บทที่ 3

การรับรู้เสียงพูด

ในการรับรู้เสียงพูด ก็เป็นลักษณะหนึ่งของการรับรู้รูปแบบ (Pattern Recognition) คือ จะเป็นการเปรียบเทียบระหว่าง แบบทดสอบ (Test Pattern) กับ แบบอ้างอิง (Reference Pattern) ซึ่งเป็นรูปแบบที่เราทราบและเก็บไว้ล่วงหน้า

ขั้นตอนในการรับรู้รูปแบบแบ่งเป็น 2 ขั้นตอน คือ (Parson T.W., 1986)

ก. ขั้นเรียนรู้ (Learning) จะเป็นการสร้างกลุ่มของแบบอ้างอิง ในการรับรู้เสียงพูด ในขั้นนี้จะทำการวิเคราะห์เสียงพูดก่อน แล้วเก็บลักษณะของเสียงในรูปของพารามิเตอร์ พร้อมกับป้ายกำกับ เพื่อจะใช้เปรียบเทียบในขั้นต่อไป รายละเอียดของการวิเคราะห์เสียงพูดได้กล่าวไว้ในบทที่ 2.

ข. ขั้นรับรู้ (Recognition) จะเป็นการทดสอบการรับรู้ ระหว่างแบบอ้างอิง กับแบบทดสอบ โดยจะทำการเปรียบเทียบพารามิเตอร์ของแบบทดสอบ กับแบบอ้างอิงทั้งหมด แบบอ้างอิงที่ถูกเลือก คือ แบบอ้างอิงที่มีพารามิเตอร์ใกล้เคียงกับแบบทดสอบที่สุด เทคนิคในการเปรียบเทียบมีอยู่หลายวิธี ดังจะได้กล่าวต่อไป

3.1 การจัดกลุ่มเพื่อสร้างแบบอ้างอิง

การสร้างกลุ่มของแบบอ้างอิงเป็นขั้นตอนในส่วนของการเรียนรู้ โดยเริ่มต้นจากการนำข้อมูลสัญญาณเสียงที่ได้จากส่วนของการวิเคราะห์ มาทำการสร้างเป็นกลุ่มของแบบอ้างอิง ในระบบการรับรู้เสียงพูดแบบต่างบุคคล จะใช้แบบอ้างอิงของคำหนึ่ง ๆ จากผู้พูดจำนวนมาก เพื่อที่จะได้ครอบคลุมถึงความแปรปวนต่าง ๆ ที่จะเกิดขึ้นระหว่างผู้พูดแต่ละคน แต่อย่างไรก็ตามผลที่เกิดขึ้นตามมาเมื่อจำเป็นจะต้องมีข้อมูลจำนวนมากในกลุ่มของแบบอ้างอิง คือ

ก. เวลาที่ใช้ในการตอบสนอง ในการรับรู้เสียงพูด ขั้นตอนการเปรียบเทียบแบบทดสอบกับแบบอ้างอิงทั้งหมดที่มี เป็นขั้นตอนที่สำคัญในการรับรู้ เมื่อแบบอ้างอิงมีจำนวนมาก การเปรียบเทียบจึงจำเป็นที่จะต้องอาศัยเวลาเพิ่มขึ้นด้วย

ข. เนื้อที่ในหน่วยความจำสำรองที่ใช้ในการเก็บแบบอ้างอิง เมื่อมีแบบอ้างอิงจำนวนมากที่จำเป็นต้องใช้ในการเปรียบเทียบ เนื้อที่ในหน่วยความจำสำรองก็จำเป็นต้องมีเพิ่มขึ้นด้วย

ค. ความถูกต้องในการรับรู้ เนื่องจากจำนวนแบบอ้างอิงในแต่ละคำไม่ได้เป็นสัดส่วนโดยตรงกับความถูกต้องในการรับรู้ กล่าวคือเมื่อเราเพิ่มจำนวนแบบอ้างอิงในแต่ละคำไปจนถึงระดับหนึ่ง ความถูกต้องในการรับรู้จะเริ่มคงที่ และในบางครั้งอาจจะมีค่าลดลงด้วย (Rabiner L.R., 1979)

จากเหตุผลดังกล่าว จึงทำให้มีความพยายามที่จะจัดกลุ่มของแบบอ้างอิงในแต่ละคำใหม่ เพื่อให้ได้แบบอ้างอิงในจำนวนที่พอเหมาะ และสามารถใช้เป็นตัวแทนของแบบอ้างอิงที่มีอยู่ทั้งหมดได้ อัลกอริทึมที่ใช้ในการแบ่งกลุ่มมีอยู่เป็นจำนวนมาก สำหรับอัลกอริทึมที่จะกล่าวถึง คือการหาค่าเฉลี่ย K (K-means algorithm) (Rabiner L.R., 1979) ซึ่งประกอบด้วยขั้นตอนต่าง ๆ ดังนี้

3.1.1 การหาจุดศูนย์กลางของกลุ่ม (Center Cluster)

ในขั้นตอนนี้จะทำการกำหนดจำนวนกลุ่มของแบบอ้างอิงให้มีค่าเท่ากับ K และเริ่มต้นจากแบบอ้างอิงที่มีความยาวมากที่สุดเป็นจุดศูนย์กลางของกลุ่มแรก จากนั้นจะหาจุดศูนย์กลางของคำถัดไปได้จาก แบบอ้างอิงที่มีระยะทางห่างจากจุดศูนย์กลางของกลุ่มแรกมากที่สุด และจุดศูนย์กลางของกลุ่มถัดไปจะได้จาก แบบอ้างอิงที่มีระยะทางห่างจากจุดศูนย์กลางของกลุ่มทั้งสองมากที่สุด ขั้นตอนนี้จะถูกทำซ้ำจนได้กลุ่มของแบบอ้างอิงครบตามจำนวน K กลุ่ม หรือค่าระยะทางที่มากที่สุดมีค่าน้อยกว่าระดับที่กำหนด วิธีการนี้เรียกว่า Farthest Neighbor

3.1.2 การจัดกลุ่มใหม่ (Reclassification)

แบบอ้างอิงที่เหลืออยู่ในแต่ละคำ จะถูกจะให้เข้าในแต่ละกลุ่ม โดยมีเงื่อนไขคือ ระยะทางของแบบอ้างอิงนี้กับแบบอ้างอิงในกลุ่มจะมีค่าเฉลี่ยน้อยที่สุด หลังจากที่แบบอ้างอิงทุกแบบถูกจัดเข้ากลุ่มเรียบร้อยแล้ว จะทำการหาจุดศูนย์กลางของแต่ละกลุ่มใหม่ ขั้นตอนนี้จะถูกทำซ้ำจนกว่าจุดศูนย์กลางของแต่ละกลุ่มจะไม่มีการเปลี่ยนแปลง อย่างไรก็ตามจากขั้นตอนดังกล่าวจะเห็นว่า การกำหนดค่าจำนวนกลุ่มที่เหมาะสมอาจจะทำได้ลำบาก ดังนั้นจึงได้มีการปรับปรุงเทคนิคนี้เป็น การกำหนดค่าระยะทางสูงสุดที่จะทำให้เกิดกลุ่มใหม่ขึ้น คือเมื่อแบบอ้างอิงใดที่ถูกเลือกให้เข้ากลุ่มด้วยระยะทางที่น้อยที่สุดแล้ว แต่ระยะทางนี้ยังคงมีค่ามากกว่าระดับที่กำหนด แบบอ้างอิงนี้จะถูกนำไปสร้างเป็นจุดศูนย์กลางของกลุ่มใหม่ทันที

3.2 การเปรียบเทียบแบบไดนามิคโปรแกรมมิ่ง

เป็นที่ทราบกันดีอยู่แล้วว่า การเปลี่ยนแปลงอัตราเร็วในการเปล่งเสียง เป็นสาเหตุของการแกว่งไปมา ของรูปแบบสัญญาณเสียงบนแกนเวลา การกำจัดผลของการแกว่งนี้เป็นปัญหาสำคัญประการหนึ่งในการรับรู้เสียงพูด ในยุคเริ่มแรกได้มีการเสนอวิธีการปรับปรุงเชิงเส้น

(Linear Normalization) เพื่อที่จะกำจัดความแตกต่างของเวลา ระหว่างรูปแบบสัญญาณเสียง 2 รูปแบบ โดยทำการเปลี่ยนรูปเชิงเส้น (Linear Transformation) บนแกนเวลา แต่อย่างไรก็ตาม วิธีนี้ยังไม่ประสบความสำเร็จในการปรับปรุงความถูกต้องของการรับรู้เสียงพูด (Sakoe H., 1978)

เทคนิคของไดนามิคโปรแกรมมิ่งเป็นแนวความคิดใหม่ ที่จะใช้ในการกำจัดความแตกต่างของเวลาระหว่างรูปแบบสัญญาณ 2 รูปแบบ โดยการสร้างแบบจำลองของฟังก์ชันที่ไม่เป็นเชิงเส้น โดยให้ชื่อว่า ฟังก์ชันแรปปิ้ง (Wrapping Function) ความแตกต่างของเวลาจะถูกกำจัดออกไปได้โดยอาศัยแบบจำลองของฟังก์ชันนี้

3.2.1 <u>ฟังก์ชันแรปปิ้ง</u> (Rabiner L.R., 1978)

เราสามารถแสดงสัญญาณเสียงออกมาให้อยู่ในรูปของเวคเตอร์ได้ คือ

$$A = a_1, a_2, \ldots, a_i, \ldots, a_I$$

$$B = b_1, b_2, \ldots, b_j, \ldots, b_J \qquad \ldots\ldots(3.2.1)$$

ในการที่จะแก้ปัญหาความแตกต่างของเวลา ระหว่างรูปแบบของสัญญาณเสียง 2 รูปแบบนั้น สามารถทำได้โดย แทนที่จะอธิบายสัญญาณด้วยธรรมชาติของการแกว่งบนแกนเวลา หรือความแตกต่างของเวลา เราจะอธิบายด้วยระนาบ i-j ดังแสดงในรูปที่ 3.2.1 โดยรูปแบบ A และ B จะถูกแสดงบนแกน i และ j ตามลำดับ จากรูปจะได้ว่า ความแตกต่างของเวลาสามารถแสดงได้ด้วยลำดับของจุด c = (i,j) จะได้ว่า

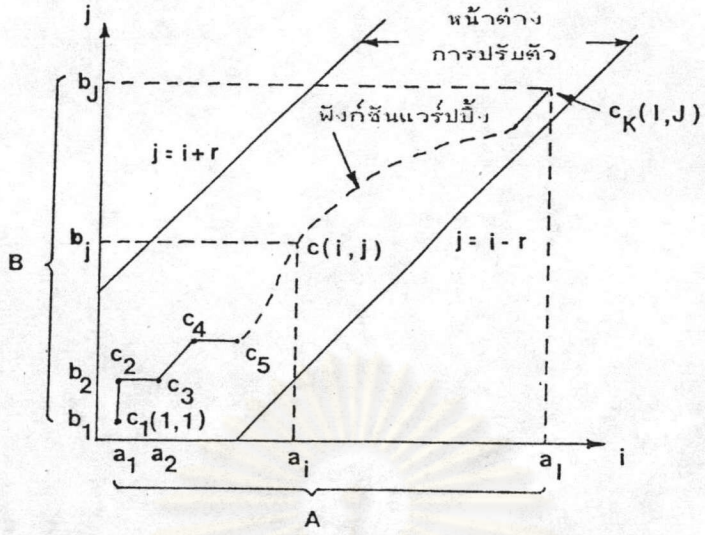
$$F = c(1), c(2), \ldots, c(k), \ldots, c(K) \qquad \ldots\ldots(3.2.2)$$

โดย $c(k) = (i(k), j(k))$

จากคู่ลำดับนี้จะสามารถมองเป็นฟังก์ชัน ซึ่งจะโยงแกนเวลาของรูปแบบสัญญาณเสียง A ไปยัง B เราเรียกฟังก์ชันนี้ว่า ฟังก์ชันแรปปิ้ง (Wrapping Function) ดังแสดงในรูปที่ 3.2.1 ถ้าไม่มีความแตกต่างของเวลาระหว่างสัญญาณทั้งสองแล้ว จะได้ว่าฟังก์ชันแรปปิ้งจะมีลักษณะเป็นเส้นทแยงมุม i=j ฟังก์ชันนี้จะมีการเปลี่ยนแปลงไปตามความแตกต่างของเวลาที่เกิดขึ้น

ให้ความแตกต่างระหว่างเวคเตอร์ a_i และ b_j เป็น

$$d(c) = d(i,j) = \|a_i - b_j\| \qquad \ldots\ldots(3.2.3)$$



รูปที่ 3.2.1 แสดงให้เห็นลักษณะของฟังก์ชันแรปปิ้ง
(Sakoe H., 1978)

ดังนั้นจะได้ว่า ผลต่างรวมของฟังก์ชันแรปปิ้ง F คือ

$$E(F) = \sum_{k=1}^{K} d(c(k)).w(k) \qquad .....(3.2.4)$$

โดยที่ w(k) เป็นสัมประสิทธิ์น้ำหนัก (Weight Coefficient) ซึ่งจะทำให้การวัด E(F) มีความยืดหยุ่นขึ้น ในการหาทางเดินของฟังก์ชันแรปปิ้งที่ดีที่สุดนั้น จะได้จากค่าผลต่างรวมที่น้อยที่สุดตามทางเดินนั้น และเพื่อที่จะให้ได้ค่าคงตัวบนแกนเวลา เราสามารถเขียนระยะทางระหว่างรูปแบบสัญญาณเสียง A และ B ได้เป็น

$$D(A,B) = \min_{F} \left[\frac{\sum_{k=1}^{K} d(c(k)).w(k)}{\sum_{k=1}^{k} w(k)} \right] \qquad .....(3.2.5)$$

โดยที่ตัวหาร $\sum w(k)$ จะเป็นตัวชดเชยผลของค่า K คือ จำนวนจุดของฟังก์ชันแรปปิ้ง จากสมการที่ (3.2.5) จะได้รูปทั่วไปของระยะทางการปรับตัวตามแกนเวลา (Time-normalized Distance) เราเรียกวิธีการที่จะแก้สมการนี้ได้อย่างมีประสิทธิภาพว่า ไดนามิคโปรแกรมมิ่ง (Dynamic Programming) อย่างไรก็ตามการหาฟังก์ชันแรปปิ้งที่เหมาะสมนั้น ยังขึ้นอยู่กับข้อกำหนดของฟังก์ชันแรปปิ้ง และการระบุถึงสัมประสิทธิ์น้ำหนักอีกด้วย

ในการหาค่าฟังก์ชันแรปปิ้ง ได้มีการกำหนดเงื่อนไขไว้ดังนี้ คือ

ก. เงื่อนไขโมโนโทนิค (Monotonic Condition)

$$i(k-1) \leq i(k) \quad \text{และ} \quad j(k-1) \leq j(k)$$

ข. เงื่อนไขความต่อเนื่อง (Continuity Condition)

$$i(k) - i(k-1) \leq 1 \quad \text{และ} \quad j(k) - j(k-1) \leq 1$$

จากเงื่อนไขทั้ง 2 นี้ ทำให้ได้ความสัมพันธ์ระหว่างจุด 2 จุดคือ

$$c(k-1) = \begin{cases} (i(k), j(k)-1) \\ (i(k)-1, j(k)-1) \\ \text{หรือ} \quad (i(k)-1, j(k)) \end{cases} \qquad \text{.....(3.2.6)}$$

ค. เงื่อนไขขอบเขต (Boundary Condition)

$$i(1) = 1 \ , \quad j(1) = 1 \quad \text{และ}$$

$$i(K) = I \ , \quad j(K) = J \qquad \text{.....(3.2.7)}$$

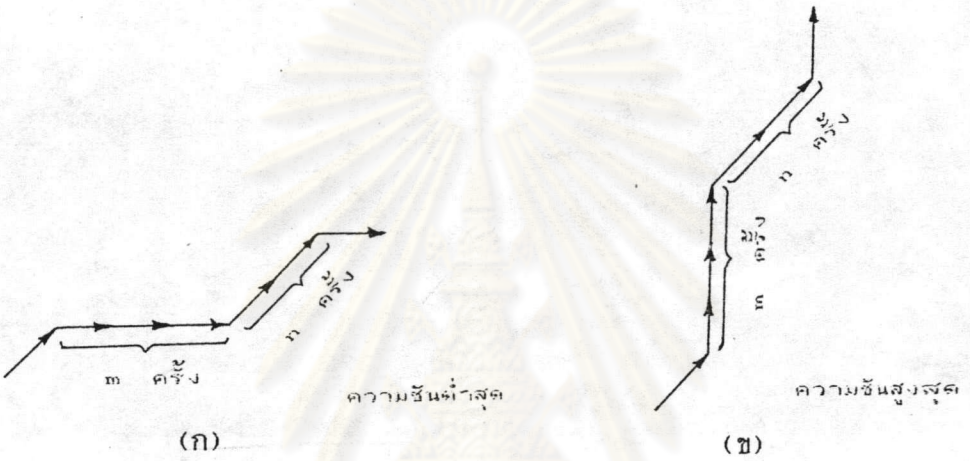
ง. เงื่อนไขการปรับขอบเขต (Boundary Adjustment)

$$|i(k) - j(k)| \leq r \qquad \text{.....(3.1.8)}$$

โดยที่ r เป็นความกว้างของหน้าต่าง ซึ่งจะมีค่าเป็นเลขจำนวนเต็มบวก

จ. เงื่อนไขการเปลี่ยนความชัน (Slope Constraint Condition) เนื่องจากความแปรปวนของความยาวในคำพูด สามารถเกิดขึ้นได้มาก เช่น เมื่อเราเปรียบเทียบคำพูด ก ซึ่งสั้นมากเมื่อเทียบกับคำพูด ข ในกรณีนี้จะทำให้ได้ฟังก์ชันแวร์ปปิ้งที่มีค่าผิดเพี้ยนไปมาก ดังนั้นจึงมีการกำหนดให้มีเงื่อนไขของการเปลี่ยนความชันขึ้น โดยเงื่อนไขนี้จะกำหนดให้จุด c(k) สามารถเคลื่อนที่ไปตามแกน i หรือ j โดยไม่เปลี่ยนทิศทางได้ m ครั้ง จากนั้นจุด c(k) จะไม่สามารถเคลื่อนที่ไปในทิศทางเดิมได้อย่างน้อย n ครั้ง ดังแสดงในรูปที่ (3.2.2ก) และ (3.2.2ข) เงื่อนไขนี้จะถูกเขียนออกมาในรูปของ

$$P = n / m \qquad .....(3.2.9)$$



รูปที่ 3.2.2 เงื่อนไขการเปลี่ยนความชันของฟังก์ชันแรปปิ้ง
(Sakoe H., 1978)

จากสมการที่ (3.2.5) ถ้าเราให้ตัวหาร ซึ่งเรียกว่าสัมประสิทธิ์การปรับตัว (Normalized Coefficient) ไม่ขึ้นอยู่กับฟังก์ชันแรปปิ้ง จะทำให้เขียนสมการนี้ใหม่ได้เป็น

$$D(A,B) = (1/N) \min_{F} \left[\sum_{k=1}^{K} d(c(k)).w(k) \right] \qquad .....(3.2.10)$$

โดย

$$N = \sum_{k=1}^{K} w(k) \qquad .....(3.2.11)$$

เพื่อที่จะทำให้สามารถแก้สมการที่ (3.2.10) ได้ง่ายขึ้น เราจึงมีการกำหนดค่าของ w(k) ซึ่งเราเรียกว่า สัมประสิทธิ์น้ำหนักแบ่งออกได้เป็น 2 แบบ คือ

ก. แบบแกนเวลาสมมาตร (Symmetric) ในแบบนี้แกนเวลาทั้งสอง จะถูกเปลี่ยนรูปให้อยู่บนแกนชั่วคราว ซึ่งกำหนดเป็นแกนร่วม ดังนั้น ผลรวมของสัมประสิทธิ์น้ำหนักก็คือผลรวมของแกนร่วมนั่นเอง (l=i+j)

$$w(k) = (i(k)-i(k-1)) + (j(k)-j(k-1)) \quad .....(3.2.12)$$

จะได้ว่า $N = I + J$ .....(3.2.13)

โดย I และ J เป็นความยาวของสัญญาณ A และ B ตามลำดับ

ข. แบบแกนเวลาไม่สมมาตร (Asymmertic) ในแบบนี้แกนเวลาจะถูกเปลี่ยนรูปจากแกนเวลาหนึ่ง ไปอยู่บนอีกแกนหนึ่ง ดังนั้นผลรวมของสัมประสิทธิ์น้ำหนัก ก็คือผลรวมของแกน i หรือ j เท่านั้น

$$w(k) = i(k) - i(k-1) \quad .....(3.2.14)$$

จะได้ว่า $N = I$ .....(3.2.15)

หรือ $w(k) = j(k) - j(k-1)$ จะได้ว่า $N = J$

3.2.2 <u>สมการไดนามิคโปรแกรมมิ่ง</u> (Sakoe H., 1978)

จากสมการที่ 3.2.10 สามารถแก้ได้โดยอาศัยทฤษฎีของไดนามิคโปรแกรมมิ่ง ดังนี้

เงื่อนไขเริ่มต้น

$$g_1(c(1)) = d(c(1)).w(1) \quad .....(3.2.16)$$

สมการไดนามิคโปรแกรมมิ่ง

$$g_k(c(k)) = \min_{c(k-1)} [g_{k-1}(c(k-1))+d(c(k)).w(k)] \quad .....(3.2.17)$$

ระยะทางการปรับตัวตามแกนเวลา

$$D(A,B) = (1/N)\, g_K(c(K)) \qquad .....(3.2.18)$$

จากสมการทั้งหมดนี้ สมมุติให้ $c(0) = (0,0)$ และคิดสัมประสิทธิ์น้ำหนักแบบสมมาตร $w(1) = 2$ แทนค่าสมการ (3.2.12) ลงในสมการ (3.2.17) จะทำให้สามารถหาค่าต่าง ๆ ได้ เช่น ถ้ากำหนดให้เงื่อนไขของการเปลี่ยนความชัน $P = 0$ จะได้ว่า

เงื่อนไขเริ่มต้น

$$g(1,1) = 2d(1,1) \qquad .....(3.2.19)$$

สมการไดนามิคโปรแกรมมิ่ง

$$g(i,j) = \min \begin{cases} g(i,j-1) + d(i,j) \\ g(i-1,j-1) + 2d(i,j) \\ g(i-1,j) + d(i,j) \end{cases} \qquad .....(3.2.20)$$

ระยะทางการปรับตัวตามแกนเวลา

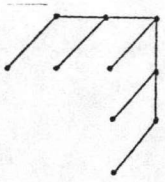
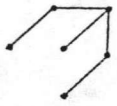
$$D(A,B) = (1/N)\, g(I,J) \quad \text{โดย } N = I + J \qquad .....(3.2.21)$$

ในตารางที่ 3.2.1 จะสรุปให้เห็นถึงสมการไดนามิคโปรแกรมมิ่งในลักษณะต่าง ๆ ทั้งแบบแกนเวลาสมมาตร และแกนเวลาไม่สมมาตร กับการเปลี่ยนแปลงเงื่อนไขความชัน

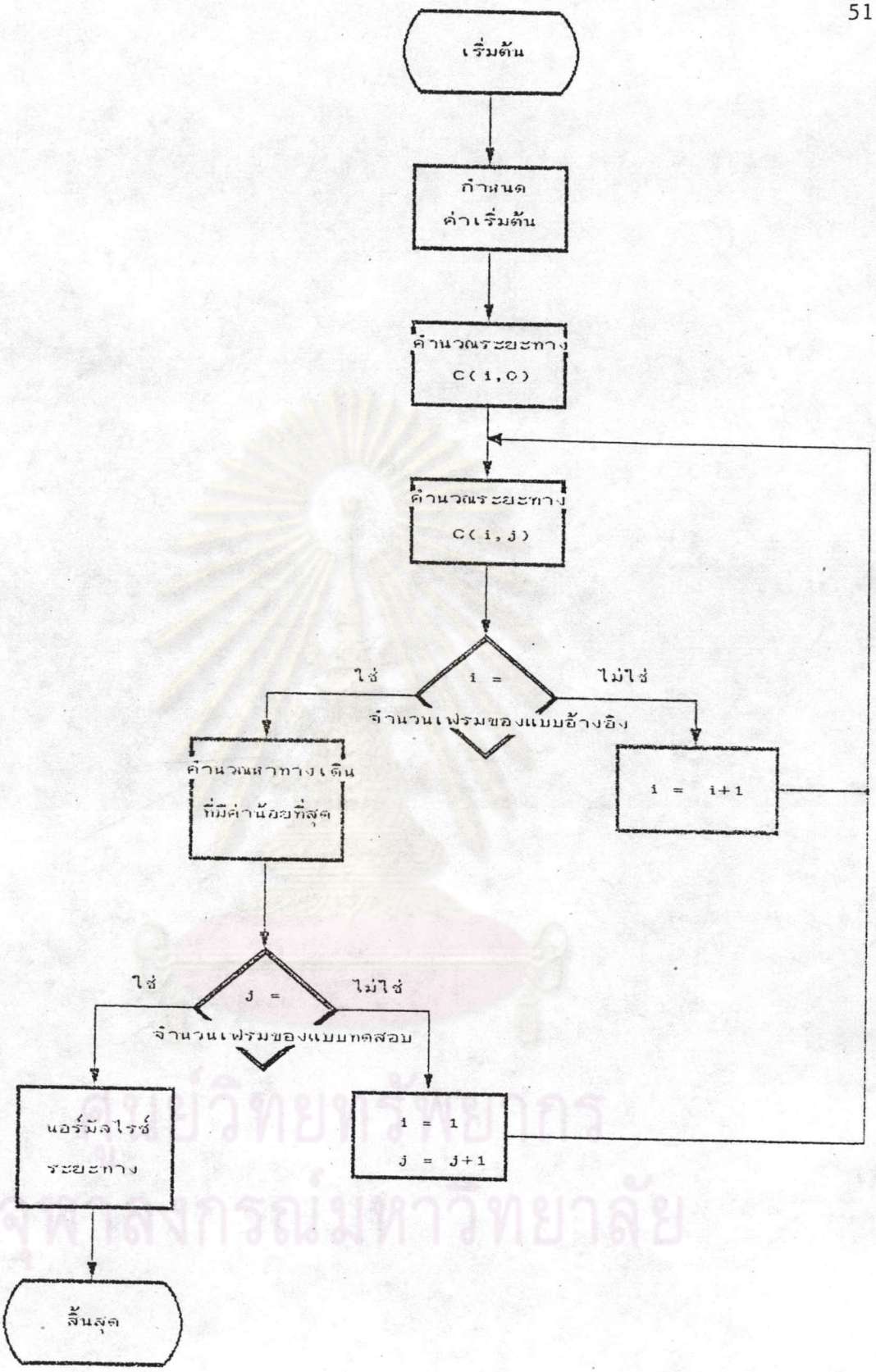
ในรูปที่ 3.2.3 จะแสดงผังของโปรแกรมย่อย DYNAMIC ซึ่งเป็นส่วนที่ใช้ในการเปรียบเทียบพารามิเตอร์ของสัญญาณเสียงพูด ระหว่างสัญญาณอ้างอิง กับ สัญญาณทดสอบ

3.3 กฎการตัดสินใจ (Decision Rules)

ผลที่ได้จากการเปรียบเทียบแบบไดนามิคโปรแกรมมิ่ง คือระยะทางระหว่างแบบทดสอบกับแบบอ้างอิงแต่ละแบบ ระยะทางที่ได้ทั้งหมดจะนำมาผ่านขั้นตอนการตัดสินใจ เพื่อหาผลลัพธ์ของการรับรู้ กฎการตัดสินใจ (Decision Rules) ที่ใช้ในการรับรู้เสียงพูดมีอยู่ 3 แบบ คือ

P	แผนภาพแสดงทางเดิน	สมมาตร / ไม่สมมาตร	สมการไดนามิคโปรแกรมมิ่ง $g(i,j)$ =
0		สมมาตร	$\min \begin{cases} g(i,j-1)+d(i,j) \\ g(i-1,j-1)+2d(i,j) \\ g(i-1,j)+d(i,j) \end{cases}$
		ไม่สมมาตร	$\min \begin{cases} g(i,j-1) \\ g(i-1,j-1)+d(i,j) \\ g(i-1,j)+d(i,j) \end{cases}$
1/2		สมมาตร	$\min \begin{cases} g(i-1,j-3)+2d(i,j-2)+d(i,j-1)+d(i,j) \\ g(i-1,j-2)+2d(i,j-1)+d(i,j) \\ g(i-1,j-1)+2d(i,j) \\ g(i-2,j-1)+2d(i-1,j)+d(i,j) \\ g(i-3,j-1)+2d(i-2,j)+d(i-1,j)+d(i,j) \end{cases}$
		ไม่สมมาตร	$\min \begin{cases} g(i-1,j-3)+(d(i,j-2)+d(i,j-1)+d(i,j))/3 \\ g(i-1,j-2)+(d(i,j-1)+d(i,j))/2 \\ g(i-1,j-1)+d(i,j) \\ g(i-2,j-1)+d(i-1,j)+d(i,j) \\ g(i-3,j-1)+d(i-2,j)+d(i-1,j)+d(i,j) \end{cases}$
1		สมมาตร	$\min \begin{cases} g(i-1,j-2)+2d(i,j-1)+d(i,j) \\ g(i-1,j-1)+2d(i,j) \\ g(i-2,j-1)+2d(i-1,j)+d(i,j) \end{cases}$
		ไม่สมมาตร	$\min \begin{cases} g(i-1,j-2)+(d(i,j-1)+d(i,j))/2 \\ g(i-1,j-1)+d(i,j) \\ g(i-2,j-1)+d(i-1,j)+d(i,j) \end{cases}$
2		สมมาตร	$\min \begin{cases} g(i-2,j-3)+2d(i-1,j-2)+2d(i,j-1)+d(i,j) \\ g(i-1,j-1)+2d(i,j) \\ g(i-3,j-2)+2d(i-2,j-1)+2d(i-1,j)+d(i,j) \end{cases}$
		ไม่สมมาตร	$\min \begin{cases} g(i-2,j-3)+2(d(i-1,j-2)+d(i,j-1)+d(i,j))/3 \\ g(i-1,j-1)+d(i,j) \\ g(i-3,j-2)+d(i-2,j-1)+d(i-1,j)+d(i,j) \end{cases}$

ตารางที่ 3.2.1 แสดงสมการไดนามิคโปรแกรมมิ่งต่าง ๆ เมื่อเงื่อนไขการเปลี่ยนความชันเปลี่ยนไป (Sakoe H., 1978)



รูปที่ 3.2.3 แสดงผังของโปรแกรม DYNAMIC

3.3.1 กฎ Nearest Neighbor (NN)

ผลลัพธ์ของการรับรู้จากกฎการตัดสินใจนี้ ได้แก่ แบบอ้างอิงที่มีระยะทางจากแบบทดสอบน้อยที่สุด กฎนี้เหมาะสำหรับระบบการรับรู้เสียงพูดแบบบุคคลเดี่ยว

3.3.2 กฎ K-Nearest Neighbor (KNN)

เป็นกฎที่ดัดแปลงมาจากกฎ NN อีกที่หนึ่ง เพื่อให้เหมาะสมกับระบบการรับรู้เสียงพูดแบบต่างบุคคล ซึ่งมีแบบอ้างอิงในแต่ละคำมากกว่า 1 แบบ กฎนี้ได้แบ่งออกเป็น 2 แบบ คือ

ก. แบบที่ 1 จะทำการหาแบบอ้างอิงจำนวน K แบบ ที่มีระยะทางห่างจากแบบทดสอบน้อยที่สุด ผลลัพธ์ของการรับรู้จะได้จาก แบบอ้างอิงที่มีจำนวนมากที่สุดในจำนวน K แบบ ที่ได้รับการเลือกมา

ข. แบบที่ 2 จะทำการหาแบบอ้างอิงจำนวน K แบบในแต่ละคำ โดยที่ทั้ง K แบบมีระยะทางห่างจากแบบทดสอบน้อยที่สุด ผลลัพธ์ของการเรียนรู้จะได้จาก แบบอ้างอิงที่มีค่าเฉลี่ยของระยะทางในจำนวน K แบบที่ได้รับเลือกมาน้อยที่สุด วิธีการนี้จะให้ผลลัพธ์ที่ถูกต้องมากขึ้นโดยกำหนดค่า K เป็น 2 หรือ 3 เมื่อเทียบกับกฎ NN (Rabiner L.R., 1979) อย่างไรก็ตามวิธีนี้จะทำให้เวลาที่ใช้ในการรับรู้เพิ่มขึ้นด้วย

ในระบบการรับรู้เสียงพูดทั่วไป จะมีการกำหนดค่าระยะทางมากที่สุดที่ยอมรับได้ (Reject Value) เพื่อเพิ่มความถูกต้องของการรับรู้ในกรณีที่ผลลัพธ์ที่ได้จากการตัดสินใจมีค่าใกล้เคียงกันมาก วิธีการกำหนดค่านี้ต้องใช้ความระมัดระวังเป็นอย่างมาก เนื่องจากถ้าเรากำหนดค่านี้สูงเกินไป หรือไม่ได้กำหนด โอกาสที่จะทำให้ผลลัพธ์ของการรับรู้ผิดพลาดไปจะมีได้มาก แต่ถ้าเรากำหนดค่านี้ต่ำเกินไป จะทำให้ผลลัพธ์ของการรับรู้สำหรับแบบอ้างอิงที่ถูกต้องเกิดขึ้นได้ยาก ดังนั้นโดยทั่วไปค่าระยะทางนี้มักจะได้มาจากการทดลอง เพื่อหาค่าที่เหมาะสมสำหรับกลุ่มของแบบอ้างอิงหนึ่ง ๆ เท่านั้น (Holtzman J., 1984)