

รายงานการปฏิบัติงานสหกิจศึกษา

เรื่อง

ระบบตรวจหาข้อความบนภาพมังงะด้วยเทคนิค Stroke Width

Transform 5555555

**DETECTING TEXT IN MANGA USING STROKE WIDTH
TRANSFORM**

ปฏิบัติงาน ณ มหาวิทยาลัยออกไกโด

โดย

บุญฤทธิ์ พริย์ไยชินกุล
รหัสประจำตัว 58070077

รายงานนี้เป็นส่วนหนึ่งของการศึกษารายวิชา สหกิจศึกษา
สาขาวิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 1 ปีการศึกษา 2561
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

รายงานการปฏิบัติงานสหกิจศึกษา
ระบบตรวจหาข้อความบนภาพมังงะด้วยเทคนิค Stroke Width
Transform 555555

**DETECTING TEXT IN MANGA USING STROKE WIDTH
TRANSFORM**

ปฏิบัติงาน ณ มหาวิทยาลัยฮอกไกโด

โดย

นุญฤทธิ์ พิริย์โยธินกุล
รหัสประจำตัว 58070077

ปฏิบัติงาน ณ มหาวิทยาลัยฮอกไกโด
Hokkaido University Kita 8, Nishi 5, Kita-ku,
Sapporo, Hokkaido, 060-0808 Japan
Web site : www.global.hokudai.ac.jp

**DETECTING TEXT IN MANGA USING STROKE WIDTH
TRANSFORM**

BOONYARITH PIRIYOTHINKUL

**A REPORT SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENT FOR COOPERATING EDUCATION PROGRAM
THE DEGREE OF BACHELOR OF SCIENCE PROGRAM IN
INFORMATION TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

COPYRIGHT 2018
FACULTY OF INFORMATION TECHNOLOGY
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

วันที่ 10 พฤษภาคม พ.ศ. 2561

เรื่อง ขอส่งรายงานการปฏิบัติงานสหกิจศึกษา
เรียน รองศาสตราจารย์ ดร. กิตติสุชาต พสุภา¹
ที่ปรึกษาสหกิจศึกษาในสาขาวิชาเทคโนโลยีสารสนเทศ

ตามที่ข้าพเจ้า บุญฤทธิ์ พิริยะчинกุล นักศึกษาสาขาวิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ได้ปฏิบัติงานสหกิจศึกษาระหว่างวันที่ 23 กรกฎาคม พ.ศ. 2561 ถึงวันที่ 30 พฤษภาคม พ.ศ. 2561 ในตำแหน่ง ผู้ช่วยนักวิจัย ณ สถานประกอบการชื่อ มหาวิทยาลัยออกไกโด และได้รับมอบหมายจากพนักงานที่ปรึกษาให้ศึกษาและจัดทำรายงาน เรื่อง ระบบตรวจหาข้อความบนภาพมังงะด้วยเทคนิค Stroke Width Transform 5555555

บันทึก การปฏิบัติงานสหกิจศึกษาได้ล้วนสุดลงแล้ว จึงได้ขอส่งรายงานการปฏิบัติงาน สหกิจศึกษาดังกล่าวมาพร้อมนี้ จำนวน 1 เล่ม เพื่อขอรับคำปรึกษาต่อไป

จึงเรียนมาเพื่อโปรดพิจารณา

ขอแสดงความนับถือ

(บุญฤทธิ์ พิริยะчинกุล)

กิตติกรรมประกาศ

ตามที่ข้าพเจ้า บุญฤทธิ์ พิริย์ไชนกุล ได้มาปฏิบัติงานสหกิจศึกษา ณ มหาวิทยาลัยออกไกโอด ตั้งแต่วันที่ 23 กรกฎาคม พ.ศ. 2561 ถึงวันที่ 30 พฤษภาคม พ.ศ. 2561 ทำให้ข้าพเจ้าได้รับความรู้ และประสบการณ์ต่าง ๆ ที่มีคุณค่ามากmany สำหรับรายงานสหกิจศึกษานั้น สำเร็จลงได้ด้วยดี จากความช่วยเหลือและความร่วมมือสนับสนุนของหลายฝ่าย ดังนี้

1. Professor Dr. Masanori Sugimoto ตำแหน่ง ศาสตราจารย์
2. Jiang Ye ตำแหน่ง นักศึกษาปริญญาเอก ปี 2

นอกจากนี้ยังมีบุคคลท่านอื่น ๆ อีกที่ไม่ได้กล่าวไว้ ณ ที่นี้ ซึ่งให้ความกรุณาแนะนำในจัดทำรายงานสหกิจศึกษานั้น ข้าพเจ้าจึงขอขอบพระคุณทุกท่านที่ได้มีส่วนร่วมในการให้ข้อมูล และให้ความเข้าใจเกี่ยวกับชีวิตของการปฏิบัติงาน รวมถึงเป็นที่ปรึกษาในการจัดทำรายงานฉบับนี้ จนเสร็จสมบูรณ์

บุญฤทธิ์ พิริย์ไชนกุล
ผู้จัดทำรายงาน
วันที่ 10 พฤษภาคม พ.ศ. 2561

ชื่อรายงานการปฏิบัติงานสหกิจศึกษา ระบบตรวจหาข้อความบนภาพมังงะด้วยเทคนิค Stroke
Width Transform 5555555

ผู้รายงาน บุญฤทธิ์ พิริยะธนกุล
คณะ เทคโนโลยีสารสนเทศ
สาขาวิชา เทคโนโลยีสารสนเทศ

.....
(รองศาสตราจารย์ ดร. กิตติสุชาต พสุภา)
อาจารย์ที่ปรึกษาสหกิจศึกษา

.....
(Professor Dr. Masanori Sugimoto)
พนักงานที่ปรึกษา

คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
อนุมัติให้นับรายงานการปฏิบัติงานสหกิจศึกษานานบัน妮 เป็นส่วนหนึ่งของการศึกษา
ตามหลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

ชื่อรายงาน	ระบบตรวจหาข้อความบนภาพมังะด้วยเทคนิค Stroke Width Transform
	5555555
ชื่อนักศึกษา	บุญฤทธิ์ พิริย์ไชยชินกุล
รหัสนักศึกษา	58070077
สาขาวิชา	เทคโนโลยีสารสนเทศ
อาจารย์ที่ปรึกษา	รองศาสตราจารย์ ดร. กิตติสุชาต พสกานา
ปีการศึกษา	2561

บทคัดย่อ

การรุ่นปูน หรือที่รู้จักกันอย่างแพร่หลายว่า มังงะ (Manga) ถูกเป็นหนึ่งในหัวข้อที่ถูกหยิบมาวิจัย ในงานวิจัยชิ้นนี้มุ่งเน้นไปที่ปัญหาการตรวจหาข้อความบนภาพมังงะ เนื่องจากปัญหาการสร้างชุดข้อมูลภาพมังงะและข้อมูลประกอบ (Annotation) อย่างเช่นการระบุขอบเขตข้อความ ซึ่งต้องใช้แรงงานคนและกินเวลาอย่างมาก ดังนั้นการพัฒนาระบบอัตโนมัติที่จะสามารถเข้ามาช่วยในงานส่วนนี้ได้จึงเป็นสิ่งที่น่าสนใจเป็นอย่างมาก โดยเราได้นำเสนอวิธีการตรวจหาข้อความบนภาพมังงะแบบใหม่ด้วยการใช้ Stroke Width Transform (SWT) ร่วมกับการใช้ Support Vector Machine (SVM) อย่างไรก็ได้ วิธี SWT โดยดึงเดินน้ำเสียงพัฒนาขึ้นเพื่อ เพื่อการตรวจหาข้อความบนภาพถ่ายทำให้ไม่สามารถประยุกต์ใช้กับภาพมังงะได้ เพราะความแตกต่างระหว่างลักษณะวัตถุกับอักษรของข้อความในภาพวาดและภาพถ่ายนั้นคล้ายคลึงมากกันเกินไป ดังนั้นเพื่อให้สามารถใช้งานกับมังงะได้ เราจึงนำวิธีการตรวจหาข้อความด้วย SWT ดึงเดินมาปรับปรุงและพัฒนาขึ้นเป็นวิธีการใหม่ของเราวิธีการ SWT ได้ปรับปรุงกฎเกณฑ์ในการค้นหาวัตถุที่คล้ายคลึงอักษร (Letter Candidates) ซึ่งช่วยเพิ่มประสิทธิภาพในการตรวจจับอักษร ได้ครอบคลุมมากขึ้น และใช้ SVM เพื่อคัดแยกวัตถุอื่น ๆ ออกจากอักษร ช่วยในการลด False Positive ของผลลัพธ์ ในท้ายที่สุดเรานำประสิทธิภาพวิธีการของเราเบรียบเทียบรวมกับวิธีการดึงเดินและวิธีอื่น ๆ รวมถึงวิธีที่ใช้ Deep Learning เป็นส่วนประกอบ ในท้ายที่สุดประสิทธิภาพของวิธีการใหม่ของเรานั้นสามารถทำคะแนน F-measure ได้สูงสุดเทียบกับวิธีการอื่น ๆ ที่ 0.506

Project Title	Detecting Text in Manga Using Stroke Width Transform
Student	Boonyarith Piriyothinkul
Student ID	58070077
Program	Information Technology
Advisor	Associate Professor Dr. Kitsuchart Pasupa
Year	2018

Abstract

The Japanese comic-book style known as manga is becoming a popular topic for researchers. This paper focuses on the problem of detecting text regions in manga pages. Because it is time-consuming and laborious to identify the text regions in images manually, an automatic approach is highly desirable. Here, we propose a new text-detection method for manga using a Stroke Width Transform (SWT) technique in conjunction with a Support Vector Machine (SVM). Conventional SWT-based text-detection techniques perform poorly with manga because both text and non-text objects have similar characteristics for strokes, lines, and shapes. To better suit manga, we propose modifying the rules for finding letter candidates, which improves the ability to capture text. An SVM is then used to classify image patches into letter and nonletter regions. We compared our proposed framework with a conventional framework and other text-detection methods including deep-learning techniques. In the results, our proposed method achieved the highest F-measure of 0.506.

สารบัญ

หน้า

สารบัญตาราง

ตารางที่

หน้า

សារណ្ឌុប្បរព

រូបភ័

អនុញ្ញាត

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

การตูนญี่ปุ่นเป็นที่รักกันอย่างแพร่หลายทั่วโลกในฐานะสื่อบันเทิง หรืออีกชื่อหนึ่งคือ “มังงะ (Manga)” ในปัจจุบันมีงานวิจัยในหัวข้อมังงะอย่างหลากหลาย ในหลาย ๆ งานวิจัย [?, ?, ?, ?, ?, ?, ?] มีการใช้ชุดข้อมูลสำหรับการทดลอง เช่น Manga109 [?] ซึ่งเป็นชุดข้อมูลที่ถูกสร้างขึ้นจากภาพมังงะจำนวน 20,260 หน้า รวบรวมจากมังงะ 109 เรื่อง มังงะที่ถูกรวมรวมมาเนี้ยเป็นผลงานของนักวาดมังงะมืออาชีพชาวญี่ปุ่น นอกจากการพองมังงะแล้ว ชุดข้อมูลนี้ยังประกอบไปด้วยข้อมูลอธิบายประกอบ หรือ Annotation ต่าง ๆ เช่น ขอบเขตและตำแหน่งของใบหน้า ร่ายกาย และ กรอบภาพ เป็นต้น นอกจากนี้ยังมีข้อมูลขอบเขตและตำแหน่งของข้อความที่ปรากฏในภาพมังงะ โดยตำแหน่ง ข้อความต่าง ๆ นั้นถูกป้อนข้อมูลด้วยแรงงานคนโดยไม่พึงพาระบบอัตโนมัติ ในการป้อนข้อมูล ดังกล่าวเน้นใช้เวลานานและต้องพึงพาแรงงานมนุษย์ ด้วยเหตุนี้ระบบอัตโนมัติที่จะสามารถเข้ามาร่วมในการระบุข้อมูล Annotation นั้นจึงมีประโยชน์และสามารถช่วยลดภาระงานในส่วนนี้ลงได้อย่างมาก

ถึงแม้ว่าสำหรับภาพมาตรฐานรูปแบบการตูนญี่ปุ่นจะมีทั้งแบบภาพวาดทั่วไปที่เป็นภาพแสดงของตัวละคร หรือทิวทัศน์ และแบบภาพมังงะ แต่ภายในงานวิจัยนี้เรามุ่งเน้นไปที่มังงะเป็นหลักเนื่องจาก ข้อความมักปรากฏบนหนังสือการตูนมากกว่าภาพวาดทั่วไปอย่างที่ทราบกันดี สำหรับวิธีการตรวจหาข้อความในภาพมังงะนั้นมีการพัฒนามาหลากหลายก่อนหน้านี้ [?, ?] แต่วิธีเหล่านี้ถูกพัฒนาให้พึงพา โครงสร้างส่วนต่าง ๆ ของภาพมังงะเป็นข้อมูลอ้างอิง เช่น กรอบช่องภาพวาด, ลักษณะของกล่องคำพูด เป็นต้น นอกจากนี้บางวิธียังคงมีความจำเป็นที่ต้องพึงพาการป้อนข้อมูลเข้าจากภายนอก ทั้งจากมนุษย์และข้อมูลอื่น ๆ ทำให้ไม่สามารถทำงานได้อัตโนมัติอย่างสมบูรณ์ อย่างไรก็ดีไม่นาน นานนี้มีการพัฒนาวิธีการใหม่ โดยใช้วิธีการ Deep Learning อย่างเช่นเทคนิค Convolutional Neural Network เพื่อช่วยในการสกัดลักษณะเด่น (Feature) ออกจากภาพมังงะเพื่อช่วยในการตรวจหา ข้อความในภาพมังงะ [?] ซึ่งวิธีการนี้สามารถเพิ่มความแม่นยำและถูกต้องในการตรวจหาข้อความ ได้โดยปราศจากการพึงพา โครงสร้างต่าง ๆ ในภาพมังงะ แต่อย่างไรก็ดี Deep Learning ยังเป็นวิธีการที่ต้องใช้ทรัพยากรของระบบเพื่อการคำนวนมากกว่าวิธีการอื่น ๆ ซึ่งเป็นข้อเสียสำคัญของการประยุกต์ [?]

ในงานวิจัยนี้มีจุดมุ่งหมายเพื่อพัฒนาระบบตรวจหาข้อความที่ทำงานได้กับมังงะอย่างหลากหลาย และไม่ถูกจำกัดด้วยโครงสร้างหรือลักษณะบางประการของภาพมังงะ เราจึงเลือกใช้ Stroke Width Transform (SWT) ใน การสกัดลักษณะเด่นของเส้นต่าง ๆ ของวัตถุที่ปรากฏในภาพออก มา โดยวิธีการนี้ถูกใช้เป็นขั้นตอนแรกของการตรวจหาข้อความบนภาพถ่ายมาก่อนหน้านี้ วิธีการนี้ทำงานโดยพึงพาสมมติฐานว่าขอบของเส้นอักษรในข้อความนั้นมีขอบที่ชัดเจนและหนาแน่น ปรากฏอยู่บนพื้นหลังที่รับเรียน [?] อย่างไรก็ได้การใช้วิธีการนี้กับการตรวจหาข้อความบนภาพมังงะ ส่งผลให้เกิดข้อผิดพลาดเชิง False Positive จำนวนมาก ปัญหานี้เกิดจากความแตกต่างของลักษณะ

เฉพาะตัวของภาพถ่ายและภาพวาดมังงะ ภาพมังงะนั้น โดยส่วนใหญ่มีลักษณะเป็นภาพขาวดำ และลักษณะของวัตถุภายในมังงะ เช่น ขนาด, เส้น, และพื้นหลัง นั้นมีความคล้ายคลึงกับตัวอักษรของข้อความ ด้วยปัญหาข้างต้นเราจึงต้องปรับปรุงและพัฒนา SWT ที่ถูกใช้ในภาพถ่าย [?] เพื่อให้สามารถทำงานกับภาพมังงะได้

วิธีการใหม่ของเรานี้ที่ถูกพัฒนาขึ้นใหม่นั้นแบ่งออกเป็น 4 ส่วนดังนี้ (i) The Stroke Width Transform (ii) คืนหายาตุที่เข้าข่ายลักษณะของตัวอักษร (iii) คัดแยกอักษร โดยใช้ Support Vector Machine (SVM) ร่วมกับ Histogram of Oriented Gradients Feature (iv) จัดกลุ่มอักษรที่ผ่านการคัดแยกแล้วให้เกิดเป็นบรรทัดหรือกลุ่มของข้อความ

1.2 วัตถุประสงค์

พัฒนาระบบค้นหาตามแน่นข้อความสำหรับมังงะ โดยนำ Stroke Width Transform ที่ถูกใช้เป็นกระบวนการเรียนรู้ในเทคนิคตรวจหาข้อความบนภาพถ่ายมาพัฒนาและปรับปรุงต่อขึ้นเพื่อให้สามารถใช้งานกับภาพมังงะได้มีประสิทธิภาพมากขึ้น

1.3 ขอบเขตของงานวิจัย

1. พัฒนาระบบตรวจหาตามแน่นข้อความซึ่งใช้สำหรับภาพมังงะ โภนสีขาวดำ
2. ภาษาของเนื้อหาในมังงะที่นำมาใช้งาน คือ ภาษาญี่ปุ่น
3. ข้อมูลที่ใช้ในการวิจัยเพื่อการเทรนและทดสอบนำมาจากฐานข้อมูล Manga109

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้วิธีการตรวจหาข้อความใหม่ที่ถูกพัฒนาขึ้นเพื่อใช้งานร่วมกับภาพมังงะโดยเฉพาะ
2. ทำให้ทราบถึงลักษณะที่เป็นเอกลักษณ์ของมังงะซึ่งแตกต่างจากภาพถ่ายทั่วไป

บทที่ 2

แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 การตรวจหาข้อความในภาพถ่ายด้วยเทคนิค Stroke Width Transform

วิธีการการตรวจหาข้อความในภาพถ่าย [?] นี้เป็นวิธีการที่เรานำมาใช้ศึกษาและเป็นต้นแบบในการพัฒนาเพื่อทำงานร่วมกับภาพมังงะ โดยมีขั้นตอนทั้งหมดแบ่งได้เป็น 3 ขั้นตอน ขั้นแรกคือการใช้ Stroke Width Transform ในการปรับเปลี่ยนข้อมูลให้แสดงลักษณะของความกว้างในแต่ละเส้นภายในภาพ ขั้นที่สอง ค้นหาวัตถุที่คล้ายคลึงกับตัวอักษรในภาพ โดยใช้กฎเกณฑ์ที่กำหนดไว้ ขั้นสุดท้าย คือ การจัดกลุ่มตัวอักษรเข้าด้วยกันเป็นบรรทัดของข้อความ

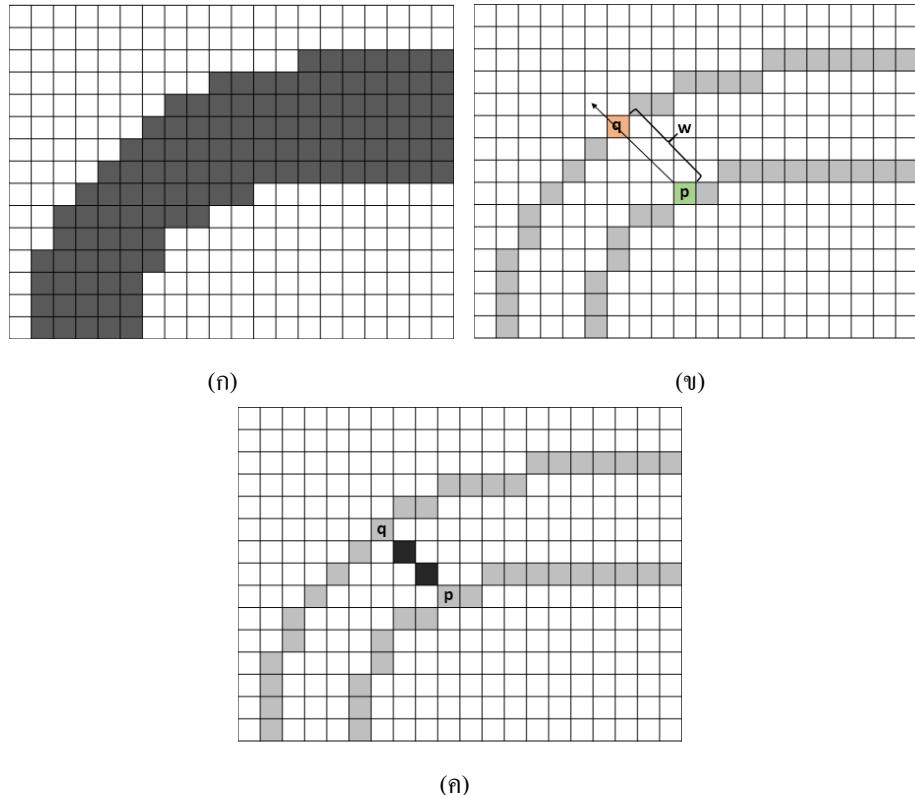
2.1.1 Stroke Width Transform

Stroke Width Transform หรือ SWT เป็นเทคนิคที่ใช้ช่วยในการทำงานของระบบตรวจหาข้อความในภาพถ่าย โดยสกัดลักษณะเด่นของเส้นต่าง ๆ ในภาพ เช่น เส้นของตัวอักษร เป็นต้น [?] ด้วยลักษณะดังกล่าว ทำให้เราสามารถใช้ในการคัดแยกวัตถุที่เป็นตัวอักษรออกจากวัตถุอื่น ๆ โดยพึ่งพาลักษณะเด่นเหล่านี้

เริ่มแรกเราสร้างภาพ Output ที่มีขนาดเท่ากับภาพที่ต้องการตรวจหาข้อความ โดยแต่ละ Pixel ในภาพถูกกำหนดให้มีค่าอนันต์ (∞) จากนั้นใช้ Canny Edge Detection [?] ตรวจหาตำแหน่งของขอบของวัตถุในภาพ ซึ่งจากตัวอย่างในภาพ หากเรามีภาพ ?? จากนั้นใช้การตรวจหาตำแหน่งของขอบของวัตถุ เราจะได้ผลลัพธ์ตามภาพ ?? เมื่อตรวจหาตำแหน่งของเส้นสิ้นต่อมาจะคำนวณความกว้างของเส้นโดยใช้ขอบที่ได้มา ความกว้างคำนวนได้จากการห่างระหว่างขอบของเส้น โดยพิจารณาทุก Pixel p ของขอบที่ได้จาก Canny Edge Detection เพื่อหา Pixel q ที่เข้าใกล้กันอย่างที่แสดงให้เห็นในภาพ ?? การหา q จาก p ทำได้โดยใช้ Gradient Direction ของ p ซึ่งคือ d_p โดย d_p จะชี้ไปหา q และหาก d_p และ d_q มีทิศทางตรงกันข้าม โดยประมาณ $d_q = -d_p \pm \pi/6$ ให้กำหนดค่าให้กับแต่ละ Pixel ที่อยู่ใกล้ได้แนวทางห่างระหว่าง p และ q ให้มีค่าเท่ากับ $\|\overrightarrow{p-q}\|$ เนื่องแต่ว่า Pixel ที่จะระบุค่าให้นั้นมีค่าเดิมน้อยกว่าค่าใหม่ที่จะระบุให้ ดังนั้นหากค่าใหม่น้อยกว่าค่าเดิมใน Pixel ก็สามารถทำการระบุค่าใหม่แทนที่ค่าเดิมให้กับ Pixel นั้นได้ อย่างที่ปรากฏในภาพ ?? เมื่อทำการบุก Pixel ในภาพ สุดท้ายจะได้เมทริกซ์ Output ขนาดเท่ากับ Input โดยมีค่าของความกว้างเส้นถูกระบุนในพื้นที่ระหว่างขอบของเส้นอย่างเช่นปรากฏในภาพ ??

2.1.2 ค้นหาวัตถุที่มีลักษณะใกล้เคียงตัวอักษร

ในขั้นตอนนี้เรานำผลลัพธ์จากขั้นตอนก่อนหน้ามาจำจัดวัตถุที่ไม่มีลักษณะคล้ายคลึงอักษร เริ่มจากจับกลุ่มแต่ละ Pixel ในผลลัพธ์ของขั้นตอนก่อนหน้า การจับกลุ่มทำได้โดยเบริชเทียบแต่ละ Pixel กับ Pixel เพื่อบันรองข้าง หาก Pixel สองตัวที่เบริชเทียบกันมีค่าของความกว้างเส้นไม่ต่างกันเกิน 3.0 เท่า ให้ถือว่า Pixel ทั้งสองเป็นล่าวของวัตถุเดียวกันซึ่งจะถูกจัดกลุ่มเข้าด้วยกัน เมื่อการจัดกลุ่ม Pixel เสร็จสิ้น ผลลัพธ์ที่ได้คือภาพวัตถุต่าง ๆ ที่ปรากฏในภาพหรือ Connected Components



รูปที่ 2.1 ขั้นตอนการทำงานของ Stroke Width Transform

อย่างไรก็ได้วัตถุที่เกิดจากการรวมกลุ่มของ Pixel นั้นใหญ่หรือเล็กเกินไปก็จะถูกคัดออก การคัดวัตถุลักษณะดังกล่าวออกไปโดยการใช้กฎสองข้อดังนี้ (i) อัตราส่วนระหว่างเส้นผ่าศูนย์กลางต่อ มัชฌานของความกว้างของเส้นตัวอักษรนั้นต้องน้อยกว่า 10 (ii) ความสูงต้องมากกว่า 10 และน้อยกว่า 300 ตามที่แสดงในสมการ ??

$$f(d, h, \tilde{s}) = \begin{cases} 1, & \text{if } \frac{d}{\tilde{s}} < 10 \text{ and } 10 < h < 300 \\ 0, & \text{otherwise} \end{cases}, \quad (2.1)$$

โดย d คือ เส้นผ่าศูนย์กลางของวัตถุ, h คือ ความสูงของวัตถุ, และ \tilde{s} คือ มัชฌานของความกว้างเส้นตัวอักษร โดยค่าความกว้างได้มาจากค่าของ Pixel ในพื้นที่ของเส้นที่ถูกระบุไปโดยขั้นตอน SWT

2.1.3 จัดกลุ่มตัวอักษร

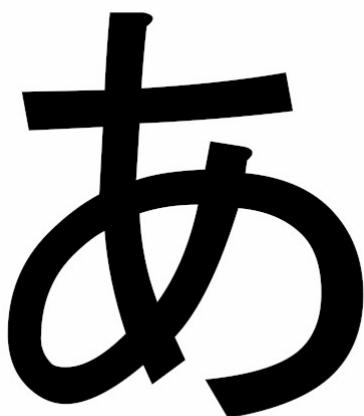
วัตถุแต่ละชิ้นที่หลักน้ำหนักคล้ายคลึงกับอักษรซึ่งผ่านการคัดกรองด้วยกฎจากขั้นตอนก่อนหน้าจะถูกนำมาจับกลุ่มเป็นบรรทัดของข้อมูลความในขั้นตอนนี้โดยใช้การเปรียบเทียบความคล้ายคลึงระหว่างลักษณะอักษรต่าง ดังนี้ ระยะห่างระหว่างวัตถุ, อัตราส่วนความกว้างของเส้นอักษร, และความสูงของอักษร โดยสองวัตถุจะถูกจัดกลุ่มกันต่อเมื่อ (i) อัตราส่วนระหว่างค่ามัชฌานความกว้างเส้นของวัตถุทั้งสองมีค่าน้อยกว่า 2 เท่า (ii) ความสูงของอักษรทั้งสองต่างกันไม่เกิน 2 เท่า (iii) ระยะห่างระหว่าง

สองวัตถุนั้นมีค่าไม่เกิน 3 เท่าของวัตถุที่กว้างที่สุดในคู่อักษรที่ใช้เปรียบเทียบ หลังจากการจัดกลุ่มนี้
 100 เราจะได้ใช้ของอักษรที่ถูกจัดกลุ่มเข้าด้วยกัน แต่ละใช้ประกอบไปด้วยอักษรส่องตัวที่ถูกจัดกลุ่ม ต่อ
 مانนั้นแต่ละใช้จะถูกรวบเข้าด้วยกันหากใช้อักษรนี้อักษรในใช้ของตนร่วมกันใช้อื่น ๆ และทิศทาง
 ของใช้มีความใกล้เคียงกัน

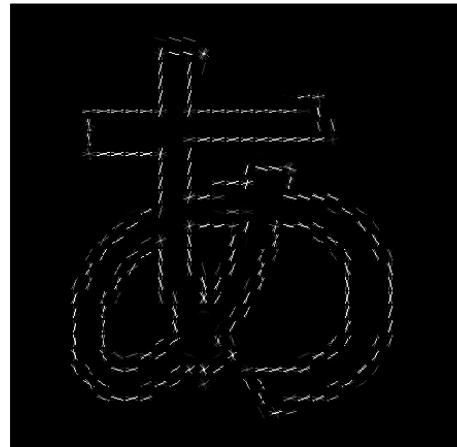
สุดท้ายขั้นตอนนี้จะบลลงเมื่อไม่มีใช้อักษรใด ๆ ถูกเชื่อมต่อเพิ่มเติม ในที่สุดเราจะได้กลุ่มหรือ
 ใช้ของอักษรที่เกิดจากการจัดกลุ่มด้วยความคล้ายคลึงของอักษรและทิศทางของข้อความ อีกนัยนึง
 ก็คือเราได้กลุ่มบรรทัดของแต่ละประโยคออกมากจากภาพถ่ายเรียบร้อยในขั้นตอนนี้

2.2 Histogram of Oriented Gradients

Histogram of Oriented Gradients (HOG) เป็นการสกัด Feature ของภาพโดยอาศัยรูปแบบ Histogram ของทิศทางเนคส์ในภาพ หรือ Gradient direction เพื่อพิจารณาลักษณะของวัตถุต่าง ๆ อย่างที่แสดงในภาพ ?? โดยภาพ ?? คือ ตัวอย่างอักษรภาษาญี่ปุ่น และภาพ ?? คือภาพแสดงทิศทางของเนคส์ของภาพอักษร โดยใช้เส้นขนาดเล็กแสดงทิศทางของเนคส์ ด้วยความสามารถนี้ จึงมีการนำ HOG มาใช้สำหรับสกัดลักษณะเด่นเพื่อใช้ในงานจำพวกการตรวจจับวัตถุอย่างหลากหลาย ทั้ง การตรวจจับท่าทางของมือ [?], การตรวจจับรถบนท้องถนน [?], การตรวจจับมนุษย์ในภาพ [?] และไม่เพียงแค่สามารถใช้กับงานตรวจจับวัตถุ แต่ยังสามารถใช้กับงานด้านตรวจหาข้อความในภาพได้เช่นกัน [?, ?, ?]



(ก)

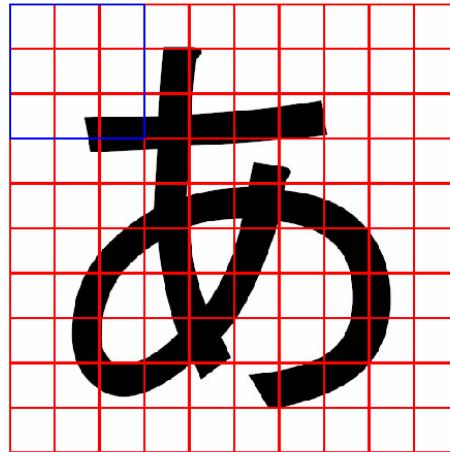


(ว)

รูปที่ 2.2 ตัวอย่างข้อมูลนำเข้าและการจำลองภาพทิศทางของ Histogram of Oriented Gradients

การสกัดลักษณะเด่นของ HOG นั้นทำได้โดยเริ่มจากแบ่งภาพเป็นส่วนเล็ก เรียกว่า Cell หรือช่องสี่เหลี่ยมตามที่แสดงในภาพ ?? จากนั้นสร้าง Histogram สำหรับ Cell นั้น ๆ ด้วยค่า Gradient Direction และ Magnitude โดย Histogram นี้จะเป็นตัวแทนของลักษณะของรูปร่างที่อยู่ภายใน Cell นั้น ๆ จากนั้นจะทำการ Normalization กับ Histogram ของแต่ละ Cell ด้วยกลุ่มของ Cell หรือที่เรียกว่า Block อย่างที่เห็นเป็นช่องสี่เหลี่ยมในภาพ ?? สุดท้ายเราจะได้ Histogram ของ Gradient Direction

จากทุก ๆ Cell ของภาพซึ่งเป็นตัวแทนของรูปร่างวัตถุแต่ละส่วน ด้วย Histogram ที่ได้มาระบุกนำไปเข้ากระบวนการ Vectorization เพื่อให้สามารถใช้ในงานอื่น ๆ ได้ต่อไป



รูปที่ 2.3 Cell และ Block ในการทำงานของ Histogram of Oriented Gradients

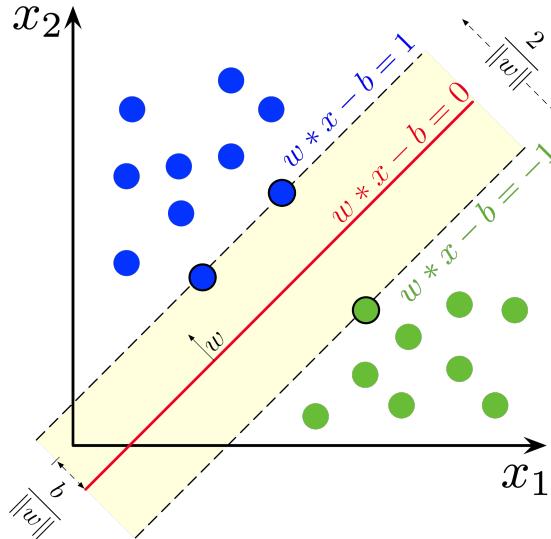
อย่างไรก็ต้องการที่จะได้ HOG ของวัตถุที่เราต้องการตรวจสอบจำเป็นต้องใช้ภาพของวัตถุนั้น ๆ ในกระบวนการ แต่ในสถานการณ์จริงภาพของวัตถุอาจอยู่ในภาพถ่ายขนาดใหญ่ที่ประกอบไปด้วยหลายวัตถุ เราจึงต้องตัดภาพของวัตถุเป็นส่วนย่อยเพื่อใช้คำนวณกับ HOG เราเรียกภาพส่วนย่อยที่ถูกตัดออกมานี้ว่า Patch ดังนั้นเราจำเป็นต้องใช้ Patch ที่มีขนาดใกล้เคียงกับวัตถุนั้น ๆ เป็นเหตุให้หากภาพมีขนาดใหญ่แต่วัตถุที่ต้องการตรวจสอบมีขนาดเล็ก จำนวน Patch ก็จะมากขึ้น

ในการนิยามมังงะ ตัวอักษรมักมีขนาดเล็ก ($20\text{px} - 40\text{px}$ โดยส่วนใหญ่อ้างอิงจาก Dataset ของเรา) เมื่อเทียบกับขนาดภาพมังงะ (1170px อ้างอิงจาก dataset ของเรา) ซึ่งมีขนาดใหญ่กว่าหลายเท่า ดังนั้นจำนวนของ Patch ที่ต้องสร้างและคำนวนด้วย HOG จึงมีมหาศาลและสร้างภาระแก่การคำนวน ลักษณะเด่นด้วย SVM อย่างมาก ซึ่งปัญหาส่วนนี้ส่งผลกระทบต่อความเร็วในการทำงานของเรา เราจึงใช้ SWT สำหรับกำหนดพื้นที่ที่คาดว่าเป็นอักษรเพื่อใช้สร้าง Patch แทนการสร้าง Patch จากทุกส่วนของภาพด้วยการใช้ Sliced Window

2.3 Support Vector Machine

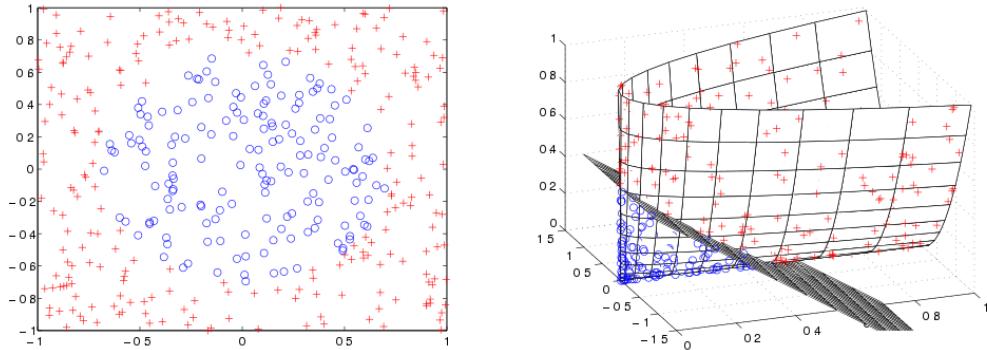
Support Vector Machine (SVM) [?] เป็นเทคนิค Pattern Recognition แบบ Supervised Learning ซึ่งถูกใช้ทั้งในงานเพื่อ Classification และ Regression ซึ่งภายในงานนี้ได้ใช้งานเพื่อ Classification โดยทำงานด้วยการสร้าง Hyper-plane ที่เหมาะสมที่สุด (Optimal) เพื่อจำแนกแยกข้อมูลสองกลุ่มอย่างที่แสดงในภาพ ??

เพื่อที่จะแยกข้อมูลทั้งสองกลุ่มด้วย Optimal Hyper-plane นั้น $w \times x - b = 0$ จะทำหน้าที่แบ่งข้อมูลสองกลุ่มออกจากกันโดยมี Support Vector ทำหน้าที่เป็นกันชนระหว่างข้อมูลที่ใกล้กันที่สุดระหว่างกลุ่มข้อมูลทั้งสอง ซึ่ง SVM นั้นจะสร้างพื้นที่การตัดสินใจขึ้นมา หรือก็คือพื้นระหว่าง $w \times x - b = 1$ และ $w \times x - b = -1$ โดยจะปรับให้ระยะห่างหรือความกว้างระหว่างทั้งสอง



รูปที่ 2.4 การแบ่งแยกกลุ่มข้อมูลด้วย Hyper-plane ของ SVM

นั้นมีค่าสูงสุด โดยจะห่างนั้นมีค่าเท่ากับ $\frac{2}{\|w\|}$ อย่างไรก็ได้หลาย ๆ ครั้งข้อมูลไม่สามารถแบ่งแยกได้ด้วยเส้นตรง และจำเป็นต้องใช้การแบ่งข้อมูลแบบ Non-linear ซึ่งสำหรับ SVM แล้วนั้นสามารถใช้ Kernel เข้ามาช่วยในการเปลี่ยนมิติของข้อมูลเพื่อให้สามารถแบ่งแยกข้อมูลทั้งสองกลุ่มได้ด้วย Linear Hyper-plan ตามที่แสดงในภาพ ??



รูปที่ 2.5 คุณสมบัติการเปลี่ยนมิติของข้อมูลด้วย Kernel

บทที่ 3

วิธีการทดลอง

ในบทนี้เรามีวิธีการใหม่ของเราระบุที่ได้ปรับปรุงและพัฒนาขึ้นมาโดยใช้ SWT เป็นส่วนหลัก ในของระบบใหม่ โดยมีจุดประสงค์เพื่อทำให้สามารถใช้งานร่วมกับภาพมังงะได้อย่างมีประสิทธิภาพ 150 และดำเนินการทดลองเพื่อวัดประสิทธิภาพของวิธีการใหม่ของเราว่าสามารถทำงานได้ดีขึ้นหรือไม่ อย่างไรเมื่อเปรียบเทียบกับวิธีการต้นฉบับ [?] และวิธีการอื่น ๆ ที่ถูกพัฒนามาก่อนหน้า

3.1 วิธีการใหม่ที่ถูกปรับปรุงและพัฒนาเพิ่มเติม

สำหรับวิธีการอย่างที่ได้กล่าวไว้ในบทที่ ?? วิธีการใหม่ของเรายังคงใช้ประโยชน์จาก SWT ร่วมกับความสามารถของ SVM โดยใช้ HOG เป็นลักษณะเด่น หรือ Feature อย่างไรก็ตามที่ได้กล่าวไปในบทที่ ?? จุดประสงค์หลักของ SWT ที่ถูกนำเสนอไปในงานวิจัยก่อนหน้านี้นั้นถูกออกแบบเพื่อการตรวจหาข้อความบนภาพถ่ายเป็นภาษาไทยหลัก ด้วยเหตุผลนี้ทำให้การทำงานร่วมกับภาพมังงะไม่สามารถทำงานได้ดีอย่างที่ควร ก่อให้เกิด False Positive จำนวนมาก ตามที่แสดงให้เห็นในภาพ ?? ซึ่งจำนวน False Positive ที่มากนั้นแสดงถึงประสิทธิภาพที่ต่ำของระบบ สาเหตุหลักคือความแตกต่างเชิงเอกลักษณ์ของวัตถุในภาพจริงและภาพมังงะ นอกเหนือนี้ยังมีปัจจัยอื่นๆ ที่สำคัญ เช่น การตรวจจับตัวอักษรในภาพมังงะ ซึ่งมีลักษณะที่แตกต่างจากตัวอักษรในภาพมาก เนื่องจากตัวอักษรในภาพมังงะมีลักษณะที่กว้างและมีเส้นผ่านศูนย์กลางที่ใหญ่กว่าตัวอักษรในภาพจริง ทำให้ระบบตรวจจับตัวอักษรในภาพมังงะได้ยากขึ้น ด้วยเหตุนี้วิธีการของเรายังคงปรับปรุงขั้นตอนของระบบดังเดิม โดยเราได้ปรับปรุงขั้นตอนการค้นหาวัตถุที่คุ้นเคย เช่น การจับคู่ตัวอักษร และการจัดเรียงตัวอักษร ให้สามารถทำงานได้ดีขึ้น รวมถึงเพิ่มความสามารถในการจัดเรียงตัวอักษรในภาพมังงะให้สามารถใช้งานได้ดีขึ้น

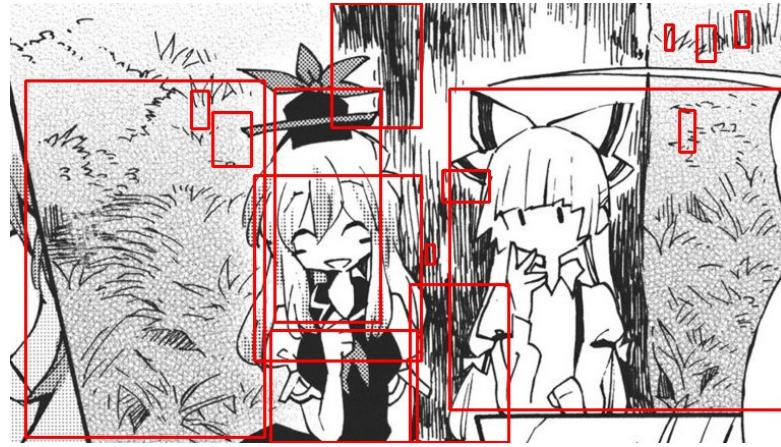
ความแตกต่างของวิธีต้นฉบับและวิธีการใหม่ของเรานั้นถูกแสดงให้เห็นในภาพ ?? อย่างที่เห็นในภาพ ?? เราได้เพิ่มขั้นตอนการคัดแยกตัวอักษรเข้ามา โดยวิธีการใหม่จะคัดแยกอักษรออกจากวัตถุอื่น ๆ ที่ไม่มีความเกี่ยวข้องกันที่จะจับคู่ตัวอักษรเข้าเป็นประกาย ขั้นตอนการคัดแยกนี้ใช้ความสามารถของ SVM Classification เพื่อช่วยลด False Positive ของผลลัพธ์การค้นหาวัตถุที่คุ้นเคยตัวอักษรจากขั้นตอนก่อนหน้า

3.1.1 The Stroke Width Transform

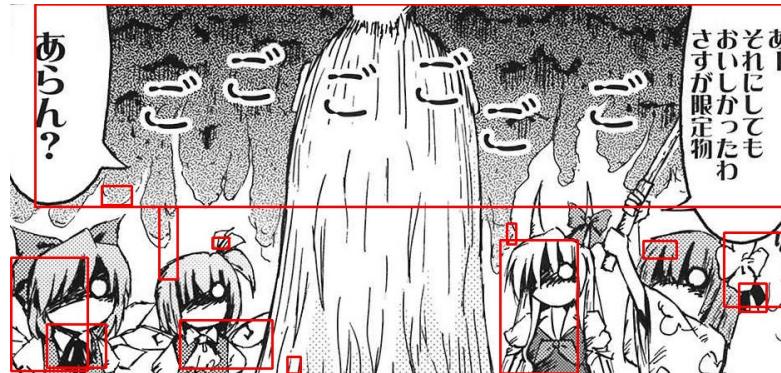
ในขั้นตอนนี้ใช้วิธีการเดียวกับงานวิจัย [?] ที่กล่าวไว้ในบทที่ ?? โดยเรานำ SWT มาดำเนินการบนภาพมังงะเพื่อให้อภูติแบบตัวดำเนินการ SWT โดยข้อมูล Output ของขั้นตอนนี้คือเมทริกซ์ขนาดเท่ากับภาพ Input ซึ่ง Output นี้จะถูกใช้ในขั้นตอนต่อไป

3.1.2 ค้นหาวัตถุที่ใกล้เคียงอักษร

ในมังงะนั้นข้อความหรืออักษรทั้งหลายมีขนาดที่หลากหลายและแตกต่างไปจากภาพถ่าย เราจึงต้องนำกฎเกณฑ์ที่ใช้ในการคัดกรองวัตถุกับตัวอักษรบนภาพถ่ายมาใช้ในการคัดแบ่งให้เหมาะสม



(n)



(v)

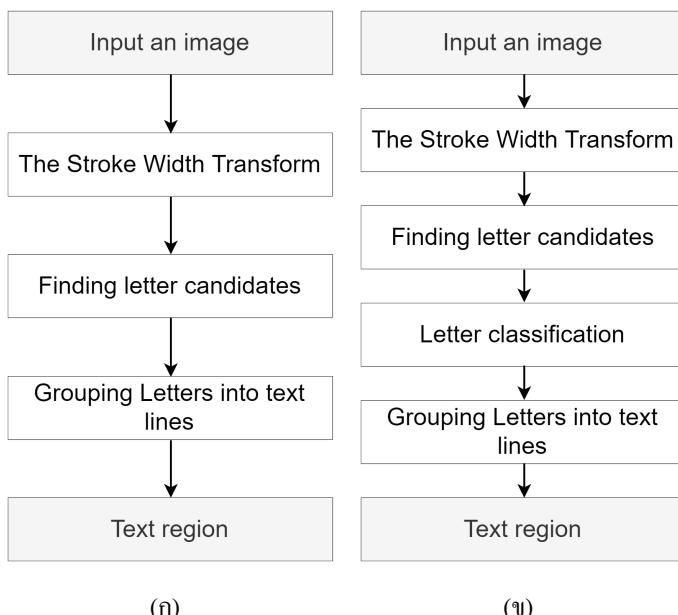
รูปที่ 3.1 ตัวอย่างผลลัพธ์จากการตรวจหาข้อความบนภาพมังงะ ด้วยวิธีต้นฉบับ [?] แสดงให้เห็น False Positive จำนวนมาก (n) นักวาด: Shinoasa (v) นักวาด: Kousei (Public Planet)

กับสภาพลักษณะเฉพาะของอักษรในมังงะ โดยกลุ่ดังกล่าวถูกดังแปลงให้อยู่ในรูปแบบสมการ ??

$$f(d, h, w, \tilde{s}) = \begin{cases} 1, & \text{if } 1 < \frac{d}{\tilde{s}} < 15 \text{ and } \tilde{s} \leq 80 \text{ and} \\ & 5 < h, w < 50 \\ 0, & \text{otherwise,} \end{cases} \quad (3.1)$$

โดยตัวแปรใหม่ที่ถูกเพิ่มเข้ามาคือ w ซึ่งคือ ค่าความกว้างของวัตถุนั้น ๆ

เมื่อเราจำจัดวัตถุที่ไม่มีลักษณะคล้ายคลึงอักษรออกไปแล้ว เราจะได้กลุ่มของวัตถุที่มีลักษณะคล้ายอักษร อย่างที่แสดงในภาพ ?? โดยในขั้นตอนนี้ของวิธีการ SWT ต้นฉบับไม่สามารถรวบรวมวัตถุที่คล้ายคลึงอักษรได้ครบถ้วนเพียงพอตามที่แสดงให้เห็นในภาพ ?? แต่ก็ใหม่ของเรานั้นที่ถูกปรับปรุงแล้วนั้นสามารถรวบรวมวัตถุที่คล้ายอักษรได้ครบถ้วนมากขึ้น อย่างไรก็ได้ถูกเกณฑ์ใหม่นั้นสร้าง False Positive ที่มากขึ้นตาม ซึ่งมากกว่าผลลัพธ์จากกลุ่มแรกที่ดังเดิมของ SWT ต้นฉบับ อย่างไรก็ได้ปัญหาดังกล่าวจะถูกแก้ไขด้วยขั้นตอนคัดแยกอักษรด้วย SVM ต่อไป ซึ่ง SVM จะทำหน้าที่กำจัด False Positive ออกໄไป



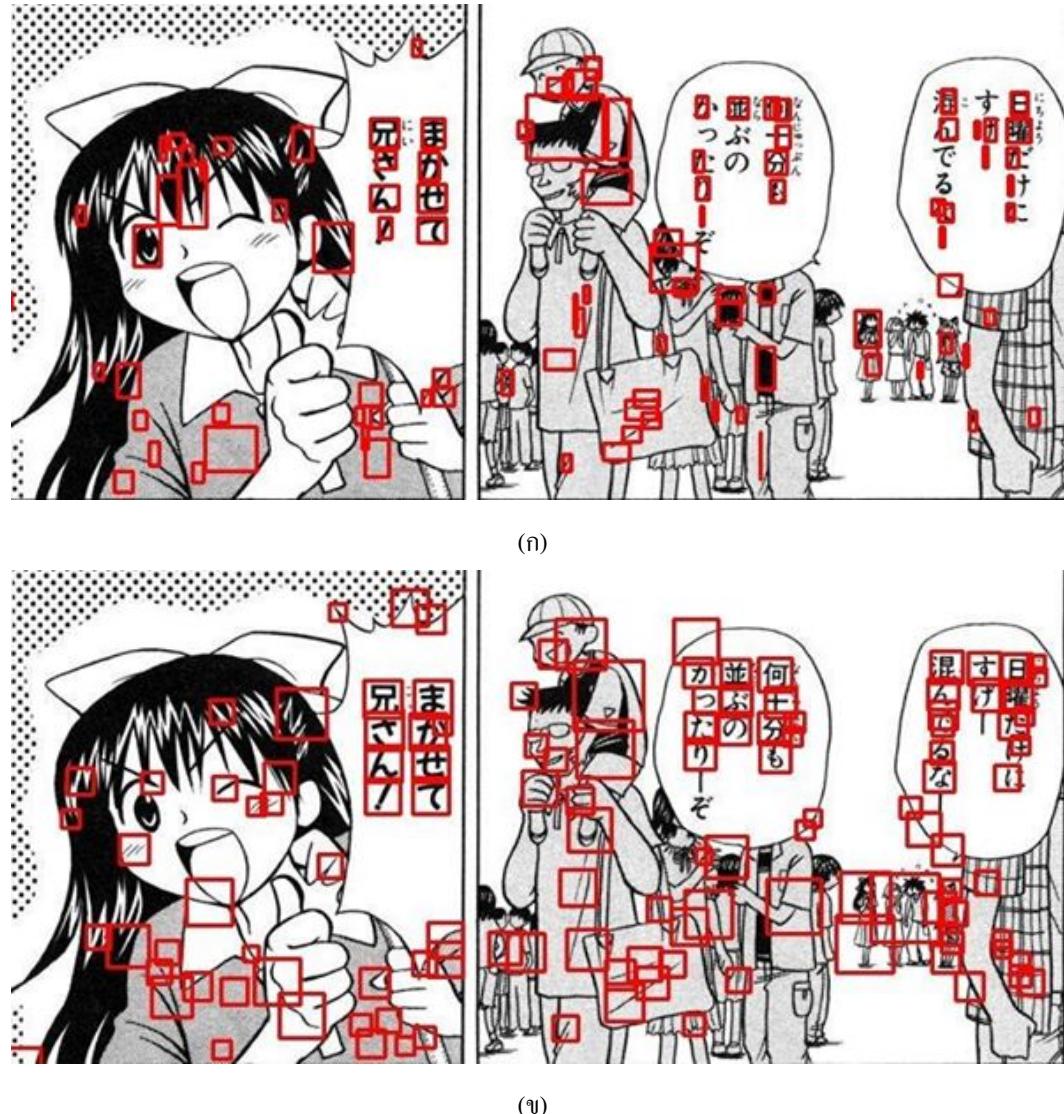
รูปที่ 3.2 แผนผังการทำงานของ (ก) วิธีการคั่งเดิม [?] และ (ข) วิธีการใหม่ของเรา

3.1.3 คัดแยกอักษรด้วย SVM

ในขั้นตอนนี้เราสร้างภาพนาดเล็ก (Patch) จากภาพมังงะ Input โดยพิ่งพาตำแหน่งและขนาดของวัตถุที่คล้ายคลึงอักษรจากขั้นตอนก่อนหน้าในการสร้างขอบเขตของภาพนาดเล็กนั้น ๆ โดยภาพนาดเล็กเหล่านี้จะถูกคัดแยกเป็นกลุ่มที่เป็นอักษรและกลุ่มที่ไม่ใช่ตัวอักษรด้วย SVM ซึ่งจะช่วยลด False Positive ให้ต่ำลงทำให้ได้ผลลัพธ์ของการตรวจหาข้อความที่แม่นยำมากขึ้น

เราได้นำ SVM มาดำเนินการในขั้นตอนนี้โดย SVM คือ เทคนิค Supervised Learning แบบหนึ่ง ซึ่งมักถูกใช้ในงานด้านคัดแยก (Classification) และ สมการลดด้อยต่อเนื่อง (Regression) [?] สำหรับ ชุดข้อมูลสำหรับเทรน โมเดลของ SVM ในขั้นตอนนี้จะถูกสร้างจากภาพขนาดเล็ก หรือ Patch โดย แบ่งออกเป็น ภาพที่เป็นอักษร (Positive) และ ภาพที่ไม่ใช้อักษร (Negative) อย่างที่แสดงในภาพ ?? ภาพขนาดเล็กสำหรับเทรนนิ่งและทดสอบเหล่านี้สร้างจาก Manga109 สำหรับภาพ Positive และ Negative ที่สร้างขึ้นมาจะถูกนำไปสกัดลักษณะเด่นหรือ Feature ด้วย Histogram of Oriented Gradients หรือ HOG [?] ซึ่งเป็นเทคนิคการสกัดข้อมูลเชิงรูป่างของวัตถุในภาพด้วยการพิ่งพารากรา ยด้วยของทิศทาง โภนสี โดยลักษณะเด่นที่ถูกสกัดมาสำหรับใช้งานในงานวิจัยนี้นั้นเป็นข้อมูลรูปแบบ เวกเตอร์ 2,916-dimension

200 เช่นเดียวกับข้อมูลสำหรับเทرن ภาพขนาดเล็กของวัตถุที่คล้ายคลึงอักษรจากขั้นตอนก่อนหน้าจะถูก HOG สถาคัลกิณณเด่นออกแบบแล้วจึงนำไปให้ SVM ดำเนินการคัดแยกภาพที่เป็นอักษรและไม่ใช่อักษรออกจากกัน หลังจากเสร็จสิ้นกระบวนการ การ ส่วนภาพที่เป็นอักษรจะถูกนำไปใช้ในการจัดกลุ่มอักษรให้เกิดเป็นข้อความในขั้นตอนต่อไป



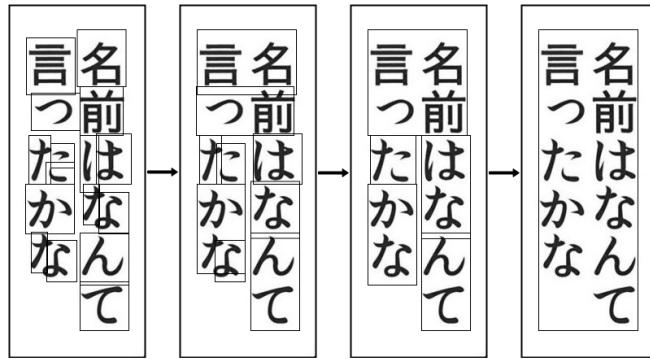
รูปที่ 3.3 ตัวอย่างแสดงการเปรียบเทียบผลลัพธ์ระหว่างขอบเขตตัวอักษรที่ตรวจพบระหว่างการใช้กฏเกณฑ์เดิมของ SWT ด้านบน (n) และกฏเกณฑ์ใหม่ในวิธีของเรา (u) ข้อมูลภาพถูกนำมาระบุจากเรื่อง Arisa ©Yagami Ken



รูปที่ 3.4 ตัวอย่างของ Patch: (g) ภาพ Positive Patches and (u) ภาพ Negative Patches

3.1.4 จัดกลุ่มอักษรเป็นข้อความ

ในวิธีการด้านบน [?] ขั้นตอนจัดกลุ่มวัดถูกที่คล้ายคลึงอักษรเข้าด้วยกันเป็นประโยชน์หรือบรรทัดของข้อความได้ใช้หลักการเปรียบเทียบความคล้ายคลึงลักษณะต่าง ๆ ของตัวอักษร ประกอบไปด้วย



รูปที่ 3.5 ตัวอย่างแสดงการจับกลุ่มของตัวอักษร

ของความสูง ขนาดเส้น ทิศทาง และ ระยะห่าง โดยวัตถุที่ไม่ถูกจับคู่จะถูกกำจัดทิ้งไป กระบวนการนี้ใช้สมมติฐานว่าประโภคหรือข้อความมักเกิดจาก การรวมตัวกันของอักษรมากกว่าหนึ่งตัวและจัดเรียงอยู่ในทิศทางเดียวกันกับตัวอักษรอื่น ๆ ที่ขนาดใกล้เคียงกันตามที่ได้กล่าวไว้ใน ?? ขั้นตอนนี้ช่วยกำจัดข้อมูลรบกวนอื่น ๆ เช่น วัตถุที่ไม่ใช่อักษรที่กระจายอยู่ในภาพ แต่อย่างไรก็ดี วิธีการของเรานี้ได้กำจัดข้อมูลรบกวนเหล่านี้ออกไปแล้วในขั้นตอนการคัดแยกอักษรด้วย SVM ดังนั้นเราวิจัยใช้เพียงระยะห่างระหว่างอักษรเป็นปัจจัยในการจับกลุ่มอักษร

วิธีการจัดกลุ่มอักษรของเรานี้จะใช้อักษรที่ถูกคัดแยกด้วย SVM และมาจัดกลุ่มเป็นประโภคโดยการจัดกลุ่มแต่ละอักษรที่อยู่ห่างกันไม่เกิน 1.5 เท่าของตัวอักษรที่คะแนนที่สุดของคู่อักษรที่ใช้เปรียบเทียบ หากตัวอักษรใดที่ห่างจากกันเกินกว่าค่าที่กำหนดจะถือว่าเป็นอักษรของคนละประโภค ซึ่งจะไม่ถูกจับกลุ่มเข้ามา โดยตัวอย่างในภาพ ?? แสดงถึงตัวอย่างขั้นตอนการจับกลุ่มด้วยระยะห่าง นอกจากนี้หลังจากการจัดกลุ่มแต่ละประโภคด้วยระยะห่างเสร็จสิ้นแล้ว กลุ่มอักษรเหล่านี้ถูกนำมาพิจารณาขนาดด้วยชั้นกัน โดยแต่ละกลุ่มของอักษรหรือแต่ละข้อความต้องมีพื้นที่ (ความกว้าง × ความสูง) เกินกว่า 2,550px อ้างอิงจากการทดลองกับชุดข้อมูลของเรา สุดท้ายเราจะได้กลุ่มของอักษรหรือประโภคข้อความจากภาพในมังงะออกมา

3.2 ชุดข้อมูลสำหรับการเทรนโมเดล SVM

เราได้นำภาพจากชุดข้อมูลภาพมังงะขนาดใหญ่ Manga109 [?] ประกอบไปด้วยภาพมังงะพร้อมข้อมูลประกอบ หรือ Annotation ของมังงะ 109 เรื่อง จัดทำโดยห้องทดลอง Aizawa Yamasaki แห่งมหาวิทยาลัยโตเกียว มังงะทั้งหมดในชุดข้อมูลนี้ถูกวัดโดยนักดามังงะมืออาชีพชาวญี่ปุ่นและถูกจัดจำแนยในช่วงปี 1970 ถึงปี 2010 แต่ละหน้าของมังงะถูกระบุตำแหน่งของข้อความในภาพซึ่งข้อมูลดังกล่าวเนี่ยหมายความว่าการใช้трен โมเดลและทดสอบวิธีการของเรามาก

3.3 การทดลอง

เราดำเนินการทดลองในรูปแบบเดียวกับงานวิจัยของ Aramaki et al. [?] ซึ่งจะทำให้เราสามารถเปรียบเทียบผลการทดลองประสิทธิภาพวิธีการของเรากับงานวิจัยอื่น ๆ ที่เกี่ยวข้องกับการตรวจหา

ข้อความในภาพมังงะซึ่งเคยถูกทดสอบมา ก่อนหน้าแล้วได้ เราได้เลือกภาพมังงะด้วยวิธีการสุ่มเลือก 100 หน้าสำหรับการเทรน และอีก 100 หน้าสำหรับทดสอบประสิทธิภาพ โดยภาพมังงะทั้งหมดนี้ถูกสุ่มเลือกจากมังงะ 6 เรื่อง ได้แก่ *Aosugiru Haru, Arisa 2, Bakuretsu Kung Fu Girl, Dollgun, Love Hina, และ Uchiha Akatsuki EvaLady*

เนื่องจากวิธีการของเราราใช้ SVM ซึ่งต้องสร้างโมเดลสำหรับใช้งานคัดแยกภาระห่วงภาพขนาดเด็ก (Patch) ระหว่างกลุ่มที่เป็นอักษรและไม่ใช่อักษรตามที่แสดงไปในภาพ ?? เราจึงได้สร้างชุดข้อมูลประกอบด้วยภาพอักษร 5,201 ภาพ และภาพขนาดเด็กอื่น ๆ ของวัตถุที่ไม่ใช้อักษรอีก 5,201 ภาพ กล่าวคือแบ่งเป็นข้อมูล Positive และ Negative ส่วนละ 50% เท่า ๆ กัน โดยภาพเหล่านี้สร้างจากภาพสำหรับการเทรนที่ได้ถูกกล่าวไปในย่อหน้าก่อนหน้า โดยใช้ขั้นตอนค้นหาวัตถุที่ใกล้เคียงอักษร ของ我们在การค้นหาตำแหน่งและขอบเขตของวัตถุที่คล้ายคลึงอักษรและนำตำแหน่งเหล่านั้นสร้างภาพขนาดเด็กเหล่านี้ออกมานะ

สำหรับ SVM เราใช้ Radial Basis Function Kernel โดย Hyperparameter ที่ร่วมใช้งานประกอบไปด้วย C และ γ เราใช้ Grid Search บนเครื่องคอมพิวเตอร์ Google Cloud Compute Engine *n1-highcpu-8* ในการค้นหาค่า Hyperparameter ที่ดีที่สุด ในช่วง 2^{-10} ถึง 2^{10} โดยได้ค่า C และ γ ที่ดีที่สุดที่ 2^5 และ $2^{-6.75}$ ตามลำดับ โมเดลที่ถูกปรับปรุงให้เหมาะสม (Optimized) และนี้จะถูกนำไปใช้ในขั้นตอนคัดแยกอักษรด้วย SVM

การประเมินวิธีการของเราราที่ถูกพัฒนาขึ้นใหม่นั้นได้ใช้รูปแบบการประเมินเดียวกับที่ใช้ใน IC-DAR 2013 Robust Reading Competition [?] โดยถ้าอัตราส่วนระหว่างพื้นที่ Overlapped ต่อ พื้นที่ Ground-truth นั้นมากกว่าค่า t_p และอัตราส่วนระหว่างพื้นที่ Overlapped ต่อ พื้นที่ Detected Region มากกว่า t_r ให้ถือว่า พื้นที่ขอบเขตอักษรที่ถูกตรวจพบจากวิธีการของเรานั้นถูกต้อง โดย t_p และ t_r นั้นมีค่าเท่ากับ 0.5 ถ้างบประมาณค่าดังกล่าวตามงานวิจัยของ Aramaki et al. [?] สำหรับ Precision และ Recall 250 เราได้คำนวณตามสมการดังต่อไปนี้ ??, ?? ตามลำดับ

$$P = \frac{\text{#Correctly Detected Rectangles}}{\text{#Detected Rectangles}} \quad (3.2)$$

$$R = \frac{\text{#Correctly Detected Rectangles}}{\text{#Rectangles of the Ground-truth}} \quad (3.3)$$

สำหรับ F-Measure เราคำนวนด้วยสมการดังนี้ ??

$$F = 2 \cdot \frac{P \cdot R}{P + R} \quad (3.4)$$

เราได้เปรียบเทียบผลการทดลองของวิธีการใหม่ของเรา rwm กับวิธีการต้นฉบับ [?] ซึ่งถูกใช้กับภาพถ่าย นอกเหนือจากนี้ยังเปรียบเทียบกับงานวิจัยการตรวจหาข้อความอื่น ๆ ด้วย ดังนี้ Basic Grouping+ImageNet Classification Model (BG+ImN) [?], Basic Grouping+Illustration2Vec Model (BG+I2V) [?], Scene Text Detection (STD) [?], Speech Balloon Detection (SBD) [?], และ Text Line Detection (TLD) [?] วิธีข้างต้นที่กล่าวถึงมีเทคนิคในการตรวจหาข้อความที่หลากหลายแตกต่างกัน เช่น การ

ขีดหลักสมมติฐานพื้นฐาน (ทิศทางของข้อความ, รูปแบบการจัดวาง, ลักษณะของกล่องคำพูด) และ Convolutional neural network

สำหรับ BG+ImN, BG+I2V, STD, SBD, และ TLD นั้นเราได้นำผลลัพธ์การทดลองจากงานวิจัยของ Aramaki et al. [?] มาใช้ในการเปรียบของเราระดับต่างๆ ซึ่งการเปรียบเทียบโดยตรงนี้สามารถทำได้เนื่องจากเราได้ดำเนินการทดลองในสภาพแวดล้อมเดียวกันงานวิจัยดังกล่าว โดยผลลัพธ์การเปรียบเทียบและตัวอย่างขอบเขตข้อความที่วิธีการของเรานั้นสามารถตรวจพบถูกแสดงให้เห็นที่ ??

บทที่ 4

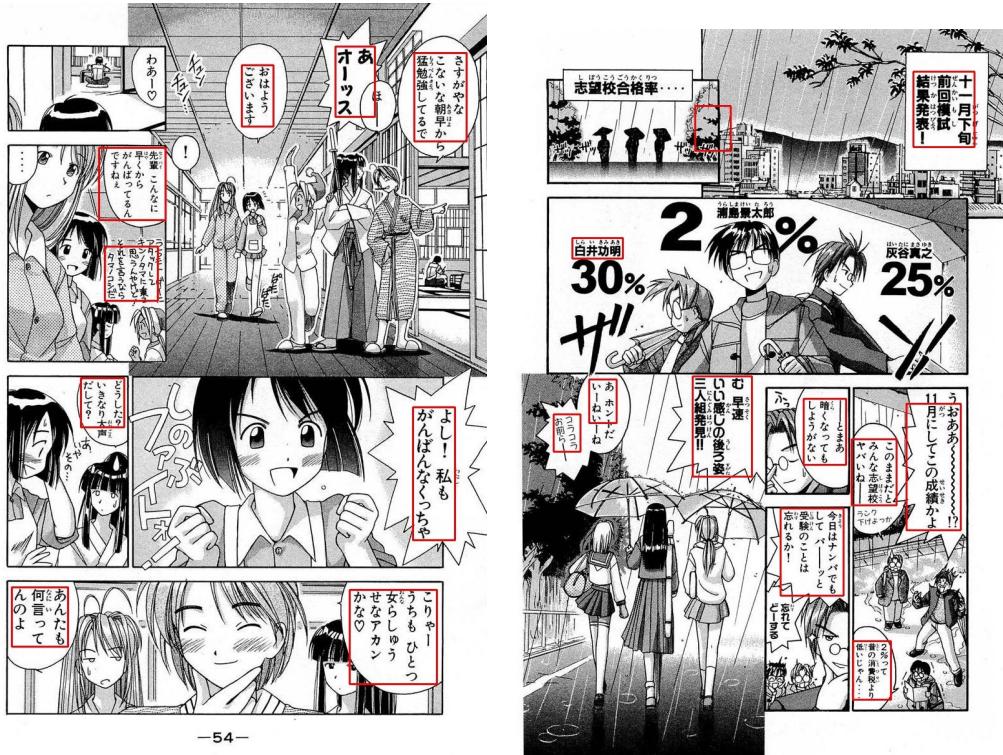
ผลการทดลอง

ผลลัพธ์การทดสอบประสิทธิภาพในวิธีการใหม่ของเราระเบริญเทียบกับงานวิจัยอื่น ๆ ที่เกี่ยวข้อง แสดงในตารางที่ ?? จากตารางดังกล่าว วิธีการของเราได้รับ F-measure สูงที่สุด ที่ 0.506 ซึ่งแสดงให้เห็นชัดเจนว่าวิธีการของเรา มีประสิทธิภาพดีกว่าวิธีการ SWT ต้นฉบับ [?] ยิ่งไปกว่านั้น วิธีการของเรา ยังได้รับ F-measure ที่มากกว่าวิธีการ BG+ImN [?] และ BG+I2V [?] ซึ่งทั้งสองวิธีนี้ ใช้เทคนิค Deep Learning เป็นส่วนหนึ่งในการตรวจหาข้อความในภาพ อย่างไรก็ได้ค่า Precision และ Recall สูงสุดของการทดลองนี้อยู่ที่ 0.715 และ 0.481 เป็นของ BG+I2V และ BG+ImN ตามลำดับ สำหรับตัวอย่างพื้นที่ของข้อความที่วิธีการใหม่ของเราตรวจพบถูกแสดงในภาพ ??

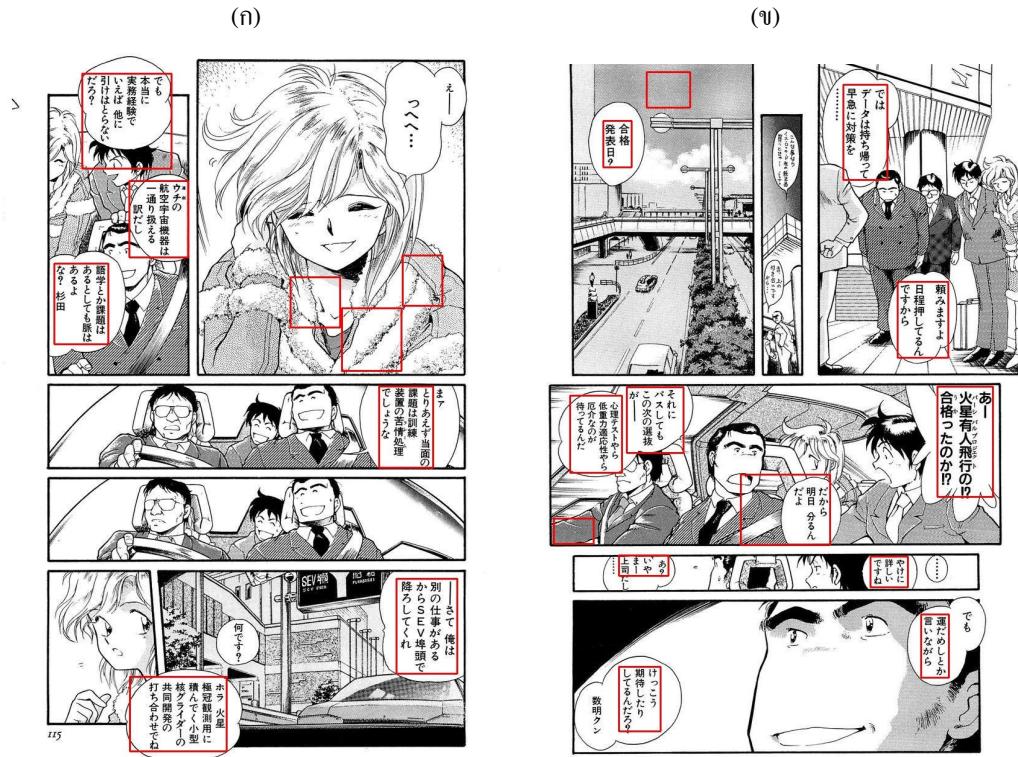
Method	Precision	Recall	F-measure
STD [?]	0.165	0.051	0.078
SBD [?]	0.180	0.102	0.130
TLD [?]	0.095	0.095	0.095
BG + ImN [?]	0.451	0.481	0.466
BG + I2V [?]	0.715	0.191	0.301
Baseline [?]	0.068	0.336	0.113
วิธีการของเรา	0.564	0.458	0.506

ตารางที่ 4.1 ตารางแสดงการเบริญเทียบประสิทธิภาพของวิธีการใหม่ของเราร่วมกับวิธีการอื่น ๆ ที่เกี่ยวข้อง

เป็นที่น่าสนใจอย่างมากที่วิธีการของเราสามารถทำงานได้ดีกว่าเทคนิค Deep Learning ทั้งสอง วิธี สมมติฐานแรกคือ BG+ImN นี้ ใช้ ImageNet Classification Model [?] ซึ่งถูกเทรนบนภาพถ่าย ของวัตถุจริง อย่างไรก็ได้ภาพวดมังงะของวัตถุต่าง ๆ นั้นมีความแตกต่างจากภาพวัตถุจริงอย่าง ชัดเจนซึ่ง ณ บุคนี้ทำให้วิธีการนี้ไม่สามารถทำงานได้เต็มประสิทธิภาพ อีกวิธีการที่ใช้ Deep Learning คือ BG+I2V ถึงแม้ว่าวิธีการนี้จะได้รับ Precision สูงที่สุดในการทดลองของเราแต่คะแนน Recall นั้นต่ำกว่าทั้ง BG+ImN และวิธีการของเรา วิธีการนี้ใช้โมเดล Illustration2Vec [?] เป็นโมเดล สำหรับคัดแยกข้อความจากวัตถุอื่น ๆ ที่ปรากฏในภาพมังงะ ล้วนของ โมเดลนี้ถูกเทรนบนภาพ วาด Anime (ภาพการ์ตูนแบบญี่ปุ่น) และภาพมังงะจากหลากหลายแหล่งแหล่งอันประกอบไปด้วย Danbooru และ Safebooru ซึ่งมีลักษณะงานคล้ายกับข้อมูลที่เราใช้ทดสอบในวิธีการของเรา แต่โมเดลนี้ ถูกออกแบบมาเพื่อการทำนายป้ายกำกับ (Tag Prediction) และค้นหาภาพที่คล้ายคลึงกัน ดังนั้น การเลือกใช้วิธีการที่ไม่ต้องลักษณะงานโดยตรงในลักษณะนี้จึงอาจเป็นเหตุผลว่าทำไม โมเดลนี้จึงไม่มี ประสิทธิภาพเท่าที่ควรในการทดลองนี้



-54-



115

รูปที่ 4.1 ตัวอย่างของเบตช์ความที่วันนี้การของเราราตรีพบ (ก-u) Love Hina ©Ken Akamatsu และ (ก-n) Eva Lady ©Miyone Shi

บทที่ 5

สรุปผล

ในการทดลองนี้ เราได้เสนอวิธีการตรวจหาข้อความบนภาพมังงะด้วยเทคนิค SWT ร่วมกับการใช้ SVM และลักษณะเด่น (Feature) HOG ในการลด False Positive ที่เกิดขึ้น การทดลองของเราดำเนินการบนชุดข้อมูล Manga109 ซึ่งเป็นชุดข้อมูลภาพมังงะที่มีกรอบ Annotation ที่เกี่ยวข้องมาเรียบร้อยแล้ว วิธีการของเรานั้นสามารถทำงานได้ผลลัพธ์ F-measure ที่ดีที่สุดในการเปรียบเทียบกับวิธี Baseline และวิธีการอื่น ๆ ที่เกี่ยวข้องรวมถึงวิธีการที่ใช้ Deep Learning เป็นส่วนประกอบในการคัดแยกอักษรหรือข้อความจากวัตถุอื่น ๆ ในภาพ ถึงแม้ว่างานของเราจะได้ทดสอบบนการ์ตูนญี่ปุ่นสามารถทำงานได้เป็นผลดีเยี่ยม แต่วิธีการของเรานั้นยังต้องมีการพัฒนาและค้นคว้าเพิ่มเติมเพื่อปรับปรุงประสิทธิภาพและทำให้สามารถใช้งานร่วมกับภาษาอื่น ๆ ได้

sdfsd

ภาคผนวก ก

การใช้ชีวิตในประเทศญี่ปุ่น

สำหรับการฝึกงานร่วมกับมหาวิทยาลัยออกไกโดในประเทศญี่ปุ่น ได้ทำการฝึกงานรวมกันเป็น 300 ระยะสี่เดือนกับอีกสิบวัน โดยได้เข้ามาทำงานในห้องทดลอง Intelligent Information System ภายใต้ การดูแลของอาจารย์ประจำห้องทดลอง Professor Dr. Masanori Sugimoto และในช่วงเวลาฝึกงาน นี้ยัง ได้รับความช่วยเหลือจากสมาชิกภายในห้องทดลองอีกท่าน Jiang Ye นักศึกษาปริญญาเอก ปี 2 จากประเทศจีน ได้ให้ความช่วยเหลือและการดูแลในเรื่องการดำเนินการเอกสารที่จำเป็นต่าง ๆ เช่น เอกสารการขยับเข้ามานักในประเทศญี่ปุ่นและการขยับออกจากประเทศญี่ปุ่น นอกจากนี้ยังได้ รับความช่วยเหลือในเรื่องความเป็นอยู่และสิ่งของทั่วไปในการใช้ชีวิตจากห้องกลุ่มนักศึกษาชาวไทย และชาวต่างชาติซึ่งล้วนศึกษาอยู่ที่มหาวิทยาลัยออกไกโด

การใช้ชีวิตในต่างแดน เช่น ประเทศญี่ปุ่นนั้นมีอุปสรรคหลายอย่าง ปัญหาประการหนึ่งที่ประสบ บ่อยครั้งที่สุด คือ เรื่องการสื่อสารภาษาญี่ปุ่น อย่างที่ทราบกันดีว่าชาวญี่ปุ่นส่วนใหญ่ไม่สามารถ สนทนากายาอังกฤษ ได้ ดังนั้นจึงมีความจำเป็นต้องเรียนรู้ภาษาญี่ปุ่นพื้นฐานเพื่อใช้ในการดำเนิน ชีวิตประจำวัน รวมถึงสมาชิกภายในห้องทดลองที่ส่วนใหญ่ไม่สามารถสื่อสารด้วยภาษาอังกฤษ เช่นกัน สมาชิกภายในห้องทดลองส่วนมากจะสามารถสื่อสาร ได้เพียงประโยชน์พื้นฐาน ไม่สามารถ สนทนาระหว่างกันได้ ดังนั้นเมื่อมีปัญหาในการพูดคุยจึงใช้การเขียนหรือพิมพ์ทดแทน อย่างไร ก็ได้ภาษาในห้องทดลองก็มีสมาชิกชาวญี่ปุ่นบางคนที่มีความสามารถภาษาอังกฤษในขั้นดีมาก อย่าง เช่นนักศึกษาปริญญาโทชาวญี่ปุ่นท่านหนึ่งที่เคยได้ฝึกงานในนิวซีแลนด์และฟิลิปปินส์มาก่อนหน้านี้ ทำให้สามารถสื่อสารภาษาอังกฤษได้เป็นอย่างดี

นอกจากตัวผมที่เป็นนักศึกษาฝึกงานจากไทย ภายในห้องทดลองที่ผมฝึกงานก็ยังมีนักศึกษาต่าง ชาติอีกสองคน คนแรกอย่างที่กล่าวไว้ในตอนต้น คือ นักศึกษาปริญญาเอกจากประเทศจีน และคนที่ สองคือนักศึกษาปริญญาโทจากบรัสเซล สำหรับชาวจีนนั้นใช้ภาษาญี่ปุ่นเป็นภาษาหลักและสามารถ สื่อสารกับคนญี่ปุ่นในห้องทดลองได้อย่างราบรื่น และยังพูดภาษาอังกฤษได้ดีอีกด้วย

ก.1 ที่อยู่อาศัย

หอพักที่ใช้อาศัยตลอดโครงการฝึกงานนี้ถูกจัดหาให้โดยทางมหาวิทยาลัยออกไกโด หอพักมีชื่อ ว่า International House Kita 8 East เป็นหอช雅ลั่วนจัดให้สำหรับนักศึกษาชาวต่างชาติเท่านั้น โดย ภายในห้องจะมีเพียงตู้, เตียง, โต๊ะ, คอมไฟ, ไฟเพคาน, ตู้เย็น, ฮีทเตอร์, และถังขยะ อย่างที่แสดงใน ภาพ ?? และ ?? สำหรับห้องน้ำ และห้องซักรีดต้องใช้ของส่วนกลางเท่านั้น โดยจะจัดแยกในแต่ละ ห้องสำหรับให้ผู้อยู่อาศัยใช้ร่วมกัน

สำหรับห้องครัวและห้องรับประทานอาหาร มีจัดเตรียมให้ที่ชั้นแรกของหอพัก โดยต้องใช้ร่วม กันห้องหอพัก เตาแก๊สและอ่างล้างจานถูกติดตั้งไว้เรียบร้อยสามารถใช้งานได้ในทันที สำหรับของใช้ ส่วนตัวรวมถึงเครื่องครัวผู้อยู่อาศัยต้องหาซื้อด้วยตนเอง ในกรณีของผมมีความจำเป็นซื้อแค่ชุดจาน

ชานชื่อนซึ่งมีห้องนอนและห้องครัว สำหรับอุปกรณ์ทำอาหารได้รับมาจากนักศึกษาชาวจีนที่กำลังจะกลับประเทศจีน ภาพห้องครัวแสดงในภาพ ?? และ ??



(ก) ห้องนอน



(ข) ห้องนอน



(ค) ห้องอาหาร



(ง) ห้องครัว

รูปที่ ก.1 ภาพหอพัก International House Kita 8 East

ภาคผนวก ๖

กิจกรรมระหว่างฝึกงาน

นอกจากการทำโครงการแล้วนั้น ระหว่างสี่เดือนนี้ผมยังได้ร่วมกิจกรรมด้านวิชาการต่าง ๆ ดัง เช่น การประชุมห้องทดลองรายสัปดาห์ และ การร่วมนำเสนอผลงานระหว่างห้องทดลอง เป็นต้น โดยรายละเอียดจะกล่าวต่อจากนี้

๔.๑ Mirai Symposium

กิจกรรมนี้จัดขึ้นที่ริบวกังหรือสอร์ทแบนญี่ปุ่น ซึ่งมีอ่อนเช่นหรือบ่อน้ำพุร้อนให้ได้แข็งด้วย ค่าใช้จ่ายทั้งหมดทางห้องทดลองออกให้ทั้งหมด ภายในงานจะมีกิจกรรมหลักคือการให้นักศึกษา ทุกคนของห้องทดลองต่าง ๆ ที่มาร่วมงานนำผลงานตนเองมาเสนอต่อไปโดยต่อในห้องประชุม หมุนเวียนไปเป็นช่วงเวลาคนละ 30 นาที หากสนใจงานของใครก็สามารถเดินไปสอบถามได้ อย่าง ที่เห็นในภาพ ?? การนำเสนอไม่ได้จำกัดว่างานต้องเป็นผลงานที่เสร็จลืนแล้ว อาจเป็นความก้าวหน้า ของงานที่พัฒนาอยู่ได้ เช่นเดียวกัน สำหรับครั้งนี้มีห้องทดลองร่วมงานสามห้องทดลอง สำหรับ นักศึกษาต่างชาติจะมีนักศึกษาท่านอื่นเข้ามาสอบถามเพียงเล็กน้อยเนื่องจากต้องสื่อสารด้วยภาษา อังกฤษและนักศึกษาญี่ปุ่นส่วนใหญ่ไม่สามารถพูดอังกฤษได้ สำหรับงานวิจัยที่ผมพัฒนาไว้ได้นำไป นำเสนอในงานนี้ด้วยเช่นเดียวกัน

เมื่อช่วงนำเสนอเสร็จสิ้น ทุกคนจะได้พักผ่อนตามอัธยาศัยโดยส่วนใหญ่จะเลือกไป เช่นบ่อน้ำพุ ร้อนกัน สำหรับบ่อน้ำพุร้อนก็มีทั้งภายนอกและภายในอาคาร จากนั้นเป็นมื้อเย็นซึ่งเป็นอาหารชุด ญี่ปุ่น ??

สำหรับวันที่สองก็มีกิจกรรมสร้างสรรค์ เป็นกิจกรรมที่ให้แยกกลุ่มกันและแก้โจทย์ปัญหาโดย ให้แก่ภายในเวลาที่กำหนด ตัวอย่างคำถาม เช่น มีเหรียญอยู่ร้อยเหรียญ มีทั้งที่หงายหน้าก้อยและหัว อย่างละครั้ง ต้องการแบ่งกลุ่มเหรียญสองกลุ่ม โดยที่แต่ละกลุ่มนี้จำนวนหน้าเหรียญและก้อยเท่า ๆ กัน โดยที่ผู้แบ่งมองไม่เห็นเหรียญจะทำได้อย่างไร เป็นต้น เมื่อเสร็จกิจกรรมสร้างสรรค์ก็ถือเป็นการ จบกิจกรรม Mirai Symposium แต่เพียงเท่านี้ และเดินทางกลับมหาวิทยาลัย

๔.๒ Lab Meeting

สำหรับห้องทดลองที่ได้เข้ามาฝึกงานนั้นมีกิจกรรม Lab Meeting ทุกวันพุธ คือ กิจกรรมที่ให้ สมาชิกภายในห้องทดลองร่วมกันนำงานวิจัยต่าง ๆ ที่ได้อ่านมานำเสนอต่อสมาชิกท่านอื่น ๆ ภายใน ห้องทดลอง โดยแต่ละสมาชิกจะถูกกำหนดวันเพื่อให้แต่ละคนที่ถูกกำหนดในวันนั้นนำ Conference Paper หรือ Journal ที่ถูกตีพิมพ์ในงานประชุมวิชาการ ซึ่งดังของ โลกมาร่วมกันนำเสนอแบบสรุป พร้อมสไลด์ กิจกรรมนี้ไม่มีคะแนนหรือรางวัลใด ๆ แต่เป็นการแลกเปลี่ยนความรู้ที่เกี่ยวข้องกับงาน ของตนเองที่ทำอยู่

สำหรับผมได้มีโอกาสนำเสนองานวิจัยที่ถูกตีพิมพ์ใน Journal เรื่อง *Text-aware balloon ex-*



(ก) บรรยาย公然ทางการนำเสนอผลงานด้วยโปสเดอร์



(ข) ชุดอาหารจัดเลี้ยงมื้อเย็น

(ค) ห้องอาหารจัดเลี้ยง

รูปที่ ๗.๑ ภาพกิจกรรมในงาน Mirai Symposium

traction from manga โดยหลังจากนำเสนอเรื่องสื้นก็ได้รับคำชมและความคิดเห็นจากสมาชิกและอาจารย์ของห้องทดลองอย่างหลากหลาย ซึ่งถือเป็นโอกาสที่ดีในการเรียนรู้มุมมองและความคิดเห็นที่เปลกใหม่ต่องานของเรา

๗.๓ กิจกรรมอื่น ๆ

นอกจากกิจกรรมเชิงวิชาการแล้ว ผู้เข้าร่วมในกิจกรรมสังสรรค์อื่น ๆ เช่น งานเลี้ยงตามโอกาสต่าง ๆ โดยผู้มีส่วนได้ร่วมงานเลี้ยงต้อนรับตัวผู้มา ซึ่งจัดในช่วงเดือนตุลาคมที่ผ่านมา สาเหตุที่จัดขึ้นมาจากเดือนที่เข้าฝึกงานวันแรกไปหลายเดือนเนื่องจากกำหนดการในตอนแรกนั้นคือหนึ่งเดือนหลังจากผู้มาอยู่ที่ประเทศไทยปัจจุบัน แต่ในช่วงกำหนดการเกิดเหตุการณ์แผ่นดินไหวใหญ่บนเกาะชอกไกโค ทำให้ต้องเลื่อนกำหนดการออกไป โดยงานเลี้ยงนี้จัดที่ร้านเนื้อย่างเจงกิสخ่าน เป็นเนื้อแกะย่างบน

เตาเหล็กลักษณะคล้ายหมุกระทะ ในประเทศไทยแตกต่างกันเพียงน้านไปที่เนื้อจะเป็นหลัก ภาพกิจกรรมแสดงในภาพ ??



(ก) สมาชิกในห้องทดลองระหว่างกินเลี้ยง



(ข) โต๊ะอาหารในร้านอาหารเงกิส่าวน



(ค) กระทะร้อนเงกิส่าวน

รูปที่ ๗.๒ ภาพระหว่างงานเลี้ยงต้อนรับ

นอกจากรงานเลี้ยงต้อนรับแล้วยังมีงานเลี้ยงต้อนรับนักศึกษาปีสามที่เข้าเป็นสมาชิกใหม่ในห้องทดลองนี้อีกด้วย ซึ่งงานนี้จัดรวมเป็นงานเลี้ยงสำลาตัวผู้พิรุณ ๆ กันเนื่องจากจัดให้ล้วนลืนสุดการฝึกงาน งานเลี้ยงจัดในห้องทดลองพร้อมด้วยอาหารหลากหลายชนิด งานไม่มีกิจกรรมอะไรมากมาย เป็นเพียงการกินเลี้ยงเพื่อทำความรู้จักกับสมาชิกใหม่ของห้องทดลองและนั่งเล่นเกมด้วยกันตามที่แสดงในภาพ ??

อธิบายเพิ่มเติมสำหรับงานเลี้ยงต้อนรับนักศึกษาปีสาม สำหรับห้องทดลองของคณะวิศวกรรมศาสตร์ที่ได้มาอยู่นี้จะมีการเปิดห้องทดลองให้นักศึกษาในคณะชั้นปีสามได้เข้าเยี่ยมชมห้องทดลองที่ตอนօงสนใจก่อนจะเลือกเข้ามาเป็นสมาชิกในห้องทดลองที่เกี่ยวข้องกับงานที่สนใจจะทำ โดยงานที่สนใจจะทำจะกลายเป็นชีวิ詹งานจากการศึกษาลักษณะในมหาวิทยาลัยของไทย โดยในช่วงเวลาหนึ่งปี



(ก) อาหารต่าง ๆ ในงานเลี้ยง



(ข) สมาชิกในห้องทดลองระหว่างเล่นเกมในงานเลี้ยง

รูปที่ ข.3 งานเลี้ยงอำลาและต้อนรับนักศึกษาปีสาม

แต่ละห้องทดลองก็จะมีการ โดยเฉพาะห้องทดลองของตัวเอง และเปิดโอกาสให้เข้าเยี่ยมชมงานภายในห้องทดลองว่าทำวิจัยเรื่องอะไร อายุเท่าไร ?? ที่เป็นป้ายเชิญเข้าชมห้องทดลองด้านเลี้ยงและมีการใช้ตัวละครจากการ์ตูนเรื่อง Kemono Friend ประกอบให้น่าสนใจและดึงดูดมากขึ้น



รูปที่ ๖.๔ ป้ายเชิญชวนชมห้องทดลอง โดยมีการใช้ตัวละครจากการ์ตูนประกอบให้น่าสนใจมากขึ้น

ภาคผนวก ค

ผลงานวิจัยที่ได้รับการตีพิมพ์

Detecting Text in Manga using Stroke Width Transform

Boonyarith Piriyothinkul*, Kitsuchart Pasupa[†]
 Faculty of Information Technology
 King Mongkut's Institute of Technology Ladkrabang
 Bangkok 10520
 Thailand
 Email: *58070077@kmitl.ac.th, [†]kitsuchart@it.kmitl.ac.th

Abstract—The Japanese comic-book style known as manga is becoming a popular topic for researchers. This paper focuses on the problem of detecting text regions in manga pages. Because it is time-consuming and laborious to identify the text regions in images manually, an automatic approach is highly desirable. Here, we propose a new text-detection method for manga using a Stroke Width Transform (SWT) technique in conjunction with a Support Vector Machine (SVM). Conventional SWT-based text-detection techniques perform poorly with manga because both text and non-text objects have similar characteristics for strokes, lines, and shapes. To better suit manga, we propose modifying the rules for finding letter candidates, which improves the ability to capture text. An SVM is then used to classify image patches into letter and nonletter regions. We compared our proposed framework with a conventional framework and other text-detection methods including deep-learning techniques. In the results, our proposed method achieved the highest F-measure of 0.506.

I. INTRODUCTION

Japanese comics are well known across the world as “manga.” There has been extensive research in manga [1]–[7] and benchmark datasets for this research, including Manga109 [8]. Manga109 contains more than 20,260 manga pages collected from 109 volumes and drawn by professional manga artists in Japan. The annotations in the dataset refer to face, body, frame, and text regions, with the text regions being identified and annotated manually. The annotations were undertaken by human without using any support from automation. This task was time-consuming and laborious. Therefore, an automatic annotation system is highly desirable.

In comparison to regular artworks, manga contains much more text and this research focuses on manga. Several text-detection methods for manga have been proposed [9], [10] but these methods are limited to specific layouts, dialogue-balloon positions, and art frames. Moreover, the proposed frameworks are not fully automated and robust to the various text styles present in manga. Recently, a deep-learning technique, the convolutional neural network, was applied to the feature-extraction aspect of text detection in manga [11]. It performed well and was able to reduce the false-positive rate without limitations on layout structure. However, its use of a deep-learning technique results in a high computational cost [11].

In this research, we focus on developing a framework for text detection that is robust to manga components. The

Masanori Sugimoto
 Graduate School of Information Science & Technology
 Hokkaido University
 Hokkaido 060-0808
 Japan
 Email: sugi@ist.hokudai.ac.jp

Stroke Width Transform (SWT) technique is used to describe the characteristics of a component “stroke.” It was proposed initially as part of a framework for detecting text in natural scenes, based on the assumptions that there would be high-density edges to text areas and smooth backgrounds [12]. However, employing the framework proposed by [12] to detect text in manga can generate many false-positive text predictions that reduce performance. This is because manga involves greyscale image and object details such as size, stroke, and background that are similar to text. We therefore aim to adopt an SWT similar to that proposed by [12] and improve its framework to suit manga.

The proposed technique can be divided into four main steps: (i) the Stroke Width Transform, (ii) finding letter candidates, (iii) letter classification, which aims to reduce false-positive results using the Support Vector Machine (SVM) with the histogram of oriented gradients (HOG) as a feature, and (iv) grouping letters into lines.

The paper is structured as follows. Detecting text in natural scenes with the SWT [12] is briefly explained in Section II. Our proposed framework is described in Section III. Section IV discusses our experiments and their results. Our conclusions are contained in Section V.

II. DETECTING TEXT IN NATURAL SCENES WITH THE SWT

A. The Stroke Width Transform

The SWT is a text-detection technique for photographs that uses the “stroke” features of an object [12], which can differentiate between text and nontext objects. The extraction process begins with the output image being initialized (by assigning ∞ to all its pixels). The output image is the same size as an input image. Next, the Canny edge detection [13] is employed to extract the edges objects in the image. The distance between the borders of a stroke can be calculated by considering each edge pixel p and finding a corresponding pixel q on the other edge. As shown in Fig. 1(b), the gradient direction of edge pixel p (d_p) points to edge pixel q . If d_p is approximately opposite d_q , $d_q = -d_p \pm \pi/6$ and any pixels that are located in between p and q will be assigned to $\|p - q\|$, as shown in Fig. 1(c). If the pixel already contains a width $\|p - q\|$ value and the new value of $\|p - q\|$ is smaller than the existing one, it will be replaced by the smaller value.

Otherwise, it remains the same. Eventually, a matrix with stroke-width values assigned to each element will be returned.

B. Finding letter candidates

In this step, the SWT output is used to find letter candidates by eliminating unrelated components. Each pixel is checked and compared with its neighbors. If their stroke-width ratio does not exceed 3.0, they will be grouped together. However, if the connected components are too big or too small, they will be eliminated according to these two rules: (i) the ratio of the diameter of the connected component to its median stroke width must be less than 10 and (ii) its height must be between 10 and 300 pixels, as shown in (1).

$$f(d, h, \tilde{s}) = \begin{cases} 1, & \text{if } \frac{d}{\tilde{s}} < 10 \text{ and } 10 < h < 300 \\ 0, & \text{otherwise} \end{cases}, \quad (1)$$

where d is the diameter of the connected component, h is its height in pixels, and \tilde{s} is its median stroke width.

C. Grouping letters into text lines

The letter candidates will be grouped into lines based on the distance between letters, the stroke-width value, and their height. Two letter candidates can be grouped together if (i) their median stroke-width ratio is less than 2.0, (ii) their height ratio does not exceed 2.0, and (iii) the distance between them does not exceed three times that of the wider candidate. After grouping, all letter chains can be merged to form a word. Chains can be merged if two chains have shared letters and the same direction. This step terminates when there are no more chains to be merged. Finally, the group or chain for a word is obtained from the image.

III. PROPOSED METHOD

In this section, we describe our proposed method, which utilizes an SWT in conjunction with an SVM and a HOG. As mentioned in Section I, the purpose of the SWT is to detect text in natural scenes. Unfortunately, when an SWT is simply applied to manga, it can lead to many false-positive results, as shown in Fig. 2. This is because of the characteristic differences between objects in manga and in the real world. Moreover, various graphic components in manga are very similar to text characters, such as grass, a character's hair, and background details, as shown in Figs. 2(a) and 2(b). Our proposal therefore modifies the steps involving finding letter candidates and grouping letters and adds a new step to make the approach more suited to manga, as shown in Fig. 3. In the last step, it utilizes an SVM to reduce the false positives in the previous finding-letter-candidates step.

A. The Stroke Width Transform

This step is similar to that proposed in [12]. The input is a manga image that has been transformed by the SWT operator. The size of the output image is the same as the input.

B. Finding letter candidates

In manga, there may be various text sizes to compare with the natural scenes. Therefore, we modify the conditions for finding letter candidates to enhance the performance, as shown in (2).

$$f(d, h, w, \tilde{s}) = \begin{cases} 1, & \text{if } 1 < \frac{d}{\tilde{s}} < 15 \text{ and } \tilde{s} \leq 80 \text{ and} \\ & 5 < h, w < 50 \\ 0, & \text{otherwise,} \end{cases} \quad (2)$$

where w is the width of the connected component in pixels.

Finally, we obtain the letter candidates shown in Fig. 4. Note that the baseline method fails to capture all the letters while our proposed method captures them all. However, there are many nonletter image patches. Their elimination is the purpose of the subsequent step.

C. Letter classification

In this step, we aim to classify image patches obtained from the previous step into letter and nonletter patches. This will reduce the number of false positives in the letter candidates. We adopt an SVM to perform this task. An SVM is a supervised learning technique that is well known and widely used for classification and regression tasks [14]. The dataset comprises two classes of letter (positive) and nonletter (negative) image patches created from Manga109 [8], as shown in Fig. 5. More details will be described in Section IV. Image-patch features are extracted by using the HOG [15], which is a descriptor for the gradient-direction patterns of objects in terms of their distribution. In this work, a HOG feature is represented by a 2,916-dimensional vector. After this process, the letter patches are grouped into lines in the next step.

D. Grouping letters into lines

The conventional (baseline) approach [12] groups the letter candidates into lines by using rules about height and median stroke-width values, as explained in Section II. It aims to remove scattered noise, i.e., the nonletter image patches. However, our approach has already dealt with nonletter image patches by using an SVM in conjunction with a HOG. Therefore, we use only the distance between letters to form groups.

Our grouping method uses the classified letters to create lines of letters. A line is created by including in the group any character that is closer to another than 1.5 times the narrower character's width. Characters spaced out less than this are considered as part of the same line, as shown in Fig. 6. Moreover, each line must contain at least three characters and have an area (width \times height) exceeding 2,550 pixels, according to the experiments using our dataset. Finally, the letters are grouped into lines.

IV. EXPERIMENTS

A. Dataset

We used images from Manga109 [8], which comprises 109 annotated volumes of manga created by the Aizawa Yamasaki Laboratory at the University of Tokyo. All manga volumes in the dataset were drawn by professional manga artists in

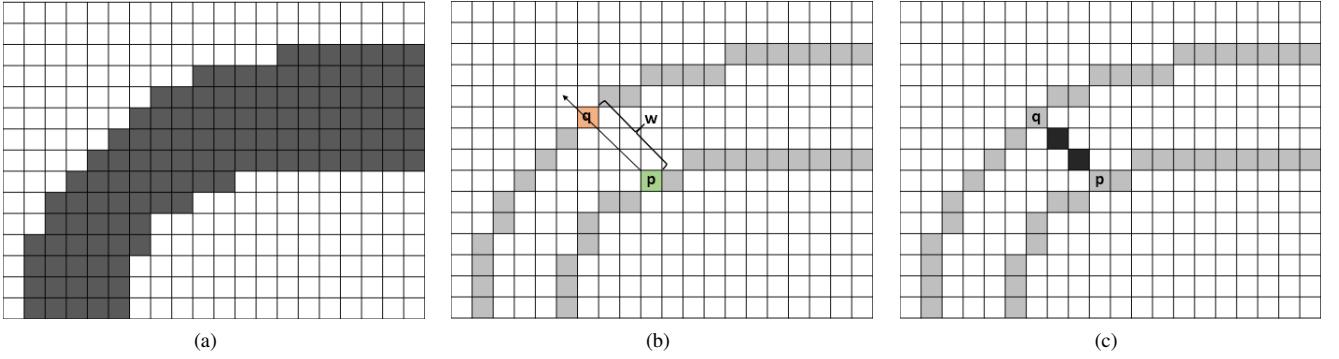


Fig. 1: (a) Grey pixels representing a stroke, (b) Pixel p is considered to be at the edge of the stroke, with the gradient direction d_p of p pointing to q , the corresponding pixel on the other edge. w is the distance between the two edge pixels. (c) The pixels along the line connecting p and q are assigned the distance w between p and q .

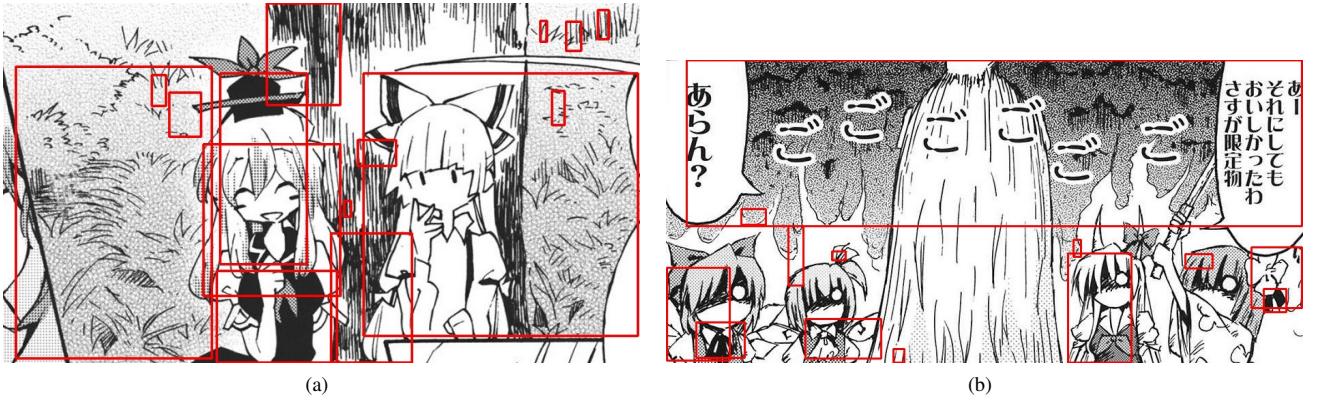


Fig. 2: Example of results generated by a conventional SWT method (the baseline method), where there are many false positives because the characteristics of text and graphic elements are very similar [12]. (a) Artist: Shinoasa (b) Artist: Kousei (Public Planet)

Japan and were available commercially to the public between the 1970s and 2010s. Each page is annotated with text-region areas suitable for training and evaluating our proposed method.

B. Experimental settings

We used experimental settings similar to those of Aramaki et al. [11], enabling us to make direct comparisons between their results and our results. We selected 100 pages of training data and 100 pages of test data at random from six manga titles, i.e., *Aosugiru Haru*, *Arisa 2*, *Bakuretsu Kung Fu Girl*, *Dollgun*, *Love Hina*, and *Uchiha Akatsuki EvaLady*.

Because our approach requires us to build a model for classifying image patches as letters or nonletters, as shown in Fig. 5, we created a dataset comprising 5,201 letter patches and 5,201 nonletter patches. These image patches were obtained from 100 pages selected at random from the Manga109 data set. The data were partitioned into 50 % for training and 50 % for validation. We used an SVM with a radial basis function kernel. This has two hyperparameters that require tuning, i.e., the regularization (C) and kernel (γ) parameters. A grid search was performed, with C and γ ranging from 2^{-10} to 2^{10} , to train the model and evaluate its F-measure on

the validation set. We performed the search using the Google Compute Engine n1-highcpu-8. The optimal C and γ values, returning the best F-measure, were 2^5 and $2^{-6.75}$, respectively. This optimized model was used in the letter-classification step for our proposed method.

Our evaluation method for text detection was the same as for the ICDAR 2013 Robust Reading Competition [16]. If the ratio of the overlapped area to the ground-truth area is greater than t_p and the ratio of the overlapped area to the detected region is greater than t_r , correct detection has been achieved. Here, t_p and t_r were set to 0.5, as for Aramaki et al. [11]. Precision and recall can be calculated by (3) and (4), respectively.

$$P = \frac{\text{\#Correctly Detected Rectangles}}{\text{\#Detected Rectangles}} \quad (3)$$

$$R = \frac{\text{\#Correctly Detected Rectangles}}{\text{\#Rectangles of the Ground-truth}} \quad (4)$$

The F-measure can then be calculated by

$$F = 2 \cdot \frac{P \cdot R}{P + R}. \quad (5)$$

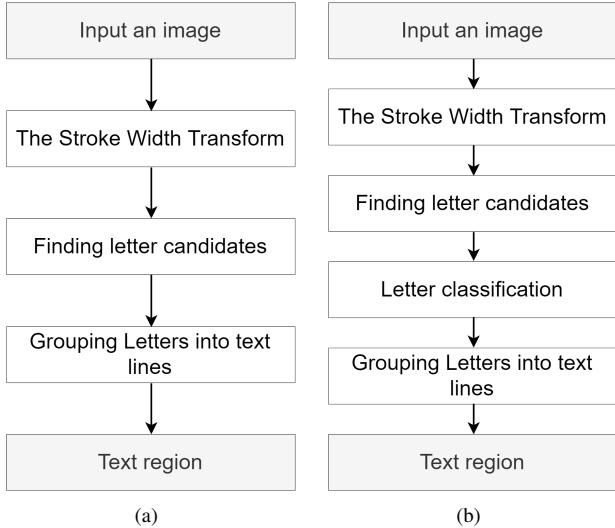


Fig. 3: The flowchart for the baseline method [12] comprises three steps whereas our proposed method has four steps: (a) baseline method and (b) proposed method.

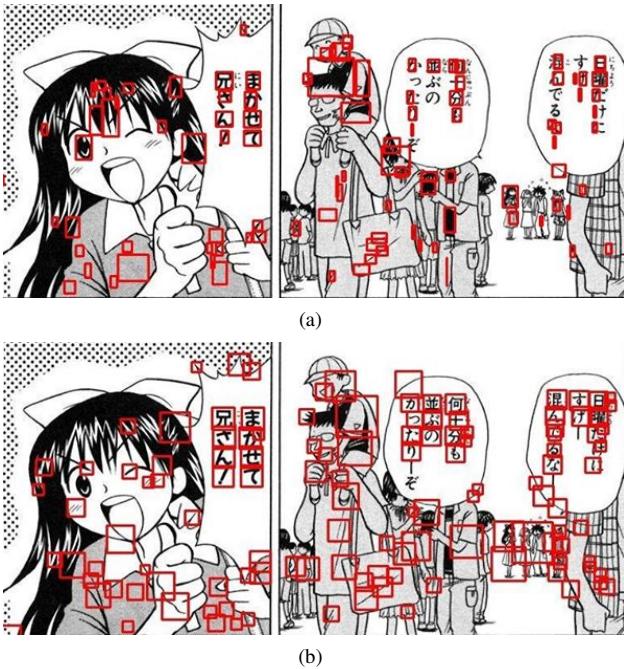


Fig. 4: The results of generating letter-candidate regions in Arisa ©Yagami Ken. The image patches with bounding boxes are the letter-candidate regions: (a) baseline method and (b) proposed method.

We compared our proposed method with a conventional (baseline) SWT method [14] and with previous published results for this dataset using various text-detection methods reported in literature. These other approaches were the Basic Grouping+ImageNet Classification model (BG+ImN) [11], the Basic Grouping+Illustration2Vec model (BG+I2V) [11],

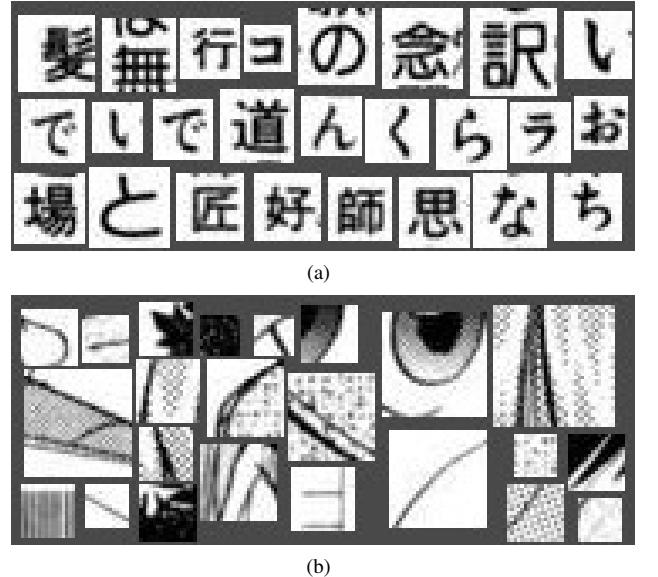


Fig. 5: Examples of letter image patches: (a) positive image patches and (b) negative image patches.

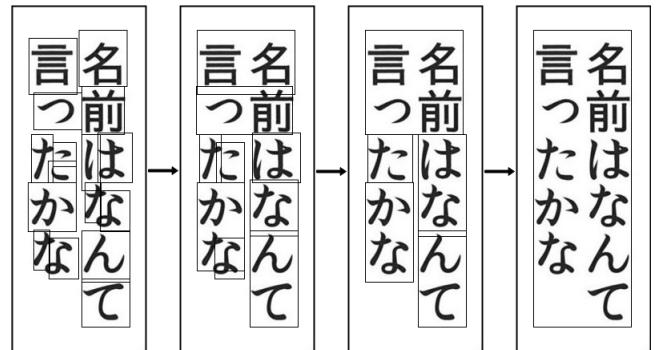


Fig. 6: An example of the process of grouping letters into lines.

Scene Text Detection (STD) [17], Speech Balloon Detection (SBD) [9], and Text Line Detection (TLD) [18]. These approaches use a variety of text-detection methods, including heuristic assumptions (direction of text, layout, dialogue balloon) and convolutional neural networks. The results are shown in Table I. Note that our proposed method yields the highest F-measure of 0.506. The proposed method clearly outperforms the baseline method for all performance measures. Moreover, its F-measure is higher than that for both BG+ImN and BG+I2V. Note also that both BG+ImN and BG+I2V utilize deep-learning methods. However, the best precision and recall is achieved by BG+I2V and BG+ImN at 0.715 and 0.481, respectively. Examples of text regions detected by our proposed technique are shown in Fig. 7.

It is interesting that our method can perform better than deep-learning techniques. This might be because BG+ImN uses the ImageNet Classification model [19], which was trained using a large number of real-world object images. However, manga objects are different from objects in natural

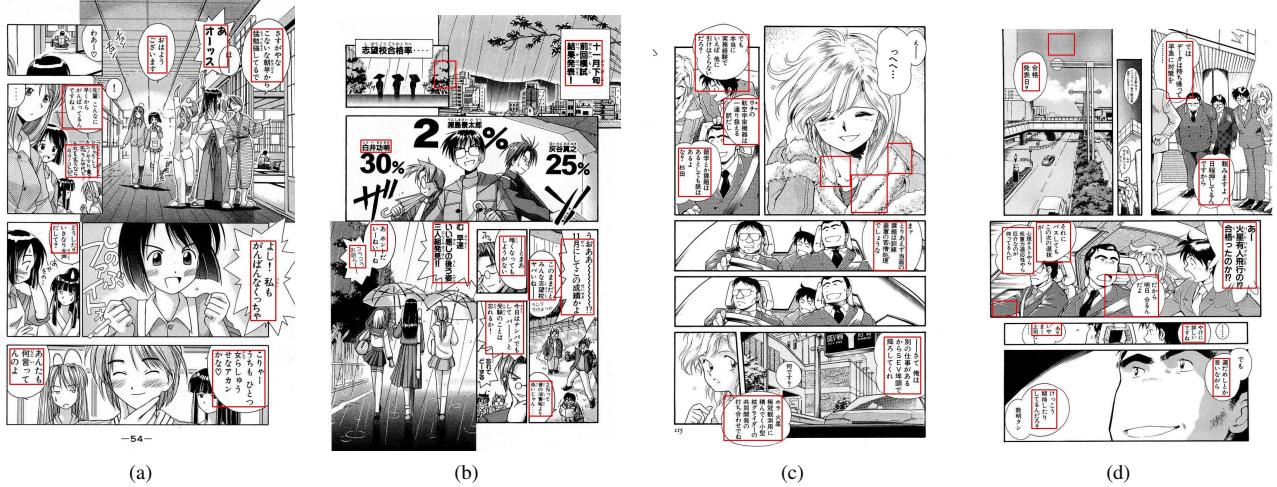


Fig. 7: Examples of text regions detected by our proposed method on (a–b) Love Hina © Ken Akamatsu and (c–d) Eva Lady © Miyone Shi.

TABLE I: The experimental results with Manga109 [8]. The proposed method is compared with the baseline method [12], the Basic Grouping+ImageNet Classification model (BG+ImN) [11], the Basic Grouping+Illustration2vec model (BG+I2V) [11], Scene Text Detection (STD) [17], Speech Balloon Detection (SBD) [9], and Text Line Detection (TLD) [18].

Method	Precision	Recall	F-measure
STD [17]	0.165	0.051	0.078
SBD [9]	0.180	0.102	0.130
TLD [18]	0.095	0.095	0.095
BG + ImN [11]	0.451	0.481	0.466
BG + I2V [11]	0.715	0.191	0.301
Baseline [12]	0.068	0.336	0.113
Our method	0.564	0.458	0.506

scenes. Another deep-learning model is BG+I2V. Although it achieved the best precision in our experiments, its recall is lower than both BG+ImN and our proposed technique. This method uses Illustration2Vec [20] as a classification model. It was trained using a dataset of manga and anime illustrations (the Japanese animation art) from several sources (e.g., Danbooru and Safebooru) that is similar to our approach. The model was not built specifically for text classification but for other purposes such as tag prediction and finding similar images. This might explain why the model appeared relatively unsuccessful in our experiments.

V. CONCLUSION

In this research, we have proposed a text-detection framework for manga using an SWT in conjunction with an SVM and a HOG. Experiments were conducted using a well-known Manga109 dataset. The proposed method yielded the best F-measure in comparison with a baseline method and those involving deep learning. To date, we have only tested our method on Japanese comics, where it works well. However, more investigations are required to find additional improvements and we aim to extend the method to other languages.

ACKNOWLEDGMENT

We would also like to show our gratitude to Xiaoyong Zhang, Sendai National College of Technology for sharing their wisdom. We also thanks to Napassorn Thammaviwatnukoon and ZUN for inspiration on this research. This research is supported by King Mongkuts Institute of Technology Ladkrabang.

REFERENCES

- [1] H. Yanagisawa, T. Yamashita, and H. Watanabe, “A study on object detection method from manga images using CNN,” in *Proceedings of the International Workshop on Advanced Image Technology (IWAIT 2018)*, Chiang Mai, Thailand., Jan 2018, pp. 1–4.
- [2] X. Liu, C. Li, H. Zhu, T.-T. Wong, and X. Xu, “Text-aware balloon extraction from manga,” *The Visual Computer*, vol. 32, no. 4, pp. 501–511, Apr 2016. [Online]. Available: <https://doi.org/10.1007/s00371-015-1084-0>
- [3] X. Pang, Y. Cao, R. W. Lau, and A. B. Chan, “A robust panel extraction method for manga,” in *Proceedings of the 22nd ACM International Conference on Multimedia (MM 2014)*. New York, NY, USA: ACM, 2014, pp. 1125–1128. [Online]. Available: <http://doi.acm.org/10.1145/2647868.2654990>
- [4] Y. Aramaki, Y. Matsui, T. Yamasaki, and K. Aizawa, “Interactive segmentation for manga using lossless thinning and coarse labeling,” in *Proceedings of the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA 2015)*, Hung Hom, Kowloon, Hong Kong, Dec 2015, pp. 293–296.
- [5] T. Ogawa, A. Otsubo, R. Narita, Y. Matsui, T. Yamasaki, and K. Aizawa, “Object detection for comics using manga109 annotations,” *CoRR*, vol. abs/1803.08670, 2018. [Online]. Available: <http://arxiv.org/abs/1803.08670>
- [6] S. Kovánen and K. Aizawa, “A layered method for determining manga text bubble reading order,” in *Proceedings of the IEEE International Conference on Image Processing (ICIP 2015)*, Quebec City, QC, Canada, Sept.
- [7] Y. Matsui, T. Shiratori, and K. Aizawa, “Drawfromdrawings: 2D drawing assistance via stroke interpolation with a sketch database,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 7, pp. 1852–1862, 2017.
- [8] Y. Matsui, K. Ito, Y. Aramaki, A. Fujimoto, T. Ogawa, T. Yamasaki, and K. Aizawa, “Sketch-based manga retrieval using manga109 dataset,” *Multimedia Tools and Applications*, vol. 76, no. 20, pp.

- 21 811–21 838, Oct 2017. [Online]. Available: <https://doi.org/10.1007/s11042-016-4020-z>
- [9] H. Tolle and K. Arai, “Manga content extraction method for automatic mobile comic content creation,” in *Proceedings of the International Conference on Advanced Computer Science and Information Systems (ICACSIS 2013)*, Bali, Indonesia, Sept 2013, pp. 321–328.
- [10] C. Rigaud, T. Le, J. . Burie, J. Ogier, S. Ishimaru, M. Iwata, and K. Kise, “Semi-automatic text and graphics extraction of manga using eye tracking information,” in *2016 12th IAPR Workshop on Document Analysis Systems (DAS)*, Santorini, Greece, April 2016, pp. 120–125.
- [11] Y. Aramaki, Y. Matsui, T. Yamasaki, and K. Aizawa, “Text detection in manga by combining connected-component-based and region-based classifications,” in *Proceedings of the IEEE International Conference on Image Processing (ICIP 2016)*, Phoenix, AZ, USA, Sept 2016.
- [12] B. Epshtain, E. Ofek, and Y. Wexler, “Detecting text in natural scenes with stroke width transform,” in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2010)*, San Francisco, CA, USA, Jun 2010, pp. 2963–2970.
- [13] J. Canny, “A computational approach to edge detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.
- [14] J. Suykens and J. Vandewalle, “Least squares support vector machine classifiers,” *Neural Processing Letters*, vol. 9, no. 3, pp. 293–300, Jun 1999. [Online]. Available: <https://doi.org/10.1023/A:1018628609742>
- [15] W. T. Freeman, W. T. Freeman, M. Roth, and M. Roth, “Orientation Histograms for Hand Gesture Recognition,” in *Proceedings of the International Workshop on Automatic Face and Gesture Recognition*, 1994, pp. 296–301. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.6.618>
- [16] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. G. i. Bigorda, S. R. Mestre, J. Mas, D. F. Mota, J. A. Almazn, and L. P. de las Heras, “Icdar 2013 robust reading competition,” in *2013 12th International Conference on Document Analysis and Recognition*, Kolkata, India, Aug 2013, pp. 1484–1493.
- [17] L. Gómez and D. Karatzas, “Multi-script text extraction from natural scenes,” in *Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR 2013)*, Washington, DC, USA, Aug 2013, pp. 467–471.
- [18] C. Rigaud, D. Karatzas, J. Van De Weijer, J.-C. Burie, and J.-M. Ogier, “Automatic text localisation in scanned comic books,” in *Proceedings of the 9th International Conference on Computer Vision Theory and Applications*, Barcelona, Spain, Feb 2013. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-00841492>
- [19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Proceedings of the 25th International Conference on Neural Information Processing Systems (NIPS 2012)*, vol. 1. Lake Tahoe, Nevada, USA: Curran Associates Inc., Dec 2012, pp. 1097–1105. [Online]. Available: <http://dl.acm.org/citation.cfm?id=2999134.2999257>
- [20] M. Saito and Y. Matsui, “Illustration2Vec: A semantic vector representation of illustrations,” in *SIGGRAPH Asia 2015 Technical Briefs*. Kobe, Japan: ACM, Nov 2015, pp. 5:1–5:4. [Online]. Available: <http://doi.acm.org/10.1145/2820903.2820907>