สำหรับเครื่องคอมพิวเตอร์นั้น การแยกแยะใบหน้าบุคคลได้จะต้องวิเคราะห์หาองค์ประกอบโครงสร้างของใบหน้า และด้วยการ วิเคราะห์นี้ นำไปสู่การสร้าง “รูปแบบ” ของใบหน้าแต่ละบุคคล ซึ่งถ้าหากมีใบหน้าของบุคคลเดิมนี้ เข้ามา ระบบก็จะสามารถทำการตอบสนองและบอกได้ว่าเป็นบุคคลเดียวกัน (แม้หากปฏิกิริยา หรือ สีหน้าท่าทางจะเปลี่ยนไปบ้าง) สิ่งเหล่านี้เราเรียกมันว่า การรู้จำรูปแบบ ซึ่งในแต่ละระบบนั้นอาจจะ มีหลาย ๆ คลาส (Class) ซึ่งเราต้องทำการเลือก หรือจำแนก ใบหน้าของแต่ละบุคคลและจัดแบ่งเป็น คลาส

คลาส คือ การรวมตัวกันของกลุ่มตัวอย่างที่มีคุณลักษณะที่คล้ายกัน แต่ไม่จำเป็นต้อง เหมือนกันทั้งหมดเลยทีเดียว และที่สำคัญคือต้องมีลักษณะร่วมที่ต่างจากคลาสนั้น ดังรูปที่ 2.3

ความสนใจในปัญหาเรื่องการรู้จำรูปแบบ และการจำแนกนั้น ได้เพิ่มขึ้นอย่างรวดเร็วในยุค ปัจจุบัน เนื่องจากมีเทคโนโลยี และ เครื่องมือที่ทันสมัยมาสนับสนุน การรู้จำนั้นยังสามารถนำไป ประยุกต์ใช้กับงานในหลาย ๆ ด้าน เช่น ด้านไบโอเมตริกซ์ นำไปใช้ในการระบุตัวบุคคล โดยใช้ คุณลักษณะทางชีวภาพ เช่น ลายนิ้วมือ ม่านตา การตรวจสอบ DNA การตรวจสอบลายมือเขียน หรือ การตรวจสอบลายมือชื่อ ด้านแมชชีนวิชัน (Machine Vision) สามารถนำไปประยุกต์ใช้กับ สายการผลิต (Manufactory Line) เช่น การตรวจสอบรอยขีดข่วนบนชิ้นงานต่าง ๆ ด้านการรู้จำ เอกสาร (Document Recognition) เช่น การตรวจสอบรู้จำรูปแบบข้อความของ Spam Mail



รูปที่ 2.3 แสดงการจัดกลุ่มที่จำแนก “A” และ “B” ออกจากกัน (Dougherty, 2012)



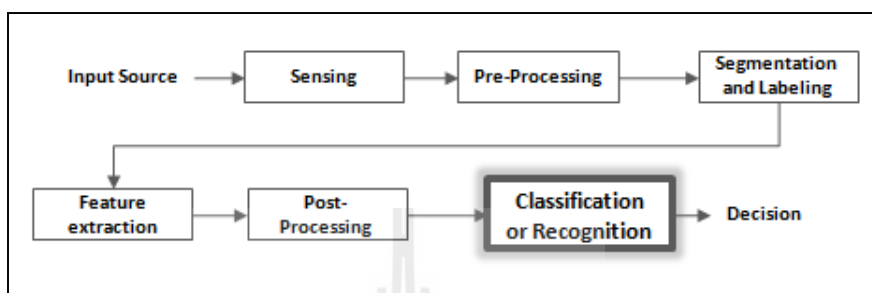
รูปที่ 2.4 การรู้จำ และองค์ความรู้ที่เกี่ยวข้อง

ในส่วนของวิธีการ และ เทคนิคที่นำมาใช้ในการพัฒนาระบบการรู้จำรูปแบบนั้น แตกต่างกันไปตามแต่ละสายงานที่นำไปประยุกต์ใช้ (รูปที่ 2.4) ซึ่งจะมีเทคนิคต่าง ๆ อันเป็นที่นิยมนำมาใช้ร่วมพัฒนา ได้แก่ โครงข่ายประสาทเทียม (Neural Network) การทำเหมืองข้อมูล การเรียนรู้ของเครื่อง รวมไปถึงเทคนิค และ องค์ความรู้อื่น ๆ ในสถิติอีกด้วย

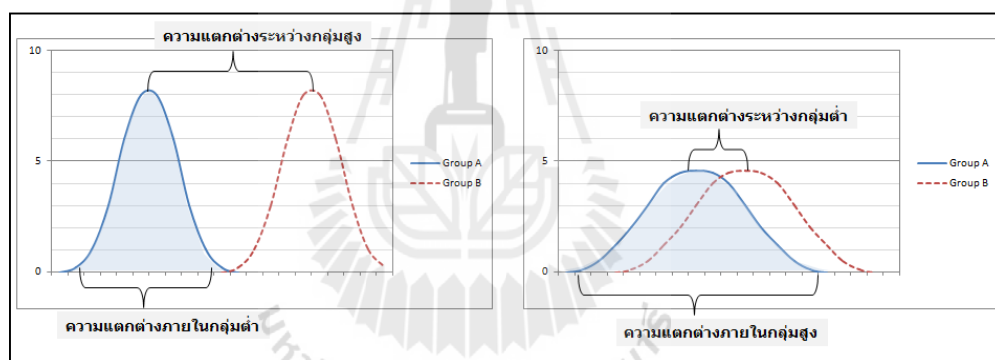
## 2.4 การทำเหมืองข้อมูลสำหรับการจำแนกประเภท (Data Mining for Classification)

การจำแนกประเภท ส่วนใหญ่จะอยู่ในขั้นตอนช่วงท้ายของกระบวนการทำเหมืองข้อมูล (Dougherty, 2012) โดยทั่วไปแล้ว การจำแนกประเภทคือการเลือกวัตถุ หรือข้อมูล ให้ไปอยู่ตามคลาสต่าง ๆ ดังที่แสดงให้เห็นในรูปที่ 2.5 โดยในกรณีของการจำแนกประเภทรูปภาพแล้ว รูปภาพที่ได้รับมาถูกจำแนกให้อยู่ในคลาสเดียวกันนั้น อาจ会有ความแตกต่างกันบ้างเล็กน้อย แต่ถ้าเทียบความแตกต่างระหว่างคลาสแล้วจะมีมากกว่าอย่างเห็นได้ชัดเจน

คุณภาพของรูปภาพ หรือ ข้อมูลที่จะนำมาใช้ในการจำแนกนั้นขึ้นอยู่กับกระบวนการเก็บรวบรวมข้อมูล รวมถึงขั้นตอนการปรับแต่งรูปภาพ ดังนั้นกระบวนการก่อนนำมาเข้าสู่การจำแนก หรือ ที่เรียกว่า Pre-processing นั้นจึงมีความสำคัญเป็นอย่างมาก โดยอาจจะมีหลายเทคนิคมาเกี่ยวข้องเพื่อให้ได้ข้อมูล หรือ รูปภาพที่มีคุณภาพมากยิ่งขึ้น และเหมาะสมกับการนำไปจำแนก



รูปที่ 2.5 ขั้นตอนการทำงานโดยทั่วไปของกระบวนการทำเหมืองข้อมูล (Dougherty, 2012)



รูปที่ 2.6 ลักษณะตัวอย่างการจำแนกประเภทที่ดี และไม่ดี

ดังตัวอย่างในรูปที่ 2.6 เป็นการแสดงถึงมาตรวัดการจำแนกที่ดีและไม่ดี โดยรูปด้านซ้ายที่แสดงถึงการจำแนกที่ดีนั้น จะมีคุณลักษณะที่แตกต่างกันมากระหว่างคลาส และในคลาสเดียวกันนั้นจะต้องมีความแตกต่างกันให้น้อยที่สุด เมื่อเปรียบเทียบกับด้านขวาซึ่งเป็นการจำแนกที่ไม่ดี ซึ่งจะเห็นได้ว่าความแตกต่างกันระหว่างคลาสจะต่ำ และความแตกต่างกันภายในคลาสจะสูง

การจำแนกในลักษณะของกระบวนการอัตโนมัติ นั้น ได้ถูกพัฒนาขึ้นอย่างต่อเนื่อง และนำไปประยุกต์ใช้งานได้ในหลาย ๆ ด้าน ยกตัวอย่างเช่น การจำแนกใบหน้าของแต่ละบุคคล ดังรูปที่ 2.7 นั้นเป็นรูปใบหน้าของบุคคลเดียวกัน ที่แตกต่างกันออกไปตาม อิริยาบถ อารมณ์ แสงเงา และเครื่องประดับบนใบหน้า ซึ่งหากตัวระบบต้องการจะให้จำแนกโดยนำไปใช้ระบุความเป็นตัวบุคคลแล้ว ภาพด้านล่างควรจะถูกจัดให้อยู่ในคลาสเดียวกัน

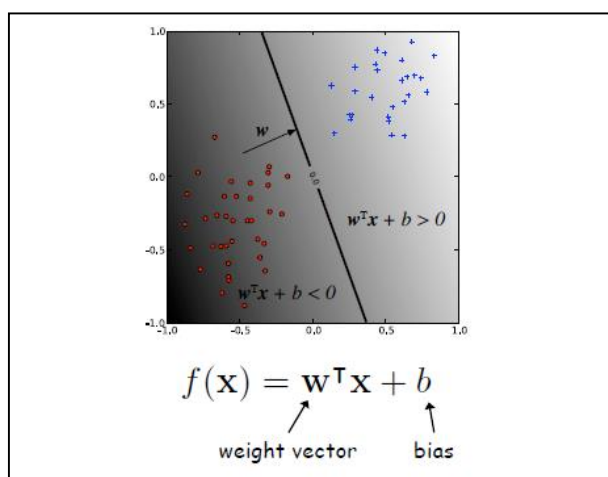


รูปที่ 2.7 ภาพใบหน้าของบุคคลเดียวกันในอิริยาบถ อารมณ์ ความสว่าง และองค์ประกอบที่ต่างกัน (Dougherty, 2012)

## 2.5 อัลกอริทึมเพอร์เซปตรอน (Perceptron)

เพอร์เซปตรอนเป็นอัลกอริทึมที่ใช้ในการจำแนกประเภท (กิตติศักดิ์ 2557; Flach, 2012; Minsky et al., 1969; Theodoridis et al., 2010) ที่นำเสนอถึงวิธีการเรียนรู้ของโครงข่าย โดยจะถูกนำไปประยุกต์ในการแก้ปัญหาในด้านการรู้จำรูปแบบ โดยการเรียนรู้นั้นจะเป็นไปอย่างอัตโนมัติและไม่ซับซ้อน นอกจากนั้นพฤติกรรมของระบบโครงข่ายจะยังมีความสามารถในการเรียนรู้ได้แม้ว่าค่าน้ำหนัก และ ค่าไบอัส (Bias) เริ่มต้นจะเป็นค่าจากการสุ่ม เพอร์เซปตรอนนั้นมีพื้นฐานมาจากแบบจำลองเชิงเส้น (Linear Model) แบบจำลองเชิงเส้น คือ การใช้สมการเส้นตรง เพื่อนำมาจำแนกกลุ่มสองกลุ่มออกจากกัน ดังรูปที่ 2.8





รูปที่ 2.8 ตัวอย่างการจำแนกด้วยแบบจำลองเชิงเส้น และสมการ (กิตติศักดิ์ 2557)

ข้อดีของแบบจำลองเชิงเส้นนั้น มีดังต่อไปนี้

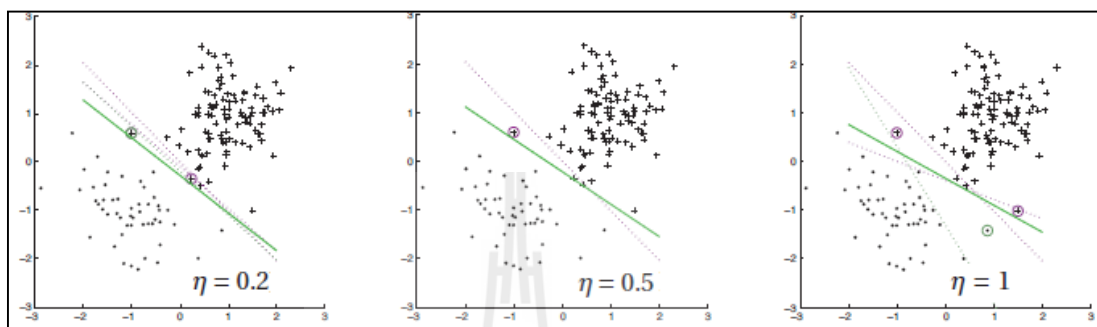
- 1) มีความเป็นจุดเริ่มต้นที่ดี ให้ค่าเริ่มต้นที่ดี
- 2) เข้าใจง่าย และใช้เวลาไม่มากในการสร้างแบบจำลอง
- 3) คงทนต่อการเปลี่ยนแปลง ไม่เปลี่ยนแปลงง่าย
- 4) มีโอกาสเกิดเหตุการณ์ Overfitting กับข้อมูลฝึกค่อนข้างน้อย
- 5) นำไปปรับใช้กับอัลกอริทึมทั่วไปได้หลากหลาย

อัลกอริทึมของเพอร์เซปตรอน แสดงได้ดังสมการต่อไปนี้

$$w' = w + \eta x_i \quad (2.1)$$

เมื่อ	$w'$	คือ ค่าน้ำหนักของข้อมูลที่ถูกปรับ
	$w$	คือ ค่าน้ำหนักของข้อมูลเดิม
	$x_i$	คือ ข้อมูลฝึกที่ได้แบ่งประเภทไว้แล้ว
	$\eta$	คือ ค่าสัมประสิทธิ์การเรียนรู้ (Learning Rate)

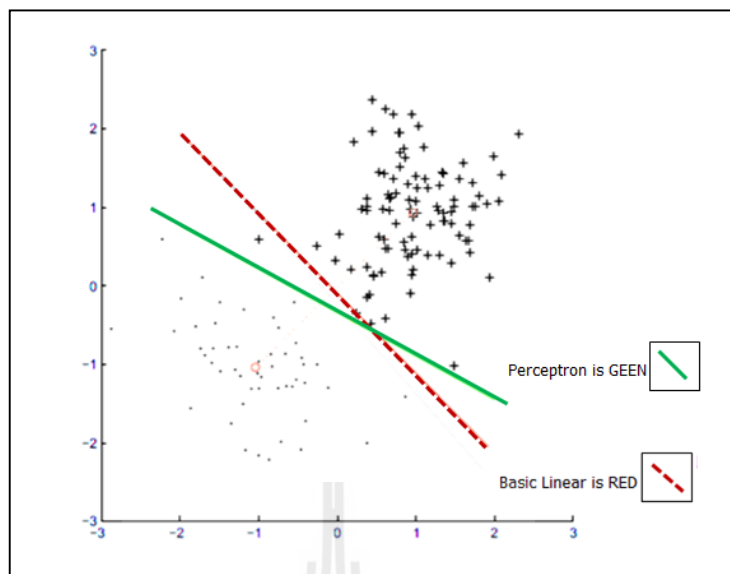
จุดสำคัญของอัลกอริทึมเพอร์เซปตรอน คือทุกครั้งที่ตัวอย่าง  $x_i$  นั้นถูกจัดกลุ่มผิด เราจะทำการถ่วงน้ำหนักของ  $x_i$  ลงไป และหลังจากการเทรนเสร็จแล้ว แต่ละตัวอย่างจะถูกจัดกลุ่มอย่างสมบูรณ์โดยจะมี  $x_i$  ที่ถูกจัดกลุ่มผิดเป็นศูนย์ หรือมีความผิดพลาดในการจัดกลุ่มผิดให้น้อยที่สุดที่เป็นไปได้



รูปที่ 2.9 แสดงการเปลี่ยนแปลงของเส้นแบ่ง ในแต่ละค่าสัมประสิทธิ์การเรียนรู้ (Flach, 2012)

จากรูปที่ 2.9 (ด้านซ้าย) แสดงถึงการเทรนของเพอร์เซปตรอน ด้วยค่าสัมประสิทธิ์การเรียนรู้ที่น้อย ๆ ( $\eta = 0.2$ ) กลุ่มตัวอย่างที่ถูกวางกลมไว้ เป็นหนึ่งในตัวอย่างที่ถูกถ่วงน้ำหนัก โดยเส้นแบ่งนั้นจะเริ่มจากวิธีการของการจำแนกด้วยแบบจำลองเชิงเส้น ในกราฟตรงกลาง แสดงการเพิ่มค่าสัมประสิทธิ์การเรียนรู้เป็น 0.5 ทำให้การแบ่งกลุ่มสามารถเกิดขึ้นได้อย่างรวดเร็ว และกราฟด้านขวานั้น เมื่อค่าสัมประสิทธิ์การเรียนรู้เท่ากับ 1 ทำให้การอัปเดตค่าน้ำหนัก มีผลในทางที่เกิดการเปลี่ยนแปลงอย่างรวดเร็วเกินไป จึงทำให้เกิดการเปลี่ยนเส้นแบ่งหลายครั้ง

ดังนั้นจึงสรุปได้ว่า เพอร์เซปตรอนเป็นตัวจำแนกเชิงเส้นประเภทหนึ่ง ซึ่งแบ่งแยกได้ดีกว่าการจำแนกเชิงเส้นทั่วไป หลักการคือการพยายามลากเส้นแบ่งระหว่างกลุ่ม โดยที่โมเดลจะพยายามหาเส้นแบ่งที่ดีที่สุด จึงต้องอาศัยการทำซ้ำในทุก ๆ จุด เป็นอีกหนึ่งวิธีที่ง่ายเหมาะสำหรับการเรียนรู้ขนาดใหญ่ โดยไม่จำเป็นต้องมีอัตราการเรียนรู้เริ่มต้น การหาเส้นแบ่งกลุ่มที่ดีที่สุดนั้น โมเดลจะพยายามปรับค่าสัมประสิทธิ์การเรียนรู้ไปเรื่อย ๆ จนถึงจุดที่มันคิดว่าเหมาะสมที่สุดแล้ว จึงหยุดคำนวณ ดังนั้นหากกลุ่มข้อมูลอยู่แยกกันแบบชัดเจน เพอร์เซปตรอนจะสามารถแบ่งแยกข้อมูลประเภทนี้ได้ดี พิจารณาได้จากรูปที่ 2.10



รูปที่ 2.10 การจำแนกข้อมูลเชิงเส้น และเพอร์เซปตรอน (Flach, 2012)

รูปที่ 2.10 นี้แสดงให้เห็นถึงความแตกต่างระหว่างการแทนด้วย การจำแนกเชิงเส้น ของชุดข้อมูลที่แทนด้วยเครื่องหมายบวก 100 ตัว และเครื่องหมายจุด 50 ตัว โดยเส้นประสีแดง คือเส้นที่เกิดจากการแบ่งกลุ่มข้อมูลด้วยเส้นตรงแบบพื้นฐาน (Basic Linear Classification) ส่วนเส้นทึบสีเขียว คือการแบ่งกลุ่มข้อมูลโดยใช้ เพอร์เซปตรอน สังเกตเห็นได้ว่าการแบ่งด้วยเพอร์เซปตรอนสามารถแยกกลุ่มเครื่องหมายบวก ออกจากกลุ่มเครื่องหมายจุด ได้อย่างสมบูรณ์แบบ แต่การตัดแบ่งกลุ่มลักษณะนี้ ในบางกรณีอาจจะทำให้เกิดการแบ่งกลุ่มแบบชิดขอบของข้อมูลมากเกินไป ซึ่งจะไม่ส่งผลดีกับข้อมูลใหม่ในอนาคต

## 2.6 อัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine)

ซัพพอร์ตเวกเตอร์แมชชีน เป็นอัลกอริทึมที่ใช้ในการจำแนกประเภทหนึ่ง มีพื้นฐานมาจากแบบจำลองเชิงเส้น (กิตติศักดิ์ 2557; Cristianini et al., 2000; Flach 2012; Garreta and Moncecchi, 2013; Theodoridis et al., 2010) ซึ่งเป็นอัลกอริทึมที่ถูกพัฒนามาจากเพอร์เซปตรอน และเพิ่มลักษณะยึดหยุ่นด้วยการพยายามปรับเส้นแบ่งให้เกิดระยะขอบ (Margin) มาก ๆ จัดเป็นเทคนิคที่นำมาใช้ในการแก้ปัญหาด้านการรู้จำอีกเทคนิคหนึ่ง และมักจะเป็นเทคนิคที่ให้ความแม่นยำในการจำแนกข้อมูลสูง แนวคิดคือการนำค่าของกลุ่มข้อมูลมาวางลงในฟีเจอร์สเปซ (Feature Space) ต่อจากนั้นนำจุดขอบ (Convex Hull) มาลากเส้นขอบเชื่อมต่อกัน ดังรูปที่ 2.11 (ซัพพอร์ตเวกเตอร์แมชชีน จะไม่ได้ใช้ข้อมูลทุกจุดมาคิดคำนวณเหมือนเพอร์เซปตรอน ซึ่งวิธีนี้เองเรียกได้ว่าเป็นข้อดี

ที่ทำให้ใช้เวลาน้อยลง) หลังจากลากเส้นเชื่อมขอบได้แล้ว ก็จะทำการสร้างเส้นตรงที่ขนานกันระหว่างจุดขอบทั้งสองกลุ่มนั้น จากนั้นจะพยายามสร้างเส้นแบ่งตรงกึ่งกลางระหว่างกลุ่มทั้งสอง โดยให้มีระยะขอบที่มากที่สุด เพื่อแบ่งข้อมูลทั้งสองออกจากกัน ดังรูปที่ 2.12 และในบางกรณี เพื่อให้เกิดระยะขอบมาก ๆ จึงต้องใช้ตัวแปรอนุโลม (Slack Variable) เข้ามาช่วย คือการยอมให้เกิดการทำนายผิดได้บ้าง หรือยอมมองข้ามจุดขอบบางจุดไป เพื่อให้ได้มาซึ่งระยะการแบ่งที่มากที่สุด วิธีแบบนี้เป็นวิธีที่ถือได้ว่าค่อนข้างยืดหยุ่น ซึ่งจะพิจารณาได้จากรูปที่ 2.13

วิธีการของซัพพอร์ตเวกเตอร์แมชชีน จะถูกอธิบายด้วยวิธีการคำนวณ โดยกำหนดให้  $(x_i, y_i), \dots, (x_n, y_n)$  เป็นข้อมูลตัวอย่างที่ใช้สำหรับการสอน โดย  $n$  คือจำนวนข้อมูลตัวอย่าง  $m$  คือจำนวนมิติข้อมูลเข้าและ  $y$  คือผลลัพธ์ที่แทนประเภทหรือกลุ่มข้อมูลมีค่า  $+1$  หรือ  $-1$  เมื่อ  $+1$  แทนความหมาย Positive และ  $-1$  แทนความหมาย Negative อธิบายดังสมการต่อไปนี้

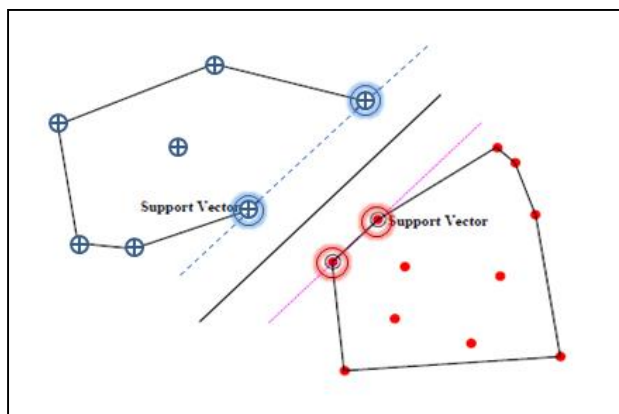
$$(x_i, y_i), \dots, (x_n, y_n) \text{ เมื่อ } x \in R^m, y \in \{+1, -1\} \quad (2.2)$$

สำหรับปัญหาเชิงเส้น 2 มิติที่เป็นพื้นฐานของการจำแนกข้อมูล สามารถคำนวณได้ ดังสมการที่ 2.3

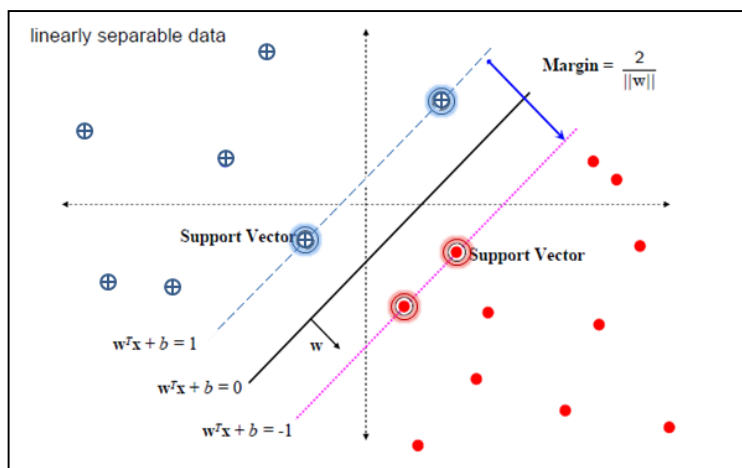
$$(w^T * x) + b \quad (2.3)$$

เมื่อ  $w$  คือค่าน้ำหนัก และ  $b$  คือค่าไบอัส สมการใช้สำหรับจำแนกประเภทของข้อมูล ด้วยวิธีการซัพพอร์ตเวกเตอร์แมชชีน ได้แก่

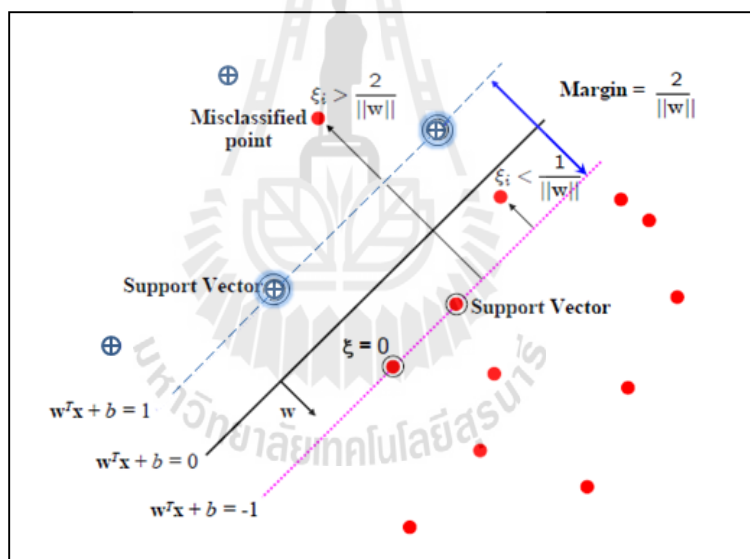
$$(w^T * x) + b > 0 \text{ ถ้า } y_i = +1 \text{ และ } (w^T * x) + b < 0 \text{ ถ้า } Y_i = -1$$



รูปที่ 2.11 รูปภาพแสดงการลากเส้นเชื่อมจุดขอบของแต่ละกลุ่มตัวอย่าง (กิตติศักดิ์ 2557)

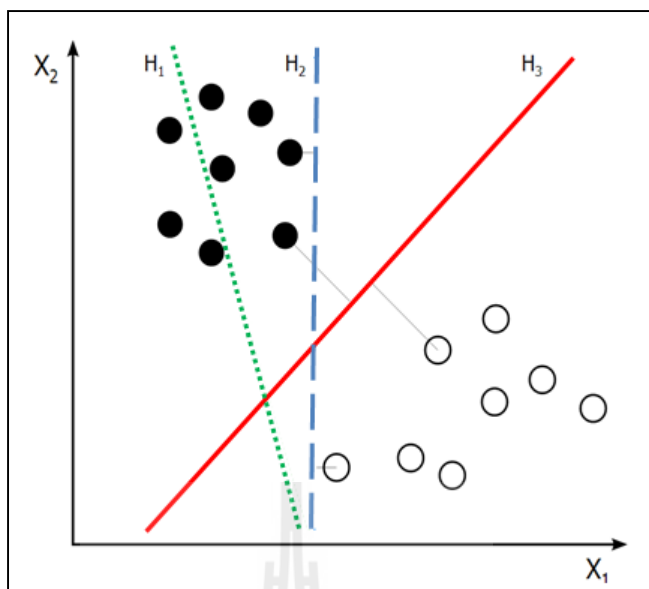


รูปที่ 2.12 การพยายามสร้างเส้นแบ่งตรงกึ่งกลางระหว่างกลุ่มทั้งสอง โดยให้มีระยะขอบที่มากที่สุด (กิตติศักดิ์ 2557)



รูปที่ 2.13 แสดงการใช้ตัวแปรอนุโลม ที่ยอมมองข้ามบางจุดขอบเพื่อให้ได้ Margin ที่มากที่สุด (กิตติศักดิ์ 2557)

นอกจากฟังก์ชันเชิงเส้นแล้ว ในข้อมูลที่มีมิติสูง และข้อมูลไม่สามารถแบ่งแยกได้โดยง่าย ซัพพอร์ตเวกเตอร์แมชชีน ยังมีเคอร์เนลฟังก์ชัน (Kernel Function) ให้ผู้ใช้สามารถประยุกต์ใช้ในการแก้ปัญหาได้หลายวิธี เช่น ฟังก์ชันพหุนาม (Polynomial Function) ฟังก์ชันเรเดียลเบสิส (Radial Basis Function) และ ฟังก์ชันซิกมอยด์ (Sigmoid Function)



รูปที่ 2.14 แสดงการแบ่งข้อมูลด้วยเส้นตรงที่ต่างกัน (Garreta and Moncecchi, 2013)

เส้นแบ่งกลุ่มข้อมูลที่เรียกว่า Hyperplane การสร้างเส้นแบ่งกลุ่มสามารถสร้างได้หลายเส้น แต่จะมีเส้นที่ดีที่สุด ที่ให้ระยะขอบมากที่สุด รูปที่ 2.14 เส้น H1 แสดงการแบ่งกลุ่มที่ไม่สามารถแบ่งจุดสีขาวออกจากจุดสีดำได้ H2 สามารถทำได้แต่จะมี Margin เพียงเล็กน้อย ในขณะที่ H3 สามารถแสดงการแบ่งกลุ่มด้วยระยะ Margin ที่มากที่สุด

## 2.7 อัลกอริทึมนาอิวเบย์ (Naïve Bayes)

นาอิวเบย์ (กิตติศักดิ์ 2557; นิตยา 2557; Flach, 2012; Mohammed et al., 2014; Oded et al., 2010) เป็นวิธีจำแนกประเภทข้อมูลที่มีประสิทธิภาพอีกวิธีหนึ่ง ซึ่งใช้งานได้ดีและเหมาะสมกับกรณีของเซตตัวอย่างที่มีจำนวนมาก และมีสมมุติฐานที่ว่าแอตทริบิวต์ (Attribute) ของตัวอย่างเป็นอิสระไม่ขึ้นต่อกัน อัลกอริทึมนี้ใช้ทฤษฎีของเบย์เข้ามาช่วยในการเรียนรู้จุดมุ่งหมาย ก็เพื่อต้องการสร้างโมเดลที่อยู่ในรูปของความน่าจะเป็นแบบมีเงื่อนไข ข้อดีของวิธีการเรียนรู้แบบนี้ คือ เราสามารถใช้ข้อมูลและความรู้ก่อนหน้า (Prior Knowledge) เข้ามาช่วยในการเรียนรู้ได้ ซึ่งพบว่าวิธีนี้ให้ประสิทธิภาพในการเรียนรู้ได้ดี ไม่ด้อยกว่าวิธีการเรียนรู้ประเภทอื่น ในด้านการคำนวณนั้น จะใช้หลักการของความน่าจะเป็น ซึ่งมีพื้นฐานมาจากทฤษฎีของเบย์ที่แสดงเป็นสมการได้ดังต่อไปนี้

$$P(C|A) = \frac{P(A|C) * P(C)}{P(A)} \quad (2.4)$$

จากสมการของ Bayes จะมี 3 ส่วนที่สำคัญ คือ

1)  $P(C|A)$  หรือ Posterior Probability คือ ค่าความน่าจะเป็นที่ข้อมูลที่มีแอตทริบิวต์เป็น  $A$  จะมีคลาส  $C$

2)  $P(A|C)$  หรือ Likelihood คือ ค่าความน่าจะเป็นที่ข้อมูล Training Data ที่มีคลาส  $C$  และมี แอตทริบิวต์  $A$  โดยที่  $A = a_1 \cap a_2 \dots \cap a_M$  โดยที่  $M$  คือจำนวนแอตทริบิวต์ใน Training Data

3)  $P(C)$  หรือ Prior Probability คือ ค่าความน่าจะเป็นของคลาส  $C$

แต่การที่แอตทริบิวต์  $A = a_1 \cap a_2 \dots \cap a_M$  ที่เกิดขึ้นในการเทรน อาจจะมีจำนวนน้อยมาก หรือไม่มีรูปแบบของแอตทริบิวต์แบบนี้เกิดขึ้นเลย ดังนั้นจึงได้ใช้หลักการที่ว่าแต่ละแอตทริบิวต์เป็นอิสระต่อกัน ทำให้สามารถเปลี่ยนสมการ  $P(A|C)$  ได้เป็น

$$P(A|C) = P(a_1|C) * P(a_2|C) * P(a_3|C) * \dots * P(a_M|C) \quad (2.5)$$

โดยตัวอย่างการคำนวณนั้นแสดงให้เห็นตามตารางตัวอย่างดังต่อไปนี้

ตารางที่ 2.1 ตัวอย่างข้อมูลสภาพอากาศที่มีผลต่อการออกไปเล่นกอล์ฟ

NO	OUTLOOK	TEMPERATURE	HUMIDITY	WINDY	PLAY
1	sunny	Hot	high	FALSE	no
2	sunny	Hot	high	TRUE	no
3	overcast	Hot	high	FALSE	yes
4	rainy	Mild	high	FALSE	yes
5	rainy	Cool	normal	FALSE	yes
6	rainy	Cool	normal	TRUE	no
7	overcast	Cool	normal	TRUE	yes
8	sunny	Mild	high	FALSE	no
9	sunny	Mild	normal	FALSE	yes
10	rainy	Mild	normal	FALSE	yes
11	sunny	Mild	normal	TRUE	yes
12	overcast	Mild	high	TRUE	yes
13	overcast	Hot	normal	FALSE	yes
14	rainy	Mild	high	TRUE	No

จากตารางที่ 2.1 เป็นตัวอย่างแสดงถึงข้อมูลประกอบการตัดสินใจออกไปเล่นกอล์ฟตามสภาพอากาศที่แตกต่างกัน โดยมีทั้งสิ้น 4 แอตทริบิวต์ได้แก่ Outlook , Temperature, Humidity และ Windy จากตารางข้อมูลที่ 2.1 จะได้  $P(\text{Play} = \text{yes}) = 9/14$  หรือ 0.64 ส่วน  $P(\text{Play} = \text{no}) = 5/14$  หรือ 0.36

ตารางที่ 2.2 ตารางแสดงค่าความน่าจะเป็นของแต่ละแอตทริบิวต์

Attribute	play = yes	play = no
Outlook = sunny	$2/9 = 0.22$	$3/5 = 0.60$
Outlook = overcast	$4/9 = 0.45$	$0/5 = 0.00$
Outlook = rainy	$3/9 = 0.33$	$2/5 = 0.40$
Temperature = hot	$2/9 = 0.22$	$2/5 = 0.40$
Temperature = mild	$4/9 = 0.45$	$2/5 = 0.40$
Temperature = cool	$3/9 = 0.33$	$1/5 = 0.20$
Humidity = high	$3/9 = 0.33$	$4/5 = 0.08$
Humidity = normal	$6/9 = 0.67$	$1/5 = 0.20$
Windy = TRUE	$3/9 = 0.33$	$3/5 = 0.60$
Windy = FALSE	$6/9 = 0.67$	$2/5 = 0.40$

จากตารางที่ 2.2 แสดงถึงความน่าจะเป็นที่จะเกิดขึ้นในแต่ละค่า เฉพาะแอตทริบิวต์ที่เราสนใจ อย่างเช่น เราสนใจว่าเมื่อ Outlook = rainy แล้วจะทำให้มีความน่าจะเป็นในการออกไปเล่นกอล์ฟเท่ากับ 0.33 ซึ่งเมื่อมีหลายแอตทริบิวต์ที่นำมาใช้ในการตัดสินใจมากขึ้น เราต้องใช้สมการ (2.4) เข้ามาช่วย เช่น ถ้าหาก

A is Outlook = sunny, Temperature = hot, Humidity = high and Windy = FALSE ความน่าจะเป็นที่จะออกไปเล่นกอล์ฟ จะสามารถคำนวณได้ดังนี้

$$P(\text{play} = \text{yes} | A) = P(\text{Outlook} = \text{sunny} | \text{play} = \text{yes}) * P(\text{Temperature} = \text{hot} | \text{play} = \text{yes}) * P(\text{Humidity} = \text{high} | \text{play} = \text{yes}) * P(\text{Windy} = \text{FALSE} | \text{play} = \text{yes}) * P(\text{play} = \text{yes})$$

ซึ่งจะได้ค่าความน่าจะเป็นในการออกไปเล่นกอล์ฟเท่ากับ 0.0068

และเมื่อมาพิจารณาในส่วนของความน่าจะเป็นของการไม่ออกไปเล่นกอล์ฟ (เมื่อ play = no) ตามการคำนวณดังต่อไปนี้

$$P(\text{play} = \text{no} | A) = P(\text{Outlook} = \text{sunny} | \text{play} = \text{no}) * P(\text{Temperature} = \text{hot} | \text{play} = \text{no}) * P(\text{Humidity} = \text{high} | \text{play} = \text{no}) * P(\text{Windy} = \text{FALSE} | \text{play} = \text{no}) * P(\text{play} = \text{no})$$



จะได้ค่าความน่าจะเป็นในการไม่ออกไปเล่นกอล์ฟเท่ากับ 0.0276 ซึ่งหากสภาพอากาศเป็นดังนี้ความน่าจะเป็นที่จะไม่ออกไปเล่นกอล์ฟจะสูงกว่าการออกไปเล่น และข้อดีอีกอย่างของอัลกอริทึมนี้คือ มีหลายฟังก์ชันให้เลือกใช้ อาทิเช่น เกาเซียน มัลติโนเมียล และเบอร์нулลี

## 2.8 อัลกอริทึมเคเนียร์เซนเบอร์ (k-Nearest Neighbors)

เคเนียร์เซนเบอร์ (นิตยา 2557; Aci et al., 2010; Oded et al., 2010) เป็นอัลกอริทึมที่ได้รับความนิยมในการใช้จำแนกประเภทข้อมูลของการรู้จำรูปแบบ โดยหลักการของอัลกอริทึมนี้ จะทำการจำแนกข้อมูลที่ได้รับมาใหม่โดยอ้างอิงจากข้อมูลเดิมที่มีคุณสมบัติที่ใกล้เคียงที่สุดจำนวน  $k$  ตัว โดยจะมีขั้นตอนดังต่อไปนี้

- 1)  $D$  เป็นชุดข้อมูลที่ประกอบไปด้วย

$$D = \{(p_0, c_0), (p_1, c_1), (p_2, c_2), \dots, (p_n, c_n)\} \quad (2.6)$$

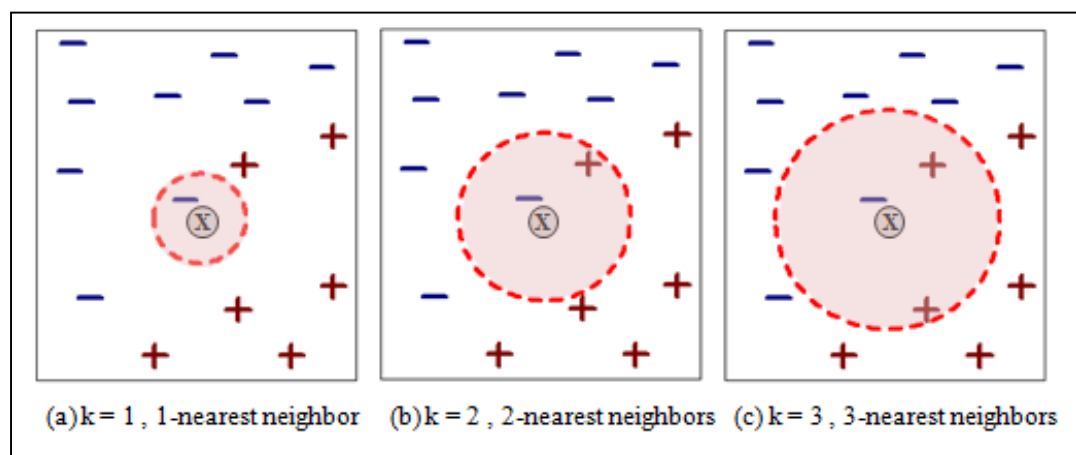
โดยที่  $p_i = \{p_{i_1}, p_{i_2}, p_{i_3}, \dots, p_{i_n}\}$  เป็นเวกเตอร์ของข้อมูล

และ  $c_i$  เป็นประเภทข้อมูล  $p_i$

- 2) ให้  $q = \{q_1, q_2, q_3, \dots, q_n\}$  เป็นเวกเตอร์ข้อมูลใหม่ที่ยังไม่ได้จำแนกประเภท
- 3) สำหรับทุกๆ  $p_i$  ใน  $D$  สามารถคำนวณหาค่าระยะห่างระหว่าง  $q$  และ  $p_i$  โดยใช้วิธียูคลิดีเนียน (Euclidian Distance,  $d$ ) ดังสมการ

$$d(p_i, q) = \sqrt{\sum_{j=1}^n (p_{i_j} - q_j)^2} \quad (2.7)$$

- 4) กำหนดประเภทของ  $q$  จากประเภทของ  $p_i$  ที่มีค่าใกล้เคียงที่สุด  $k$  จำนวน โดยเลือกจากประเภทที่พบบ่อยที่สุด



รูปที่ 2.15 แสดงการจำแนกประเภทด้วยค่า  $k$  ที่ต่างกัน

จากรูปที่ 2.15 แสดงให้เห็นถึงการจำแนกประเภทด้วย เคนีย์เรสเนเบอร์ ด้วยค่า  $k$  ที่ต่างกัน ผลลัพธ์ที่ได้จะเปลี่ยนไปตามประเภทส่วนใหญ่จำนวนของข้อมูลที่อยู่ใกล้ที่สุด  $k$  ตัว อย่างเช่น เมื่อ  $k$  มีค่าเท่ากับ 1 ข้อมูลที่เข้ามาใหม่ (แทนด้วย  $X$ ) จะถูกจำแนกเป็น (-) เมื่อ  $k$  มีค่าเท่ากับ 2 ข้อมูลที่เข้ามาใหม่จะถูกจำแนกเป็น (-) หรือ (+) ก็ได้ ส่วนถ้า  $k$  เท่ากับ 3 ข้อมูลที่เข้ามาใหม่จะถูกจำแนกเป็น (+)

## 2.9 การตรวจหาขอบภาพ และการทำภาพให้บางลง (Edge Detection and Thining)

### 2.9.1 การตรวจหาขอบภาพ

การตรวจหาขอบภาพเป็นวิธีการหาเส้นแสดงรอบรูปร่างของวัตถุ โดยตัดรายละเอียดอื่น ๆ ทิ้งไปเช่น สี หรือ ลวดลาย ซึ่งภาพที่ใช้แสดงเส้นรอบรูปร่างของวัตถุนั้นจะแสดงเป็นภาพไบนารี (Binary Image) การตรวจหาขอบภาพมีหลากหลายวิธี ซึ่งมีหลักการคล้าย ๆ กัน คือ การหาความแตกต่างของความเข้มสีระหว่างภายในภาพที่เป็นระดับสีเทา (Gray Scale) จากจุดหนึ่งไปอีกจุดหนึ่ง หากภาพนั้นมีค่าความเข้มสีแตกต่างกันมาก ก็จะได้ขอบภาพที่ชัดเจน แต่หากความเข้มสีแตกต่างกันน้อย ขอบภาพที่ได้ก็จะไม่ชัดเจน ประโยชน์ของการตรวจหาขอบภาพนั้น สามารถนำไปใช้ในคอมพิวเตอร์วิชัน ด้านอื่น ๆ ได้ เช่น การแบ่งแยกขอบเขตระหว่างวัตถุกับพื้นหลัง และการรู้จำรูปร่างของวัตถุ เป็นต้น (Gonzalez, 2001)

การตรวจหาขอบภาพนั้นสามารถทำหลายวิธี ซึ่งแต่ละวิธีมีขั้นตอนที่คล้ายกัน คือ การใช้ฟิลเตอร์แต่ละแบบในการทำคอนโวลูชัน (Convolution) กับพิกเซล (Pixel) ทั้งหมดภายในภาพ ทั้งในแนวแกน  $X$  และแกน  $Y$  ซึ่งแต่ละวิธีนั้นจะใช้ค่าในฟิลเตอร์แตกต่างกันไป ทำให้ได้ผลลัพธ์ของภาพแสดงเส้นขอบที่แตกต่างกัน ในหัวข้อนี้จะกล่าวถึงวิธีการตรวจหาขอบภาพทั้งหมด 4 วิธี ซึ่งเป็นวิธีที่นิยมใช้ในปัจจุบัน คือ การตรวจหาขอบภาพแบบ โซเบล (Sobel Edge Detection) พรูวิท (Prewitt Edge Detection) โรเบิร์ต (Robert Edge Detection) และ แคนนี่ (Canny Edge Detection)

#### 2.9.1.1 การตรวจหาขอบภาพแบบโซเบล

การตรวจหาขอบภาพแบบโซเบล (Gonzalez, 2001) เป็นการหาขอบภาพโดยใช้ฟิลเตอร์ขนาด  $3 \times 3$  สองฟิลเตอร์ โดยฟิลเตอร์แรกจะใช้หาค่าความแตกต่างในแนวแกน  $X$  และฟิลเตอร์ที่สองจะใช้หาค่าความแตกต่างในแนวแกน  $Y$  แสดงรูปของฟิลเตอร์ดังรูปที่ 2.16 และสมการที่ 2.8

$$M = \sqrt{S_x^2 + S_y^2} \quad (2.8)$$

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad S_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

รูปที่ 2.16 แสดงโซเบลฟิลเตอร์

### 2.9.1.2 การตรวจหาขอบภาพแบบพรีวิต

การตรวจหาขอบภาพแบบพรีวิต (Gonzalez, 2001) จะคล้ายกับวิธีการแบบโซเบลโดยใช้ขนาดของฟิลเตอร์ 3X3 จำนวนสองฟิลเตอร์ด้วยกัน และทำการเปลี่ยนค่าตัวเลขในฟิลเตอร์ซึ่งแสดงรูปของฟิลเตอร์ดังรูปที่ 2.17 และสมการที่ 2.9

$$h_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad h_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

รูปที่ 2.17 แสดงพรีวิตฟิลเตอร์

$$M = \sqrt{h_x^2 + h_y^2} \quad (2.9)$$

### 2.9.1.3 การตรวจหาขอบภาพแบบโรเบิร์ต

สำหรับการตรวจหาขอบภาพแบบโรเบิร์ตนั้นจะใช้ฟิลเตอร์ขนาด 2X2 จำนวน 2 ฟิลเตอร์ โดยเป็นการหาความแตกต่างของความเข้มของสีแต่ละพิกเซลภาพในภาพ ในแนวเส้นทแยงมุม ซึ่งแสดงรูปของฟิลเตอร์ดังรูปที่ 2.18 และสมการที่ 2.10 (Gonzalez, 2001)

$$G_x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad G_y = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

รูปที่ 2.18 แสดงโรเบิร์ตฟิลเตอร์

$$G(F(i,j)) = |f(i+1,j+1) - f(i,j)| + |f(i+1,j) - f(i,j+1)| \quad (2.10)$$

### 2.9.1.4 การตรวจหาขอบภาพแบบแคนนี่

ในขั้นตอนแรกของการตรวจหาขอบภาพแบบแคนนี่นั้นจะต้องทำการกำจัดสัญญาณรบกวน (Noise) ออกไปก่อน โดยใช้ตัวกรองเกาส์เซียน (Gaussian Filter) เพื่อปรับให้ภาพมีความเรียบขึ้น ซึ่งสัญญาณรบกวนนั้น ก็คือเส้นเล็ก ๆ ที่เกิดขึ้นหลังจากการตรวจหาขอบภาพ ซึ่งถ้าหากเราใช้ตัวกรองเกาส์เซียนเพื่อทำให้ภาพเบลอในระดับหนึ่งแล้วนั้น เส้นเล็ก ๆ หรือสัญญาณรบกวนเหล่านี้ จะได้ถูกกำจัดออกไปได้จำนวนหนึ่ง สำหรับการคำนวณหาภาพที่ได้จากการใช้ตัวกรองเกาส์เซียน เป็นดังสมการที่ 2.11 (Gonzalez, 2001)

$$S[i, j] = G[i, j, \sigma] * I[i, j] \quad (2.11)$$

เมื่อ  $S[i, j]$  คือ ภาพภายหลังการกรอง  
 $I[i, j]$  คือ ภาพที่ต้องการหาขอบ  
 $G[i, j]$  คือ ตัวกรองแบบเกาส์เซียน  
 $\sigma$  คือ ค่าเบี่ยงเบนมาตรฐานของการกระจาย

หลังจากทำการปรับภาพให้เรียบเพื่อกำจัดสัญญาณรบกวนแล้ว ขั้นตอนต่อไปคือการคำนวณหาทิศทางที่เป็นขอบภาพ โดยนำภาพ  $S[i, j]$  มาทำการสร้าง  $P[i, j]$  และ  $Q[i, j]$  ดังสมการที่ 2.12 และ 2.13

$$P[i, j] \approx (S[i, j + 1] - S[i, j] + S[i + 1, j + 1] - S[i + 1, j])/2 \quad (2.12)$$

$$Q[i, j] \approx (S[i, j] - S[i + 1, j] + S[i, j + 1] - S[i + 1, j + 1])/2 \quad (2.13)$$

เมื่อ  $P[i, j]$  คือ ค่าความแตกต่างในแกนแนวนอน  
 $Q[i, j]$  คือ ค่าความแตกต่างในแกนแนวตั้ง  
 $S[i, j]$  คือ ค่าความเข้มแสงของจุดภาพ

และหลังจากนั้นจะแปลงรูปแบบจาก Rectangular ไปเป็น Polar เพื่อทำการหาขนาดของการเปลี่ยนความเข้มของสี ดังสมการที่ 2.14

$$M[i, j] = \sqrt{P[i, j]^2 + Q[i, j]^2} \quad (2.14)$$

### 2.9.2 การทำภาพให้บางลง

การทำขอบภาพให้บางลงนั้น เป็นกระบวนการจัดเตรียม (Pre-processing) ที่สำคัญอย่างหนึ่ง ที่นิยมในการปรับรูปภาพที่ได้ทำการหาขอบภาพแล้ว นำมาทำให้ขอบภาพที่มีความหนาบางลง เป็นการกำจัดขอบของพิกเซลที่หนาเกินไป ซึ่งปกติแล้วการทำให้ขอบภาพบางลง นิยมใช้ในการรู้จำตัวอักษร และการรู้จำลายมือชื่อ ซึ่งจะช่วยให้เหลือเส้นขอบภาพบางที่สุดเพียงแค่นึงพิกเซลเท่านั้น

การทำโอเพอเรชันการปรับขอบภาพให้บาง สามารถทำได้โดยอาศัยฟิลเตอร์ P1 และ P2 วิธีการทำจะมี 2 ขั้นตอน คือขั้นแรกจะใช้ฟิลเตอร์ P1 โดยการนำเทมเพลต (Template) ขนาด 3X3 สแกนไปตามข้อมูลภาพ และทำการพิจารณาพิกเซลบริเวณขอบภาพว่าสามารถลบได้หรือไม่ ถ้าลบได้ให้หมายเหตุไว้แต่ยังไม่ต้องลบ หลังจากทีสแกนทั่วทั้งภาพก็ให้ทำการลบข้อมูลภาพดังที่ได้หมายเหตุไว้ ขั้นที่สองใช้ฟิลเตอร์ P2 และดำเนินการเหมือนการใช้ฟิลเตอร์ P1 เมื่อทำการลบข้อมูลภาพที่มีไว้ในหมายเหตุแล้ว ก็ให้ทำซ้ำต่อไปเรื่อย ๆ จนไม่สามารถลบข้อมูลภาพออกได้อีก (Gonzalez, 2001)

ลักษณะของข้อมูลที่นำมาพิจารณาจะกระทำไปทั้งภาพโดยใช้ฟิลเตอร์ขนาดเท่ากับ 3x3 เลื่อนทำการพิจารณาตามเงื่อนไขที่กำหนดไปจนหมดทั้งภาพ ดังรูปที่ 2.19 โดยที่พิกเซลปัจจุบันคือพิกเซลตรงกลาง (p0)

p8	p1	p2
p7	<b>p0</b>	p3
p6	p5	p4

รูปที่ 2.19 แสดงทีนนิ่งฟิลเตอร์

สำหรับฟิลเตอร์ P1 และ P2 จะมีเงื่อนไขในการพิจารณา ดังสมการที่ 2.15 และ สมการที่ 2.16

$$P1: (2 \leq N(p_0) \leq 6) \& \& (T(p_0) = 1) \& \& (p1.p3.p5 = 0) \& \& (p3.p5.p7 = 0) \quad (2.15)$$

$$P2: (2 \leq N(p_0) \leq 6) \& \& (T(p_0) = 1) \& \& (p1.p3.p7 = 0) \& \& (p1.p5.p7 = 0) \quad (2.16)$$

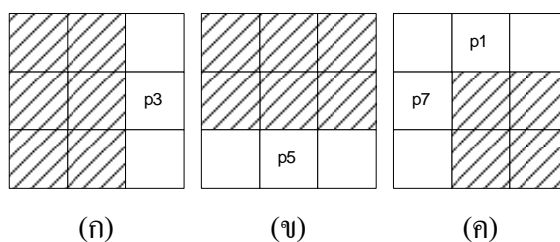
เมื่อ กำหนดให้  $N(p_0) = \sum_{i=1}^8 p_i$  เป็นจำนวนของพิกเซลรอบ  $p_0$

เมื่อ  $p_0 = 0,1$  และ  $i=0,1,2,\dots,8$

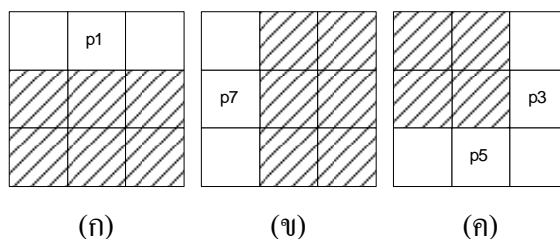
$T(p_0)$  = การเปลี่ยนแปลงของข้อมูลจาก 0 ไปเป็น 1

เมื่อพิจารณาข้อมูลใน  $p_1, p_2, p_3, \dots, p_8, p_1$  ตามลำดับ

สำหรับเงื่อนไขแรกของ P1 และ P2 คือพิกเซลตรงกลาง ( $p_0$ ) สามารถลบได้ก็ต่อเมื่อค่าของ  $p_1$  ถึง  $p_8$  มีค่าเท่ากับ 1 (เป็นสีดำ) มากกว่า 1 ตำแหน่งและต้องไม่มากกว่า 6 ตำแหน่ง สำหรับเงื่อนไขที่ 2 กำหนดให้  $T(p_0) = 1$  หมายถึงต้องมีการเปลี่ยนแปลงของข้อมูลจาก 0 ไปเป็น 1 ของข้อมูล  $p_1, p_2, p_3, \dots, p_8, p_1$  เพียงครั้งเดียวเท่านั้น และในเงื่อนไขที่ 3 และ 4 คือ  $(p1.p3.p5 = 0)$  และ  $(p3.p5.p7 = 0)$  จะใช้ได้สำหรับเงื่อนไขที่  $p3=0$  หรือ  $p5=0$  หรือ ถ้า  $p1=0$  และ  $p7=0$  ตามตัวอย่างดังกล่าวนี้แสดงไว้ดังรูปที่ 2.20 ซึ่งจะเห็นว่าเมื่อ  $p3=0$  จะเป็นลักษณะของขอบด้านขวา เมื่อ  $p5=0$  จะเป็นขอบด้านล่าง และเมื่อ  $p1=0$  และ  $p7=0$  จะเป็นมุมบนซ้าย สำหรับใน P2 นั้นเราสามารถลบพิกเซล  $p_0$  ได้ก็ต่อเมื่อ  $p1=0$  หรือ  $p7=0$  หรือ  $(p3$  และ  $p5=0)$  ดังแสดงในรูปที่ 2.21 ซึ่งจะเห็นว่าเมื่อ  $p1=0$  จะเป็นขอบบน  $p7=0$  จะเป็นขอบด้านซ้าย และเมื่อ  $(p3=0$  และ  $p5=0)$  จะเป็นมุมล่างขวา



รูปที่ 2.20 เงื่อนไขของฟิลเตอร์ P1 เมื่อตำแหน่ง (ก)  $p3=0$  (ข)  $p5=0$  และ (ค)  $p1=0$  และ  $p7=0$



รูปที่ 2.21 เงื่อนไขของฟิลเตอร์ P2 เมื่อตำแหน่ง (ก)  $p1=0$  (ข)  $p7=0$  และ (ค)  $p3=0$  และ  $p5=0$

## 2.10 โปรแกรม MATLAB

MATLAB เป็นซอฟต์แวร์ที่ได้รับความนิยมอย่างแพร่หลายในหมู่นักวิทยาศาสตร์ นักคณิตศาสตร์ และวิศวกร ซึ่งตัวโปรแกรมนั้นมีความสามารถที่กว้างขวาง ครอบคลุมตั้งแต่ การพัฒนาอัลกอริทึม การสร้างแบบจำลองทางคณิตศาสตร์ การสร้างระบบควบคุม และ กระบวนการปรับแต่งรูปภาพ

ด้านการติดต่อกับผู้ใช้งานนั้น สามารถทำงานได้ทั้งในลักษณะของการติดต่อโดยตรง คือ การเขียนคำสั่งเข้าไปทีละคำสั่ง เพื่อให้ MATLAB ประมวลผลไปเรื่อย ๆ หรือสามารถที่จะรวบรวม ชุดคำสั่งเหล่านั้นเป็นโปรแกรมก็ได้ ข้อสำคัญอย่างหนึ่งของ MATLAB ก็คือข้อมูลทุกตัวจะถูกเก็บ ในลักษณะของแถวลำดับ คือในแต่ละตัวแปรจะได้รับการแบ่งเป็นส่วนย่อยเล็ก ๆ ขึ้น ซึ่งการใช้ตัวแปรเป็นแถวลำดับ ในการเขียน โปรแกรมในภาษาขั้นต่ำทั่วไปนั้นทำการต้องจองมิติ ในขณะที่ MATLAB นั้นไม่จำเป็น ซึ่งทำให้เราสามารถที่จะแก้ปัญหของตัวแปรที่อยู่ในลักษณะของเมทริกซ์ และเวกเตอร์ได้โดยง่าย ซึ่งทำให้เราลดเวลาการทำงานลงได้อย่างมากเมื่อเทียบกับการเขียน โปรแกรมโดยภาษาซีหรือภาษาฟอร์แทรน

MATLAB เป็นภาษาคอมพิวเตอร์ระดับสูงที่ใช้สำหรับคำนวณเชิงตัวเลข (Numerical Computing) แสดงผลกราฟฟิก และเขียนแอปพลิเคชัน ภายในตัวโปรแกรม MATLAB นั้น ประกอบด้วยกล่องเครื่องมือ (Toolbox) ที่มีการจัดกลุ่มเป็นฟังก์ชันสำเร็จรูปตามแต่ละสาขาวิชา รวมไปถึงฟังก์ชันพื้นฐานจำนวนมาก ทำให้การวิเคราะห์ทำได้หลากหลายวิธี พร้อมกับคำตอบที่รวดเร็วกว่าโปรแกรมตารางคำนวณ (อย่างเช่น Spreadsheet หรือ Excel) เราสามารถนำ MATLAB ไปประยุกต์ใช้งานได้หลายสาขามาก ทั้งการประมวลผลสัญญาณ (Signal Processing) การสื่อสาร (Communication) การประมวลผลภาพและวิดีโอ (Image and Video Processing) ระบบควบคุม (Control System) ระบบการวัดควบคุม (Instruments) การคำนวณทางเศรษฐศาสตร์ (Economic) การคำนวณทางชีววิทยา (Biology) และอื่น ๆ

### ข้อดีของ MATLAB

- 1) เนื่องจากเป็น โปรแกรมที่ได้รับความนิยม จึงมีตัวอย่างและกรณีศึกษา รวมถึงการ แสดงความคิดเห็นเป็นกันอย่างแพร่หลาย ทำให้มีแหล่งที่ให้ผู้ใช้งาน โปรแกรม MATLAB ศึกษา หรือนำมาประกอบงานวิจัยได้เป็นจำนวนมาก
- 2) โปรแกรมใช้งานง่าย ทำงานได้หลากหลายรูปแบบด้วยสภาพแวดล้อมเชิงโต้ตอบ คล้ายเครื่องคิดเลขสามารถตรวจสอบค่าต่าง ๆ ซ้ำได้อย่างรวดเร็ว ประยุกต์ใช้ในการออกแบบและ แก้ปัญหาทางคณิตศาสตร์ได้ง่ายและมีประสิทธิภาพสูงมาก อีกทั้งเครื่องมือสำหรับสร้างแอปพลิเคชันบน MATLAB นั้น สามารถสร้างด้วย GUI (Graphic User Interface) ซึ่งจะมีลักษณะคล้าย กับ Visual Basic ซึ่งได้รับความนิยมอย่างแพร่หลายมากในอดีต
- 3) มี Toolbox และ ฟังก์ชันทางคณิตศาสตร์มากมาย สำหรับ พีชคณิต สถิติ การวิเคราะห์ ฟูเรีย การประมวลผลภาพและวิดีโอ การคำนวณทางเศรษฐศาสตร์และชีววิทยา เป็นต้น
- 4) ประกอบด้วยฟังก์ชันสำหรับแสดงผลรูปภาพขั้นสูง ในการแสดงผลข้อมูลที่หลากหลายได้อย่างสวยงาม นอกจากนี้ยังสามารถปรับแต่งได้ง่าย
- 5) MATLAB สามารถเชื่อมการทำงานร่วมกับภาษาอื่น ๆ ได้

### ข้อเสียของ MATLAB

- 1) ตัวโปรแกรมต้องการเครื่องคอมพิวเตอร์ที่มีประสิทธิภาพสูง
- 2) โปรแกรมมีส่วนประกอบเยอะ ดังนั้นฟังก์ชันที่มาพร้อมกับ MATLAB จึงมากมาย บางครั้งอาจทำให้หาฟังก์ชันที่ต้องการลำบาก
- 3) ตัวโปรแกรมมีราคาสูง

## 2.11 ภาษาไพธอน และเครื่องมือในการพัฒนาโปรแกรมสไปเดอร์ (Python Language and Spyder IDE)

ภาษาไพธอนเป็นภาษาทางการเขียน โปรแกรมคอมพิวเตอร์ ที่มีลักษณะเป็นภาษา สคริปต์ ซึ่งสามารถทำงานได้ทั้งรูปแบบโครงสร้างแบบเก่าและการเขียนเชิงวัตถุ (Dynamic Object Oriented) และ ยังสามารถทำงานได้ทั้งแบบบนคอมพิวเตอร์เดี่ยว (Standalone Computer) และ บนเครือข่ายอินเทอร์เน็ต รวมไปถึง โปรแกรมประยุกต์บนเว็บ อีกทั้งสามารถที่จะทำงานได้หลาย ระบบปฏิบัติการ เช่น Linux, Windows, FreeBSD รวมไปถึง Mac/OS จุดเด่นที่สำคัญอีกอย่างหนึ่งของภาษาไพธอน คือ เป็น Open Source ทำให้บริษัทผู้ผลิตซอฟต์แวร์ชั้นนำทั่วโลก นำเอาภาษา ไพธอนไปพัฒนาเป็นโปรแกรมต่าง ๆ ให้ได้ใช้กันทั่วโลก เช่น [www.google.com](http://www.google.com) ซึ่งเป็นเว็บ



สำหรับสืบค้นข้อมูลที่ได้รับความนิยมสูงสุด ของบริษัท Google นอกจากนี้องค์การ NASA ได้ประกาศให้ภาษาไพธอนเป็นภาษาที่ใช้ในการพัฒนาซอฟต์แวร์ประเภท CAD/CAE/PDM เป็นหลักด้วย ภาษาไพธอน ถูกสร้างขึ้นโดยชาวเนเธอร์แลนด์ชื่อ นาย กีโด ฟาน รอสซัม (Guido Van Rossum) ลักษณะโดยทั่วไปของภาษาไพธอนนั้นจะคล้ายกับภาษา Perl, PHP, JAVA และ ASP เนื่องจากภาษาไพธอน เป็นภาษาที่ใหม่จึงมีคุณสมบัติที่ดีดังต่อไปนี้

1) สามารถใช้ได้ทุกแพลตฟอร์ม กล่าวคือ ทำงานได้กับคอมพิวเตอร์ทุกประเภท หลากหลายระบบปฏิบัติการ และหลากหลายแพลตฟอร์ม ตัวอย่างเช่น ไพธอนบนแพลตฟอร์มภาษาซี ไพธอนบนแพลตฟอร์มภาษาจาวา และไพธอนบนแพลตฟอร์มภาษาคอทเน็ต

2) เป็น Open Source ทำให้ไม่ต้องเสียค่าใช้จ่ายในการจัดซื้อตัวโปรแกรมมาติดตั้ง โปรแกรมภาษาไพธอนสามารถดาวน์โหลดจาก [www.python.org](http://www.python.org) ได้โดยตรง แล้วนำมาติดตั้งและศึกษาการใช้งานได้ด้วยตนเอง นอกจากนี้การเป็น Open Source ทำให้กลุ่มนักพัฒนาโปรแกรมเข้าถึงง่าย มีตัวอย่างให้ค้นหาหลากหลาย และทำให้สามารถนำความรู้มาต่อยอดเพื่อพัฒนางานวิจัยของแต่ละบุคคล

3) ภาษาไพธอนมีโครงสร้างภาษาที่ไม่ซับซ้อน ทำให้นักพัฒนาโปรแกรมเรียนรู้ได้เร็วกว่าโปรแกรมภาษาอื่น ๆ ในส่วนของโครงสร้างภาษาจะคล้ายคลึงกับภาษาซี แต่ภาษาไพธอนจะมีความกระชับกว่าภาษาซี

4) มีความปลอดภัยสูง เนื่องจากภาษาไพธอนทำงานอยู่ด้าน Server เป็นหลัก เมื่อมีการร้องขอจากเครื่อง Client จะประมวลผลที่เครื่อง Server ทำให้ผู้ใช้ทั่วไปไม่สามารถเข้าถึงเครื่อง Server ได้โดยตรง จึงมีความปลอดภัยสูงกว่า

5) มีความเป็น Glue Language หรือภาษาสำหรับการเชื่อมต่อ กล่าวคือในปัจจุบันการพัฒนาซอฟต์แวร์ได้เน้นที่มีการแลกเปลี่ยนข้อมูลซึ่งกันและกัน ทั้งในองค์กรเดียวกันหรือแม้แต่ต่างองค์กรกัน ภาษาไพธอนสามารถเรียกใช้ภาษาโปรแกรมอื่น ๆ ได้หลายภาษา ทำให้เหมาะที่จะใช้เขียนเพื่อประสานงานโปรแกรมที่เขียนในภาษาต่างกันได้ ทำให้เกิดความสะดวกไม่ต้องใช้ซอฟต์แวร์อื่น ๆ มาแปลงข้อมูลเพื่อให้เข้ากันได้อีกต่อไป

ในการพัฒนาโปรแกรมไพธอนของวิทยานิพนธ์นี้จะใช้สภาพแวดล้อมในการพัฒนา (Integrated Development Environment : IDE) ด้วยซอฟต์แวร์สไปเดอร์ ซึ่งเป็น Editor ชนิดหนึ่งที่ใช้สำหรับการพัฒนาโปรแกรมภาษาไพธอน สามารถทำงานได้บนหลายแพลตฟอร์ม เช่น Windows, MAC และ Linux ซึ่งในความจริงแล้วผู้สร้างสไปเดอร์ขึ้นมานั้นต้องการให้มันทำงานคล้ายกับโปรแกรม MATLAB ดังนั้นเราจึงเห็นตัวโปรแกรมสไปเดอร์มี GUI ที่คล้ายกับโปรแกรม MATLAB ข้อดีของโปรแกรมสไปเดอร์นั้นอยู่ที่มันเป็นโปรแกรมที่มีขนาดเล็ก และ ใช้ทรัพยากรของ

เครื่องคอมพิวเตอร์น้อยกว่าโปรแกรมอื่น อีกด้วย นอกจากนั้นยังเป็นโปรแกรมที่เป็น Open Source ซึ่งทำให้นักวิจัยหลายคนเลือกใช้เพราะ สามารถเข้าถึงได้ง่าย และมีการนำความรู้มาอภิปราย และ แสดงข้อคิดเห็นกันในวงกว้างในอินเทอร์เน็ต

## 2.12 งานวิจัยที่เกี่ยวข้อง (Literature Review)

ในปัจจุบันงานวิจัยที่เกี่ยวข้องกับเรื่องการรู้จำลายมือชื่อนั้นมีมากมาย ซึ่งทางผู้วิจัยได้ ทำการศึกษา และคัดเลือกนำมาประกอบงานวิจัย 6 เรื่องดังต่อไปนี้

Vargas et al. (2014) ได้อธิบายการตรวจสอบลายมือชื่อที่เขียนด้วยลายมือ ในระดับ ภาพรวม โดยมุ่งเน้นที่การวัดระดับ Grey Scale ซึ่งมีทั้งการใช้ คุณสมบัติของพื้นผิวด้านสถิติ (Statistical Texture Features) เทคนิค Co-Occurrence Matrix และ Local Binary Pattern มาใช้ใน งานวิจัย ซึ่งงานวิจัยนี้จะเริ่มโดยการลบภาพพื้นหลังออกก่อน จึงทำให้สามารถนำข้อมูลที่ทางผู้วิจัย ดังกล่าวสนใจ มาวิเคราะห์ข้อมูล และ นำไปใช้ใน โมเดลได้อย่างถูกต้องแม่นยำ อีกทั้งในงานวิจัย ชื่อนี้มีการใช้กราฟิสโตแกรม เพื่อลดทอนอิทธิพลของการใช้หมึกปากกาต่างกัน ในผู้ลงนามคน เดียวกันอีกด้วย แหล่งอ้างอิงภาพลายมือชื่อของงานวิจัยชื่อนี้ได้มาจากสองฐานข้อมูลคือ MCYT-75 และ GPDS-100 ซึ่งจะมีทั้งลายมือชื่อของแท้ (Genuine) และ ลายมือชื่อจากการปลอมแปลง (Forgeries) ด้านผลลัพธ์ที่ออกมานั้น ได้รับค่า EER (Equal Error Rate) = 16.27% เมื่อลายมือชื่อที่ มาจากการปลอมแปลงนั้นมาจากผู้เชี่ยวชาญ (Skilled Forgeries) ซึ่งหากนำลายมือชื่อปลอมแปลงแบบ ที่มาจากการสุ่ม (Random Forgeries) มาเข้าโมเดลแล้วนั้น จะได้ค่า EER = 6.7% ซึ่งทางผู้วิจัยได้มี การใช้ Local Binary Pattern เข้าช่วยเพื่อเพิ่มความแม่นยำในการตรวจสอบลายมือชื่อ โดยผลลัพธ์ที่ ได้นั้นออกมามีค่าขึ้น โดยเมื่อตรวจสอบกับ Skilled Forgeries จะมีค่า EER = 12.8% และ EER = 3.6% ใน Random Forgeries เมื่อมี 5 Samples

Guerbai et al. (2014) นำเสนอการใช้งานที่มีประสิทธิภาพของ OC-SVM เพื่อตรวจสอบ ลายมือชื่อที่เขียนด้วยลายมือ ตามพารามิเตอร์ของผู้ลงนาม (Writer) โดยใช้ลายมือชื่อของแท้เท่านั้น ประโยชน์หลักของการใช้ HSVS (Handwritten Signature Verification System) คือ จะใช้ลายมือชื่อ เพียงแค่จำนวนน้อย มันจะยอมให้มีการกำหนด Optimal Threshold จากลายมือชื่อแท้ (Genuine Signatures) และ ลายมือชื่อที่เกิดจากการสมมติขึ้น (Fictitious Signatures) จาก Combination of Distance ซึ่งต้องผ่านการปรับค่าอย่างระมัดระวัง และเมื่อมีผู้ลงนามคนใหม่เข้ามาในระบบ ค่าพารามิเตอร์ของ OC-SVM จะถูกใช้โดยปราศจากการ หาค่า Optimal Threshold ด้านผลลัพธ์จาก การทดลองนั้น เราจะเห็นค่า AER (Average Error Rate) อยู่ที่ 5 – 7% ใน CEDAR Dataset ส่วนใน

GPDS Dataset จะมีค่า AER อยู่ที่ 15 – 17% โดยทั้งสอง Dataset จะมีค่า AER ลดลงเรื่อย ๆ ตามจำนวนของผู้ลงนาม ในการทดลองนี้จำนวนจะอยู่ที่ 4 – 12 ลายมือชื่อ

Frias-Martinez et al. (2006) กล่าวถึงปัญหาของการรู้จำลายมือชื่อโดยอัตโนมัติ (Signature Recognition) ได้รับความสนใจเพียงเล็กน้อยเมื่อเทียบกับปัญหาของการตรวจสอบลายมือชื่อ (Signature Verification) แม้จะมีการใช้งานที่มีศักยภาพในการเข้าถึงสิ่งอำนวยความสะดวก การรักษาความปลอดภัยที่มีความสำคัญ รวมไปถึงการประมวลผลเอกสารทางกฎหมาย และทางประวัติศาสตร์ งานวิจัยชิ้นนี้ได้นำเสนอการรู้จำลายมือชื่อของมนุษย์ บนพื้นฐานของ Support Vector Machine (SVM) และ นำไปเปรียบเทียบกับ เทคนิคการแบ่งกลุ่มแบบดั้งเดิม อย่าง Multi-Layer Perceptron (MLP) งานวิจัยชิ้นนี้ยังแสดงถึงกลไก (Mechanism) การตรวจสอบถึง Variation ของแต่ละลายมือชื่อในตัวบุคคลเดียวกัน โดยนำเสนอผ่านสองกระบวนการ คือ 1) การสร้าง Feature Vector โดยใช้เซตของ Global Geometric และ Moment-Based Characteristics ของลายมือชื่อ 2) การสร้าง Feature Vector โดยใช้ Bitmap ของ ลายมือชื่อที่ตรงกัน (Corresponding) ซึ่งจากผลการทดลองแสดงให้เห็นว่า SVM นั้นให้ค่าความแม่นยำสูงถึง 71% และ ดีกว่าเทคนิค MLP

Suman and Kumar (2013) นำเสนอเกี่ยวกับวิธีการระบุตัวตนด้วยลายมือชื่อแบบออฟไลน์ ซึ่งเป็นที่สนใจมากขึ้นในปีที่ผ่านมา เพราะมีความสามารถนำไปใช้ในชีวิตประจำวันในเชิงปฏิบัติ และลายมือชื่อจะต้องมีการตรวจสอบทันที เช่น งานรักษาความปลอดภัย หรือ สำหรับการทำธุรกรรมเชิงพาณิชย์ แม้กระทั่งการตรวจสอบการเข้าร่วมประชุมอื่น ๆ โดยตัวงานวิจัยชิ้นนี้ จะวิจัยเกี่ยวกับ การระบุลายมือชื่อในภาษาปัญหา บนพื้นฐานของเทคนิค Neural Network ข้อได้เปรียบหลักสำหรับการใช้การระบุลายมือชื่อออฟไลน์ คือเป็นระบบที่มีการปรับตัวได้ดี และในการดำเนินการนั้นสามารถกระทำได้อย่างง่ายดาย ใช้ต้นทุนต่ำ รวมไปถึงมีความสะดวกของการฝังระบบในองค์กร โดยไม่ต้องมีผลกระทบต่อระบบที่มีอยู่ ผลการทดลองนี้ วัดค่าความแม่นยำโดยใช้ค่า FAR (False Acceptance Ratio) และ FFR (False Rejection Ratio) ซึ่งค่าทั้งสองจะมากหรือน้อยนั้นขึ้นอยู่กับค่าการปรับแต่งค่า Threshold ซึ่งถ้าปรับค่า Threshold ให้สูง FFR จะลดลง และ FAR สูงขึ้น (Over Accept) โดยในงานวิจัยที่นำเสนอนี้ได้ใช้ค่า Threshold ที่ 90%

Zheng et al. (2004) ทำการตรวจหาขอบร่วมกับการไล่ระดับสี ซึ่งเป็นนวัตกรรมใหม่ เพื่อที่จะหาตำแหน่งขอบให้ชัดเจนมากยิ่งขึ้น บทความนี้ใช้ Least Squares Support Vector Machine (LS-SVM) ร่วมกับ เคอร์เนลฟังก์ชันเรเดียลเบสิส และใช้วิธีการตรวจหาขอบแบบ Sobel และ Canny พบว่าทำให้มีประสิทธิภาพเพิ่มขึ้น

Niu et al. (2012) ได้นำเสนอเทคนิคใหม่ในการรู้จำลายมือตัวเลข โดยใช้อัลกอริทึมตัวจำแนก 2 ชนิดเข้าร่วมกัน นั่นคือ CNN (Convolution Neural Network) เพื่อใช้เป็นตัว Feature

Extractor และ SVM เพื่อเป็นตัว Recognizer ด้วยทั้งสองเทคนิคที่ผสมผสานกันนี้ทำให้ได้ค่า Recognition Rate สูงถึง 99.81% และ เมื่อทางผู้วิจัยได้ทดลองทำการ Reject ภาพที่คลุมเครือออกไป (ทำการเลือกที่จะไม่ทำนายลายมือที่มีความคลุมเครือ) ได้ผลลัพธ์คือ Recognition Rate ที่ลดลงเหลือ 94.4% โดยที่มี Reject Rate อยู่ที่ 5.6% นั้นหมายความว่าในผลงานวิจัยชิ้นนี้ให้ค่าความแม่นยำที่น่าเชื่อถือ (Reliability Rate) ที่ 100% ซึ่งนับว่ามีความสำคัญมากในการนำไปใช้ในแวดวงอุตสาหกรรม และการเงิน

Mangini et al. (2014) นำเสนอเกี่ยวกับการใช้เทคนิคแบบใหม่ในการสร้าง Prototype ในการรู้จำลายมือตัวเลข โดยงานวิจัยชิ้นนี้จะมีเป้าหมายทั้งในแง่ของความแม่นยำในการทำนาย และ เวลาที่ใช้ในการจำแนกลายมือตัวเลข การดำเนินงานวิจัยจะแบ่งออกเป็นสองขั้นตอนหลัก คือ ขั้นตอนการคัดเลือก Prototype ซึ่งในขั้นตอนนี้จะใช้อัลกอริทึมที่เรียกว่า ART1 (Adaptive Resonance Theory 1) และ ขั้นตอนที่สองคือ การกรอง Prototype ซึ่งวัตถุประสงค์ของขั้นตอนนี้จะเน้นไปที่การตัด Prototype ที่มีความแม่นยำในการทำนายต่ำทิ้ง หลังจากได้ชุดของ Prototype เรียบร้อยแล้ว ข้อมูลจะถูกนำไปเข้าอัลกอริทึม k-NN เพื่อสร้างโมเดลการรู้จำ และผลลัพธ์ของงานวิจัยนี้ออกมาค่อนข้างน่าพอใจ โดยที่ความแม่นยำในการทำนายอยู่ที่ 98.73% ซึ่งนับว่าค่อนข้างต่ำ เมื่อเทียบกับงานวิจัยชิ้นอื่น ๆ ในด้านการรู้จำลายมือตัวเลข แต่เวลาในการทำนายนั้นลดลงอย่างมากจาก 511.97 วินาที เหลือ 6.15 วินาที

จากงานวิจัยที่เกี่ยวข้องที่ได้กล่าวมาข้างต้น มีทั้งที่อยู่ในลักษณะของการตรวจสอบลายมือ และการรู้จำลายมือ ในส่วนของฐานข้อมูลที่น่ามาใช้ นั้น มีทั้งที่นำมาจากฐานข้อมูลที่มีอยู่แล้ว และ เก็บรวบรวมข้อมูลเอง โดยขอบเขตของแต่ละงานวิจัยจะมุ่งเน้น ไปที่การทดสอบเปรียบเทียบ ประสิทธิภาพ และการปรับปรุงเพื่อเพิ่มประสิทธิภาพ ซึ่งจะมีเพียงส่วนน้อยที่จะทำการนำเสนอ แนวคิดใหม่ลงไปด้วย โดยเทคนิคที่นิยมนำมาใช้ในงานวิจัยส่วนมากจะเป็น SVM การพัฒนา งานวิจัยด้านการเรียนรู้ลายมือชื่อของวิทยานิพนธ์นี้ เมื่อเปรียบเทียบกับงานวิจัยอื่นสามารถสรุปได้ ดังตารางที่ 2.3

ตารางที่ 2.3 สรุปเปรียบเทียบงานวิจัยที่เกี่ยวข้อง

กระบวนการทำงาน	งานวิจัยที่เกี่ยวข้อง						
	ก	ข	ค	ง	จ	ฉ	ช*
<b>เทคนิค/ อัลกอริทึมที่นำมาใช้</b>							
Support Vector Machine	✓	✓	✓		✓		✓
Perceptron			✓				✓
Neural Network				✓	✓		
Naïve Bayes							✓
k- Nearest Neighbors		✓				✓	✓
Local Binary Pattern	✓	✓					
Grey Level Co-occurrence Matrix	✓						
Curvelet Transform		✓					
Additional Intensive Data							✓
<b>ขอบเขตงานวิจัย</b>							
เพื่อทดสอบประสิทธิภาพ	✓	✓	✓	✓	✓	✓	✓
เพื่อเปรียบเทียบประสิทธิภาพระหว่างเทคนิค/อัลกอริทึม	✓	✓	✓		✓		✓
เพื่อปรับปรุงหรือเพิ่มประสิทธิภาพ	✓	✓		✓	✓		✓
เพื่อเสนอแนวคิดใหม่	✓				✓	✓	✓
<b>ลักษณะการจำแนกไปโอเมตริกซ์</b>							
การตรวจสอบลายมือ	✓	✓		✓			
การรู้จำลายมือ			✓		✓	✓	✓
<b>Dataset / การเก็บรวบรวมข้อมูล</b>							
นำมาจากแหล่งอ้างอิงที่มีอยู่แล้ว	✓	✓			✓	✓	
เก็บรวบรวมข้อมูลด้วยผู้วิจัย			✓	✓			✓
<b>การพัฒนา</b>							
ภาษาไพธอน							✓
โปรแกรม MATLAB	✓			✓	✓	✓	✓
ภาษาหรือโปรแกรมอื่น ๆ		✓	✓				

**หมายเหตุ** งานวิจัยที่เกี่ยวข้อง ประกอบด้วย

- ก แทนงานวิจัยของ Vargas et al. (2014)  
 ข แทนงานวิจัยของ Guerbai et al. (2014)  
 ค แทนงานวิจัยของ Frias-Martinez et al. (2006)  
 ง แทนงานวิจัยของ Suman and Kumar (2013)

- จ แทนงานวิจัยของ Niu et al. (2012)
- ฉ แทนงานวิจัยของ Mangini et al. (2014)
- ซ\* แทนงานวิจัยการเพิ่มประสิทธิภาพของโมเดลรู้จำภาพลายมือชื่อ  
(งานวิจัยของวิทยานิพนธ์ฉบับนี้)



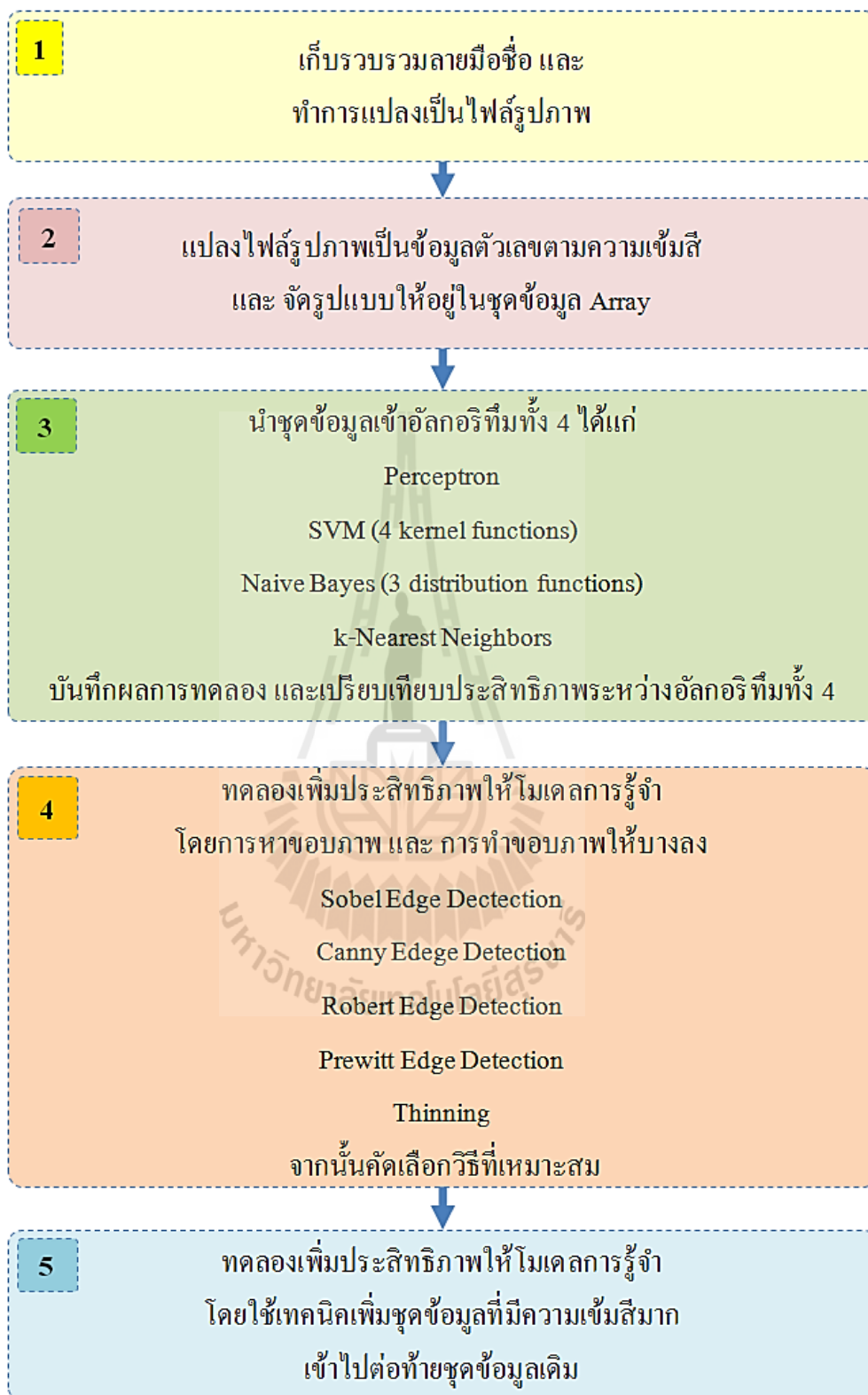
## บทที่ 3

### วิธีดำเนินการวิจัย

งานวิจัยนี้มีวัตถุประสงค์เพื่อทำการศึกษเปรียบเทียบประสิทธิภาพของโมเดลที่ได้จากอัลกอริทึมการเรียนรู้ทั้ง 4 อัลกอริทึม รวมถึงการพัฒนาเพื่อเพิ่มประสิทธิภาพของการรู้จำอีกด้วย โดยในบทนี้จะกล่าวถึง กรอบแนวคิดวิธีการวิจัย เครื่องมือที่ใช้ในการวิจัย และกระบวนการต่าง ๆ ของการวิจัย โดยมีรายละเอียดดังนี้

#### 3.1 กรอบแนวคิดของการวิจัย

แนวคิดหลักของงานวิจัยนี้ ถูกแบ่งออกเป็น 5 ส่วน ได้แก่ 1) วิธีเก็บข้อมูลลายมือชื่อ และการแปลงภาพลายมือชื่อให้เป็นไฟล์ภาพ 2) การแปลงข้อมูลภาพลายมือชื่อให้เป็นข้อมูลทางตัวเลข 3) การทดสอบเปรียบเทียบประสิทธิภาพของแต่ละอัลกอริทึม 4) การทดลองเพิ่มประสิทธิภาพการรู้จำโดยใช้เทคนิคการหาขอบภาพ และการทำขอบภาพให้บางลง 5) การทดลองเพิ่มประสิทธิภาพการรู้จำโดยใช้เทคนิคเพิ่มชุดข้อมูลที่มีความเข้มสูง ดังที่แสดงให้เห็นในรูปที่ 3.1

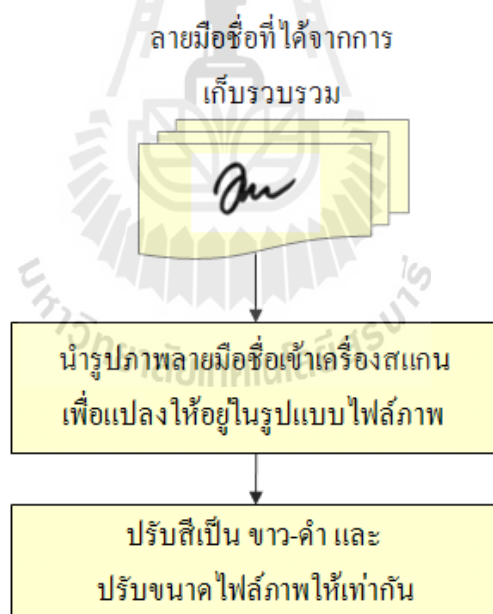


รูปที่ 3.1 แผนภาพแสดงขั้นตอนการดำเนินงานวิจัยทั้ง 5 ขั้นตอน



### 3.1.1 กรอบแนวคิดขั้นตอนที่ 1: วิธีเก็บข้อมูลตัวอย่างลายมือชื่อ และการแปลงภาพลายมือชื่อให้เป็นไฟล์ภาพ

ขั้นตอนนี้จะเกี่ยวกับการเก็บรวบรวมข้อมูล รวมไปถึงการแปลงข้อมูลเพื่อให้ง่ายต่อการทำงานวิจัย โดยทางผู้วิจัยได้ทำการเก็บรวบรวมข้อมูลจริง จากนักศึกษาชั้นปีที่สี่ คณะมนุษยศาสตร์และสังคมศาสตร์ สาขาสารสนเทศศึกษา มหาวิทยาลัยบูรพา โดยทางผู้วิจัยได้ทำการคัดกรองเอาแต่บุคคลที่ใช้ลายมือชื่ออยู่เป็นประจำ ภายหลังจากการคัดกรองดังกล่าว ผู้วิจัยได้ร้องขอให้ผู้ที่เข้าร่วมงานวิจัยทำการเขียนลายมือชื่อคนละ 20 ครั้ง โดยให้เขียนในตอนแรก 10 ครั้งจากนั้นเว้นระยะไป 14 วันแล้วจึงทำการร้องขอให้เขียนลายมือชื่ออีก 10 ครั้ง เพื่อเป็นการยืนยันว่าใช้ลายมือชื่อในชีวิตประจำวันจริง และเป็นการแสดงถึงว่างานวิจัยนี้สามารถรองรับความแปรปรวนของลายมือชื่อตามสภาพแวดล้อม (ซึ่งในที่นี้คือเวลา) จากผู้เข้าร่วม 30 คน ดังนั้นจำนวนลายมือชื่อที่ใช้ในงานวิจัยนี้จะมีทั้งหมด 600 ลายมือชื่อ โดยขั้นตอนการเตรียมข้อมูลได้ถูกอธิบายไว้ในแผนภาพ ดังรูปที่ 3.2



รูปที่ 3.2 วิธีการเก็บข้อมูลลายมือชื่อ และการแปลงข้อมูลให้อยู่ในรูปแบบไฟล์ภาพ

จากกรอบแนวคิดข้างต้นนั้นจะถูกแบ่งออกเป็น 2 ขั้นตอนย่อยดังต่อไปนี้  
 ขั้นตอนที่ 1 นำภาพถ่ายมือชื่อที่ได้ทั้งหมดมาผ่านเครื่องสแกน FUJI XEROX รุ่น ApeosPort-IV5070 เพื่อทำเป็นไฟล์รูปภาพ  
 ขั้นตอนที่ 2 ทำการปรับแต่งสีให้เป็นลักษณะขาวดำ  
 ขั้นตอนที่ 3 ทำการปรับขนาดรูปภาพลายมือชื่อให้มีขนาดเท่ากันทุกภาพ กว้าง 144 พิกเซล และ สูง 38 พิกเซล ดังตัวอย่างในรูปที่ 3.3



รูปที่ 3.3 ตัวอย่างภาพถ่ายลายมือชื่อที่ปรับให้เป็นสีขาว ดำ และทำให้มีขนาดเท่ากัน

### 3.1.2 กรอบแนวคิดขั้นตอนที่ 2: การแปลงข้อมูลภาพถ่ายลายมือชื่อให้เป็นข้อมูลทางตัวเลข

ขั้นตอนที่ 1 ใช้โปรแกรมไพธอน ทำการประมวลผล และแปลงภาพถ่ายลายมือชื่อที่นำเข้ามา ให้เป็นรูปแบบตารางที่มีขนาด 144 คอลัมน์ และ 38 แถว ตามขนาดภาพที่เราได้ปรับแต่งไว้ จากนั้นแทนที่ค่าความเข้มสีลงในทุกช่องของตารางนั้น ค่าความเข้มสีแสดงอยู่ในลักษณะตัวเลข มีค่าระหว่าง 0 ถึง 255 โดยที่ 0 คือ ค่าระดับความเข้มสูงที่สุด (ในที่นี้คือสีดำ) และตัวเลข 255 คือความเข้มต่ำสุด (ในที่นี้คือสีขาว) ดังตัวอย่างในรูปที่ 3.4

ขั้นตอนที่ 2 จัดเรียงข้อมูลตัวเลขของไฟล์ภาพแต่ละลายมือชื่อ ให้อยู่ในรูปของ Array 1 มิติ โดยที่ลายมือชื่อหนึ่งรูปจะมีค่าข้อมูล 5,472 คอลัมน์



รูปที่ 3.4 ตัวอย่างภาพถ่ายมือชื่อที่อยู่ในรูปของตาราง และค่าความเข้มถูกแทนที่ด้วยตัวเลข 0 – 255

The screenshot shows the Spyder Python IDE interface. The main window displays a Python script with the following code:

```
6
7
8
9
10 from PIL import image
11 import numpy as np
12 from sklearn.cross_validation import cross_val_score, Kfold
13 from sklearn.cross_validation import StratifiedFold
14 from scipy.stats import sun
15 from sklearn import svm
16 from sklearn.svm import svm
17 from sklearn import metrics
18
19 for x in range(1,60):
20
21     imgstr=str(x)+'.jpg'
22     original=Image.open(imgstr)
23     converted_img=original.convert('L')
24     ndarr=np.array(converted_img.getdata())
25     if x==1:
26         imgarr = np.array(ndarr)
27     else:
28         X = np.array(ndarr)
29         imgarr = np.concatenate((imgarr,X))
30
31 target = np.array([
```

The Variable explorer on the left shows the state of the program, including variables like 'converted\_img', 'imgstr', 'ndarr', and 'pi'. The console window on the right displays the output of the script, which is a grid of numerical values representing the grayscale image data.

รูปที่ 3.5 ภาพตัวอย่างแสดงการแปลงไฟล์ภาพให้เป็นข้อมูลตัวเลขตามความเข้มสีด้วยไพธอน

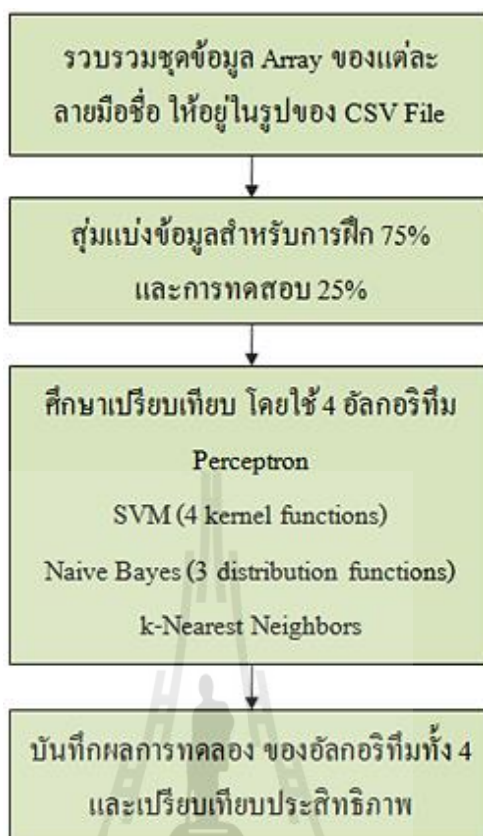


รูปที่ 3.6 ขั้นตอนการแปลงไฟล์รูปภาพให้เป็นข้อมูลตัวเลข

### 3.1.3 กรอบแนวคิดขั้นตอนที่ 3: การทดสอบเปรียบเทียบประสิทธิภาพของแต่ละอัลกอริทึม

ขั้นตอนนี้จะเป็นการนำข้อมูล Array ที่ได้จาก กรอบแนวคิดขั้นตอนที่ 2 มารวมกัน และทำให้อยู่ในรูปของ CSVไฟล์ เพื่อสะดวกต่อการนำไปทำการศึกษาเปรียบเทียบประสิทธิภาพ โดยมีขั้นตอนย่อย ๆ ตามแผนภาพในรูปที่ 3.7





รูปที่ 3.7 การทดสอบเปรียบเทียบประสิทธิภาพของแต่ละอัลกอริทึม

จากกรอบแนวคิดของขั้นตอนนี้จะประกอบด้วย 4 ขั้นตอนย่อยดังต่อไปนี้  
 ขั้นตอนที่ 1 นำข้อมูลซึ่งอยู่ในรูปแบบของ Array มารวบรวม และ Export ออกมา  
 ให้อยู่ในรูปแบบไฟล์ CSV เพื่อความสะดวกในการนำไปใช้งาน

ขั้นตอนที่ 2 แบ่งข้อมูลออกเป็น สองส่วนเพื่อการฝึก (Training data) และการ  
 ทดสอบ (Test data)

ขั้นตอนที่ 3 นำไฟล์ชุดข้อมูลฝึกที่ได้ ไปเข้าอัลกอริทึมทั้ง 4 อัลกอริทึม เพื่อให้ได้  
 โมเดลการเรียนรู้ จากนั้นทดสอบความแม่นยำของโมเดลด้วยชุดข้อมูลทดสอบ

ขั้นตอนที่ 4 บันทึกผลการทดลอง จากนั้นนำผลลัพธ์ที่ได้ของแต่ละอัลกอริทึมมา  
 ทำการศึกษา และเปรียบเทียบประสิทธิภาพ

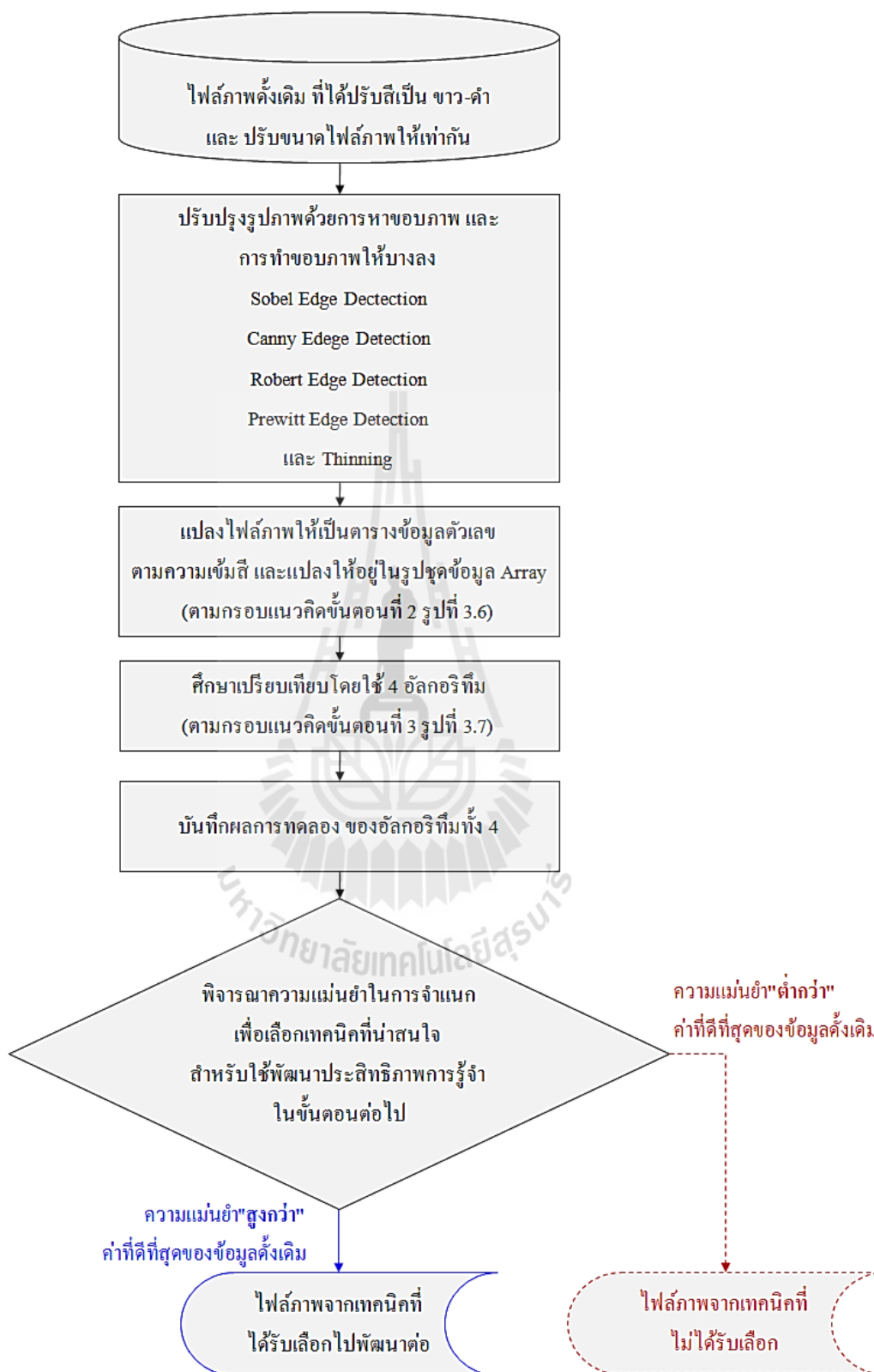
### 3.1.4 กรอบแนวคิดของขั้นตอนที่ 4 : การทดลองเพิ่มประสิทธิภาพการรู้จำโดยใช้เทคนิคการหาขอบภาพ และการทำขอบภาพบาง

การดำเนินงานในขั้นตอนนี้จะนำผลลัพธ์ที่ได้จาก กรอบแนวคิดของขั้นตอนที่ 1 มาทำการปรับปรุงไฟล์ภาพให้มีความคมชัดยิ่งขึ้นด้วยเทคนิคต่าง ๆ รวม 5 เทคนิคด้วยกัน จากนั้นจึงทำการคัดเลือกเทคนิคที่น่าสนใจมาใช้ในการพัฒนาโมเดลการรู้จำต่อไป โดยมีแผนผังแสดงไว้ดังรูปที่ 3.8

จากกรอบแนวคิดของขั้นตอนนี้จะประกอบด้วย 3 ขั้นตอนย่อยดังต่อไปนี้

- ขั้นตอนที่ 1 ปรับปรุงภาพด้วยเทคนิคการหาขอบภาพ และ ทำขอบภาพให้บางลง
- ขั้นตอนที่ 2 นำไฟล์ภาพที่ได้รับการปรับปรุงแล้วมาแปลงให้อยู่ในรูปของตารางตัวเลขตามความเข้มสี และ จากนั้นแปลงข้อมูลตัวเลขให้เป็นชุดข้อมูล Array
- ขั้นตอนที่ 3 ศึกษาเปรียบเทียบโดยใช้อัลกอริทึมทั้ง 4 แบบ
- ขั้นตอนที่ 4 บันทึกผลการทดลอง
- ขั้นตอนที่ 5 คัดเลือกไฟล์รูปภาพที่ได้จากเทคนิคการปรับปรุงรูปภาพ โดยตัดสินใจเลือกจากการนำไปเปรียบเทียบกับค่าที่สูงที่สุดของชุดข้อมูลดั้งเดิม (ผลลัพธ์ที่ดีที่สุดจากกรอบแนวคิดขั้นตอนที่ 3) โดยภาพที่ได้จากเทคนิคเหล่านั้นจะถูกนำไปใช้ในขั้นตอนถัดไป



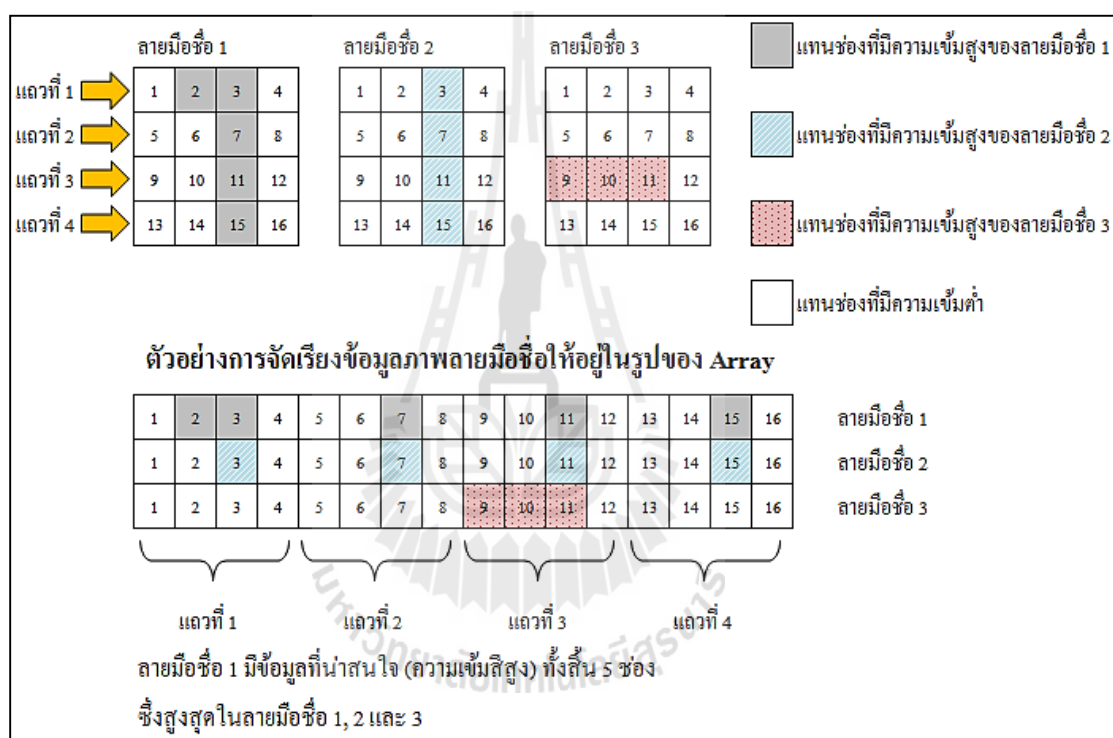


รูปที่ 3.8 ขั้นตอนการเพิ่มประสิทธิภาพการรู้จำด้วยเทคนิคการปรับปรุงรูปภาพ

### 3.1.5 กรอบแนวคิดของขั้นตอนที่ 5 : การเพิ่มประสิทธิภาพการรู้จำด้วยเทคนิคการเพิ่มชุดข้อมูลที่มีความเข้มสูง

การดำเนินงานในขั้นตอนนี้จะเป็นการเพิ่มประสิทธิภาพโมเดลการรู้จำ โดยจะมุ่งเน้นไปที่การนำข้อมูลมาใช้งานให้เกิดประโยชน์สูงสุด ซึ่งข้อมูลที่เราให้ความสนใจนั้นจะเป็นข้อมูลที่มีความเข้มสูงซึ่งนั่นหมายถึงตัวลายมือชื่อนั่นเอง (เมื่อข้อมูลที่มีความเข้มต่ำแสดงถึงที่ว่างระหว่างเส้นของตัวลายมือชื่อ) โดยได้แสดงการเพิ่มข้อมูลที่มีความเข้มสูงไว้ในรูปที่ 3.9 และ

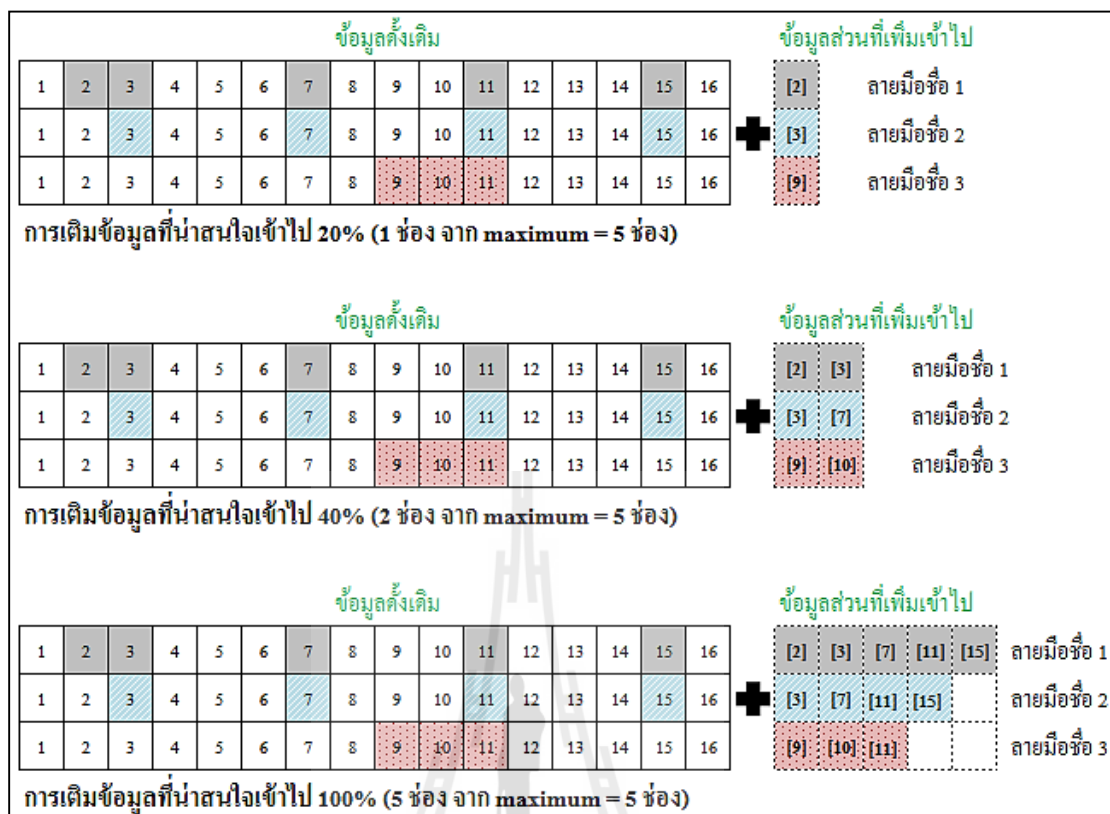
3.10



รูปที่ 3.9 ภาพตัวอย่างแสดงการเพิ่มข้อมูลความเข้มสูงในแต่ละลายมือชื่อ

รูปที่ 3.9 แสดงถึงการแปลงนำภาพลายมือชื่อให้อยู่ในชุดข้อมูล Array โดยตัวอย่างเป็นภาพลายมือชื่อขนาด 4x4 ทั้งสามภาพถูกนำมาจัดเรียงให้อยู่ในรูปของ Array ขนาด 16 คอลัมน์ ซึ่งจากชุดข้อมูลตัวอย่างนี้ จะเห็นได้ว่าลายมือชื่อที่ 1 มีจำนวนข้อมูลที่มีความเข้มสูง (ข้อมูลที่น่าสนใจ) เป็นจำนวนช่องทั้งหมด 5 ช่อง (ลายมือชื่อที่ 2 มี 4 ช่อง และ ลายมือชื่อที่ 3 มี 3 ช่อง) โดยจำนวนช่อง เท่ากับ 5 ช่องในที่นี้จะถูกนำไปอ้างอิงในขั้นตอนย่อยต่อไป

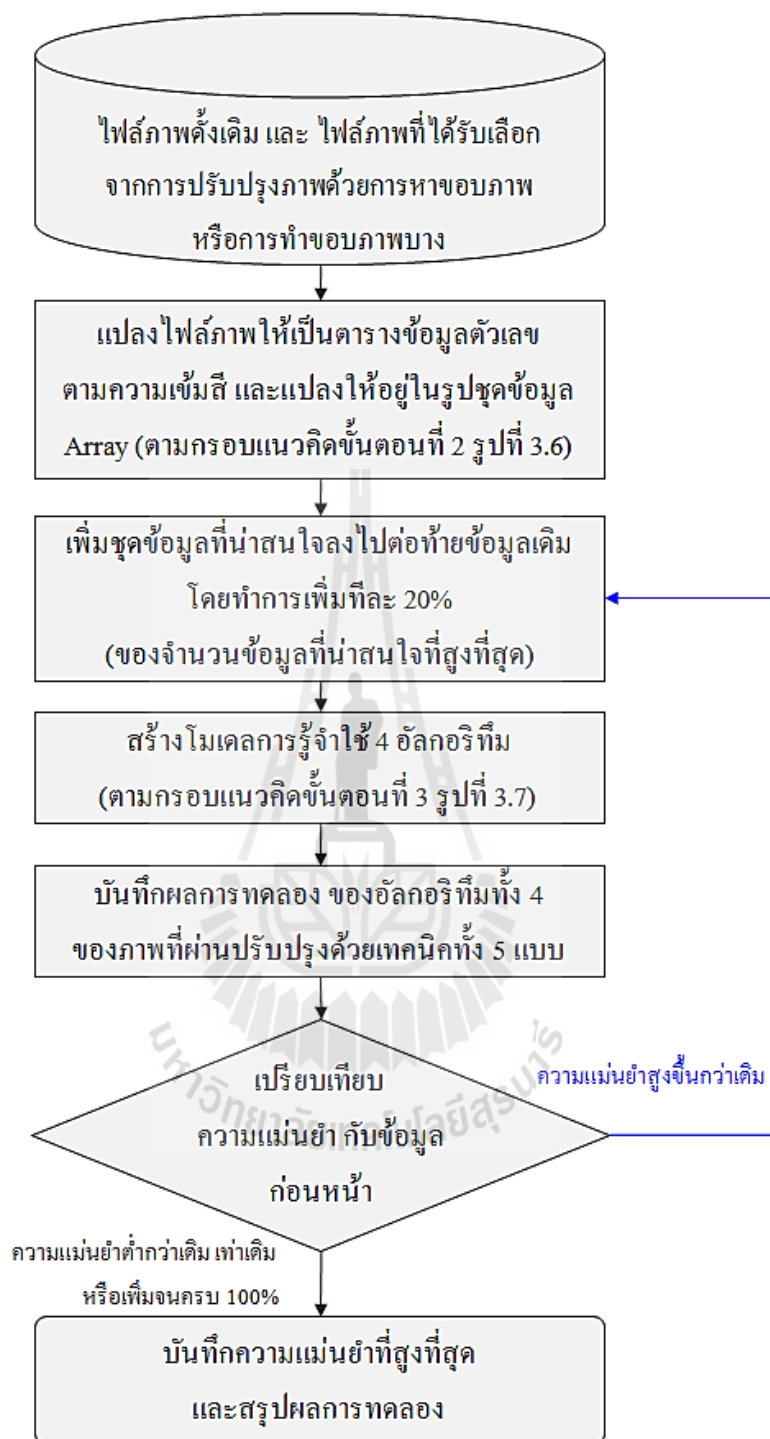




รูปที่ 3.10 ภาพตัวอย่างแสดงการเติมข้อมูลที่มีความเข้มสูงลงไปในแต่ละลายมือชื่อ ตามเปอร์เซ็นต์ของจำนวนข้อมูลที่มีความเข้มสูงที่สุด

กรอบแนวคิดในขั้นตอนนี้ยังได้ให้ความสนใจเกี่ยวกับ เวลาที่ใช้ในการสร้างโมเดลการรู้จำ เนื่องจากการที่เพิ่มข้อมูลลงไปในนั้นอาจจะทำให้เกิดการใช้เวลาในการสร้างโมเดลที่มากขึ้น

โดยในขั้นตอนนี้จะทำการเพิ่มข้อมูลที่มีความน่าสนใจลงไปทีละ 20% จากนั้นจะทำการบันทึกผลการทดลองเพิ่มนำไปเปรียบเทียบกับความแม่นยำของข้อมูลเดิม ซึ่งหากได้ความแม่นยำที่มากขึ้นจะทำต่อไปเรื่อยๆ จนกว่าความแม่นยำจะคงที่ หรือ จนกว่าจะข้อมูลที่เพิ่มเข้าไปจะครบ 100% ดังตัวอย่างที่แสดงใน รูปที่ 3.10 ข้อมูลภาพลายมือชื่อที่อยู่ในรูปของชุดข้อมูล Array ทั้ง 3 ลายมือชื่อ จะถูกเพิ่มชุดข้อมูลที่มีความเข้มสูงลงไป โดยแสดงตัวอย่างทั้งในระดับเพิ่มเข้าไป 20% (เพิ่ม 1 ช่อง จาก maximum = 5 ช่อง), 40% (เพิ่ม 2 ช่อง จาก maximum = 5 ช่อง) และ 100% (เพิ่ม 5 ช่อง จาก maximum = 5 ช่อง)



รูปที่ 3.11 ภาพแผนผังแสดงการทำงานของกรอบแนวคิดขั้นตอนที่ 5

### 3.2 เครื่องมือที่ใช้ในการวิจัย

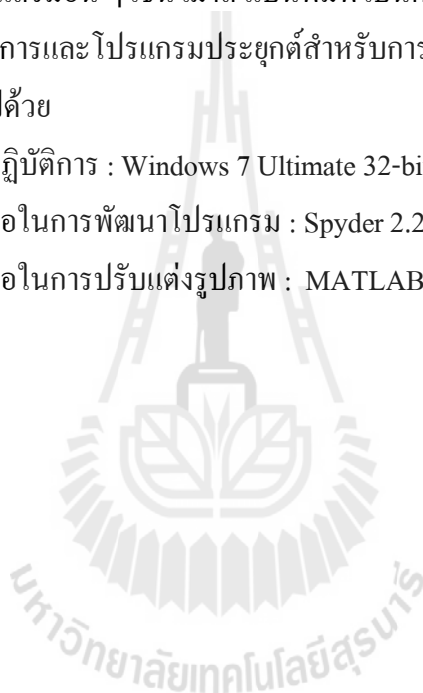
เครื่องมือที่ใช้ในงานวิจัยนี้ ประกอบด้วยฮาร์ดแวร์และซอฟต์แวร์ ดังนี้

1) เครื่องคอมพิวเตอร์สำหรับพัฒนาโปรแกรม และทดสอบอัลกอริทึมโดยมีรายละเอียด ดังนี้

- หน่วยประมวลผลกลาง : Intel® Core i5
- หน่วยความจำสำรอง : 750 GB
- หน่วยความจำหลัก : 2 GB
- อุปกรณ์เสริมอื่น ๆ เช่น เมาส์ แป้นพิมพ์ เป็นต้น

2) ระบบปฏิบัติการและโปรแกรมประยุกต์สำหรับการเพิ่มประสิทธิภาพของโมเดลรู้จำภาพลายมือชื่อประกอบไปด้วย

- ระบบปฏิบัติการ : Windows 7 Ultimate 32-bit Operating System
- เครื่องมือในการพัฒนาโปรแกรม : Spyder 2.2.5 (Python 2.7)
- เครื่องมือในการปรับแต่งรูปภาพ : MATLAB version 7.14 (R2012a)



## บทที่ 4

### การทดสอบและอภิปรายผล

ในการทดสอบประสิทธิภาพโมเดลการรู้จำลายมือชื่อในงานวิจัยชิ้นนี้นั้น จะใช้อัลกอริทึมทั้ง 4 แบบ และจะมีการใช้เทคนิคการปรับแต่งรูปภาพ รวมถึงเทคนิคการเพิ่มชุดข้อมูลที่น่าสนใจลงไป และทำการสรุปผลการทดลองโดยจะมีการเปรียบเทียบทั้งความแม่นยำ และทรัพยากรเวลาที่ใช้ในการสร้างโมเดลและทำนาย

#### 4.1 ข้อมูลที่ใช้ในการทดสอบ

ข้อมูลภาพลายมือชื่อทั้ง 600 ภาพที่ใช้ในงานวิจัยชิ้นนี้จะถูกเก็บลงในเว็บไซต์ <https://sites.google.com/site/nhinganusarapesut/signature/datasets> โดยจะเป็นภาพลายมือชื่อที่เกิดจากการเก็บรวบรวมด้วยตัวผู้วิจัยเอง จากผู้เข้าร่วมทั้งสิ้น 30 คน โดยแต่ละคนจะถูกร้องขอให้เขียนลายมือชื่อคนละ 20 ครั้ง โดยในแต่ละภาพลายมือชื่อจะมีขนาด 38x144 พิกเซล เท่ากันทั้งหมด ดังรูปที่ 4.1



รูปที่ 4.1 ตัวอย่างข้อมูลภาพลายมือชื่อ

## 4.2 การทดสอบประสิทธิภาพระหว่างโมเดลการเรียนรู้จำที่สร้างขึ้นจากอัลกอริทึมทั้ง 4 แบบ

การทดลองนี้จะมีทั้งการเปรียบเทียบประสิทธิภาพความแม่นยำในการจำแนกข้อมูลของอัลกอริทึมทั้ง 4 แบบ ได้แก่ ซัพพอร์ตเวกเตอร์แมชชีน (เคอร์เนลฟังก์ชันสี่ชนิด คือ ฟังก์ชันเชิงเส้น ฟังก์ชันพหุนาม ฟังก์ชันเรเดียลเบสิส และฟังก์ชันซิกมอยด์) เพอร์เซปตรอน นาอิวเบย์ (ฟังก์ชันการกระจายข้อมูลสามชนิด คือ เกาเซียน มัลติโนเมียล และเบอร์นูลลี) และ เคเนียร์สเนเบอร์ โดยเฉพาะอย่างยิ่งในอัลกอริทึม เคเนียร์สเนเบอร์ นั้นจะทำการทดลองด้วยค่า  $k = 1, 2, 3, \dots, 20$  เพื่อหาค่า  $k$  ที่ให้ค่าความแม่นยำที่มีค่ามาก เพื่อนำไปใช้ในการพัฒนาประสิทธิภาพโมเดลการเรียนรู้จำในขั้นตอนถัดไป

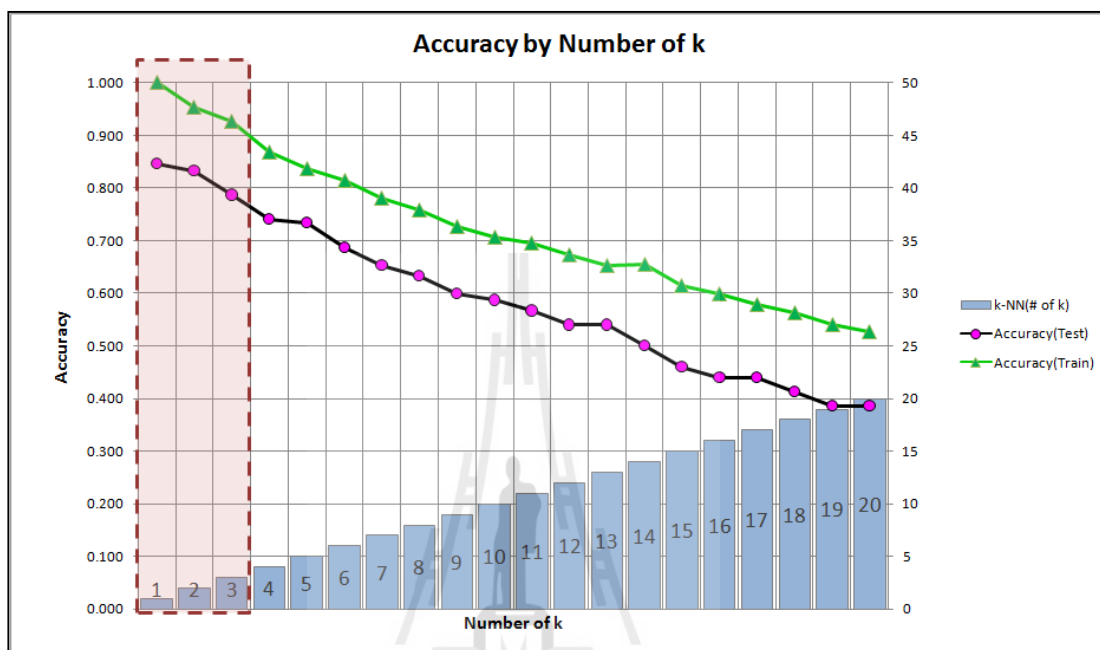
### 4.1.1 ผลการทดลองการเปรียบเทียบประสิทธิภาพระหว่างอัลกอริทึมทั้ง 4 แบบ

จากการทดลอง มีผลลัพธ์ดังต่อไปนี้ อัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีนเป็นอัลกอริทึมที่มีความแม่นยำในการทำนายมากที่สุดอยู่ที่ 95.33% ด้วยเคอร์เนลฟังก์ชันพหุนาม รองลงมาจะเป็น 94.67% ด้วยเคอร์เนลฟังก์ชันเชิงเส้น ซึ่งในอัลกอริทึมอื่น ๆ จะให้ค่าความแม่นยำที่ต่ำลง ซึ่งลดหลั่นกันลงไป เช่น มัลติโนเมียลนาอิวเบย์ อยู่ที่ 82.67% เพอร์เซปตรอน 68.67% ดังที่แสดงให้เห็นในตารางที่ 4.1

ตารางที่ 4.1 แสดงค่าความแม่นยำของแต่ละอัลกอริทึม โดยใช้ภาพดั้งเดิม

Algorithms	Functions	Traditional Method (5,472 columns)	
		Accuracy	
		Training data	Test data
Perceptron	Linear	1.000	0.687
SVM	Linear	1.000	0.947
	Poly	1.000	<b>0.953</b>
	RBF	1.000	0.007
	Sigmoid	0.042	0.007
Naive Bayes	Gaussian	1.000	0.567
	Multinomial	0.973	0.827
	Bernoulli	0.042	0.007
k-NN	k=1	1.000	0.847
	k=2	0.953	0.833
	k=3	0.927	0.787

อัลกอริทึมเคเนียร์เซนเบอร์ เป็นอัลกอริทึมที่มีความน่าสนใจมาก คือเราจะสามารถพบได้ว่าค่า  $k$  นั้นมีความสัมพันธ์แบบแปรผกผันกับค่าความแม่นยำ โดยความแม่นยำจะมีค่ามากที่สุดเมื่อ  $k=1$  (อยู่ที่ 84.7%) และน้อยที่สุด 38.7% เมื่อ  $k = 20$  ดังจะเห็นได้จากรูปที่ 4.2



รูปที่ 4.2 กราฟแสดงความสัมพันธ์ของจำนวน  $k$  ในอัลกอริทึมเคเนียร์เซนเบอร์ และความแม่นยำ

### 4.3 การทดสอบประสิทธิภาพโมเดลการรู้จำโดยใช้ภาพถ่ายมือชื่อที่ผ่านเทคนิคการหาขอบภาพ และการทำขอบภาพให้บางลง

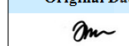

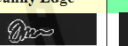



ความแม่นยำของแต่ละอัลกอริทึม เมื่อผ่านเทคนิคการหาขอบภาพ และการทำขอบภาพให้บางลงเข้ามาช่วยนั้น พบว่าอัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีนฟังก์ชันเชิงเส้นนั้นค่าความแม่นยำสูงขึ้นด้วยเทคนิคการตรวจหาขอบภาพแบบโซเบล โดยให้ค่าความแม่นยำ 96.00% ซึ่งเป็นค่าที่สูงที่สุดของการทดลองใช้เทคนิคการหาขอบภาพ และการทำขอบภาพให้บางลงเข้ามาช่วย และยังสูงกว่าความแม่นยำสูงสุดของชุดข้อมูลดั้งเดิมด้วย

จากผลการทดลองที่ได้แสดงได้แสดงไว้ในตารางที่ 4.2 นั้นจะเห็นได้ว่าด้วยเทคนิคการตรวจหาขอบแบบโซเบลนั้น อัลกอริทึมเคเนียร์เซนเบอร์มีค่าความแม่นยำสูงขึ้น 10.66%, 10.67% และ 12.66% ด้วย  $k = 1, 2$  และ 3 ตามลำดับ โดยเฉพาะอย่างยิ่งที่ค่า  $k=1$  นั้น มีความแม่นยำสูงถึง 95.33% ซึ่งเทียบเท่ากับค่าความแม่นยำสูงสุดของชุดข้อมูลเดิม

ในส่วนของความแม่นยำที่ได้จากเทคนิคการหาขอบภาพนั้น จะมีความแม่นยำในการทำนายค่อนข้างน้อย อย่างเช่น เทคนิคการตรวจหาขอบภาพแบบแคนนี่นั้น จะมีความแม่นยำสูงสุดอยู่ที่ 85.33% ด้วยอัลกอริทึมนาอิวเบย์ ฟังก์ชันการกระจายข้อมูลแบบมัลติโนเมียล ซึ่งนับว่ายังน้อยกว่าค่าความแม่นยำสูงสุดของชุดข้อมูลดั้งเดิมถึง 12% และด้วยเทคนิคการหาขอบภาพแบบอื่น ๆ ที่เหลือ ไม่ว่าจะเป็นการตรวจหาขอบภาพแบบโรเบิร์ต การตรวจหาขอบภาพแบบพรีวิตต์ และการทำขอบภาพบางนั้นจะมีค่าความแม่นยำที่ต่ำ โดยที่ไม่มีผลลัพธ์ที่มีค่าความแม่นยำสูงกว่า 90% เลย ไม่ว่าจะใช้อัลกอริทึมใดในการสร้างโมเดลการรู้จำ

จากการทดลองนี้แสดงให้เห็นว่า การใช้เทคนิคการตรวจหาขอบภาพแบบโซเบลนั้นเป็นเพียงเทคนิคเดียวที่ให้ผลลัพธ์ที่สูงกว่าค่าความแม่นยำสูงสุดของชุดข้อมูลดั้งเดิม เราจึงนำภาพถ่ายมือชื่อที่ผ่านการทำขอบภาพแบบโซเบลเท่านั้นเพื่อเข้าสู่กระบวนการพัฒนาประสิทธิภาพโมเดลการรู้จำในขั้นต่อไป

ตารางที่ 4.2 แสดงค่าความแม่นยำของแต่ละอัลกอริทึม เปรียบเทียบระหว่าง 6 วิธีการ

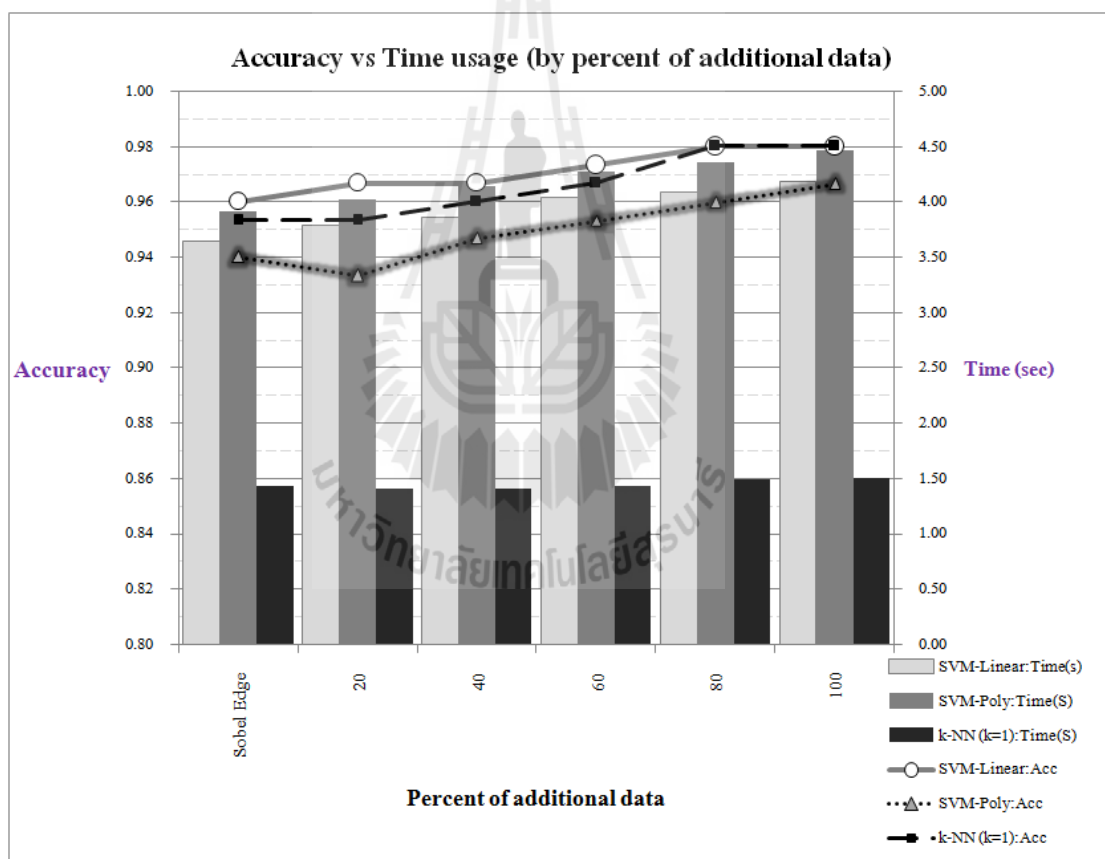
Algorithm	Function	Original Data		Sobel Edge		Canny Edge		Robert Edge		Prewitt Edge		Thining	
													
		Accuracy		Accuracy		Accuracy		Accuracy		Accuracy		Accuracy	
		Training data	Test data	Test data	GAP from Original	Test data	GAP from Original	Test data	GAP from Original	Test data	GAP from Original	Test data	GAP from Original
Perceptron	linear	1.000	68.67%	86.67%	18.00%	64.00%	-4.67%	72.00%	3.33%	65.33%	-3.34%	45.33%	-23.34%
SVM	linear	1.000	94.67%	<b>96.00%</b>	1.33%	83.33%	-11.34%	<b>89.33%</b>	-5.34%	82.00%	-12.67%	86.00%	-8.67%
	poly	1.000	<b>95.33%</b>	94.00%	-1.33%	2.00%	-93.33%	14.67%	-80.66%	2.00%	-93.33%	<b>88.67%</b>	-6.66%
	rbf	1.000	0.67%	0.67%	0.00%	0.67%	0.00%	0.67%	0.00%	0.67%	0.00%	0.67%	0.00%
	sigmoid	0.042	0.67%	0.67%	0.00%	0.67%	0.00%	0.67%	0.00%	0.67%	0.00%	0.67%	0.00%
Naive Bayes	gaussian	1.000	56.67%	20.67%	-36.00%	65.33%	8.66%	72.00%	15.33%	65.33%	8.66%	71.33%	14.66%
	multinomial	0.973	82.67%	89.33%	6.66%	<b>85.33%</b>	2.66%	86.67%	4.00%	<b>87.33%</b>	4.66%	75.33%	-7.34%
	bernoulli	0.042	0.67%	17.33%	16.66%	67.33%	66.66%	72.00%	71.33%	68.67%	68.00%	0.67%	0.00%
k-NN	k = 1	1.000	84.67%	<b>95.33%</b>	10.66%	9.33%	-75.34%	40.00%	-44.67%	12.67%	-72.00%	25.33%	-59.34%
	k = 2	0.953	83.33%	94.00%	10.67%	9.33%	-74.00%	<b>37.33%</b>	-46.00%	14.00%	-69.33%	20.67%	-62.66%
	k = 3	0.927	78.67%	91.33%	12.66%	6.00%	-72.67%	28.00%	-50.67%	8.00%	-70.67%	9.33%	-69.34%

#### 4.4 การทดสอบประสิทธิภาพโมเดลการรู้จำ โดยใช้ภาพถ่ายมือชื่อที่ผ่านเทคนิคการตรวจหาขอบภาพแบบโซเบล และใช้เทคนิคการเพิ่มชุดข้อมูลความเข้มสูง

ในขั้นตอนนี้จะเป็นการทดสอบประสิทธิภาพ โดยจะมีทั้งการทดลองเพื่อหาความสัมพันธ์ระหว่างความแม่นยำ และเวลาที่ใช้ในการสร้างโมเดลการรู้จำ ตามจำนวนเปอร์เซ็นต์ของข้อมูลที่นำเสนอสนใจที่เพิ่มเข้าไป รวมถึงการหาความแม่นยำของแต่ละอัลกอริทึม เมื่อใช้เทคนิคการเพิ่มชุดข้อมูลที่มีความเข้มสูงเข้าช่วย

#### 4.4.1 การทดสอบหาความสัมพันธ์ระหว่างความแม่นยำ และเวลาที่ใช้ กับจำนวนเปอร์เซ็นต์ของข้อมูลความเข้มสูงที่เพิ่มเข้าไป

ในขั้นตอนนี้จะมีการทดสอบหาความสัมพันธ์ระหว่างความแม่นยำ และเวลาที่ใช้ในการสร้างโมเดลการรู้จำ โดยจะทำการเพิ่มข้อมูลเข้าไปทีละ 20% จากจำนวนข้อมูลที่มีความเข้มสูงทั้งหมด และผู้วิจัยพบว่าจำนวนข้อมูลความเข้มสูงที่สุดจากข้อมูลทั้งหมด คือ 717 คอลัมน์ ซึ่งเราจะสามารถเห็นได้จากกราฟในรูปที่ 4.3 ว่าความแม่นยำในการทำนายของแต่ละโมเดลนั้นจะมีค่าสูงขึ้นตามลำดับ โดยจะมีค่าสูงที่สุดเมื่อเพิ่มข้อมูลที่มีความเข้มสูงเข้าไปที่ 100% ในขณะที่เวลาที่เพิ่มขึ้นมาจากการเพิ่มชุดข้อมูลเข้าไบนั้นจะเพิ่มขึ้นเพียงแค่นิดหน่อย โดยในอัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีนจะเพิ่มขึ้น 0.5 วินาที และอัลกอริทึมเคเนียร์เสนเบอร์จะเพิ่มเพียง 0.08 วินาที



รูปที่ 4.3 กราฟแสดงความสัมพันธ์ระหว่างความแม่นยำ เวลาที่ใช้ในการสร้าง โมเดล และเปอร์เซ็นต์ของข้อมูลที่นำสนใจที่เพิ่มเข้าไป



#### 4.4.2 การทดสอบประสิทธิภาพด้านความแม่นยำของการใช้เทคนิคการเพิ่มชุดข้อมูล ความเข้มสูง

จากผลการทดลองที่แสดงไว้ในตารางที่ 4.3 ผลปรากฏว่าการเพิ่มชุดข้อมูลที่มีความเข้มสูงเข้าไปนั้น ช่วยให้ความแม่นยำสูงขึ้นในอัลกอริทึมส่วนใหญ่ เช่น อัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีนจะเพิ่มขึ้นประมาณ 2.1% โดยมีค่าความแม่นยำสูงสุดที่ 97.33% ด้วยฟังก์ชันพหุนาม นอกจากนั้นในขั้นตอนนี้เรายังพบอีกว่า อัลกอริทึมเคเนียร์สเตนเบอร์เป็นอีกอัลกอริทึมที่ให้ค่าความแม่นยำที่สูงที่สุดรวมกันที่ 97.33% เมื่อใช้ค่า  $k=1$  โดยมีความแม่นยำสูงขึ้นไปถึง 14.95% เมื่อเปรียบเทียบกับข้อมูลชุดดั้งเดิม โดยถือว่าการเพิ่มขึ้นอย่างมีนัยสำคัญ และเมื่อพิจารณาที่  $k=2$  และ 3 นั้นจะพบว่ามีความแม่นยำเพิ่มสูงขึ้นเช่นกันจาก 83.3% และ 78.7% เป็น 97.3% และ 93.3% ตามลำดับ

ตารางที่ 4.3 แสดงค่าความแม่นยำของแต่ละอัลกอริทึมเมื่อเพิ่มชุดข้อมูลความเข้มสูง

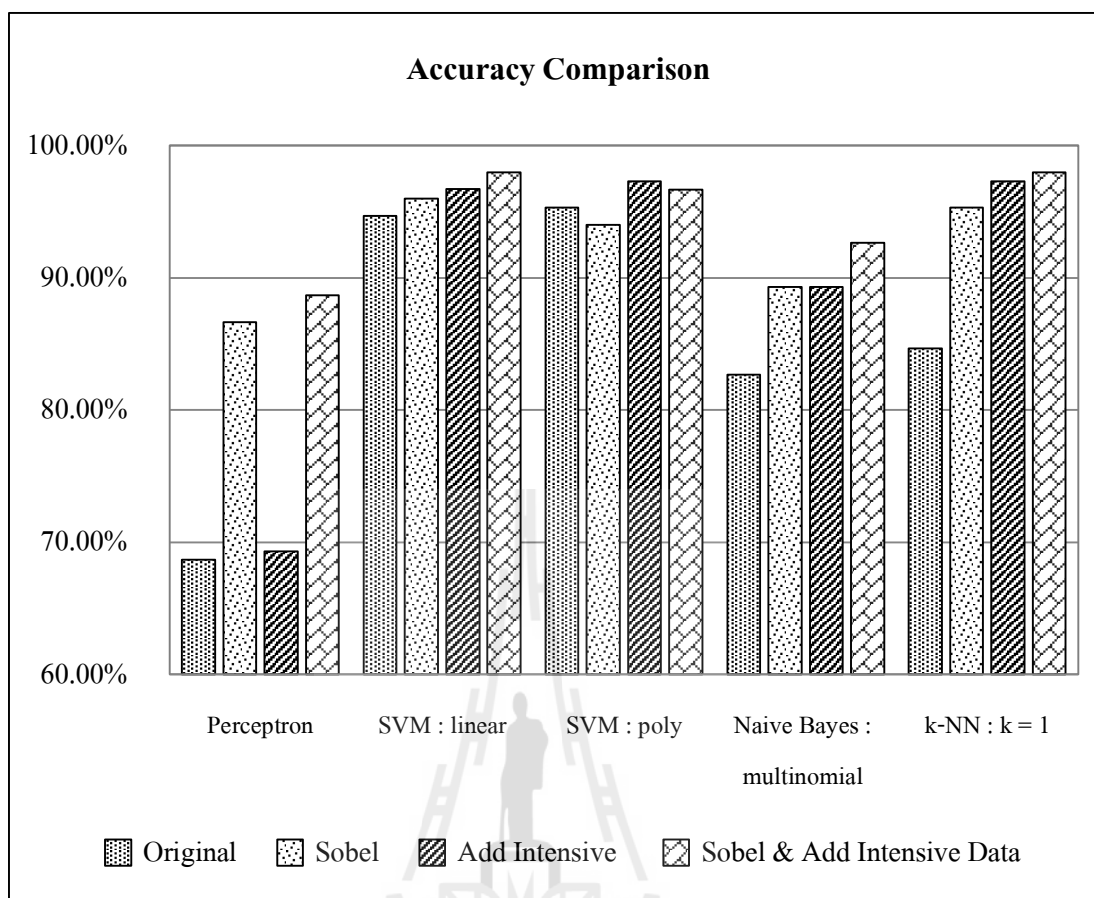
Algorithm	Function	Traditional Method		Intensive Data Method		Improvement
		(5,472 columns)		(5,472+717 columns)		
		Accuracy		Accuracy		
		Training data	Test data	Training data	Test data	
Perceptron	linear	1.000	0.687	1.000	0.693	0.96%
SVM	linear	1.000	0.947	1.000	0.967	2.11%
	poly	1.000	0.953	1.000	0.973	2.10%
	rbf	1.000	0.007	1.000	0.007	0.00%
	sigmoid	0.042	0.007	0.042	0.007	0.00%
Naive Bayes	gaussian	1.000	0.567	1.000	0.567	0.00%
	multinomial	0.973	0.827	0.987	0.893	8.06%
	bernoulli	0.042	0.007	0.042	0.007	0.00%
k-NN	k = 1	1.000	0.847	1.000	0.973	14.95%
	k = 2	0.953	0.833	0.982	0.953	14.40%
	k = 3	0.927	0.787	0.980	0.933	18.63%

จากตารางผลการทดลองที่ 4.4 แสดงให้เห็นถึงผลลัพธ์ที่ดีที่สุดในงานวิจัยชิ้นนี้ซึ่งอยู่ที่ 98% โดยพบว่าการใช้ภาพถ่ายมือชื่อที่ผ่านการหาขอบภาพแบบโซเบล และด้วยเทคนิคการเพิ่มชุดข้อมูลความเข้มสูงเข้าไปนั้น ทำให้ค่าความแม่นยำสูงขึ้นอย่างมีนัยสำคัญโดยในการทดลองนี้เราจะได้รับค่าความแม่นยำที่ 98% ถึงสองอัลกอริทึม นั่นคือ อัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีนด้วยฟังก์ชันเชิงเส้น และอัลกอริทึมเคเนียร์สเนเบอร์ เมื่อค่า  $k = 1$

ด้วยเทคนิคการเพิ่มประสิทธิภาพโมเดลการเรียนรู้จำทั้งสองอย่างที่น่ามาผสมผสานกันในงานวิจัยชิ้นนี้ จะทำให้อัลกอริทึมส่วนใหญ่มีค่าความแม่นยำสูงขึ้น มีเพียงอัลกอริทึมนาอิวเบย์ฟังก์ชันการกระจายข้อมูลแบบเกาเซียนเท่านั้น ที่ให้ค่าความแม่นยำที่ลดลงจาก 56.67% เหลือเพียง 21.33%

ตารางที่ 4.4 แสดงค่าความแม่นยำของภาพถ่ายมือชื่อที่ผ่านกระบวนการตรวจหาขอบแบบโซเบล และเพิ่มชุดข้อมูลความเข้มสูง

Algorithm	Function	Original		Sobel Edge		Sobel Edge + Additional Intensive Data Method	
		Accuracy		Accuracy		Accuracy	
		Train	Test	Test	Improvement from original	Test	Improvement from original
Perceptron	linear	100%	68.67%	86.67%	26.21%	88.67%	29.12%
SVM	linear	100%	94.67%	96.00%	1.40%	98.00%	3.52%
	poly	100%	95.33%	94.00%	-1.40%	96.67%	1.41%
	rbf	100%	0.67%	0.67%	0.00%	0.67%	0.00%
	sigmoid	4.22%	0.67%	0.67%	0.00%	0.67%	0.00%
Naive Bayes	gaussian	100.0%	56.67%	20.67%	-63.53%	21.33%	-62.36%
	multinomial	97.33%	82.67%	89.33%	8.06%	92.67%	12.10%
	bernoulli	4.22%	0.67%	17.33%	2486.57%	18.00%	2586.57%
k-NN	k = 1	100.00%	84.67%	95.33%	12.59%	98.00%	15.74%
	k = 2	95.33%	83.33%	94.00%	12.80%	95.33%	14.40%
	k = 3	92.67%	78.67%	91.33%	16.09%	96.00%	22.03%



รูปที่ 4.4 กราฟแสดงการเปรียบเทียบความแม่นยำระหว่างเทคนิคต่าง ๆ กับข้อมูลดั้งเดิม

#### 4.5 อภิปรายผล

จากข้อมูลตั้งต้นที่เป็นภาพลายมือชื่อทั้งสิ้น 600 ภาพจากผู้เข้าร่วม 30 คน และผลการทดลองที่ได้นำเสนอมาข้างต้นนั้น สามารถสรุปการเปรียบเทียบประสิทธิภาพ และนำมาวิเคราะห์ได้ดังต่อไปนี้

1) วิธีและภาพรูปภาพลายมือชื่อดั้งเดิม (Original Data) ซัพพอร์ตเวกเตอร์แมชชีนด้วยฟังก์ชันพหุนาม เป็นอัลกอริทึมที่มีความแม่นยำในการทำนายมากที่สุดอยู่ที่ 95.33% ซึ่งจะพบได้จากการศึกษางานวิจัยด้านการรู้จำลายมือชื่ออยู่แล้วว่า ซัพพอร์ตเวกเตอร์แมชชีนมักจะเป็นอัลกอริทึมที่ให้ค่าความแม่นยำสูงสุด หรืออยู่ในอันดับต้น ๆ ของงานวิจัยแต่ละชิ้น โดยค่าความแม่นยำที่ 95.33% นั้นจะถูกใช้เป็นค่าความแม่นยำที่เอาไว้อ้างอิงในขั้นตอนการพัฒนาเพิ่มความแม่นยำ สิ่งที่สำคัญอีกข้อที่สามารถสังเกตเห็นได้จากขั้นตอนนี้คือ การที่อัลกอริทึมเคเนียร์สเนเบอร์ในการทดลองนี้ให้ค่าความแม่นยำที่มีความสัมพันธ์กับค่า  $k$  อย่างมีนัยสำคัญ โดยความสัมพันธ์นี้มีลักษณะการแปรผันแบบผกผัน

2) ด้วยเทคนิคการหาขอบภาพและการทำขอบให้บางลง ในขั้นตอนนี้ผลลัพธ์ที่ได้นั้น จะเห็นได้ว่าวิธีการตรวจ หาขอบภาพแบบ โซเบลนั้นเป็นวิธีที่ให้ค่าความแม่นยำที่ดีที่สุด คือ 96.00% และเป็นเพียงวิธีเดียวที่มีค่าความแม่นยำสูงกว่า ค่าความแม่นยำสูงสุดของข้อมูลชุดเดิม ซึ่งได้มาจาก อัลกอริทึมซอฟต์แวร์แมชชีนฟังก์ชันเชิงเส้น ส่วนในอัลกอริทึมเคเนียร์สเนเบอร์นั้น ได้ค่าเท่ากับค่าสูงสุดของชุดข้อมูลเดิมที่ 95.33% โดยอัลกอริทึมส่วนใหญ่จะมีค่าความแม่นยำสูงขึ้น ซึ่งทางผู้วิจัยคาดว่าน่าจะเกิดจากการที่การตรวจหาขอบภาพแบบ โซเบลนั้นทำให้ขอบของตัวลายมือชื่อ มีความหนามากขึ้น จึงทำให้มีข้อมูลที่น่ามาใช้ในการสร้างโมเดลการรู้จำมากขึ้น ซึ่งจะสอดคล้องกับผลลัพธ์ที่ได้จากเทคนิคการหาขอบภาพให้บางลง ซึ่งความแม่นยำที่ได้มานั้นลดลงในเกือบทุก อัลกอริทึม

3) ด้วยเทคนิคการเพิ่มชุดข้อมูลที่มีความเข้มสูง (Additional Intensive Data Method) วิธีการนี้ทางผู้วิจัยได้ทำการคิดค้นวิธีการใหม่ ที่คาดหวังว่าจะทำให้การทำงานของแต่ละอัลกอริทึม มีค่าความแม่นยำมากขึ้นกว่าเดิม โดยการเพิ่มชุดข้อมูลที่มีค่าความเข้มสีค่อนข้างมากเพิ่มเข้าไป ต่อท้ายกับชุดข้อมูลเดิม กล่าวคือสนใจระดับความเข้มสีที่อยู่ระหว่าง 0 ถึง 200 เท่านั้น จากนั้นนำ ข้อมูลใหม่ที่ผ่านมาการคัดกรองแล้วเข้าไปต่อท้ายจากชุดข้อมูล Array เดิมอีก 717 คอลัมน์ รวมกับ ของเดิม 5,472 คอลัมน์ เป็นทั้งสิ้น 6,189 คอลัมน์ โดยภาพถ่ายมือชื่อที่นำมาใช้นั้นจะเป็นภาพถ่ายมือชื่อที่ได้จากการใช้เทคนิคการตรวจหาขอบภาพแบบ โซเบลเข้าช่วยก่อนแล้ว ซึ่งผลการ ทดลองจากวิธีการใหม่นี้พบว่า อัลกอริทึมเพอร์เซปตรอนเป็นอัลกอริทึมที่มีค่าความแม่นยำเพิ่มขึ้น มากที่สุด โดยเพิ่มจาก 68.67% เป็น 88.67% ซึ่งเพิ่มจากเดิม 20% ทั้งนี้ น่าจะเป็นเพราะค่าความ แม่นยำด้วยวิธีการดั้งเดิมนั้นต่ำอยู่แล้ว จึงสามารถเห็นการเพิ่มขึ้นของความแม่นยำได้อย่างชัดเจน ในส่วนของค่าที่สูงที่สุดในงานวิจัยชิ้นนี้นี้อยู่ที่ 98% ซึ่งมาจากสองอัลกอริทึม ได้แก่ อัลกอริทึม เคเนียร์สเนเบอร์ ที่ค่า  $k=1$  มีค่าความแม่นยำเพิ่มขึ้น จาก 84.67% เป็น 98.00% ซึ่งเพิ่มขึ้นถึง 13.33% อีกอัลกอริทึมคือ ซัพพอร์ตเวกเตอร์แมชชีนฟังก์ชันเชิงเส้น ซึ่งเป็นอัลกอริทึมที่ให้ค่าความ แม่นยำที่สูงตั้งแต่เดิมอยู่แล้ว

ในส่วนของเวลาที่ใช้ในการสร้างโมเดลนั้น พบว่าจะเพิ่มขึ้นเพียงเล็กน้อย เมื่อเทียบกับ ความแม่นยำที่สูงขึ้น นับได้ว่าเป็นการเพิ่มขึ้นที่คุ้มค่ามาก ตัวอย่างเช่น อัลกอริทึมเคเนียร์สเนเบอร์ จะมีความแม่นยำสูงขึ้น 3% โดยใช้เวลาเพิ่มขึ้นเพียง 0.08 วินาที

## บทที่ 5

### สรุปผลการวิจัยและข้อเสนอแนะ

ในปัจจุบันนั้นการตรวจสอบความถูกต้องของตัวบุคคลโดยใช้ข้อมูลทางชีวภาพ หรือไบโอเมตริกซ์ได้รับความนิยมเป็นอย่างมากเช่น ลายนิ้วมือ ใบหน้า ม่านตา ลายมือชื่อ เป็นต้น สิ่งเหล่านี้จะถูกเก็บรวบรวมเป็นข้อมูล เพื่อนำมาเปรียบเทียบจำแนกความแตกต่างของแต่ละบุคคลนั้นออกจากกัน หรือเพื่อจดจำ รวมไปถึงการรู้จำลักษณะเฉพาะของแต่ละบุคคล

การใช้ไบโอเมตริกซ์ ในการตรวจสอบตัวบุคคล ทำให้เกิดความสะดวกในการที่ไม่ต้องพกเอกสารเพื่อระบุตัวบุคคล และยังสามารถลดปัญหาเรื่องการปลอมแปลงเอกสาร ได้อีกด้วย การรู้จำลายมือชื่อนั้นก็เป็นหนึ่งในสิ่งที่เป็นที่นิยมอย่างมาก เนื่องจากไม่ต้องมีอุปกรณ์เฉพาะทางเพื่อมาใช้ร่วม อย่างเช่น เครื่องสแกนลายมือ หรือเครื่องสแกนม่านตา

งานวิจัยที่เกี่ยวข้องกับการตรวจสอบลายมือชื่อ และการรู้จำลายมือชื่อนั้น ส่วนใหญ่จะมุ่งเน้นไปที่การศึกษาเปรียบเทียบ และการปรับปรุงเพื่อเพิ่มประสิทธิภาพของโมเดลด้วยเทคนิคการหาขอบภาพและการทำขอบภาพให้บางลง ซึ่งเป็นเทคนิคที่ผู้วิจัยส่วนใหญ่นิยมนำมาใช้ ทางผู้วิจัยจึงได้สังเกตเห็นถึงความสำคัญ ของการเพิ่มประสิทธิภาพความแม่นยำในการรู้จำภาพลายมือชื่อด้วยเทคนิคใหม่ โดยในงานวิจัยนี้เป็นการเพิ่มชุดข้อมูลที่มีความเข้มสีสูงเข้าไปต่อท้ายชุดข้อมูลเดิม

ในส่วน of ฐานข้อมูลที่น่ามาใช้ในงานวิจัยนั้น ทางผู้วิจัยเลือกที่จะเก็บรวบรวมตัวอย่างลายมือชื่อด้วยตนเอง จากผู้เข้าร่วมทำการวิจัย ที่เป็นนักศึกษาซึ่งมีการใช้ลายมือชื่อจริงในชีวิตประจำวัน นับได้ว่าเป็นแหล่งข้อมูลที่แสดงให้เห็นถึงการนำไปใช้งานได้จริง

#### 5.1 ขั้นตอนการดำเนินงานวิจัย

ขั้นตอนการดำเนินงานของงานวิจัยชิ้นนี้ ถูกแบ่งออกเป็น 5 ขั้นตอนหลักแนวคิดหลักของงานวิจัยนี้ โดยในแต่ละขั้นตอนจะมีส่วนที่น่าสนใจ และจุดที่มุ่งเน้นที่แตกต่างกันดังนี้

- 1) วิธีเก็บข้อมูลลายมือชื่อ และการแปลงภาพลายมือชื่อให้เป็นไฟล์ภาพ ในขั้นตอนนี้จะเน้นที่การเก็บลายมือชื่อจากผู้ใช้งานจริง และมีการเว้นระยะในการเขียนลายมือชื่อ 14 วัน เพื่อแสดงให้เห็นว่าผลลัพธ์ของงานวิจัยได้คำนึงถึงความแตกต่างเพียงเล็กน้อยของลายมือชื่อในตัวบุคคลเดียวกันตามช่วงเวลา

2) การแปลงข้อมูลภาพถ่ายมือชื่อให้เป็นข้อมูลทางตัวเลข ในงานวิจัยชิ้นนี้จะเป็นการนำเอาภาพถ่ายมือชื่อมาแปลงเป็นตารางที่เป็นข้อมูลตัวเลขตามความเข้มสีก่อน โดยใช้ IDE Spyder และภาษาไพธอน โดยหลักการคือนำเอาภาพถ่ายมือชื่อทั้งหมด 600 ภาพมาหาค่าความเข้มสีทั้งหมดพร้อม ๆ กัน และทำการแบ่งออกเป็น 256 ระดับ ตั้งแต่ 0 (สีดำ) ไปจนถึง 255 (สีขาว)

3) การทดสอบเปรียบเทียบประสิทธิภาพของแต่ละอัลกอริทึม โดยในขั้นตอนนี้จะสนใจค่าความแม่นยำที่สูงที่สุด เพื่อนำไปเป็นค่าอ้างอิงในการเพิ่มประสิทธิภาพโมเดลการเรียนรู้จำในขั้นตอนถัด ๆ ไป และนอกจากนี้ในขั้นตอนนี้จะทำการศึกษาถึงฟังก์ชันการใช้งานย่อยในแต่ละอัลกอริทึมอีกด้วย

4) การเพิ่มประสิทธิภาพการเรียนรู้ โดยใช้เทคนิคการหาขอบภาพและการทำภาพให้บางลงเป็นขั้นตอนแรกของการเพิ่มประสิทธิภาพโมเดลการเรียนรู้ ซึ่งจะมุ่งเน้นไปที่การปรับปรุงรูปภาพลายมือชื่อที่ผ่านกระบวนการเหล่านั้นจะแตกต่างไปจากเดิม ซึ่งจะแตกต่างกันในแต่ละเทคนิคอีกด้วย โดยผลลัพธ์ที่เราคาดหวังจากขั้นตอนนี้คือ เทคนิคใดที่จะสามารถทำให้ความแม่นยำในการทำนายเพิ่มมากขึ้นจากการใช้ภาพถ่ายมือชื่อดั้งเดิม

5) การเพิ่มประสิทธิภาพการเรียนรู้ โดยการใช้เทคนิคเพิ่มชุดข้อมูลที่มีความเข้มสูง ขั้นตอนนี้เป็นเทคนิคใหม่ที่ทางผู้วิจัยคิดค้นขึ้นมา โดยได้มาจากแนวความคิดที่ว่า การที่เราแปลงภาพถ่ายลายมือชื่อให้เป็นตารางข้อมูลนั้น ส่วนใหญ่จะเป็นข้อมูลที่มีความเข้มสีดำหรือสีขาวอยู่มาก และมีข้อมูลความเข้มสีสูงเพียงเล็กน้อย ซึ่งทางผู้วิจัยเห็นว่าข้อมูลที่มีความเข้มสีสูงนั้นเป็นข้อมูลที่น่าสนใจ จึงไปทำการทดลองเพิ่มชุดข้อมูลเหล่านี้ลงไปต่อท้ายชุดข้อมูลเดิม ในขั้นตอนนี้ นอกจากสนใจที่ความแม่นยำแล้วยังจะทำการเปรียบเทียบในเรื่องเวลาที่ใช้อีกด้วย

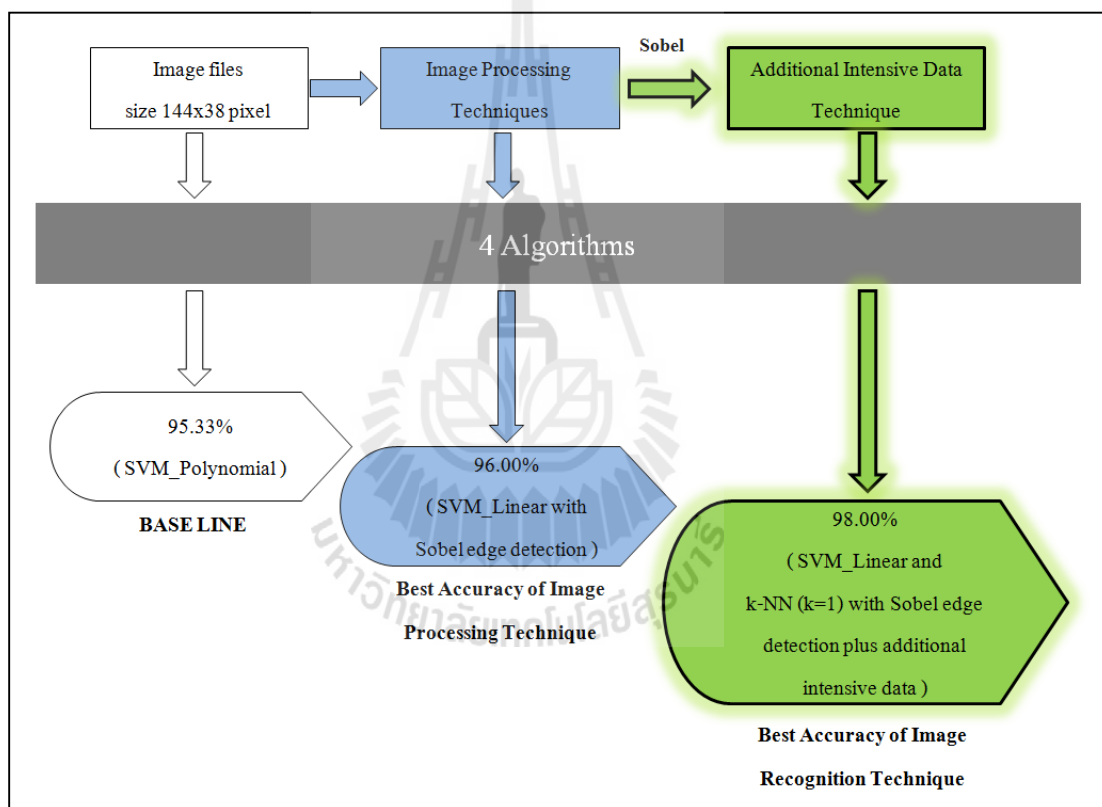
## 5.2 สรุปผลการวิจัย

จากผลการทดลองด้วยวิธีการใช้ภาพถ่ายมือชื่อดั้งเดิม อัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีน ฟังก์ชันพหุนามให้ค่าความแม่นยำสูงสุด อยู่ที่ 95.33% และนอกจากนั้นยังสังเกตเห็นได้อีกว่า อัลกอริทึมเคเนียร์เรสเนเบอร์ในการทดลองนี้ ให้ค่าความแม่นยำที่มีความสัมพันธ์กับค่า  $k$  อย่างมีนัยสำคัญ โดยความสัมพันธ์ที่ได้มีลักษณะการแปรผันแบบผกผัน

จากการทดลองด้วยเทคนิคการหาขอบภาพและการทำภาพให้บางลง จะเห็นได้ว่าวิธีการตรวจหาขอบภาพแบบโซเบลนั้นเป็นวิธีที่ให้ค่าความแม่นยำที่ดีที่สุด คือ 96% และเป็นเพียงวิธีเดียวที่มีค่าความแม่นยำสูงกว่า ค่าความแม่นยำสูงสุดของข้อมูลชุดเดิม ซึ่งได้มาจากอัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีนฟังก์ชันเชิงเส้น ส่วนในอัลกอริทึมเคเนียร์เรสเนเบอร์นั้น ได้ค่าเท่ากับค่าสูงสุดของชุดข้อมูลเดิมที่ 95.33% โดยอัลกอริทึมส่วนใหญ่จะมีค่าความแม่นยำสูงขึ้น

ในการทดลองสุดท้ายด้วยเทคนิคการเพิ่มชุดข้อมูลที่มีความเข้มสูงพบว่า อัลกอริทึมเพอร์เซปตรอนเป็นอัลกอริทึมที่มีค่าความแม่นยำเพิ่มขึ้นมากที่สุด โดยเพิ่มจาก 68.67% เป็น 88.67% ซึ่งเพิ่มจากเดิม 20% ในส่วนของค่าที่สูงที่สุดในงานวิจัยชิ้นนี้นั้นอยู่ที่ 98% โดยมาจากสองอัลกอริทึม ได้แก่ อัลกอริทึมเคเนียร์เรสเนเบอร์ ที่ค่า  $k=1$  และซัพพอร์ตเวกเตอร์แมชชีนฟังก์ชันเชิงเส้น ดังที่แสดงให้เห็นในรูปที่ 5.1

ในส่วนของทรัพยากรเวลาที่ใช้ในการสร้างโมเดลนั้น จะพบได้ว่าการใช้เทคนิคการเพิ่มชุดข้อมูลที่มีความเข้มสูงเข้าช่วยนั้น ทำให้เวลาเพิ่มขึ้นมาเพียงเล็กน้อยเท่านั้น โดยเวลาที่เพิ่มขึ้นมามากที่สุดคือ 0.5 วินาที



รูปที่ 5.1 แผนภาพแสดงผลการทดลองที่ดีที่สุด ในแต่ละขั้นตอนของการปรับปรุงประสิทธิภาพ

### 5.3 ปัญหาและข้อเสนอแนะ

ในขั้นตอนการเก็บข้อมูลนั้นทางผู้วิจัยสังเกตเห็นว่าอาจจะสามารถเพิ่มจำนวนของคนที่จะเข้าร่วมงานวิจัยให้มากขึ้น โดยอาจจะเพิ่มกลุ่มของเด็กวัยรุ่น คนวัยทำงาน และคนวัยเกษียณ ซึ่งจะสามารถแสดงถึงความสามารถรองรับการใช้งานของกลุ่มบุคคลทุกกลุ่ม

งานวิจัยชิ้นนี้มีลักษณะการทำงานแบบออนไลน์ ซึ่งหากมีการพัฒนาไปเป็นการใช้งานแบบออนไลน์จะทำให้เกิดความสะดวกต่อการใช้งาน และเหมาะสมกับการใช้งานในยุคปัจจุบันมากยิ่งขึ้น

เนื่องจากเทคนิคการปรับปรุงรูปภาพในยุคปัจจุบันนั้นหลากหลาย และมีการพัฒนาไปอย่างรวดเร็ว ทางผู้วิจัยจึงสังเกตเห็นว่าในอนาคตอาจจะมีเทคนิคการปรับปรุงรูปภาพแบบอื่นที่เหมาะสมกว่าเทคนิคที่งานวิจัยนี้เลือกใช้





## รายการอ้างอิง

- กิตติศักดิ์ เกิดประสพ. (2557). **Data Mining Application** [ออนไลน์]. ได้จาก:  
[https://sites.google.com/site/kittisakthailand56/home/3\\_56-dm-applications](https://sites.google.com/site/kittisakthailand56/home/3_56-dm-applications)
- นิตยา เกิดประสพ. (2557). **Knowledge Discovery and Data Mining** [ออนไลน์]. ได้จาก:  
<https://sites.google.com/site/nittayak/home/kdd>
- Aci M., Inan C., & Avci M. (2010). A hybrid classification method of k nearest neighbor, Bayesian methods and genetic algorithm. **Expert Systems with Applications**. 37:5061–5067.
- Arya, S., Mount, D. M., Netanyahu, N. S., Silverman, R., & Wu, A. Y. (1998). An optimal algorithm for approximate nearest neighbor searching fixed dimensions. **Journal of the ACM (JACM)**. 45(6):891-923.
- Cristianini, N., & Shawe-Taylor, J. (2000). **An introduction to support vector machines and other kernel-based learning methods**. Cambridge university press.
- Dougherty, G. (2012). **Pattern Recognition and Classification: An Introduction**. Springer.
- Flach, P. (2012). **Machine learning: The art and science of algorithms that make sense of data**. Cambridge University Press.
- Frias-Martinez, E., Sanchez, A., & Velez, J. (2006). Support vector machines versus multi-layer perceptrons for efficient off-line signature recognition. **Engineering Applications of Artificial Intelligence**. 19(6):693-704.
- Garreta, R., & Moncecchi, G. (2013). **Learning scikit-learn: Machine Learning in Python**. Published by Packt Publishing Ltd.
- Gonzalez R. C., & Woods R. E. (2001). **Digital Image Processing**. Prentice Hall.
- Guerbai, Y., Chibani, Y., & Hadjadji, B. (2014). The effective use of the one-class SVM classifier for handwritten signature verification based on writer-independent parameters. **Pattern Recognition**.

- Mangini, F. M., Impedovo, S., & Barbuzzi, D. (2014). A novel prototype generation technique for handwriting digit recognition. **Pattern Recognition**. 47:1002–1010.
- Minsky, M., L., & Papert, S. (1969). **Perceptrons**. MIT Press. Cambridge MA. 105-110.
- Mitchell, T. M. (1997). **Machine Learning**. Boston: McGraw-Hill.
- Mohammed, J. Z., & Wagner M. Jr. (2014). **Data Mining and Analysis: Fundamental Concepts and Algorithms**. Cambridge University Press
- Niu, X. X., & Suen, C. Y. (2012). A novel hybrid CNN–SVM classifier for recognizing handwritten digits. **Pattern Recognition**. 45:1318–1325.
- Oded Maimon & Lior Rokach (2010). **Data Mining and Knowledge Discovery Handbook**. Springer
- Suman R. and Kumar D. (2013). Punjabi Offline Signature Verification System Using Neural Network. **International Journal of Engineering and Advanced Technology (IJEAT)**. 3(2):419-424.
- Theodoridis S., Pikrakis A., Koutroumbas K., & Cavouras D. (2010). **An Introduction to Pattern Recognition: A MATLAB Approach**. Academic Press
- Unar, J. A., Seng, W. C., & Abbasi, A. (2014). A review of biometric technology along with trends and prospects. **Pattern Recognition**. 47(8):2673-2688.
- Vargas, J. F., Ferrer, M. A., Travieso, C. M., & Alonso, J. B. (2011). Off-line signature verification based on grey level information using texture features. **Pattern Recognition**. 44(2):375-385.
- Zheng S., Liu J., & Tian J. W. (2004). A new efficient SVM-based edge detection method. **Pattern Recognition Letters**. 25:1143-1154.



ภาคผนวก ก

การใช้งานโปรแกรม

## การใช้งานโปรแกรม

ในส่วนนี้จะอธิบายการใช้งานของโปรแกรมสำหรับการทำงานวิจัยขั้นนี้ตามหัวข้อที่จะกล่าวถึงดังต่อไปนี้

### ก.1 การสร้างโมเดลการเรียนรู้จำด้วยภาษาไพธอน

ในขั้นตอนนี้จะเป็นการแบ่งข้อมูลออกเป็น 2 ชุดด้วยการสุ่ม โดยจะแบ่ง 75% ให้เป็นข้อมูลสำหรับการฝึก และอีก 25% ที่เหลือ เป็นข้อมูลสำหรับการทดสอบ หลังจากนั้นจะเป็นการสร้างโมเดลการเรียนรู้จำด้วยอัลกอริทึมทั้ง 4 แบบ ซึ่งทั้งหมดของขั้นตอนนี้จะกระทำด้วย IDE Spyder โดยใช้ภาษาไพธอน ตามรูปที่ ก.1 และ ก.2

```

79
80 #data = np.genfromtxt('E:\Edge1_signature_data_sobel_Add20Percent.csv', dtype=None, delimiter=',', use
81 #target = np.genfromtxt('E:\Edge1_signature_data_sobel_Add20Percent.csv', dtype=None, delimiter=',', u
82
83 #####
84
85 X_train, X_test, y_train, y_test = train_test_split(data, target, test_size=0.25, random_state=42)
86 from time import time
87

```

รูปที่ ก.1 ตัวอย่างแสดงการสุ่มแบ่งข้อมูลสำหรับการฝึก และการทดสอบ

Name	Type	Size	Value
X_test	int32	(150, 5471)	array([[255, 255, 255, ..., 255, 255, 255], [255, 255, 255, ..., 255, 255, 255 ...
X_train	int32	(450, 5471)	array([[255, 255, 255, ..., 255, 255, 255], [254, 255, 255, ..., 255, 255, 255 ...
acc1	float64	1	1.0
acc2	float64	1	0.8466666666666668
data	int32	(600, 5471)	array([[255, 255, 255, ..., 255, 255, 255], [255, 255, 255, ..., 255, 255, 255 ...
e	float	1	2.718281828459045

```

88 # Perceptron
89 from sklearn.linear_model import Perceptron
90 from sklearn import metrics
91 t0 = time()
92 clf = Perceptron(n_iter=100).fit(X_train , y_train)
93 y1_pred = clf.predict(X_train)
94 y2_pred = clf.predict(X_test)
95 acc1 = metrics.accuracy_score(y_train, y1_pred)
96 acc2 = metrics.accuracy_score(y_test, y2_pred)
97 print("#####")
98 print("done in %.3fs" % (time() - t0))
99 print("Perceptron")
100 print("Accuracy[Training]: %.4f" % acc1)
101 print("Accuracy[Testing]: %.4f" % acc2)
102 print("#####\n")
103 #####
104 # SVM
105 svm_mode = ['linear', 'poly', 'rbf', 'sigmoid']
106 for ex in svm_mode:
107
108     t1 = time()
109     clf1 = svm.SVC(gamma=0.001, C=100., kernel=ex)
110     #clf = svm.SVC(kernel=ex)
111     clf1.fit(X_train, y_train)
112     y1_pred = clf1.predict(X_train)
113     y2_pred = clf1.predict(X_test)
114     acc1 = metrics.accuracy_score(y_train, y1_pred)
115     acc2 = metrics.accuracy_score(y_test, y2_pred)
116     print("#####")
117     print("done in %.3fs" % (time() - t1))
118     print("SVM Mode: %s" % ex)
119     print("Accuracy[Training]: %.4f" % acc1)
120     print("Accuracy[Testing]: %.4f" % acc2)
121     print("#####\n")

```

รูปที่ ก.2 ตัวอย่างการสร้างโมเดลการเรียนรู้จำด้วยภาษาไพธอน

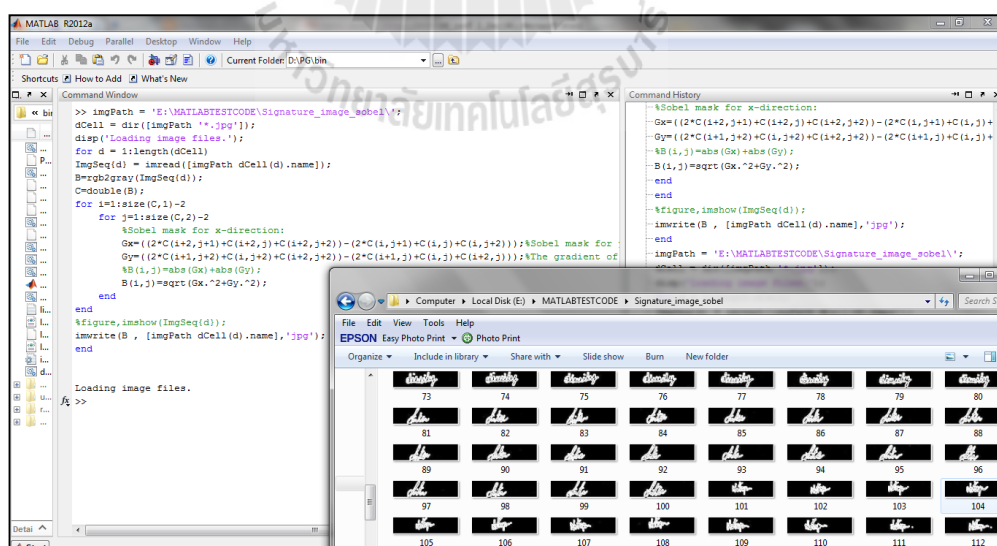
## ก.2. การหาขอบภาพและการทำขอบภาพให้บางลงโดยใช้ MATLAB

โปรแกรม MATLAB ถูกนำมาใช้ในงานวิจัยชิ้นนี้ โดยการนำมาใช้ในการหาขอบภาพและการทำขอบภาพให้บางลง ซึ่งในแต่ละเทคนิคนั้นจะทำให้ภาพที่ได้รับต่างออกไปจากต้นฉบับเล็กน้อยดังที่แสดงไว้ในรูปที่ ก.3

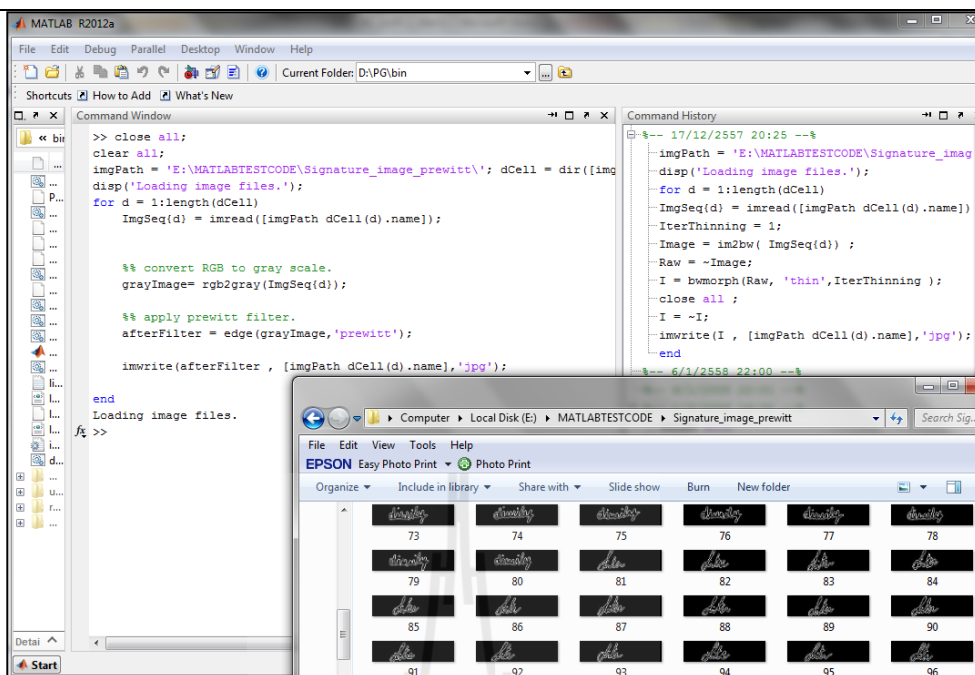
Original			
Sobel Edge Detection			
Prewitt Edge Detection			
Robert Edge Detection			
Canny Edge Detection			
Thinning			

รูปที่ ก.3 ตัวอย่างภาพลายมือชื่อดั้งเดิม และภาพที่ผ่านเทคนิคการปรับปรุงรูปภาพ

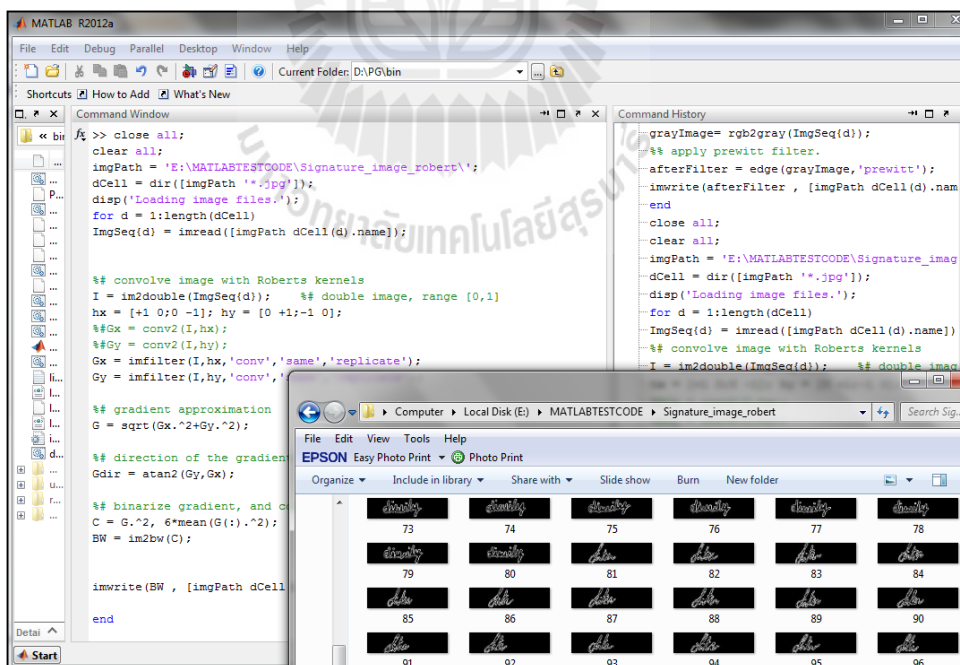
ในส่วนของการใช้งานโปรแกรม MATLAB เพื่อใช้ในการหาขอบภาพและการทำขอบภาพให้บางลงนั้น ได้แสดงไว้ในรูปภาพที่ ก.4 ก.5 ก.6 ก.7 และ ก.8



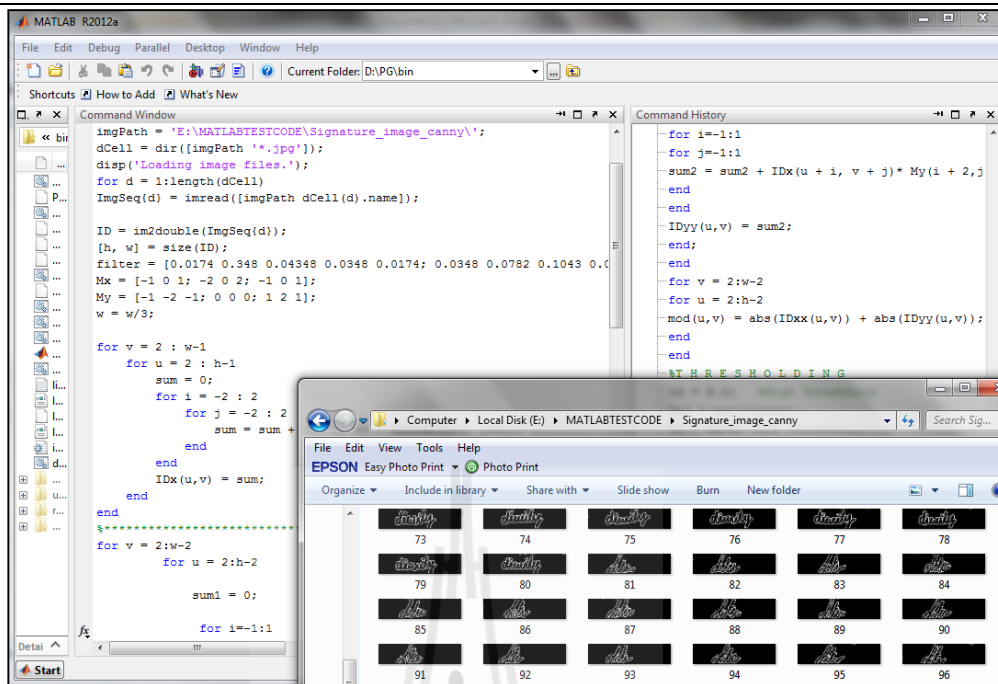
รูปที่ ก.4 ตัวอย่างการใช้โปรแกรม MATLAB เพื่อหาขอบภาพด้วยเทคนิคการตรวจหาขอบแบบโซเบล



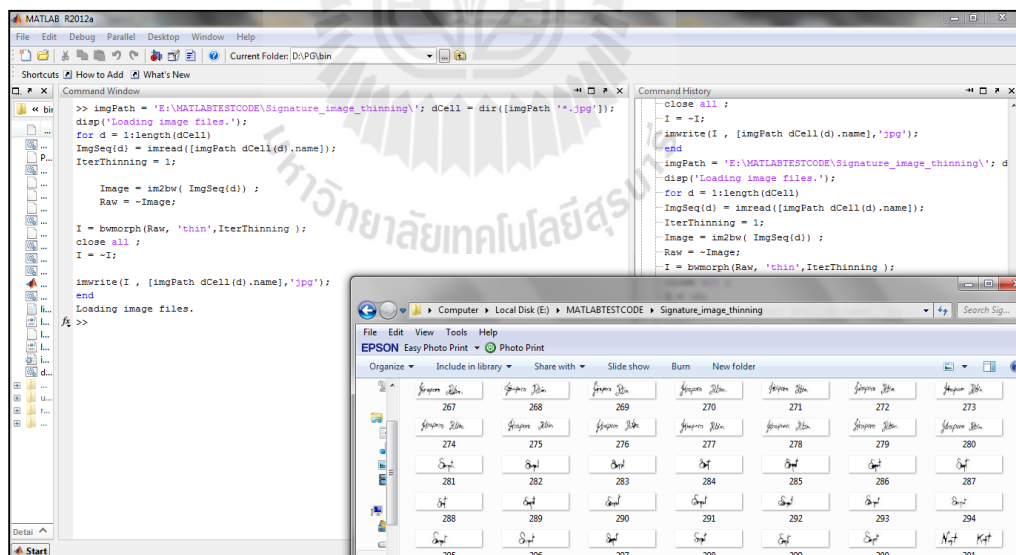
รูปที่ ก.5 ตัวอย่างการใช้โปรแกรม MATLAB เพื่อหาขอบภาพด้วย  
เทคนิคการตรวจหาขอบแบบเพรวิต



รูปที่ ก.6 ตัวอย่างการใช้โปรแกรม MATLAB เพื่อหาขอบภาพด้วย  
เทคนิคการตรวจหาขอบแบบโรเบิร์ต



รูปที่ ก.7 ตัวอย่างการใช้โปรแกรม MATLAB เพื่อหาขอบภาพด้วยเทคนิคการตรวจหาขอบแบบแคนนี่



รูปที่ ก.8 ตัวอย่างการใช้โปรแกรม MATLAB ในเทคนิคการทำขอบภาพบาง

ภาคผนวก ข

บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างการศึกษา

มหาวิทยาลัยเทคโนโลยีสุรนารี



## รายชื่อบทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างการศึกษา

- อนุสรฯ หิรัญวานากุล, รติพร จันทร์กลั่น, กิตติศักดิ์ เกิดประสพ, นิตยา เกิดประสพ. 2557. **การศึกษาเปรียบเทียบเทคนิคเพื่อเพิ่มประสิทธิภาพโมเดลจำแนกภาพ.** ในการประชุมวิชาการระดับชาติมหาวิทยาลัยเทคโนโลยีราชมงคล ครั้งที่ 6. มหาวิทยาลัยเทคโนโลยีราชมงคลสุวรรณภูมิ ศูนย์หันตรา. 23 - 25 กรกฎาคม 2557
- รติพร จันทร์กลั่น, อนุสรฯ หิรัญวานากุล, กิตติศักดิ์ เกิดประสพ, นิตยา เกิดประสพ. 2557. **การจำแนกหน้าชายหญิงโดยใช้อัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีน.** ในการประชุมวิชาการระดับชาติมหาวิทยาลัยเทคโนโลยีราชมงคล ครั้งที่ 6. มหาวิทยาลัยเทคโนโลยีราชมงคลสุวรรณภูมิ ศูนย์หันตรา. 23 - 25 กรกฎาคม 2557
- อนุสรฯ หิรัญวานากุล, กิตติศักดิ์ เกิดประสพ, นิตยา เกิดประสพ. 2557. **เทคนิคการเพิ่มประสิทธิภาพของโมเดลรู้จำภาพลายเซ็น.** ในการประชุมวิชาการระดับชาติ “มศว วิจัย” ครั้งที่ 8. มหาวิทยาลัยศรีนครินทรวิโรฒ. 26-27 พฤศจิกายน 2557
- Anusara Hirunyanakul, Kedkarn Chaiyakhan, Ratiporn Chanklan, Kittisak Kerdprasop, Nittaya Kerdprasop. 2015. **A New Efficient Method to Improve Handwritten Signature Recognition.** In Proceedings of the 3rd International Conference on Industrial Application Engineering 2015, Kitakyushu, Japan. 28-31 March 2015.
- Ratiporn Chanklan, Kedkarn Chaiyakhan, Anusara Hirunyanakul, Kittisak Kerdprasop, Nittaya Kerdprasop. 2015. **Fingerprint Recognition with Edge Detection and Dimensionality Reduction Techniques.** In Proceedings of the 3rd International Conference on Industrial Application Engineering 2015, Kitakyushu, Japan. 28-31 March 2015.
- Kedkarn Chaiyakhan, Anusara Hirunyanakul, Ratiporn Chanklan, Kittisak Kerdprasop, Nittaya Kerdprasop. 2015. **Traffic Sign Classification using Support Vector Machine and Image Segmentation.** In Proceedings of the 3rd International Conference on Industrial Application Engineering 2015, Kitakyushu, Japan. 28-31 March 2015.

การศึกษาเปรียบเทียบเทคนิคเพื่อเพิ่มประสิทธิภาพโมเดลจำแนกภาพ  
A Comparative Study of Techniques to Improve Image Classification Model

อนุสรฯ หิรัญวานากุล\*, รัตพร จันทร์กลิ่น, กิตติศักดิ์ เกิดประสพ, นิตยา เกิดประสพ  
Anusara Hirunyanakul\*, Ratiporn Chanklan, Kittisak Kerdprasop, Nittaya Kerdprasop

บทคัดย่อ

กรณีศึกษาในงานวิจัยชิ้นนี้จะเป็นการจำแนกภาพใบหน้าชาย และ หญิง ด้วยอัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีน โดยงานวิจัยชิ้นนี้ นำเสนอการศึกษาเปรียบเทียบระหว่างเทคนิคต่าง ๆ ในการนำมาเพิ่มประสิทธิภาพความแม่นยำของโมเดลจำแนกภาพ เพื่อจำแนกโดยเน้นในขั้นตอนการเตรียมข้อมูล เทคนิคที่จะนำมาทำการเปรียบเทียบ ได้แก่ การวิเคราะห์องค์ประกอบหลัก (พีซีเอ), การวิเคราะห์จำแนกประเภทเชิงเส้น (แอลดีเอ) และการคัดเลือกฟีเจอร์ บนพื้นฐานของไคสแควร์ ซึ่งผู้วิจัยได้ทำการเปรียบเทียบประสิทธิภาพระหว่างเคอร์เนลฟังก์ชันสี่ชนิด คือ ฟังก์ชันเชิงเส้น, ฟังก์ชันพหุนาม, ฟังก์ชันเรเดียลเบสิส และฟังก์ชันซิกมอยด์ จากผลการทดลองพบว่า การใช้เทคนิคพีซีเอ ที่จำนวนองค์ประกอบหลักเท่ากับ 3 ในเคอร์เนลฟังก์ชันเรเดียลเบสิส ให้ความแม่นยำสูงถึง 95% ซึ่งเป็นความแม่นยำที่สูงที่สุดในเทคนิคต่าง ๆ ในงานวิจัยชิ้นนี้ โดยเพิ่มขึ้นถึง 10% เมื่อเปรียบเทียบกับ ความแม่นยำของการจำแนกใบหน้าชาย และ หญิง โดยไม่ได้ใช้เทคนิคอื่นช่วยในขั้นตอนการเตรียมข้อมูล ซึ่งจะมีความแม่นยำอยู่เพียง 85%

**คำสำคัญ:** โมเดลจำแนกภาพ, การจำแนกใบหน้าระหว่างชาย และ หญิง, ซัพพอร์ตเวกเตอร์แมชชีน, พีซีเอ, แอลดีเอ, การคัดเลือกฟีเจอร์

Abstract

The case study in this research is about male-female image classification using Support Vector Machine (SVM). This research demonstrates a comparison between pre-processing techniques including Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Feature Selection based on Chi-square. We also compare the efficiency of image recognition and classification among four kinds of kernel functions, that is, Linear Function, Polynomial Function, Radial Basis Function, and Sigmoid Function. From the experimental results, we found that the classification accuracy of PCA technique with 3 components and trained with the RBF kernel function is 95%. This is the highest accuracy of compared to all techniques studied in this research. The increase is 10% from the baseline 85%, which is the accuracy of male-female classification without any pre-processing technique.

**Keywords:** Image recognition model, Classification between male and female faces, Support Vector Machine, PCA, LDA, Feature Selection

สาขาวิชาวิศวกรรมคอมพิวเตอร์ สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี นครราชสีมา 30000  
School of Computer Engineering, Institute of Engineering, Suranaree University of Technology, Nakhon Ratchasima, Thailand.

\*Corresponding author. E-mail: Nhingsan@gmail.com

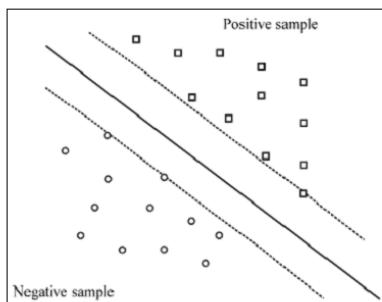
## 1. บทนำ

ในปัจจุบันข้อมูลภาพเป็นข้อมูลที่มีความละเอียดสูงจึงมีจำนวนมิติสูง ส่งผลให้เมื่อนำข้อมูลภาพไปประมวลผลด้วยคอมพิวเตอร์จะใช้เวลาประมวลผลนาน อีกทั้งยังต้องใช้หน่วยความจำสำหรับการประมวลผลที่สูงมาก งานวิจัยนี้จึงใช้การลดขนาดข้อมูลเป็นกระบวนการหนึ่งในขั้นตอนการเตรียมข้อมูลเป็นการทำให้ข้อมูลตั้งต้นมีขนาดลดลงโดยสูญเสียลักษณะสำคัญของข้อมูลน้อยที่สุดและสูญเสียความถูกต้องของผลลัพธ์น้อยที่สุด เนื่องจากข้อมูลแต่ละตัวจะมีความสำคัญต่อการจัดกลุ่มข้อมูลไม่เท่ากัน ด้วยเทคนิคการเลือกข้อมูลที่ดียิ่งจะทำให้สามารถเลือกข้อมูลที่มีความสำคัญและสามารถใช้เป็นตัวแทนของข้อมูลส่วนใหญ่ สามารถลดเวลาในการเรียนรู้ข้อมูล และลดความต้องการของหน่วยความจำ [1] ซึ่งเทคนิคการลดขนาดข้อมูลที่นำมาใช้ ได้แก่ การวิเคราะห์องค์ประกอบหลัก (พีซีเอ), การวิเคราะห์จำแนกประเภทเชิงเส้น (แอลดีเอ) และการคัดเลือกพีเจอร์ บนพื้นฐานของโคสแควร์ หลังจากนั้นจึงส่งข้อมูลภาพที่ผ่านการลดมิติแล้วเข้าประมวลผลด้วยอัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีนซึ่งเป็นอัลกอริทึมในการจำแนกที่มีการนำมาใช้กันอย่างกว้างขวางในด้านการประมวลผลภาพดิจิทัล [2] และเป็นวิธีการที่มีประสิทธิภาพในการจำแนกประเภทข้อมูล และถูกนำมาใช้ในการรู้จำรูปแบบอย่างง่าย [9] จากนั้นงานวิจัยนี้จะทำการเปรียบเทียบประสิทธิภาพระหว่างการใช้เคอร์เนลฟังก์ชันในซัพพอร์ตเวกเตอร์แมชชีนสี่ชนิด ได้แก่ ฟังก์ชันเชิงเส้น, ฟังก์ชันพหุนาม, ฟังก์ชันเรเดียลเบสิส และฟังก์ชันซิกมอยด์

วัตถุประสงค์หลักของงานวิจัยฉบับนี้ เพื่อศึกษาเปรียบเทียบเทคนิคการลดมิติเพื่อลดเวลาและเพิ่มความถูกต้องในการจำแนก โดยเน้นในขั้นตอนการเตรียมข้อมูล เทคนิคที่จะนำมาทำการเปรียบเทียบ ได้แก่ การวิเคราะห์องค์ประกอบหลัก (พีซีเอ), การวิเคราะห์จำแนกประเภทเชิงเส้น (แอลดีเอ) และการคัดเลือกพีเจอร์ บนพื้นฐานของโคสแควร์ แล้วส่งข้อมูลที่ผ่านการลดมิติเข้าประมวลผลด้วยซัพพอร์ตเวกเตอร์แมชชีนต่าง ๆ เพื่อเปรียบเทียบประสิทธิภาพความแม่นยำของโมเดลสำหรับจำแนกข้อมูล

## 2. งานวิจัยและทฤษฎีที่เกี่ยวข้อง

ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine: SVM) [3] เป็นวิธีการจำแนกกลุ่มข้อมูลที่อาศัยระนาบการตัดสินใจมาใช้ในการแบ่งข้อมูลออกเป็น 2 ส่วน โดยพยายามสร้างเส้นแบ่งกลางระหว่างกลุ่มให้มีระยะห่างระหว่างขอบเขตทั้งสองกลุ่มมากที่สุด ดังตัวอย่างในรูปที่ 1 ซึ่งซัพพอร์ตเวกเตอร์แมชชีนจะใช้ฟังก์ชันแมปปิง (Mapping Function) เพื่อแปลงข้อมูลจากโดเมนเดิมไปยังโดเมนที่เรียกว่า ฟีเจอร์สเปซ (Feature Space) และใช้ฟังก์ชันเคอร์เนล (Kernel Function) ในการวัดความคล้ายกันของข้อมูลในฟีเจอร์สเปซ ในงานวิจัยนี้ใช้เคอร์เนลฟังก์ชัน คือ ฟังก์ชันเชิงเส้น, ฟังก์ชันพหุนาม, ฟังก์ชันเรเดียลเบสิส และฟังก์ชันซิกมอยด์ในการทดสอบเปรียบเทียบประสิทธิภาพในการจำแนกข้อมูล



รูปที่ 1 ระบายการตัดสินใจของซัพพอร์ตเวกเตอร์แมชชีน (Ali, Shamsuddin and Ismail, 2011)

การวิเคราะห์องค์ประกอบหลัก (Principal Component Analysis : PCA ) [4] เป็นวิธีการที่ใช้ในการแปลงข้อมูลไปสู่ระนาบใหม่ (เรียกว่า องค์ประกอบ) ที่สามารถกระจายข้อมูลออกจากกันได้มากที่สุด โดยมีขั้นตอนการทำงานคือ 1) ทำการแปลงโครงสร้างเมตริกซ์ข้อมูลไปเป็นเวกเตอร์แถว 2) คำนวณหาไอเกนเวกเตอร์ที่สอดคล้องกันกับค่าไอเกน 3) นำไปคำนวณหาลักษณะเด่นของภาพเพื่อนำไปใช้ในการรู้จำ

ถ้ากำหนดให้ภาพที่ใช้ในการฝึกสอนมีจำนวน  $N$  ภาพ และมีขนาด  $N \times N$  โดยข้อมูลภาพจะมีรูปแบบเป็น  $X_1, X_2, \dots, X_n$  และแปลงเมตริกซ์ภาพเป็นเวกเตอร์ขนาด  $1 \times D$  โดยที่  $D = N^2$  จะทำให้ได้ชุดข้อมูลฝึกสอนดังนี้ [5]

$$\mu = \{\mu_1, \mu_2, \dots, \mu_N\} \quad (1)$$

นำข้อมูลไปคำนวณหาค่าความแปรปรวนร่วมของข้อมูลเมตริกซ์ (Covariance Matrix)

$$C = AA^T \\ = \frac{1}{N} \sum_{i=1}^N (\mu_i - \mu)(\mu_i - \mu)^T \quad (2)$$

เมื่อ  $\mu = \frac{1}{N} \sum_{i=1}^N \rho_i$  แทนค่าเฉลี่ยโดยที่  $C$  เป็นเมตริกซ์ความแปรปรวนร่วม  $\mu$  ได้มาจากภาพที่  $i$  ลบด้วยค่าเฉลี่ยภาพ  $\mu_i = \mu_i - \mu$  และนำค่า  $C$  ไปคำนวณหาไอเกนเวกเตอร์ที่สอดคล้องกับค่าไอเกนจากสมการที่(3)

$$C u = \lambda u \quad (3)$$

เรียงลำดับไอเกนเวกเตอร์ที่สอดคล้องกับค่าไอเกนจากมากไปน้อยแล้วคัดเอาเฉพาะค่าที่ไม่เท่ากับศูนย์  $U_{PCA} = [u_1, u_2, u_3, \dots, u_{N-1}]$  แล้วทำการสกัดเอาลักษณะเด่นของภาพ (Feature extraction) เพื่อนำไปใช้หาการรู้จำ จากสมการที่(4)

$$Y_k = U_d^T(u_k - \mu) \text{ โดยที่ } k = 1, 2, \dots, N \quad (4)$$

เมื่อ  $Y_k$  แทน ลักษณะเด่นภาพที่  $k$  ส่วน  $d$  เป็นค่ามิติที่เลือกระหว่าง 1 ถึง  $N-1$  และ  $k$  เป็นจำนวนภาพ จากทั้งหมด  $N$  ภาพ

การวิเคราะห์จำแนกประเภทเชิงเส้น (Linear Discriminant Analysis : LDA ) [6] เป็นเทคนิคที่ใช้สำหรับการเรียนรู้แบบมีผู้สอน ใช้เทคนิคการหากลุ่มของปริมาณ (Base vector) โดย  $W_k$  เป็นเวกเตอร์ของกลุ่มตัวอย่าง สามารถแสดงสมการ  $W_k$  ได้ดังนี้

$$\begin{aligned} W_{opt} &= \frac{\text{argmax}_W \frac{|W^T S_C W|}{|W^T S_V W|}}{W} \\ &= [W_1, W_2, \dots, W_L] \end{aligned} \quad (5)$$

โดยที่  $\{W_X | 1 \leq K \leq L\}$  กำหนดให้  $L$  เป็นมิติของชุดข้อมูล  $S_C$  และ  $S_V$  เป็นเมตริกซ์ที่อยู่ใน class สามารถคำนวณหาเมตริกซ์  $S_C$  และ  $S_V$  ได้จากสมการที่(6) และสมการที่(7)

$$S_C = \sum_{k=1}^a M_k (\mu_k - \mu)(\mu_k - \mu)^T \quad (6)$$

$$S_V = \sum_{k=1}^a \sum_{(x_u = x_k)} (x_u - \mu_k)(x_u - \mu_k)^T \quad (7)$$

เมื่อ  $a$  แทน จำนวน class และ  $X \in R^n$  เป็นข้อมูล

$x_k$  แทน ข้อมูลทดสอบของ class  $k$

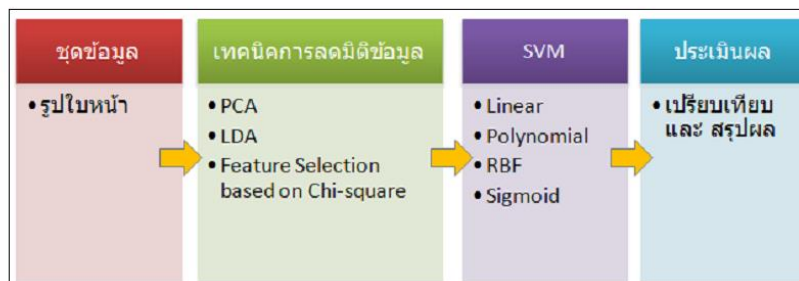
$\mu_k$  แทน ข้อมูลทั้งหมดของ class  $k$

$M_k$  คือ จำนวนข้อมูลที่ใช้ทดสอบใน class  $k$  ถ้า  $S_V$  เป็น non-singular ที่ใช้ค้นหา  $W_k$  จากสมการที่(5) ซึ่ง  $L$  แรกจะมีขนาดใหญ่กว่าไอเกน  $\{W_X | 1 \leq K \leq L\}$

**วิธีการคัดเลือกคุณลักษณะ (Feature Selection Methods)** [7] การคัดเลือกคุณลักษณะเป็นกระบวนการที่เลือกสับเซตจากเซตของคุณลักษณะ (Feature Set) ของต้นฉบับ จะทำให้ได้คุณลักษณะที่เหมาะสมเพื่อนำไปใช้ในการจำแนกหมวดหมู่ โดยที่เซตของคุณลักษณะ คือ เซตของ term ที่เกิดขึ้นในข้อมูลทั้งหมด ซึ่งวิธีการคัดเลือกคุณลักษณะนี้จะช่วยเพิ่มประสิทธิภาพในการจำแนกข้อมูลและหลีกเลี่ยงการเกิด overfitting ได้ ซึ่งงานวิจัยนี้ได้เลือกใช้วิธีการคัดเลือกคุณลักษณะบนพื้นฐานของไคสแควร์ (Chi Squared : CHI) [8] เป็นการวัดความสัมพันธ์ระหว่าง term และ category ซึ่งคำนวณได้จากสมการดังนี้

$$X^2(t, c) = \frac{N \times (p(t,c) \times p(\bar{t}, \bar{c}) - p(t, \bar{c}) \times p(\bar{t}, c))^2}{p(t) \times p(\bar{t}) \times p(c) \times p(\bar{c})} \quad (8)$$





รูปที่ 3 โครงร่างขั้นตอนการดำเนินงาน

#### 4. ผลการทดลอง

จากข้อมูลรูปใบหน้าคนทั้งหมดจำนวน 400 รูป การจำแนกใบหน้าชายและหญิง โดยใช้ภาษา Python 2.7 เขียนโปรแกรม Spyder จะได้ผลการทดลองทั้งหมดแสดงได้ดังตารางที่ 1

ตารางที่ 1 แสดงค่าความแม่นยำ ของแต่ละเทคนิคในการลดมิติข้อมูล

Accuracy of testing datasets									
		PCA				LDA			
SVM (kernel)		Linear	Poly	Rbf	Sigmoid	Linear	Poly	Rbf	Sigmoid
n-component	1	0.40	0.45	0.50	0.50	0.70	0.60	0.70	0.50
	3	0.25	0.65	0.95	0.50	0.70	0.60	0.70	0.50
	5	0.80	0.85	0.95	0.50	0.70	0.60	0.70	0.50
	7	0.85	0.80	0.90	0.50	0.70	0.60	0.70	0.50
	9	0.90	0.80	0.85	0.50	0.70	0.60	0.70	0.50
	10	0.85	0.85	0.90	0.50	0.70	0.60	0.70	0.50
	15	0.70	0.85	0.95	0.50	0.70	0.60	0.70	0.50
	20	0.75	0.75	0.50	0.50	0.70	0.60	0.70	0.50
	25	0.80	0.80	0.50	0.50	0.70	0.60	0.70	0.50
	30	0.80	0.80	0.50	0.50	0.70	0.60	0.70	0.50
	35	0.80	0.80	0.50	0.50	0.70	0.60	0.70	0.50
	40	0.75	0.75	0.50	0.50	0.70	0.60	0.70	0.50
	45	0.80	0.80	0.50	0.50	0.70	0.60	0.70	0.50
	50	0.75	0.75	0.50	0.50	0.70	0.60	0.70	0.50
	55	0.75	0.75	0.50	0.50	0.70	0.60	0.70	0.50
	60	0.75	0.75	0.50	0.50	0.70	0.60	0.70	0.50
	80	0.80	0.80	0.50	0.50	0.70	0.60	0.70	0.50
	90	0.80	0.80	0.50	0.50	0.70	0.60	0.70	0.50
	100	0.80	0.80	0.50	0.50	0.70	0.60	0.70	0.50
10%	410	0.80	0.80	0.50	0.50	0.70	0.60	0.70	0.50
25%	1025	0.80	0.80	0.50	0.50	0.70	0.60	0.70	0.50
50%	2025	0.80	0.80	0.50	0.50	0.70	0.60	0.70	0.50
75%	3075	0.80	0.80	0.50	0.50	0.70	0.60	0.70	0.50
100%	4096	0.80	0.80	0.45	0.50	0.70	0.60	0.70	0.50

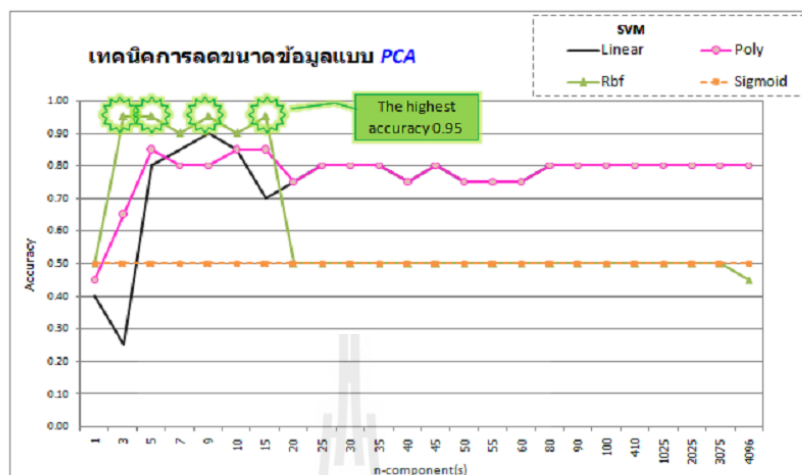
Feature Selection					
SVM (kernel)	Linear	Poly	Rbf	Sigmoid	
percentile	1	0.35	0.45	0.50	0.50
	3	0.55	0.45	0.50	0.50
	5	0.50	0.45	0.50	0.50
	7	0.45	0.45	0.50	0.50
	9	0.35	0.45	0.55	0.50
	10	0.40	0.45	0.55	0.50
	15	0.40	0.45	0.45	0.50
	20	0.35	0.45	0.45	0.50
	25	0.45	0.45	0.50	0.50
	30	0.60	0.50	0.55	0.50
	35	0.75	0.50	0.55	0.50
	40	0.65	0.50	0.55	0.50
	45	0.80	0.55	0.55	0.50
	50	0.80	0.55	0.55	0.50
	55	0.85	0.55	0.55	0.50
	60	0.85	0.55	0.55	0.50
	65	0.85	0.55	0.55	0.50
	70	0.85	0.55	0.60	0.50
	75	0.85	0.55	0.60	0.50
	80	0.85	0.55	0.60	0.50
	85	0.85	0.55	0.60	0.50
	90	0.85	0.55	0.60	0.50
	95	0.85	0.55	0.60	0.50
	100	0.85	0.55	0.60	0.50

ด้วยเทคนิคการลดมิติข้อมูลแบบ PCA ดังแสดงผลในรูปที่ 4 จะสังเกตได้ว่าที่ kernel Linear จะมีค่า accuracy แกว่งในช่วงจำนวน component 1 ถึง 15 หลังจากนั้นค่าความแม่นยำจะเริ่มคงที่ ที่ประมาณ 0.8

ที่ kernel Rbf ค่า accuracy เมื่อจำนวน component = 3, 5, 9 และ 15 สูงถึง 0.95

ที่ kernel Poly จะมีค่า accuracy แกว่งในช่วง component 1 ถึง 15 หลังจากนั้นค่าความแม่นยำจะเริ่มคงที่ ที่ประมาณ 0.8

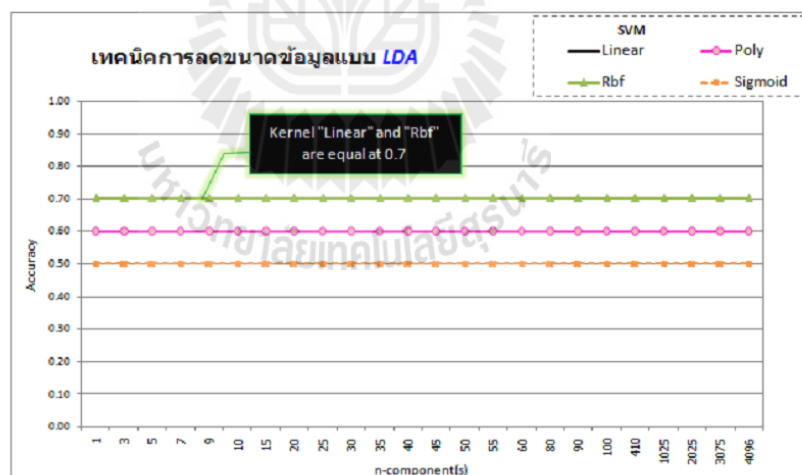
ที่ kernel sigmoid ค่า accuracy จะคงที่ ที่ 0.5 ไม่ว่าจะมิจำนวน component จะเป็นเท่าใด



รูปที่ 4 กราฟแสดงการเปรียบเทียบ ความแม่นยำระหว่างแต่ละ kernel ในเทคนิคลดมิติด้วย PCA

ด้วยเทคนิคการลดมิติข้อมูลแบบ LDA ดังแสดงผลในรูปที่ 5 ค่า accuracy จะคงที่ โดยจะแตกต่างกันออกไปตามแต่ละ kernel โดยค่าที่ได้ มีดังต่อไปนี้ ที่ kernel Linear และ Rbf จะมีความแม่นยำเท่ากันอยู่ที่ 0.7 และคงที่ ไม่ว่าจำนวน component จะเปลี่ยนแปลงไปมากเท่าไร

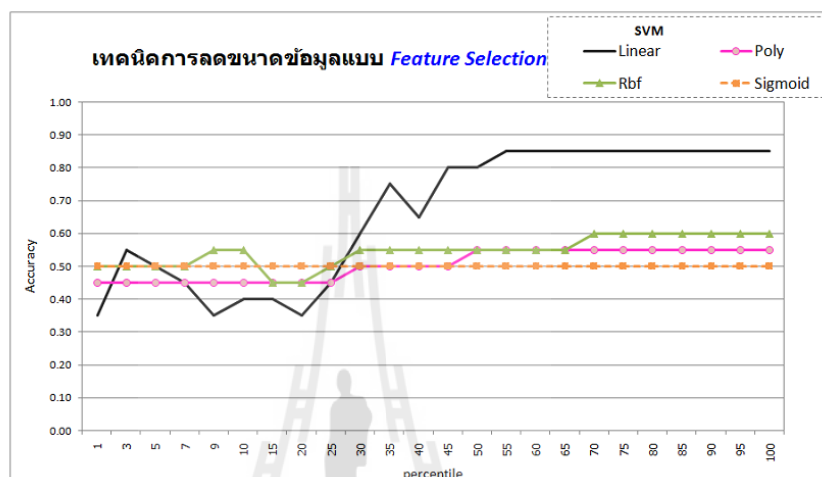
ที่ kernel Poly ค่า accuracy จะคงที่อยู่ที่ 0.6 และ ที่ kernel Sigmoid ค่า accuracy คงที่อยู่ที่ 0.5 (ซึ่งจะเท่ากับ เทคนิคการลดมิติข้อมูลแบบ PCA)



รูปที่ 5 กราฟแสดงการเปรียบเทียบ ความแม่นยำระหว่างแต่ละ kernel ในเทคนิคลดมิติด้วย LDA



ในส่วนของเทคนิคการลดมิติข้อมูลด้วยวิธี Feature Selection บนพื้นฐานของโคสแควร์ ดังรูปที่ 6 พบว่าที่ Kernel Linear ค่า accuracy นั้นจะสูงขึ้นเรื่อย ๆ ตาม percentile ของจำนวนมิติข้อมูล อย่างเห็นได้ชัดเจน และ สูงสุดที่ 0.85 โดยจะเริ่มคงที่ เมื่อ percentile=55 ส่วนใน Kernel อื่น ๆ ค่า accuracy จะเพิ่มขึ้นเล็กน้อยตาม percentile



รูปที่ 6 กราฟแสดงการเปรียบเทียบ ความแม่นยำระหว่าง Kernel ต่างๆ ในเทคนิค Feature Selection บนพื้นฐานโคสแควร์

#### อภิปรายผล

ผลการทดลองนี้แสดงให้เห็นถึงความสำคัญในการลดมิติข้อมูล โดยในแต่ละเทคนิค จะให้ผลลัพธ์ต่างกันออกไป โดยในแต่ละ kernel จะมีค่าความแม่นยำเปลี่ยนไปตาม จำนวน component (หรือ percentile ในกรณีของ การลดมิติแบบ Feature Selection บนพื้นฐานโคสแควร์)

#### 5. สรุปผลงานวิจัย

งานวิจัยนี้มุ่งเน้นไปในด้านการเพิ่มประสิทธิภาพการจำแนกรูปใบหน้าระหว่างชาย-หญิง โดยทางผู้วิจัยจะทำการศึกษาเปรียบเทียบ ระหว่างแต่ละเทคนิคของการลดขนาดมิติของข้อมูลได้แก่ PCA, LDA และ Feature Selection บนพื้นฐานโคสแควร์ โดยรวมไปถึงการศึกษาเปรียบเทียบ ระหว่างแต่ละ kernel ของ SVM ด้วย

จากการทดลองพบว่า การใช้เทคนิคการลดขนาดมิติข้อมูลแบบ PCA โดยใช้ kernel SVM เป็น RBF จะให้ค่าความแม่นยำสูงถึง 0.95 ที่ จำนวน component = 3, 5, 9 และ 15 ซึ่งเป็นค่าที่สูงที่สุดใน การทดลองนี้ เมื่อเปรียบเทียบกับ ความแม่นยำของการจำแนกใบหน้าชาย และ หญิง โดยไม่ได้ใช้เทคนิคอื่นช่วยในขั้นตอนการเตรียมข้อมูล ซึ่งจะมีค่าความแม่นยำอยู่เพียง 0.85

#### 6.เอกสารอ้างอิง

- [1] ธรรมศักดิ์ เขียรนิเวศน. “การลดขนาดข้อมูลด้วยน้ำหนักความหนาแน่นเพื่อการจัดกลุ่มข้อมูลขนาดใหญ่”, วิทยานิพนธ์วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยเทคโนโลยีสุรนารี ปการศึกษา 2548.
- [2] พรพล ธรรมรงค์รัตน์, ลัดดา ปรีชาวีรกุล, และวิภาดา เวทย์ประสิทธิ์. “การจำแนกประเภทเว็บเพจโดยใช้ค่าความถี่เอกสารและซอฟต์แวร์เวกเตอร์แมชชีน”, The 12th National Computer Science and Engineering Conference, 2008.
- [3] Ali, W., Shamsuddin, S.M., and Ismail A.S. (2011). “Web Proxy Cache Content Classification based on Support Vector Machine”. *Journal of Artificial Intelligence* 4(1): 100-109.
- [4] Dinh Minh Nhat, V., and Young Lee, S. (2006). “Two dimensional Weighted PCA algorithm for Face Recognition”. *IEEE International Symposium on Computational Intelligence in Robotics and Automation*. Page 219-223.
- [5] Turk, M., and Pentland, A. (1991). “Eigenfaces for recognition”. *Journal of Cognitive Neuroscience*. 3: Page 71-86, from <http://www.sciencedirect.com>
- [6] Tang, F., and Tao, H. (2006) “Fast linear discriminant analysis using binary bases”. *Proceedings of the 18<sup>th</sup> International Conference on Pattern Recognition (ICPR'06)*, Page 52-55.
- [7] Yang Y. and Pedersen J.O., “A Comparative Study on Feature Selection in Text Categorization,” *Proceedings of The Fourteenth International Conference on Machine Learning (ICML'97)*, Page 412-420.
- [8] Galavotti, L., Sebastiani, F. and Simi, M. (2000) “Feature selection and negative evidence in automated text categorization”, *Proceedings of KDD-00*.
- [9] Cristianini N. and Shawe-Taylor J. (2000) “An Introduction to Support Vector Machines and Other Kernel-based Learning Methods”, U.K. Cambridge University Press.

การจำแนกหน้าชายหญิงโดยใช้อัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีน  
Male-Female Face Classification Using Support Vector Machine Algorithm

รติพร จันทร์กลิ่น<sup>\*</sup>, อนุสรา หิรัญวานากุล, กิตติศักดิ์ เกิดประสพ, นิตยา เกิดประสพ  
Ratiporn Chanklan, Anusara Hirunyanakul, Kittisak Kerdprasop, Nittaya Kerdprasop

**บทคัดย่อ**

งานวิจัยนี้นำเสนอวิธีการจำแนกรูปใบหน้าระหว่างชาย-หญิง ด้วยอัลกอริทึม ซัพพอร์ตเวกเตอร์แมชชีนโดยงานวิจัยจะมุ่งเน้น การปรับปรุงโมเดลเพื่อให้ตรวจจับรูปใบหน้าที่ตรวจจับได้ยากและถ้ารูปบางรูปมีจำนวนน้อย จะต้องสามารถตรวจจับได้ถูกต้องมากที่สุด เทคนิคการปรับปรุงโมเดล เพื่อให้การจำแนกรูปใบหน้าที่มีประสิทธิภาพสูงขึ้นประกอบด้วย 3 วิธี ได้แก่ วิธีการปรับอัตราส่วนระหว่างหน้าชาย-หญิงให้สมดุล, วิธีการคัดเลือกใบหน้าผู้หญิงที่เด่นชัดเพื่อใช้ในโมเดล และ วิธีการคัดเลือกใบหน้าผู้หญิงที่ไม่เด่นชัดเพื่อใช้ในโมเดล การทดสอบประสิทธิภาพจะใช้วิธีการเปรียบเทียบความแม่นยำในการจำแนก โดยเทียบกับวิธีการดั้งเดิมที่ไม่มีการปรับปรุงโมเดล ผลการทดลองพบว่าการปรับปรุงโมเดลด้วยวิธีการคัดเลือกใบหน้าผู้หญิงที่เด่นชัดเพื่อใช้ในโมเดลให้ความแม่นยำเพิ่มขึ้นจาก 50% เป็น 85%

**คำสำคัญ:** การจำแนกใบหน้าระหว่างชายและหญิง, ซัพพอร์ตเวกเตอร์แมชชีน, ภาษาไพธอน

**Abstract**

This research demonstrates a method of classification between male and female faces using support vector machine algorithm. It focuses on improving the model to detect faces that are difficult to detect and if the number of some pictures is limited, the method is able to classify faces with high accuracy. The proposed techniques to improve classification accuracy are balancing male-female ratio, using obvious female face shape, and using unobvious female shape. To evaluate efficiency of the proposed techniques, we compare classification accuracy of the models obtained from the proposed three techniques against the traditional face classification method. From the experimental results, we found that the use of obvious female face shape can improve model accuracy from 50% to 85%.

**Keywords:** classification between male and female faces, Support Vector Machine, Python language

สาขาวิชาวิศวกรรมคอมพิวเตอร์ สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี นครราชสีมา 30000  
School of Computer Engineering, Institute of Engineering, Suranaree University of Technology,  
Nakhon Ratchasima, Thailand.

<sup>\*</sup>Corresponding author. E-mail: mootigger@gmail.com

## 1. บทนำ

ในปัจจุบันรูปภาพดิจิทัลได้เพิ่มปริมาณขึ้นอย่างรวดเร็วด้วยเทคโนโลยีทางการถ่ายภาพนั้นได้พัฒนาขึ้นอย่างต่อเนื่อง และหลากหลาย รูปถ่ายเหล่านี้จะถูกเก็บไว้อย่างกระจัดกระจาย ดังนั้นการจัดกลุ่มหรือจำแนกนั้นจะเป็นไปได้ยากเนื่องจากโดยปกติแล้วคอมพิวเตอร์ไม่สามารถมองวัตถุแล้วสามารถบอกได้ว่าวัตถุนั้นคืออะไร ซึ่งแตกต่างจากมนุษย์ที่มองรูปภาพแล้วสามารถบอกได้ทันที ดังนั้นจึงได้มีการนำวิธีการ และ โมเดลต่าง ๆ ในเชิงวิชาการเข้ามาช่วยในการแก้ปัญหาดังกล่าว

ระบบรู้จำใบหน้าถูกออกแบบมาให้ทำการเปรียบเทียบใบหน้าบุคคลที่ตรวจจับมาทำการประมวลผลแล้วนำมาเปรียบเทียบกับฐานข้อมูล ซึ่งถือว่าเป็นเทคนิคการจำแนกรูปภาพที่สามารถนำมาใช้ให้เกิดประโยชน์ในชีวิตประจำวันได้ เช่น “การจำแนกใบหน้าระหว่างคนที่ยิ้มและไม่ยิ้ม” เพื่อใช้ประโยชน์ในการตั้งโปรแกรมถ่ายรูปแบบอัตโนมัติทั้งยังมีกรนำ “การจำแนกใบหน้าของแต่ละบุคคล” เพื่อระบุตัวตนในการ Log-on อุปกรณ์อิเล็กทรอนิกส์ส่วนตัว เช่น คอมพิวเตอร์ และ โทรศัพท์มือถือ ซึ่งทางผู้วิจัยได้เล็งเห็นถึงประโยชน์ในการจำแนก “ใบหน้าระหว่างชายและหญิง” เช่นการแต่งรูปภาพเฉพาะใบหน้าของเพศใดเพศหนึ่งในรูปหมู่หรือการนำไปใช้ช่วยสืบสวนคดีอาชญากรรมต่าง ๆ ซึ่งงานวิจัยชิ้นนี้ได้นำเสนอเทคนิคการเพิ่มประสิทธิภาพโมเดลเพื่อการจำแนกใบหน้าจากภาพถ่ายของหญิงและชาย ซึ่งจากการค้นคว้างานวิจัยที่เกี่ยวข้องยังไม่พบงานวิจัยในด้านนี้โดยตรง ทางผู้วิจัยจึงนำเสนอเทคนิคการจำแนกภาพถ่ายของหญิงและชาย ด้วยวิธีซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine) ซึ่งเป็นอัลกอริทึมในการคัดแยกที่มีความชัดเจนในการแบ่งมากและมีการนำมาใช้กันอย่างกว้างขวางในด้านการประมวลผลและการจำแนกภาพดิจิทัล โดยมุ่งเน้นการปรับปรุงโมเดลเพื่อให้ความถูกต้องมากขึ้น

### วัตถุประสงค์ของงานวิจัย

งานวิจัยชิ้นนี้มีวัตถุประสงค์เพื่อศึกษาการจำแนกใบหน้าระหว่างชาย-หญิงด้วย เทคนิค Support Vector Machine (SVM) เพื่อที่จะพยายามตรวจจับรูปใบหน้าที่มีความคล้ายคลึงกันทำให้ตรวจจับยากให้ตรวจจับได้ถูกต้อง และถ้ารูปบางรูปมีจำนวนน้อยทำให้มีการตรวจจับได้ยาก ซึ่งผู้วิจัยพยายามจะแก้ไขปัญหาโดยจะมุ่งเน้นไปที่การปรับปรุงโมเดลด้วยวิธีการต่างๆ เช่น การปรับอัตราส่วนระหว่างจำนวนใบหน้าของผู้ชาย และ ผู้หญิง หรือการเลือกใบหน้าผู้หญิงที่มีความเด่นชัดเพื่อใช้ในการสร้างโมเดลเพื่อให้โมเดลมีความแม่นยำมากยิ่งขึ้น

## 2. งานวิจัยและทฤษฎีที่เกี่ยวข้อง

### 2.1 การรู้จำใบหน้า (Face recognition)

ระบบรู้จำใบหน้า[3] ถูกออกแบบมาให้ทำการเปรียบเทียบใบหน้าบุคคลที่ตรวจจับได้และประมวลผลแล้วจากขั้นตอนการตรวจ มาเปรียบเทียบกับฐานข้อมูลใบหน้าที่มีอยู่เพื่อระบุว่าใบหน้าที่ตรวจจับได้ตรงกับบุคคลใดโดยอัลกอริทึมที่ใช้ในขั้นตอนการสร้างแม่แบบและขั้นตอนการเปรียบเทียบอาจแตกต่างกันไปแล้วแต่การออกแบบระบบของแต่ละระบบ

ระบบรู้จำใบหน้าจะประกอบไปด้วย 2 ขั้นตอนคือ

2.1.1 การตรวจจับใบหน้า (Face Detection) คือกระบวนการค้นหาใบหน้าของบุคคลจากภาพหรือวิดีโอ หลังจากนั้นก็จะทำการประมวลผลภาพใบหน้าที่ได้สำหรับขั้นตอนถัดไปเพื่อให้ภาพใบหน้าที่ตรวจจับได้

ง่ายต่อการจำแนก ในขั้นตอนการตรวจจับใบหน้าที่ดี ต้องมีความแม่นยำและรวดเร็วในการค้นหารูป  
ใบหน้าในระบบโดยจะอาศัย อัลกอริทึมต่าง ๆ มาช่วยในการแบ่งแยกและจำแนกใบหน้า

2.1.2 การรู้จำใบหน้า (Face Recognition) คือกระบวนการที่ได้นำภาพใบหน้าที่ตรวจจับได้และนำไป  
ประมวลผลแล้วจากขั้นตอนการตรวจจับใบหน้ามาเปรียบเทียบกับฐานข้อมูลของใบหน้าเพื่อระบุว่าใบหน้า  
ที่ตรวจจับได้ตรงกับบุคคลใด

### 2.2 อัลกอริทึมการจำแนกประเภท (Classification Algorithm)

การจำแนกข้อมูลจัดเป็นการทำเหมืองข้อมูล ที่สำคัญเทคนิคหนึ่ง เป็นกระบวนการสร้างโมเดล  
จัดการข้อมูลให้อยู่ในกลุ่มที่กำหนดมาให้ โดยการสร้างกฎ หรือโมเดลในรูปแบบอื่น ๆ เพื่อช่วยในการ  
ตัดสินใจจากข้อมูลที่มีอยู่ เพื่อให้ทำนายแนวโน้มการเกิดขึ้นของข้อมูลที่ยังไม่เกิดขึ้น การจำแนกข้อมูล  
ประเภทเรียนรู้แบบมีผลเฉลย (Supervised Learning) สามารถแบ่งขั้นตอนวิธีการจำแนกประเภทได้เป็น  
2 ขั้นตอนคือ การเรียนรู้เพื่อสร้างกลุ่มต้นแบบและ จำแนกประเภทของกลุ่มตัวอย่างที่สนใจ โดยการ  
ตรวจสอบหาความคล้ายกับกลุ่มตัวอย่างต้นแบบ

งานวิจัยนี้ศึกษากระบวนการรู้จำใบหน้า โดยใช้ซัพพอร์ตเวกเตอร์แมชชีน หรือ SVM เป็นอัลกอริทึมหลัก  
ในการจำแนกประเภทใบหน้าบุคคล เป็นอัลกอริทึมในการตัดแยกที่มีการนำมาใช้กันอย่างกว้างขวางใน  
ด้านการประมวลผลเป็นภาพดิจิทัล [2], [4] แนวคิดหลักของวิธีการนี้ใช้เพื่อหาระนาบการตัดสินใจในการ  
แบ่งข้อมูลออกเป็นสองกลุ่ม โดยจะพยายามสร้างเส้นแบ่งตรงกึ่งกลางระหว่างกลุ่มให้มีระยะห่างระหว่าง  
ขอบเขตของทั้งสองกลุ่มมากที่สุด SVM จะใช้ฟังก์ชันแมปสำหรับย้ายข้อมูลจาก Input Space ไปยัง  
Feature Space และสร้างฟังก์ชันวัดความคล้ายที่เรียกว่าเคอร์เนลฟังก์ชัน (Kernel Function) บน  
Feature Space เหมาะใช้สำหรับข้อมูลที่มีมิติของข้อมูลสูง

กำหนดให้  $(x_1, y_1), \dots, (x_n, y_n)$  เป็นข้อมูลตัวอย่างที่ใช้สำหรับการสอน  $n$  คือจำนวนข้อมูลตัวอย่าง  
 $m$  คือจำนวนมิติข้อมูลเข้าและ  $y$  คือผลลัพธ์มีค่า +1 หรือ -1  
เมื่อ +1 แทนกลุ่มของสมาชิกจำนวนเต็มบวก และ -1 แทนกลุ่มของสมาชิกจำนวนเต็มลบ ดังสมการ  
ต่อไปนี้

$$(x_i, y_i), \dots, (x_n, y_n) \text{ เมื่อ } x \in R^m, y \in \{+1, -1\}$$

สำหรับปัญหาเชิงเส้น 2 มิติ สามารถคำนวณได้ ดังสมการ

$$(w * x) + b$$

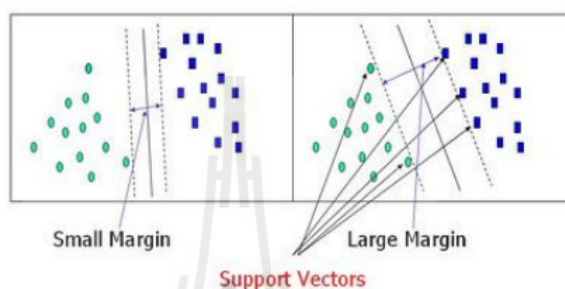
เมื่อ  $w$  คือค่าน้ำหนักและ  $b$  คือค่า bias สมการใช้สำหรับจำแนกประเภทของข้อมูล ได้แก่

$$(w * x) + b > 0 \text{ ถ้า } y_i = +1 \text{ และ } (w * x) + b < 0 \text{ ถ้า } y_i = -1$$

โดย SVM มีเคอร์เนลฟังก์ชัน ให้ผู้ใช้สามารถประยุกต์ใช้ในการแก้ปัญหาได้หลายฟังก์ชัน เช่น Linear,  
Polynomial, Radial Basis Function โดยงานวิจัยนี้ได้ใช้ Linear Kernel ในการทดลอง

ยกตัวอย่างง่าย ๆ จากรูปที่ 1 หากเราต้องการตัดแยกข้อมูลออกเป็น 2 กลุ่ม กลุ่มที่เป็นวงกลม  
และกลุ่มที่เป็นสี่เหลี่ยม โดยใช้เส้นแบ่งที่เป็นเส้นตรง จะพบว่าไม่มีเส้นตรงจำนวนมากที่สามารถตัดแยกกลุ่ม  
ทั้งสองออกจากกันได้ เราจะนิยาม Margin เป็นผลรวมระยะห่างของเส้นตรงที่เป็นเส้นแบ่ง ถึงเส้นตรงที่

ผ่านข้อมูลที่ใกล้ที่สุดและขนานกับเส้นแบ่งของทั้งสองกลุ่ม ซึ่งหากเราใช้โมเดลอื่นในการแบ่งกลุ่มเราอาจได้รูปทางฝั่งซ้ายซึ่งสามารถแบ่งกลุ่มได้ก็จริง แต่จะมีระยะขอบที่ค่อนข้างแคบ (Small margin) แต่หากเราเลือกใช้โมเดลซัพพอร์ตเวกเตอร์แมชชีนในการแบ่งกลุ่ม เราจะได้รูปทางฝั่งขวา ซึ่งนอกจากจะแบ่งกลุ่มทั้งสองได้แล้วนั้น โมเดลยังทำให้เกิดระยะขอบที่กว้างขึ้นอีกด้วย (Large margin)



รูปที่ 1 ตัวอย่าง SVM ในสองมิติ

แหล่งที่มา: <http://www.dtreg.com/svm.htm>

### 2.3 Confusion Matrix

Confusion Matrix [1] คือการประเมินผลลัพธ์การทำนายของโมเดล หรือผลลัพธ์จากโปรแกรม (Predicted Labels) เปรียบเทียบกับผลลัพธ์จริงๆ ที่ทำโดยคน หรือจากผลเฉลยที่ทราบค่าที่แท้จริงอยู่ก่อนแล้ว (Actual Labels)

ในกรณีที่ข้อมูลจำแนกประเภทได้เป็น 2 กลุ่มคือ Positive และ Negative ค่าที่ได้จาก Confusion Matrix จะมี 4 ค่า (รูปที่ 2) ได้แก่

True Positive (TP) คือ สิ่งที่โปรแกรมทำนายว่าจริงและคนบอกว่ามันจริง ถือว่าเป็นการทำนายถูก

True Negative (TN) คือ สิ่งที่โปรแกรมทำนายว่าไม่จริงและคนบอกว่ามันไม่จริง ถือว่าเป็นการทำนายถูก

False Positive (FP) คือ สิ่งที่โปรแกรมทำนายว่าจริงแต่คนบอกว่ามันไม่จริง ถือว่าเป็นการทำนายผิด

False Negative (FN) คือ สิ่งที่โปรแกรมทำนายว่าไม่จริงแต่คนบอกว่ามันจริง ถือว่าเป็นการทำนายผิด

		Predicted Labels :	
		True	False
Actual Labels :	True	TP	FN
	False	FP	TN

รูปที่ 2 ตาราง Confusion Matrix

ในงานวิจัยครั้งนี้จะใช้มาตรวัดที่คำนวณได้จาก Confusion Matrix ด้วยกัน 3 มาตรวัด ได้แก่ Accuracy, Recall และ Precision

2.3.1 Accuracy คือ ค่าความแม่นยำที่บอกว่าโปรแกรมทำนาย สามารถสร้างโมเดลที่ใช้ทำนายข้อมูลได้ถูกต้องขนาดไหน

$$\text{ค่าความแม่นยำคำนวณได้จาก} \quad (TP + TN) / (TP + TN + TP + FN)$$

2.3.2 Recall (True Positive Rate) คือ ค่าระลึที่บอกว่าโปรแกรมทำนายได้ว่าข้อมูลเป็นประเภท positive จริง เป็นอัตราส่วนเท่าไรของข้อมูลจริงในกลุ่ม positive ทั้งหมด

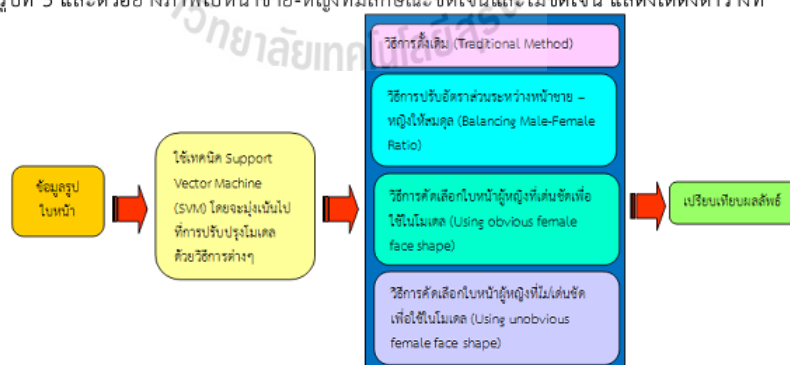
$$\text{ค่าระลึนี้ คำนวณได้จาก} \quad TP / (TP + FN)$$

2.3.3 Precision คือ ค่าความเที่ยงที่บอกว่าโปรแกรมทำนายว่าข้อมูลเป็นประเภท positive ได้เที่ยงตรงมากน้อยเพียงใด โดยการทำนายผิดที่ระบุว่าข้อมูลเป็น positive ทั้งที่ข้อมูลนั้นมีประเภทที่แท้จริงเป็น negative จะส่งผลให้ค่าความเที่ยงของการทำนายลดลง

$$\text{ค่าความเที่ยงนี้ คำนวณได้จาก} \quad TP / (TP + FP)$$

### 3. ขั้นตอนดำเนินงานวิจัย

งานวิจัยครั้งนี้ ใช้ฐานข้อมูลที่เป็นรูปใบหน้าคนจาก <http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html> จำนวน 400 รูป เป็นรูปใบหน้าผู้ชายจำนวน 360 รูปและ 40 รูปที่เหลือเป็นรูปใบหน้าผู้หญิง ซึ่งทั้งหมด 400 รูปนี้มีความละเอียด 64x64 pixels (4096 pixels) โมเดลจำแนกใบหน้าในงานวิจัยนี้ จะทำขึ้นเพื่อแบ่งแยกรูปใบหน้าระหว่างชาย-หญิง โดยใช้อัลกอริทึม SVM และเรียกโมเดลที่ได้จากอัลกอริทึม SVM นี้ว่าโมเดลด้วยวิธีการดั้งเดิม (Traditional Method) จากนั้นผู้วิจัยทำการปรับปรุงคุณภาพการจำแนกของโมเดลด้วยวิธีการต่าง ๆ 3 วิธีคือ วิธีการปรับอัตราส่วนระหว่างหน้าชาย-หญิงให้สมดุล (Balancing Male-Female Ratio), วิธีการคัดเลือกใบหน้าผู้หญิงที่เด่นชัดเพื่อใช้ในโมเดล (Using obvious female face shape) และ วิธีการคัดเลือกใบหน้าผู้หญิงที่ไม่เด่นชัดเพื่อใช้ในโมเดล (Using unobvious female face shape) แผนภาพการดำเนินงานแสดงได้ ดังรูปที่ 3 และตัวอย่างภาพใบหน้าชาย-หญิงที่มีลักษณะชัดเจนและไม่ชัดเจน แสดงได้ดังตารางที่ 1



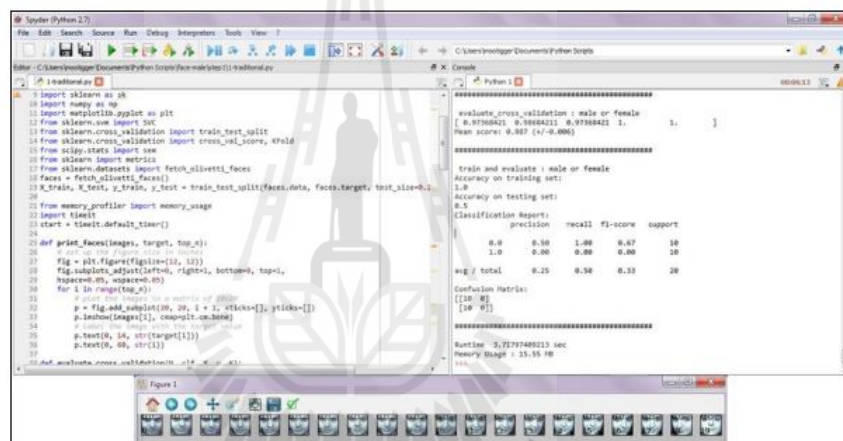
รูปที่ 3 โครงร่างขั้นตอนการดำเนินงาน

ตารางที่ 1 ตัวอย่างรูปแบบหน้าที่มีลักษณะการบ่งบอกเพศที่ชัดเจน และไม่ชัดเจน

เพศ ลักษณะรูป	ชัดเจน	ไม่ชัดเจน
ชาย		
หญิง		

#### 4. ผลการทดลอง

จากข้อมูลรูปแบบหน้าคนทั้งหมด 400 รูป ผู้วิจัยใช้ภาษา python 2.7 เขียนบนโปรแกรม spyder ดังจะเห็นตัวอย่างได้จากรูปที่ 4 และได้ผลการทดลองดังต่อไปนี้



```

1: import sklearn as sk
2: import numpy as np
3: import matplotlib.pyplot as plt
4: from sklearn.cross_validation import train_test_split
5: from sklearn.cross_validation import cross_val_score, KFold
6: from sklearn.metrics import confusion_matrix
7: from sklearn.datasets import fetch_olivetti_faces
8: from sklearn import metrics
9: from sklearn.cross_validation import cross_val_score
10: X_train, X_test, y_train, y_test = train_test_split(faces.data, faces.target, test_size=0.1)
11: X_train, X_test, y_train, y_test = train_test_split(faces.data, faces.target, test_size=0.1)
12: from memory_profiler import memory_usage
13: import timeit
14: start = timeit.default_timer()
15:
16:
17: def print_faces(images, target, top_n):
18:     # Sort on the figure size in pixels
19:     fig = plt.figure(figsize=(10, 10))
20:     fig.subplots_adjust(left=0, right=1, bottom=0, top=1,
21:                         hspace=0.05, wspace=0.05)
22:     for i in range(top_n):
23:         # Get the indices of the top_n images
24:         p = fig.add_subplot(20, 20, i + 1, ticks=[])
25:         p.imshow(images[i], cmap=plt.cm.gray)
26:         p.text(0, 14, str(target[i]))
27:         p.text(0, 16, str(i))
28:
29: def main():
30:     print_faces(X_train, y_train, 10)
31:     print_faces(X_test, y_test, 10)
32:
33: if __name__ == '__main__':
34:     main()
35:
36:
37:
38:
39:
40:
41:
42:
43:
44:
45:
46:
47:
48:
49:
50:
51:
52:
53:
54:
55:
56:
57:
58:
59:
60:
61:
62:
63:
64:
65:
66:
67:
68:
69:
70:
71:
72:
73:
74:
75:
76:
77:
78:
79:
80:
81:
82:
83:
84:
85:
86:
87:
88:
89:
90:
91:
92:
93:
94:
95:
96:
97:
98:
99:
100:
101:
102:
103:
104:
105:
106:
107:
108:
109:
110:
111:
112:
113:
114:
115:
116:
117:
118:
119:
120:
121:
122:
123:
124:
125:
126:
127:
128:
129:
130:
131:
132:
133:
134:
135:
136:
137:
138:
139:
140:
141:
142:
143:
144:
145:
146:
147:
148:
149:
150:
151:
152:
153:
154:
155:
156:
157:
158:
159:
160:
161:
162:
163:
164:
165:
166:
167:
168:
169:
170:
171:
172:
173:
174:
175:
176:
177:
178:
179:
180:
181:
182:
183:
184:
185:
186:
187:
188:
189:
190:
191:
192:
193:
194:
195:
196:
197:
198:
199:
200:
201:
202:
203:
204:
205:
206:
207:
208:
209:
210:
211:
212:
213:
214:
215:
216:
217:
218:
219:
220:
221:
222:
223:
224:
225:
226:
227:
228:
229:
230:
231:
232:
233:
234:
235:
236:
237:
238:
239:
240:
241:
242:
243:
244:
245:
246:
247:
248:
249:
250:
251:
252:
253:
254:
255:
256:
257:
258:
259:
260:
261:
262:
263:
264:
265:
266:
267:
268:
269:
270:
271:
272:
273:
274:
275:
276:
277:
278:
279:
280:
281:
282:
283:
284:
285:
286:
287:
288:
289:
290:
291:
292:
293:
294:
295:
296:
297:
298:
299:
300:
301:
302:
303:
304:
305:
306:
307:
308:
309:
310:
311:
312:
313:
314:
315:
316:
317:
318:
319:
320:
321:
322:
323:
324:
325:
326:
327:
328:
329:
330:
331:
332:
333:
334:
335:
336:
337:
338:
339:
340:
341:
342:
343:
344:
345:
346:
347:
348:
349:
350:
351:
352:
353:
354:
355:
356:
357:
358:
359:
360:
361:
362:
363:
364:
365:
366:
367:
368:
369:
370:
371:
372:
373:
374:
375:
376:
377:
378:
379:
380:
381:
382:
383:
384:
385:
386:
387:
388:
389:
390:
391:
392:
393:
394:
395:
396:
397:
398:
399:
400:
401:
402:
403:
404:
405:
406:
407:
408:
409:
410:
411:
412:
413:
414:
415:
416:
417:
418:
419:
420:
421:
422:
423:
424:
425:
426:
427:
428:
429:
430:
431:
432:
433:
434:
435:
436:
437:
438:
439:
440:
441:
442:
443:
444:
445:
446:
447:
448:
449:
450:
451:
452:
453:
454:
455:
456:
457:
458:
459:
460:
461:
462:
463:
464:
465:
466:
467:
468:
469:
470:
471:
472:
473:
474:
475:
476:
477:
478:
479:
480:
481:
482:
483:
484:
485:
486:
487:
488:
489:
490:
491:
492:
493:
494:
495:
496:
497:
498:
499:
500:
501:
502:
503:
504:
505:
506:
507:
508:
509:
510:
511:
512:
513:
514:
515:
516:
517:
518:
519:
520:
521:
522:
523:
524:
525:
526:
527:
528:
529:
530:
531:
532:
533:
534:
535:
536:
537:
538:
539:
540:
541:
542:
543:
544:
545:
546:
547:
548:
549:
550:
551:
552:
553:
554:
555:
556:
557:
558:
559:
560:
561:
562:
563:
564:
565:
566:
567:
568:
569:
570:
571:
572:
573:
574:
575:
576:
577:
578:
579:
580:
581:
582:
583:
584:
585:
586:
587:
588:
589:
590:
591:
592:
593:
594:
595:
596:
597:
598:
599:
600:
601:
602:
603:
604:
605:
606:
607:
608:
609:
610:
611:
612:
613:
614:
615:
616:
617:
618:
619:
620:
621:
622:
623:
624:
625:
626:
627:
628:
629:
630:
631:
632:
633:
634:
635:
636:
637:
638:
639:
640:
641:
642:
643:
644:
645:
646:
647:
648:
649:
650:
651:
652:
653:
654:
655:
656:
657:
658:
659:
660:
661:
662:
663:
664:
665:
666:
667:
668:
669:
670:
671:
672:
673:
674:
675:
676:
677:
678:
679:
680:
681:
682:
683:
684:
685:
686:
687:
688:
689:
690:
691:
692:
693:
694:
695:
696:
697:
698:
699:
700:
701:
702:
703:
704:
705:
706:
707:
708:
709:
710:
711:
712:
713:
714:
715:
716:
717:
718:
719:
720:
721:
722:
723:
724:
725:
726:
727:
728:
729:
730:
731:
732:
733:
734:
735:
736:
737:
738:
739:
740:
741:
742:
743:
744:
745:
746:
747:
748:
749:
750:
751:
752:
753:
754:
755:
756:
757:
758:
759:
760:
761:
762:
763:
764:
765:
766:
767:
768:
769:
770:
771:
772:
773:
774:
775:
776:
777:
778:
779:
780:
781:
782:
783:
784:
785:
786:
787:
788:
789:
790:
791:
792:
793:
794:
795:
796:
797:
798:
799:
800:
801:
802:
803:
804:
805:
806:
807:
808:
809:
810:
811:
812:
813:
814:
815:
816:
817:
818:
819:
820:
821:
822:
823:
824:
825:
826:
827:
828:
829:
830:
831:
832:
833:
834:
835:
836:
837:
838:
839:
840:
841:
842:
843:
844:
845:
846:
847:
848:
849:
850:
851:
852:
853:
854:
855:
856:
857:
858:
859:
860:
861:
862:
863:
864:
865:
866:
867:
868:
869:
870:
871:
872:
873:
874:
875:
876:
877:
878:
879:
880:
881:
882:
883:
884:
885:
886:
887:
888:
889:
890:
891:
892:
893:
894:
895:
896:
897:
898:
899:
900:
901:
902:
903:
904:
905:
906:
907:
908:
909:
910:
911:
912:
913:
914:
915:
916:
917:
918:
919:
920:
921:
922:
923:
924:
925:
926:
927:
928:
929:
930:
931:
932:
933:
934:
935:
936:
937:
938:
939:
940:
941:
942:
943:
944:
945:
946:
947:
948:
949:
950:
951:
952:
953:
954:
955:
956:
957:
958:
959:
960:
961:
962:
963:
964:
965:
966:
967:
968:
969:
970:
971:
972:
973:
974:
975:
976:
977:
978:
979:
980:
981:
982:
983:
984:
985:
986:
987:
988:
989:
990:
991:
992:
993:
994:
995:
996:
997:
998:
999:
1000:

```

รูปที่ 4 ตัวอย่างผลลัพธ์ของวิธีการดั้งเดิม (Traditional Method)

4.1 วิธีการดั้งเดิม (Traditional Method) เป็นวิธีการที่ให้โมเดลเลือกใบหน้ามาทำการ train และ test แบบสุ่มซึ่งผลลัพธ์ที่ได้คือ โมเดลทำนายได้ค่าความถูกต้องน้อยมาก อยู่ที่ 50% และส่วนใหญ่ยอมทำนายข้อมูลใหม่ให้เป็นใบหน้าของผู้ชาย ด้วยเหตุผลที่ว่า จำนวนใบหน้าผู้ชายมีเยอะกว่าใบหน้าผู้หญิงมาก จะเห็นได้ว่าค่าระลึก (Recall) และค่าความเที่ยง (Precision) รวมกันต่ำ ดังจะเห็นได้จากรูปที่ 5 และหากพิจารณาต่อไปในเฉพาะเพศหญิงจะพบว่าทั้งสองค่าที่กล่าวมา มีค่าเป็น 0 ดังจะเห็นได้จากตารางที่ 3

4.2 วิธีการปรับอัตราส่วนระหว่างหน้าชาย - หญิงให้สมดุล (Balancing Male-Female Ratio) วิธีการนี้เราจะลดจำนวนหน้าผู้ชายที่ใช้การ train และ test ให้มีอัตราส่วนเท่ากับจำนวนหน้าของผู้หญิงผลลัพธ์ที่ได้นั้น คือ โมเดลเริ่มยอมทำนายข้อมูลใหม่ว่าเป็นใบหน้าผู้หญิงมากขึ้นกว่าเดิม แต่ความถูกต้องยังถือว่ายังไม่สูงนัก อยู่ที่ 65%



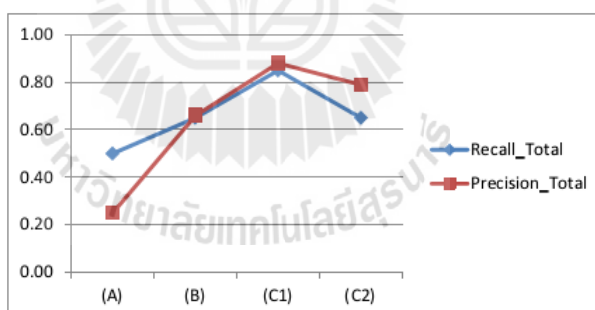
4.3 วิธีการคัดเลือกใบหน้าผู้หญิงที่เด่นชัดเพื่อใช้ในโมเดล (Using obvious female face shape) วิธีการนี้ผู้วิจัยจะทำการเลือกใบหน้าผู้หญิงที่มีความเด่นชัด มาเพื่อใช้ในการ train และ test รวมถึงยังใช้อัตราส่วนที่เท่ากัน ระหว่างใบหน้าชาย และ หญิงผลลัพธ์ที่ได้นั้น มีความถูกต้องและแม่นยำสูงชันมาก โดยสามารถทำนายผลใบหน้าผู้หญิงด้วยความแม่นยำถึง 80% ดังจะเห็นได้จากตารางที่ 2 ไม่เพียงแค่นั้น ค่าระลึก (Recall) และค่าความเที่ยง (Precision) รวมนั้นสูงชันมาก ดังจะเห็นได้จากรูปที่ 5

4.4 วิธีการคัดเลือกใบหน้าผู้หญิงที่ไม่เด่นชัดเพื่อใช้ในโมเดล (Using unobvious female face shape) วิธีการนี้ ผู้วิจัยจะทำการเลือกใบหน้าผู้หญิงที่มีความไม่เด่นชัด มาเพื่อใช้ในการ train และ test รวมถึงยังใช้อัตราส่วนที่เท่ากัน ระหว่างใบหน้าชาย และ หญิงผลลัพธ์ที่ได้นั้น มีความแม่นยำอยู่ในระดับที่ไม่สูงนัก ซึ่งค่อนข้างคล้ายกับ ผลลัพธ์ในข้อ 4.2

จากผลลัพธ์การทดลองข้างต้น ผู้วิจัยได้ทำการแสดงผลให้อยู่ในรูปแบบตาราง และกราฟเพื่อวัดผลระหว่างวิธีการทั้ง 4 ซึ่ง (A) คือวิธีการดั้งเดิม, (B) คือวิธีการปรับอัตราส่วนระหว่างหน้าชาย - หญิง ให้สมดุล, (C1) คือวิธีการคัดเลือกใบหน้าผู้หญิงที่เด่นชัดเพื่อใช้ในโมเดล และ (C2) คือวิธีการคัดเลือกใบหน้าผู้หญิงที่ไม่เด่นชัดเพื่อใช้ในโมเดลตามรูปด้านล่างนี้

ตารางที่ 2 แสดงค่าความแม่นยำ (Accuracy) ของแต่ละวิธีการ

Method	(A = traditional)	(B = balancing)	(C1 = obvious)	(C2 = unobvious)
Accuracy	0.50	0.65	0.85	0.65



รูปที่ 5 กราฟแสดงค่าระลึก (Recall) และค่าความเที่ยง (Precision) ของแต่ละวิธีการ

ตารางที่ 3 แสดงค่าระลึก (Recall) และค่าความเที่ยง (Precision) ของแต่ละวิธีการโดยแยกชาย-หญิง

Method	Male				Female			
	(A)	(B)	(C1)	(C2)	(A)	(B)	(C1)	(C2)
Recall	1.0	0.55	1.00	1.00	0.00	0.75	0.70	0.30
Precision	0.5	0.69	0.77	0.59	0.00	0.62	1.00	1.00

จากการทดลองจะเห็นได้ว่าในขั้นตอนแรก ได้นำข้อมูลมาทำการตรวจจับโดยใช้วิธีการดั้งเดิม ผลลัพธ์ที่ได้มีประสิทธิภาพค่อนข้างต่ำ โดยเฉพาะกลุ่มใบหน้าเพศหญิง ด้วยสาเหตุที่ในข้อมูลมีจำนวน ใบหน้าผู้หญิงค่อนข้างน้อยจึงได้ทดลองลดจำนวนหน้าผู้ชายที่ใช้การ train และ test ให้มีอัตราส่วน เท่ากับจำนวนหน้าของผู้หญิง ผลลัพธ์ที่ได้ดีขึ้นแต่ค่าความถูกต้องยังค่อนข้างต่ำจึงคิดว่าความชัดเจนของ รูปภาพน่าจะมีส่วนในการตรวจจับรูปจึงได้แบ่งรูปภาพที่ชัดเจนและไม่ชัดเจนออกจากกันเพื่อใช้ในโมเดล ผลลัพธ์วิธีการคัดเลือกใบหน้าผู้หญิงที่เด่นชัดเพื่อใช้ในการสร้างโมเดลได้ผลลัพธ์ที่ดี ส่วนวิธีการคัดเลือก ใบหน้าผู้หญิงที่ไม่เด่นชัดเพื่อใช้ในโมเดลได้ผลลัพธ์เท่ากับวิธีการปรับอัตราส่วนระหว่างหน้าชาย - หญิงให้ สมดุล จากผลการทดลองทำให้ทราบว่า การตรวจจับใบหน้าให้มีประสิทธิภาพนั้นขึ้นอยู่กับจำนวนข้อมูล และความชัดเจนในการบ่งบอกเพศของรูปใบหน้าซึ่งรูปใบหน้าผู้หญิงบางรูปมีความคล้ายคลึงกับรูปใบหน้า ผู้ชายทำให้แยกแยะออกยาก

#### อภิปรายผล

จากผลการทดลองข้างต้นนี้แสดงให้เห็นถึง ความสำคัญในกระบวนการ Data Preparation ซึ่ง เราจะสามารถเห็นได้อย่างชัดเจนว่า การเตรียมข้อมูลให้มีอัตราส่วนระหว่างใบหน้าชาย-หญิง ด้วย อัตราส่วนที่เท่า ๆ กันจะทำให้เห็นผลลัพธ์ที่ดีขึ้น นอกจากนั้นผู้วิจัยยังพบอีกว่า ความเด่นชัดในการบ่งบอก เพศของใบหน้าผู้หญิงนั้น มีผลเป็นอย่างมากในการสร้างโมเดลเพื่อทำให้ความถูกต้องและแม่นยำสูงขึ้น ซึ่งเป็นวิธีการที่ดีที่สุดในการคัดเลือกข้อมูลเพื่อใช้ในอัลกอริทึมจำแนกประเภท

#### 5. สรุปผลงานวิจัย

งานวิจัยนี้นำเสนอวิธีการจำแนกรูปใบหน้าระหว่างชาย-หญิง ซึ่งเท่าที่ค้นคว้าวรรณกรรมที่ เกี่ยวข้อง ยังไม่พบว่ามีการวิจัยใด นำเสนอเทคนิค Support Vector Machine เพื่อที่จะจำแนกใบหน้า ชาย-หญิงมาก่อน โดยงานวิจัยชิ้นนี้จะมุ่งเน้นการปรับปรุงโมเดลเพื่อให้ตรวจจับรูปใบหน้าที่ตรวจจับได้ ยากและถ้ารูปในบางกลุ่มมีจำนวนน้อย โมเดลจะต้องสามารถตรวจจับได้ถูกต้องมากที่สุด ซึ่งในการทำงาน วิจัยชิ้นนี้พบกับความยากลำบากในการแยกใบหน้าชาย-หญิง อาทิเช่นข้อมูลตัวอย่างนั้นมีจำนวนใบหน้า ของผู้หญิงมีน้อยมากเมื่อเทียบอัตราส่วนกับใบหน้าของผู้ชาย นอกจากนี้รูปหน้าของชายและหญิงบางรูป ไม่มีความเด่นชัดในการบ่งบอกเพศ โดยทางผู้วิจัยได้พยายามปรับปรุงโมเดลด้วยวิธีการต่าง ๆ เพื่อให้การ จำแนกเพศจากรูปใบหน้าที่มีประสิทธิภาพยิ่งขึ้น จากนั้นจึงนำมาวัดผลถึงประสิทธิภาพการจำแนกด้วย confusion matrix โดยวิธีการปรับปรุงโมเดล ประกอบไปด้วย วิธีการปรับอัตราส่วนระหว่างหน้าชาย- หญิงให้สมดุล, วิธีการคัดเลือกใบหน้าผู้หญิงที่เด่นชัด และ วิธีการคัดเลือกใบหน้าผู้หญิงที่ไม่เด่นชัด

จากผลการทดลอง ผู้วิจัยพบว่า วิธีการคัดเลือกใบหน้าที่บ่งบอกลักษณะความเป็นผู้หญิงที่เด่นชัด หลังจากการปรับอัตราส่วนระหว่างจำนวนภาพใบหน้าชายและหญิงให้สมดุลแล้ว เป็นวิธีการให้ความ แม่นยำที่ดีที่สุด โดยมีค่าความแม่นยำ 0.85 หรือ 85% เมื่อเปรียบเทียบกับวิธีการจำแนกภาพที่ไม่มีการ ปรับปรุงกระบวนการจำแนกจะให้ความแม่นยำเพียง 50%

#### 6. เอกสารอ้างอิง

- [1] กิตติศักดิ์ เกิดประสพ และ นิตยา เกิดประสพ (2014) Data Mining Application. from: [https://sites.google.com/site/kittisakthailand56/home/3\\_56-dm-applications](https://sites.google.com/site/kittisakthailand56/home/3_56-dm-applications) Retrieved Mar 01, 2014
- [2] พรพล ธรรมรงค์รัตน์, ถัดดา ปรีชาวีรกุล, และวิภาดา เวทย์ประสิทธิ์. “การจำแนกประเภทเว็บเพจโดยใช้ค่าความถี่เอกสารและซัพพอร์ตเวกเตอร์แมชชีน”, The 12th National Computer Science and Engineering Conference, 2008.
- [3] M. A. Turk and A. P. Pentland, “Face recognition using Eigen Faces”, Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR '91, pp. 586-591, 1991.
- [4] Joachims, “Text categorization with support vector machines: Learning with many relevant features”, Proceedings of the 10 th European Conference on Machine Learning, pp.137-142, 1998.



การประชุมวิชาการ "มศว วิจัย" ครั้งที่ 8  
26-27 พฤศจิกายน 2557 มหาวิทยาลัยศรีนครินทรวิโรฒ

**SWURES8-103 เทคนิคการเพิ่มประสิทธิภาพของโมเดลรู้จำภาพลายเซ็น**  
**A TECHNIQUE TO IMPROVE IMAGE RECOGNITION PERFORMANCE OF**  
**HANDWRITTEN SIGNATURE**

อนุสรา หิรัญวานากุล<sup>1</sup>, กิตติศักดิ์ เกิดประสพ<sup>2</sup>, นิตยา เกิดประสพ<sup>3</sup>  
Anusara Hirunyanakul<sup>1</sup>, Kittisak Kerdprasop<sup>2</sup>, Nittaya Kerdprasop<sup>3</sup>

สาขาวิชาวิศวกรรมคอมพิวเตอร์ สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี  
School of Computer Engineering, Institute of Engineering, Suranaree University of Technology, Thailand.  
Corresponding author, E-mail: Anusara.cpe8@gmail.com

**บทคัดย่อ**

งานวิจัยชิ้นนี้ ผู้วิจัยได้ศึกษาปัญหาเกี่ยวกับเรื่อง การรู้จำภาพซึ่งใช้ในชีวิตจริง นั่นก็คือการรู้จำภาพลายเซ็น งานวิจัยชิ้นนี้มาจากการเก็บข้อมูลจริง จากผู้เข้าร่วมที่เป็นนักศึกษาในระดับมหาวิทยาลัย 30 คน โดยให้ผู้เข้าร่วมวิจัยเซ็นลายเซ็น 20 ครั้ง โดยใช้ 5 ลายเซ็นจาก 20 ลายเซ็นในการสุ่มเลือกเพื่อทำการทดสอบ ภาพลายเซ็นจะถูกสแกนจากกระดาษด้วย เครื่องสแกนภาพ Fuji ApeosPort-IV5070 โดยแต่ละภาพจะมีขนาด 38x144 พิกเซลเท่ากันทั้งหมด โดยทั้ง 5,472 พิกเซลในแต่ละภาพจะถูกแปลงค่าเป็นค่าความเข้ม จาก 0 ถึง 255 โดย 0 คือค่าความเข้มที่มากที่สุด หรือก็คือสีดำนั่นเอง ส่วนค่า 255 หมายถึงสีขาว ภาพลายเซ็นเหล่านี้จะถูกนำไปเทรน และ ทดสอบความแม่นยำโดยอัลกอริทึม ทั้ง 4 แบบ ได้แก่ ซัพพอร์ตเวกเตอร์แมชชีน (เคอร์เนลฟังก์ชันสี่ชนิด คือ ฟังก์ชันเชิงเส้น, ฟังก์ชันพหุนาม, ฟังก์ชันเรเดียลเบส และฟังก์ชันซิกมอยด์) , เคเนียร์เรสเนเบอร์, เพอร์เซปตรอน , นาอ์ฟเบย์ (ฟังก์ชันสามชนิด คือ เกาเซียน , มัลติโนเมียล และเบอร์นูลลี) จากผลการทดลองแรกพบว่า อัลกอริทึม SVM (ซัพพอร์ตเวกเตอร์แมชชีน) ที่เคอร์เนลฟังก์ชันพหุนาม ได้ให้ค่าความแม่นยำที่สูงที่สุด (โดยมีค่าความแม่นยำ ที่ชุดข้อมูลการทดสอบสูงถึง 95.33%) โดยทั้งนี้ผู้วิจัยได้เพิ่มประสิทธิภาพโดยการที่เพิ่มชุดข้อมูลที่นำสนใจเข้าไป ซึ่งในที่นี้เราสนใจในข้อมูลบริเวณที่มีความเข้มสูง ดังนั้นเราจึงทำการเพิ่มข้อมูลที่มีค่าความเข้ม 0-200 เข้าไปต่อกับข้อมูลในแต่ละชุด (แต่ละภาพลายเซ็น) ทั้งสิ้น 717 พิกเซล ซึ่งด้วยวิธีการนี้เองสามารถทำให้ 2 อัลกอริทึม (SVM ที่เคอร์เนลฟังก์ชันเชิงเส้น และ SVM ที่เคอร์เนลฟังก์ชันพหุนาม) มีความแม่นยำสูงขึ้นมากกว่า 95.33% (โดยค่าสูงสุดคือ 97.33%) และ เคเนียร์เรสเนเบอร์ เป็นอีกอัลกอริทึม ที่มีค่าความแม่นยำสูงขึ้นมากอย่างมีนัยสำคัญ

**คำสำคัญ:** การรู้จำภาพ ลายเซ็น ซัพพอร์ตเวกเตอร์แมชชีน เคเนียร์เรสเนเบอร์ นาอ์ฟเบย์

**Abstract**

In this research we study the problem of image recognition that has been applied to the real world application, that is, handwritten signature identification. We collected data from the 30 university students. Each student has been requested to sign 20 times; 5 of these are randomly selected as test data. We then scanned the images using Fuji ApeosPort-IV5070 scanner. Each picture has the same size, 38x144. Each of the 5472 pixels is to be encoded with the intensity value ranging from 0 to 255. The value 0 is the highest intensity (black color), whereas 255 represents white. These image data are

then trained and identified by 4 different algorithms. Algorithm SVM with polynomial kernel is the most accurate algorithm (accuracy on test data = 95.33%). We then boosted the recognition process by adding more intensive data. We are interested in the dark area, so we repeat the pixel with intensity value 0-200. Additional 717 pixels are then added to the image data. It turned out that two algorithms (SVM-linear and SVM-polynomial) can improve accuracy rate of signature recognition to be higher than 95.33% (the highest one is 97.33%). And the one of most significant improved algorithm is the k-nearest neighbor algorithm.

**Keywords:** image recognition, handwritten signature, support vector machine, k-nearest neighbor, naive bayes

#### บทนำ

เทคโนโลยีในปัจจุบันนั้น ให้ความสนใจเกี่ยวกับการตรวจสอบความถูกต้องของตัวบุคคล โดยใช้ข้อมูลทางชีวภาพ ทั้งลักษณะเฉพาะทางกายภาพ รวมไปถึงลักษณะทางพฤติกรรม มาใช้ในการตรวจสอบหรือแสดงตน เรียกว่า ไบโอมेटริกซ์ (Biometrics) ตัวอย่างเช่น ลายนิ้วมือ, ใบหน้า, ม่านตา, ลายเซ็น เป็นต้น สิ่งเหล่านี้จะถูกเก็บรวบรวมเป็นข้อมูลเพื่อนำมาเปรียบเทียบกับความแตกต่างของแต่ละบุคคลนั้นออกจากกัน หรือ เพื่อจดจำ รวมไปถึงการรู้จำแต่ละบุคคล

การใช้ไบโอมेटริกซ์ในการตรวจสอบตัวบุคคล หรือระบุตัวบุคคลนั้น ทำให้เกิดความสะดวกสบายในการที่ไม่ต้องพกเอกสารเพื่อระบุตัวบุคคล และยังสามารถลดปัญหาเรื่องการปลอมแปลงเอกสารได้อีกด้วย การรู้จำ (Recognition) คืองานวิจัยลักษณะหนึ่งที่มีความสำคัญ ซึ่งเป็นแขนงหนึ่งของปัญญาประดิษฐ์ (Artificial Intelligence) การรู้จำมีหลายชนิด และการรู้จำลายเซ็นนั้นเป็นหนึ่งในสิ่งที่เป็นที่นิยมอย่างสูง เนื่องจากลายเซ็นนั้นคือ อัตลักษณ์ภายนอกอย่างหนึ่ง ที่ใช้ในการระบุตัวตนของบุคคล ซึ่งนิยมใช้กันทั่วไปเนื่องจากไม่ต้องมีอุปกรณ์เฉพาะทางเพื่อมาใช้ร่วม อย่างเช่น เครื่องสแกนลายมือ หรือเครื่องสแกนม่านตา ลายเซ็นของแต่ละบุคคลมีลักษณะที่ต่างกัน การปลอมแปลงหรือลอกเลียนแบบนั้น จึงเป็นไปได้ยาก งานวิจัยด้านการวิเคราะห์และรู้จำลายเซ็นจึงได้ถูกพัฒนาขึ้น เพื่อให้สามารถวิเคราะห์ความเฉพาะตัวของลายเซ็นมนุษย์แต่ละคน (Human Identification) แต่ปัญหาของลายเซ็นที่มีความคล้ายคลึงกันของแต่ละบุคคล หรือ ลายเซ็นของบุคคลเดียวกัน ก็ยังคงมีความแตกต่างกัน อันเนื่องมาจากสภาวะแวดล้อมต่าง ๆ ที่ส่งผลให้การเซ็นของบุคคลในแต่ละครั้งผิดเพี้ยนไปจากเดิม จึงส่งผลทำให้ความถูกต้องของการระบุตัวบุคคลนั้นต่ำลง

งานวิจัยที่เกี่ยวข้องกับการตรวจสอบลายเซ็น และการรู้จำลายเซ็นนั้น ส่วนใหญ่จะมุ่งเน้นไปที่การศึกษาเปรียบเทียบ และการปรับปรุงเพื่อเพิ่มประสิทธิภาพของโมเดล ซึ่งทางผู้วิจัยนั้นเล็งเห็นถึงการเพิ่มประสิทธิภาพทางด้านความแม่นยำในการทำนาย โดยการนำเสนอเทคนิคใหม่ด้วยการเพิ่มชุดข้อมูลที่มีความน่าสนใจเข้าไป ต่อท้ายชุดข้อมูลเดิม ในส่วนของฐานข้อมูลก็นำมาใช้ในงานวิจัยนั้น ทางผู้วิจัยเลือกที่จะเก็บรวบรวมตัวอย่างลายเซ็นด้วยตนเองจากผู้เข้าร่วมทำการวิจัย ที่เป็นนักศึกษาซึ่งมีการใช้ลายเซ็นจริงในชีวิตประจำวัน นับได้ว่าเป็นแหล่งข้อมูลที่แสดงให้เห็นถึงการนำไปใช้งานได้จริง

#### งานวิจัยและทฤษฎีที่เกี่ยวข้อง

##### การเรียนรู้ของเครื่อง (Machine Learning)

การเรียนรู้ของเครื่อง คือ การพยายามทำให้เครื่องคอมพิวเตอร์ หรือเครื่องอิเล็กทรอนิกส์ มีความสามารถในการทำงานที่มีความฉลาดได้ เป็นสาขาหนึ่งของปัญญาประดิษฐ์ ที่เกี่ยวข้องกับการพัฒนาเทคนิค

วิธี เพื่อให้คอมพิวเตอร์สามารถเรียนรู้จากข้อมูลเข้า โดยเรียนรู้จากข้อมูลบางส่วน โดยเน้นที่วิธีการวิเคราะห์ชุดข้อมูล เทคนิคการเรียนรู้ของเครื่อง [1] ถูกใช้เพื่อเพิ่มประสิทธิภาพในการแก้ปัญหาต่าง ๆ เช่น การสร้างให้คอมพิวเตอร์สามารถแยกแยะวัตถุ เสียง ตัวอักษร หรือจำแนกรูปภาพต่าง ๆ ได้ กระบวนการจำแนกข้อมูลจำนวนมากที่มีมนุษย์ต้องอาศัยระยะเวลาอย่างมากในการทำนั้น เราสามารถฝึกสอนกระบวนการดังกล่าวนี้ให้แก่เครื่องจักรหรือคอมพิวเตอร์ให้สามารถประมวลผล หรือทำแทนมนุษย์ได้

#### การรู้จำแบบ (Pattern Recognition)

มนุษย์เรานั้นมีความสามารถในการรู้จำ [2] วัตถุ สิ่งของ รวมไปถึงคุณลักษณะของสิ่งเหล่านั้นได้ จากการรับรู้ และ จดจำ เนื่องจากเป็นความสามารถที่เราได้รับมาตั้งแต่กำเนิด เป็นเรื่องยากมากที่เราจะนำความสามารถเหล่านี้มาวิเคราะห์และนำไปใส่ในระบบ หรือ กระบวนการใดกระบวนการหนึ่ง อย่างเช่น มันเป็นเรื่องที่ไม่ยากเลยสำหรับมนุษย์ที่จะบอกว่า "เสียงที่ได้ยินนั้นเป็นเสียงไวโอลิน", "ลายมือที่เราเห็นนี้เป็นเลขสาม หรือ เลขแปด", "กลิ่นนี้เป็นกลิ่นหัวหอม หรือ กลิ่นดอกกุหลาบ" หรือแม้กระทั่ง เราสามารถจดจำและแยกแยะบุคคลแต่ละบุคคลรอบ ๆ ตัวเรา ได้จากใบหน้าโดยเพียงแค่ใช้สติตรรกะ แต่สำหรับเครื่องคอมพิวเตอร์นั้น การแยกแยะใบหน้าบุคคลได้จะต้องวิเคราะห์หาองค์ประกอบโครงสร้างของใบหน้า และด้วยการวิเคราะห์เหล่านั้นเอง นำไปสู่การสร้าง"รูปแบบ"ของใบหน้าแต่ละบุคคล ซึ่งถ้าหากมีใบหน้าของบุคคลเดิมนั้นเข้ามา ระบบก็จะสามารถทำการตอบสนองและบอกได้ว่าเป็นบุคคลเดียวกัน (แม้อากัปกิริยา หรือ สีหน้าท่าทางจะเปลี่ยนไปบ้าง) สิ่งเหล่านี้เราเรียกมันว่า การรู้จำรูปแบบ

#### ไบโอเมตริกซ์ (Biometrics)

ระบบไบโอเมตริกซ์เป็นหนึ่งในระบบการจดจำรูปแบบ [3] ซึ่งจะมีการตรวจสอบ หรือ วิเคราะห์ คุณสมบัติเด่น หรือ ลักษณะที่แตกต่างกัน ของข้อมูลที่ได้มา กับคุณสมบัติของข้อมูลที่เก็บไว้ในฐานข้อมูลอยู่แล้ว ระบบไบโอเมตริกซ์จะมีโหมดในการทำงานโหมดใดโหมดหนึ่งดังต่อไปนี้

##### ○ โหมดการตรวจสอบ (Verification Mode)

หลังจากผู้ใช้ส่งลายเซ็นให้กับระบบไบโอเมตริกซ์ ระบบจะทำการตรวจสอบตัวตนของแต่ละบุคคล ผ่าน PIN, login name ฯลฯ ในการตอบสนอง ระบบจะทำการรับรู้และทำการตรวจสอบ กับข้อมูลของผู้ใช้ร้องขอเข้ามาแบบ 1:1 (หนึ่งต่อหนึ่ง) โดยจะเปรียบเทียบระหว่าง ไบโอเมตริกซ์ลายเซ็นที่ส่งเข้ามา และ ไบโอเมตริกซ์ลายเซ็นของกับผู้ใช้งานที่ได้ทำการลงทะเบียนไว้

##### ○ โหมดระบุตัวตน (Identification Mode)

ในโหมดนี้ระบบจะพยายามรับรู้และจดจำ (Recognize) ลักษณะของผู้ใช้ โดยการเปรียบเทียบลายเซ็นไบโอเมตริกซ์ส่งไปยังทุกลายเซ็นที่ลงทะเบียนในฐานข้อมูลโดยเป็นปฏิบัติการแบบ 1: N (หนึ่งต่อหลายคน) เปรียบเทียบโดยไม่ต้องเรียกร่องเอกลักษณ์เฉพาะจากผู้ใช้

##### ○ โหมดการคัดกรอง (Screening Mode)

โหมดนี้เป็นส่วนขยายของโหมดระบุตัวตน ที่ระบบไบโอเมตริกซ์มั่นใจว่า บุคคลที่ถูกระบุอยู่นั้นจะถูกตรวจสอบและผ่านการคัดกรอง โดยการทำงานจะยังคงเป็นแบบ 1: N (หนึ่งต่อหลายคน)

#### การทำเหมืองข้อมูลสำหรับการจำแนกแยกประเภท (Data Mining for Classification)

การจำแนกแยกประเภท ส่วนใหญ่จะอยู่ในขั้นตอนสุดท้ายของกระบวนการ [2] โดยทั่วไปมันคือการเลือกวัตถุ หรือ ข้อมูล ให้ไปอยู่ตามคลาสต่าง ๆ โดยในกรณีของรูปภาพที่ได้รับมา แล้วจำแนกให้อยู่ในคลาสเดียวกันนั้น อาจมีความแตกต่างกันบางเล็กน้อย แต่ถ้าเทียบความแตกต่างระหว่างคลาสแล้ว จะมีมากกว่าอย่างเห็นได้ชัดเจน คุณภาพของรูปภาพ หรือ ข้อมูลที่จะนำมาใช้ในการจำแนกนั้นขึ้นอยู่กับกระบวนการเก็บรวบรวมข้อมูล รวมถึงขั้นตอนการปรับแต่งรูปภาพ ดังนั้นกระบวนการก่อนนำมาเข้าสู่การจำแนก หรือ ที่เรียกว่า Pre-processing

นั้นจึงมีความสำคัญเป็นอย่างมาก โดยอาจจะมียหลายเทคนิคมาเกี่ยวข้องเพื่อให้ได้ข้อมูล หรือ รูปภาพที่มีคุณภาพมากยิ่งขึ้น และเหมาะสมกับการนำไปใช้งาน

#### เพอร์เซปตรอน (Perceptron)

เพอร์เซปตรอน เป็นอัลกอริทึมที่ใช้ในการจำแนกประเภท หนึ่ง [4-5] มีพื้นฐานมาจากแบบจำลองเชิงเส้น ซึ่งแบ่งแยกได้ดีกว่าการจำแนกเชิงเส้นทั่วไป หลักการคือ การพยายามลากเส้นแบ่งระหว่างกลุ่ม โดยที่โมเดลจะพยายามหาเส้นแบ่งที่ดีที่สุด จึงต้องอาศัยการทำซ้ำในทุก ๆ จุด เป็นอีกหนึ่งวิธีที่ง่ายเหมาะสำหรับการเรียนรู้ขนาดใหญ่ โดยไม่จำเป็นต้องมีอัตราการเรียนรู้เริ่มต้น การหาเส้นแบ่งกลุ่มที่ดีที่สุดนั้น โมเดลจะพยายามปรับค่าสัมประสิทธิ์การเรียนรู้ (Learning rate) ไปเรื่อย ๆ จนถึงจุดที่มันคิดว่าเหมาะสมที่สุดแล้ว จึงหยุดคำนวณ ดังนั้นหากกลุ่มข้อมูลอยู่แยกกันแบบชัดเจน เพอร์เซปตรอนจะสามารถแบ่งแยกข้อมูลประเภทนี้ได้ดี

#### ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine)

ซัพพอร์ตเวกเตอร์แมชชีน เป็นอัลกอริทึมที่ใช้ในการจำแนกประเภทหนึ่ง มีพื้นฐานมาจากแบบจำลองเชิงเส้น [ 5-6] ซึ่งเป็นอัลกอริทึม ที่ถูกพัฒนามาจากเพอร์เซปตรอน และเพิ่มลักษณะการพยายามปรับเส้นแบ่งให้เกิดระยะขอบ (Margin) มาก ๆ จัดเป็นเทคนิคที่นำมาใช้ในการแก้ปัญหาด้านการรู้จำอีกเทคนิคหนึ่งแนวคิดคือการนำค่าของกลุ่มข้อมูลมาวางลงในฟีเจอร์สเปซ (Feature space) ต่อจากนั้นนำจุดขอบ (Convex hull) มาลากเส้นขอบเชื่อมต่อกัน ข้อมูลทุกจุดมาคิดค่านวนเหมือนเพอร์เซปตรอน ซึ่งวิธีนี้เองเรียกได้ว่าเป็นข้อดี ที่ทำให้ใช้เวลา น้อยลง) หลังจากลากเส้นเชื่อมขอบได้แล้ว ก็จะทำการสร้างเส้นตรงที่ขนานกัน ระหว่างจุดขอบทั้งสองกลุ่มนั่นเอง จากนั้นจะพยายามสร้างเส้นแบ่งตรงกึ่งกลางระหว่างกลุ่มทั้งสอง โดยให้มีระยะขอบที่มากที่สุด เพื่อแบ่งข้อมูลทั้งสองออกจากกัน และในบางกรณีเพื่อให้เกิดระยะขอบมาก ๆ จึงต้องใช้ตัวแปรอนุโลม (Slack variable) เข้ามาช่วย คือการยอมให้เกิดการทำนายผิดได้บ้าง หรือยอมมองข้ามจุดขอบบางจุดไป เพื่อให้ได้มาซึ่งระยะการแบ่งที่มากที่สุด วิธีแบบนี้เป็นวิธีที่ถือได้ว่าค่อนข้างยืดหยุ่น

#### นาอีฟเบย์ (Naïve Bayes)

นาอีฟเบย์ เป็นวิธีจำแนกประเภทข้อมูลที่มีประสิทธิภาพอีกวิธีหนึ่ง โดยใช้หลักการของความน่าจะเป็น เป็นหลัก ซึ่งมีพื้นฐานมาจากทฤษฎีของเบย์ ( Bayes theorem) ซึ่งใช้งานได้ดีและเหมาะสมกับ กรณีของเซต ตัวอย่างที่มีจำนวนมาก และมี Attribute ของตัวอย่างไม่ขึ้นต่อกัน เข้ามาช่วยในการเรียนรู้จุดมุ่งหมายก็ เพื่อต้องการสร้างอัลกอริทึมที่อยู่ในรูปของความน่าจะเป็นแบบมีเงื่อนไข ข้อดีของวิธีการเรียนรู้แบบนี้ คือ เราสามารถใช้ข้อมูลและความรู้ก่อนหน้า ( Prior knowledge) เข้ามาช่วยในการเรียนรู้ได้ซึ่งพบว่าวิธีนี้ให้ประสิทธิภาพในการเรียนรู้ได้ดี ไม่ด้อยกว่าวิธีการเรียนรู้ประเภทอื่น นอกจากนี้ยังมีหลายฟังก์ชันให้เลือกใช้ อาทิเช่น เกาเซียน , มัลติ โนเมียล และเบอร์นูลลี

#### เคเนียร์สเนเบอร์ (k-Nearest Neighbors)

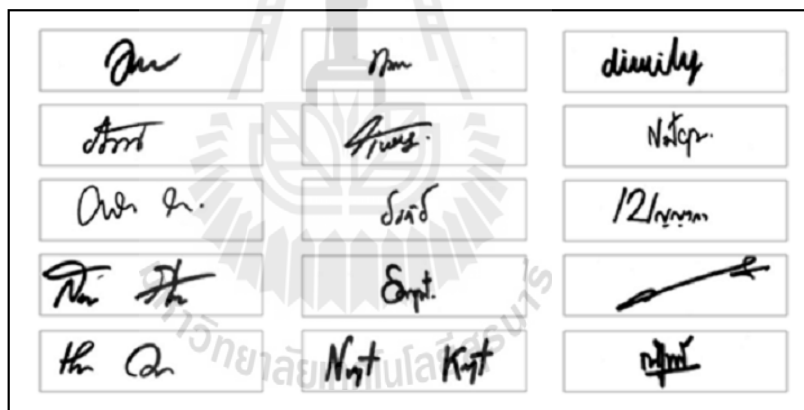
เคเนียร์สเนเบอร์ [7] เป็นอัลกอริทึมที่ใช้ในการจัดกลุ่มข้อมูลที่อยู่ใกล้กัน ให้เป็นกลุ่มเดียวกัน ซึ่งเทคนิคนี้จะทำให้ตัดสินใจได้ว่า คลาสไหนที่จะแทนเงื่อนไขใหม่ ๆ ได้บ้าง โดยการตรวจสอบจำนวน k ซึ่งถ้าหากเงื่อนไขของการตัดสินใจมีความซับซ้อน วิธีนี้สามารถสร้างโมเดลที่มีประสิทธิภาพได้ โดยเริ่มจาก การกำหนดขนาดของ k ค่าณระยะห่าง ( Distance) ของข้อมูลที่ต้องการพิจารณากับกลุ่มข้อมูลตัวอย่าง จากนั้นจัดเรียงลำดับของระยะห่าง และเลือกพิจารณาข้อมูลที่อยู่ใกล้ที่สุดที่ต้องการพิจารณาตามจำนวน k ที่กำหนดไว้ ต่อไปเป็นการพิจารณาข้อมูลจำนวน k ตัว หลังจากนั้นรวบรวมสมาชิกที่ใกล้เคียงที่สุด k ตัว แล้วเลือกกลุ่มที่สมาชิกส่วนใหญ่ที่สุดในกลุ่ม k ดังกล่าวสังกัดอยู่มากที่สุดให้แก่สมาชิกใหม่

**วัตถุประสงค์ของการวิจัย**

- 1) เพื่อศึกษาและเปรียบเทียบโมเดลการจำรู้อยู่ลายเซ็น ที่ได้จากอัลกอริทึมทั้ง 4 แบบ
- 2) เพื่อพัฒนาอัลกอริทึมใหม่ ที่จะช่วยเพิ่มประสิทธิภาพของโมเดลการจำรู้อยู่ลายเซ็น

**วิธีดำเนินงานวิจัย**

งานวิจัยชิ้นนี้ ผู้วิจัยได้ทำการเก็บรวบรวมข้อมูลจริง จากนักศึกษาชั้นปีที่สี่ คณะมนุษยศาสตร์และสังคมศาสตร์ สาขาสารสนเทศศึกษา มหาวิทยาลัยบูรพา จากนั้นได้ทำการคัดกรองเอาแต่บุคคลที่ใช้ลายเซ็นอยู่เป็นประจำ และยังได้ใช้ลายเซ็นค่อนข้างบ่อยในชีวิตประจำวัน ภายหลังการคัดกรองดังกล่าว ทำให้มีจำนวนลายเซ็นที่ใช้ในงานวิจัยนี้ทั้งหมด 600 ลายเซ็น จากผู้เข้าร่วม 30 คน (สามารถไปดาวน์โหลดข้อมูลนี้ได้จาก <https://sites.google.com/site/nhinganusarapesut/signature/datasets>) หลังจากนั้นนำลายเซ็นที่ได้ทั้งหมด มาผ่านเครื่องสแกน FUJI XEROX รุ่น ApeosPort-IV5070 เพื่อทำเป็นไฟล์รูปภาพ โดยทางผู้วิจัยได้ทำการปรับแต่งสีให้เป็นลักษณะขาวดำ และได้ทำการปรับขนาดรูปภาพลายเซ็นให้มีขนาดกว้างยาวเท่ากัน (38x144 pixels) (จากภาพที่ 1) จากนั้นโปรแกรมจะทำการประมวลผล ไฟล์ภาพลายเซ็นที่นำเข้ามา ให้เป็นรูปแบบตารางกว้าง x ยาวตามที่เรากำหนดไว้เบื้องต้น ต่อจากนั้นโปรแกรมจะแทนที่ค่าความเข้มสีลงไปในทุกส่วนของช่องตารางนั้น ค่าความเข้มสีแสดงอยู่ในลักษณะตัวเลข มีค่าระหว่าง 0 ถึง 255 โดยที่ 0 คือ ค่าระดับความเข้มสูงสุด (ในที่นี้คือสีดำ) และตัวเลข 255 คือความเข้มต่ำสุด (ในที่นี้คือสีขาว) (จากภาพที่ 2) ขั้นตอนต่อไป คือการจัดเรียงข้อมูลตัวเลขของไฟล์ภาพแต่ละลายเซ็น ให้อยู่ในรูปของ Array 1มิติ โดยที่ลายเซ็นหนึ่งรูปจะมีค่าข้อมูล 5,472 คอลัมน์



ภาพที่ 1 ตัวอย่างภาพลายเซ็นที่ถูกปรับให้เป็นสีขาว ดำ และทำให้มีขนาดกว้างยาวเท่ากัน (38x144 pixels)

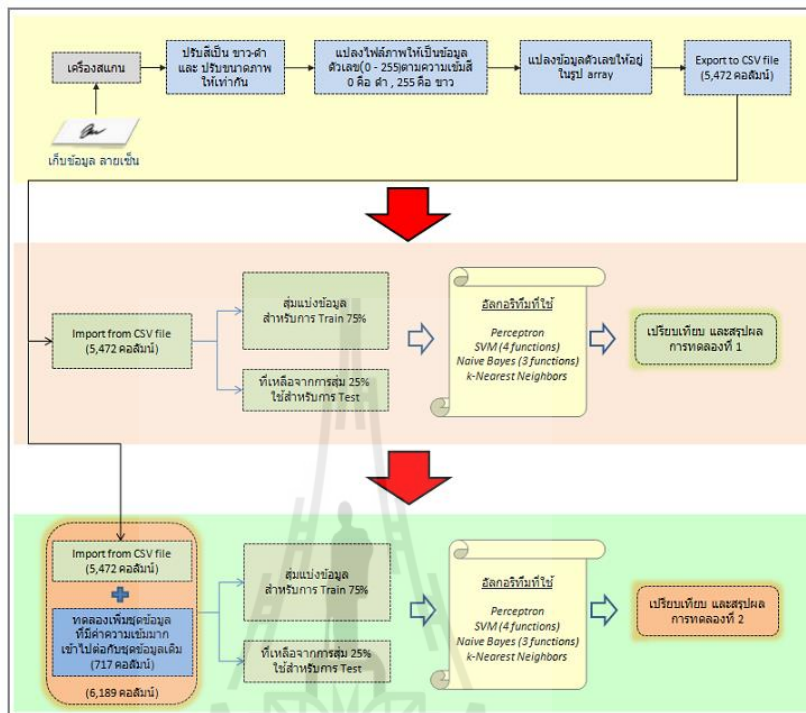


การประชุมวิชาการ "มศว วิจัย" ครั้งที่ 8  
26-27 พฤศจิกายน 2557 มหาวิทยาลัยศรีนครินทรวิโรฒ

ภาพที่ 2 ตัวอย่างภาพลายเซ็นที่อยู่ในรูปของตาราง และ ค่าความเข้มถูกแทนที่ด้วยตัวเลข 0 ถึง 255

หลังจากได้ข้อมูลในรูปแบบตัวเลขของ Array แล้วจึงทำการ export เป็น csv file เพื่อง่ายต่อการนำไปใช้ในขั้นตอนต่อไป ถึงตรงนี้เรามีข้อมูลทั้งหมด 5,472 คอลัมน์ 600 เรกคอร์ด จากนั้นก่อนจะไปสู่ขั้นตอนการหาโมเดลที่ดีที่สุด ผู้วิจัยได้ทำการสุ่มแบ่งข้อมูล 75% เพื่อใช้ในการ Train และอีก 25% ที่เหลือไว้ใช้ในการ Test โดยอัลกอริทึมที่นำมาใช้นั้นได้แก่ SVM (เคอร์เนลฟังก์ชันเชิงเส้น, ฟังก์ชันพหุนาม, ฟังก์ชันเรเดียลเบส และ ฟังก์ชันซิกมอยด์), เคเนียร์เรสนเนเบอร์, เพอร์เซปตรอน, นาอ็พไบเย่ (ฟังก์ชันสามชนิด คือ เกาเซียน มัลติโนเมียล และเบอร์นูลลี) รวมทั้งสิ้น 4 อัลกอริทึม ผลการทดลองที่ได้ก็แตกต่างกันออกไปตามแต่ละอัลกอริทึม โดยทางผู้วิจัยได้ทำการทดลองเพิ่มขึ้นอีกขั้น เพื่อเพิ่มความแม่นยำในการทำนาย โดยอาศัยหลักการ การเพิ่มชุดข้อมูลที่มีค่าความเข้มมาก (สนใจค่าความเข้มที่อยูระหว่าง 0 ถึง 200 เท่านั้น) จากนั้นนำชุดข้อมูลนี้เข้าไปต่อท้ายจากชุดข้อมูลเดิม ซึ่งจะช่วยให้ข้อมูลลายเซ็นแต่ละ Array มีข้อมูลเพิ่มขึ้น จาก 5,472 คอลัมน์ เป็น 6,189 คอลัมน์ และนำข้อมูลชุดใหม่ไปเข้าสู่ขั้นตอนการหาโมเดลที่ดีที่สุดอีกครั้งหนึ่ง (จากภาพที่ 3)

การประชุมวิชาการ "มศว วิจัย" ครั้งที่ 8  
 26-27 พฤศจิกายน 2557 มหาวิทยาลัยศรีนครินทรวิโรฒ



ภาพที่ 3 โครงร่างขั้นตอนการดำเนินงาน

**ผลการวิจัย และอภิปรายผล**

จากข้อมูลลายเซ็นทั้งหมด 600 ลายเซ็น จากบุคคล 30 คน เราได้ใช้ ภาษา Python2.7 เขียนด้วยโปรแกรม Spyder เพื่อทำการทำนายผล ซึ่งได้ผลลัพธ์ดังตารางที่ 1

ตารางที่ 1 ตารางแสดงค่าความแม่นยำของแต่ละอัลกอริทึม เปรียบเทียบระหว่าง 2 วิธีการ

Algorithm	Function	Traditional Method (5,472 columns)		Additional Intensive data Method (5,472+717 columns)		Improvement
		Accuracy		Accuracy		
		Training data	Test data	Training data	Test data	
Perceptron	linear	100%	68.67%	100%	69.33%	0.96%
SVM	linear	100%	94.67%	100%	96.67%	2.11%
	poly	100%	95.33%	100%	97.33%	2.10%
	rbf	100%	0.67%	100%	0.67%	0.00%
	sigmoid	4.22%	0.67%	4.22%	0.67%	0.00%
Naive Bayes	gaussian	100%	56.67%	100%	56.67%	0.00%
	multinomial	97.33%	82.67%	98.67%	89.33%	8.06%
	bernoulli	4.22%	0.67%	4.22%	0.67%	0.00%
kNN	k = 3	92.67%	78.67%	98.00%	93.33%	18.63%

การประชุมวิชาการ "มศว วิจัย" ครั้งที่ 8  
26-27 พฤศจิกายน 2557 มหาวิทยาลัยศรีนครินทรวิโรฒ

1. วิธีดั้งเดิม (Traditional Method) SVM เป็นอัลกอริทึมที่มีความแม่นยำในการทำนายมากที่สุดอยู่ที่ 95.33% (โดยใช้ เคอร์เนลฟังก์ชันพหุนาม) และ 94.67% (โดยใช้ เคอร์เนลฟังก์ชันเชิงเส้น) ซึ่งในอัลกอริทึมอื่น ๆ จะให้ค่าความแม่นยำที่ต่ำลง ซึ่งลดหลั่นกันไป เช่น มัลติโนเมียลนาอ์ฟเบย์ อยู่ที่ 82.67%, เพอร์เซปตรอน 68.67% และ เคเนียร์เรสเนเบอร์ คือ 78.67%

2. วิธีเพิ่มชุดข้อมูลที่มีระดับความเข้มข้น (Additional Intensive data Method) วิธีการนี้ทางผู้วิจัยได้ทำการคิดค้นวิธีการใหม่ ที่คาดหวังว่าจะทำให้การทำนายของแต่ละอัลกอริทึมมีค่าความแม่นยำมากขึ้นกว่าเดิม โดยการเพิ่มชุดข้อมูลที่มีความเข้มข้นค่อนข้างมากเพิ่มเข้าไปต่อท้ายกับชุดข้อมูลเดิม กล่าวคือสนใจระดับความเข้มข้นที่อยู่ระหว่าง 0 ถึง 200 เท่านั้น จากนั้นนำข้อมูลใหม่ผ่านการคัดกรองแล้วเข้าไปต่อท้ายจากชุดข้อมูล Array เดิมอีก 717 คอลัมน์ รวมกับของเดิม 5,472 คอลัมน์ เป็นทั้งสิ้น 6,189 คอลัมน์ ซึ่งผลการทดลองจากวิธีการใหม่นี้พบว่าอันดับหนึ่งที่มีความแม่นยำเพิ่มขึ้นสูงที่สุด คืออัลกอริทึม เคเนียร์เรสเนเบอร์ มีค่าความแม่นยำเพิ่มขึ้น จาก 78.67% เป็น 93.33% ซึ่งเพิ่มขึ้นถึง 14.66% ลำดับถัดมาคือ มัลติโนเมียลนาอ์ฟเบย์ มีค่าความแม่นยำเพิ่มขึ้นสูงรองลงมา จาก 82.67% เป็น 89.33% ซึ่งเพิ่มขึ้น 6.66%, ทางด้านอัลกอริทึม SVM ในส่วนเคอร์เนลฟังก์ชันเชิงเส้น และ เคอร์เนลฟังก์ชันพหุนาม ที่มีความแม่นยำมากอยู่แล้วนั้น พบว่าดีขึ้นจากเดิมเล็กน้อย จาก 94.67% เป็น 96.67% และจาก 95.33% เป็น 97.33% ตามลำดับ

```

signature_from_CSV.py
34 X_train, X_test, y_train, y_test = train_test_split(data, target, test_size=0.25, random_stat
35 # Perceptron
36 from sklearn.linear_model import Perceptron
37 from sklearn import metrics
38 clf = Perceptron(n_iter=100).fit(X_train, y_train)
39 y1_pred = clf.predict(X_train)
40 y2_pred = clf.predict(X_test)
41 acc1 = metrics.accuracy_score(y_train, y1_pred)
42 acc2 = metrics.accuracy_score(y_test, y2_pred)
43
44 print("=====")
45 print("Perceptron")
46 print("Accuracy[Training]: %.4f" % acc1)
47 print("Accuracy[Testing]: %.4f" % acc2)
48 print("=====")
49 # SVM
50 svm_mode = ['linear', 'poly', 'rbf', 'sigmoid']
51 for ex in svm_mode:
52     cffi = svm.SVC(kernel=ex)

```

Variable explorer

Name	Type	Size	Value
X_test	int32	(150, 6188)	array([[255, 255, 255, ..., 255, 255, 255], [255, 255, 255, ..., 0, 24 ...
X_train	int32	(450, 6188)	array([[255, 255, 255, ..., 255, 255, 255], [254, 255, 255, ..., 5, 184 ...
acc1	float64	1	0.04222222222222223
acc2	float64	1	0.006666666666666671
data	int32	(600, 6188)	array([[255, 255, 255, ..., 136, 134, 131],

Console

```

=====
SVM Mode: linear
Accuracy[Training]: 1.0000
Accuracy[Testing]: 0.9667
=====
SVM Mode: rbf
Accuracy[Training]: 1.0000
Accuracy[Testing]: 0.0067
=====
SVM Mode: poly
Accuracy[Training]: 0.0422
Accuracy[Testing]: 0.0067
=====
KNeighborsClassifier
Accuracy[Training]: 0.9000
Accuracy[Testing]: 0.9333

```

ภาพที่ 4 ตัวอย่างผลลัพธ์จากโปรแกรม ของเทคนิคใหม่ (Additional Intensive data Method)

จากผลการทดลองของงานวิจัยชิ้นนี้แสดงให้เห็นว่า การเพิ่มชุดข้อมูลที่มีความเข้มข้นเข้าไปต่อท้ายกับชุดข้อมูลเดิมนั้น ส่งผลให้การทำนายของหลายอัลกอริทึมมีค่าความแม่นยำสูงขึ้น โดยส่วนใหญ่จะพบว่ามีความแม่นยำสูงขึ้นกว่าเดิม มีเพียงบางส่วนของความแม่นยำยังคงเท่าเดิม ส่วนเคเนียร์เรสเนเบอร์ เป็นอัลกอริทึมที่เห็นได้ชัดเจนที่สุด หากใช้วิธีการเพิ่มข้อมูลที่มีความเข้มข้นเข้าไปนั้น ทำให้ค่าความแม่นยำสูงถึง 93.33% เมื่อเปรียบเทียบกับวิธีเดิมอยู่ที่ 78.67% ซึ่งเพิ่มสูงขึ้นมากที่สุด โดยคิดเป็น 18.63%

### สรุปผลงานวิจัย

งานวิจัยชิ้นนี้ได้เก็บรวบรวมลายเซ็นจากนักศึกษาที่มีการใช้งานลายเซ็นจริงในชีวิตประจำวัน ซึ่งถือได้ว่าเป็นแหล่งข้อมูลที่น่าสนใจสำหรับการทำงานวิจัย จากนั้นได้นำมาแปลงข้อมูลให้อยู่ในรูปของระดับความเข้มของแต่ละพิกเซล และได้ทำการเปรียบเทียบการทำนายการรู้จำลายเซ็นในแต่ละอัลกอริทึม ซึ่งพบว่าอัลกอริทึมที่มีความแม่นยำสูงสุดคือ SVM (ที่เคอร์เนลฟังก์ชันพหุนาม) ซึ่งมีค่าความแม่นยำอยู่ที่ 95.33% งานวิจัยนี้ยังรวมไปถึงการเพิ่มประสิทธิภาพในการทำนายของแต่ละอัลกอริทึม ซึ่งเป็นสิ่งที่มุ่งเน้นสำหรับงานวิจัยชิ้นนี้ วิธีการดังกล่าวคือ การเพิ่มชุดข้อมูลที่มีความเข้มสูงเข้าไปต่อท้าย ซึ่งให้ผลลัพธ์ที่น่าสนใจ โดยเห็นผลลัพธ์ได้จากซัพพอร์ตเวกเตอร์แมชชีน เคอร์เนลฟังก์ชันเชิงเส้น และพหุนาม ที่ให้ค่าความแม่นยำมากกว่าค่าที่มากที่สุดของวิธีดั้งเดิม ซึ่งค่าความแม่นยำที่มากที่สุดของวิธีการใหม่นี้สูงถึง 97.33%

### เอกสารอ้างอิง

- [1] Tom Michael Mitchell (1997). *Machine Learning*. Boston: McGraw-Hill.
- [2] Geoff Dougherty (2013). *Pattern Recognition and Classification*. New York: Springer.
- [3] J.A. Unar, Woo Chaw Seng, Almas Abbasi (2014, August). A review of biometric technology along with trends and prospects Original Research Article. *Pattern Recognition*. Volume 47, Issue 8: 2673-2688
- [4] Minsky M.; & Papert S. (1969). *Perceptrons*. Cambridge, MA: MIT Press.
- [5] Peter Flash. (2012), *Machine learning: The Art and Science of Algorithms that Make Sense of Data*. New York: Cambridge University Press.
- [6] Nello Cristianini.; & John Shawe-Taylor. (2000). *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. New York: Cambridge University Press.
- [7] Arya, S.; D. M. Mount.; N. S. Netanyahu.; R. Silverman.; & A. Y. Wu. (1998, November). An Optimal Algorithm for Approximate Nearest Neighbor Searching in Fixed Dimensions. *Journal of the ACM*, Volume 45 Issue 6: 891–923.

## A New Efficient Method to Improve Handwritten Signature Recognition

Anusara Hirunyanakul\*, Kedkarn Chaiyakhan, Ratiporn Chanklan, Kittisak Kerdprasop, Nittaya Kerdprasop

Data Engineering Research Unit, School of Computer Engineering, Institute of Engineering,  
Suranaree University of Technology, NakhonRatchasima 30000, Thailand.

\*Corresponding author: Anusara.cpe8@gmail.com

### Abstract

In this research we demonstrate the improvement for handwritten recognition using edge detection technique and our novel technique of adding intensive data. We collect totally 600 signatures from 30 people. Then we transform the hand written signatures images to be image file and resize to 144 x 38 pixels along the width and the height, respectively. Every pixel is encoded its intensity value from 0 to 255. The value 0 is the highest intensity (black) and 255 is white. Next, we use 4 different algorithms: Support Vector Machine (with linear, polynomial, radial basis, and sigmoid kernel functions), k-Nearest Neighbors, Perceptron, and Naïve Bayes (using Gaussian, multinomial, and Bernoulli density functions). From the experiment result, SVM with polynomial kernel function shows the highest accuracy (95.33%). Then we use 4 techniques of edge detection: Sobel, Prewitt, Robert, Canny and Thinning technique. With Sobel edge detection technique, we found that the accuracy is gained to 96% (higher than the highest of original data). We also observe that Sobel technique can improve the accuracy of k-NN with a significant level (from 78.67% to 91.33%). Moreover, we try to append the high intensity color data. And by this technique, we notice significant improvement of k-NN accuracy up to 96%. In SVM with linear function, after applying our technique the accuracy is improved to 98.00% which is the highest accuracy of this research.

**Keywords:** image recognition, handwritten signature, support vector machine, k-nearest neighbor, naïve Bayes

### 1. Introduction

The use of biometric in authentication or individual identification receives much attention in the current. It provides convenience of not having to carry identification

documents, which reduces the problem of document falsification. The signature is external identity which is widely used for identifying individual. Signature of a person is distinct and it is hardly to be forged or counterfeited.

Vargas et al.<sup>(1)</sup> reviewed the handwritten signatures focusing on the grey-scale measurement and co-occurrence matrix technique and local binary pattern base on MCYT-75 and GPDS-100 databases. The result was that the EER (Equal Error Rate) = 16.27%.

Guerbai et al.<sup>(2)</sup> proposed the powerful use of OC-SVM for handwritten signature verification. The result from the experiment was 5-7% AER (Average Error Rate) in CEDAR dataset, whereas 15 - 17% AER in GPDS dataset.

Frias-Martinez et al.<sup>(3)</sup> demonstrated the handwritten recognition based on the Support Vector Machine (SVM) and compared to a traditional classification technique like Multi-Layer Perceptron (MLP). The experimental results showed that SVM could provide up to 71% accuracy rate, which is better than the MLP technique.

Zheng et al.<sup>(4)</sup> conducted edges and gradients detection, which was an innovative method for finding clearer edges. They used Least Squares Support Vector Machine (LS-SVM) with radial basis kernel function and Sobel and Canny edge detection. The outcome revealed that these techniques were even more effective than applying only a single machine learning technique.

Most researches on signature recognition often focused on a comparative study to find algorithms suitable for the signature recognition. However, we often encounter problems of a similar signature of different individuals or slightly different signatures of the same individual. Due to various environmental conditions, the accuracy of signature identification turns derogated. We have realized the importance of pre-processing. It is the importance step that can affect the accuracy rate. This paper proposes a

importance of pre-processing. It is the importance step that can affect the accuracy rate. This paper proposes a technique to enhance the signature recognition by focusing the improvement of signature images. The signature images will be improved by edge detection technique and thinning edgetechnique. In addition, we propose a novel concept, which is never seen in any previous signature recognition researches; that is, to append the interesting area of imagedataset. And in this case is high intensity color data.

## 2. Related Theorems and Algorithms

### 2.1 Pattern Recognition

The pattern recognition<sup>(5)</sup> is the study about object classification with respect to "Feature" of each "Class." The method can be applied to various fields, for example, the individual identification using biological data, e.g. fingerprint, face, iris, DNA, or even a signature, as well as the recognition of documents, e.g. pattern recognition of spam mail.

Issues related to pattern recognition and classification have been of great interest at present. As a result, technology and various advanced tools have been developed to be applied. The classification often requires knowledge of various branches, e.g. data mining, artificial neural network, machine learning, data improvement processes such as image-data improvement by edge detection.

### 2.2 Image Edge Detection

The image edge detection<sup>(6)</sup> is used to detect lines showing around the shape of an object by cutting away any other details, e.g. color or streaked. The image used to represent the shape of the object is represented as a "Binary Image." Edge detection can be done in a variety of ways with similar principles; that is, to find the difference of color between the "Gray Scale" of one point and the other point. If the light intensity is very different, the edges will be clearer; however, if the color difference is less, the edges can be vague. The edge detection can be applied in computer vision, e.g. boundary separation between object and background or object recognition etc.

#### 2.2.1 Sobel Edge Detection

Sobel edge detection<sup>(6)</sup> is the edge detection method by using the 2 filters with the size 3 x 3 called "Sx" and "Sy" to separate objects and background. The gradient

values of each band will be computed and create the filters. Example of Sx and Sy filter are in figure 1.

$$\begin{array}{|c|c|c|} \hline -1 & 0 & 1 \\ \hline -2 & 0 & 2 \\ \hline -1 & 0 & 1 \\ \hline \end{array} \quad S_x \qquad \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 0 & 0 & 0 \\ \hline -1 & -2 & -1 \\ \hline \end{array} \quad S_y$$

Fig.1.Sobelfilter.

The calculation of sobel gradient value is shown in equation 1.

$$\text{Sobel gradient} = \sqrt{(S_x^2 + S_y^2)}(1)$$

#### 2.2.2 Prewitt Edge Detection

Prewitt edge detection<sup>(6)</sup> is detection technique using the same concept as the Sobel edge detection. The differential is the value in the filters that shown in figure 2.

$$\begin{array}{|c|c|c|} \hline -1 & 0 & 1 \\ \hline -1 & 0 & 1 \\ \hline -1 & 0 & 1 \\ \hline \end{array} \quad h_x \qquad \begin{array}{|c|c|c|} \hline -1 & -1 & -1 \\ \hline 0 & 0 & 0 \\ \hline 1 & 1 & 1 \\ \hline \end{array} \quad h_y$$

Fig.2.Prewitt filter.

Prewitt gradient can be calculated as shown in equation 2.

$$\text{Prewitt gradient} = \sqrt{(h_x^2 + h_y^2)}(2)$$

#### 2.2.3 Robert Edge Detection

Robert edge detection<sup>(6)</sup> is a technique using 2x2 size filters called "Gx" and "Gy". The concept of this edge detection is to calculate the gradient of an image which is summarized from the differences between diagonally adjacent pixels. The filter of Robert edge detection is shown in figure 3.

$$\begin{array}{|c|c|} \hline 1 & 0 \\ \hline 0 & -1 \\ \hline \end{array} \quad G_x \qquad \begin{array}{|c|c|} \hline 0 & 1 \\ \hline -1 & 0 \\ \hline \end{array} \quad G_y$$

Fig. 3.Robert filter.

Robert gradient can be calculated as shown in equation 3.

$$G(F(i,j)) = |f(i+1,j+1) - f(i,j)| + |f(i+1,j) - f(i,j+1)| \quad (3)$$

#### 2.2.4 Canny Edge Detection

The first step of Canny edge detection<sup>(6)</sup> is to eliminate noise. Noise can be removed by using Gaussian Filter to clear the speckles and smooth the edge of image. In the second step, a gradient operator will be applied to achieve the gradient's intensity and direction. Then, The non-maximum suppression is used for thinning the images' edge by determining if the pixel is a better candidate than its neighbors. The final step is using double thresholding algorithm to specify contour pixels and make the edge continuous.

The calculation of Gaussian Filter which is applied in Canny edge detection can be explained by equation 4.

$$S[i,j] = G[i,j,\sigma] * I[i,j] \quad (4)$$

where  $S[i,j]$  is image after filtered,

$I[i,j]$  is image for finding edge detection,

$G[i,j,\sigma]$  is Gaussian filter, and

$\sigma$  is a variable to control the distribution of smoothing the image.

The image  $S[i,j]$  is used in equation 5 and 6 to find the x, y partial derivatives  $P[i,j]$  and  $Q[i,j]$ . The size and direction gradient are explained in equations 7 and 8, respectively.

$$P[i,j] \approx (S[i,j+1] - S[i,j] + S[i+1,j+1] - S[i+1,j])/2 \quad (5)$$

$$Q[i,j] \approx (S[i,j] - S[i+1,j] + S[i,j+1] - S[i+1,j+1])/2 \quad (6)$$

$$M[i,j] = \sqrt{P[i,j]^2 + Q[i,j]^2} \quad (7)$$

$$\theta[i,j] = \arctan(Q[i,j]/P[i,j]) \quad (8)$$

#### 2.3 Thinning Edge

The edge thinning<sup>(6)</sup> is an important preparation process (pre-processing) that is widely used to slenderize image with thick edges, which is produced from the edge detection. This is to remove the excessively thick edge pixels. Usually, excessively thick edge slenderizing is used in character recognition and signature recognition to eventually generate thinnest edge lines with only one pixel.

The thinning edge operation can be done by using P1 and P2 filters. The first step in doing that is to use the P1 filter; using the 3x3 template to scan the image data and then decide whether or not the pixels around the edges can be deleted. If the pixels can be deleted, mark them but do not delete them yet. After scanning throughout the image, delete the marked pixels. In the final step, P2 filter is used as when using the P1 filter. After deleting the marked pixels, repeat these steps until no more image data can be deleted.

#### 2.4 Perceptron

Perceptron<sup>(7,8)</sup> is one of the most popular algorithms used in classification. This algorithm is based on basic linear function model to classify the data with centroid as representative population. The linear function model mechanism is to create a line connecting the centroid of two groups and then create a perpendicular line to break the groups apart. Perceptron employs this method of linear function model as an initiative separation line. After that, the algorithm will check for the misclassified data point. If a fault exists, the counterbalance to weight of that data point to achieve the accurate classification, or to achieve classification with least erroneous data.

Another advantage of the Perceptron is that we can tune the learning rate to determine for the algorithm's accuracy. If learning rate is too small value, the weighting in erroneous data will be also small and we will see gradual changes of the separation line. On the other hand, if the learning rate is too high in value, it will result in too aggressive changes that will affect other data and that data must be classified many times. This can be explained by figure 4, where  $\eta$  is the learning rate values.

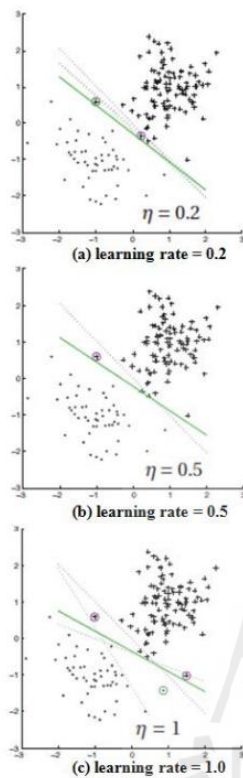


Fig. 4. The difference between 3 cases of learning rate in Perceptron algorithm<sup>(8)</sup>

## 2.5 Support Vector Machine

Support Vector Machine<sup>(7,10)</sup> or SVM is an algorithm based on a linear function model, which is developed from the Perceptron algorithm. It is a way to increase flexibility of classification to acquire large margin as much as possible. The concept of this algorithm is to place the data onto feature space and draw lines connecting the edges of each group. And then, the algorithm uses these data points on the edge to represent groups. The nearest data points of each group are called “support vector”. Then data separation lines of both groups are created to classify the data with the largest margins as possible, that shown in figure 5. In some cases, this algorithm can allow for misclassification to achieve the larger margin by using slack variable.

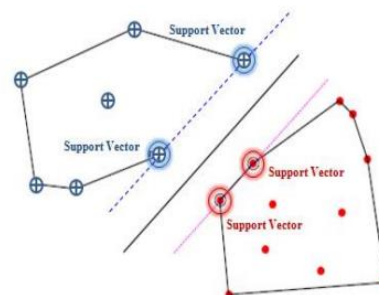


Fig. 5. The support vectors and classification in SVM algorithm

Another advantage of the Support Vector Machine is that the processing time is less than Perceptron algorithm because Support Vector Machine does not require all the data points to be calculated. In addition, the Support Vector Machine provides various kind of functions, called kernel, to fit a specific type of data distribution. These kernel functions include linear, polynomial, radial basis, and sigmoid.

## 2.6 Naïve Bayes

Naive Bayes<sup>(7-8,11)</sup> is an algorithm that uses the Bayes theorem to assist in classification. It is based on the assumption that the attributes of the sample are independent. The algorithm is suitable for the set of large sample. The modeling is in the form of conditional probability. The advantage of this method of learning is that we can use the data and “Prior knowledge” to help in learning. This algorithm gives good performance when compare with the other algorithms. In terms of the calculation, the principles of probability will be used and will be based on the theory of Bayes.

## 2.7 k-Nearest Neighbors

k-Nearest Neighbors<sup>(7,11)</sup> or k-NN is a popular classification algorithm in the field of pattern recognition. The concept of this algorithm is classifying the new data base on the k closest training examples. And the class of new data will be assigned by the majority class label of the k closest training data.



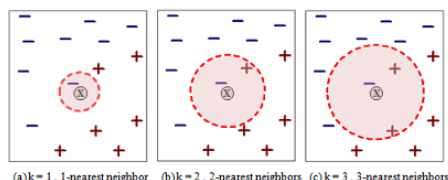


Fig. 6. The example of k-NN algorithm when k=1, 2 and 3

Figure 6 shows the classification of k-Nearest Neighbors with the different k values. Results will vary depending on the number of the closest k, for example when k=1 for the incoming data (represented by x) will be classified as (-). When k=2, the incoming data can be classified as either (-) or (+). When k=3, the incoming data will be classified as (+).

#### The objectives of this research are:

1) To study and compare the effectiveness of hand-written signature recognition models from 4 learning algorithms: Perceptron, SVM, Naïve Bayes, and k-NN.

2) To improve the accuracy of hand-written signature recognition model by using image improvement and by addition of high intensity data.

### 3. Research Framework

This research consists of 6 stages (and diagrammatically shown in figure 7) as follows:

1) Collecting 600 signatures from 30 university students who use the hand-written signature in daily life and turn to images file by scanning device. We adjust the image's color to black and white. Then, equalize their size. The raw data of all hand-written signature image files are stored at the main author's website:

<https://sites.google.com/site/nhinganusaracpesut/signature/datasets>

2) Using of edge detection technique and thinning edges to sharpen signature images.

3) Converting the data into a numeric table in accordance with color intensity, and then converting the numeric table to the array data.

4) Input the array data to the 4 learning algorithms to recognize correct owner of each signature. Studying and

comparing the performances and drawing the initial conclusions.

5) Adding each image data with high intensity area appending to the original image and choosing edge detection technique that gives highest accuracy in the image improvement process (the experimental results show that the Sobel edge detection technique gives highest precision).

6) Comparing the learning accuracy of models that are created from a training data set annotated with high intensity area. Then observe results and draw the conclusions.

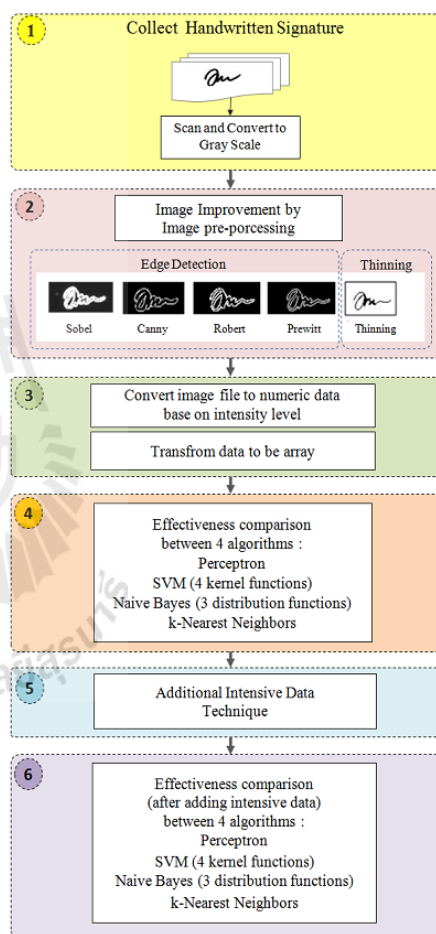


Fig. 7. Research framework

#### 4. Experimental Results

With all 600 signatures from 30 individuals, we use Python 2.7 Language on Editor Spyder to predict the results of signature recognition.

The image improvement at the second step of our proposed framework yields the results as shown in figure 8.

From table 1, the four algorithms used in this study include Perceptron algorithm, Support Vector Machine algorithm (Linear Function, Polynomial Function, Radial

Basis Function, and Sigmoid Function), Naive Bayes algorithm (Gaussian Function, Multinomial Function, and Bernoulli Function), and k-Nearest Neighbors algorithm. These algorithms are used in comparative test. The results obtained indicate that, in using the original image files, the SVM-Polynomial Function provides the highest accuracy of 95.33% and we see 94.67% by SVM-Linear Function. The Naive Bayes algorithm with Multinomial Function gives 82.67% of accuracy, and the k-Nearest Neighbors algorithm gives 78.67% of accuracy.



Fig. 8. Example images after applying edge detection and thinning techniques to signature images of three persons

Table 1. Experimental results of signature recognition with image improvement techniques

Algorithm	Function						
		Original	Sobel	Prewitt	Robert	Canny	Thinning
Perceptron	linear	68.67%	86.67%	65.33%	72.00%	64.00%	45.33%
SVM	linear	94.67%	<b>96.00%</b>	82.00%	89.33%	83.33%	86.00%
	poly	<b>95.33%</b>	94.00%	2.00%	14.67%	2.00%	88.67%
	rbf	0.67%	0.67%	0.67%	0.67%	0.67%	0.67%
	sigmoid	0.67%	0.67%	0.67%	0.67%	0.67%	0.67%
Naive Bayes	gaussian	56.67%	20.67%	65.33%	72.00%	65.33%	71.33%
	multinomial	82.67%	89.33%	87.33%	86.67%	85.33%	75.33%
	bernoulli	0.67%	17.33%	68.67%	72.00%	67.33%	0.67%
kNN	k = 3	78.67%	<b>91.33%</b>	8.00%	28.00%	6.00%	9.33%

Table 2. Experimental results when using additional intensive data technique

Algorithm	Function	Original		Sobel Edge		Sobel Edge + Additional Intensive Data Method	
		Accuracy		Accuracy		Accuracy	
		Train	Test	Train	Test	Train	Test
Perceptron	linear	100%	68.67%	100%	86.67%	100%	88.67%
SVM	linear	100%	94.67%	100%	96.00%	100%	98.00%
	poly	100%	95.33%	100%	94.00%	100%	96.67%
	rbf	100%	0.67%	100%	0.67%	100%	0.67%
	sigmoid	4.22%	0.67%	4.22%	0.67%	4.22%	0.67%
Naive Bayes	gaussian	100.0%	56.67%	99.56%	20.67%	99.56%	21.33%
	multinomial	97.33%	82.67%	94.00%	89.33%	99.73%	92.67%
	bernoulli	4.22%	0.67%	49.78%	17.33%	48.44%	18.00%
kNN	k = 3	92.67%	78.67%	98.22%	91.33%	99.11%	96.00%

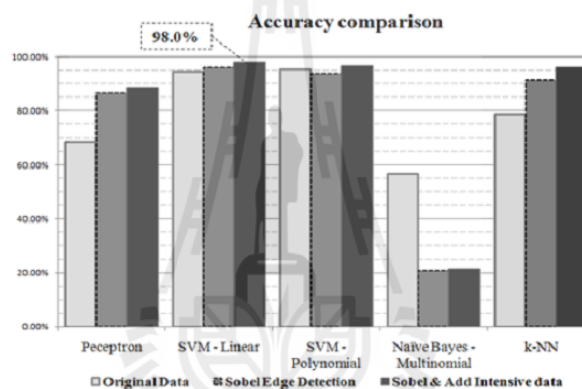


Fig. 9. Accuracy comparisons of original signature image data recognition, recognition from Sobel edge detected data, and recognition from both Sobel and additional intensive data techniques

The accuracy improvement of learning algorithms after using image processing with edge detection technique and thinning reveal that k-Nearest Neighbors algorithm's accuracy is increased by 12.66%; that is, from 78.67% to 91.33% with the use of Sobel edge detection technique. This increment is very significant. The accuracy of SVM-Linear Function is increased by applying the Sobel edge detection technique as well and the improved accuracy is 96.00%. Sobel edge detection is the best image processing technique applied prior to the signature recognition with learning algorithms.

Table 2 shows the results of using the additional intensive data technique with Sobel edge detection: SVM-Linear Function provides the accuracy of 98.00%,

which is the highest accuracy in this research. Moreover, we also found that the k-Nearest Neighbors algorithm provides higher accuracy by using the additional intensive data; that is the accuracy increases from 91.33% to 96.00% (which is higher than the maximum value of the original data).

The accuracy comparisons of signature image recognition without any other techniques, recognition with edge detection technique, and recognition with both edge detection and our additional intensive data techniques are shown in figure 9. It can be noticed that the combination of Sobel edge detection technique and our novel proposed additional intensive data technique yields the highest recognition rate at 98%.

## 5. Conclusions

We study the problem of handwritten signature recognition with the main objective of devising techniques to improve recognition accuracy rate. According to the signatures collected from hand-written users for this research, the SVM-Linear Function is the most suitable learning algorithm for modeling the signature recognition with the edge detection technique applied for image improvement and the additional intensive data technique newly proposed for accuracy improvement. This combination of edge detection and additional intensive data techniques provides the accuracy rate of up to 98.00%. For the technique of image improvement, the researchers note that we have possibility to achieve higher accuracy if we study more advanced techniques of image processing.

## Acknowledgment

This research has been supported by grant from the National Research Council of Thailand (NRCT) and Suranaree University of Technology through the funding of research unit.

## References

- (1) Vargas J.F., Ferrer M.A, Travieso C.M. and Alonso J.B.: "Off-line signature verification based on grey level information using texture features", *Pattern Recognition*, Vol.44, pp.375-385, 2011
- (2) Guerbai Y., Chibani Y., and Hadjadji B.: "The effective use of the one-class SVM classifier for handwritten signature verification based on writer-independent parameters", *Pattern Recognition*, Vol. 48, pp. 103-113, 2014
- (3) Frias-Martinez E., Sanchez A., and Velez J.: "Support vector machines versus multi-layer perceptrons for efficient off-line signature recognition", *Engineering Applications of Artificial Intelligence*, Vol. 19, No. 6, pp.693-704, 2006
- (4) Zheng S., Liu J., and Tian J. W.: "A new efficient SVM-based edge detection method", *Pattern Recognition Letters*, Vol. 25, pp. 1143-1154, 2004
- (5) Dougherty G.: "Pattern Recognition and Classification: An Introduction", Springer, 2012
- (6) Gonzalez R. C., and Woods R. E. : "Digital Image Processing", Prentice Hall, 2001
- (7) Theodoridis S., Pikrakis A., Koutroumbas K., and Cavouras D.: "An Introduction to Pattern Recognition: A MATLAB Approach", Academic Press, 2010
- (8) Flach P.: "Machine Learning: The Art and Science of Algorithms That Make Sense of Data", Cambridge University Press, 2012
- (9) Cristianini N., and Shawe-Taylor J.: "An Introduction to Support Vector Machines and Other Kernel-based Learning Methods", Cambridge University Press, 2000
- (10) Garreta R., and Moncecchi G.: "Learning scikit-learn: Machine Learning in Python", Packt Publishing, 2013
- (11) Aci M., Inan C., and Avci M.: "A hybrid classification method of k nearest neighbor, Bayesian methods and genetic algorithm", *Expert Systems with Applications* Vol. 37, pp. 5061-5067, 2010

## Fingerprint Recognition with Edge Detection and Dimensionality Reduction Techniques.

Ratipom Chanklan<sup>\*</sup>, Kedkarn Chaikyakhan, Anusara Hirunyanakul, Kittisak Kerdprasop, Nittaya Kerdprasop

Data Engineering Research Unit, School of Computer Engineering, Suranaree University of Technology,  
Nakhon Ratchasima 30000

<sup>\*</sup>Corresponding Author: arc\_angle@hotmail.com

### Abstract

In present an implementing fingerprint has been used widely. Such as fingerprint scanners first used your phones, used fingerprint scanner to save working hours. But the performance for implementing fingerprint also mistake. We assume that the fingerprint images to edge detection of the image to make the image more clearly, and may cause performance image recognition to better. We thus study the four edges detection: sobel, prewitt, robert and canny. For faster classification we apply two dimensionality reduction techniques: principal component analysis and linear discriminant analysis. Then, we identify biometrics with the algorithm support vector machine using kernel function linear. Experimental results showed that pre-processing fingerprint images using canny edge detection with principal component analysis the rate increased accuracy and highest from 64.3% to 88% and using canny edge detection with linear discriminant analysis the rate increased accuracy and highest from 73.8% to 88%

**Keywords:** fingerprint, edge detection, support vector machine, principal component analysis, linear discriminant analysis

### 1. Introduction

The combination of biological, medical and computer technology together by using the unique features of each person used to identify the person. And bring unique features to compare the unique features that have been recorded in the database for identify them from others. The system is called system identification biotechnology or biometric. The physical characteristics of the people do not change over time but the physical behavioral may have

changed. Thus identify the person by physical characteristics are reliable more than physical behavioral. In present fingerprint identification used to access devices such as cell phones is popular. The biometric system is not security enough, because the accuracy can't be a hundred percent. In the past, many researchers proposed edge detection techniques used to enhance the recognition. Improved performance the recognition by the application of wavelet transform with prewitt edge detection<sup>(1)</sup>. Have taken edge detection used with gray level watershed approach makes fast data classification and better performance<sup>(2)</sup>. Edge detection is important to recognize the fingerprint. It influences on the image extraction and affects the matching of images. The algorithm should be chosen according to the characteristics of the image is detected to be implemented perfectly<sup>(3)</sup>.

We thus propose edges detection, which is important in image processing and widely used to enhance recognition. We uses fingerprint images in the experiment. However, fingerprint images each individuals will look very similar it making classification difficult. First, we should be separation element of image. The element is maintaining characteristics of fingerprint. We used separation element of image by finding four edges detection method: sobel, prewitt, robert and canny. Because it make a fingerprint images look clear. And we apply two dimensionality reduction techniques: principal component analysis and linear discriminant analysis. Dimensionality reduction is making a classification less time and save a memory. Then dimensionality reduction image data into the algorithm that is used in the classification. We identify by support vector machine with linear kernel and compared the performance of each model.

## 2. Theories

### 2.1 Edge detection

Edge detection<sup>(4)</sup> is find line around the object in the image. When know line around the object we can calculate the area (size) or recognition type the object. However, finding correct image edge detection perfect not easy. In particular, find edge of image low quality or uneven light. Edges caused by the difference of light intensity from one point to another point. If has different of light intensity many the edges look clearly. If has the difference of light intensity less the edges is not clear. Edges detection has two main groups: Gradient method and Laplacian method.

In this study, we use four gradient methods: sobel edge detection, prewitt edge detection, robert edge detection and canny edge detection

#### 2.1.1 Sobel edge detection

Sobel<sup>(4)</sup> is find change of colors in image between object and background. The gradient value in each band is calculated by convolution image with filters size 3X3.

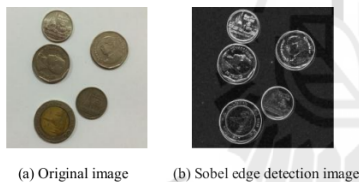


Fig. 1. Sobel edge image samples

#### 2.1.2 Prewitt edge detection

Prewitt<sup>(4)</sup> is calculated gradient vector of each all point on original image. The gray level intensity higher show border between object and background. Calculate gradient by filters size 3X3.

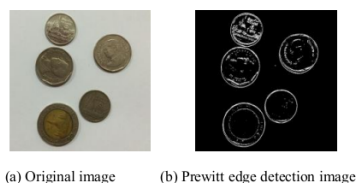


Fig. 2. Prewitt edge image samples

#### 2.1.3 Robert edge detection

Robert<sup>(4)</sup> is similar to the sobel edge detection. But using filters size 2X2.

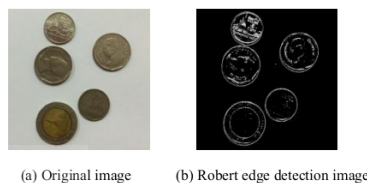


Fig. 3. Robert edge image samples

#### 2.1.4 Canny edge detection

Canny<sup>(4)</sup> is adjustments to smoothing image with Gaussian filter to remove noise in images. It makes finding edge clearly better. Then calculate magnitude size and orientation of gradient. Next step used Nonmaxima suppression with Gradient magnitude to make the edges thinner. And used double thresholding algorithm to identify edge pixels and helped connect edges consecutive.

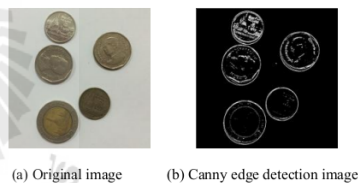


Fig. 4. Canny edge image samples

### 2.2 Dimensionality Reduction

Dimensionality reduction is action to replace data with compact data. It reduces memory usage and less time for classification. Then the results are equal or equivalent with a classification of all the data.

In this study, we use two dimensionality reduction techniques: principal component analysis and linear discriminant analysis.

#### 2.2.1 Principal Component Analysis (PCA)

PCA<sup>(5,6)</sup> is the techniques of multivariate data analysis without segmentation variables. Commonly used to reduce size the matrix of variables to small size

or find relations of data. It will create a new variable, which is made up of variable or the variance of original variables. Find relations data used to create the matrix of covariance from image data used to build an Eigen faces instead of the vector (Eigenvector).

### 2.2.2 Linear Discriminate Analysis (LDA)

LDA<sup>(7)</sup> is techniques used for supervised learning. It is commonly used for dimensionality reduction data and classifies data. It will be treat a number groups by will be maintained data as much as possible. But it will try to find the direction data to sorted out best. It is a function project onto a subspace in such a way data coming from different class are separate. And data same class are closer together to allow for easy classification. It consider a distribution within the group and distribution between the groups. And it can identify different picture because of other factors such as Lights and shooting characteristics.

### 2.3 Support Vector Machine (SVM)

SVM<sup>(8,9)</sup> is process of selecting the optimal model for the patterns in learning. Support vector machines popular in pattern recognition. The pattern recognition is use support vector machine for classification method. Support vector machines for classification use optimal hyperplane in classification. To build over planes can be created in various ways. But will have one optimal hyperplane can maintain the greatest distance between two groups. Find optimal hyperplane by find the location of the support vector is used for representative of the entire set. Which is used to divide the data by the principle is to take a plane has a many distance as possible between data two groups. Then find a plane at a safe distance from the edge (Maximum Margin) and assume that plane is plane best suited for classification.

We set  $(x_i, y_i), \dots, (x_n, y_n)$  when  $x \in R^m, y \in \{-1, 1\}$ ,  $n$  is number of data,  $m$  is dimension,  $x$  is data input and  $y$  is class -1 or +1. Creating a plane to split data can be calculated using Equation 1 and 2.

$$(w * x_1) + b > 0 \text{ where } y_i = +1 \quad (1)$$

$$(w * x_2) + b < 0 \text{ where } y_i = -1 \quad (2)$$

Consider a data set  $x$  by assigned  $x_1$  is data  $x_i$  where class +1,  $x_2$  is data  $x_i$  where class -1,  $w$  is weight and  $b$  is bias value.

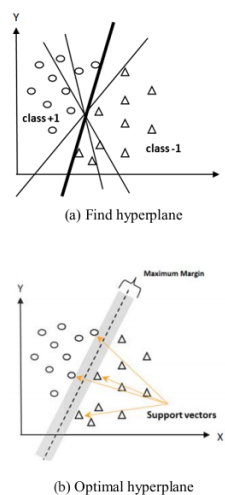


Fig. 5. Optimal hyperplane for classification

## 3. Research Process

This research has used Fingerprint data set from <http://fvs.sourceforge.net/download.html>. It have 168 fingerprint images as all 21 fingers, 1 finger have 8 images with a resolution of 80x80 pixels (6400 pixels).

In this paper using edges detection with dimensionality reduction to identification execution time. The process of implementation is shown in Figure 6.

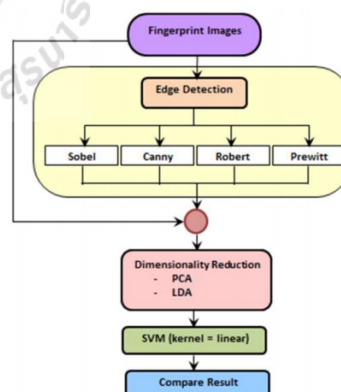
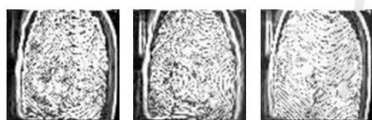


Fig. 6. The process of implementation.

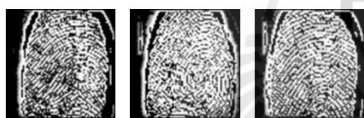
Figure 6 is a concept in research. In process finding fingerprint images edges detection 4 method: sobel, prewitt, robert and canny using matlab program and using python language by spyder editor for convert images edges data to array is level number color in each pixel, convert to array 1 dimensional, using dimensionality reduction techniques. In the last step using support vector machine algorithm to classify images data



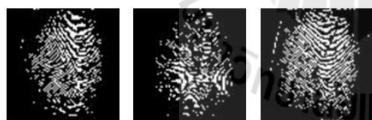
(a) Fingerprint image original samples.



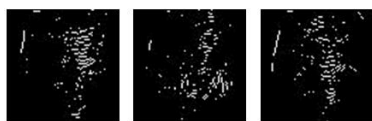
(b) Fingerprint image samples by sobel edge detection.



(c) Fingerprint image samples by canny edge detection.



(d) Fingerprint image samples by robert edge detection.



(e) Fingerprint image samples by prewitt edge detection.

Fig. 7. Fingerprint image samples.

## 4. Experimental Results

This research measure the performance of the model by comparing the accuracy of the data classification with support vector machine algorithm. The experimental results before and after applying by edges detection as follows in table 1, 2 and 3:

### 4.1 Fingerprint Recognition Accuracy of Non-edge versus Edges Detection

We split data by training data 126 images and test data 42 images. Then we use classification by support vector machine algorithm with linear kernel give which accuracy 52.4%.

### 4.2 Recognition Accuracy of Full Feature Fingerprint versus Dimensionality Reduction

Table 1. Before applying by edges detection

Dimension reduce	component	accuracy
LDA	32	73.80%
PCA	8	64.30%

When we use dimension reduction techniques, the classification appears to make increase accuracy. LDA increase accuracy from 52.4% to 73.80% and PCA increase accuracy from 52.4% to 64.30%.

### 4.3 Accuracy Improvement with Edges Detection and Dimensionality Reduction

Table 2. After applying by LDA with edges detection

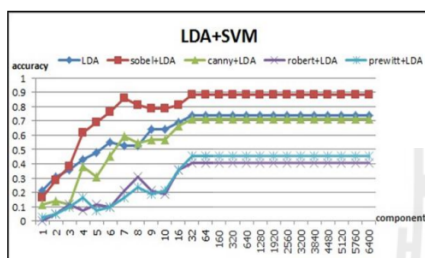
edge detection	component	accuracy
Sobel	32	88%
Canny	32	71.4%
Robert	16	40.5%
Prewitt	32	45.2%

Table 3. After applying by PCA with edges detection

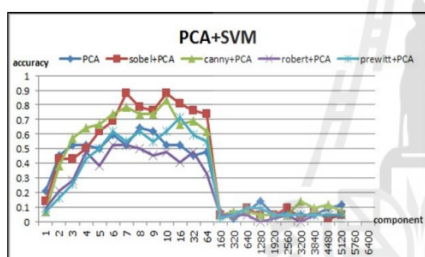
edge detection	component	accuracy
Sobel	7	88%
Canny	10	83.3%
Robert	6	52.5%
Prewitt	16	71.4%



We have shown results before and after edges detection with two dimensionality reduction techniques: principal component analysis (PCA) and linear discriminant analysis (LDA) in a graph in Figure 8.



(a) This graph shows the accuracy with the components by edges detection and LDA.



(b) This graph shows the accuracy with the components by edges detection and PCA.

Fig. 8. Graph shows all results.

Figure 8 is a graph showing the accuracy with the components. We are interested in the most accuracy to compare with edges detection method and non-edges detection method. Non-edges LDA + SVM is the most correct 73.8% after using edges sobel + LDA + SVM model increase accuracy is 88%. Non-edges PCA + SVM has the highest accuracy of 64.3% after using edges sobel + PCA + SVM model increase accuracy is 88%. The accuracy is increased after finding edges detection the fingerprints clearly. Will be seen from Figure 7 a sample fingerprint image before the edge image is then compared to using edges detection in sobel edge detection is most clear and images is gray scale, when using dimensionality reduction is increasing the accuracy and highest.

#### 4 Conclusions

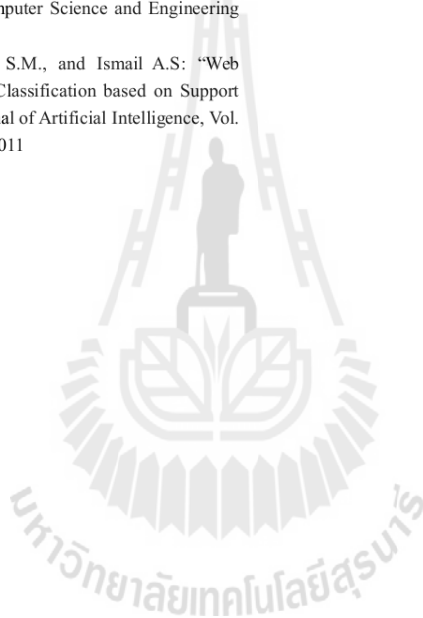
The fingerprint recognition technology is comfortable. Adopting fingerprints used for identification before entering the mobile devices are used widely. Which fingerprint images have splice many line, if using edges detection before classification by the algorithm may be effective in identifying increased. We using four edges detection: sobel, prewitt, robert and canny. Use dimensional reduction techniques to help identification execution time two techniques: principal component analysis and linear discriminant analysis. And using support vector machine kernel linear for measure the performance of the model. Adopting fingerprint image size 80x80 pixels (6400 components) of 168 images to identify people 21 people. The experimental results LDA + SVM model has the highest accuracy 73.8% after using edges detection sobel + LDA + SVM model increased accuracy to 88%. And PCA + SVM model has the highest accuracy 64.3% after using edges detection sobel + PCA + SVM model accuracy increased to 88%. The results can be summarized as sobel edge detection when used in conjunction with dimensional reduction techniques: principal component analysis or linear discriminant enhance the recognition of fingerprint image.

#### References

- (1) Zhou Weina, Han Jun, Zeng Xiaoyang and Yan Weiwei : "Fingerprint verification based on wavelet and edge detection", 9<sup>th</sup> International Conference on Electronic Measurement & Instruments, Vol 3, pp. 1001-1004, 2009
- (2) G.Sambasiva Rao, C. NagaRaju, Dr.L.S.S. Reddy and Dr.E.V. Prasad : "A Novel Fingerprints Identification System Based on The Edge Detection", Journal of Computer Science and Network Security, Vol. 8, No.12, pp. 394-397, 2008
- (3) Wei Cui, Guoliang Wu, Rongjin Hua and HaoYang: "The research of edge detection algorithm for Fingerprint images", Automation Congress (WAC 2008), Vol. 1, pp. 1-5, 2008
- (4) Gonzalez, R. C., and Woods, R. E : "Digital Image Processing", Addison-Wesley, 1992
- (5) Dinh Minh Nhat, V., and Young Lee, S: "Two dimensional Weighted PCA algorithm for Face Recognition", IEEE International Symposium on

Computational Intelligence in Robotics and Automation. pp. 219-223, 2006

- (6) Turk, M., and Pentland, A: "Eigenfaces for recognition", Journal of Cognitive Neuroscience. Vol. 3, pp. 71-86, 1991
- (7) Balakrishnama, S., and Ganapathiraju, A: "Linear discriminant analysis: a brief tutorial", Mississippi State (MS): Institute for Signal and information Processing, Mississippi State University, 1998
- (8) Pornpon Thamrongrat, Ladda Preechaweerakul and Wipada Wetpasit: "Web Page Classification Using Document Frequency and Support Vector Machine", The 12th National Computer Science and Engineering Conference, 2008.
- (9) Ali, W., Shamsuddin, S.M., and Ismail A.S: "Web Proxy Cache Content Classification based on Support Vector Machine", Journal of Artificial Intelligence, Vol. 4, No.1, pp. 100-109, 2011



## Traffic Sign Classification using Support Vector Machine and Image Segmentation

Kedkarn Chaiyakhon\*, Anusara Hirunyanakul, Ratiporn Chanklan,  
Kittisak Kerdprasop, Nittaya Kerdprasop

Data Engineering Research Unit, School of Computer Engineering, Institute of Engineering,  
Suranaree University of Technology, Nakhon Ratchasima 30000, Thailand.  
Rajamangala University of Technology Isan, Nakhon Ratchasima 30000, Thailand.

\*Corresponding Author: kedkarnc@hotmail.com

### Abstract

Traffic sign classification system is a part of driving assistance system that automatically alerts and instructs the driver about the meaning of traffic signs. In this paper, we proposed the idea for classifying automatically each type of traffic signs. The proposed method has been tested with a publicly available dataset: German Traffic Sign Recognition Benchmark (GTSRB). We use 360 images for testing by grouping traffic signs into 12 groups and images are taken by short distance from camera. In the first stage of our method, we have to separate particular traffic sign from background using color analysis, edge detection, and region of interest (ROI). In the process of ROI, we use Hough Transform algorithm to detect the different shapes of circle, square, and triangle. After we obtained edge images and images from ROI, in the subsequent stage we use 75% of images for training and 25% of images for testing using support vector machine classification algorithm. From ROI image we use feature extraction to get normal direction from edge image and also use support vector machine (SVM) to classify and compare results between ROI and ROI with normal direction feature. In our experimental results, using ROI and SVM can improve the efficiency of classification which has the accuracy 73%, ROI with normal direction feature and SVM can improve the efficiency of classification which has the accuracy 88.60%, when compare with the original involved background 67.80%. The improvement from SVM is about 5% and 20.8% respectively.

**Keywords:** image classification, support vector machine, hough transform, feature extraction, ROI.

### 1. Introduction

Image segmentation is the important process to subdivide an image into several regions. It is an important topic in computer vision and image processing. The result of segmentation is objects that are separated from background. The separated objects can decrease processing time and processing steps. Nowadays image segmentation has been widely used in several kinds of applications such as industrial images, geography and traffic safety. In this paper, we propose the experiment results of classification for several kinds of traffic signs by performing segmentation using image processing technique and classifying with support vector machine.

Traffic sign was designed for human to recognize and interpret the meaning of the abstract sign which using uncomplicated shape, character and symbol. The development of traffic sign recognition has been used for automatic driven vehicle or warning system for driver. Then it is an interesting task to make the computer identify, distinguish, and recognize the traffic sign correctly.

Color space is primarily used to obtain image information and detection. The Hue-Saturation-Value<sup>(1,2)</sup> (HSV) color space has been the most popular one as it based on human color perception. In addition it is considered largely invariant to illumination changes and also used to classify traffic sign by color separation. Many researchers use Red-Green-Blue (RGB) color space<sup>(3)</sup> to get image information and then convert it to a gray scale image to find edge contour of traffic sign region. However RGB is not a good representation with color separated between

objects and background. Thus, the conversion of RGB to gray scale image is required and then uses gray scale image to find contour edge. The contour edge can describe the shape properties of object and finally the position of object can be derived.

Currently, there are many researches in traffic sign segmentation and recognition that using several algorithms. Amplitude segmentation methods is the simplest method to separate the region of traffic sign using a proper threshold value, but the result is not good since the intensity of image and background are similar. Boundary detection method<sup>(4)</sup> is rather be popular and get good efficiency result of image segmentation is edge detection. This method takes quite a long time in processing, because after obtaining edge contour, the next step is to find the arbitrary shape in image.













## 2. Related Work

Many methodologies and researches have been proposed for traffic sign classification system. Sekanian et al.<sup>(5)</sup> proposed the detection of Norwegian speed limit signs. They design for 3 steps. Color classification using HSV color space, position detection of speed limit sign using template matching, and speed limit number recognition using neural network. Liu et al.<sup>(6)</sup> proposed vision-based stop sign detection and recognition systems for intelligent vehicle based on HSV color space, size analysis, and ratio of traffic sign. They used neural network technique in recognition process. Huda et al.<sup>(7)</sup> proposed real-time detection and recognition of Indian traffic signs using matlab. They also used gaussian filter and median filter to remove noise in image and used YCbCr color space in segmentation process. In the next step they used feature extraction before classification and recognition using multi-layer perceptron neural network method. Another method has also been proposed by Surinwarangkoon et al.<sup>(8)</sup> based on thresholding. They use RGB color space in segmentation process. They also use feature selection before classification process in traffic sign recognition system in poor condition.

## 3. Characteristics of Traffic Sign

Since traffic signs provide important information for driver about caution and hazard, their color and shape are chosen by traffic authorities to make them easy to understand by humans. In this paper, we selected traffic signs from German Traffic Sign Recognition Benchmark

Table 1. Different categories of traffic signs from GTSRB.

Traffic Sign Types	Image	Traffic Sign Types	Image
Speed limit (30 km/h)		Speed limit (120 km/h)	
Speed limit (50 km/h)		No overtaking	
Speed limit (60 km/h)		No overtaking by lorries	
Speed limit (70 km/h)		Right of way at the next crossroads	
Speed limit (80 km/h)		Priority road	
Speed limit (100 km/h)		Give way	

(GTSRB). The dataset used in our experiment contains 360 images, which are separated into 12 categories. There are mainly 3 shapes of traffic signs in our experiment such as circle, square, and triangle. Short distance photos were taken in shady, normal and bright light conditions by camera for every image. The size of the traffic signs were normalized to 80 X 80 pixels for size testing of signs. Different category of traffic signs from GTSRB are shown in Table 1.

## 4. Materials and Methods

### 4.1 Canny Edge Detection

In this paper we use canny edge detection to compute gradient direction of image. Canny edge detection enhances the image using smoothing process by Gaussian filter to blur image for eliminating noises in image. Then compute the gradient magnitude and gradient orientation using first-order derivative. Next step is to use non-maxima suppression and gradient magnitude to find thinnest contour. The last step is to use double thresholding algorithm to specify contour pixels and make the contour continuously.

Noise reduction is the first step using Gaussian filter.

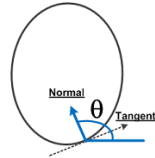


Fig. 1. Normal direction of edge pixel.

Given that  $I[i,j]$  is the image which require edge contour,  $G[i,j, \sigma]$  is Gaussian smoothing filter, and  $\sigma$  is the variable to control the distribution of smoothing. The smoothing process is as follows:

$$S[i, j] = G[i, j, \sigma] * I[i, j] \quad (1)$$

When smooth the image with noise reduction, the next step is to take the  $S[i, j]$  image to compute  $x, y$  partial derivatives  $P[i, j]$  and  $Q[i, j]$  respectively

$$P[i, j] \approx (S[i, j+1] - S[i, j] + S[i+1, j+1] - S[i+1, j])/2 \quad (2)$$

$$Q[i, j] \approx (S[i, j] - S[i+1, j] + S[i, j+1] - S[i+1, j+1])/2 \quad (3)$$

In the final step  $P[i, j]$  and  $Q[i, j]$  can be compute by converting rectangular coordinates to polar coordinates to find the size and direction of gradient.

$$M[i, j] = \sqrt{P[i, j]^2 + Q[i, j]^2} \quad (4)$$

$$\theta[i, j] = \arctan(Q[i, j], P[i, j]) \quad (5)$$

Thus the normal direction of edge pixel can be compute in the process of Canny edge detection. These normal direction values are the feature extraction that has been used in image segmentation by support vector machine. Normal direction on each edge pixel shown in Fig.1.

#### 4.2 Region of Interest (ROI)

Region of Interest is the method to crop the specific region of object in the image by eliminating background. ROI can be obtained by using binary edge contour image to find the position of circle, triangle, and square in image. The Hough transform is the algorithm to find the position of object. Since this paper uses 3 shapes of traffic sign, thus 2 methods of Hough transform algorithm were used to find sign position: Hough transform for circle detection and Hough transform for line detection to detect triangle and square.

##### (a) Hough transform for circle detection

Circle detection using Hough transform is the method

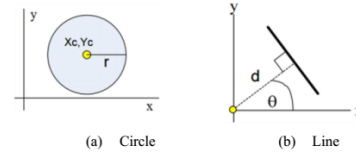


Fig. 2. Coordinate of circle and line.

for circle detection in image. Every edge pixel in binary image was substituted in geometric circle equation called Hough space to obtain a center of circle and radius in an image using maximum vote value of  $Xc, Yc$ , and  $r$ .

$$Xc = x + r \cos \theta \quad (6)$$

$$Yc = y + r \sin \theta \quad (7)$$

Where  $Xc$  and  $Yc$  are the possible center point of circle (range of  $Xc$  is 1 to image width,  $Yc$  is 1 to image height),  $x$  and  $y$  are edge pixel coordinate,  $r$  is radius of circle (range of  $r$  is 1 to half diagonal distance of image), and  $\theta$  is possible angle when measure along  $x$  coordinate (0-360 degrees) as shown in Fig. 2.

##### (b) Hough transform for line detection

Line detection using Hough transform is the method for line detection in image. Every edge pixel in binary image was substituted in geometric line equation called Hough space to derive the maximum vote of each line. If there are 3 maximum votes and 3 intersection in Hough space the object is triangle. Likewise, if there are 4 maximum votes and 4 intersection in Hough space the objects is square.

$$d = x \cos \theta + y \sin \theta \quad (8)$$

Where  $d$  is distance from the origin point when perpendicular measuring from specific line,  $\theta$  is possible angle (0-360 degrees) of line when dragging from origin point to perpendicular with line along  $x$  coordinate,  $x$  and  $y$  are coordinates of edge pixel as shown in Fig. 2.

#### 4.3 Support Vector Machine (SVM)

In this section we introduce some basic concepts of SVM and different types of kernel function.

##### (a) Overview of SVM

Support vector machine is supervised machine learning algorithm used for classification and regression problems by generating the optimal separation hyperplane in a multi-dimensional space. The basic idea

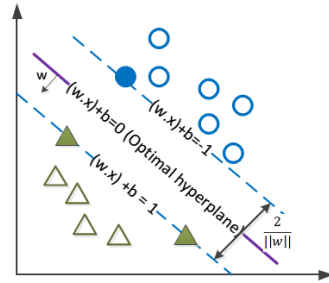


Fig. 3. Optimal hyperplane with maximum margin.

of SVM is to find the optimal hyperplane to separate datasets. Two parallel hyperplanes are constructed on each side of the hyperplane that separate the data. The separating hyperplane is the hyperplane that maximizes the distance between the two parallel hyperplanes. An assumption is made that the larger of margin or distance between these parallel hyperplanes, the better of classification as shown in Fig. 3.

We consider 2 datasets of the form

$$D = \{ (x_1, y_1), \dots, (x_l, y_l) \}, x_i \in R^m, y_i \in \{-1, 1\} \quad (9)$$

Where  $l$  denote the amount of dataset,  $i$  denote the sequence of data,  $m$  is number of dimensions, and  $y$  is classification result consisting of +1 and -1 to denote each group of dataset after separation process.

If the training data are linearly separable, we consider the dataset by specifying  $x_1$  to be data of  $x_i$  and value is positive. Likewise  $x_2$  is data of  $x_i$  and value is negative. This can be written as

$$\begin{aligned} (w * x_1) + b &> 0 \text{ where, } y_i = +1 \\ (w * x_2) + b &< 0 \text{ where, } y_i = -1 \end{aligned} \quad (10)$$

Where  $w$  denotes weight of vector,  $x_1$  is positive data vector,  $x_2$  is negative data vector, and  $b$  denotes bias value.

#### (b) Kernel Function of SVM

Training vectors can be represented in a higher dimensional space. There are many kernel functions available in SVM. So how to select an optimal kernel function is also a research problem. However, for general purpose, there are some popular kernel functions:

- Linear kernel:

$$K(x_i, x_j) = x_i^T x_j \quad (11)$$

- Polynomial kernel:

$$K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \quad \gamma > 0 \quad (12)$$

- RBF kernel:

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \quad \gamma > 0 \quad (13)$$

- Sigmoid kernel:

$$K(x_i, x_j) = \tanh \gamma x_i^T x_j + r \quad (14)$$

Here,  $\gamma$ ,  $r$  and  $d$  are kernel function parameters.

## 5. Proposed Work

A framework of proposed work for segmentation and classification of traffic sign is shown in Fig. 4. The main stages of our implementation are: image pre-processing using edge detection, image pre-processing using Hough transform to get region of interest (ROI), and classification process using support vector machine with different kinds of kernel functions.

### 5.1 Image Pre-processing using Canny Edge Detection

In Canny edge detection process; the first step is to decrease the dimensions of color space by converting color image to gray scale image to the remaining of only 256 colors, since much color is not necessary to find edge. Then the Canny edge is detected from gray scale image, as shown in Fig. 5

### 5.2 Image Pre-processing using Region of Interest

In the process of finding region of interest (ROI) by eliminating background, there are two main steps to detect region of traffic sign: convert color image to gray scale image and edge detection using canny edge detection, because it can remove noise from image by blurring to improve the quality of edge image. After that, the edge image is then sent to Hough transform to detect circle, triangle, and square. So, we can obtain the region of interest of traffic sign and then in the final step, region of interest has been sent to classification process using support vector machine as shown in Fig. 6.

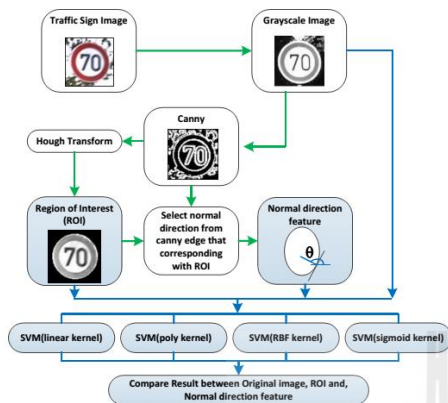


Fig. 4. A framework of proposed work.



Fig. 5. Canny edge image.

### 5.3 Feature Extraction

After we obtained the edge image we also obtained the normal direction feature. The normal direction of gradient or normal vectors are derived from the line that drags to touch every edge pixel from edge image which we call tangent vector, and then draw the perpendicular line with tangent vector. We will get the normal direction degree when measure angle along x axis. The normal direction in every edge pixels are the feature that will use in SVM classification process. Because normal direction feature can represents the shape and curve of object in degree of angles (0-360). Normal direction feature is shown is Fig.7.

### 5.4 SVM Classification

In SVM training process, we have 3 models for training images including original images that are converted to gray scale, ROI and, ROI with normal direction feature. All 3

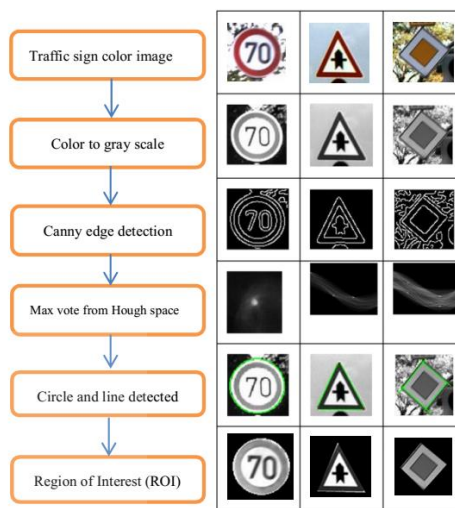


Fig. 6. Find ROI using Hough transform.

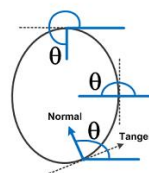


Fig. 7. Normal direction feature on every edge pixel.

models were trained from 270 images. In classification evaluation process, 90 images were used for testing. These tested images had been performed with image processing steps the same as those in the training process. Finally, we use 4 kernels functions (Linear, Polynomial, Sigmoid, and RBF) in SVM classification process.

## 6. Experimental Results

In this paper, we use 360 images from 12 categories and 3 different shapes such as circle, triangle, and square.

We use Matlab for pre-processing images and Python 2.7 use for the classification process. SVM is the efficiency method that has been widely used for classification and we also use 4 kernel functions in classification process as shown the experimental results in Table 2.

Linear kernel function gives rather high accuracy when

compared with other kernel functions. The accuracy rates are ranging between 67.80% - 88.60%. But original image gives low accuracy.

Polynomial kernel function obtains medium accuracy between 46.70% - 54.50%. Original image gives low accuracy for this kernel function.

RBF and sigmoid kernel function get very low accuracy in all methods of pre-processing image. The accuracy rates are between 4.40% - 28.70%.

The accuracy of classification when using ROI with linear kernel function is 73% and the accuracy of classification when using ROI with normal direction feature is 88.60%. The results of Polynomial, RBF and Sigmoid kernel functions are rather low as shown in Table 2.

In Table 2, we found that the classification accuracy of linear kernel function with region of interest (ROI) and region of interest with normal direction feature techniques is rather good on 3 methods of image pre-processing. Accuracy of original image is 67.80%, ROI 73% and, ROI with normal direction feature 88.60%. A comparative graph of classification accuracy using image pre-processing techniques with 4 kernel functions is shown in Fig. 8.

## 7. Conclusions

Traffic signs classification using support Vector machine and image segmentation is the main contribution presented in this paper. The image segmentation algorithm namely region of interest (ROI) and region of interest (ROI) with normal direction feature using SVM linear kernel function is quite efficient for classification. It can improve the accuracy of classification from 67.80% to 73% and 88.60%; the increase is about 5% and 20.8%. This increase is due to the fact that ROI will crop image only the region of traffic signs and eliminate the unnecessary background. We also use feature extraction with normal direction and bring the results to compare with ROI and original image. The accuracy of original image when used in classification process is not good. We found that the classification result using original image get low accuracy because original image consists of noisy background. On the other hand, ROI and ROI with normal direction feature are the gray scale image which has possible 256 intensity values, and we crop the specific region of traffic sign and also eliminate noisy background from image. Hence, ROI and ROI with normal direction feature when using with SVM linear kernel function already shows good accuracy in terms of segmentation and classification.

Table 2. Classification results for each method.

SVM	Original		ROI		Normal direction	
	Accuracy		Accuracy		Accuracy	
	Train	Test	Train	Test	Train	Test
Linear	1.000	0.678	1.000	<b>0.730</b>	1.000	<b>0.886</b>
poly	1.000	0.467	1.000	0.511	0.958	0.545
RBF	1.000	0.067	1.000	0.067	1.000	0.365
Sigmoid	0.960	0.044	0.096	0.044	0.0975	0.287

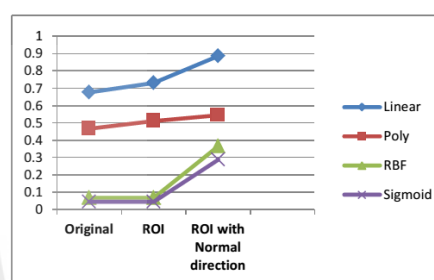


Fig. 8. Accuracy of classification using image.

In future work, we will mainly focus on improving the efficiency of the traffic sign segmentation and classification in order to use other information in image such as texture feature and morphological features. We will also apply feature extraction and feature selection before classification process to improve accuracy and reduce misclassification data.

## Acknowledgment

Financial assistance for Kedkarn Chaiyakhan has been provided by a scholarship from Rajamangala University of Technology Isan, Nakhorn Ratchasima, Thailand.

## References

- (1) G. Piccioli, E. De Micheli, P. Parodi, and M. Campani : "A robust method for road sign detection and recognition", *Image and Vision Computing*, Vol. 14, No. 1, pp. 209-223, 1996
- (2) X. W. Gao, L. Podladchikova, D. Shaposhnikov, K. Hong, and N. Shevtsova : "Recognition of traffic signs based on their colour and shape features extracted



- using human vision models”, *Journal of Visual Communication and Image Representation*, Vol. 17, pp. 675-685, 2006
- (3) A. de la Escalera, L. E. Moreno, M. A. Salichs, and J. M. Armingol : “Road traffic sign detection and classification”, *IEEE Transactions on Industrial Electronics*, Vol. 44, pp. 848-859, 1997
  - (4) T. Surinwarangkoon, S. Nitsuwat, and E. J. Moore : “Traffic sign recognition by color filtering and particle swarm optimization”, 4<sup>th</sup> International Conference on Computer Research and Development, pp. 55-59, 2012
  - (5) L. Sekanina, and J. Torrensen : “Detection of Norwegian Speed Limit Signs”, In Proc. Of the 16<sup>th</sup> European Simulation Multiconference, pp. 337-340, 2002
  - (6) X. Lui, and B. Ran : “Vision-based Stop Sign Detection and Recognition System for Intelligent Vehicle”, *Journal of the Transportation Research Board*, Vol. 1748, pp. 161-166, 2001
  - (7) H. Dean, and Jabir : “Real Time Detection and Recognition of Indian Traffic Signs using Matlab”, *International Journal of Scientific & Engineering Research*, Vol. 4, pp. 684-690, 2013
  - (8) T. Surinwarangkoon, S. Nitsuwat, and E. J. Moore : “Traffic Sign Recognition System for Roadside Images in Poor Condition”, *International Journal of Machine Learning and Computing*, Vol. 3(1), pp. 121-126, 2013



## ประวัติผู้เขียน

นางสาวอนุสรฯ หิรัญวานากุล เกิดเมื่อวันที่ 17 กันยายน พ.ศ. 2528 ที่จังหวัดแพร่ เริ่มเข้าศึกษาระดับชั้นอนุบาล 1 ถึงชั้นมัธยมศึกษาตอนต้นปีที่ 3 ที่โรงเรียนภัทรวิทยา อำเภอแม่สอด จังหวัดตาก จากนั้นได้เข้าศึกษาต่อในระดับมัธยมศึกษาตอนปลาย ที่โรงเรียนสรรพวิทยาคม อำเภอแม่สอด จังหวัดตาก ปีการศึกษา 2546 ได้เข้าศึกษาต่อระดับปริญญาตรีในสาขาวิชาวิศวกรรมคอมพิวเตอร์ สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี และสำเร็จการศึกษาเมื่อปี พ.ศ. 2549

ภายหลังสำเร็จการศึกษาในระดับปริญญาตรี ได้เข้าทำงานที่บริษัท VIZRT ในตำแหน่ง Broadcast System Engineer เมื่อปี พ.ศ. 2550 - 2551 จากนั้นได้เข้าทำงานที่บริษัท Hitachi Global Storage Technologies (Thailand) Ltd. ในตำแหน่ง Data Mining Expert Engineer เมื่อปี พ.ศ. 2551 - 2554 หลังจากนั้นได้เข้าทำงานที่บริษัท Seagate Technology (Thailand) Ltd. - Korat Plant ในตำแหน่ง Process Engineer เมื่อปี พ.ศ. 2554 - 2556

หลังจากนั้นได้เข้าศึกษาในระดับปริญญาโท สาขาวิชาวิศวกรรมคอมพิวเตอร์ สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี ในปี 2556

ในระหว่างการศึกษาได้รับความอนุเคราะห์อย่างยิ่ง จากอาจารย์ประจำวิชา Knowledge Discovery and Data Mining, Formal Methods and Computability, Data Structures and Algorithms ทั้งยังได้รับความไว้วางใจให้เป็นผู้ช่วยสอนปฏิบัติการ จากอาจารย์ประจำวิชา Database System และได้รับการตีพิมพ์เผยแพร่บทความวิชาการซึ่งรายละเอียดสามารถดูได้ที่ภาคผนวก ข