

บทที่ 3

ทฤษฎีที่ใช้ในการแก้ปัญหา

ในบทนี้จะกล่าวถึงทฤษฎีที่ใช้ในงานวิจัยโดยแบ่งออกเป็น 2 ส่วนได้แก่ ส่วนของการหา
ลักษณะเฉพาะจากเฟรม และ ส่วนที่ใช้สร้างตัวแทนของข้อมูล โดยส่วนแรกจะใช้ Scale
Invariant Feature Transform (SIFT) ในการหาลักษณะเฉพาะในแต่ละเฟรมของตัวอักษรและ
สระ สำหรับส่วนที่สองจะเป็นการจัดกลุ่มลักษณะเฉพาะของเฟรมทั้งหมดในแต่ละกลุ่มตัวอักษร
โดยจะใช้ Hard C-Means และ Fuzzy C-Means ในการจัดกลุ่ม

3.1. Scale invariant feature transform[2]

Scale Invariant Feature Transform หรือเรียกอีกอย่างหนึ่งว่า ชิฟต์ (SIFT) ใช้สำหรับการ
สร้างลักษณะเฉพาะของดิสคริปเตอร์แบบเฉพาะส่วน (local feature descriptors) โดยลักษณะเฉพาะ
ที่ได้ จะมีคุณสมบัติที่ทนทานต่อการเปลี่ยนแปลงขนาดวัตถุ การหมุนวัตถุ การบิดเบี้ยวรูปร่างของ
วัตถุ และการเปลี่ยนแปลงความเข้มแสงของวัตถุ นั่นคือ ลักษณะเฉพาะที่ได้จากวิธีการนี้ จะยังคง
คุณสมบัติของวัตถุเดิมไว้ได้นั่นเอง สำหรับขั้นตอนการหาลักษณะเฉพาะแบ่งเป็น 4 ขั้นตอนดังนี้

1) Scale space extrema detection

เป็นขั้นตอนแรกในการระบุจุดสำคัญในสเกลสเปซ โดยจะทำการค้นหาทุกๆ จุดรูป ด้วย
การใช้ฟังก์ชันผลต่างของเกาส์เซียน (difference of Gaussian) เพื่อหาจุดที่ไม่เปลี่ยนแปลงเมื่อมีการ
สเกลรูปและการเปลี่ยนแปลงตำแหน่งของรูป

ขั้นตอนนี้จะทำการตรวจหาจุดที่เกิดขึ้นซ้ำๆ ของวัตถุเมื่อมีการเปลี่ยนแปลงรูปร่างของวัตถุ
การตรวจสอบความเสถียรของจุดในวัตถุสามารถทำได้ด้วยการใช้ทฤษฎีสเกลสเปซ [11]

สำหรับฟังก์ชันของสเกลสเปซที่สามารถนำมาใช้งานได้ พบว่ามีเพียงฟังก์ชันเดียว คือ
ฟังก์ชันของเกาส์เซียน (Gaussian function) ซึ่งถูกเสนอโดย Koenderink [12] และ Lindeberg [13]

ถ้าให้สเกลสเปซของรูปแทนด้วย $L(x, y, \sigma)$ โดยเกิดจากการคอนโวลูชันของฟังก์ชัน
เกาส์เซียนกับรูป ดังสมการที่ (3.1)

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (3.1)$$

เมื่อให้ * คือ ตัวกระทำการคอนโวลูชัน ซึ่งเป็นค่าสี่ของรูป ณ ตำแหน่ง x และ y โดยที่

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \quad (3.2)$$

ในการตรวจหาจุดที่มีความเสถียรในสเกลสเปซ Lowe [14] ได้นำเสนอวิธีการหาจุดที่มีค่าสูงสุดหรือต่ำสุดในสเกลสเปซ $D(x, y, \sigma)$ ที่ซึ่งสามารถคำนวณได้จากการหาผลต่างของค่าสเกลสองค่าที่ใกล้เคียงกัน โดยค่าคงที่ที่ใช้ในการคูณในที่นี้คือ k

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (3.3)$$

สำหรับเหตุผลที่เลือกใช้ฟังก์ชันนี้มีสองข้อคือ

1. $L(x, y, \sigma)$ เป็นฟังก์ชันที่ใช้ในการวัดประสิทธิภาพของการเกลี่ยภาพ (smooth images) และ $D(x, y, \sigma)$ ก็ถูกคำนวณจากการหาผลต่างของรูป
2. Lindeberg [13] ได้แสดงให้เห็นว่า ผลต่างของฟังก์ชันของเกาส์เซียน (difference of Gaussian function) มีลักษณะใกล้เคียงกับฟังก์ชันสเกลนอร์มัลไลซ์ของลาปลาเซียนเกาส์เซียน ฟังก์ชัน $\sigma^2 \nabla^2 G$ โดยความสัมพันธ์ของ D และ $\sigma^2 \nabla^2 G$ สามารถอธิบายได้ด้วยสมการการแพร่ของความร้อน โดยแทน t ด้วย σ^2

$$\frac{\delta G}{\delta \sigma} = \sigma \nabla^2 G \quad (3.4)$$

$$\text{หรือ } \sigma \nabla^2 G = \frac{\delta G}{\delta \sigma} \approx \frac{G(x, y, k\sigma) - G(x, y, \sigma)}{k\sigma - \sigma} \quad (3.5)$$

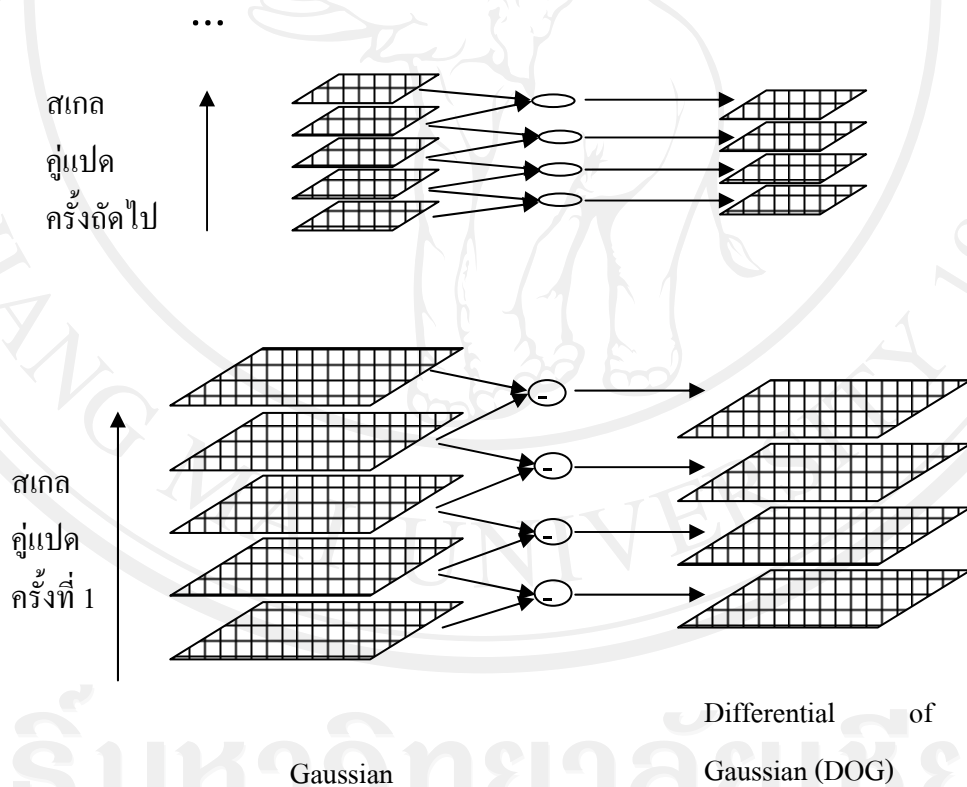
$$\text{จะได้ } (k-1)\sigma^2 \nabla^2 G = G(x, y, k\sigma) - G(x, y, \sigma) \quad (3.6)$$

ค่า $(k-1)$ เป็นค่าคงที่สำหรับทุกๆ การสเกล ดังนั้นจึงไม่มีผลกับการค้นหาค่าสูงสุดต่ำสุด และค่าความผิดพลาดจะมีค่าเป็น 0 เมื่อ k มีค่าเป็น 1 แต่ในการทดลองพบว่าค่า k จะไม่มีผลต่อการหาจุดเด่นในการสเกลที่มีค่านัยสำคัญที่แตกต่างกัน เช่น $k = \sqrt{2}$

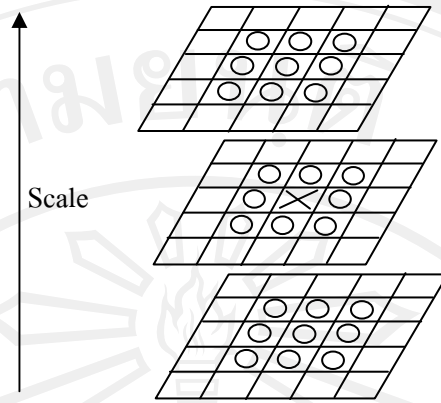
การหาค่า $D(x, y, \sigma)$ ถูกแสดงไว้ในรูปที่ 3.1 โดยรูปภาพตอนเริ่มต้น จะถูกนำไปคอนโวลูชันกับฟังก์ชันเกาส์เซียน โดยแบ่งเป็นแต่ละค่าคงที่ k ในสเกลสเปซ ถูกแสดงไว้ตรงคอลัมน์ด้านซ้าย จากนั้นทำการหารแต่ละคู่แปดด้วยค่าสองเท่าของ σ ให้เป็นจำนวนเต็ม s ทำให้ระยะห่างแต่ละคู่ของสเกลสเปซมีค่าเป็น $k = 2^{1/s}$ ดังนั้นรูปที่จะต้องสร้างขึ้นทั้งหมดจะมีอยู่ $s+3$ รูปใน 1 คู่

แปด เพื่อให้กระบวนการตรวจหาจุดที่เกิดขึ้นซ้ำๆ ครอบคลุมในแต่ละคู่แปด รูปที่อยู่ติดกันจะถูกนำมาลบกัน เพื่อหาผลต่างของฟังก์ชันเกาส์เซียนของแต่ละรูป โดยแสดงไว้ด้านขวาของรูปที่ 3.1 ถือเป็นอันเสร็จสิ้นกระบวนการคู่แปดแรก ในคู่แปดถัดไปจะเกิดขึ้นจากการชักตัวอย่างรูปเกาส์เซียนที่มีค่า σ เริ่มต้นเป็นสองเท่า (ในที่นี้คือ 2 รูปด้านบนสุด) โดยจะเก็บทุกๆ พิกเซลที่สองของแต่ละคอลัมน์และแถว ความแม่นยำจากการชักตัวอย่างเมื่อเทียบกับ σ ถือว่าไม่แตกต่างเมื่อใช้จุดเริ่มต้นของคู่แปดที่แล้ว

local extrema detection เป็นการเลือกจุดที่มากที่สุดหรือน้อยที่สุดของ $D(x, y, \sigma)$ คือ การเปรียบเทียบจุดตัวอย่างกับแปดจุดที่ถูกล้อมรอบในรูปปัจจุบัน และอีก 2 รูปที่อยู่ด้านบนและด้านล่างที่ติดกับรูปปัจจุบัน (ดังรูปที่ 3.2) โดยจุดนี้จะถูกเลือกก็ต่อเมื่อมีค่ามากที่สุดหรือน้อยที่สุดกว่า 27 จุด



รูปที่ 3.1 คู่แปด (octave) ของแต่ละสเกลสเปซ



รูปที่ 3.2 การหา Local extrema detection

2) Keypoint localization

เป็นการหาตำแหน่งของคีย์พอยต์ เป็นขั้นตอนที่เปรียบเทียบจุดที่ได้จากขั้นตอนที่ 1 เพื่อนำไปเป็นคีย์พอยต์ โดยขึ้นกับความเสถียรของแต่ละจุด

หลังจากได้ตัวแทนคีย์พอยต์จากการเปรียบเทียบกับพิกเซลที่อยู่ใกล้เคียงในขั้นตอนที่ 1 ขั้นตอนต่อไป คือ การหาข้อมูลของตำแหน่ง สเกล และอัตราส่วนของความสัมพันธ์ระหว่างข้อมูล โดยข้อมูลเหล่านี้จะถูกกำจัดทิ้งไป ถ้าจุดนั้นไม่มีความโดดเด่น (คือเกิดสัญญาณรบกวนได้ง่าย) หรือมีตำแหน่งที่ไม่ได้บ่งบอกถึงขอบ

Lowe [14] ได้เสนอวิธีการหาตำแหน่งของคีย์พอยต์ที่ตำแหน่งและสเกลของจุดที่กำลังพิจารณา โดย Brown [15] ได้สร้างฟังก์ชันกำลังสองแบบ 3 มิติเพื่อใช้ในการประมาณค่าให้ได้จุดที่มีค่ามากที่สุด การทดลองของเขาได้แสดงถึงควมมีเสถียรภาพและการจับคู่ที่ดี ด้วยการใช้การกระจายของเทเลอร์ (Taylor expansion) ในการเลื่อนฟังก์ชันสเกลสเปซ $D(x, y, \sigma)$ ทำให้จุดกำเนิดไปอยู่ที่จุดตัวอย่างที่กำลังพิจารณา

$$D(\mathbf{X}) = D + \frac{\partial D^T}{\partial \mathbf{X}} \mathbf{X} + \frac{1}{2} \mathbf{X}^T \frac{\partial^2 D}{\partial \mathbf{X}^2} \mathbf{X} \quad (3.7)$$

โดยที่ D และอนุพันธ์ของ D หาได้จากจุดตัวอย่าง และ $\mathbf{X} = (x, y, \sigma)^T$ จะเป็นระยะออฟเซตตำแหน่งของจุดวิกฤติ (extremum) $\mathbf{X}$ สามารถหาได้โดยการแก้สมการอนุพันธ์ของสมการที่ (3.7) โดยให้ $D(\mathbf{X})$ มีค่าเป็นศูนย์ จะได้ผลลัพธ์ดังสมการที่ (3.8)

$$\mathbf{X} = \frac{\partial^2 D^{-1}}{\partial \mathbf{X}^2} \frac{\partial D}{\partial \mathbf{X}} \quad (3.8)$$

Brown ได้เสนอว่า Hessian และอนุพันธ์ของ D สามารถหาได้จากผลต่างของจุดตัวอย่างที่อยู่รอบๆ ถ้าออฟเซตของ $\mathbf{X}$ มีค่ามากกว่า 0.5 ในแกนใดๆ ก็ตาม หมายความว่าจุดวิกฤติอยู่ระหว่าง

ผลต่างของจุดตัวอย่าง ในกรณีนี้ จุดตัวอย่างจะถูกเปลี่ยนและประมาณค่าให้มาแทนที่จุดนั้น ค่าออฟเซตสุดท้ายของ $\mathbf{X}$ จะถูกเพิ่มให้กับตำแหน่งของจุดตัวอย่างของ $\mathbf{X}$ เพื่อที่จะได้หาค่าประมาณตำแหน่งของจุดวิกฤติ

ค่าของฟังก์ชัน ณ จุดวิกฤติ $D(\mathbf{X})$ จะถูกนำไปใช้ในการกำจัดจุดสูงสุดหรือต่ำสุดที่ไม่เสถียรและมีค่าความต่างของสี่ด้านออก ซึ่งสามารถหาได้โดยการแทนสมการที่ (3.8) ลงไปในสมการที่ (3.7) จะได้

$$D(\mathbf{X}) = D + \frac{1}{2} \frac{\partial D^T}{\partial \mathbf{X}} \mathbf{X} \quad (3.9)$$

โดยถ้า $|D(\mathbf{X})|$ มีค่าน้อยกว่า 0.3 จะถูกกำจัดทิ้งไป

3) Orientation assignment

เป็นการหาทิศทางของคีย์พอยต์โดยจะขึ้นกับทิศทางของเกรเดียนต์ในแต่ละรูป โดยทุกๆ จุดมีลักษณะเป็นไปตามทิศทาง ขนาด และตำแหน่งของแต่ละลักษณะเฉพาะ ซึ่งทำให้สามารถทดแทนต่อการเปลี่ยนแปลงรูปร่างของวัตถุ

การกำหนดทิศทางของแต่ละคีย์พอยต์จะขึ้นกับคุณสมบัติของรูปนั้นๆ คีย์พอยต์คิสกริปเตอร์สามารถนำเสนอด้วยความสัมพันธ์ของทิศทางและทำให้เกิดการไม่แปรเปลี่ยนต่อการหมุนของรูป เป้าหมายของวิธีการนี้จะแตกต่างจากวิธีการของ Schmid และ Mohr [16] โดยที่คุณสมบัติของแต่ละรูปขึ้นอยู่กับความไม่แปรเปลี่ยนต่อการหมุน ข้อเสียของวิธีการนี้คือจำนวนของคิสกริปเตอร์ที่จำกัด

วิธีระบุทิศทางที่ให้ผลลัพธ์ที่มีความเสถียรมากที่สุด ทำได้โดยการสเกลคีย์พอยต์ของรูปที่ผ่านการเกลี่ยด้วยฟังก์ชันเกาส์เซียน (L) ด้วยสเกลที่มีค่าใกล้เคียงที่สุด ดังนั้นทุกๆ การคำนวณจะอยู่ในรูปแบบของความไม่แปรเปลี่ยนต่อการสเกล โดยสามารถคำนวณหาขนาดของเกรเดียนท์ $m(x,y)$ และทิศทาง $\theta(x,y)$ ดังต่อไปนี้

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \quad (3.10)$$

$$\theta(x,y) = \tan^{-1} \left((L(x,y+1) - L(x,y-1)) / (L(x+1,y) - L(x-1,y)) \right) \quad (3.11)$$

ฮิสโตแกรมของทิศทางมาจากทิศทางของเกรเดียนท์ของจุดตัวอย่าง ภายในบริเวณรอบๆ คีย์พอยต์ ฮิสโตแกรมของทิศทางจะมี 36 แท่งครอบคลุม 360 องศา แต่ละจุดตัวอย่างที่เพิ่มให้กับฮิส

โตแกรมจะถูกถ่วงน้ำหนักโดยขนาดของเกรเดียนต์โดยค่าน้ำหนักของเกาส์เซียนที่ล้อมรอบหน้าต่าง ด้วย σ จะมีค่า 1.5 เท่าของสเกลของคีย์พอยต์

ค่าสูงสุดในฮิสโตแกรมนั้นจะตรงกับทิศทางที่หลักของ เกรเดียนต์เฉพาะรูปร่างนั้นๆ เมื่อยอดที่สูงที่สุดในฮิสโตแกรมถูกตรวจพบและหลังจากนั้นยอดของฮิสโตแกรมที่มีค่าภายใน 80% ของยอดที่สูงที่สุดจะนำมาใช้สร้างคีย์พอยต์ด้วยทิศทางนั้น เพราะฉะนั้นสำหรับตำแหน่งที่มีค่าสูงสุดเท่ากันหลายค่า ก็จะมีคีย์พอยต์หลายคีย์พอยต์ที่สร้างมาที่ตำแหน่งและสเกลเดียวกันแต่มีทิศทางต่างกัน ซึ่งมีเพียง 15% ของจุดเท่านั้นที่จะให้ทิศทางหลายทิศทาง แต่สิ่งเหล่านี้ก็จะเป็นส่วนสำคัญที่จะช่วยให้การจับคู่มีความเสถียรมากขึ้น

4) Keypoint descriptor

เป็นการหาคีย์พอยต์ดิสคริปเตอร์ โดยเกิดขึ้นจากการเปลี่ยนแปลงรูปร่างของกลุ่มของคีย์พอยต์ โดยแต่ละคีย์พอยต์ประกอบไปด้วยค่าเกรเดียนต์ ใน 1 วัตถุ จะประกอบไปด้วยหลายๆ คีย์พอยต์เดสคริปเตอร์ ซึ่งก็คือลักษณะเฉพาะที่เกิดจากวิธีฝึกฝนตัวเอง ในที่นี้จะอธิบายการหาคีย์พอยต์ดิสคริปเตอร์แบ่งเป็นสองส่วนดังนี้

Local image descriptor

จากหัวข้อที่ได้อธิบายก่อนหน้านี้เป็นการหาตำแหน่ง, สเกล, และทิศทางของแต่ละคีย์พอยต์ ซึ่งตัวแปรเหล่านี้จะเป็นตัวกำหนดกระบวนการหาพิคคในสองมิติซึ่งจะใช้บอกรายละเอียดของรูป ขั้นตอนต่อไปคือการคำนวณหาดีสคริปเตอร์ของรูปที่มีลักษณะเฉพาะที่จะไม่แปรเปลี่ยนกับความสว่างของรูปหรือมุมมองที่เปลี่ยนไปในสามมิติ

วิธีที่จะทำให้ถึงเป้าหมายได้แบบไม่ซับซ้อน คือการหาตัวอย่างของความสว่างของรูปในรูปนั้นที่อยู่รอบคีย์พอยต์โดยมีการสเกลที่เหมาะสม และจะทำการจับคู่สิ่งเหล่านี้โดยการทำให้ค่าสหสัมพันธ์เป็นบรรทัดฐานเดียวกัน (normalized correlation) อย่างไรก็ตามการทำสหสัมพันธ์ (correlation) จะทำให้เกิดการเปลี่ยนแปลงของรูป ซึ่งจะเป็นสาเหตุที่ทำให้เกิดการจดจำตัวอย่างที่ผิดพลาด เช่น มุมมองที่เปลี่ยนไปในสามมิติ หรือวัตถุเกิดการบิดรูป

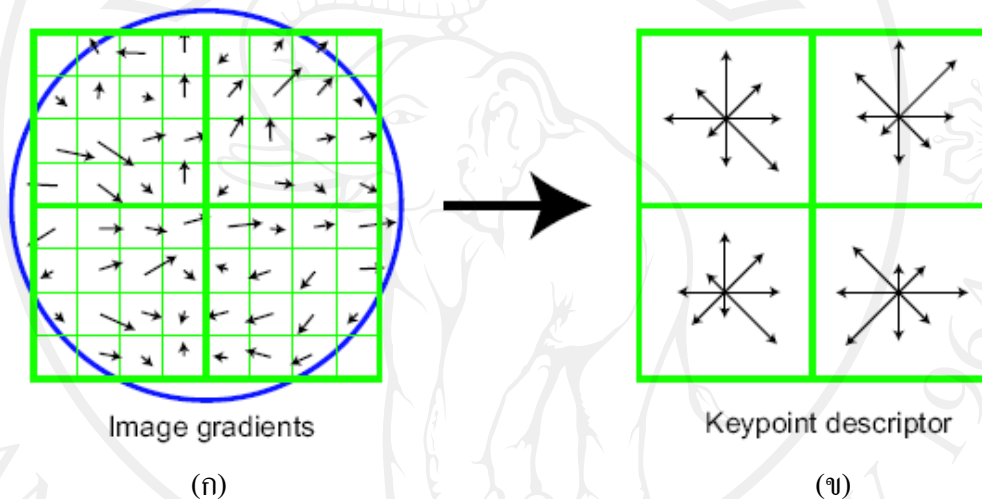
Descriptor representation

จาก แสดงการคำนวณของคีย์พอยต์ดิสคริปเตอร์ เริ่มจากการหาขนาดและมุมของเกรเดียนต์ ที่ถูกชักตัวอย่าง (sample) รอบๆตำแหน่งของคีย์พอยต์ โดยใช้สเกลของคีย์พอยต์ ในการเลือกระดับของ การเบลอดด้วยเกาส์เซียนของรูป เพื่อที่จะทำให้ทิศทางนั้นมีความคงที่ พิกคของดีสคริปเตอร์และมุมเกรเดียนต์จะถูกหมุนให้สัมพันธ์กับทิศทางของคีย์พอยต์ ในการที่จะทำให้เกิด

ประสิทธิภาพจะต้องทำการคำนวณหาเกรเดียนท์ก่อนสำหรับทุกๆ ระดับของพีรามิดดังที่ได้อธิบายในหัวข้อแรก ซึ่งจะแสดงด้วยลูกศรเล็กๆ ที่แต่ละตำแหน่งถูกแสดงไว้ดังรูปที่ 3.3 (ก)

ฟังก์ชันเกาส์เซียนที่มีการถ่วงน้ำหนักด้วย σ มีค่าเท่ากับ 1.5 เท่าของความกว้างของหน้าต่าง descriptor ซึ่งจะถูกกำหนดให้เป็นน้ำหนักของขนาดของแต่ละจุดตัวอย่าง สามารถแสดงได้ดังวงกลมรูปที่ 3.3 (ก)

เป้าหมายของหน้าต่างเกาส์เซียนคือหลีกเลี่ยงการเปลี่ยนแปลงของดิสคริปเตอร์กับการเปลี่ยนแปลงของตำแหน่งของหน้าต่าง และให้ความสนใจกับเกรเดียนท์ที่อยู่ห่างไกลจากจุดศูนย์กลางของดิสคริปเตอร์น้อยลง



รูปที่ 3.3 การหาคีย์พอยต์ดิสคริปเตอร์

จากรูปที่ 3.3 คีย์พอยต์ดิสคริปเตอร์จะถูกสร้างโดยการคำนวณขนาดของเกรเดียนท์และมุมของแต่ละจุดตัวอย่าง ในบริเวณรอบๆ คีย์พอยต์ ดังที่แสดงทางด้านซ้าย ซึ่งดิสคริปเตอร์เหล่านี้จะถูกให้น้ำหนักโดยหน้าต่างจากฟังก์ชันเกาส์เซียน โดยแสดงด้วยวงกลมที่ทับอยู่บนรูปด้านซ้าย ตัวอย่างเหล่านี้จะถูกรวมไว้ในฮิสโตแกรมของทิศทางบนพื้นที่ย่อยขนาด 4×4 ดังที่แสดงในรูปด้านขวามือ โดยที่ความยาวของลูกศรแต่ละอันจะมีขนาดเท่ากับผลรวมของขนาดของเกรเดียนท์ ที่อยู่ใกล้กับทิศทางนั้นภายในพื้นที่เดียวกัน ซึ่งรูปนี้จะแสดงอารีย์ของดิสคริปเตอร์ขนาด 2×2 ที่คำนวณจากเซตของตัวอย่าง ขนาด 8×8

จากรูปที่ 3.3 (ข) แสดงคีย์พอยต์ดิสคริปเตอร์ พิจารณาความสำคัญของตำแหน่งของเกรเดียนท์ โดยการสร้างฮิสโตแกรมของทิศทางที่ครอบคลุมพื้นที่ของตัวอย่าง ขนาด 4×4 ซึ่งในรูปที่ 3.3 (ข) แสดงทิศทางแปดทิศทางสำหรับแต่ละฮิสโตแกรมทิศทาง ซึ่งความยาวของแต่ละลูกศรมีขนาดเท่ากับขนาดของฮิสโตแกรมนั้น

สำหรับการหลีกเลี่ยงผลกระทบของขอบเขตในการเปลี่ยนแปลงแบบจับพลันของคิสคริปเตอร์ นั้นมีความสำคัญเช่นเดียวกับการย้ายตัวอย่าง จากฮิสโตแกรมหนึ่งไปอีกฮิสโตแกรมหนึ่ง หรือจากทิศทางหนึ่งไปอีกทิศทางหนึ่ง ดังนั้นการประมาณค่าแบบไตรลิเนียร์ (trilinear interpolation) จะถูกนำมาใช้กระจายค่าตัวอย่างของเกรเดียนท์ให้กับแท่งที่อยู่ติดกันของฮิสโตแกรม สามารถอธิบายได้อีกอย่างว่า แต่ละทิศทางที่อยู่ในแท่งของฮิสโตแกรมจะถูกนำมาคูณโดยน้ำหนักของ $1-d$ ของแต่ละมิติ โดยที่ d คือระยะของตัวอย่างจากค่ากลางของแท่งของฮิสโตแกรม

คิสคริปเตอร์ถูกสร้างมาจากเวกเตอร์ที่ประกอบด้วยค่าของทุกๆ ฮิสโตแกรมทิศทาง โดยใช้ความยาวของลูกศรดังรูปที่ 3.3 (ก) ซึ่งในรูปแสดงอาร์เรย์ขนาด 2×2 ของฮิสโตแกรมทิศทาง โดยที่การใช้งานจริงจะให้ผลดีที่สุดเมื่ออาร์เรย์ของฮิสโตแกรมมีขนาดเป็น 4×4 โดยมีทั้งหมดอยู่ 8 ฮิสโตแกรม ดังนั้นคีย์พอยต์จะมีทั้งหมด $4 \times 4 \times 8 = 128$ ลักษณะเฉพาะ

3.2. Hard C-Means[17]

Hard C-Means หรือเรียกอีกอย่างหนึ่งว่า HCM เมื่อทำการจัดกลุ่มสิ้นสุดลง แต่ละเวกเตอร์ข้อมูลจะเป็นสมาชิกได้เพียงหนึ่งกลุ่มเท่านั้น ในที่นี้จะขอยกตัวอย่างวิธีการของ HCM ดังต่อไปนี้ เริ่มจากพิจารณาชุดข้อมูล $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ โดยที่ $\mathbf{x}_k$ เป็นเวกเตอร์ใน d มิติ และมีเซตของความเป็นสมาชิก (Membership Values) $P = \{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_c\}$ เมื่อทำการจัดกลุ่มข้อมูลออกเป็น c กลุ่ม โดยที่ $\mathbf{u}_i(\mathbf{x}_k)$ คือ ค่าความเป็นสมาชิกของเวกเตอร์ $\mathbf{x}_k$ ในกลุ่มข้อมูลที่ i ซึ่ง $\mathbf{u}_i(\mathbf{x}_k) \in \{0, 1\}$ โดย i มีค่าตั้งแต่ 1 ถึง c และ k มีค่าตั้งแต่ 1 ถึง n

โดยมีเงื่อนไขของค่าความเป็นสมาชิกคือ $\sum_{i=1}^c \mathbf{u}_i(\mathbf{x}_k) = 1$

นิยามสมการจุดประสงค์ (Objective Function) ของ HCM คือ

$$J(P) = \sum_{k=1}^n \sum_{i=1}^c \mathbf{u}_i(\mathbf{x}_k) \|\mathbf{x}_k - \mathbf{v}_i\|^2 \quad (3.12)$$

โดยที่สัญลักษณ์ $\|\bullet\|$ คือ Inner Product Norm

เมื่อ $\mathbf{V} = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_1, \dots, \mathbf{v}_c\}$ เป็นเซตของโปรโตไทป์ของกลุ่มข้อมูล สมการที่ใช้ในการปรับปรุง (update) ค่าของความเป็นสมาชิกดังสมการที่ (3.13) ถูกสร้างขึ้นจากเงื่อนไขว่าแต่ละชุดข้อมูล จะอยู่ใกล้กับโปรโตไทป์ได้เพียงแค่โปรโตไทป์เดียวเท่านั้น นั่นคือ

$$\mathbf{u}_i(\mathbf{x}_k) = \begin{cases} 1 & \text{if } \|\mathbf{x}_k - \mathbf{v}_i\|^2 = \min_{i=1, \dots, c} \|\mathbf{x}_k - \mathbf{v}_i\|^2 \\ 0 & \text{other} \end{cases} \quad (3.13)$$

และทำการหาค่าที่เหมาะสมที่สุด (optimization) ของ $J(P)$ เทียบกับแต่ละโปรโตไทป์ จะสามารถหาตัวแทนของแต่ละกลุ่มข้อมูลดังสมการที่ (3.14)

$$\mathbf{v}_i = \frac{\sum_{k=1}^n \mathbf{u}_i(\mathbf{x}_k) \mathbf{x}_k}{\sum_{k=1}^n \mathbf{u}_i(\mathbf{x}_k)} \quad (3.14)$$

สำหรับวิธีการนำไปใช้แก้ปัญหา สามารถเริ่มได้สองวิธีคือ การกำหนดตัวแทนของชุดข้อมูล หรือ กำหนดค่าความเป็นสมาชิกตอนเริ่มต้น ในที่นี้จะยกตัวอย่างโดยการเริ่มต้นด้วยการกำหนดตัวแทนของชุดข้อมูลก่อน

ขั้นตอนที่ 1 ให้ $t=0$ กำหนดค่าตัวแทนของชุดข้อมูล $\mathbf{V}^t$ ทั้ง c กลุ่ม

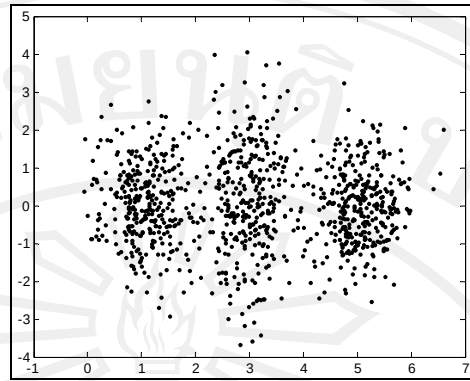
ขั้นตอนที่ 2 หาค่าความเป็นสมาชิก $P^{(t+1)}$ สำหรับทุกค่าของ $\mathbf{x}_k \in \mathbf{X}$ โดยใช้สมการที่ (13)

ขั้นตอนที่ 3 หาค่าตัวแทนชุดข้อมูล $\mathbf{V}^{t+1}$ ของทั้ง c กลุ่ม $[\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_c]$ สำหรับ $P^{(t+1)}$ โดยใช้สมการที่ (14)

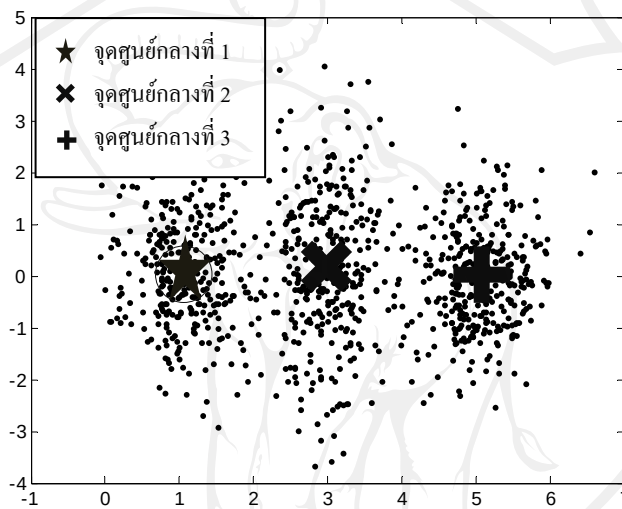
ขั้นตอนที่ 4 เปรียบเทียบ $\mathbf{v}_i^{(t)}$ กับ $\mathbf{v}_i^{(t+1)}$ ถ้า $\max_{i=1, \dots, c} |\mathbf{v}_i^{(t+1)} - \mathbf{v}_i^{(t)}| \leq \varepsilon$ ให้หยุดการทำงาน (ε เป็นค่าขอบเขตขั้นต่ำเพื่อให้หยุดการทำงาน) แต่ถ้านอกเหนือจากเงื่อนไขนี้ ให้เพิ่มค่า t เป็น $t+1$ แล้วกลับไปทำซ้ำในขั้นตอนที่ 2 โดยอาจกำหนดให้หยุดการทำงานถ้าค่า t เท่ากับจำนวนรอบสูงสุดที่กำหนดไว้

รูปที่ 3.4 ข) แสดงตัวอย่างของการจัดกลุ่มด้วยวิธี HCM โดยที่ รูปที่ 3.4 ก) แสดงตัวอย่างชุดข้อมูลที่ได้จากการสุ่มข้อมูล 3 กลุ่มโดยมีจุดศูนย์กลางที่ $[1,0]$ $[3,0]$ และ $[5,0]$ การกระจายของข้อมูลในแนวแกน x และ y เป็น $[1,2]$ $[1,3]$ และ $[1,2]$ ตามลำดับ รูปที่ 3.4 ข) แสดงตัวแทนของกลุ่มข้อมูลทั้ง 3 กลุ่มที่ได้จากวิธีการของ HCM

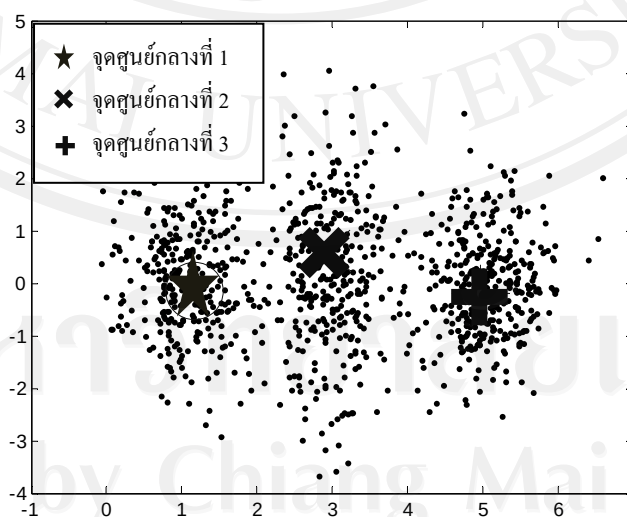
วิธีการ HCM นี้เป็นวิธีการจัดกลุ่มแบบทั่วไปโดยมีข้อเสียคือ ในกรณีที่ข้อมูลบางตัวอย่างเป็นสมาชิกมากกว่าหนึ่งกลุ่ม แต่วิธีการ HCM จะไม่สามารถจัดกลุ่มได้เนื่องจากเงื่อนไขของความเป็นสมาชิกคือ เป็นสมาชิกได้เพียงหนึ่งกลุ่ม กรณีที่มีข้อมูลในรูปแบบนี้ได้มีการคิดค้นวิธีการ FCM ดังรูปที่ 3.4 ค) แสดงตัวแทนของกลุ่มข้อมูลทั้ง 3 กลุ่มที่ได้จากวิธีการของ FCM โดยจะถูกระบุไว้ในขั้นตอนต่อไป



(ก) ตัวอย่างชุดข้อมูลที่ได้จากการสุ่มข้อมูล



(ข) ตัวแทนกลุ่มข้อมูลทั้ง 3 กลุ่มที่ได้จากวิธีการของ HCM



(ค) ตัวแทนกลุ่มข้อมูลทั้ง 3 กลุ่มที่ได้จากวิธีการของ FCM

รูปที่ 3.4 ตัวอย่างของกลุ่มข้อมูล และวิธีการจัดกลุ่มด้วย HCM และ FCM

3.3. Fuzzy C-Means (FCM)[17]

Fuzzy C-Means หรือเรียกอีกอย่างหนึ่งว่า FCM จัดได้ว่าเป็นวิธีการที่นิยมใช้มากในปัจจุบัน ซึ่งได้พัฒนาขึ้นมาจากวิธีการของ HCM ที่มีข้อจำกัดว่า $u_i(\mathbf{x}_k) \in \{0,1\}$ เปลี่ยนเป็น $u_i(\mathbf{x}_k) \in [0,1]$ โดย i มีค่าตั้งแต่ 1 ถึง c และ k มีค่าตั้งแต่ 1 ถึง n

สิ่งที่เปลี่ยนไปของ HCM กับ FCM คือสมาชิกของข้อมูลในวิธีการ HCM จะสามารถเป็นสมาชิกได้เพียงหนึ่งกลุ่มเท่านั้น แต่วิธีการของ FCM จะเป็นสมาชิกมากกว่าหนึ่งกลุ่มได้ ทำให้ความยืดหยุ่นของวิธีการจัดกลุ่มของ FCM สูงกว่าวิธีการ HCM

วิธีการ FCM มีตัวแปรที่เพิ่มขึ้นมาคือค่า fuzzifier หรือค่า m

โดยที่ m จะเป็นค่าคงที่ ที่มีผลต่อ Membership Values $m \in \mathbf{R}^+$ และ $m \geq 1$

โดยมีนิยามสมการจุดประสงค์ (Objective Function) ของ FCM คือ

$$J_m(P) = \sum_{k=1}^n \sum_{i=1}^c [u_i(\mathbf{x}_k)]^m \|\mathbf{x}_k - \mathbf{v}_i\|^2 \quad (3.15)$$

โดยที่สัญลักษณ์ $\|\bullet\|$ คือ Inner Product Norm และมีเงื่อนไขสำหรับ Membership Values

ว่า

$$\sum_{i=1}^c u_i(\mathbf{x}_k) = 1 \quad k = 1, 2, \dots, n \quad (3.16)$$

$$0 < \sum_{k=1}^n u_i(\mathbf{x}_k) < n \quad i = 1, 2, \dots, c \quad (3.17)$$

เป้าหมายของการจัดกลุ่มข้อมูลคือ การหาเซตของค่าความเป็นสมาชิก ที่ทำให้สมการจุดประสงค์ $J_m(P)$ มีค่าน้อยที่สุด โดยใช้วิธีการหาค่าที่เหมาะสมที่สุด (Optimization) ในการแก้ไขปัญหานี้ โดยสามารถอธิบายวิธีการได้ดังต่อไปนี้

ขั้นตอนที่ 1 ให้ $t=0$ กำหนดค่าตั้งต้นของตัวแทนชุดข้อมูลทั้ง c กลุ่ม และเลือกค่า m

ขั้นตอนที่ 2 หาค่า $P^{(t+1)}$ สำหรับทุกๆค่าของ $\mathbf{x}_k \in \mathbf{X}$

ถ้า $\|\mathbf{x}_k - \mathbf{v}_i^{(t)}\|^2 > 0$ ให้

$$u_i^{(t+1)}(\mathbf{x}_k) = \left[\sum_{j=1}^c \left(\frac{\|\mathbf{x}_k - \mathbf{v}_i^{(t)}\|^2}{\|\mathbf{x}_k - \mathbf{v}_j^{(t)}\|^2} \right)^{\frac{1}{m-1}} \right]^{-1} \quad (3.18)$$

ถ้า $\|\mathbf{x}_k - \mathbf{v}_i^{(t)}\|^2 = 0$ สำหรับบางค่า $i \in I$ ให้ $\mathbf{u}_i^{(t+1)}(\mathbf{x}_k)$ สำหรับ $i \in I$ เป็นจำนวนจริงบวกใดๆ

$$\text{โดยที่ } \sum_{i \in I} \mathbf{u}_i^{(t+1)}(\mathbf{x}_k) = 1 \quad (3.19)$$

และให้ $\mathbf{u}_i^{(t+1)}(\mathbf{x}_k) = 0$ สำหรับค่า i ค่าอื่นๆ

ขั้นตอนที่ 3 คำนวณตัวแทนชุดข้อมูลของทั้ง c กลุ่ม $[\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_c]$ สำหรับ $P^{(t+1)}$

$$\mathbf{v}_i = \frac{\sum_{k=1}^n [\mathbf{u}_i(\mathbf{x}_k)]^m \mathbf{x}_k}{\sum_{k=1}^n [\mathbf{u}_i(\mathbf{x}_k)]^m} \text{ เมื่อ } i = 1, 2, \dots, c \quad (3.20)$$

ขั้นตอนที่ 4 เปรียบเทียบ $\mathbf{v}_i^{(t)}$ กับ $\mathbf{v}_i^{(t+1)}$ ถ้า $\max_{i=1, \dots, c} |\mathbf{v}_i^{(t+1)} - \mathbf{v}_i^{(t)}| \leq \varepsilon$ ให้หยุดการทำงาน (ε เป็นค่าขอบเขตขั้นต่ำเพื่อให้หยุดการทำงาน) แต่ถ้านอกเหนือจากเงื่อนไขนี้ ให้เพิ่มค่า t เป็น $t+1$ แล้วกลับไปทำซ้ำในขั้นตอนที่ 2 โดยอาจกำหนดให้หยุดการทำงานถ้าค่า t เท่ากับจำนวนรอบสูงสุดที่กำหนดไว้

สำหรับตัวอย่างของวิธีการจัดกลุ่มข้อมูลด้วย FCM ถูกแสดงไว้ดังรูปที่ 3.4 ค)