

## การปรับปรุงโครงสร้างรูปภาพตัวอักษรเพื่อการรู้จำรูปแบบ

### Modification of image character structures for pattern recognition

เศรษฐไกร พ่อคำ<sup>1</sup>, ศิวกรณ์ อาจรัชชา<sup>2</sup> และ วัชรพงศ์ ทองตัน<sup>3</sup>

สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาการสารสนเทศ มหาวิทยาลัยบูรพา ชลบุรี

Emails: 56660201@go.buu.ac.th<sup>1</sup>, 56660009@go.buu.ac.th<sup>2</sup>, 56660147@go.buu.ac.th<sup>3</sup>

#### บทคัดย่อ

งานวิจัยนี้เป็นการนำเสนอวิธีเพื่อปรับปรุงโครงสร้างรูปภาพตัวอักษรและนำลักษณะโครงสร้างที่ถูกปรับปรุงใหม่เหล่านี้ไปใช้เป็นข้อมูลนำเข้าในฟังก์ชัน OCR (Optical Character Recognition) เนื่องจากวิธีการรู้จำรูปแบบตัวอักษรของฟังก์ชัน OCR ยังมีข้อจำกัดในด้านมาตรฐานของรูปแบบตัวอักษร หากข้อมูลนำเข้ามีรูปแบบของตัวอักษรที่ไม่เป็นมาตรฐานจะทำให้การรู้จำในฟังก์ชันมีประสิทธิภาพน้อยลง ซึ่งขั้นตอนการทำงานในงานวิจัยนี้จะประกอบด้วย 3 ขั้นตอนหลัก คือ ขั้นตอนที่ 1 กระบวนการในการคัดแยกรูปแบบของตัวอักษรเพื่อเลือกใช้วิธีเฉพาะของตัวอักษร ขั้นตอนที่ 2 กระบวนการก่อนการประมวลผลภาพ (Pre-processing) จากการปรับปรุงโครงสร้างตัวอักษรโดยใช้หลักการคาดเดาก่อนเข้าฟังก์ชัน OCR และขั้นตอนสุดท้ายเรียกใช้ฟังก์ชัน OCR เพื่อรู้จำรูปแบบของตัวอักษรที่ไม่เป็นมาตรฐาน จากผลการทดลองงานวิจัยนี้จะทำการเปรียบเทียบความถูกต้องของวิธีการในฟังก์ชัน OCR ปกติกับการเปรียบเทียบการรู้จำรูปแบบมาเทียบกัน (Template Matching) และวิธีการที่แนะนำให้เสนอพบว่ามีความถูกต้องเพิ่มขึ้นจากการเปรียบเทียบโดยวัดความถูกต้องจากกลุ่มคำถึง 70% จากข้อมูลนำเข้าในรูปแบบกลุ่มคำ

**คำสำคัญ**– โครงสร้างตัวอักษร ; กระบวนการก่อนการประมวลผลภาพ ; การรู้จำตัวอักษร

#### Abstract

This research is to propose how to modify characteristic image structure for pattern recognition of text image for input of OCR (Optical Character Recognition). Because of OCR function's font recognition work only in standard font pattern. If the

input data is the form of character that is not standard pattern the 3 processing steps is used. First, classify the type of character for each algorithm. Second, image enhancement for OCR function. Third, OCR function is used to recognize. According to the result, the accuracy is increased to 70% when compares with normal OCR and Template Matching.

**Keywords**- Character structures; Pre-processing; Optical Character Recognition

#### 1. บทนำ

การรู้จำ (Recognition) [1] มีความสำคัญและเป็นศาสตร์แขนงหนึ่งของระบบปัญญาประดิษฐ์ (Artificial Intelligence) [2] การรู้จำตัวอักษรมีขั้นตอนการทำงานที่สำคัญคือการวิเคราะห์ภาพที่ประกอบไปด้วยตัวอักษร แล้วทำการดึงตัวอักษรออกมาทีละตัวตามลำดับ หากการแยกตัวอักษรไม่มีประสิทธิภาพและไม่ถูกต้องจะทำให้ผลลัพธ์ที่ได้จากขั้นตอนผิดพลาด การใช้งานระบบรู้จำตัวอักษร (OCR :Optical Character Recognition) นั้นจะช่วยลดระยะเวลาในการนำข้อมูลประเภทรูปภาพของตัวอักษรไปจัดเก็บโดยวิธีการพิมพ์หรือการนำไปสแกนภาพซึ่งข้อมูลที่ได้อาจไม่สามารถแก้ไขข้อมูลเพื่อจัดเก็บได้ เช่นการแก้คำผิดและการเพิ่มข้อความลงไป เป็นต้น ทำให้เกิดความไม่สะดวกในการใช้งานข้อมูลที่ต้องการ อย่างไรก็ตามการประมวลผลของ OCR ยังมีรูปแบบตัวอักษรที่เป็นมาตรฐานทำให้ความหลากหลายของตัวอักษรที่นำเข้า OCR นั้นมีผลกระทบต่อความถูกต้องของตัวอักษร

เนื่องจากการพัฒนาตัวอักษรให้มีความสวยงามมากขึ้นจากอักษรมาตรฐานทั่วไปและถูกนำมาใช้งานในการเขียนลงบนสิ่งพิมพ์มากขึ้นทำให้การจัดเก็บข้อมูลจากสิ่งพิมพ์ลงคอมพิวเตอร์ด้วย OCR นั้นมีความถูกต้องน้อยลงด้วยเหตุนี้จึง

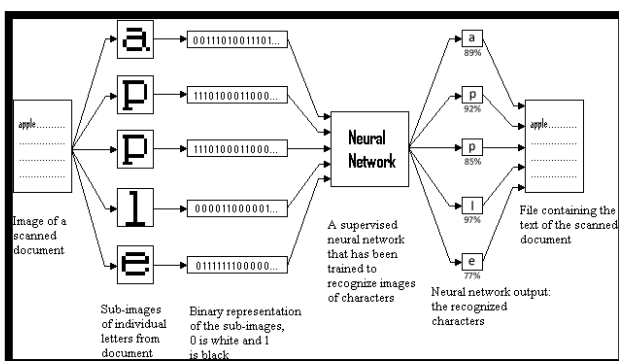
นำเสนอการทำ Pre-processing ของภาพให้เข้าเงื่อนไขของรูปแบบ OCR และให้มีลักษณะใกล้เคียงกับตัวอักษรมาตรฐาน โดยทั่วไปแล้วอักษรนั้นเกิดจากการขีดเขียนโครงสร้างที่มีความใกล้เคียงกับรูปทรงซึ่งจะมีลักษณะเฉพาะของแต่ละตัวอักษรได้จากเรขาคณิตในบางตัวอักษรทำให้เกิดรูปแบบของตัวอักษรโดยขอบเขตตัวอักษรที่ใช้เป็นประเภทอักษรศิลป์ของยี่ห้อสินค้ามาทำการทดสอบ

## 2. ทฤษฎีที่เกี่ยวข้อง

จากการศึกษาปัญหาของตัวอักษรศิลป์จากข้อมูลนำเข้านั้นพบว่ารูปแบบตัวอักษรของศิลป์มีรูปแบบคล้ายคลึงกับอักษรมาตรฐานเพียงเล็กน้อยแต่ยังคงลักษณะเฉพาะของตัวอักษรไว้ทำจึงได้ใช้ทฤษฎีเบื้องต้นของการ Recognition ของ OCR ใน MATLAB<sup>®</sup> ทฤษฎีการทำงานก่อนการประมวลผล (Pre-processing) ทฤษฎี จับคู่แม่แบบ (Template Matching)

### 2.1 OCR (Optical Character Recognition)

โปรแกรมการรู้จำอักษรด้วยแสง ซึ่งเป็นโปรแกรมที่สามารถอ่านตัวอักษรของรูปภาพที่อยู่บนคอมพิวเตอร์หรือไฟล์ภาพที่ถูกสแกนเข้ามาเพื่อนำมาแปลงเป็นภาษาเครื่องของคอมพิวเตอร์ทำให้สามารถแก้ไขข้อมูลของตัวอักษรได้หรือจัดเก็บลงคอมพิวเตอร์ได้โดยที่ OCR จะมีขั้นตอนกระบวนการสกัดคุณลักษณะของตัวอักษร (Feature Extraction) การแยกวัตถุที่สนใจ (Segmentation) การจำแนกหมวดหมู่ (Classification) โดยที่โปรแกรม OCR จะนำข้อมูลนำเข้าที่เป็นรูปภาพตัวอักษร โปรแกรม OCR จะทำการตัดแบ่งคำให้มีอักษรเพียงหนึ่งตัวและแปลงภาพให้อยู่ในข้อมูลของเมทริกซ์ 0 และ 1 แล้วนำข้อมูลไปจำแนกหมวดหมู่ (Classification) ซึ่งจะมีกระบวนการทำงานตามรูปภาพที่ 1



รูปที่ 1. ภาพกระบวนการของโปรแกรม OCR จาก

<http://how-ocr-works.com/history/neural-networks.html>

### 2.2 การทำงานก่อนการประมวลผล (Pre-processing)

ก่อนการประมวลผล จากข้อมูลรูปภาพที่เข้ามามีลักษณะรูปแบบตัวอักษรที่ไม่ได้มาตรฐาน จึงทำการปรับปรุงโครงสร้างของรูปแบบตัวอักษรเพื่อให้ใกล้เคียงกับตัวอักษรแบบมาตรฐาน โดยใช้หลักการของเรขาคณิตมาช่วยปรับปรุงโครงสร้างข้อมูลรูปภาพเพื่อให้ได้คุณภาพตัวอักษรที่ดีขึ้น ซึ่งจากการสังเกตของรูปแบบตัวอักษรมีการใช้โครงสร้างทางเรขาคณิตเป็นแบบในการสร้างตัวอักษร โดยใช้การประมวลผลรูปร่างกับโครงสร้างของภาพ (Morphological Image Processing)

#### 2.2.1 การประมวลผลรูปร่างกับโครงสร้างของภาพ (Morphological Image Processing)

เป็นการประมวลผลภาพโดยเกี่ยวกับการแยกส่วนประกอบของภาพออกเพื่อใช้ในการแสดงรูปร่างในเมทริกซ์จะประกอบไปด้วยค่าระดับขาว-ดำคือ 0 และ 1 ซึ่งได้นิยามวิธีการ การทำให้ภาพผอมลง (Thinning) มาใช้สามารถทำได้โดยอาศัย กฎ 2 ข้อ คือ  $n_1$  และ  $n_2$  ดังสมการต่อไปนี้

$$2 \leq \min\{n_1(p), n_2(p)\} \leq 3 \quad (1)$$

เมื่อ

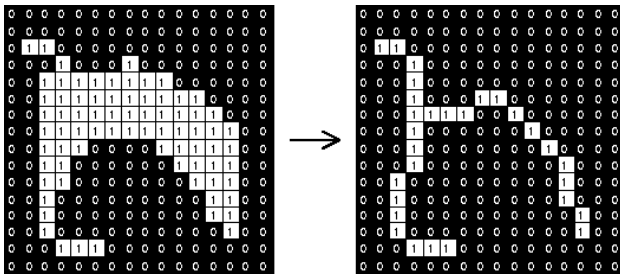
$$n_1(p) = \sum_{k=1}^4 x_{2k-1} \vee x_{2k} \quad (2)$$

$$n_2(p) = \sum_{k=1}^4 x_{2k} \vee x_{2k+1} \quad (3)$$

$n$  คือ Inf (Infinity)

$p$  คือ พิกเซล (Pixel)

$x$  คือ ค่าของ 8 neighbors ของ  $p$  โดยเริ่มจากทางขวาและวนตามเข็มนาฬิกา

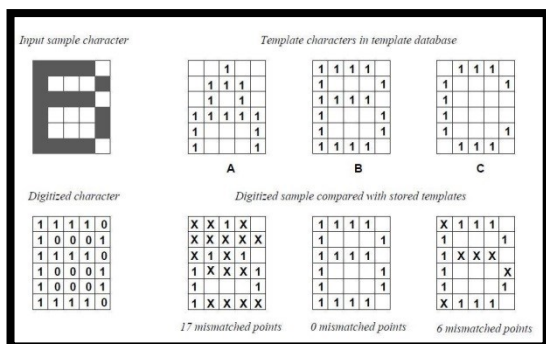


รูปที่ 2. ภาพแสดงผลลัพธ์ของการดำเนินการด้วยวิธี Thinning จาก <http://homepages.inf.ed.ac.uk/rbf/HIPR2/thin.htm>

วิธีการทำ Thinning จะมี 2 ขั้นตอน ซึ่งขั้นตอนแรกจะใช้อักษร  $n_1$  โดยการนำ Template ขนาด 3x3 สแกนไปตามข้อมูลภาพและทำการพิจารณาพิกเซลบริเวณขอบภาพว่าสามารถลบได้หรือไม่ถ้าลบได้ให้หมายเหตุไว้แต่ยังไม่ต้องลบหลังจากที่สแกนทั่วทั้งภาพก็ให้ทำการลบข้อมูลภาพดังที่ได้หมายเหตุไว้ ขั้นตอนที่สองใช้อักษร  $n_2$  และดำเนินการเหมือนการใช้  $n_1$  เมื่อทำการลบข้อมูลภาพที่มีไว้ในหมายเหตุแล้วก็ให้ทำซ้ำต่อไปจนไม่สามารถลบข้อมูลภาพออกได้อีก

### 2.3 การจับคู่แม่แบบ (Template Matching)

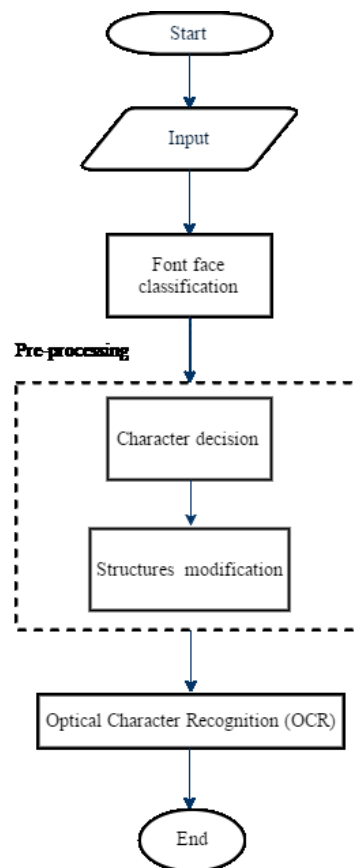
แม่แบบการจับคู่เป็นเทคนิคการมองเห็นระดับสูงที่ระบุส่วนต่างๆ บนรูปภาพที่ตรงกับแม่แบบที่กำหนดไว้ล่วงหน้า ขั้นตอนวิธีการจับคู่แม่แบบขั้นสูงช่วยให้การหาแม่แบบโดยไม่คำนึงถึงความสอดคล้องและความสว่างเฉพาะเทคนิคการจับคู่แม่แบบมีความยืดหยุ่นและค่อนข้างตรงไปตรงมาซึ่งเป็นหนึ่งในวิธีการที่นิยมมากที่สุดของการจำกัดวัตถุโดยมีการทำงานแบบสังเขปดังรูปภาพที่ 3



รูปที่ 3. ภาพการทำงานของ Template Matching จาก <https://www.hackster.io/kapildevkumar/smart-pen-e04e06>

### 3. วิธีดำเนินงาน

จากการศึกษาของทฤษฎีต่างๆ ดังกล่าวจะทำให้ได้แนวคิดและวิธีการแก้ไขปัญหาตามแผนการดำเนินการตามรูปภาพที่ 4 ว่าต้องมีขั้นตอนการดำเนินงานไปในทิศทางใด

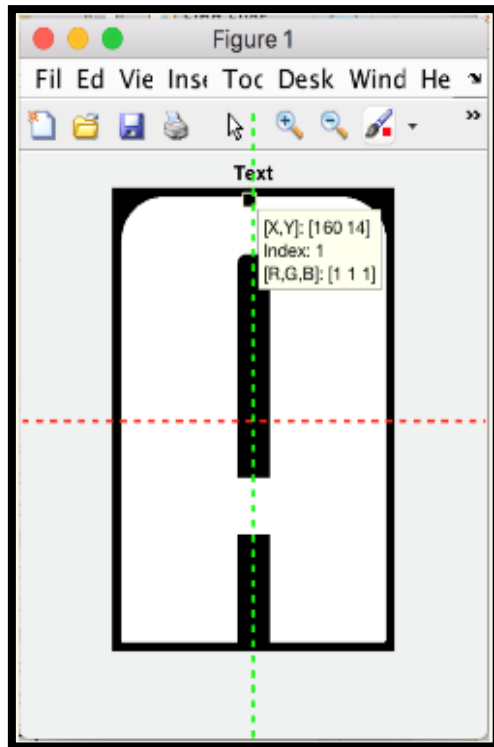


รูปที่ 4. ขอบเขตการดำเนินงาน

#### 3.1 การเลือกกลุ่มตัวอักษร (Font face classification)

โดยใช้การความหนาของช่วงพิกเซลของรูปภาพในการหาลักษณะเฉพาะโดยนำตัวอักษรข้อมูลทั้ง 26 ตัวของกลุ่มตัวอักษรนั้นมาทดสอบ ในที่นี้เลือกข้อมูลจากตัวอักษรศิลป์ของ PUMA, ADIDAS และ NIKE โดยคณะผู้วิจัยได้เลือกตัวอักษรของ PUMA, ADIDAS และ NIKE เพราะเป็นกลุ่มเป้าหมายในปัจจุบันที่มีการนำอักษรไปใช้งานในการทำสินค้าและเป็นกลุ่มสินค้ายอดนิยมในธุรกิจของเสื้อผ้า รองเท้า โดยเมื่อได้นำอักษร มาจัดกลุ่ม โดยดูลักษณะจากภาพความใกล้เคียงกลุ่มตัวอักษรซึ่งจากผลการทดลองพบว่า ลักษณะของชนิดตัวอักษร PUMA นั้นมีความแตกต่างจากลักษณะของชนิดตัวอักษรของ ADIDAS และ NIKE เช่นรูปแบบของตัวอักษรที่แตกต่างกัน และ ความหนาของรูปแบบตัวอักษรที่ไม่มีความสมส่วนในทุกด้านของชนิดตัวอักษร PUMA มีความหนาที่ต่างกันจึงทำให้ผลลัพธ์ของ OCR แสดงผลไม่ถูกต้อง

โดยขั้นตอนการหาค่าความแตกต่างคณะผู้วิจัยจะนำตัวอักษร ศิลป์มาหาโดยการวัดจากแนวแกน X และ แนวแกน Y เพื่อเก็บ ค่าของช่องเมทริกซ์ ค่า 1 และ 0 ของเมทริกซ์มาใช้งาน



รูปที่ 5. ภาพอักษร A ของ PUMA ระยะความแตกต่าง

เมื่อได้ค่าเมทริกซ์ของแนวแกน X และ แนวแกน Y แล้วจะนำค่า เมทริกซ์ที่ได้ในแต่ละแกนมาลบกันทั้ง 2 ค่า โดยค่าที่ได้จะต้องไม่ติดลบจะใช้ค่าสัมบูรณ์ (Absolute value) เก็บค่าใส่ตัวแปรเมื่อได้ทั้ง 26 ตัวอักษรจะมาทำการหาค่าเฉลี่ยของอักษรทั้งหมดเพื่อหาค่าที่อยู่นอกกลุ่มตัวอักษรนี้ แต่ยังเป็นกลุ่มตัวอักษรเดียวกันโดยปรับจุดเริ่มต้นของการแบ่งครึ่งจาก 1 ใน 2 เป็น 1 ใน 3 แทน

### 3.2 การนำภาพมาลบออกจากกัน (Subtraction)

จากการดำเนินงานใน 3.1 นั้นเราจะได้กลุ่มตัวอักษรที่แบ่งแยกกันชัดเจนแล้วต่อไปจะทำการ Subtraction ภาพเพื่อระบุวิธีการว่าควรใช้รูปแบบวิธีคิดของตัวอักษรชนิดไหน โดยการนำภาพใน ฐานข้อมูลภาพมาลดขนาดรูปภาพ (Resize) ให้เท่ากันกับภาพนำเข้า แล้วนำมาลบกันจะทำให้ค่าของภาพนั้นมีค่าเข้าใกล้ 0 ซึ่งก็คือตัวอักษรตัวนั้นออกมา

### 3.3 เลือกวิธีการเฉพาะของกลุ่มตัวอักษร

ใช้เรขาคณิตโดยจัดกลุ่มได้ตามตารางที่ 1

ตาราง 1. การจัดกลุ่มเลือกใช้เรขาคณิตในการพิสูจน์ตัวอักษร

สามเหลี่ยม	สี่เหลี่ยม	วงรี	อื่นๆ
A	C	P	B
M	O		D
N	S		H
V			U
W			ฯลฯ

เมื่อได้ภาพตัวอักษรที่ต้องการแล้วจะทำการเลือกวิธีการของเป้าหมายจากขั้นตอน การนำภาพมาลบออกจากกัน (Subtraction) ซึ่งได้ผลลัพธ์เท่ากับตัวอักษรที่มีอยู่ใน Template นำภาพมา ตัดออก (Crop) โดยการหาพื้นที่ของสี่เหลี่ยม (Bounding Box) ของรูปภาพเมื่อได้สัดส่วนของรูปภาพมาแล้วจึงนำเข้าวิธีการลดรูปตัดแปลงตามลักษณะเฉพาะของตัวอักษร ซึ่งจะถูกรับโครงสร้าง(Mask) แบบสามเหลี่ยม สี่เหลี่ยม วงรีหรือเรขาคณิตแบบอื่นๆ ตามความเหมาะสมของตัวอักษร ซึ่งในที่นี้จะขอยกตัวอย่างการทำอักษรตัว M โดยจะใช้สมการพีทาโกรัสมาช่วยในการ Pre-processing [6] จากสมการ

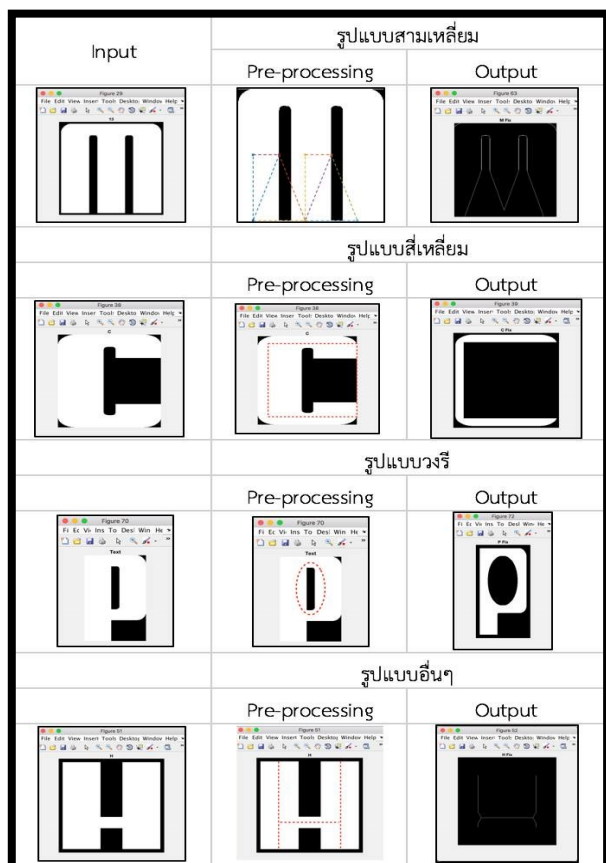
$$C^2 = X^2 + Y^2 \quad (4)$$

$C$  คือ ผลลัพธ์ของด้านตรงข้ามมุมฉาก

$X$  คือ ความยาวของฐานภาพ สามเหลี่ยม

$Y$  คือ ความสูงของภาพจากจุดยอดที่หาได้

จากสมการที่ (4) จะทำให้เรารู้ความยาวของด้านตรงข้ามมุมฉาก เพื่อช่วยในปรับโครงสร้าง (Mask) แบบสามเหลี่ยมมาและทำแบบเดียวกันกับสามเหลี่ยมรูปถัดไปจากนั้นจึงหาหาขอบของวัตถุในภาพด้วยวิธีการของ Canny (Canny Edge Detection)



ตาราง 2. การปรับโครงสร้างของรูปภาพ(Mask) และการหาขอบภาพแบบต่าง ๆ

### 3.4 การเลือกนำเข้า OCR

กระบวนการ OCR เป็นกระบวนการที่สามารถประมวลผลแปลงภาพของข้อความจากการเขียนหรือจากการพิมพ์ให้อยู่ในรูปแบบของตัวอักษรมาตรฐานได้ถึง 90% ของทั้งหมด เนื่องจากชุดข้อมูลภาพที่ใช้ทดลองมีลักษณะและชนิดเป็นตัวอักษรจึงได้นำกระบวนการ OCR มาใช้ในการวิเคราะห์ผลลัพธ์

เมื่อได้ผลลัพธ์จากการปรับปรุงโครงสร้างของกระบวนการ Pre-processing แล้วจะทำให้ข้อมูลเหล่านั้นมีลักษณะข้อมูลที่เหมาะสมกับกระบวนการในการนำเข้า OCR ได้ ซึ่งในขั้นตอนส่วนนี้ไม่ได้มีความซับซ้อนมากนักเพราะทางคณะผู้วิจัยได้ใช้เทคนิค OCR เป็นขั้นตอน Post-processing สำหรับการประมวลผลลักษณะของตัวอักษรจากชุดข้อมูลนำเข้าจากกระบวนการ Pre-processing หลังจากนั้นจะนำผลลัพธ์ที่ได้ไปดำเนินการในขั้นตอนถัดไป

### 3.5 จัดเก็บผลลัพธ์เทียบอักษรแต่ละตัวจากผลลัพธ์ของ OCR

ผลลัพธ์ที่ได้เมื่อนำภาพนำเข้าผ่านกระบวนการ OCR จะถูกแปลงให้อยู่ในรูปแบบของตัวอักษรหรือชุดตัวอักษร แต่ความถูกต้องอาจจะยังไม่สมบูรณ์เนื่องจากกระบวนการ OCR จำเป็นต้องอ่าน

ชุดข้อมูลนำเข้าเทียบกับกลุ่มคำในฐานข้อมูลที่เก็บไว้เพื่อดูว่าตัวอักษรมีความถูกต้องตามรูปแบบของตัวอักษรหรือคำที่มีความหมายหรือไม่ แต่ผลลัพธ์อาจไม่ถูกต้องเท่าที่ควรจึงจำเป็นต้องกระบวนการ Pre-processing เพื่อปรับปรุงโครงสร้างของภาพนำเข้าให้มีรูปแบบใกล้เคียงกับฐานข้อมูลของ OCR มากที่สุด

## 4. ผลการดำเนินงาน

เริ่มต้นกระบวนการวิจัย ทางคณะผู้วิจัยได้ทำการทดสอบนำชุดข้อมูลภาพ PUMA, ADIDAS และ NIKE จำนวน 30 ภาพ เข้าสู่กระบวนการ OCR โดยแปลงเป็นข้อมูลภาพเป็นภาพไบนารี ซึ่งผลลัพธ์ของการทดสอบการนำเข้า OCR ไม่ถูกต้องเท่าที่ควร ซึ่งจากการทดลองพบว่า ลักษณะของชนิดตัวอักษร PUMA นั้นมีความแตกต่างจากลักษณะของชนิดตัวอักษรของ ADIDAS และ NIKE เช่นรูปแบบของตัวอักษรที่แตกต่างกันและความหนาของรูปแบบตัวอักษรที่ไม่มีความสมส่วนในทุกด้านของชนิดตัวอักษร และมีมุมมองของตัวอักษรในข้อมูลภาพนำเข้าที่เอียงมากกว่าหรือน้อยกว่า 90 องศา ดังนั้นข้อมูลนำเข้าจึงควรมีมุม 90 องศาและมีขนาดที่เหมาะสมเพื่อให้ได้ผลลัพธ์ที่ชัดเจน

คณะผู้วิจัยจึงได้นำเสนอการหาลักษณะภาพก่อน (Pre-processing) เพื่อจัดกลุ่มตัวอักษรให้เหมาะสมโดยการหาค่าความหนาแน่นของแนวแกน X และ แนวแกน Y ของแต่ละตัวอักษรในข้อมูลภาพ โดยวิธีการที่นำเสนอจะแตกต่างกันออกไปตามชนิดของตัวอักษรเนื่องจากชนิดของตัวอักษรมีลักษณะเฉพาะในบางจุดและมีความแตกต่างกัน จากนั้นนำผลลัพธ์ที่ได้จากการทำกระบวนการ Pre-processing เข้าสู่กระบวนการ OCR ทำให้ได้ผลลัพธ์ที่แตกต่างออกไปจากเดิมที่นำชุดข้อมูลภาพเข้ากระบวนการ OCR เลย โดยภาพที่ผ่านการทำ Pre-processing ก่อนนำเข้ากระบวนการ OCR ให้ผลลัพธ์ความถูกต้องมากกว่าเดิมถึง 80% ซึ่งผลลัพธ์จากการทดลองยังมีบางตัวอักษรที่ยังไม่ได้ผลลัพธ์ตามที่ต้องการจึงจำเป็นต้องใช้วิธีอื่นในการเข้ามาช่วยเพิ่มประสิทธิภาพในการอ่านตัวอักษรเพื่อให้ความถูกต้องเพิ่มมากขึ้น

ตาราง 3. ประสิทธิภาพความถูกต้องในการอ่านตัวอักษร

ตัวอักษร PUMA		
Text	OCR	Preprocessing
A	'I' (ไอ)	A
B	'I'	B
C	'E'	C
D	'I' (ไอ)	D
E	'E'	E
F	'F'	F
G	' '	G
H	' '	H
I(i)	' '	I
J	'J'	J
K	' '	K
L	'L'	L
M	' '	M
N(n)	' '	N
O	'I' (แอล)	O
P	'I' (ไอ)	P
Q	'I' (ไอ)	Q
R	'I' (ไอ)	R
S	' '	S
T	'T'	T
U	' '	U
V	' '	V
W	' '	W
X	' '	X
Y	'[]'	Y
Z	'Z'	Z

ตาราง 4. ประสิทธิภาพความถูกต้องในการอ่านตัวอักษร PUMA แต่ละขนาด

ตัวอักษรชนิด PUMA			
Size	OCR	Temple Matching	Preprocessing + OCR
50 พ้อย (101*72)	23.07%	80.77%	92.30%
100 พ้อย (181*123)	23.07%	88.46%	92.30%
150 พ้อย (261*174)	23.07%	88.46%	92.30%
200 พ้อย (341*225)	23.07%	88.46%	92.30%
250 พ้อย (421*277)	23.07%	96.15%	92.30%
300 พ้อย (500*328)	23.07%	88.46%	92.30%
350 พ้อย (581*380)	23.07%	92.31%	92.30%
400 พ้อย (661*431)	23.07%	88.46%	92.30%



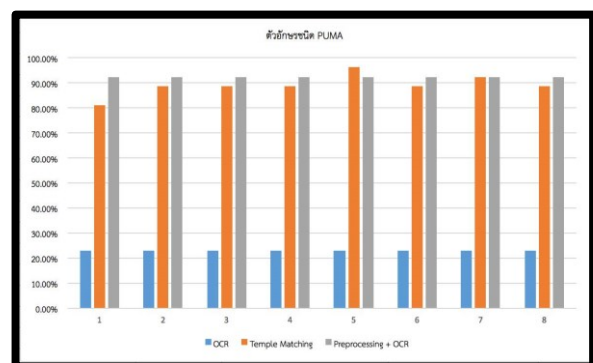
รูปที่ 6. ภาพตัวอย่างของตัวอักษร ADIDAS



รูปที่ 7. ภาพตัวอย่างของตัวอักษร NIKE



รูปที่ 8. ภาพตัวอย่างของตัวอักษร PUMA



รูปที่ 9. ประสิทธิภาพความถูกต้องแต่ละขนาด

## 5. สรุปผลทดลอง

จากปัญหาการพัฒนาตัวอักษรให้มีความสวยงามมากขึ้นจากอักษรมาตรฐานทั่วไปและถูกนำมาใช้งานในการเขียนลงบนสิ่งพิมพ์มากขึ้นทำให้การจัดเก็บข้อมูลจากสิ่งพิมพ์หรือรูปภาพลงคอมพิวเตอร์ด้วยโปรแกรม OCR นั้นมีปัญหาเกี่ยวกับการอ่านตัวอักษรศิลป์เป็นอย่างมาก จึงได้เสนอการวิเคราะห์หารูปลักษณะของตัวอักษรศิลป์เพื่อเพิ่มประสิทธิภาพให้กับโปรแกรม OCR และได้ใช้ขั้นตอนการทำการประมวลผลขั้นต้น (Pre-processing) เพิ่มขึ้นมาเป็นตัวช่วยเพิ่มประสิทธิภาพ ผลทดลองกับตัวอักษรชนิด PUMA ,ADIDAS และ NIKE พบว่าอักษรศิลป์ที่นำเข้า OCR โดยไม่ทำ Pre-processing ก่อนนั้นให้ผลลัพธ์ลดลงจากการเปรียบเทียบกับตัวอักษรมาตรฐานถึง 80% ของรูปภาพตัวอักษรที่ใช้ในการทดลองและการทดลองใช้วิธีการ

เปรียบเทียบรูปแบบ (Template Matching) ตัวอักษรนั้นให้ผลลัพธ์ในรูปแบบที่เป็นแค่ตัวอักษรที่ไม่ใช่กลุ่มคำถึง 100% แต่ถ้าหากเป็นกลุ่มคำความถูกต้องในการเปรียบเทียบจะน้อยลงตามลำดับเพราะไม่มีการจัดรูปแบบของตัวอักษรเพื่อใช้ในการเปรียบเทียบทำให้ตัวอักษรมีการซ้อนทับกันในบางครั้งและเนื่องจากต้องคอยกำหนดรูปแบบตัวอักษรเพื่อใช้เป็นตัวเปรียบเทียบนั้นจึงจำเป็นต้องมีการเก็บชนิดของตัวอักษรไว้ให้ได้มากที่สุดเพื่อความถูกต้องของการนำไปใช้แต่หากนำตัวอักษรที่ผ่านขั้นตอนการประมวลผลขั้นต้นโดยใช้วิธีวิเคราะห์รูปแบบของตัวอักษรแล้วนำเข้า OCR นั้น มีผลลัพธ์ที่ดีขึ้นเมื่อเทียบกับตัวอักษรมาตรฐานประมาณ 96% จากรูปภาพที่ใช้ในการทดลองจะเห็นได้ว่าการทำ Pre-processing ก่อนนั้นสามารถช่วยเพิ่มประสิทธิภาพของโปรแกรม OCR เพิ่มขึ้นจากเดิมถึง 96% และจากการเปรียบเทียบ (Template Matching) ถึง 10% ทำให้ข้อมูลในการจัดเก็บลงคอมพิวเตอร์มีความถูกต้องหรือนำไปใช้ประโยชน์ในด้านต่างๆได้อย่างมีประสิทธิภาพ

## 6. เอกสารอ้างอิง

- [1] OCR Introduction". Dataid.com. Retrieved 2013-06-16.
- [2] พิษญา แจ่มจันทร์ และ ญัฐชา เดชดำรง “ระบบรู้จำชนิดแบบอักษรด้วยแสง”วารสารคอมพิวเตอร์ และเทคโนโลยี ชั้นสูง ฉบับที่ 12 ต.ค. 2554 หน้า 57-62.
- [3] ความหมายของ OCR [https://en.wikipedia.org/wiki/Optical\\_character\\_recognition](https://en.wikipedia.org/wiki/Optical_character_recognition)
- [4] การทำงานErosion [http://www.wbi.msu.ac.th/file/721/doc\\_42.pdf](http://www.wbi.msu.ac.th/file/721/doc_42.pdf)
- [5] โปรแกรม OCR <https://www.mathworks.Com/products/matlab.html>
- [6] Font Download MY PUMA แหล่งที่มา: <http://www.dafont.com/my-puma.font>
- [7] Font Download MY PUMA แหล่งที่มา: <http://www.dafont.com/my-puma.font>
- [8] Font Download ADIDAS แหล่งที่มา: <http://famfonts.com/adidas>
- [9] Font Download NIKE แหล่งที่มา: <http://www.dafont.com/forum/read/72741/nike-font>
- [10] โครงข่ายประสาทเทียม คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยพระจอมเกล้าพระนครเหนือ