การรู้จำตัวอักษรลายมือเขียนโดยใช้ตัวจำแนกข้อมูลด้วยวิธีการคำนวณเพื่อนบ้านใกล้ที่สุด *k* ตำแหน่ง – วิจัยเชิงสำรวจ

Handwritten Character Recognition Using *k*-Nearest Neighbors Classifier – A Survey

โอพาริก สุรินต๊ะ[1*], กวีพจน์ บรรลือวงศ์[2]

Olarik Surinta[1*], Kaveepoj Bunluewong[2]

Received: 5 February 2016; Accepted: 23 May 2016

บทคัดย่อ

การวิจัยเชิงสำรวจฉบับนี้ ได้นำเสนอการนำวิธีการคำนวณเพื่อนบ้านใกล้ที่สุด k ตำแหน่ง (k-Nearest Neighbors: kNN) เพื่อใช้ในงานด้านการรู้จำตัวอักษรลายมือเขียน เนื่องจาก kNN เป็นขั้นตอนวิธีการเรียนรู้ของเครื่องจักรที่นิยมใช้สำหรับการจัดหมวดหมู่ของข้อมูล และยังเป็นขั้นตอนวิธีที่ไม่มีความซับซ้อน ขั้นตอนวิธี kNN ทำงานโดยนำข้อมูลมาเปรียบเทียบเพื่อหาค่าความใกล้เคียงระหว่างข้อมูลที่ต้องการจำแนก และข้อมูลทั้งหมดจากฐานข้อมูล ทั้งนี้ข้อมูลที่มีค่าระยะห่างน้อยที่สุดจำนวนทั้งสิ้น k ข้อมูลจะถูกนำมาพิจารณาคัดเลือกโดยใช้เทคนิคเสียงส่วนใหญ่ ดังนั้น ข้อมูลที่ต้องการจำแนกจะถูกกำหนดกลุ่มโดยพิจารณาจากกลุ่มของข้อมูลที่ปรากฏบ่อยที่สุด จากการสำรวจพบว่า ชุดข้อมูลที่ใช้ในงานวิจัยทางด้านตัวอักษรลายมือเขียนได้เก็บรวบรวมตัวอักษรลายมือเขียนที่มีลักษณะที่แตกต่างกัน เช่น โครงสร้างของตัวอักษรที่มีความคล้ายคลึงกัน การเขียนตัวอักษรที่มีความหลากหลาย ตัวอักษรที่เขียนไม่สมบูรณ์ หรือแม้กระทั่งข้อมูลมีจำนวนไม่เพียงพอ เป็นต้น ส่งผลต่อประสิทธิภาพของการรู้จำตัวอักษรลายมือเขียน ดังนั้นเมื่อนำคุณลักษณะพิเศษที่มีความคงทนต่อตัวอักษรที่เขียนในลักษณะที่แตกต่างกันมาใช้งานร่วมกับขั้นตอนวิธี kNN สามารถเพิ่มประสิทธิภาพการรู้จำตัวอักษรลายมือเขียน

คำสำคัญ: การรู้จำตัวอักษรลายมือเขียน ขั้นตอนวิธีการคำนวณเพื่อนบ้านใกล้ที่สุด *k* ตำแหน่ง ขั้นตอนวิธีที่ใช้ในการแยกประเภทข้อมูล คุณลักษณะพิเศษ การประมวลผลเบื้องต้น

Abstract

The use of *k*-Nearest Neighbors (*k*NN) algorithm for recognizing handwritten character scripts is presented in this survey paper. The kNN algorithm, the simple and well-known algorithm for machine learning, is suitable for classification scheme. According to the *k*NN algorithm, unknown data is firstly compared with the training samples to compute the similarity function. The most similar *k* neighbors, which have the smallest distance values, are subsequently selected by using the majority vote. The unknown data is then assigned by determining the number of its occurrences in each class. The most frequent class in *k* neighbors is finally selected as the output of the classifier. In this survey, it is found that the handwritten character datasets are collected from the different types of handwriting,

[1] *ผู้ช่วยศาสตราจารย์ ห้องปฏิบัติการมัลติเอเจนท์ ระบบอัจฉริยะ และการจำลองสถานการณ์ คณะวิทยาการสารสนเทศ มหาวิทยาลัยมหาสารคาม อำเภอกันทรวิชัย จังหวัดมหาสารคาม 44150 ประเทศไทย*

[2] *อาจารย์ หน่วยวิจัยการประมวลชั้นสูงสำหรับงานด้านปัญญาประดิษฐ์ การประมวลผลภาพและหุ่นยนต์ คณะวิทยาการสารสนเทศ มหาวิทยาลัยมหาสารคาม อำเภอกันทรวิชัย จังหวัด มหาสารคาม 44150 ประเทศไทย*

[1] *Assistant Professor, Multi-Agent Intelligent Simulation Laboratory (MISL), Faculty of Informatics, Mahasarakham University, Kantharawichai district, Maha Sarakham 44150, Thailand. E-mail: olarik.s@msu.ac.th*

[2] *Lecturer, (High) Processing Object for Learning Artificial intelligence, Image processing and Robot (POLAR), Faculty of Informatics, Mahasarakham University, Kantharawichai district, Maha Sarakham 44150, Thailand. E-mail: kaveepoj.b@msu.ac.th*

** Corresponding author; Olarik Surinta, olarik.s@msu.ac.th*

including different character sets with the similar structure, a variety of the handwriting styles, incomplete handwritten characters and insufficient number of handwritten character samples. These factors have a directl impact on the performance of the handwritten character recognition. Thus, the robust feature extraction techniques for handwritten recognition can be combined with the *k*NN algorithm in order to obtain the effective recognition performance.

Keywords: Handwritten Character Recognition, *k*-Nearest Neighbors Algorithm, Classifier, Feature Extraction, Pre-Processing

บทนำ

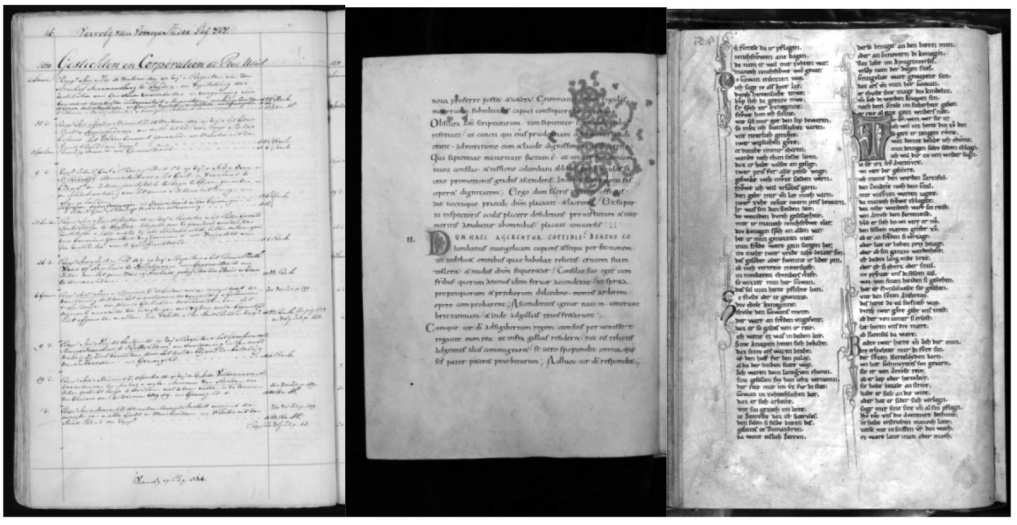
งานวิจัยด้านการรู้จำตัวอักษร (Character Recognition) นักวิจัยได้เริ่มค้นคว้าเมื่อประมาณปี ค.ศ. 1940 ซึ่งเป็นช่วงเวลาเดียวกับการพัฒนาเครื่องคอมพิวเตอร์แบบดิจิตอลซึ่งการรู้จำตัวอักษรเริ่มต้นพัฒนาเพื่อใช้สำหรับงานด้านประมวลผลของเครื่องคอมพิวเตอร์เพียงเท่านั้น ซึ่งต่อมานักวิจัยต่างให้ความสนใจศึกษา ค้นคว้า และพัฒนา เพื่อนำไปใช้ประโยชน์ในเชิงพาณิชย์ (1) โดยงานวิจัยด้านการรู้จำตัวอักษร เป็นกระบวนการในการประมวลผลของเครื่องคอมพิวเตอร์ ซึ่งทำงานควบคู่กับโปรแกรมคอมพิวเตอร์ที่พัฒนาขึ้นมาโดยเฉพาะและมีลักษณะการทำงานแบบอัตโนมัติ กระบวนการในการประมวลผลเริ่มตั้งแต่การสแกนเอกสารการแปลงรูปภาพของตัวอักษร จนกระทั่งการรู้จำตัวอักษร ทั้งนี้เพื่อบันทึกข้อมูลรูปภาพตัวอักษรให้อยู่ในรูปแบบของเอกสารอิเล็กทรอนิกส์ ทำให้ผู้ใช้งานสามารถที่จะนำข้อมูลตัวอักษรไปเปลี่ยนแปลงแก้ไขได้อย่างสะดวก และรวดเร็ว (2)

ในปัจจุบันบริษัทต่าง ๆ ได้นำงานวิจัยด้าน การรู้จำตัวอักษรไปพัฒนา และใช้งานในเชิงพาณิชย์กันอย่างแพร่หลาย ตัวอย่างเช่น ABBYY Fine Reader, Wondershare PDF element, Nuance Omni Page และ Readiris เป็นต้น อีกทั้งยังได้พัฒนาให้สามารถใช้งานผ่านสมาร์ทโฟนเพื่ออำนวยความสะดวกผู้ใช้ให้ใช้งานได้ง่ายขึ้น เช่น Atalasoft Mobile Image และ Scanner Pro เป็นต้น โดยโปรแกรมที่กล่าวมาข้างต้นเป็นโปรแกรมเชิงพาณิชย์ที่พัฒนาเพื่อใช้ในการรู้จำตัวอักษร ซึ่งสามารถนำไปประยุกต์ใช้ได้กับเอกสารหลากหลายประเภท และทำงานในลักษณะของการรู้จำตัวอักษรพิมพ์เพียงเท่านั้น ในส่วนของการรู้จำตัวอักษรภาษาไทย ในปี ค.ศ. 1996 ศูนย์เทคโนโลยีอิเล็กทรอนิสก์และคอมพิวเตอร์แห่งชาติ (NECTEC) และสำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ ได้ร่วมกันพัฒนาโปรแกรมอ่านไทย (Arn Thai) เพื่อสนับสนุนการทำงานในส่วนของตัวอักษรพิมพ์ภาษาไทย

งานวิจัยทางด้านการรู้จำตัวอักษรเป็นงานวิจัยที่ได้รับความนิยม ซึ่งต้องอาศัยความรู้ทางด้านการรู้จำรูปแบบ (Pattern Recognition) (3) ซึ่งเป็นส่วนย่อยของงานวิจัยทางด้านการเรียนรู้ของเครื่อง (Machine Learning: ML) โดยทั่วไปแล้วการรู้จำตัวอักษรสามารถแบ่งออกเป็น 2 ประเภทหลัก ได้แก่ การรู้จำตัวอักษรพิมพ์ (Optical Character Recognition: OCR) และการรู้จำตัวอักษรลายมือเขียน (Handwritten Character Recognition: HCR) (4) งานวิจัยทางด้านการรู้จำตัวอักษรนั้น ผู้วิจัยต้องวิเคราะห์ถึงโครงสร้างของเอกสาร (Document Layout Analysis) เนื่องจากเอกสารที่นำมาวิเคราะห์อาจมีหลากหลายประเภท เช่นหนังสือ หนังสือพิมพ์ เอกสารใบลาน และหลักศิลาจารึก เป็นต้น ผลลัพธ์จากการวิเคราะห์โครงสร้างก็คือพื้นที่ที่ให้ความสนใจ โดยผู้วิจัยต้องนำพื้นที่ส่วนนั้นมาพิจารณา เพื่อแบ่งพื้นที่ที่เป็นข้อความ (Textual Information) และพื้นที่ที่เป็นรูปภาพ (Nontextual Information) ออกจากกัน จากนั้นจึงแบ่งกลุ่มพื้นที่เหล่านั้นให้ตรงกับกลุ่ม ที่ได้กำหนดไว้ ทั้งนี้ขึ้นอยู่กับวัตถุประสงค์ของโปรแกรมที่พัฒนาขึ้นมา (5, 6) ตัวอย่างของเอกสารโบราณ แสดงใน (Figure 1) ตัวอย่างของเอกสาร Cabinet of the King (Kdk) ซึ่งเขียนขึ้นในปี ค.ศ. 1893 โดยเขียนเป็นภาษาดัชต์ (7) แสดงดัง (Figure 1a) ตัวอย่างเอกสารของ Saint Gallen ที่เขียนขึ้นในช่วงศตวรรษที่ 9 เขียนเป็นภาษาลาติน และเขียนโดย ผู้เขียนเพียงคนเดียว (8) แสดงดัง (Figure 1b) และ (Figure 1c) เป็นตัวอย่างเอกสารของ Parzival เขียนขึ้นในช่วงศตวรรษที่ 13 โดยเขียนเป็นภาษาเยอรมัน และเขียนเอกสารด้วยผู้เขียนจำนวน 3 คน ซึ่งเป็นลักษณะการเขียนในรูปแบบโกธิค (9)

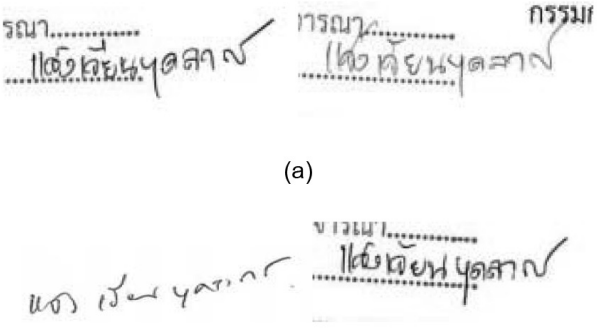
บทความฉบับนี้มุ่งเน้นศึกษางานวิจัยทางด้าน HCR เป็นหลัก เนื่องจากเป็นงานวิจัยที่มีความท้าทาย และมักพบปัญหาที่เกิดจากการเขียนตัวอักษรได้อยู่ตลอดเวลา ตัวอย่างเช่น ตัวอักษรที่เขียนติดกัน ตัวอักษรเลือน ตัวอักษรที่เขียนไม่สมบูรณ์ ตัวอักษรที่ซ้อนทับกันระหว่างบรรทัด และตัวอักษรที่เขียนลากหางยาว เป็นต้น ส่งผลให้นักวิจัยต้องคิดค้น และแก้ไขปัญหาที่เกิดขึ้นอยู่อย่างสม่ำเสมอ สิ่งสำคัญของงานวิจัยทางด้าน HCR นั้น ผู้วิจัยต้องวิเคราะห์ถึงลักษณะของตัวอักษรลายมือเขียนในลักษณะที่แตกต่างกัน ตั้งแต่ตัวอักษรที่เขียนโดยบุคคลเดียวกันหรือต่างบุคคล ตัวอักษรที่เขียนโดยบุคคลเดียวกันแต่ต่างกันในเรื่องของช่วงเวลา สภาวะอารมณ์ และ

สภาพแวดล้อม เป็นต้น ปัจจัยต่าง ๆ ที่กล่าวมาล้วนส่งผลต่อลักษณะของตัวอักษรที่เขียนขึ้นมาในแต่ละครั้ง (Figure 2a) ซึ่งเป็นผลทำให้ตัวอักษรที่เขียนนั้นผิดแผกไปจากลายมือเดิมของผู้เขียนนั้น ๆ โดยเฉพาะอย่างยิ่ง เมื่อนำตัวอักษรลายมือเขียนของบุคคลหนึ่งไปเปรียบเทียบกับบุคคลอื่น ทำให้เห็นถึงความแตกต่างของลายมือทั้งสองได้อย่างชัดเจน (Figure 2b) ทั้งนี้ผู้วิจัยยังต้องคำนึงถึงตัวอักษรที่มีลักษณะของโครงสร้างที่ใกล้เคียงกันอีกด้วย (Figure 3)



(a) (b) (c)

Figure 1 Some examples of the historical manuscripts: manuscript from (a) MLS dataset (Cabinet of the King, KdK 1893), (b) Saint Gall dataset (page 4, 9th century) and (c) Parzival dataset (page 124).



(a)

(b)

Figure 2 An illustration of Thai handwritten text which is written by (a) the same person and (b) two different persons.

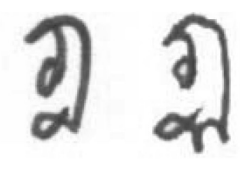


Figure 3 The similarities of structure between different character sets of Thai handwriting.

นอกเหนือจากนี้ งานวิจัยทางด้าน HCR สามารถนำไปพัฒนา และประยุกต์ใช้ในงานด้านธุรกิจได้อย่างกว้างขวาง เช่น การตรวจสอบข้อมูลที่ปรากฏอยู่บนเช็คธนาคาร การจัดเรียงและการคัดแยกไปรษณีย์โดยอัตโนมัติ เป็นต้น (10–14) อีกทั้งยังสามารถนำไปประยุกต์กับงานวิจัยหลากหลายประเภททั้งที่มีลักษณะที่เหมือน และแตกต่างกัน ได้แก่ การระบุตัวตนผู้เขียน (Writer Identification) การยืนยันตัวตนผู้เขียน (Writer Verification) และงานทางด้านนิติวิทยาศาสตร์ (Forensic Science) (15–18) เป็นต้น โดยทั่วไปแล้ว งานวิจัยด้าน HCR ต้องใช้ทฤษฎีที่เกี่ยวข้องกับการประมวลผลภาพ (Image Processing) และ ML ทำงานควบคู่กันเพื่อให้การรู้จำตัวอักษรลายมือเขียนมีความถูกต้องมากที่สุด

หลักการเบื้องต้นของการรู้จำตัวอักษร เริ่มต้นด้วยการหาคุณลักษณะพิเศษของตัวอักษรลายมือเขียน เพื่อนำข้อมูลเอกลักษณ์ที่ได้จากการคำนวณมาใช้เป็นตัวแทนของตัวอักษร จากนั้นจึงจำแนกประเภทของตัวอักษรให้ตรงกับกลุ่มที่ได้จัดเตรียมไว้ โดยใช้เทคนิคทางด้าน ML ทั้งนี้ขั้นตอนของ ML จำเป็นต้องใช้ข้อมูลจำนวนมากเพื่อใช้ในการประมวลผล (19)

ชุดข้อมูลที่ใช้ในงานวิจัยทางด้าน HCR

ในงานวิจัยทางด้าน HCR ผู้วิจัยอาจใช้ชุดข้อมูลมาตรฐานเพื่อทดสอบประสิทธิภาพของทฤษฎี และขั้นตอนวิธีที่ได้คิดค้น โดยชุดข้อมูลมาตรฐานที่สามารถนำมาใช้ทดสอบได้แก่ ชุดข้อมูล USPS (The U.S. Postal Service) ซึ่งเป็นชุดข้อมูลที่นิยมใช้ในยุคเริ่มต้นของงานวิจัยทางด้าน HCR ข้อมูลชุดนี้เป็นข้อมูลตัวเลขที่ถูกเก็บรวบรวมโดยไปรษณีย์ของประเทศสหรัฐอเมริกา ซึ่งเมื่อนำซองจดหมายมาผ่านการสแกน โปรแกรมจะหาตำแหน่งของรหัสไปรษณีย์ และตัด ตัวเลขออกมาจากซองจดหมาย และปรับเปลี่ยนขนาด ของตัวเลขให้มีขนาด 16x16 พิกเซล โดยจัดเก็บในรูปแบบของภาพสีเทา โดยประกอบด้วยข้อมูลชุดเรียนรู้มีจำนวนทั้งสิ้น 7,291 ตัวอักษร และจำนวน 2,007 ตัวอักษรสำหรับข้อมูลชุดทดสอบ (20)

ข้อมูลชุดมาตรฐานที่นิยมใช้กันอยู่ในปัจจุบัน คือ MNIST (Mixed National Institute of Standards and Technology) (21) ประกอบด้วยตัวเลข (0-9) ที่เป็นลายมือเขียน โดยตัวเลขมีขนาด 28x28 พิกเซล และจัดเก็บอยู่ในรูปแบบของภาพสีเทา ชุดข้อมูลประกอบไปด้วยข้อมูลชุดเรียนรู้ และชุดทดสอบ จำนวน 60,000 และ 10,000 ตัวอักษร ตามลำดับ

ในปี ค.ศ. 2009 ชุดข้อมูลสำหรับการรู้จำตัวอักษรภาษาลาตินถูกนำเสนอโดย Laurens van der Maaten (22) ได้รวบรวมตัวอักษรภาษาลาตินทั้งหมดมาจากชุดข้อมูล Firemaker ซึ่งต้นฉบับเป็นของ Schomaker and Vuurpijl (23) ข้อมูลชุดนี้ถูกนำไปใช้งานในด้านนิติวิทยาศาสตร์ โดยมีวัตถุประสงค์เพื่อการระบุตัวตนผู้เขียน ตัวอักษรลาตินที่ถูกเก็บรวบรวม แบ่งออกเป็นตัวอักษรลายมือเขียนตัวพิมพ์ใหญ่และตัวเลข แต่ในชุดข้อมูลตัวอักษรลาตินขาดตัวอักษร ‘X’ ทำให้ข้อมูลชุดนี้มีอักษรลาตินทั้งสิ้น 25 ตัวอักษร และตัวเลขจำนวน 10 ตัวเลข ข้อมูลตัวอักษรทั้งหมดถูกจัดเก็บในรูปแบบของภาพสีเทา และขนาดของตัวอักษรถูกเปลี่ยนเป็น 50x50 พิกเซล โดยข้อมูลถูกแบ่งออกเป็นข้อมูลชุดเรียนรู้ทั้งสิ้น 27,966 ตัวอักษร และข้อมูลชุดทดสอบจำนวน 12,167 ตัวอักษร

สำหรับข้อมูลภาษาไทย ยังไม่มีชุดข้อมูลมาตรฐานที่ใช้กันอย่างแพร่หลาย โดยในปี ค.ศ. 2003 หน่วยงาน NECTEC ได้นำเสนอชุดข้อมูลที่เรียกว่า NECTEC และในปี ค.ศ. 2006 ชุดข้อมูล Thai CAM ถูกนำเสนอโดย *Nopsuwan et al.* (24) ทั้งนี้เพื่อใช้สำหรับการรู้จำตัวอักษรลายมือเขียนภาษาไทย โดยทั้งสองชุดข้อมูลประกอบด้วย ตัวอักษรลายมือเขียนภาษาไทยจำนวน 77 ตัวอักษร ซึ่งประกอบด้วย ตัวอักษร สระ วรรณยุกต์ ตัวเลขไทย และสัญลักษณ์พิเศษ ตัวอักษรทั้งหมดถูกปรับเปลี่ยนให้มีขนาด 64x64 พิกเซล ซึ่งชุดข้อมูล NECTEC ประกอบด้วยข้อมูลชุดเรียนรู้จำนวน 9,702 และชุดทดสอบ 4,851 ตัวอักษร และชุดข้อมูล ThaiCAM ประกอบด้วย ข้อมูลชุดเรียนรู้จำนวน 4,620 ตัวอักษร และชุดทดสอบจำนวน 4,955 ตัวอักษร

ในปี ค.ศ. 2015 ชุดข้อมูลตัวอักษรลายมือเขียนภาษาไทย ALICE-THI ถูกนำเสนอโดย Surinta et al. (25) ในชุดข้อมูลประกอบด้วยตัวอักษร (THI-C68) (พยัญชนะ สระ วรรณยุกต์ และสัญลักษณ์) และตัวเลข (THI-D10) จำนวนทั้งสิ้น 68 และ10 ตัวอักษร ตามลำดับ ผู้วิจัยได้ตัดตัวอักษร และสัญลักษณ์บางตัวที่ไม่เป็นที่นิยมใช้งานออกไป ข้อมูลชุด ALICE-THI ถูกเปลี่ยนขนาดให้เป็น 36x36 พิกเซล โดย THI-C68 ประกอบไปด้วยข้อมูลชุดเรียนรู้จำนวน 13,130 ตัวอักษร และข้อมูลชุดทดสอบจำนวน 1,360 ตัวอักษร และ THI-D10 มีจำนวนของตัวอักษรในชุดเรียนรู้ และชุดทดสอบจำนวน 8,055 และ 1,500 ตามลำดับ ภาพรวมของชุดข้อมูลตัวอักษรลายมือเขียน แสดงดัง (Table 1)

จากการค้นคว้าพบว่ามีงานวิจัยที่ศึกษา และพัฒนางานทางด้าน HCR อยู่เป็นจำนวนมาก บทความนี้ศึกษาเฉพาะงานวิจัยทางด้าน HCR ที่ใช้ขั้นตอนวิธีการค้นหาสมาชิกที่ใกล้ที่สุดเพื่อเข้ามาช่วยในการจำแนกข้อมูลของตัวอักษรลายมือเขียนเท่านั้น

ขั้นตอนวิธีสำหรับการค้นหาสมาชิกที่ใกล้ที่สุด (***K*-**Nearest Neighbors (*k*NN) Algorithm)

การค้นหาสมาชิกที่ใกล้ที่สุด หรือเรียกโดยย่อว่า *k*NN เป็นเทคนิคหนึ่งทางด้าน ML ที่เป็นวิธีการแบบนอนพาราเมตริก (26) ดังนั้นการทดสอบจะไม่คำนึงถึงลักษณะการแจกแจงของกลุ่มข้อมูล และไม่ทดสอบเกี่ยวกับค่าพารามิเตอร์ของข้อมูลกลุ่มตัวอย่าง โดยวิธี *k*NN ไม่จำเป็นต้องสร้างแบบจำลองเพื่อนำมาใช้ทำนายค่าของข้อมูล แต่จะนำข้อมูลทั้งหมดมาคำนวณเปรียบเทียบกับข้อมูลที่ต้องการจำแนกเพื่อหาสมาชิกที่ใกล้ที่สุด โดย *k*NN เป็นเทคนิคที่นำไปใช้สำหรับการจัดหมวดหมู่ข้อมูลที่นิยมใช้งานกันอย่างแพร่หลายในงานวิจัยทางด้าน HCR และ CR

ข้อมูลที่ใช้สำหรับนำเข้า ใน *k*NN คือจำนวนข้อมูลชุดเรียนรู้ที่อยู่ในรูปแบบของข้อมูลที่เป็นคุณลักษณะพิเศษทั้งสิ้นจำนวน *k* ชุดข้อมูล โดยที่ผลลัพธ์ จะขึ้นอยู่กับจำนวน *k* ที่ถูกเลือก โดยทั่วไปแล้ว *k* จะถูกกำหนดให้เป็นจำนวนเลขคี่ เช่น 1, 3, 5, 7 และ 9 เป็นต้นซึ่งผลลัพธ์ที่ได้จากการจัดหมวดหมู่โดยขั้นตอนวิธี *k*NN นั้นเรียกว่า “สมาชิกของกลุ่ม” (c_i) โดยนำข้อมูลจากชุดเรียนรู้ (x_i) ทั้งหมดมาคำนวณกับข้อมูลที่ต้องการจำแนก (y) เพื่อคำนวณหาระยะห่าง (d_i) ระหว่าง x_i และ y จากนั้นเลือกข้อมูลที่มีค่า d_i ที่น้อยที่สุดจำนวนทั้งสิ้น *k* ชุด

การคำนวณหาค่าระยะห่างระหว่างชุดข้อมูลและข้อมูลที่ต้องการจำแนกที่เป็นที่นิยม (25,27) ก็คือการหาระยะห่างแบบยุคลิด (Euclidean Distance) สามารถคำนวณได้ดังสมการที่ (1)

$$d(x,y)=\sqrt{\sum_{i=1}^{N}(x_i-y_i)^2} \qquad (1)$$

โดยที่ N คือจำนวนมิติของข้อมูลที่เป็นคุณลักษณะพิเศษ ขณะที่ x และ y คือข้อมูลชุดเรียนรู้ และข้อมูลที่ต้องการจำแนก ตามลำดับ

ท้ายที่สุดแล้ว ข้อมูลที่ต้องการจำแนกจะถูกกำหนดให้อยู่ในกลุ่มที่เหมาะสม โดยวิธีที่เรียกว่า “เสียงส่วนใหญ่” (Majority Vote: MV) ซึ่งตรวจสอบจากข้อมูล x_i โดยที่ *i = 1, ..., k* ดังนั้น กลุ่มของข้อมูลที่ปรากฏบ่อยที่สุดจะถูกกำหนดให้เป็นผลลัพธ์ของขั้นตอนวิธี *k*NN (28) สามารถคำนวณได้จากสมการที่ (2)

$$y(d_i)=arg\,max_k \sum_{x_j \in kNN} S(d_i,x_j)\,y(x_j,c_k) \qquad (2)$$

โดยที่ d_i คือข้อมูลที่ต้องการจำแนกที่จะนำมาจัดกลุ่ม

x_j คือข้อมูลจากชุดเรียนรู้ที่ j

$S(d_i, x_j)$ คือ ฟังก์ชันที่ใช้หาค่าความใกล้เคียงของข้อมูล (Similarity Function) ระหว่าง d_i และ x_j ซึ่งวิธีการที่ใช้กันอย่างแพร่หลายก็คือการหาระยะห่างแบบยุคลิด ดังสมการที่ (1)

$y(x_j, c_k)$ เป็นสมาชิกของ {0, 1} มีค่าเป็น 0 เมื่อ x_j อยู่ในกลุ่มของ c_k และมีค่าเป็น 1 เมื่อx_j ไม่อยู่ในกลุ่มของ c_k ดังนั้น x_j จะถูกจัดให้อยู่ในกลุ่ม c_k ที่มีจำนวนมากที่สุด

ตัวอย่างการจำแนกกลุ่มของ kNN แสดงดัง Figure 4 จากตัวอย่างแสดงให้เห็นถึงข้อมูลชุดทดสอบจำนวนทั้งหมด 3 กลุ่ม ประกอบไปด้วยกลุ่ม a, b และ c ซึ่งผลลัพธ์ของการจัดแบ่งกลุ่มเมื่อ k มีค่าเป็น 1, 3 และ 5 สามารถทำได้ดังต่อไปนี้

กรณีที่ 1) เมื่อกำหนดให้ k=1 สังเกตุได้ว่า กลุ่มข้อมูล a ณ ตำแหน่ง d_1 มีค่าระยะห่างระหว่างข้อมูลที่ต้องการจำแนกน้อยที่สุด ดังนั้น ในกรณีนี้ ข้อมูลที่ต้องการจำแนกจะถูกจัดให้อยู่ในกลุ่ม a

Table 1 Overview of Handwritten Character Datasets.

Dataset	Language	Year	Color Format	Number of Writers	Number of Classes	Train Data	Test Data
USPS	digit	1990	Binary	Multi	10	7,291	2,007
MNIST	digit	1998	Grayscale	250	10	60,000	10,000
NECTEC	Thai and digit	2003	Binary	63	77	9,702	4,851
ThaiCAM	Thai and digit	2006	Binary	20	77	4,620	4,955
LATIN	Latin and digit	2009	Grayscale	251	35	27,966	12,167
ALICE-THI	Thai and digit	2015	Grayscale	150	78	21,185	2,860

กรณีที่ 2) กำหนดให้ k=3 ชุดข้อมูลที่มีระยะห่างถึงข้อมูลที่ต้องการจำแนกน้อยที่สุดจำนวน 3 อันดับ (d_1, d_2 และ d_3) ก็คือ ข้อมูลของกลุ่ม a จำนวน 1 ข้อมูล และ b จำนวน 2 ข้อมูล ดังนั้น เมื่อกำหนดให้ k=3 ข้อมูลที่ต้องการจำแนกจะถูกจัดให้อยู่ในกลุ่ม b

กรณีที่ 3) กำหนดให้ k=5 ชุดข้อมูลที่มีระยะห่างถึงข้อมูลที่ต้องการจำแนกน้อยที่สุดจำนวน 5 อันดับ (d_1, d_2, d_3, d_4 และ d_5) ซึ่งก็คือ ข้อมูลของกลุ่ม a จำนวน 3 ข้อมูล และ b จำนวน 2 ข้อมูล ดังนั้นในกรณี k=5 ข้อมูลที่ต้องการจำแนกจะถูกจัดให้อยู่ในกลุ่ม *a*

งานวิจัยที่เกี่ยวข้อง

อ้างอิงจากหลากหลายงานวิจัย สามารถกล่าวได้ว่า kNN เป็นขั้นตอนวิธีที่มีการประมวลผลที่รวดเร็ว อีกทั้งประสิทธิภาพในการรู้จำตัวอักษรยังอยู่ในระดับดี (28–30) ทั้งนี้ งานวิจัยของ Elglaly and Quek (28) ได้ทำการทดลองการรู้จำตัวอักษรอารบิกที่เป็นตัวอักษรโดด โดยใช้เพอร์เซปตรอนหลายชั้น (Multilayer Perceptrons: MLP) และ kNN เป็นขั้นตอนวิธีที่ช่วยในการแยกประเภทข้อมูล เป็นที่น่าสนใจว่าการรู้จำตัวอักษรอารบิกด้วยขั้นตอนวิธี kNN ให้ความถูกต้องถึง 90% ในขณะที่ MLP ให้ความถูกต้องเพียง 60% เท่านั้น ในกรณีนี้จะเห็นได้ว่า kNN ให้ประสิทธิภาพในการรู้จำที่สูงกว่า MLP ถึง 30%

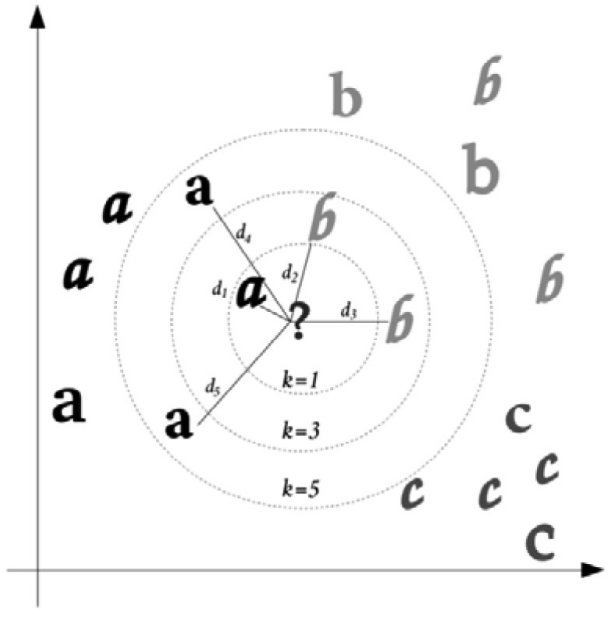


Figure 4 Finding the most similar examples between the training data and the unknown data (for example, k = 1, 3 and 5).

แต่ทั้งนี้ kNN ยังให้คำตอบที่ผิดพลาดในกรณีที่โครงสร้างของตัวอักษรใกล้เคียงกัน และที่สำคัญ ในกรณีที่ใช้เข้ามูลนำเข้าชุดเดียวกันkNN มีความเร็วมากกว่า MLP ถึง 150 เท่า

เมื่อนำ *k*NN ไปเปรียบเทียบกับเทคนิคอื่น เช่น ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine: SVM) จากงานวิจัยของ Kale et al., (14) แสดงให้เห็นว่า SVM เป็นขั้นตอนวิธีที่มีประสิทธิภาพดีกว่า *k*NN ทั้งนี้เนื่องจากการคำนวณของ SVM นั้นมีการคำนวณที่ซับซ้อน และใช้เวลาประมวลผลนานกว่า *k*NN แต่งานวิจัยก็แสดงให้เห็นว่าขั้นตอนวิธี *k*NN ให้ประสิทธิภาพสูงมากกว่า 95% ซึ่งงานวิจัยนี้ได้ทดลองรู้จำตัวอักษรเทวานาครี (Devanagari) ในการทดลองได้ใช้ตัวอักษรจำนวนทั้งสิ้น 27,000 ตัวอักษร โดยที่ตัวอักษรนั้นถูกทำการปรับเปลี่ยนขนาดให้เป็น 90x90 พิกเซล จากนั้นจึงหาคุณลักษณะพิเศษของตัวอักษรด้วยวิธีการเซอนิเกอร์โมเมนต์ (Zernike Moment) เมื่อได้คุณลักษณะพิเศษจากตัวอักษรแล้วจึงนำไปจัดแบ่งกลุ่มด้วยวิธี SVM และ *k*NN ผลจากการทดสอบประสิทธิภาพปรากฏว่า SVM ให้ประสิทธิภาพสูงถึง 98.37% ในขณะที่ *k*NN มีความถูกต้องอยู่ที่ 95.82%

นอกจากนั้น ในงานวิจัยของ Vyas and Goswami (31) ได้นำขั้นตอนวิธี *k*NN, SVM และโครงข่ายประสาทเทียม (Artificial Neural networks: ANN) มาเปรียบเทียบอัตราการรู้จำตัวเลขคุรชาต (Gujarati) โดยใช้คุณลักษณะพิเศษทั้งสิ้น 3 วิธี ประกอบด้วย การดัดแปลงรหัสโซ่ (Modified Chain Code: MCC), การแปลงฟูริเยร์แบบไม่ต่อเนื่อง (Discrete Fourier Transform: DFT) และการแปลงโคไซน์แบบไม่ต่อเนื่อง (Discrete Cosine Transform: DCT) ซึ่งชุดข้อมูลตัวเลข Gujarati ที่ใช้ในการทดลองประกอบด้วย 3,000 ตัวเลข ซึ่งการทดสอบจะแบ่งข้อมูลออกเป็น 3 ชุดข้อมูล 1) ข้อมูลจำนวน 900 ตัวเลขที่เขียนด้วยปากกาที่มีเส้นหนา 2) ข้อมูลจำนวน 2,100 ตัวเลขที่เขียนด้วยปากกาที่มีเส้นปกติ และ 3) นำข้อมูลชุดที่ 1 และ 2 มารวมกัน จากนั้นนำข้อมูลทั้งสามชุดไปคำนวณด้วยคุณลักษณะพิเศษทั้ง 3 วิธี สุดท้ายนำคุณลักษณะพิเศษที่ได้ไปผ่านกระบวนการรู้จำด้วยวิธี *k*NN, SVM และ ANN จากการทดลองโดยใช้คุณลักษณะพิเศษที่ได้จากเทคนิค MCC ปรากฏว่า*k*NN ให้ผลการรู้จำที่สูงกว่าทั้ง SVM และ ANN โดยอัตรการรู้จำของ *k*NN, SVM และ ANN เมื่อทดสอบกับข้อมูลชุดที่ 3 คือ 85.67%, 83.63% และ 84.89% ตามลำดับ อย่างไรก็ตามเมื่อใช้คุณลักษณะพิเศษที่ได้จากการคำนวณด้วย DFT และ DCT ปรากฏว่า SVM มีอัตราการรู้จำที่สูงกว่าทั้ง *k*NN และ ANN

Borji et al. (32) นำเสนอการรู้จำตัวอักษร ฟาซี (Farsi) และตัวเลขจากชุดข้อมูล MNIST โดย Farsi นั้นเป็นตัวอักษรในภาษาเปอเซียร์ และยังเป็นภาษาทางราชการของประเทศอิหร่าน ข้อมูลชุด Farsi ที่ใช้ในการทดลองประกอบไปด้วยข้อมูลชุดเรียนรู้จำนวน 60,000 ตัวอักษร และข้อมูลชุดทดสอบจำนวน 20,000 ตัวอักษร งานวิจัยนี้เพื่อหาคุณลักษณะพิเศษของตัวอักษรโดยใช้เทคนิคซีทู (C2 Feature) โดยศึกษาถึงตัวกรองแบบกาเบอร์ (Gabor) และการหาความแตกต่างของเกาส์เซียน (Difference of Gaussian: DoG) ที่อยู่ในกระบวนการทำงานของ HMAX (Hierarchical Model and X) โดยตัวอักษรที่ใช้นั้นถูกเปลี่ยนขนาดให้เป็น 32x32 พิกเซล จากนั้นจึงนำเข้ากระบวนการ C2 เพื่อหาคุณลักษณะพิเศษ โดยการทดลองนี้ใช้ ANN, SVM และ *k*NN เป็นขั้นตอนวิธีสำหรับการรู้จำ จากการทดลองพบว่า SVM ให้ประสิทธิภาพดีกว่า ANN ใน ชุดข้อมูล MNIST ส่วนชุดข้อมูล Farsi ปรากฏว่า ANN ให้ประสิทธิภาพดีกว่า SVM เพียงเล็กน้อย แต่เมื่อมองในส่วนของขั้นตอนวิธี *k*NN ปรากฏว่าประสิทธิภาพในการรู้จำในข้อมูลชุด MNIST และ Farsi อยู่ที่ 87.8% และ 91.5% ตามลำดับ

Kumar et al. (3) นำเสนอการรู้จำตัวอักษร เกอมูวกี่ (Gurmukhi) ตัวอักษรนี้ถูกเขียนขึ้นโดย ชาวซิกข์เพื่อใช้สำหรับสื่อสารในภาษา ปัญจาบซึ่งเป็นรัฐที่มีผู้คนที่ใช้ภาษานี้ถึง 130 ล้านคน ในงานวิจัยฉบับนี้ ได้เลือกผู้เขียนจำนวน 100 คน จากทั้งโรงเรียน วิทยาลัย สถานที่ราชการ และสถานที่อื่น ๆ โดยตัวอักษรที่ใช้ในงานวิจัยจะถูกปรับเปลี่ยนให้มีขนาดที่ใหญ่ถึง 100x100 พิกเซล จากนั้นเปลี่ยนให้อยู่ในรูปแบบของเส้นแสดงรูปร่าง เพื่อนำไปใช้หาคุณลักษณะพิเศษด้วยวิธีการหาค่าความหนาแน่นของแต่ละโซนที่เส้นทแยงมุม (Diagonal Features) ลากผ่าน และตัวอักษรจะถูกเปลี่ยนให้เป็นเส้นแสดงโครงร่าง เพื่อหาคุณลักษณะพิเศษด้วยวิธีการคำนวณหาค่าการเปลี่ยนแปลงจากพื้นหลังไปยังวัตถุ (Transition Features) โดยคำนวณจากด้านซ้ายไปยังด้านขวา และจากด้านบนลงด้านล่าง จากนั้นจึงใช้ขั้นตอนวิธี *k*NN ทดสอบประสิทธิภาพของทั้งสองคุณลักษณะพิเศษ ซึ่งพบว่าอัตราการรู้จำของ Diagonal Features อยู่ที่ 94.12% และอัตราการรู้จำของ Transition Features อยู่ที่ 86.57%

Surinta et al. (33) ได้นำเสนอการหาคุณลักษณะพิเศษด้วยขั้นตอนวิธีฮอตสปอต (Hotspot) ซึ่งเป็นวิธีที่ไม่ยุ่งยาก และใช้ *k*NN เป็นขั้นตอนวิธีในการแบ่งกลุ่มข้อมูล โดยจุดเด่นของ เทคนิค Hotspot คือมีเพียง 2 พารามิเตอร์ที่ต้องปรับแต่ง เท่านั้น ประกอบด้วย 1) จำนวนของจุด Hotspot และ 2) จำนวนทิศทางของรหัสโซ่ และได้นำขั้นตอนวิธี Hotspot ไปทดสอบกับข้อมูลจำนวน 3 ชุด ได้แก่ ข้อมูลตัวอักษรลายมือเขียนภาษาไทย ข้อมูลตัวเลขแบงกลา (Bangla) และข้อมูลชุด MNIST ซึ่งงานวิจัยได้สุ่มเลือกข้อมูลจาก MNIST มาเพียง 10,000 ตัวอักษร โดยทุกชุดข้อมูลถูกเปลี่ยนขนาดให้เป็น 40x40 พิกเซล และใช้ขั้นตอนวิธีที่ทำให้ตัวอักษรบาง ทั้งนี้เพื่อให้ตัวอักษรมีขนาดกว้างเพียง 1 พิกเซลเท่านั้น เมื่อนำขั้นตอนวิธี Hotspot, ทิศทางของมาร์ก (Mark Direction: MD) และทิศทางของรหัสโซ่ (Direction of Chain Code: DCC) มาทำการเปรียบเทียบ ปรากฏว่าขั้นตอนวิธี Hotspot ให้ประสิทธิภาพสูงที่สุดในชุดข้อมูล MNIST และ Bangla โดยมีอัตราการรู้จำที่ 89.9% และ 90.1% ตามลำดับ สำหรับข้อมูลตัวอักษรภาษาไทยปรากฏว่าขั้นตอนวิธี MD (อัตราการรู้จำ 88.0%) กลับให้ประสิทธิภาพที่เหนือกว่า Hotspot (อัตราการรู้จำ 83.36%) และ DCC (71.3%) จากผลของการทดลองแสดงให้เห็นว่าขั้นตอนวิธี Hotspot เหมาะสำหรับนำไปใช้กับข้อมูลที่เป็นชนิดตัวเลข

Rathi et al. (34) นำเสนอผลการรู้จำ ตัวอักษร Devanagari เฉพาะตัวสระเพียงเท่านั้น โดยใช้ข้อมูลตัวอักษรลายมือเขียนจากสถาบัน ISI (Indian Statistical Institute) เขียนโดยบุคคลที่อาศัยในเมือง โกลกาตา ซึ่งมีตัวสระของอักษรเทวานาครีจำนวน 9,191 สระ จากนั้นจึงแบ่งข้อมูลออกเป็นสี่ส่วน โดย 3 ส่วนถูกแบ่งให้เป็นข้อมูลชุดเรียนรู้ และอีก 1 ส่วนถูกแบ่งให้เป็นข้อมูลชุดทดสอบ งานวิจัยนี้ได้ใช้ขั้นตอนวิธีฟีเจอร์ไมนิ่ง (Feature Mining) เพื่อหาคุณลักษณะพิเศษของตัวอักษร เนื่องจากในตัวอักษรลายมือเขียนนั้น ลักษณะของการเขียนตัวอักษรย่อมมีความ แตกต่างกัน ซึ่งเทคนิคของ Feature Mining นั้นใช้ประโยชน์จากลักษณะรูปร่างของตัวอักษรเป็นหลัก ในกระบวนการของการรู้จำได้ใช้ขั้นตอนวิธี *k*NN สำหรับการจัดกลุ่มของตัวอักษร โดยรู้จำตัวสระผิดเพียง 88 ตัวเท่านั้น จากทั้งหมด 2,281 ตัว และมีความถูกต้องสูงถึง 96.14%

Dhandra et al. (35) นำเสนอการรู้จำเฉพาะตัวสระของภาษากันนาดา (Kannada) และภาษาอังกฤษ โดยใช้เทคนิคในการแบ่งพื้นที่ของตัวอักษรเพื่อหาคุณลักษณะพิเศษของตัวอักษรทั้งสองภาษา โดยตัวอักษรถูกปรับเปลี่ยนให้มีขนาด 32x32 พิกเซล จากนั้นตัวอักษรจะถูกแบ่งออกเป็น 64 โซน ซึ่งค่าความหนาแน่นของจุดพิกเซลในแต่ละโซนจะถูกนำมาใช้เป็นคุณลักษณะพิเศษ ตัวสระในภาษากันนาดานั้นมีทั้งสิ้น 14 ตัวอักษร ในงานวิจัยนี้ใช้ตัวอักษรในชุดเรียนรู้ และชุดทดสอบจำนวนชุดละ 700 ตัวอักษร ซึ่งอัตราการรู้จำโดยใช้ขั้นตอนวิธี *k*NN นั้นอยู่ที่ 92.71% สำหรับตัวอักษรภาษาอังกฤษนั้นเป็นตัวอักษรพิมพ์ใหญ่ ข้อมูลชุดเรียนรู้ และชุดทดสอบมีจำนวนชุดละ 1,300 ตัวอักษร ความถูกต้องที่ได้อยู่ที่ 97.51%

และงานวิจัยนี้ยังได้นำทั้งสระของภาษา Kannada และภาษาอังกฤษมารวมเข้าด้วยกัน ทำให้กลุ่มของตัวอักษรเพิ่มขึ้นเป็น 40 กลุ่ม จากการทดสอบ ความถูกต้องอยู่ที่ 95.77%

Dhandra et al. (36) นำเสนอขั้นตอนวิธีที่ใช้ลักษณะรูปร่างของตัวอักษรลายมือเขียนมาคำนวณเพื่อหาคุณลักษณะพิเศษ ได้แก่ วิธีการปรับเปลี่ยนช่วงข้อมูลของรหัสโซ่ (Normalized Chain Code) และเวฟเล็ตฟิลเตอร์ (Wavelet Filters) โดยได้นำไปใช้กับ ตัวสระของภาษา Kannada ซึ่งมีข้อมูลทั้งสิ้น 1,400 ตัวอักษร โดยข้อมูลถูกแบ่งออกเป็นข้อมูลชุดทดสอบและชุดเรียนรู้ จำนวนชุดละ 700 ตัวอักษร โดย ตัวอักษรถูกแปลงให้เป็นภาพสีขาวดำ และเปลี่ยนให้มีขนาด 40x40 พิกเซล จากนั้นใช้ขั้นตอนวิธี *k*NN เพื่อทำการรู้จำตัวอักษร จากผลของการทดสอบมีอัตราการรู้จำอยู่ที่ 95.08%

Babu et al. (37) ได้นำเสนอการหาคุณลักษณะพิเศษโดยใช้โครงสร้างของตัวอักษร (Structural Features) และหลักทางสถิติ (Statistical Features) เข้ามาช่วยในการรู้จำตัวเลข ซึ่งใช้ชุดข้อมูล MNIST จำนวน 60,000 ตัวอักษรเพื่อใช้ในการเรียนรู้ และใช้ตัวอักษรเพียง 5,000 จากทั้งหมด 10,000 ตัวอักษรเพื่อใช้ในการทดสอบ โดยคุณลักษณะพิเศษที่ใช้ในงานวิจัยนี้ได้แก่ 1) การหาองค์ประกอบที่เกี่ยวโยงกัน (Connected Component) เพื่อหาจำนวนของเส้นแสดงรูปร่าง 2) การหาอัตราส่วนระหว่างความสูงและความกว้างของตัวอักษร 3) การหาค่าการกระจายของพิกเซลของตัวอักษรที่ปรากฏอยู่ในแต่ละพื้นที่ที่ถูกกำหนดไว้ (Distribution Features) และ 4) วิธีการหาคุณลักษณะพิเศษจากเส้นแสดงโครงร่าง (Skeleton Features) วิธีนี้ถูกแบ่งออกเป็น 4.1 การหาจำนวนของจุดตัดระหว่างเส้นที่ลากผ่านทั้งในแนวตั้งและแนวนอน 4.2 การหาจุดสิ้นสุด การหาจุดเชื่อมต่อ และการการจุดตัดของตัวอักษร จากนั้นจึงนำข้อมูลคุณลักษณะพิเศษที่ได้คำนวณไว้ข้างต้น มาทำการจัดแบ่งกลุ่มโดยนำขั้นตอนวิธี *k*NN ไปเปรียบเทียบกับโครงข่ายประสาทเทียมแบบแพร่ย้อนกลับ (Back Propagation Neural Network: BPNN) ผลการทดลองปรากฏว่า *k*NN และอัตรา BPNN มีอัตราการรู้จำ 98.42% และ 95.4% ตามลำดับ จึงสรุปได้ว่า *k*NN ให้ประสิทธิภาพที่ดีกว่า BPNN ถึง 3%

Surinta et al. (25) ได้ศึกษาขั้นตอนวิธีที่ใช้คำนวณเพื่อหาคุณลักษณะพิเศษด้วยเทคนิค Local Gradient จำนวน 2 ขั้นตอนวิธี ได้แก่ Histograms of Oriented Gradients (HOG) และ Scale Invariant Feature Transform (SIFT) โดยนำขั้นตอนวิธีทั้ง 2 มาหาคุณลักษณะพิเศษของชุดข้อมูลตัวอักษรลายมือเขียน จำนวน 3 ชุดข้อมูล ได้แก่ ALICE-THI, LATIN และ Bangla ในงานวิจัยนี้ได้ทำการทดลองเปลี่ยนขนาดของตัวอักษรลายมือเขียนเป็น 28x28 และ 36x36 พิกเซล โดยใช้ *k*NN และ SVM เป็นขั้นตอนวิธีในการแบ่งกลุ่มข้อมูล จากการทดลองพบว่าขั้นตอนวิธีทั้ง HOG และ SIFT เป็นขั้นตอนวิธีที่มีความมั่นคงต่อการย่อขยาย หมุนของตัวอักษร ผลการทดลองโดยใช้ SVM เป็นเครื่องมือในการแบ่งกลุ่มข้อมูล พบว่า SIFT ให้ประสิทธิภาพที่เหนือกว่า HOG โดยที่ขั้นตอนวิธี SIFT ให้อัตราการรู้จำสูงมากกว่า 98% ในทุกชุดข้อมูล ยกเว้นเพียงข้อมูลชุด Bangla ที่เป็นตัวอักษรลายมือเขียน (BANG-C45) เท่านั้น โดยมีอัตราการรู้จำอยู่ที่ 85.6% ทั้งนี้เนื่องจากจำนวนของข้อมูลชุด BANG-C45 มีลักษณะการเขียนตัวอักษรนั้นมีความแตกต่างกันเป็นอย่างมาก มีจำนวนข้อไม่เพียงพอ และตัวอักษรบางตัวมีลักษณะที่ใกล้เคียงกัน เมื่อนำคุณลักษณะพิเศษที่ได้จากขั้นตอนวิธี SIFT และ HOG มาผ่านกระบวนการรู้จำโดยใช้ขั้นตอนวิธี *k*NN ปรากฏว่าอัตราการรู้จำสูงมากกว่า 95% ยกเว้นเพียงข้อมูลชุด BANG-C45 เท่านั้น ที่มีอัตราการรู้จำอยู่ที่ 74.50% งานวิจัยนี้ยังแสดงให้เห็นถึงประสิทธิภาพของขั้นตอนวิธีแบบ Local gradient ที่ใช้สำหรับหาคุณลักษณะพิเศษของตัวอักษรลายมือเขียน แม้ว่าจะนำคุณลักษณะพิเศษจากทั้ง SIFT และ HOG มาผ่านการรู้จำด้วย *k*NN ซึ่งเป็นขั้นตอนวิธีที่มีลักษณะของกระบวนการที่ไม่ซับซ้อน สิ่งที่น่าสนใจก็คือยังสามารถให้ประสิทธิภาพในการรู้จำตัวอักษรในอัตราที่สูง

จากงานวิจัยที่ศึกษาถึงขั้นตอนวิธี *k*NN เพื่อใช้ในการรู้จำตัวอักษรลายมือเขียนข้างต้น แสดงให้เห็นถึงประสิทธิภาพในการทำงานของ *k*NN ซึ่งอยู่ในระดับดี และให้อัตราการรู้จำที่สูงมากกว่า 90% ยกเว้นข้อมูลบางชุดเท่านั้น ทั้งนี้อาจเนื่องมาจากตัวแปรที่หลากหลาย เช่นการหาคุณลักษณะพิเศษของตัวอักษรอาจไม่สามารถนำคุณลักษณะที่โดดเด่นของตัวอักษรแต่ละตัวมาใช้งานได้เพียงพอ ข้อมูลมีจำนวนไม่เพียงพอต่อการประมวลผล ตัวอักษรในชุดข้อมูลมีลักษณะโครงสร้างที่ใกล้เคียงกัน หรือแม้แต่ตัวอักษรในกลุ่มเดียวกันอาจเขียนออกมาในลักษณะที่มีความแตกต่างกันมาก เป็นต้น ที่กล่าวมาทั้งหมดเป็นเพียงข้อมูลสมมุติฐานที่อัตราการรู้จำโดยใช้ขั้นตอนวิธี *k*NN ต่ำกว่า 90% อย่างไรก็ตาม จากการศึกษาพบว่า ขั้นตอนวิธี *k*NN ให้อัตราการรู้จำที่สูงที่สุดถึง 98.42% โดยทดสอบกับข้อมูลชุด MNIST ที่ใช้ข้อมูลชุดเรียนรู้จำนวน 60,000 และข้อมูลชุดทดสอบจำนวน 5,000 ตัวเลข โดยใช้คุณลักษณะพิเศษที่แตกต่างกันถึง 4 เทคนิคด้วยกัน (37) การศึกษาการรู้จำตัวอักษรลายมือเขียนโดยใช้ขั้นตอนวิธี *k*NN แสดงดัง (Table 2)

นอกจากจะใช้เพียงขั้นตอนวิธี *k*NN ในการ รู้จำตัวอักษรลายมือเขียน ซึ่งสามารถเรียกวิธีการแบบนี้ว่าการใช้เพียงขั้นตอนวิธีเดียวในการรู้จำซึ่งอาจเป็นเรื่องยากที่จะทำให้

มีอัตราการรู้จำที่สูงยังมีอีกหลากหลายงานวิจัยที่ใช้เอ็นเซมเบิลเทคนิค (Ensemble Technique) ซึ่งเป็นที่นิยมใช้กันอย่างกว้างขวางในงานด้านการจำแนกรูปแบบ (Pattern Classification) และ ML (38) วิธีนี้ทำงานด้วยการนำผลที่ได้จากการคาดการณ์ของแต่ละขั้นตอนวิธีมาจัดกลุ่มใหม่ โดยในแต่ละ Classifier อาจจะแสดงค่าการคาดการณ์ที่อาจจะแตกต่าง หรือเหมือนกัน ดังนั้นค่าการคาดการณ์ที่ปรากฏบ่อยที่สุดจาก Classifier ทั้งหมด นั่นก็คือผลลัพธ์ของ Ensemble Technique

Kumar et al. (39) นำเสนอการรู้จำตัวเลขจากข้อมูลชุด MNIST โดยใช้ขั้นตอนวิธีที่ประกอบด้วย *k*NN, ANN และการวิเคราะห์จำแนกประเภทเชิงเส้น (Linear Discriminant Analysis: LDA) ในการแบ่งกลุ่มข้อมูล และคุณลักษณะพิเศษที่ใช้ในประกอบด้วย Profile-Based Feature และ Kirsch Operator Based Feature จากการทดลองปรากฏว่าอัตราการรู้จำสูงที่สุดคือ 93% เมื่อใช้ Classifier เพียงตัวเดียว และเมื่อใช้ MV ปรากฏว่าอัตราการรู้จำเพิ่มสูงขึ้นเป็น 98.05%

Azad et al. (38) ได้นำเสนอการรู้จำตัวเลขเปอร์เซียด้วยวิธี Classifier Fusion โดยใช้ *k*NN, ตัวจำแนกประเภทเชิงเส้น (Linear Classifier: LC) และ SVM เป็นขั้นตอนวิธีในการรู้จำตัวอักษร โดยค่าการคาดการณ์จากทั้ง 3 ขั้นตอนวิธีจะถูกนำไปหาคำตอบโดยวิธีการโหวตข้อมูลชุดนี้ประกอบด้วยข้อมูลชุดเรียนรู้ 15,000 และข้อมูลชุดทดสอบ 5,000 ตัวเลข ในงานวิจัยนี้ได้คำนวณหาคุณลักษณะพิเศษของตัวเลขจำนวน 3 เทคนิคประกอบด้วย Directional Chain Code Frequencies, Modified Edge Maps Features และ Transition Features จากการทดลองปรากฏว่าอัตราการรู้จำสูงถึง 99.90% จากนั้นทดสอบด้วยการใช้ข้อมูลทั้งสิ้น 20,000 ตัวเลข และแบ่งกลุ่มข้อมูลออกเป็น 4 ส่วน ทำให้มีอัตราการรู้จำเพิ่มขึ้นเป็น 99.97%

นอกเหนือไปจากนั้น งานวิจัยหลายฉบับแสดงให้เห็นถึงการนำ *k*NN ไปใช้งานร่วมกัน (Combination) กับขั้นตอนวิธีอื่น ๆ เพื่อเพิ่มประสิทธิภาพในการรู้จำตัวอักษรลายมือเขียน ดังต่อไปนี้

Pramod Kumar Sharma (40) นำเสนอการใช้งาน Classifier มากกว่าหนึ่งประเภท (Classifier Combination Method: CCM) เพื่อทำการรู้จำตัวเลขลายมือเขียน ซึ่งประกอบไปด้วย MLP, Learning Vector Quantization (LVQ) และ *k*NN โดยผลลัพธ์จากทั้ง 3 Classifier ถูกนำมารวมกันทั้งนี้เพื่อใช้เป็นข้อมูลนำเข้าสำหรับการคำนวณด้วย Discriminant Function การทดลองแบ่งออกเป็น 2 ส่วน คือMV ซึ่งมีอัตราการรู้จำ 97.2% และ Classifier และ CCM มีอัตราการรู้จำ 98.1% จากการทดลองแสดงให้เห็นว่าการรู้จำโดยใช้ CCM กับข้อมูลประเภทตัวเลขลายมือเขียนมีอัตราการรู้จำที่สูงกว่า MV

Zanchettin et al. (41) ได้นำขั้นตอนวิธี *k*NN มารวมเข้ากับ SVM ซึ่งเรียกวิธีการนี้ว่า “KNN-SVM” ทั้งนี้ SVM สามารถเพิ่มประสิทธิภาพของขั้นตอนวิธี *k*NN ในการรู้จำตัวอักษรลายมือเขียน จากข้อสังเกตุ ผลลัพธ์ที่ถูกต้องของ *k*NN โดยทั่วไปนั้นจะอยู่ที่ข้อมูลจำนวน 2 ข้อมูลเท่านั้น (*k=2*) ดังนั้น SVM จึงถูกนำมาเพื่อช่วยตัดสินใจ แบ่งกลุ่มให้กับข้อมูลในกรณีที่ *k*NN ไม่สามารถแยกแยะได้ เช่น โครงสร้างของ ตัวอักษรที่คล้ายกัน ทำให้มักจะเกิดข้อผิดพลาดเมื่อเขียนตัวอักษรหนึ่งและไปคล้ายกับอีกตัวอักษรหนึ่ง เป็นต้น ผลของการทดลองโดยการใช้ขั้นตอนวิธี *k*NN มีอัตราการรู้จำ 82.26% เมื่อทดสอบกับชุดข้อมูล ตัวอักษรลายมือเขียนที่เขียนด้วยตัวพิมพ์เล็กและเขียนด้วยตัวพิมพ์ใหญ่ซึ่งรวมกันเป็น 52 กลุ่ม และเมื่อนำขั้นตอนวิธี KNN-SVM มาใช้ปรากฏว่ามีอัตราการรู้จำเพิ่มขึ้นเป็น 83.76% ซึ่งเพิ่มขึ้นถึง 1.5%

สรุป

งานวิจัยฉบับนี้ได้สำรวจงานวิจัยที่เกี่ยวข้องกับการนำวิธีการคำนวณเพื่อนบ้านใกล้ที่สุด ***k*** ตำแหน่งไปใช้เพื่อรู้จำตัวอักษรลายมือเขียน โดย *k*NN เป็นวิธีการที่ทำงานไม่ซับซ้อน เหมาะสำหรับการจัดหมวดหมู่ข้อมูล ขั้นตอนวิธี *k*NN สามารถทำได้โดยหาค่าความใกล้เคียงระหว่างข้อมูลที่ต้องการจำแนก และข้อมูลทั้งหมด โดยนำข้อมูลที่มีค่าระยะห่างน้อยที่สุดจำนวน *k* ข้อมูลมาพิจารณาจัดกลุ่ม จากการสำรวจพบว่า การรู้จำตัวอักษรลายมือเขียนด้วยขั้นตอนวิธี *k*NN มีอัตราการรู้จำมากกว่า 90% ทั้งนี้อาจเนื่องจากหลายปัจจัย เช่น คุณลักษณะพิเศษ ที่ใช้ในงานวิจัย ลายมือเขียนที่อาจมีลักษณะที่แตกต่างกัน หรือตัวอักษรที่มีโครงสร้างที่ใกล้เคียงกัน อย่างไรก็ตามเมื่อรวมวิธีการคำนวณหาคุณลักษณะพิเศษที่คงทน ต่อการย่อขยาย และการหมุน เป็นต้น และขั้นตอนวิธี *k*NN เข้าด้วยกัน มีผลต่อการเพิ่มขึ้นของประสิทธิภาพการรู้จำตัวอักษร ท้ายที่สุด ขั้นตอนวิธี *k*NN ยังมีจุดเด่นทางด้านความเร็วเมื่อเทียบกับขั้นตอนวิธีประเภทอื่น เช่น ANN หรือ SVM เป็นต้น

Table 2 Study of various handwritten character recognition systems using *k*NN algorithm

Authors	Dataset				Image Size (Pixel)	Pre-processing	Feature Extraction	Performances
	Name	Train	Test	Classes				
Borji et al. (2008) (32)	MNIST	60,000	10,000	10	32x32	- N/A	- C2 features	87.80%
	Farsi	60,000	20,000	10				91.50%
Kumar et al. (2011) (3)	Gurmukhi	3,150	350	56	100x100	- Normalization - Contour Image	- Diagonal features - Transition features	94.12%
Surinta et al. (2012) (33)	Thai Character	5,310	590	68	40x40	- Cropping to exceeding part - Binarization - Normalization - Thinning	- Hotspot technique - Mark direction technique - Direction of chain code technique	88.00%
	MNIST	9,000	1,000	10				89.90%
	Bangla numeric	8,636	959	10				90.10%
Rathi et al. (2012) (34)	Devanagari Vowel	6,910	2,281	13	90x90	- Binarization - Removing isolated object - Median filter	- Feature mining algorithm	96.14%
Dhandra et al. (2012) (35)	Kannada vowel and English	2,000	2,000	40	32x32	- Noise removal - Normalization	- Density based zone features	95.77%
Dhandra et al. (2014) (36)	Kannada vowel	700	700	14	40x40	- Binarization	- Normalized chain code - Wavelet filters	95.07%
Elglaly and Quek (2014) (28)	Arabic	196	84	28	64x64	- Noise removal - Binarization - Normalization	- Height/Width - Number of black pixels / Number of - Number of horizontal transitions - Number of vertical transitions	90.00%

Table 2 Study of various handwritten character recognition systems using *k*NN algorithm (Cont.)

Authors	Dataset				Image Size (Pixel)	Pre-processing	Feature Extraction	Performances
	Name	Train	Test	Classes				
Kale et al. (2014) (14)	Devanagari Marathi	21,600	5,400	108	30x30	- Binarization - Noise removal - Boundary Tracing - Normalization - Skeletonization	- Zernike moment features	95.82%
Babu et al. (2014) (37)	MNIST	60,000	5,000	10	28x28	- Binarization - Noise removal	- The number of contours in the image - Area of the digit - Height, Width and Width to Height ratio features - Distribution features - Skeleton features	98.42%
Vyas and Goswami (2015) (31)	Gujarati Numeral	2,100	900	10	20x20	- Noise removal - Binarization - Normalization - Thinning	- Modified chain code - Discrete Fourier Transform (DFT) - Discrete Cosine Transform (DCT)	93.60%
Surinta et al. (2015) (25)	THI-C68	13,130	1,360	68	36X36	- Normalization - Convert to grayscale image	- Histograms of Oriented Gradients (HOG) - Scale Invariant Feature Transform (SIFT)	91.91%
	THI-D10	8,055	1,500	10				97.83%
	BANG-C45	4,627	900	45				69.67%
	BANG-C10	9,161	1,500	10				96.07%
	LATIN-C25	23,329	11,287	25				96.12%
	LATIN-C10	1,637	880	10				96.48%

เอกสารอ้างอิง

1. Mantas J. An Overview of Character Recognition Methodologies. Pattern Recognit. 1986;19(6):425–30.
2. Bag S, Harit G. A survey on Optical Character Recognition for Bangla and Devanagari Scripts. In: Sadhana Academy Proceedings in Engineering Sciences. 2013. p. 133–68.
3. Kumar M, Jindal M.K, Sarma R.K. *k* -Nearest Neighbor Based Offline Handwritten Gurmukhi Character Recognition. In: International Conference on Image Information Processing (ICIIP). 2011. p. 1–4.4.
   Borovikov E. A survey of Modern Optical Character Recognition Techniques. In: International Workshop on Applied Modelling & Simulation. 2004. p. 1–38.
5. Esposito F, Malerba D, Semeraro G. A knowledge-based approach to the layout analysis. In: International Conference on Document Analysis and Recognition (ICDAR). 1995. p. 466–71.
6. Singh V, Kumar B. Document Layout Analysis for Indian Newspapers Using Contour Based Symbiotic Approach. In 2014. p. 1–4.
7. Surinta O, Holtkamp M, Karabaa F, Oosten JP Van, Schomaker L, Wiering M. A Path Planning for Line Segmentation of Handwritten Documents. In: International Conference on Frontiers in Handwriting Recognition (ICFHR). IEEE; 2014. p. 175–80.
8. Fischer A, Frinken V, Fornés A, Bunke H. Transcription Alignment of Latin Manuscripts Using Hidden Markov Models. In: The Workshop on Historical Document Imaging and Processing (HIP). 2011. p. 29–36.
9. Fischer A, Keller A, Frinken V, Bunke H. Lexicon-Free Handwritten Word Spotting Using Character HMMs. Pattern Recognit Lett. 2012;33(7):934–42.
10. Tang H, Augustin E, Suen C.Y, Baret O, Cheriet M. Recognition of Unconstrained Legal Amounts Handwritten on Chinese Bank Checks. In: International Conference on Pattern Recognition (ICPR). 2004. p. 610–3.
11. Gaceb D, Eglin V, Lebourgeois F, Emptoz H. Physical Layout Segmentation of Mail Application Dedicated to Automatic Postal Sorting System. In: IAPR International Workshop on Document Analysis Systems (DAS). 2008. p. 408–14.
12. Pal U, Roy R.K, Kimura F. Bangla and English City Name Recognition for Indian Postal Automation. In: International Conference on Pattern Recognition. 2010. p. 1985–8.
13. Salimi H, Giveki D. Farsi/Arabic Handwritten Digit Recognition Based on Ensemble of SVD Classifiers and Reliable Multi-Phase PSO Combination Rule. Int J Doc Anal Recognit. 2013;16(4):371–86.
14. Kale K.V, Deshmukh P.D, Chavan S.V, Kazi M.M, Rode Y.S. Zernike Moment Feature Extraction for Handwritten Devanagari (Marathi) Compound Character Recognition. Int J Adv Res Artif Intell. 2014;3(1):68–76.
15. He S, Wiering M, Schomaker L. Junction Detection in Handwritten Documents and Its Application to Writer Identification. Pattern Recognit. Elsevier; 2015;48(12):4036–48.
16. Guerbai Y, Chibani Y, Hadjadji B. The Effective Use of the One-Class SVM Classifier for Handwritten Signature Verification Based on Writer-Independent Parameters. Pattern Recognit. 2015;48(1):103–13.
17. Bertolini D, Oliveira L.S, Justino E, Sabourin R. Texture-Based Descriptors for Writer Identification and Verification. Expert Syst Appl. Elsevier Ltd; 2013;40(6):2069–80.
18. Akao Y, Yamamoto A, Higashikawa Y. Assisting Forensic Writer Verification by Visualizing Diversity of Digit Handwritings. In: International Conference on Frontiers in Handwriting Recognition (ICFHR). IEEE; 2014. p. 110–5.
19. Wolfshaar J van de, Karaaba M.F, Wiering M.A. Deep Convolutional Neural Networks and Support Vector Machines for Gender Recognition Deep Convolutional Neural Networks and Support Vector Machines for Gender Recognition. In: IEEE Symposium Series on Computational Intelligence. IEEE; 2015. p. 188–95.
20. LeCun Y, Matan O, Boser B, Denker J.S, Henderson D, Howard RE, et al. Handwritten Zip Code Recognition with Multilayer Networks. In: International Conference on Pattern Recognition. 1990. p. 35–40.

21. LeCun Y, Cortes C, Burges C.J.C. The MNIST database of handwritten digits [Internet]. 1998. Available from: http://yann.lecun.com/exdb/mnist/
22. der Maaten L van. A New Benchmark Dataset for Handwritten Character Recognition. Technical Report. Tilburg University, Tilburg, The Netherlands. 2009.
23. Schomaker L.R.B, Vuurpijl L. Forensic Writer Identification: A Benchmark Data Set and a Comparison of Two Systems. Technical Report. Nijmegen University, Tilburg, The Netherlands. 2000.
24. Nopsuwanchai R, Biem A, Clocksin W.F. Maximization of Mutual Information for Offline Thai Handwriting Recognition. IEEE Trans Pattern Anal Mach Intell. 2006;28(8):1347–51.
25. Surinta O, Karaaba M.F, Schomaker L.R.B, Wiering M.A. Recognition of Handwritten Characters Using Local Gradient Feature Descriptors. Eng Appl Artif Intell. Elsevier; 2015;45:405–14.
26. Carmen Mart ınez, Fuentes O. Face Recognition Using Unlabeled Data. Iberoam J Comput Sci Res. 2003;7(2):123–9.
27. AlKhateeb J.H, Khelifi F, Jiang J, Ipson SS. A new approach for Off-Line Handwritten Arabic Word Recognition Using *k*NN Classifier. In: IEEE International Conference on Signal and Image Processing Applications. IEEE; 2009. p. 191–4.
28. Elglaly Y, Quek F. Isolated Handwritten Arabic Characters Recognition Using Multilayer Perceptrons and *k*-Nearest Neighbor Classifiers. In 2012. p. 1–6.
29. Cordella L, Stefano C De, Fontanella F, Marrocco C. A Feature Selection Algorithm for Handwritten Character Recognition. In: International Conference on Pattern Recognition (ICPR). 2008. p. 1–4.
30. Rashad M, Semary N.A. Isolated Printed Arabic Character Recognition Using *k*NN and Random Forest Tree Classifiers. Adv Mach Learn Technol Appl. 2014;488:11–7.
31. Vyas AN, Goswami MM. Classification of Handwritten Gujarati Numerals. In: International Conference on Advances in Computing, Communications and Informatics (ICACCI). 2015. p. 1231–7.
32. Borji A, Hamidi M, Mahmoudi F. Robust handwritten character recognition with features inspired by visual ventral stream. Neural Process Lett. 2008;28(2):97–111.
33. Surinta O, Schomaker L.R.B, Wiering M.A. Handwritten Character Classification Using the Hotspot Feature Extraction Technique. In: International Conference on Pattern Recognition Applications and Methods (ICPRAM). Vilamoura, Algarve, Portugal: SciTePress - Science and and Technology Publications; 2012. p. 261–4.
34. Rathi R, Krishan Pandey R, Chaturvedi V, Jangid M. Offline Handwritten Devanagari Vowels Recognition using *k*NN Classifier. Int J Comput Appl. 2012;49(23):11–6.
35. Dhandra B.V, Mukarambi G, Hangarge M. Handwritten Kannada Vowels and English Character Recognition System. Int J Image Process Vis Sci. 2012;1(1):12–7.
36. Dhandra B.V, Parameshwarapa S, Mukarambi G. Kannada Handwritten Vowels Recognition Based on Normalized Chain Code and Wavelet Filters. Int J Comput Appl Recent Adv Inf Technol. 2014;21–4.
37. Babu UR, Chintha A.K, Venkateswarlu Y. Handwritten Digit Recognition Using Structural, Statistical Features and K-nearest Neighbor Classifier. Int J Inf Eng Electron Bus. 2014;6(1):62–8.
38. Azad R, Azad B, Mogharreb I, Jamali S. Classifier Fusion Method to Recognize Handwritten Persian Numerals. Int J Cybern Informatics. 2014;3(3):1–10.
39. Kumar R, Goyal M.K, Ahmed P, Kumar A. Unconstrained Handwritten Numeral Recognition using Majority Voting Classifier. In: IEEE International Conference on Parallel, Distributed and Grid Compu ting (PDGC). 2012. p. 284–9.
40. Sharma P.K. Multiple Classifiers for Unconstrained Offline Handwritten Numeral Recognition. In: International Conference on Computational Intelligence and Multimedia Applications, (ICCIMA). 2007. p. 344–8.
41. Zanchettin C, Bezerra B.L.D, Azevedo W.W. A *k*NN-SVM Hybrid Model for Cursive Handwriting Recognition. In: International Joint Conference on Neural Networks (IJCNN). 2012. p. 1–8.