สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

รายงานการวิจัย

การแยกตัวอักษรติดกันในภาษาล้านนา

Segmentation of Touching Lanna Characters

รศ. ดร. อาริต ธรรมโน

ได้รับทุนสนับสนุนงานวิจัยจากเงินงบประมาณรายได้ ประจำปีงบประมาณ 2554

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

RCH
PN
A251
.N63
@655n

เลขหมู่.....
เลขทะเบียน.....131176
วัน,เดือน,ปี.....22 มี.ค. 2557

.b.....12602103
.i.....

ชื่อโครงการ การแยกตัวอักษรติดกันในภาษาล้านนา

Segmentation of Touching Lanna Characters

ได้รับทุนอุดหนุนการวิจัยจากเงินงบประมาณรายได้ ประจำปี 2554 จำนวนเงิน 50,000 บาท
ระยะเวลาทำการวิจัย 1 ปี ตั้งแต่ 1 ตุลาคม 2553 ถึง 30 กันยายน 2554

หน่วยงานและผู้ดำเนินการวิจัยพร้อมหน่วยงานที่สังกัดและเลขหมายโทรศัพท์
รศ. ดร. อาริต ธรรมโน
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
โทร. 02-723-4964

บทคัดย่อ

ในการรู้จำตัวอักษรเขียนขั้นตอนแรก คือขั้นตอนการประมวลผลภาพเบื้องต้น ในขั้นตอนนี้ปัญหาที่พบได้แก่ปัญหาตัวอักษรติดกัน ซึ่งจะมีผลต่อการเรียนรู้จำ ดังนั้นในงานวิจัยครั้งนี้ได้มุ่งเน้นในการแก้ปัญหาตัวอักษรติดกันของภาษาล้านนา โดยได้มีการพัฒนาอัลกอริทึมในการแยกตัวอักษรติดกันในภาษาล้านนา ขั้นแรก เริ่มต้นโดยการสแกนภาพเอกสารและทำการแปลงให้อยู่ภาพไบนารีจากนั้นได้ใช้บาวดิงบอกซ์ในการตัดตัวอักษรจากเอกสาร ใช้สเกลของตัวอักษรเดี่ยวในการแยกตัวอักษรที่ติดกันแล้วทำการสร้างโครงกระดูกของตัวอักษรที่ติดกัน จากนั้นทำการแยกตัวอักษรติดกันออกเป็นชิ้นย่อยๆ และเมื่อผ่านอัลกอริทึมที่พัฒนาแล้วก็นำเอาชิ้นส่วนของตัวอักษรย่อยๆ เหล่านั้นกลับเข้ามาประกอบเป็นตัวอักษรเดี่ยว โดยจากผลการทดลองมีความถูกต้อง 73.5 เปอร์เซ็นต์

Abstract

Character segmentation is an important preprocessing step for character recognition. Incorrectly segmented characters are not likely to be correctly recognized. Touching characters is one of the most difficult segmentation cases which arise when handwritten characters are being segmented. Therefore, this paper emphasizes the interest to the segmentation of touching and overlapping characters. In the proposed character segmentation process, the bounding box analysis is initially employed to segment the document image into images of isolated characters and images of touching characters. The thinning

algorithm is applied to extract the skeleton of the touching characters. Next, the skeleton of the touching characters is separated into several pieces. Finally, the separated pieces of the touching characters are put back to reconstruct two isolated characters. The proposed algorithm achieves an accuracy of 75.3%.

สารบัญ

หน้า

บทคัดย่อ I

สารบัญ III

สารบัญตาราง V

สารบัญรูป VI

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญของปัญหา 1

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา 2

1.3 ขอบเขตของการวิจัย 2

1.4 ประโยชน์ที่คาดว่าจะได้รับ 2

1.5 ขั้นตอนของการดำเนินการวิจัย 2

บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง 4

2.1 การแปลงภาพ 6

2.1.1 การแปลงภาพสี เป็นภาพสีเทา 6

2.1.2 การแปลงภาพสีเทา เป็นภาพขาวดำ 6

2.2 สมการถดถอย 8

2.3 เวกเตอร์ 10

บทที่ 3 โมเดลที่นำเสนอ 11

3.1 ขั้นตอนการแบ่งตัวอักษรล้านนาที่ติดกัน 11

3.1.1 การแปลงภาพสีไปเป็นภาพขาวดำ 11

3.1.2 การตัดตัวอักษรโดยการใช้ Bounding box analysis 13

3.1.3 ขั้นตอนการหาตัวอักษรติดกัน 13

3.1.4 การสร้างโครงกระดูกตัวอักษร 14

สารบัญ (ต่อ)

หน้า

3.1.5 ผลลัพธ์จากการแยกตัวอักษรติดกัน 16

3.2 อัลกอริทึม 16

3.2.1 การสร้างโครงกระดูกและหาจุดสิ้นสุดและจุดแยกตัวอักษร 16

3.2.2 การแบ่งภาพหลักให้เป็นภาพย่อย 17

3.2.3 การสร้างเวกเตอร์อ้างอิงหนึ่งหน่วย 18

3.2.4 การหาขนาดของรูปภาพย่อย 18

3.2.5 การสร้างเส้นตรงจากภาพย่อย 21

3.2.6 ผลลัพธ์ของการแยกตัวอักษรที่ได้จากอัลกอริทึม 22

บทที่ 4 การทดลองและผลการทดลอง 23

บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ 26

5.1 สรุปผลการวิจัย 26

5.2 ข้อเสนอแนะ 26

บรรณานุกรม 27

สารบัญตาราง

ตารางที่ **หน้า**

4.1 ภาพตัวอย่างผลลัพธ์ของการแยกตัวอักษติดกันที่ถูกต้อง.....25
4.2 ภาพตัวอย่างผลลัพธ์ของการแยกตัวอักษติดกันที่ไม่ถูกต้องและเป้าหมายที่ต้องแยก.....25

สารบัญรูป

รูปที่ หน้า

2.1 ฮิสโตแกรมแสดงค่าเธรดโชลด์ 8
3.1 ฮิสโตรแกรมที่แสดงระดับความเข้มของภาพ 11
3.2 การแปลงภาพสี (RGB Image) ไปเป็นภาพไบนารี (Binary Image) 12
3.3 ผลลัพธ์ของการตัดคำโดยการใช้ Bounding Box Analysis 13
3.4 ขั้นตอนการหาตัวอักษรติดกัน 14
3.5 ผลลัพธ์ของการสร้างโครงกระดูกของตัวอักษร 15
3.6 ผลลัพธ์ของการสร้างโครงกระดูกของตัวอักษร 15
3.7 ผลลัพธ์สุดท้ายของการแยกตัวอักษรติดกัน 16
3.8 ภาพ A จุดปลายและจุดทางแยกของตัวอักษรติดกัน
ภาพ B การแบ่งภาพตัวอักษรติดกันให้เป็นภาพย่อย 17
3.9 ผลลัพธ์ของคอนทัวร์ภาพย่อย 18
3.10 ภาพเวกเตอร์อ้างอิงหนึ่งหน่วยในแนวแกน x และแนวแกน y 18
3.11 ขั้นตอนการคอนทัวร์ของกิ่งย่อย 19
3.12 ขั้นตอนการหามุมในการคอนทัวร์แต่ละพิกเซล 20
3.13 ผลลัพธ์ของเวกเตอร์ของคอนทัวร์ในแต่ละกิ่งย่อย 20
3.14 ผลลัพธ์ของกิ่งย่อยหลังจากผ่านสมการถดถอย 21
3.15 ผลลัพธ์ของการหามุมระหว่างกิ่งย่อยและกิ่งจุดแยก 22
3.16 ผลลัพธ์ของการแยกตัวอักษรติดกันจากอัลกอริทึมที่ได้พัฒนาขึ้น 22
4.1 ภาพตัวอักษรล้านนา 6 รูปแบบที่ใช้ในการศึกษาวิจัย 24

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ภาษาล้านนาเป็นภาษาที่ใช้กันในอาณาจักรล้านนาในศตวรรษที่ 13-18 แต่หลังจากที่อาณาจักรล้านนาถูกยึดครองโดยอาณาจักรสยามในปี ค.ศ. 1774 ตัวอักษรล้านนาได้ถูกละเลยและถูกแทนที่ด้วยตัวอักษรไทยจนในปัจจุบันมีเพียงไม่กี่คนเท่านั้นที่รู้วิธีอ่านและเขียนภาษานี้ จากการสืบค้นข้อมูลทางประวัติศาสตร์พบว่าคนล้านนาจะใช้ใบลาน พับสาหรือศิลาจารึกในการจารึกอักษรล้านนาเพื่อบันทึกเกี่ยวกับวิถีชีวิต ความเชื่อ กฎหมาย เรื่องเล่า สูตรยาสมุนไพร ประวัติศาสตร์ โหราศาสตร์ และความรู้ทั่วไปอื่นๆ เมื่อกาลเวลาผ่านไป เอกสารโบราณเหล่านี้ได้ผุพัง เสียหาย ถูกทำลาย หรือสูญหายไป ดังนั้นเพื่อที่จะอนุรักษ์ข้อมูลทางประวัติศาสตร์อันล้ำค่าที่เอกสารเหล่านี้ได้บันทึกไว้ จึงควรมีการนำเอาระบบคอมพิวเตอร์มาใช้เพื่อแปลตัวอักษรเหล่านี้ให้เป็นตัวอักษรไทยที่ใช้กันในปัจจุบัน

จากการศึกษางานวิจัยต่างๆ ที่ผ่านมาพบว่าการพัฒนาอัลกอริทึมในการรู้จำตัวอักษร (OCR) จะประกอบไปด้วยขั้นตอน 3 ขั้นตอนได้แก่ การประมวลผลภาพเบื้องต้น การดึงคุณสมบัติของตัวอักษร และการเรียนรู้จำตัวอักษร ในขั้นตอนของการประมวลผลภาพเบื้องต้นจะเป็นกระบวนการที่ทำการปรับปรุงคุณภาพของรูปภาพตัวอักษรให้ดีขึ้นและทำการตัดตัวอักษรให้เป็นตัวอักษรเดี่ยวเพื่อเข้าสู่ขั้นตอนการดึงคุณสมบัติและขั้นตอนการเรียนรู้จำตัวอักษรต่อไป ในการตัดตัวอักษรนั้นปัญหาแรกที่มักจะพบคือปัญหาตัวอักษรสัมผัสกันโดยเฉพาะตัวเขียน ตัวอักษรล้านนาก็พบปัญหานี้เช่นกัน ดังนั้นการที่จะทำให้การรู้จำตัวอักษรมีประสิทธิภาพที่ดีจึงจำเป็นที่จะต้องแก้ไขปัญหาตัวอักษรติดกันเสียก่อน ในการศึกษาและการวิจัยในครั้งนี้ได้มุ่งเน้นการแยกตัวอักษรสัมผัสกันโดยได้มีการนำเอาศาสตร์ทางด้านปัญญาประดิษฐ์มาทำการพัฒนาในการแยกตัวอักษรที่สัมผัสกันและเป็นที่ยอมรับว่ามีประสิทธิภาพ ได้แก่อัลกอริทึมการเรียนรู้แบบมีข้อมูลในการสอน อัลกอริทึมการเรียนรู้แบบไม่มีข้อมูลในการสอน และอัลกอริทึมการใช้โครงสร้างของตัวอักษร มาช่วยในการแยกตัวอักษรที่ติดกัน

การศึกษาในครั้งนี้มุ่งเน้นการประยุกต์ใช้อัลกอริทึมการใช้โครงสร้างของตัวอักษร ในการแยกตัวอักษรที่ติดกันให้เป็นตัวอักษรเดี่ยว

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

เพื่อที่จะทำการพัฒนาวิธีที่เหมาะสมในการตัดแยกตัวอักษรที่สัมผัสกันของตัวอักษรล้านนาให้ออกเป็นตัวอักษรเดี่ยวได้อย่างถูกต้อง

1.3 ขอบเขตของการวิจัย

นำเสนอโมเดลสำหรับการแยกตัวอักษรที่สัมผัสกันของตัวอักษรล้านนา โดยทำการทดลองกับชุดข้อมูลภาพถ่ายตัวอักษรล้านนาที่บันทึกในใบลานของภาคเหนือตอนบน

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. มีความรู้ความเข้าใจอัลกอริทึมในการแยกตัวอักษรที่สัมผัสกัน

2. ได้โมเดลใหม่สำหรับการแยกตัวอักษรที่สัมผัสกันของตัวอักษรล้านนา ซึ่งจะนำไปสู่งานวิจัยด้านการรู้จำตัวอักษรล้านนาต่อไป

1.5 ขั้นตอนของการดำเนินงานวิจัย

1. ศึกษางานวิจัยทางด้านการแยกตัวอักษรที่สัมผัสกัน

2. ศึกษาทฤษฎีที่เกี่ยวข้องกับการพัฒนาโมเดลการแยกตัวอักษรสัมผัสกันที่มีประสิทธิภาพ และทดลองนำโมเดลเหล่านั้นมาตัดแยกตัวอักษรล้านนา

3. วิเคราะห์และออกแบบโมเดลใหม่

4. พัฒนาโปรแกรมตามรูปแบบโมเดลที่ได้ออกแบบไว้ โดยใช้โปรแกรม MATLAB

5. จัดเตรียมข้อมูลสำหรับการทดลองวิจัยแล้วดำเนินการทดลองเพื่อวัดประสิทธิภาพของโมเดลที่นำเสนอ

6. สรุปผลการทดลอง

7. จัดทำรายงานการวิจัย

บทที่ 2
หลักการและทฤษฎีที่เกี่ยวข้อง

การศึกษาอัลกอริทึมเรื่องการแยกตัวอักษรลายมือเขียนที่เขียนสัมผัสกันเป็นการหาจุดสัมผัสระหว่างตัวอักษรเพื่อหาจุดตัดในการแบ่งแยกตัวอักษรออกจากกันเพื่อให้ได้ตัวอักษรที่ถูกต้องและสมบูรณ์ ผู้วิจัยจึงได้ศึกษาอัลกอริทึมที่เหมาะสมกับการแยกตัวอักษรที่เขียนสัมผัส เช่นการนำโครงข่ายประสาทเทียม (neural network) ทำงานร่วมกับโปรเจ็คชั่นโปรไฟล์ในการแยกตัวอักษรภาษาไทยสัมผัส โดยขั้นตอนในการแยกตัวอักษรสัมผัสกันจะใช้วิธีการทำโปรเจ็คชั่นในแนวนอนก่อนเพื่อที่จะทำการแยกตัวอักษรระหว่างบรรทัด จากนั้นก็จะใช้วิธีการทำโปรเจ็คชั่นในแนวตั้งซึ่งจะได้ภาพที่เป็นฮิสโตแกรม จากนั้นทำการหาค่าโปรไฟล์ว่าจุดไหนที่มีค่าน้อยที่สุดระหว่างตัวอักษร ก็ทำการตัดแยกตรงจุดนั้น หลังจากนั้นทำให้ตัวอักษรที่ได้เป็นตัวผอมและแปลงเป็นข้อมูลเข้าสู่โครงข่ายประสาทเทียมเพื่อทำการพยากรณ์ว่าเป็นตัวอักษรอะไรโดยวิธีการดังกล่าวข้างต้นจะให้ความถูกต้องในการแยกตัวอักษรถึง 94.94 เปอร์เซ็นต์ งานวิจัยของ Rui และคณะ ได้นำเอาวิธี Drop-fall algorithm มาทำงานร่วมกับ Pattern-Oriented startegy ในการแยกตัวเลขลายมือเขียนที่สัมผัสกัน ในขั้นแรกจะทำการจัดกลุ่มลักษณะของตัวอักษรสัมผัสกันออกเป็น 2 กลุ่มคือกลุ่มแรกการสัมผัสกันแบบง่าย (Simple touching) คือมีจุดที่ต้องทำการแยกเพียงที่เดียว (single-segment touching) หรือมีจุดที่สัมผัสกันเพียงจุดเดียว (single-point touching) กลุ่มที่สองคือกลุ่มที่มีจุดสัมผัสกันหลายจุด (multiple touching) วิธีการนี้จะเริ่มต้นโดยทำการหาพื้นหลังระหว่างตัวอักษรโดยใช้วิธีการ water reservoirs concept จากตัวเลขที่สัมผัสกันจะทำให้มีพื้นที่ในการรองรับน้ำไม่ว่าจะเป็นปลายเปิดด้านบน ด้านล่างหรือด้านข้าง จากนั้นนำเอาลักษณะของรูปแบบการสัมผัสกันมาทำการแยกโดยการใช้ Descending-left algorithm, Descending-right algorithm, Ascending-left algorithm และ Ascending-right algorithm มาช่วยในการพิจารณาจุดแยกของตัวเลขซึ่งสามารถให้ความถูกต้องได้ถึง 95.60 เปอร์เซ็นต์ งานวิจัยของ Yang และคณะ ได้ทำการศึกษาและปรับปรุงระบบรู้จำตัวอักษรเขียนภาษาจีนที่มีลักษณะกลุ่มตัวอักษรที่ขาดจากกัน ลักษณะกลุ่มตัวอักษรที่ห่างกันและลักษณะ

กลุ่มตัวอักษรที่สัมผัสกันโดยการนำทฤษฎีฟัซซี่มาปรับใช้โดยกลุ่มของตัวอักษรที่มีลักษณะที่ขาดจากกันและกลุ่มตัวอักษรที่มีลักษณะตัวอักษรที่ห่างกันให้ทำการรวมตัวอักษรให้มีความถูกต้อง ในส่วนของกลุ่มตัวอักษรที่มีลักษณะที่สัมผัสกันให้ทำการแยกตัวอักษรออกจากกัน ขั้นแรกทำการแยกตัวอักษรแบบหยาบโดยการใช้วิธีการทางสถิติมาช่วยในการหาขนาดความกว้างและความสูงของตัวอักษร ขั้นตอนต่อมาทำการจัดกลุ่มของตัวอักษรที่ได้ออกเป็นกลุ่มตัวอักษรที่ขาดจากกัน กลุ่มตัวอักษรที่ห่างกันและกลุ่มตัวอักษรที่สัมผัสกัน จากนั้นนำค่าเทรสโฮลที่มีค่ามากที่สุดของแต่ละกลุ่มนำมาสร้างค่าความเป็นสมาชิกของแต่ละกลุ่ม (membership functions) และในขั้นตอนสุดท้ายเป็นการการนำเอาข้อมูลที่ได้ทำการทดสอบโดยใช้ fuzzy threshold method และทำการตรวจสอบความถูกต้องในแก้ปัญหาดังกล่าวโดยผู้ทำงานวิจัย ซึ่งผลที่ได้มีการแก้ปัญหาดังกล่าวดีขึ้น งานวิจัยของ Alamri และคณะ ได้ทำการแยกตัวอักษรเขียนที่เป็นตัวเลขอาระบิคที่สัมผัสกันและในส่วนของการรู้จำได้ทำการเลือกใช้ทฤษฎีซัพพอร์ตเว็คเตอร์แมชชีน (Support Vector Machines) ขั้นตอนแรกเริ่มต้นด้วยการแยกตัวอักษรเขียนที่เป็นตัวเลขอาระบิคสัมผัสกันโดยหาขนาดความยาวทั้งหมดของตัวอักษรที่สัมผัสกันทั้งสองตัวเลข แทนด้วยตัวแปร W_f จากนั้นใช้วิธีการทางสถิติเข้ามาช่วยโดยทำการสร้างค่าพารามิเตอร์ขึ้นมาสองค่าคือพารามิเตอร์สำหรับความยาวตัวอักษรตัวแรกแทนด้วย (α) และพารามิเตอร์สำหรับความยาวระหว่างตัวอักษรตัวแรกและตัวที่สองแทนด้วย (β) จากนั้นทำการคำนวณความยาวของตัว อักษรตัวแรกจากสมการ $W_f = \alpha \cdot W_i$ และความยาวของตัวอักษรที่สองจากสมการ $W_s = W_i - \alpha \cdot W_i - \beta$ จากนั้นนำตัวอักษรที่ได้ไปเข้าสู่กระบวนการดึงคุณสมบัติของตัวอักษรและทำการเรียนรู้จำโดยใช้ทฤษฎีซัพพอร์ตเว็คเตอร์แมชชีนทำให้ประสิทธิภาพในการรู้จำมีความถูกต้องถึง 98.48 เปอร์เซ็นต์เป็นต้น

ในงานวิจัยครั้งนี้ได้ศึกษาทฤษฎีพื้นฐานที่เกี่ยวข้องกับงานวิจัยได้แก่การประมวลผลภาพเบื้องต้น (Image Processing) การประมวลผลภาพถูกพัฒนาขึ้นมาเพื่อแก้ไขปัญหาเกี่ยวกับการมองเห็น ซึ่งการมองเห็นของมนุษย์เป็นสิ่งสำคัญและมีกลไกที่ซับซ้อนอย่างหนึ่ง การที่จะได้ผลลัพธ์ของการรู้จำให้มีประสิทธิภาพนั้นขั้นตอนนี้มีความสำคัญอย่างมากเพราะคุณภาพของภาพตัวอักษรที่จะนำไปเรียนรู้จำนั้นถ้าคุณภาพของภาพตัวอักษรดีผลลัพธ์ที่ได้ก็ดี ถ้าคุณภาพของภาพตัวอักษรไม่ดีผลลัพธ์ที่ได้ก็จะไม่ดีตามไปด้วย

2.1 การแปลงภาพ

การแปลงภาพเป็นขั้นตอนที่ใช้แปลงจากภาพสี ให้เป็นภาพขาวดำ การแปลงภาพสีเป็นภาพสีเทาทำได้โดยการแยกระดับสีแต่ละพิกเซลออกจากกัน ในรูปแบบภาพสีจากนั้นนำค่าจากภาพสีมาเข้าสู่สมการเพื่อคำนวณหาค่าสีเทา และนำค่าที่ได้ไปแทนในจุดพิกเซลเดิม

2.1.1 การแปลงภาพสี เป็นภาพสีเทา

การแปลงภาพสีเป็นภาพสีเทาคือกระบวนการในการแปลงภาพสีแต่ละพิกเซลให้กลายไปเป็นสีเทาและนำค่าที่ได้ไปแทนที่จุดพิกเซลเดิม โดยคำนวณได้จากสมการที่ 1 และสมการที่ 2 ดังต่อไปนี้

$$G' = 0.3R + 0.59G + 0.11B \quad (1)$$

หรือ

$$G' = \frac{R + G + B}{3} \quad (2)$$

โดยกำหนดให้

G' คือค่าระดับสีเทา

R คือค่าระดับสีแดง

G คือค่าระดับสีเขียว

B คือค่าระดับสีน้ำเงิน

2.1.2 การแปลงภาพสีเทา เป็นภาพขาวดำ

วิธีที่นิยมใช้แปลงภาพสีเทาเป็นภาพขาวดำนั้นวิธีการที่นิยมได้แก่ Otsu's method โดยอาศัยวิธีการทำเธรดโชลด์ ซึ่งเป็นวิธีที่ใช้ในการแปลงภาพสีเทาให้อยู่ในรูปแบบของภาพไบนารีทำให้สามารถ

จำแนกวัตถุและพื้นหลังออกจากกันได้ จากฮิสโตรแกรมภาพที่ 2.1 แสดงระดับความเข้มของภาพนั้นจะเห็นว่าส่วนที่เป็นวัตถุจะมีเข้มมากกว่าส่วนที่เป็นพื้นหลังทำให้สามารถแยกกลุ่มของวัตถุและพื้นหลังออกจากกันได้อย่างชัดเจน การเลือกค่าเธรดโชลด์ต้องทำการเลือกค่าที่มีระดับความเข้มอยู่ระหว่างกลุ่มที่เป็นวัตถุและกลุ่มที่เป็นพื้นหลัง

จากนั้นทำการเปรียบเทียบค่าของแต่ละจุดภาพหากค่า $f_{old}(i,j)$ น้อยกว่าค่าเธรดโชลด์จุดภาพนั้นจะถูกปรับให้เป็นสีดำหรือส่วนของวัตถุ และหากค่า $f_{old}(i,j)$ มากกว่าหรือเท่ากับค่าเธรดโชลด์จุดภาพนั้นจะถูกปรับให้เป็นสีขาวหรือส่วนของพื้นหลัง หรือสามารถเขียนแทนด้วยสมการที่ 3 ดังต่อไปนี้

$$f_{new}(i,j) = \begin{cases} 1 & if \quad f_{old}(i,j) < T \\ 0 & if \quad f_{old}(i,j) \geq T \end{cases} \tag{3}$$

โดยกำหนดให้

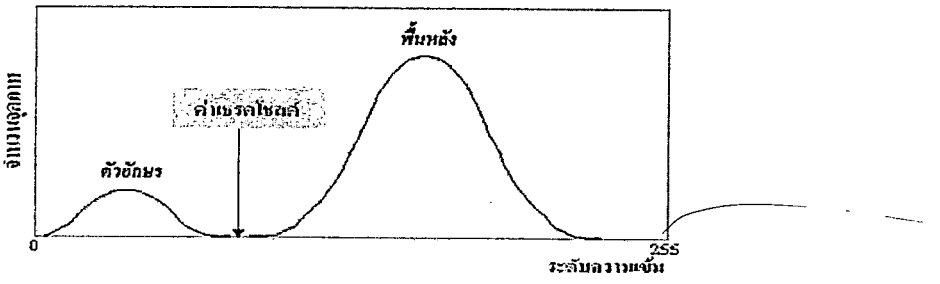
1 คือ สีดำ ซึ่งเป็นส่วนของตัวอักษร

0 คือ สีขาว ซึ่งเป็นส่วนของพื้นหลัง

$f_{new}(i,j)$ คือค่าใหม่ของภาพในลำดับ (i,j)

$f_{old}(i,j)$ คือค่าเก่าของภาพในลำดับ (i,j)

T คือค่าเธรดโชลด์ของภาพ



ภาพที่ 2.1 ฮิสโตแกรมแสดงค่าเธรดโฮลด์

เมื่อได้ภาพสีขาวดำ จะนำภาพเหล่านั้นมาทำการตัดแยกบรรทัดและตัวอักษร ซึ่งเป็นวิธีการหนึ่งในกระบวนการประมวลผลภาพ

2.2 สมการถดถอย (Linear Regression)

สมการถดถอยนั้นเป็นการดูค่าตัวแปร 2 ตัวแปรที่เป็นคู่ๆ กันว่ามีความสัมพันธ์กันในลักษณะใด โดยการสร้างแผนภาพการกระจาย (Scatter Diagram) ระหว่างตัวแปร 2 ตัวนั้น ซึ่งภาพที่ได้อาจจะเป็นเส้นตรง, เส้นโค้ง, ไม่เป็นเส้นตรง ฯลฯ

ในกรณีที่ แผนภาพการกระจายนั้นเป็นเส้นตรง เราสามารถสร้างเป็นความสัมพันธ์ในรูปแบบของสมการที่ 4 ได้ดังนี้

$$Y_i = \alpha + \beta X_i + \varepsilon_i \qquad (4)$$

ซึ่งสมการดังกล่าวเรียกว่าสมการถดถอยอย่างง่าย

โดยกำหนดให้

Y_i คือค่าของตัวแปรตาม

α คือค่าของ Y_i เมื่อ X_i เป็นศูนย์ (Y- intercept)

β คือค่าความชัน (Slope) ของสมการ

ε_i คือค่าความคลาดเคลื่อนระหว่างค่าจริงกับค่าประมาณของ Y_I

วิธีหาค่า α และค่า β จะประมาณโดยวิธีกำลังสองน้อยที่สุด ซึ่งเป็นวิธีที่ได้รับความนิยมกันมาก โดยมีหลักการคือ จะทำให้ผลรวมกำลังสองของความแตกต่างระหว่าง Yi และ Y ที่ได้จากการประมาณมีค่าต่ำที่สุดซึ่งสมการประมาณการจะมีรูปแบบดังสมการที่ 5

$$\hat{Y}_i = a + bX_i \tag{5}$$

โดยกำหนดให้

a คือค่า Y-intercept เป็นค่าประมาณของ α

b คือค่า Slope เป็นค่าประมาณของ β

$\hat{Y}_i$ คือค่าประมาณการของ Yi เมื่อตัวแปรอิสระคือ Xi

ค่า a สามารถหาได้จากสมการที่ 6

$$a = \bar{y} - b\bar{x} \quad \text{หรือ} \quad a = \frac{\sum Y_i}{n} - b\frac{\sum X_i}{n} \tag{6}$$

ค่า b สามารถหาได้จากสมการที่ 7 สมการที่ 8 และสมการที่ 9

$$b = \frac{S_{xy}}{S_{xx}} \tag{7}$$

$$S_{xx} = \sum {X_i}^2 - \frac{(\sum X_i)^2}{n} \tag{8}$$

$$S_{xy} = \sum X_i Y_i - \frac{\sum X_i Y_i}{n} \tag{9}$$

2.3 เวกเตอร์

เวกเตอร์เป็นปริมาณที่กำหนดขึ้นเพื่อใช้อธิบายเกี่ยวกับขนาดและทิศทาง โดยจะต้องบอกทั้งขนาดและทิศทางจึงจะได้ความหมายสมบูรณ์ เช่น ความเร่ง ความเร็ว แรง โมเมนตัม เป็นต้น การหาผลลัพธ์ของปริมาณเวกเตอร์ ต้องอาศัยวิธีการทางเวคเตอร์ โดยต้องหาผลลัพธ์ทั้งขนาดและทิศทาง ขนาดเวกเตอร์แทนด้วยความยาวเส้นตรง ทิศทางเวกเตอร์แทนด้วยหัวลูกศรใช้สัญลักษณ์ $\overrightarrow{AB}$ แทนส่วนของเส้นตรงที่ระบุทิศจาก A ไป B อ่านว่า **เวกเตอร์เอบี** เรียก A ว่าจุดเริ่มต้น (initial point) ของ $\overrightarrow{AB}$ เรียก B ว่าจุดสิ้นสุด (terminal point) ของ $\overrightarrow{AB}$ หัวลูกศรจะอยู่ที่จุดสิ้นสุด ขนาดของเวกเตอร์ $\overrightarrow{AB}$ คือความยาวของส่วนของเส้นตรง AB ใช้สัญลักษณ์ $|\overrightarrow{AB}|$ขนาดนี้จะเป็นปริมาณสเกลาร์ อาจใช้สัญลักษณ์ u แทนเวกเตอร์ได้

การคูณเชิงสเกลาร์

การคูณเชิงสเกลาร์ของเวกเตอร์สองเวกเตอร์ผลลัพธ์ที่ได้เป็นจำนวนจริงไม่ใช่เวกเตอร์ เช่น เวกเตอร์ A เขียนแทนด้วย $\vec{A}$ คูณกับเวกเตอร์ B เขียนแทนด้วย $\vec{B}$ ในปริภูมิสองมิติผลคูณเชิงสเกลาร์ของ $\vec{A}$ และ $\vec{B}$ เขียนแทนด้วย $\vec{A} \cdot \vec{\mathrm{B}}$ และคำนวณได้จากสมการที่ 10 สมการที่ 11 สมการที่ 12 และสมการที่ 13

$$\vec{\mathrm{A}} \cdot \vec{\mathrm{B}} = \|\vec{A}\|\|\vec{B}\| \cos\theta \tag{10}$$

โดยกำหนดให้

$$\vec{\mathrm{A}} \cdot \vec{\mathrm{B}} = (x_a \times x_b) + (y_a \times y_b) \tag{11}$$

$$\|\vec{A}\| = \sqrt{{x_a}^2 + {y_a}^2} \tag{12}$$

$$\|\vec{B}\| = \sqrt{{x_b}^2 + {y_b}^2} \tag{13}$$

ดังนั้นการหามุมระหว่างเวกเตอร์สองเวกเตอร์จะได้ดังสมการที่ 14

$$\theta = \frac{\vec{A} \cdot \vec{B}}{\|\vec{A}\|\|\vec{B}\|} \tag{14}$$

บทที่ 3
โมเดลที่นำเสนอ

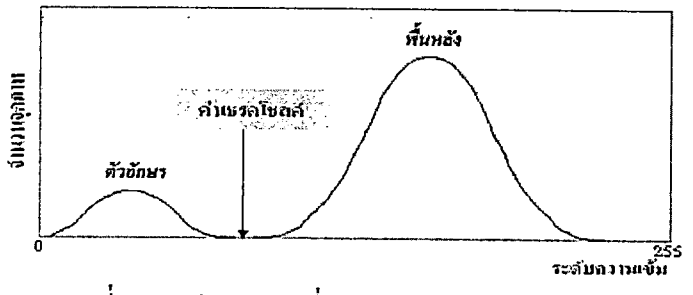
งานวิจัยนี้นำเสนอวิธีการแยกตัวอักษรติดกันโดยการใช้โครงสร้างของตัวอักษรเป็นแนวทางในการแยกตัวอักษร ซึ่งเนื้อหาในบทนี้จะนำเสนอขั้นตอนและอัลกอริทึมที่ใช้ในการแยกตัวอักษรล้านนาที่ติดกันให้เป็นตัวอักษรเดี่ยว

3.1 ขั้นตอนการแบ่งตัวอักษรล้านนาที่ติดกัน

ขั้นตอนการแบ่งตัวอักษรล้านนาที่ติดกันให้เป็นตัวอักษรเดี่ยว แบ่งออกได้เป็น 5 ขั้นตอนได้แก่

3.1.1 การแปลงภาพสีไปเป็นภาพขาวดำ

ทำการแปลงไฟล์ภาพตัวอักษรล้านนาจากภาพสี (RGB image) ไปเป็นภาพขาวดำ (Binary image) ซึ่งในกระบวนการนี้อาศัยวิธีการทำเธรดโชลด์ (Otsu's method) ซึ่งเป็นวิธีที่ใช้ในการแปลงภาพสีให้อยู่ในรูปแบบของภาพขาวดำทำให้สามารถจำแนกตัวอักษรและพื้นหลังออกจากกันได้ ซึ่งหากดูจากภาพที่ 3.1 แสดงฮิสโตรแกรมที่แสดงระดับความเข้มของภาพ $f(i, j)$ นั้นจะเห็นว่าส่วนที่เป็นพื้นหลังจะมีเข้มมากกว่าส่วนที่เป็นตัวอักษร ดังนั้นจะทำให้สามารถแยกกลุ่มของตัวอักษรและพื้นหลังออกจากกันได้อย่างชัดเจนและเลือกค่าเธรดโชลด์ที่มีระดับความเข้มอยู่ระหว่างกลุ่มทั้งสองของฮิสโตรแกรมได้



ภาพที่ 3.1 ฮิสโตรแกรมที่แสดงระดับความเข้มของภาพ

จากนั้นทำการเปรียบเทียบค่าของแต่ละจุดภาพ หากค่า $f(i, j)$ น้อยกว่าค่าเธรดโชลด์จุดภาพนั้นจะถูกปรับให้เป็นสีดำ ซึ่งจะมีค่าเท่ากับ 1 และหากค่า $f(i, j)$ มากกว่าหรือเท่ากับค่าเธรดโชลด์

จุดภาพนั้นจะถูกปรับให้เป็นสีขาวซึ่งจะมีค่าเท่ากับ 0 หรือสามารถเขียนแทนด้วยสมการที่ 15 ดังต่อไปนี้

$$f_{new}(i,j) = \begin{cases} 1 & if \quad f_{olf}(i,j) < T \\ 0 & if \quad f_{old}(i,j) \geq T \end{cases} \tag{15}$$

โดยกำหนดให้

1 คือ สีดำ ซึ่งเป็นส่วนของตัวอักษร

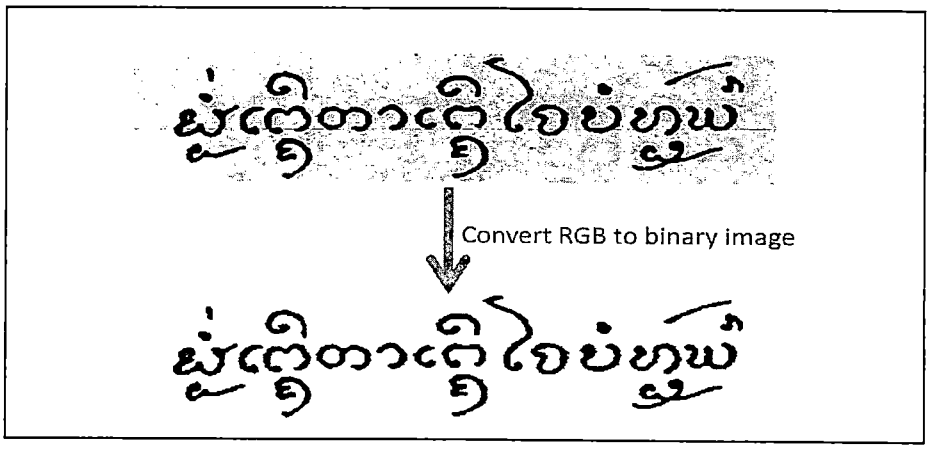
0 คือ สีขาว ซึ่งเป็นส่วนของพื้นหลัง

$f_{new}(i,j)$ คือค่าใหม่ของภาพในลำดับ (i,j)

$f_{old}(i,j)$ คือค่าเก่าของภาพในลำดับ (i,j)

T คือค่าเก่าของภาพในลำดับ (i,j)

ผลลัพธ์ที่ได้จากการทำในกระบวนนี้แสดงได้ดังภาพที่ 3.2



ภาพที่ 3.2 การแปลงภาพสี (RGB Image) ไปเป็นภาพไบนารี (Binary Image)

3.1.2 การตัดตัวอักษรโดยการใช้ Bounding box analysis

วิธีการระบุกรอบของตัวอักษรโดยอัลกอริทึมที่ใช้ได้แก่ Bounding box analysis เพื่อทำการตัดตัวอักษรให้เป็นตัวอักษรเดี่ยว ขั้นตอนของการตัดตัวอักษรโดยการใช้ Bounding box analysis แสดงได้ดังนี้

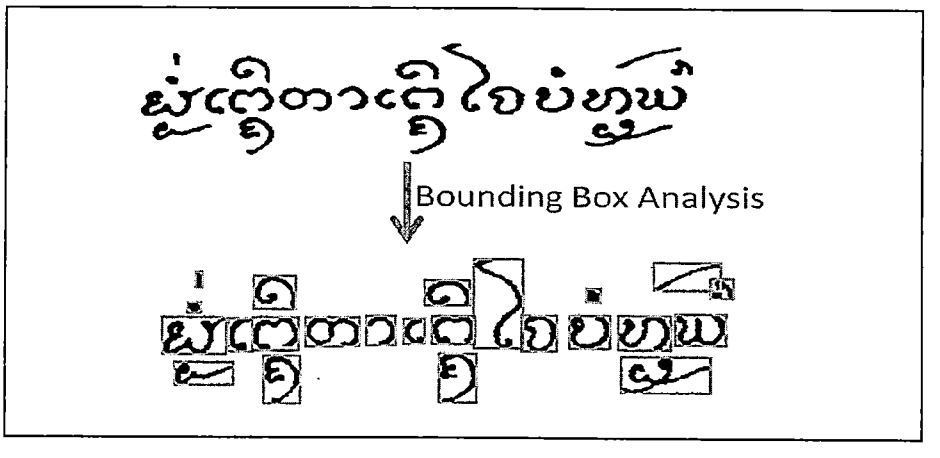
3.1.2.1 ทำการค้นหาจุดสีดำของตัวอักษรโดยเริ่มคอลัมน์ที่ 1 จากด้านซ้ายของภาพและจากด้านแถวบนลงไปแถวล่าง

3.1.2.2 เมื่อพบจุดแรกของตัวอักษรก็ทำการไล่ตามเส้นขอบตัวอักษรที่พบและจะหยุดเมื่อกลับมายังจุดเริ่มต้นโดยในขั้นตอนนี้จะทำให้ได้จุด x และจุด y ของขอบภาพและทำการค้นหาจุดที่มีค่า x มากที่สุดและจุดที่มีค่า x น้อยที่สุด จุดที่มีค่า y มากที่สุดและจุดที่มีค่า y น้อยที่สุด

3.1.2.3 ทำการสร้างบล็อคตัดตัวอักษรจากพิกัดจุด x และจุด y ที่ได้จากข้อก่อนหน้า

3.1.2.4 ทำจนครบทุกภาพตัวอักษร

ผลลัพธ์ที่ได้จากการตัดตัวอักษรโดยการใช้ Bounding box analysis แสดงได้ดังภาพที่ 3.3



ภาพที่ 3.3 ผลลัพธ์ของการตัดคำโดยการใช้ Bounding Box Analysis

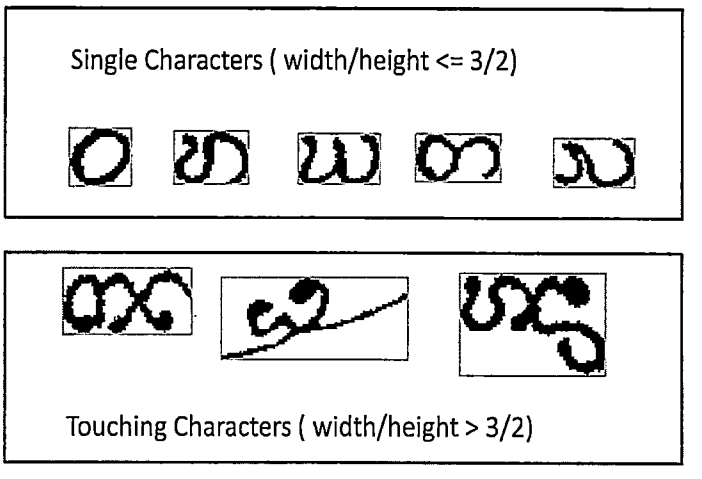
3.1.3 ขั้นตอนการหาตัวอักษรติดกัน

หลังจากที่ผ่านขั้นตอนที่สองมาแล้วก็จะทำการหาภาพตัวอักษรที่มีลักษณะติดกันโดยมีการกำหนดค่าที่จะใช้ในการหาดังนี้

3.1.3.1 ขนาดของภาพตัวอักษรที่มีขนาดใหญ่มากกว่าภาพตัวอักษรทั่วๆไปจะถือว่าเป็นตัวอักษรติดกัน

3.1.3.2 ขนาดของตัวอักษรเดี่ยวของตัวอักษรล้านนาจะมีขนาดน้อยกว่าหรือเท่ากับ 3/2 ถ้าภาพตัวอักษรที่ได้มีอัตราส่วนระหว่างความกว้างและความสูงมากกว่า 3/2 จะถือว่าเป็นตัวอักษรติดกัน

ตัวอย่างของการหาตัวอักษรที่ตัดกันโดยการใช้ขั้นตอนที่ 1.3.1 และขั้นตอนที่ 1.3.2 แสดงได้ดังภาพที่ 3.4



ภาพที่ 3.4 ขั้นตอนการหาตัวอักษรติดกัน

3.1.4 การสร้างโครงกระดูกตัวอักษร

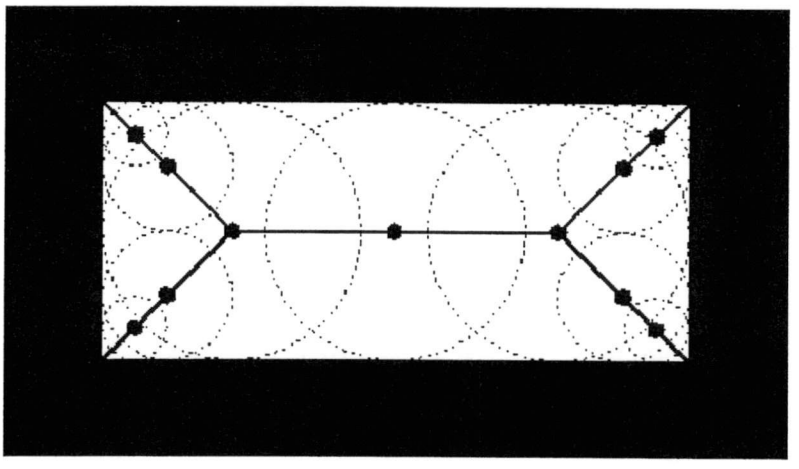
ทำการสร้างแกนหลักของโครงกระดูกตัวอักษรที่ติดกันที่ได้มาจากขั้นตอนที่สาม โดยมีขั้นตอนการสร้างโครงกระดูตัวอักษรดังนี้

3.1.4.1 ทำการสร้างวงกลมให้อยู่ในพื้นที่ของตัวอักษรโดยมีเงื่อนไขว่าต้องมีด้านของวงกลมอย่างน้อยสองด้านขึ้นไปที่ติดกับเส้นขอบของตัวอักษรและทำการสร้างจากด้านซ้ายไปด้านขวา

3.1.4.2 ทำการหาจุด x และ y ของศูนย์กลางวงกลม

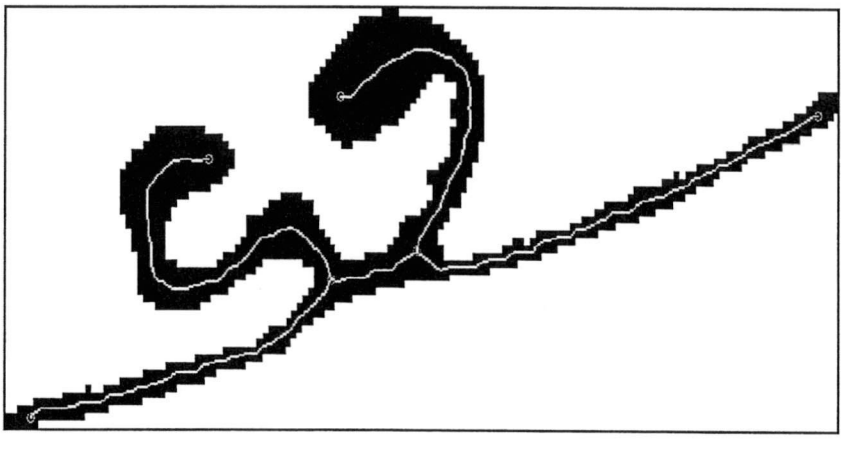
3.1.4.3 สร้างโครงกระดูกของตัวอักษรโดยการนำเอาจุด x และ y ของศูนย์กลางวงกลมมาเชื่อมต่อกัน

วิธีการหาโครงกระดูกตัวอักษรในขั้นตอนดังกล่าวแสดงได้ดังภาพที่ 3.5



ภาพที่ 3.5 ผลลัพธ์ของการสร้างโครงกระดูกของตัวอักษร

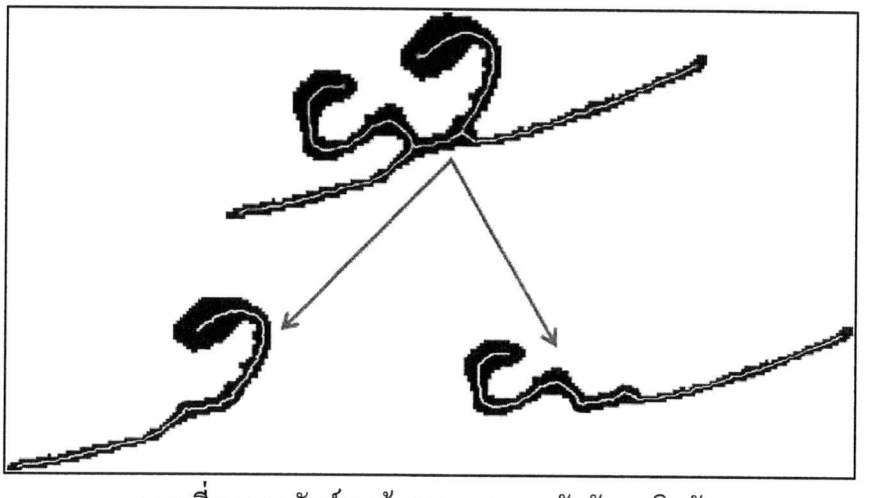
ผลลัพธ์ของการสร้างโครงกระดูกตัวอักษรแสดงได้ดังภาพที่ 3.6



ภาพที่ 3.6 ผลลัพธ์ของการสร้างโครงกระดูกของตัวอักษร

3.1.5 ผลลัพธ์จากการแยกตัวอักษรติดกัน

ส่งเข้าอัลกอริทึมที่พัฒนาขึ้นเพื่อทำการแยกตัวอักษรติดกันให้เป็นตัวอักษรเดี่ยว แสดงได้ดังภาพที่ 3.7



ภาพที่ 3.7 ผลลัพธ์สุดท้ายของการแยกตัวอักษรติดกัน

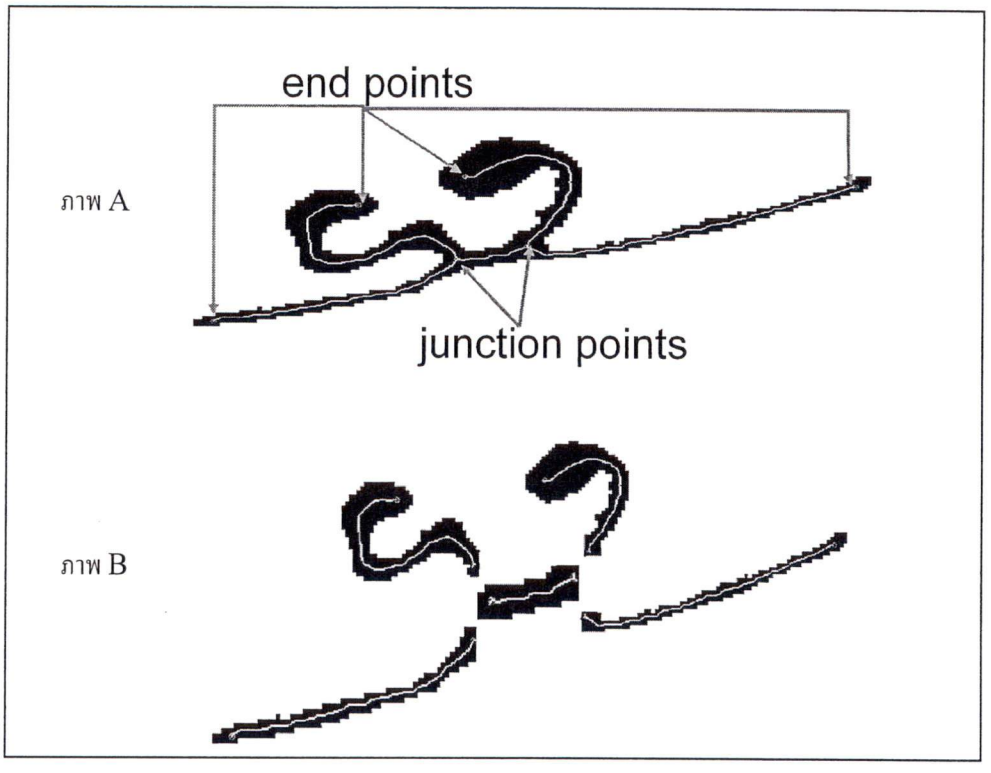
3.2 อัลกอริทึม

อัลกอริทึมที่ใช้ในการแยกตัวอักษรล้านนาที่ติดกันให้เป็นตัวอักษรเดี่ยว ในงานวิจัยครั้งนี้แสดงได้ดังต่อไปนี้

3.2.1 การสร้างโครงกระดูกและหาจุดสิ้นสุดและจุดแยกตัวอักษร

นำภาพตัวอักษรล้านนาที่ติดกันมาทำการสร้างโครงกระดูกของภาพตัวอักษร จากนั้นทำการหาจุดปลายของตัวอักษรและจุดทางแยกของตัวอักษรที่ติดกัน จากนั้นใช้จุดทางแยกเป็นแกนหลักในการแบ่งภาพตัวอักษรให้เป็นภาพย่อยๆ แสดงได้ดังภาพที่ 3.8

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



ภาพที่ 3.8 ภาพ A จุดปลายและจุดทางแยกของตัวอักษรติดกัน
ภาพ B การแบ่งภาพตัวอักษรติดกันให้เป็นภาพย่อย

3.2.2 การแบ่งภาพหลักให้เป็นภาพย่อย

เมื่อได้รูปภาพชิ้นย่อยๆ จากกระบวนการในขั้นตอนที่ 3.2.1 ในขั้นตอนนี้จะทำการหาว่าชิ้นส่วนของภาพนั้นเป็น สมาชิกของจุดทางแยกด้านไหนโดยการหาค่าความเป็นสมาชิกจากสมการที่ 16

$$T\begin{bmatrix} C_{ix} \\ C_{iy} \end{bmatrix} = \begin{bmatrix} C_{ix} \\ C_{iy} \end{bmatrix} - \begin{bmatrix} J_{ix} \\ J_{iy} \end{bmatrix} \qquad (16)$$

โดยกำหนดให้

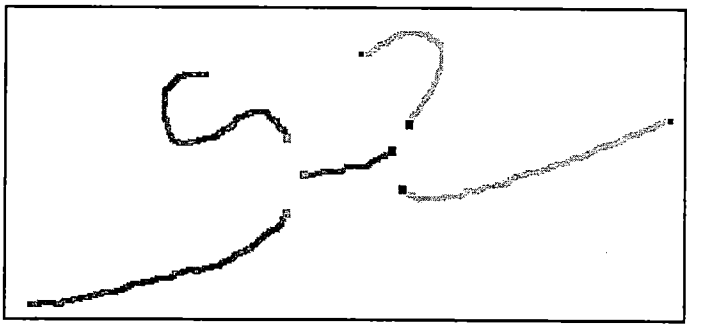
T คือ Translation Operator

C_{ix} และ C_{iy} คือจุดตำแหน่งของเส้นโครงกระดูกตัวอักษรลำดับ (x,y)

J_{ix} และ J_{iy} คือจุดตำแหน่งทางแยกของเส้นโครงกระดูกตัวอักษรลำดับ (x,y)

131176

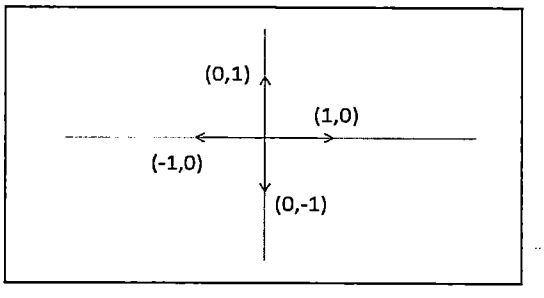
ผลลัพธ์ที่ได้จากการแบ่งภาพตัวอักษรออกเป็นคอนทัวร์ย่อยแสดงได้ดังภาพที่ 3.9



ภาพที่ 3.9 ผลลัพธ์ของคอนทัวร์ภาพย่อย

3.2.3 การสร้างเวกเตอร์อ้างอิงหนึ่งหน่วย

ทำการสร้างเวกเตอร์อ้างอิงหนึ่งหน่วยยังจุดเริ่มต้น (0,0) ไปในทิศทางของแกน x และแกน y แสดงได้ดังภาพที่ 3.10



ภาพที่ 3.10 ภาพเวกเตอร์อ้างอิงหนึ่งหน่วยในแนวแกน x และแนวแกน y

3.2.4 การหาขนาดของรูปภาพย่อย

ทำการหาขนาดของแต่ละคอนทัวร์ย่อยเพื่อที่จะนำไปสร้างเป็นเวกเตอร์สำหรับกระบวนการแยกตัวอักษรติดกันโดยการหามุมของคู่ลำดับ (x,y) ในคอนทัวร์ย่อยนั้นกับเวกเตอร์อ้างอิงหนึ่งหน่วย ทำโดยการหามุมตามเข็มนาฬิกาจากเวกเตอร์ของจุดพิกเซลบนคอนทัวร์กับเวกเตอร์อ้างอิงหนึ่งหน่วยในแนวแกน x หรือ y ที่ใกล้ที่สุดในทิศทางตามเข็มนาฬิกาและจะทำการหยุดก็ต่อเมื่อมุมที่ได้มีการซ้ำกัน 3 ครั้ง การหามุมของเวกเตอร์ดังกล่าวหาได้จากสมการที่ 17

$$\theta_n = \cos^{-1} \frac{U \cdot V_n}{\|U\|\|V_n\|} \tag{17}$$

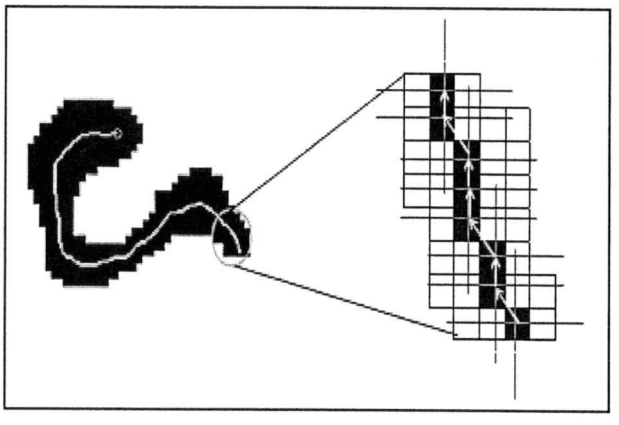
โดยกำหนดให้

θ_n คือมุมที่หาได้จากเวกเตอร์ของจุดพิกเซลบนคอนทัวร์กับเวกเตอร์อ้างอิง

U คือเวกเตอร์อ้างอิงหนึ่งหน่วย

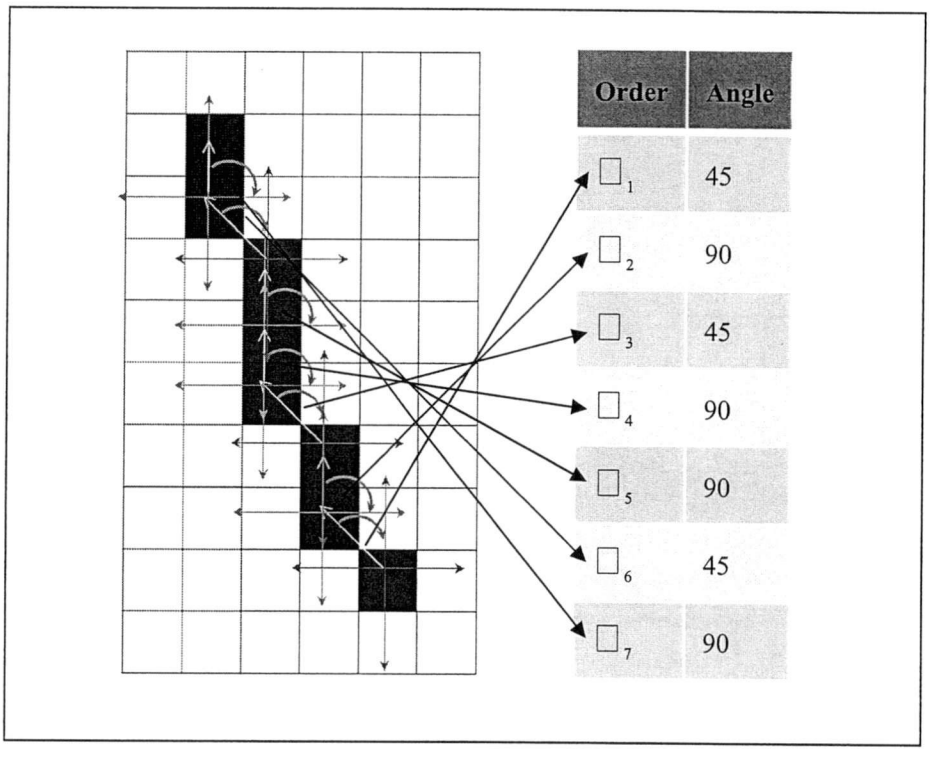
V_n คือเวกเตอร์ของจุดพิกเซลบนคอนทัวร์

จากขั้นตอนดังกล่าวจะแสดงวิธีการการคอนทัวร์ได้ดังภาพที่ 3.11



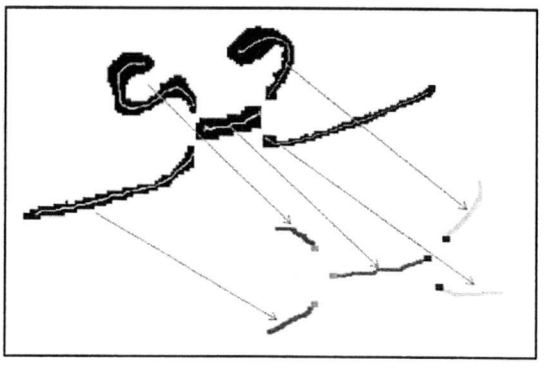
ภาพที่ 3.11 ขั้นตอนการคอนทัวร์ของกิ่งย่อย

และการหามุมที่หาได้จากเวกเตอร์ของจุดพิกเซลในภาพกับเวกเตอร์อ้างอิงแสดงได้ดังภาพที่ 3.12



ภาพที่ 3.12 ขั้นตอนการหามุมในการคอนทัวร์แต่ละพิกเซล

เมื่อทำการคอนทัวร์ครบทุกรูปภาพย่อยตัวอักษรแล้วก็จะได้ขนาดของกิ่งที่จะนำไปสร้างเป็นเวกเตอร์แสดงได้ดังภาพที่ 3.13



ภาพที่ 3.13 ผลลัพธ์ของเวกเตอร์ของคอนทัวร์ในแต่ละกิ่งย่อย

3.2.5 การสร้างเส้นตรงจากภาพย่อย

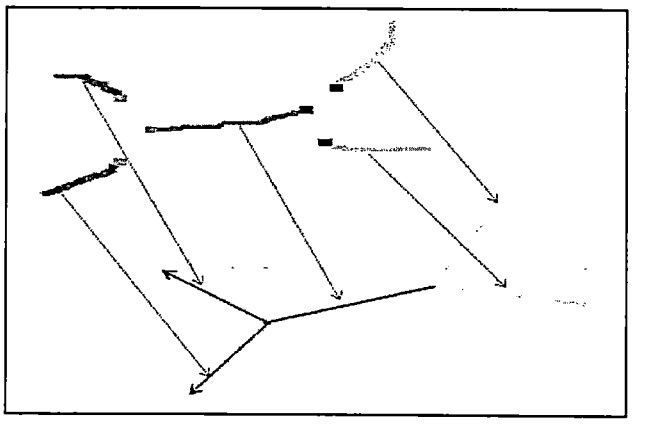
นำกิ่งย่อยที่ได้จากขั้นตอนที่ 3.2.4 มาทำการปรับให้เป็นเส้นตรงโดยการใช้สมการถดถอย (linear regression) ตามสมการที่ 18, 19 และ 20

$$y = a + bx \tag{18}$$

$$b = \frac{\sum(x - \bar{x})(y - \bar{y})}{\sum(x - \bar{x})^2} \tag{19}$$

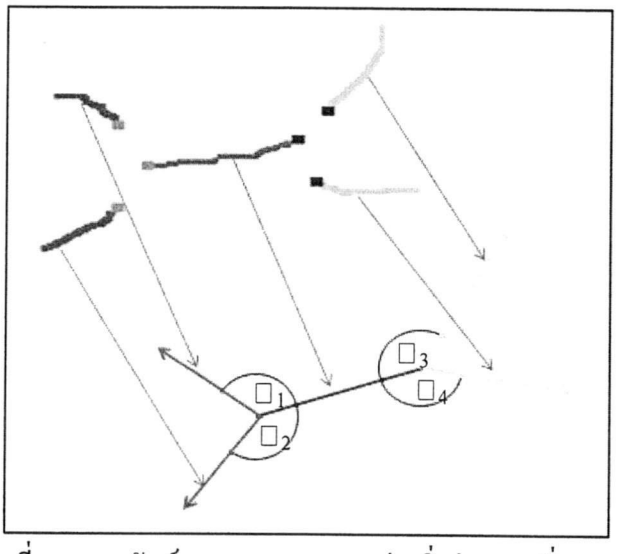
$$a = y - bx \tag{20}$$

หลังจากที่นำกิ่งย่อยของภาพมาทำการปรับเป็นเส้นตรงโดยการใช้สมการถดถอยแล้วผลลัพธ์ที่ได้แสดงได้ดังภาพที่ 3.14



ภาพที่ 3.14 ผลลัพธ์ของกิ่งย่อยหลังจากผ่านสมการถดถอย

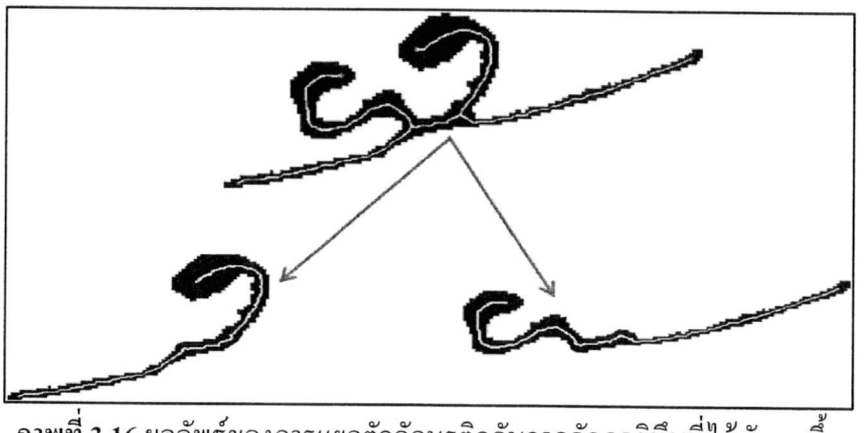
เมื่อทำการแปลงกิ่งย่อยให้เป็นเส้นตรงแล้วก็ทำการหามุมระหว่างกิ่งย่อยกับกิ่งแกนกลางของตัวอักษรที่ติดกันแสดงได้ดังภาพที่ 3.15



ภาพที่ 3.15 ผลลัพธ์ของการหามุมระหว่างกิ่งย่อยและกิ่งจุดแยก

3.2.6 ผลลัพธ์ของการแยกตัวอักษรที่ได้จากอัลกอริทึม

จากการศึกษาลักษณะการเขียนตัวอักษรล้านนาจะพบว่าจะมีลักษณะเป็นเส้นโค้งที่ไม่มีการเปลี่ยนแปลงทิศทางอย่างทันทีทันใด ดังนั้นแนวคิดที่ใช้ในการแยกตัวอักษรติดกันของตัวอักษรล้านนาจึงได้มีการเปรียบเทียบมุมที่ได้จากจุดแยกด้านหนึ่งกับมุมที่ได้จากจุดแยกด้านตรงข้าม ถ้ามุมระหว่างจุดแยกใดมีค่าใกล้กันมากที่สุด จะทำการเชื่อมต่อเส้นที่เกี่ยวข้องกับคู่ของมุมนั้นๆ เข้าด้วยกัน ผลลัพธ์ของการแยกตัวอักษรติดกันแสดงได้ดังภาพที่ 3.16



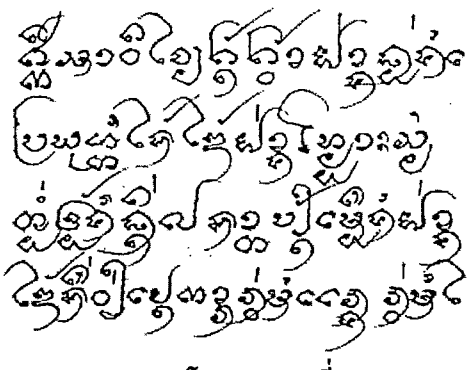
ภาพที่ 3.16 ผลลัพธ์ของการแยกตัวอักษรติดกันจากอัลกอริทึมที่ได้พัฒนาขึ้น

บทที่ 4

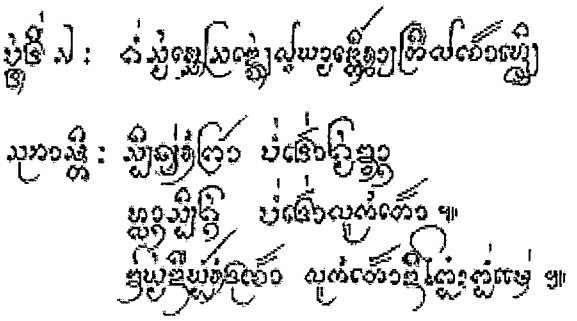
การทดลองและผลการทดลอง

ในการศึกษาวิจัยครั้งนี้ได้มีการนำเอาเอกสารล้านนาโบราณที่มีตัวอักษรต่างกันอยู่จำนวน 6 รูปแบบ แต่ละรูปแบบมีการนำเอกสารมาใช้สี่หน้า รวมเป็นเอกสารทั้งหมด 24 หน้า ซึ่งตัวอย่างเอกสารล้านนาโบราณทั้ง 6 รูปแบบแสดงได้ดังภาพที่ 4.1

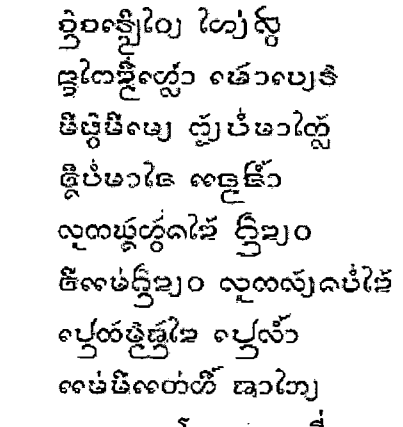
จากการศึกษาและประยุกต์ใช้อัลกอริทึมที่ได้พัฒนาขึ้นมากับตัวอักษรที่ติดกันจำนวน 85 ตัวอักษรเพื่อทำการทดสอบแยกให้เป็นตัวอักษรเดี่ยว ผลการทดลองปรากฏว่ามีความถูกต้องในการแยกตัวอักษรติดกันให้เป็นอักษรเดี่ยว 75.3 เปอร์เซ็นต์ ซึ่งตัวอย่างของผลลัพธ์ของการแยกตัวอักษรติดกันแสดงได้ดังตารางที่ 4.1 และตารางที่ 4.2



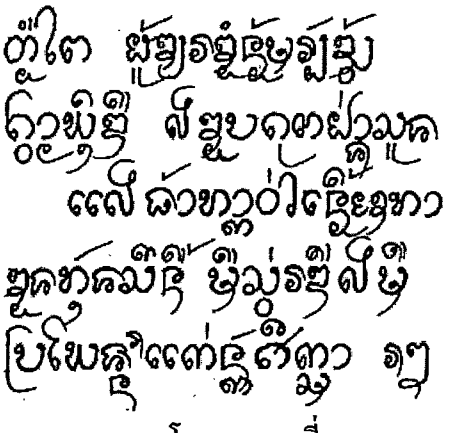
เอกสารโบราณแบบที่ 1



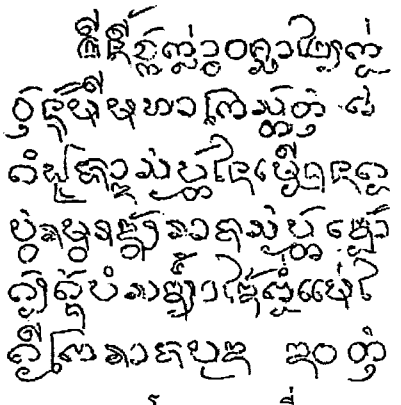
เอกสารโบราณแบบที่ 2



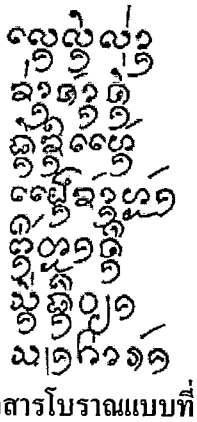
เอกสารโบราณแบบที่ 3



เอกสารโบราณแบบที่ 4

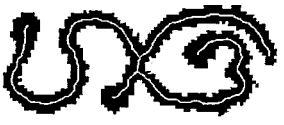
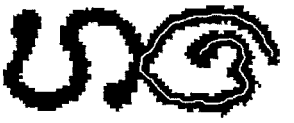


เอกสารโบราณแบบที่ 5



เอกสารโบราณแบบที่ 6

ภาพที่ 4.1 ภาพตัวอักษรล้านนา 6 รูปแบบที่ใช้ในการศึกษาวิจัย

ตัวอักษรที่ติดกัน	ตัวอักษรที่แยกได้ถูกต้อง	
		

ตารางที่ 4.1 ภาพตัวอย่างผลลัพธ์ของการแยกตัวอักษติดกันที่ถูกต้อง

ตัวอักษรที่ติดกัน	ตัวอักษรที่แยกได้ไม่ถูกต้อง		เป้าหมายที่ต้องการแยก	

ตารางที่ 4.2 ภาพตัวอย่างผลลัพธ์ของการแยกตัวอักษติดกันที่ไม่ถูกต้องและเป้าหมายที่ต้องแยก

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

งานวิจัยนี้นำเสนอวิธีการแยกตัวอักษรล้านนาติดที่กันโดยการใช้รูปแบบโครงสร้างของตัวอักษรเป็นแนวทางในการช่วยตัดสินใจ โดยในขั้นแรกจะทำการแปลงภาพเอกสารอักษรล้านนาจากภาพสีให้อยู่ในภาพขาวดำ จากนั้นใช้ Bounding Box Analysis มาเป็นตัวช่วยในการตัดแยกตัวอักษรล้านนาจากเอกสารออกเป็นภาพของตัวอักษรเดี่ยวและตัวอักษรติดกัน จากนั้นใช้อัตราส่วนของตัวอักษรเดี่ยวมาเป็นเกณฑ์ในการแยกเอาตัวอักษรที่ติดกันออกมา จากนั้นนำเอาตัวอักษรที่ติดกันมาทำการหาโครงกระดูกของตัวอักษรและทำการหาจุดปลายและจุดทางแยกของตัวอักษรและทำการแบ่งตัวอักษรที่ติดกันให้อยู่ในภาพย่อย เมื่อได้ภาพย่อยของตัวอักษรแล้วก็ทำการหาขนาดของกิ่งภาพย่อยที่จะนำไปสร้างเวกเตอร์โดยการสร้างเวกเตอร์หนึ่งหน่วยมายังเวกเตอร์อ้างอิง จากนั้นทำการหามุมของเส้นโครงกระดูกโดยการสร้างเวกเตอร์จากพิกเซลที่ติดกับจุดแยกย้อนขึ้นไปเรื่อยๆ ตามลำดับ และจะทำการหยุดก็ต่อเมื่อมีมุมซ้ำกันสามครั้ง เมื่อได้ขนาดของกิ่งครบทุกภาพย่อยแล้วก็นำไปเข้าสู่สมการถดถอยเพื่อทำการปรับให้กิ่งของภาพย่อยๆ ให้อยู่ในรูปของเส้นตรงจากนั้นก็ทำการหามุมระหว่างเวกเตอร์ของกิ่งกับเวกเตอร์ของจุดแยกและนำมุมที่ได้มาเปรียบเทียบกันซึ่งตัวอักษรล้านนาที่เป็นตัวเดี่ยวกันจะมีขนาดของมุมเท่ากับหรือใกล้เคียงกันจากนั้นก็นำเอาภาพย่อยที่ถูกแบ่งเข้ามาประกอบเป็นตัวอักษรซึ่งมีความถูกต้อง 75.3 เปอร์เซ็นต์

5.2 ข้อเสนอแนะ

ในการวิจัยครั้งนี้จะเห็นว่าค่าความถูกต้องยังไม่สูงมากนักจากการสังเกตุและวิเคราะห์จะเห็นว่าขั้นตอนในการทำการแยกภาพพื้นหลังออกจากตัวอักษรนั้นยังมีคุณภาพที่ไม่ดีหรืออาจจะเป็นที่คุณภาพของเอกสารล้านนาที่ใช้ในงานวิจัยมีคุณภาพที่ไม่ดี จึงทำให้เกิดข้อผิดพลาดในการสร้างโครงร่างกระดูกของตัวอักษรซึ่งเมื่อทำการหามุมก็จะทำให้มุมมีความผิดพลาดได้

บรรณานุกรม

Bhowmik, T. K., Roy, A., Roy, U., 2005. Character Segmentation for Handwritten Bangla Words Using Artificial Neural Network. *In Proceedings of the International Workshop on Neural Networks and Learning in Document Analysis and Recognition.*

Casey, R. G., Lecolinet, E., 1996. A Survey of Methods and Strategies in Character Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 7, pp. 690-706.

Chen, J. L., Wu, C. H., Lee, H. J., 1998. Chinese Handwritten Character Segmentation in Form Documents. *Document Analysis Systems: Theory and Practice*, LNCS 1655, pp. 348-362.

Yang, F., Si, J. H., Liu, Y. H., Tian, X. D., 2009. Fuzzy Threshold Method for Off-line Chinese Handwritten Character Segmentation. *In Proceedings of the 8th International Conference on Machine Learning and Cybernetics.*

Hoang, T. V., Tabbone, S., Pham, N., 2009. Recognition-based Segmentation of Nom Characters from Body Text Regions of Stele Images Using Area Voronoi Diagram. *In Proceedings of the 13th International Conference on Computer Analysis of Images and Patterns.*

Alamri, H., He, C. L., Suen, C. Y., 2009. A New Approach for Segmentation and Recognition of Arabic Handwritten Touching Numeral Pairs. *In Proceedings of the 13th International Conference on Computer Analysis of Images and Patterns.*

Marinai, S., Gori, M., Soda, G., 2005. Artificial Neural Networks for Document Analysis and Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 1, pp. 23-35.

Rui, M., Yingnan, Z., Yongquan, X., Yunyang, Y., 2008. A Touching Pattern-Oriented Strategy for Handwritten Digits Segmentation. *In Proceedings of the 2008 International Conference on Computational Intelligence and Security.*

Soba, T., Sulong, G., Rehman, A., 2010. A Survey on Methods and Strategies on Touched Characters Segmentation. *International Journal of Research and Reviews in Computer Science*, vol. 1, no. 2, pp. 103-114.

Tseng, L. Y., Chen, R. C., 1998. Segmenting Handwritten Chinese Characters Based on Heuristic Merging of Stroke Bounding Boxes and Dynamic Programming. *Pattern Recognition Letter*, vol. 19, pp. 963-973.

Xiao, X., Leedham, G., 2000. Knowledge-based English Cursive Script Segmentation. *Pattern Recognition Letters*, vol. 21, pp. 945-954.