

บทที่ 3

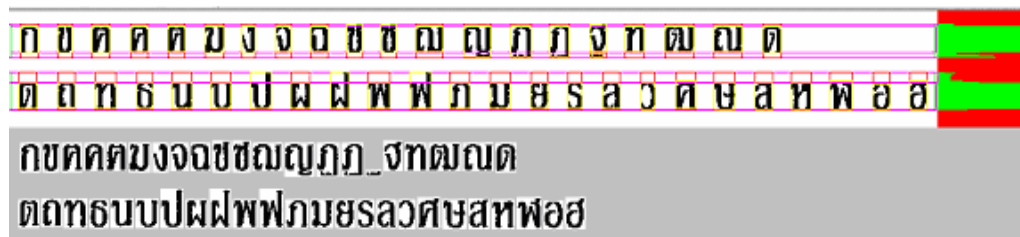
การรู้จำรูปแบบแนวทางใหม่โดยใช้โครงสกรอรีเลชั่น

3.1 กระบวนการก่อนการรู้จำ

กระบวนการก่อนการรู้จำนับว่าเป็นกระบวนการแรกในระบบการรู้จำที่ทำการประมวลผลเบื้องต้นเกี่ยวกับภาพตัวอักษรเพื่อให้ได้ข้อมูลเป็นอินพุตส่งต่อเข้าสู่กระบวนการรู้จำ โดยกระบวนการก่อนการรู้จำนี้จะประกอบไปด้วย 5 ขั้นตอนดังต่อไปนี้

ขั้นตอนที่ 1

เริ่มต้นจะมีการสแกนภาพเอกสารตัวอักษรเข้ามาเป็นข้อมูลอินพุตของระบบการรู้จำ ซึ่งจะเป็นภาพเอกสารที่ประกอบไปด้วยบรรทัดหลายบรรทัด ในขั้นตอนนี้จะทำการแยกบรรทัดและแยกตัวอักษรแต่ละตัวออกจากบรรทัด โดยใช้วิธีการของฮิสโตแกรม (Histogram) และการหาขอบภาพ (Contour Algorithm) ตามรูปที่ 3.1 ซึ่งในขั้นตอนนี้จะได้ตำแหน่งและขนาดของภาพตัวอักษรแต่ละตัวออกมาเรียกว่า “กรอบตัวอักษร”

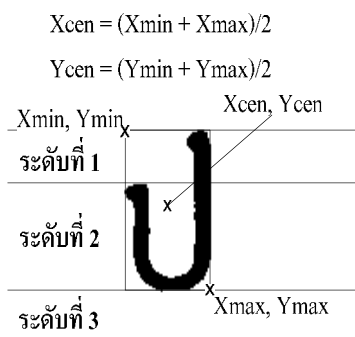


รูปที่ 3.1 การแยกบรรทัดและตัวอักษรโดยใช้ฮิสโตแกรม และการหาขอบภาพ

จากนั้นจะทำการจัดกลุ่มของกรอบตัวอักษรที่ได้เป็น 3 กลุ่มคือกลุ่มตัวอักษรที่มีจุดศูนย์กลางของกรอบตัวอักษรอยู่ในระดับที่ 1, 2, และ 3 ตามลำดับยกตัวอย่างเช่นรูปที่ 3.2 เป็นการแบ่งกลุ่มระดับโดยการหาจุดศูนย์กลางของกรอบภาพตัวอักษร “ป” และจัดอยู่ในกลุ่มระดับที่ 2 สำหรับกลุ่มตัวอักษรระดับที่ 1, 2, และ 3 นี้จะเรียกว่าเป็นกลุ่มระดับบน กลาง และล่างตามลำดับตามตารางที่ 3.1 พบว่าพยัญชนะไทยและตัวเลขอารบิกนั้นอยู่ในระดับกลางเหมือนกันทั้งหมด ส่วนวรรณยุกต์ทั้งหมดก็อยู่ในระดับบน แต่สำหรับสระไทยนั้นมีปรากฏอยู่ทั้ง 3 ระดับ

ขั้นตอนที่ 2

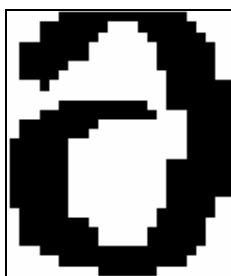
จากนั้นจะนำเอารอบภาพตัวอักษรที่ได้มาดังรูปที่ 3.3 มาทำการปรับขนาดให้เท่ากับ 25x29 Pixels ด้วยวิธีการ Resize โดยอาศัย Interpolation/resampling filter แบบ Hermite Filter แล้วทำการลดขนาดของเส้นตัวอักษรจากการหาจุดกึ่งกลางของวงกลมที่มีจุดปลายเส้นผ่านศูนย์กลางทั้งสองจุดอยู่บนจุดปลายของขอบตัวอักษรตามรูปที่ 3.4 ทั้งนี้เพื่อปรับภาพตัวอักษรให้อยู่ในรูปแบบของภาพเส้นของตัวอักษรดังรูปที่ 3.5 โดยจะไม่มีเปลี่ยนแปลงโครงสร้างพิเศษต่าง ๆ ของภาพตัวอักษรด้วยวิธีการ Skeletonization โดยใช้ Hilditch's Algorithm



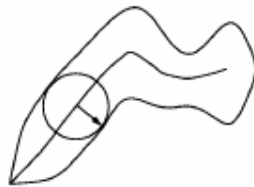
รูปที่ 3.2 การแบ่งกลุ่มระดับโดยการหาจุดศูนย์กลางของกรอบภาพตัวอักษร

ตารางที่ 3.1 กลุ่มระดับตัวอักษรภาษาไทยที่แบ่งออกเป็น 3 กลุ่ม

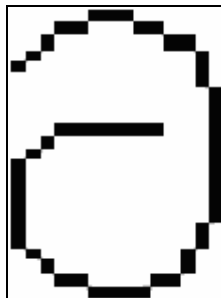
กลุ่มระดับบน 12 ตัว	๐ ๑ ๒ ๓ ๔ ๕ ๖ ๗ ๘ ๙ + , . ' °
กลุ่มระดับกลาง 66 ตัว	ก ข ช ด ต ข ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ท ฒ ฌ ด ต ก ท ธ น บ ป ผ ฝ พ ฟ ม ย ร ล ว ต ษ ส ห ฬ อ ฮ ฤ ฦ ๐ ๑ ๒ ๓ ๔ ๕ ๖ ๗ ๘ ๙ ะ ำ เ แ โ ใ ไ ใ ๆ ๆ
กลุ่มระดับล่าง 2 ตัว	๐ ๑



รูปที่ 3.3 กรอบภาพตัวอักษร “อ”



รูปที่ 3.4 การหาจุดกึ่งกลางของวงกลมที่มีจุดปลายเส้นผ่านศูนย์กลางทั้งสองจุดอยู่บนจุดปลายของ
ขอบตัวอักษรสำหรับการทำ Skeleton



รูปที่ 3.5 รูปแบบของภาพลายเส้นตัวอักษรหลังจากการทำ Skeleton

ประโยชน์ของการทำ Skeleton ตัวอักษรเพื่อจุดประสงค์ของการได้มาซึ่งตำแหน่งของพิกเซลที่ต่อเนื่องกันในรูปแบบของตัวอักษรหนึ่ง ๆ ซึ่งยังคงความเป็นรูปแบบลายเส้นของตัวอักษรหลังจากที่มีการทำ Skeleton และยังสามารถที่จะส่งเข้าสู่กระบวนการรู้จำได้ หรืออาจจะกล่าวได้ว่ายังสามารถรู้จำออกมาได้ว่าเป็นรูปแบบของตัวอักษรอะไร อัลกอริทึมที่นำมาใช้ก็คือ Hilditch's Algorithm เริ่มจากการพิจารณา 8-neighborhood ของพิกเซล P1 ตามรูปที่ 3.6 จากนั้นจะเลือกที่จะพิจารณาให้ P1 นั้นเป็นพิกเซลที่ผ่านการ Skeleton หรือไม่ หรือว่าจะลบพิกเซลนั้นทิ้งไปถ้าพิกเซลนั้นไม่นับว่าเป็น Skeleton Pixel จะมีการกำหนดฟังก์ชัน 2 ฟังก์ชันที่ใช้ในการพิจารณาคือ

1. $B(P1)$ เป็นฟังก์ชันที่ใช้ับจำนวนของ non-zero neighbors ของพิกเซล P1
คือ $B(P1) = P2 + P3 + P4 + P5 + P6 + P7 + P8 + P9$
2. $A(P1)$ เป็นฟังก์ชันที่ใช้ับจำนวนของการเปลี่ยนแปลงค่า 0-1 Patterns ในลำดับของ P2, P3, P4, P5, P6, P7, P8, P9, P2

Hilditch's Algorithm นั้นจะมีการตรวจสอบเงื่อนไขในการนำมาพิจารณาและหากพบว่า เป็นไปตามเงื่อนไขก็จะสามารถทำการลบพิกเซลที่ตำแหน่ง P1 นั้นทิ้งไปได้ แต่หากไม่ผ่านเงื่อนไขข้อใดข้อหนึ่งหรือมากกว่าหนึ่งเงื่อนไขแสดงว่าผลลัพธ์เงื่อนไขไม่เป็นจริง ก็จะกำหนดให้พิกเซลที่ตำแหน่ง P1 เป็น Skeleton Pixel และพิจารณาต่อไปเรื่อย ๆ จนกระทั่งไม่มีพิกเซลที่สามารถจะลบทิ้งออกไปได้อีกแล้ว ยกตัวอย่างผลลัพธ์ที่ได้ออกมาหลังจากที่ผ่านการทำ Skeleton ดังรูปที่ 3.5 ข้างต้น เงื่อนไขดังกล่าวมีทั้งหมด 4 ข้อได้แก่

1. เงื่อนไขข้อที่ 1: $2 \leq B(P1) \leq 6$ จะเป็นการทดสอบจุดปลายและจุดบริเวณขอบเขตของรูปแบบตัวอักษร ซึ่งจากรูปที่ 3.7 ไม่เป็นไปตามเงื่อนไขข้อที่ 1 แสดงให้เห็นว่าถ้า $B(P1) = 1$ แล้วจุดพิกเซล P1 จะเป็นจุดปลายของการทำ Skeleton และหาก $B(P1) = 0$ แสดงว่าจุดพิกเซล P1 นั้นเป็นจุดแยกออกมาจุดหนึ่งซึ่งไม่ควรที่จะลบทิ้ง และหาก $B(P1) = 7$ แสดงว่าจุดพิกเซล P1 นั้นเป็นจุดที่เส้นขอบของตัวอักษรที่ไม่ควรลบทิ้งด้วยเช่นกัน

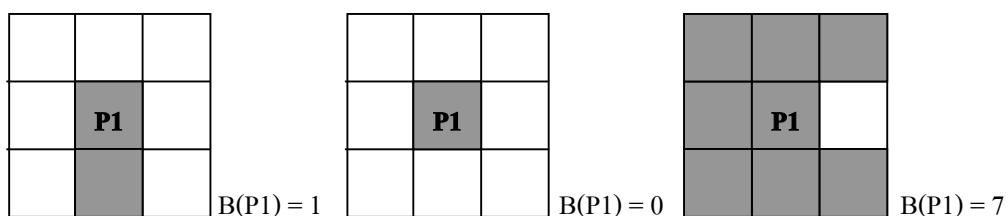
2. เงื่อนไขข้อที่ 2: $A(P1) = 1$ จะเป็นการทดสอบความต่อเนื่องของพิกเซลในรูปแบบตัวอักษร ซึ่งจากรูปที่ 3.8 ไม่เป็นไปตามเงื่อนไขข้อที่ 2 แสดงให้เห็นว่าถ้า $A(P1) > 1$ แล้วหากทำการลบจุดพิกเซล P1 ทิ้งไป จะทำให้รูปแบบของเส้นตัวอักษรไม่ต่อเนื่องกัน

3. เงื่อนไขข้อที่ 3: $P2 \times P4 \times P8 = 0$ หรือ $A(P2) \neq 1$ จะเป็นการทดสอบเพื่อให้แน่ใจว่า Pattern ที่กว้าง 2 พิกเซลติดกันในแนวตั้งจะไม่ถูกตัดทิ้งหายไป ซึ่งจากรูปที่ 3.9 ไม่เป็นไปตามเงื่อนไขข้อที่ 3

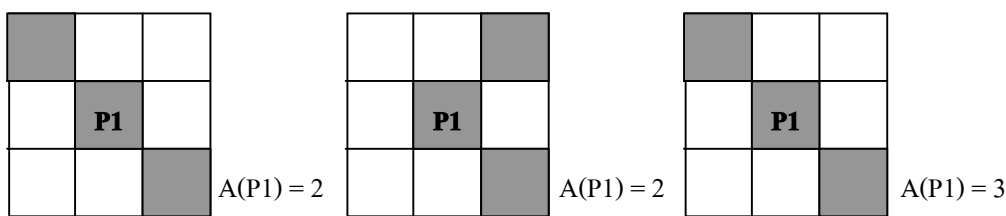
4. เงื่อนไขข้อที่ 4: $P2 \times P4 \times P6 = 0$ หรือ $A(P4) \neq 1$ จะเป็นการทดสอบเพื่อให้แน่ใจว่า Pattern ที่กว้าง 2 พิกเซลติดกันในแนวนอนจะไม่ถูกตัดทิ้งหายไป ซึ่งจากรูปที่ 3.10 ไม่เป็นไปตามเงื่อนไขข้อที่ 4

P9	P2	P3
P8	P1	P4
P7	P6	P5

รูปที่ 3.6 ภาพ 8-neighborhood ของพิกเซล P1



รูปที่ 3.7 แสดงรูปภาพที่จะทำให้เงื่อนไขข้อที่ 1 เป็นเท็จ



รูปที่ 3.8 แสดงรูปภาพที่จะทำให้เงื่อนไขข้อที่ 2 เป็นเท็จ

		P12
	P2	
	P1	P4

$$A(P2) \neq 1$$

	P1	

$$P2 \times P4 \times P8 \neq 0 \text{ และ } A(P2) = 1$$

P9		P3
	P1	
	P6	P5

$$P2 \times P4 \times P8 = 0$$

รูปที่ 3.9 แสดงรูปภาพที่จะทำให้เงื่อนไขข้อที่ 3 เป็นเท็จ

			P10
	P1	P4	
			P12

$$A(P4) \neq 1$$

	P1	

$$P2 \times P4 \times P6 \neq 0 \text{ และ } A(P4) = 1$$

P9		
P8	P1	
P7		P5

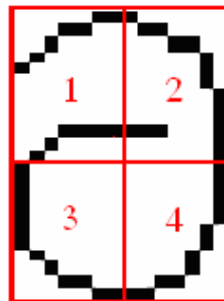
$$P2 \times P4 \times P6 = 0$$

รูปที่ 3.10 แสดงรูปภาพที่จะทำให้เงื่อนไขข้อที่ 4 เป็นเท็จ

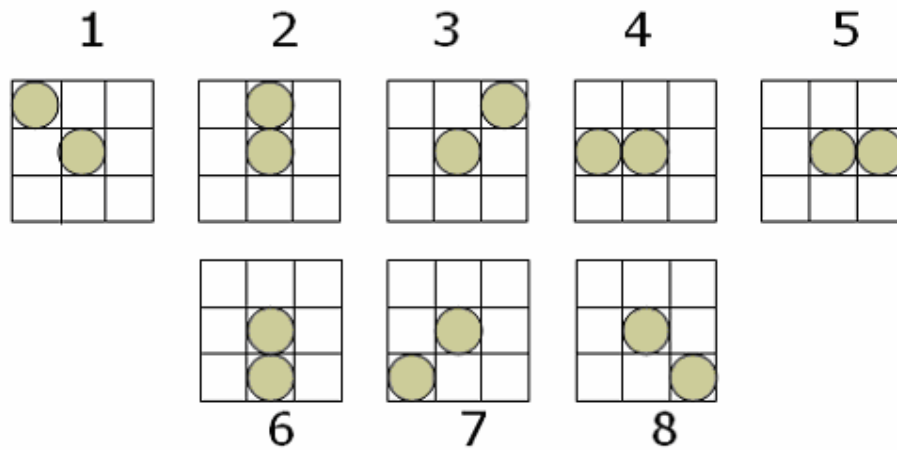
ขั้นตอนที่ 3

ทำการหาจุด Endpoint ของแต่ละกรอบภาพตัวอักษรและกำหนดจุดเริ่มต้นและจุดสิ้นสุดของเส้นภาพตัวอักษรซึ่งประกอบไปด้วย 4 ขั้นตอนดังต่อไปนี้

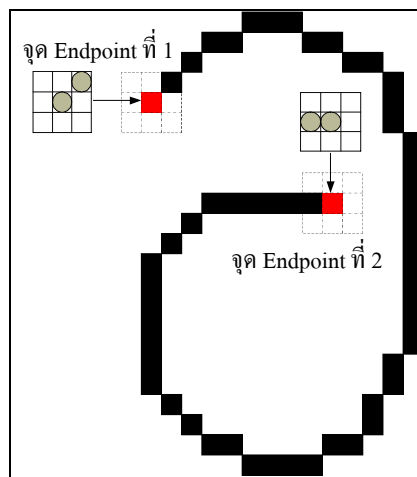
ขั้นตอนที่ 3.1 แบ่งกรอบภาพตัวอักษรออกเป็น 4 Quadrants แล้วกำหนดลำดับของกรอบภาพตัวอักษรที่ถูกแบ่งจากซ้ายไปขวา และบนลงล่างคือ 1, 2, 3 และ 4 ตามรูปที่ 3.11



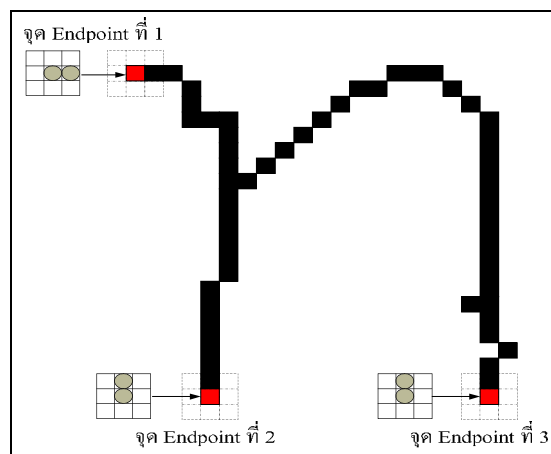
รูปที่ 3.11 การแบ่งกรอบภาพตัวอักษรออกเป็น 4 Quadrants



รูปที่ 3.12 เมตริกขนาด 3x3 Pixels ที่ใช้ในการหาจุด Endpoint มีทั้งหมด 8 รูปแบบ



รูปที่ 3.13 กรอบภาพตัวอักษรที่มีจุด Endpoint 2 จุด

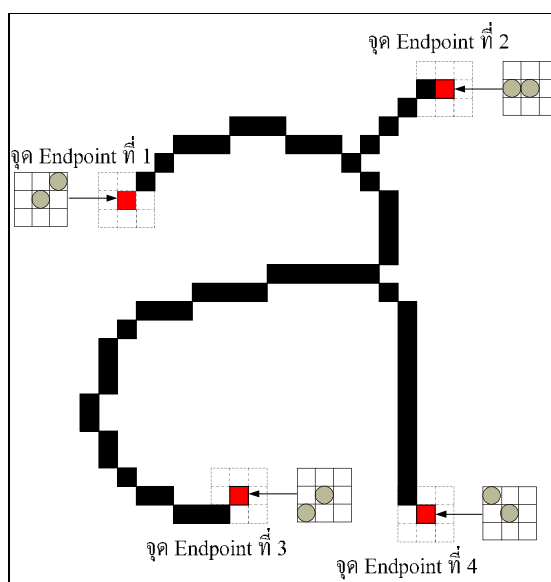


รูปที่ 3.14 กรอบภาพตัวอักษรที่มีจุด Endpoint 3 จุด

ขั้นตอนที่ 3.2 กำหนดเมตริกขนาด 3x3 Pixels สำหรับใช้ในการหาจุด Endpoint ดังรูปที่ 3.12 ซึ่งจะมียู่ทั้งหมด 8 รูปแบบ

ขั้นตอนที่ 3.3 นำเอารูปแบบของเส้นภาพตัวอักษรในกรอบภาพที่ผ่านการทำ Skeleton แล้ว มาเปรียบเทียบกับจุดศูนย์กลางของรูปแบบเมตริก 3x3 ที่มีอยู่ทั้งหมด 8 รูปแบบที่ใช้ในการหาจุด Endpoint และเมื่อได้ Pattern ที่ตรงกันแล้วจะกำหนดให้ตำแหน่งจุดที่อยู่บนเส้นภาพตัวอักษรนั้น เป็นจุด Endpoint ของลายเส้นตัวอักษร ยกตัวอย่างเช่นรูปที่ 3.13 แสดงกรอบภาพตัวอักษรที่มีจุด Endpoint 2 จุด หรือรูปที่ 3.14 แสดงกรอบภาพตัวอักษรที่มีจุด Endpoint 3 จุด หรือรูปที่ 3.15 แสดงกรอบภาพตัวอักษรที่มีจุด Endpoint 4 จุด

ขั้นตอนที่ 3.4 จากนั้นจะทำการกำหนดจุดเริ่มต้นและจุดสิ้นสุดตามลำดับความสำคัญโดย เริ่มจากจุด Endpoint ที่อยู่ที่จุดศูนย์กลางของกรอบภาพตัวอักษร จากนั้นตามด้วยจุด Endpoint ที่อยู่ใน Quadrant ที่ 1, 2, 3 และ 4 ตามลำดับดังตัวอย่างรูปที่ 3.16 แสดงจุดเริ่มต้นและจุดสิ้นสุดของตัวอักษร “อ” รูปที่ 3.17 แสดงจุดเริ่มต้นและจุดสิ้นสุดของตัวอักษร “ท” และรูปที่ 3.18 แสดงจุดเริ่มต้นและจุดสิ้นสุดของตัวอักษร “ส”

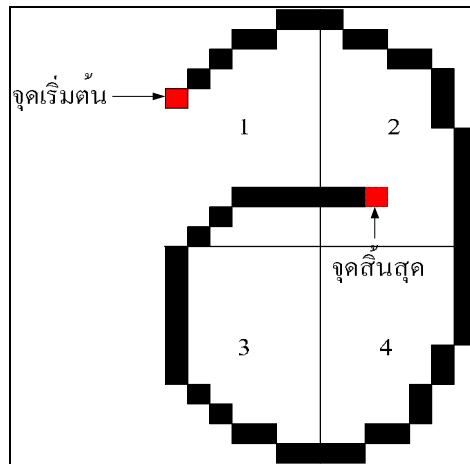


รูปที่ 3.15 กรอบภาพตัวอักษรที่มีจุด Endpoint 4 จุด

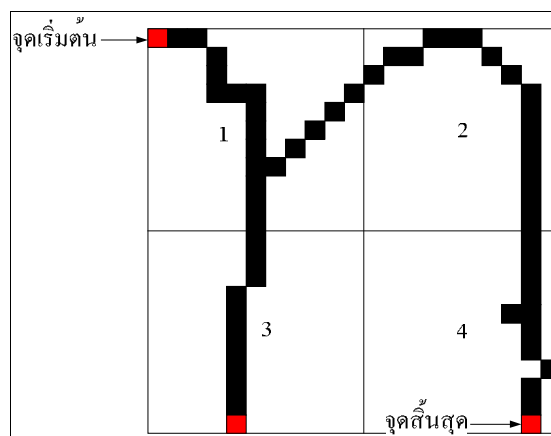
ขั้นตอนที่ 4

ทำการกำหนดรหัสแทนเส้นตัวอักษรซึ่งประกอบไปด้วย 4 ขั้นตอนดังต่อไปนี้

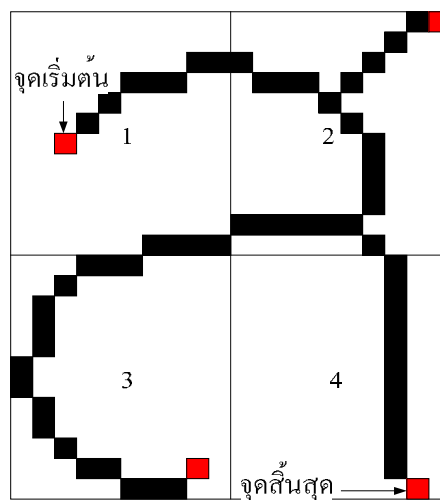
ขั้นตอนที่ 4.1 กำหนดรหัสแทนเส้นตัวอักษรทั้งหมด 9 รหัสตามรูปที่ 3.19 และรหัสแทนเส้นตัวอักษรหรือเรียกว่า “Directional Code” นี้จะแสดงในสมการที่ 3.1 โดยจะพิจารณามุมระหว่างเส้นตัวอักษรกับเส้น Base line $\theta = 0$



รูปที่ 3.16 แสดงจุดเริ่มต้นและจุดสิ้นสุดของตัวอักษร “อ”



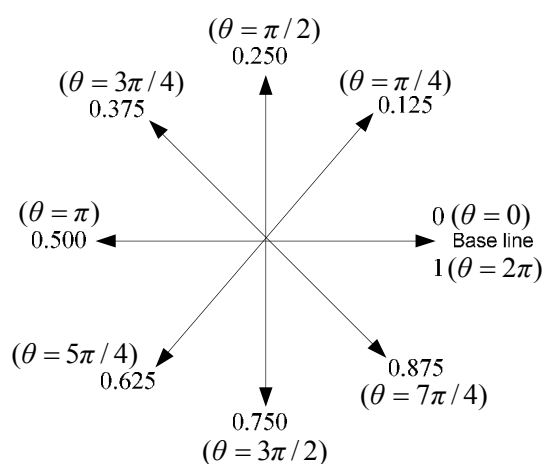
รูปที่ 3.17 แสดงจุดเริ่มต้นและจุดสิ้นสุดของตัวอักษร “ท”



รูปที่ 3.18 แสดงจุดเริ่มต้นและจุดสิ้นสุดของตัวอักษร “ส”

ขั้นตอนที่ 4.2 แบ่งภาพกรอบตัวอักษรที่มีการกำหนดจุดเริ่มต้นและจุดสิ้นสุดของเส้นตัวอักษรออกเป็นสามส่วนในแนวตั้งและสามส่วนในแนวนอนก็จะได้กรอบย่อยที่ถูกแบ่งออกเป็น 9 กรอบย่อย จากนั้นทำการกำหนดให้กรอบย่อยที่มีจุดเริ่มต้นของเส้นตัวอักษรนั้นเป็นกรอบย่อยที่ 1 และกำหนดกรอบย่อยต่อไปที่มีเส้นตัวอักษรเชื่อมต่อในแนวทิศทางของตัวอักษรให้เป็นกรอบย่อยที่ 2, 3, 4, 5, 6, 7, 8 และ 9 ตามลำดับ ยกตัวอย่างตามรูปที่ 3.20 แสดงทิศทางของเส้นตัวอักษร “อ” ตามการจัดเรียงกรอบย่อยของภาพตัวอักษรตามลำดับของทิศทางเส้นตัวอักษรเพื่อกำหนดรหัสแทนเส้นตัวอักษร นอกจากนี้หากกรอบตัวอักษรนั้นมีจุดเริ่มต้นและจุดสิ้นสุดแล้วยังมีจุด Endpoint อื่น ๆ อยู่อีกด้วย จะทำการกำหนดลำดับกรอบย่อยเริ่มจากจุดเริ่มต้นไปยังจุดสิ้นสุด จากนั้นจะพิจารณาจุด Endpoint ต่อไปให้เป็นจุดเริ่มต้นในการกำหนดทิศทางตัวอักษรและกำหนดลำดับกรอบย่อยตัวอักษรต่อไปเรื่อย ๆ จนครบ 9 กรอบย่อยยกตัวอย่างตามรูปที่ 3.21 แสดงทิศทางของเส้นตัวอักษร “ท” ตามการจัดเรียงกรอบย่อยของภาพตัวอักษรตามลำดับของทิศทางเส้นตัวอักษรเพื่อกำหนดรหัสแทนเส้นตัวอักษร และรูปที่ 3.22 แสดงทิศทางของเส้นตัวอักษร “ส” ตามการจัดเรียงกรอบย่อยของภาพตัวอักษรตามลำดับของทิศทางเส้นตัวอักษรเพื่อกำหนดรหัสแทนเส้นตัวอักษร

ขั้นตอนที่ 4.3 ค่าข้อมูลสำหรับในแต่ละกรอบย่อยตัวอักษรที่แบ่งออกเป็นทั้งหมด 9 กรอบย่อย จะแทนด้วยค่ารหัสแทนเส้นตัวอักษรโดยเริ่มจากจุดเริ่มต้นของเส้นทางการเดินไปตามเส้นตัวอักษรที่อ้างอิงทิศทางแต่ละพิกเซลหนึ่งไปยังอีกพิกเซลหนึ่งที่มีการเชื่อมต่อของเส้นตัวอักษรไปเรื่อย ๆ ในกรอบย่อยแต่ละกรอบตามลำดับของกรอบย่อยที่กำหนดไว้ ยกตัวอย่างเช่นการแทนค่ารหัสแทนเส้นทิศทางของตัวอักษร “อ” แสดงตามตารางที่ 3.2 และการแทนค่ารหัสแทนเส้นทิศทางของตัวอักษร “ท” แสดงตามตารางที่ 3.3 และการแทนค่ารหัสแทนเส้นทิศทางของตัวอักษร “ส” แสดงตามตารางที่ 3.4



รูปที่ 3.19 รหัสแทนเส้นตัวอักษรทั้งหมด 9 รหัส

$$\text{Directional Code} = \begin{cases} 0 & \rightarrow \theta = 0 \\ 0.125 & \rightarrow \theta = \pi/4 \\ 0.250 & \rightarrow \theta = \pi/2 \\ 0.375 & \rightarrow \theta = 3\pi/4 \\ 0.5 & \rightarrow \theta = \pi \\ 0.675 & \rightarrow \theta = 5\pi/4 \\ 0.750 & \rightarrow \theta = 3\pi/2 \\ 0.875 & \rightarrow \theta = 7\pi/4 \\ 1 & \rightarrow \theta = 2\pi \end{cases} \quad (3.1)$$

กำหนดให้ Directional Code แทนรหัสแทนเส้นตัวอักษร
 θ แทนมุมระหว่างเส้นตัวอักษรกับเส้น Base line $\theta = 0$

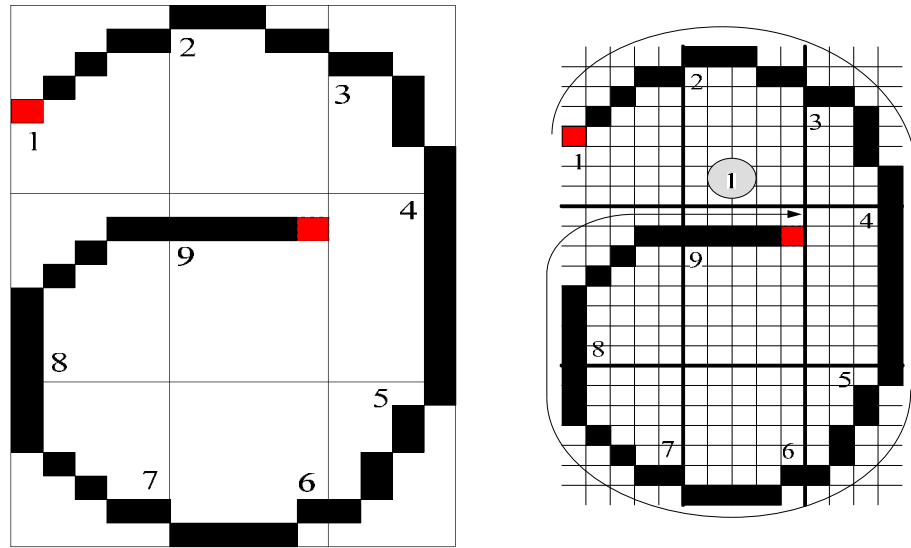
ขั้นตอนที่ 4.4 เป็นขั้นตอนการกำหนดค่าข้อมูลโดยพิจารณาค่ารหัสแทนเส้นตัวอักษรที่จะนำมาใช้คำนวณในฟังก์ชันครอสคอร์รีเลชัน เพื่อแก้ปัญหาที่อาจจะเกิดขึ้น โดยมีสาเหตุมาจากการเปรียบเทียบค่าความคล้ายคลึงกันระหว่างสองค่าชุดข้อมูลนั้นจำเป็นต้องกำหนดให้ชุดข้อมูลทั้งสองมีขนาดความยาวเท่ากัน และค่ารหัสแทนเส้นตัวอักษรที่มีค่าเป็น 0 ในทางทฤษฎีครอสคอร์รีเลชันนั้นจะหมายความว่าค่าข้อมูลที่เป็น 0 นั้นไม่มีความหมาย จึงต้องมีการแปลงค่าข้อมูลที่เป็นค่า 0 ดิจ ๆ กัน จากนั้นก็จะได้ออกมาเป็นค่า Input Vector สำหรับส่งต่อเข้าสู่กระบวนการรู้จำที่อาศัยฟังก์ชันครอสคอร์รีเลชัน จึงมีหลักในการพิจารณาดังต่อไปนี้

1. หากค่าความยาวของค่าข้อมูลไม่เท่ากันจะแทนค่าด้วยค่าสุดท้ายของชุดข้อมูลนั้น ๆ โดยที่ขนาดของ Input Vector จะมีขนาดเท่ากับขนาดของชุดข้อมูลย่อยที่มีขนาดยาวที่สุด
2. พิจารณาชุดข้อมูลที่มีค่า 0 ติดกันมากกว่า 2 ตัวขึ้นไป เช่น [... 0 0 0 ...] , [... 0 0 0 0 ...] และ [... 0 0 0 0 0 ...] จะทำการแปลงค่า 0 ระหว่างนั้นให้เป็น 1 จะทำให้ได้ Input Vector คือ [... 0 1 0 ...] , [... 0 1 1 0 ...] และ [... 0 1 1 1 0 ...] ตามลำดับเป็นต้น
3. ถ้าค่าข้อมูลมีค่าเป็น [0] หรือ [0 0] จะแทนค่าข้อมูลด้วย [0 1 0]
4. หากไม่มีค่าข้อมูลในบางกรอบส่วนย่อย จะกำหนดให้ Input Vector ที่กรอบย่อยนั้นมีค่าเป็น 0 ทั้งหมด โดยที่ขนาดของ Input Vector จะมีขนาดเท่ากับขนาดของชุดข้อมูลย่อยที่มีขนาดยาวที่สุด (ซึ่งจะไม่มีการแปลงค่า 0 ที่ติดกันไปเป็นค่า 1 ตามข้อที่ 2 ข้างต้น)

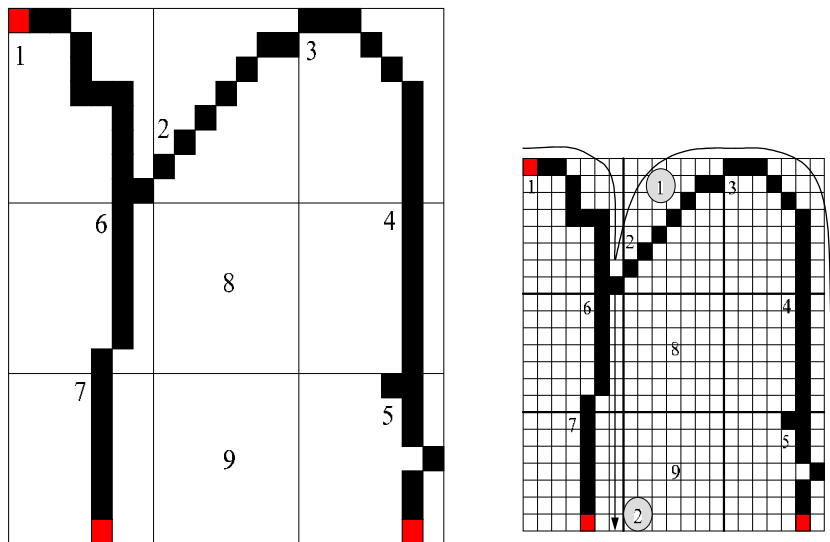
ขั้นตอนที่ 5

แนวคิดของวิธีใหม่นี้นำเสนอเริ่มต้นด้วยการจัดแบ่งแยกแยะข้อมูลภาพตัวอักษรออกเป็น 3 กลุ่มระดับตัวอักษร ได้แก่ กลุ่มระดับตัวอักษรบน กลุ่มระดับตัวอักษรกลาง และกลุ่มระดับตัว

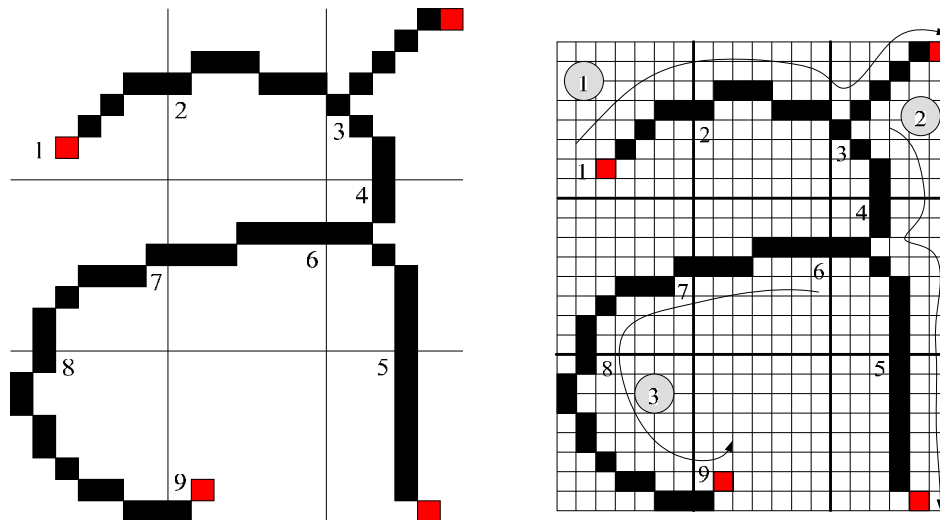
อักษรล่าง โดยแต่ละกลุ่มจะมีประเภทที่กำหนดไว้ตามกลุ่มตัวอักษรระดับบน กลาง และล่าง ตามลำดับแสดงดังรูปที่ 3.23 จากนั้นในแต่ละกลุ่มที่ถูกแบ่งจะส่งรูปแบบตัวอักษรเข้าไปสู่กระบวนการรู้จำแบบโครงข่ายประสาทเทียมตามวิธีการของฟีดฟอร์เวิร์ดแบบสี่ชั้น (Four Layers Feedforward Neural Network) ตามที่แสดงในรูปที่ 3.24



รูปที่ 3.20 แสดงทิศทางของเส้นตัวอักษร “อ” ตามการจัดเรียงกรอบย่อยของภาพตัวอักษรตามลำดับของทิศทางเส้นตัวอักษรเพื่อกำหนดรหัสแทนเส้นตัวอักษร



รูปที่ 3.21 แสดงทิศทางของเส้นตัวอักษร “ท” ตามการจัดเรียงกรอบย่อยของภาพตัวอักษรตามลำดับของทิศทางเส้นตัวอักษรเพื่อกำหนดรหัสแทนเส้นตัวอักษร



รูปที่ 3.22 แสดงทิศทางของเส้นตัวอักษร “ส” ตามการจัดเรียงกรอบย่อยของภาพตัวอักษรตามลำดับของทิศทางเส้นตัวอักษรเพื่อกำหนดรหัสแทนเส้นตัวอักษร

ตารางที่ 3.2 การแทนค่ารหัสแทนเส้นทิศทางของตัวอักษร “อ”

กรอบย่อยที่	ค่าข้อมูล	Input Vector มีขนาดเท่ากับ 8
1 (I^1)	0.125 0.125 0.125 0 0.125	0.125 0.125 0.125 0 0.125 0.125 0.125 0.125
2 (I^2)	0 0 0.875 1 0.875	0 0 0.875 1 0.875 0.875 0.875 0.875
3 (I^3)	1 0.875 0.75 0.75 0.875 0.75 0.75	1 0.875 0.75 0.75 0.875 0.75 0.75 0.75
4 (I^4)	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
5 (I^5)	0.625 0.75 0.625 0.75 0.625 0.5	0.625 0.75 0.625 0.75 0.625 0.5 0.5 0.5
6 (I^6)	0.625 0.5 0.5 0.5 0.375	0.625 0.5 0.5 0.5 0.375 0.375 0.375 0.375
7 (I^7)	0.5 0.375 0.375 0.375 0.25 0.25 0.25	0.5 0.375 0.375 0.375 0.25 0.25 0.25 0.25
8 (I^8)	0.25 0.25 0.25 0.125 0.125 0.125 0 0	0.25 0.25 0.25 0.125 0.125 0.125 0 0
9 (I^9)	0 0 0 0	0 1 1 1 1 1 1 0

ตารางที่ 3.3 การแทนค่ารหัสแทนเส้นทางของตัวอักษร “ท”

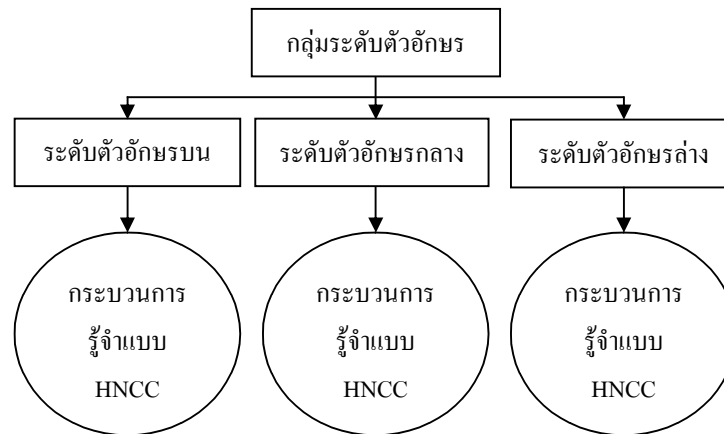
กรอบย่อยที่	ค่าข้อมูล	Input Vector มีขนาดเท่ากับ 13
1 (I^1)	0 0 0.875 0.75 0.75 1 1 0.75 0.75 0.75 0.75 1 0.125	0 0 0.875 0.75 0.75 1 1 0.75 0.75 0.75 0.75 1 0.125
2 (I^2)	0.125 0.125 0.125 0.125 0.125 0 0.125	0.125 0.125 0.125 0.125 0.125 0 0.125 0.125 0.125 0.125 0.125 0.125 0.125
3 (I^3)	0 0 0.875 0.875 0.875 0.75 0.75 0.75 0.75 0.75	0 0 0.875 0.875 0.875 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
4 (I^4)	0.75 0.75 0.75 0.75 0.75 0.75 0.625 0.625 0.625 0.625 0.625 0.625 0.625	0.75 0.75 0.75 0.75 0.75 0.75 0.625 0.625 0.625 0.625 0.625 0.625 0.625
5 (I^5)	0.875 0.75 0.875 0.625 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75	0.875 0.75 0.875 0.625 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
6 (I^6)	0.75 0.75 0.75 0.75 0.75 0.625 0.75 0.75 0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.625 0.75 0.75 0.75 0.75 0.75 0.75 0.75
7 (I^7)	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
8 (I^8)	-	0 0 0 0 0 0 0 0 0 0 0 0 0
9 (I^9)	-	0 0 0 0 0 0 0 0 0 0 0 0 0

ตารางที่ 3.4 การแทนค่ารหัสแทนเส้นทางของตัวอักษร “ส”

กรอบย่อยที่	ค่าข้อมูล	Input Vector มีขนาดเท่ากับ 10
1 (I^1)	0.125 0.125 0.125 0 0	0.125 0.125 0.125 0 1 1 1 1 1 0
2 (I^2)	0.125 0 0 0.875 1 1 0.875 0.875	0.125 0 0 0.875 1 1 0.875 0.875 0.875 0.875
3 (I^3)	0.125 0.125 0.125 0.125 0 0.875 0.75 0.75	0.125 0.125 0.125 0.125 0 0.875 0.75 0.75 0.75 0.75
4 (I^4)	0.75 0.625 0.5 0.875 0.75 0.75 0.75 0.75	0.75 0.625 0.5 0.875 0.75 0.75 0.75 0.75 0.75 0.75
5 (I^5)	0.75 0.75 0.75 0.75 0.75 0.75 0.875 0.875 0.875 0.875	0.75 0.75 0.75 0.75 0.75 0.75 0.875 0.875 0.875 0.875

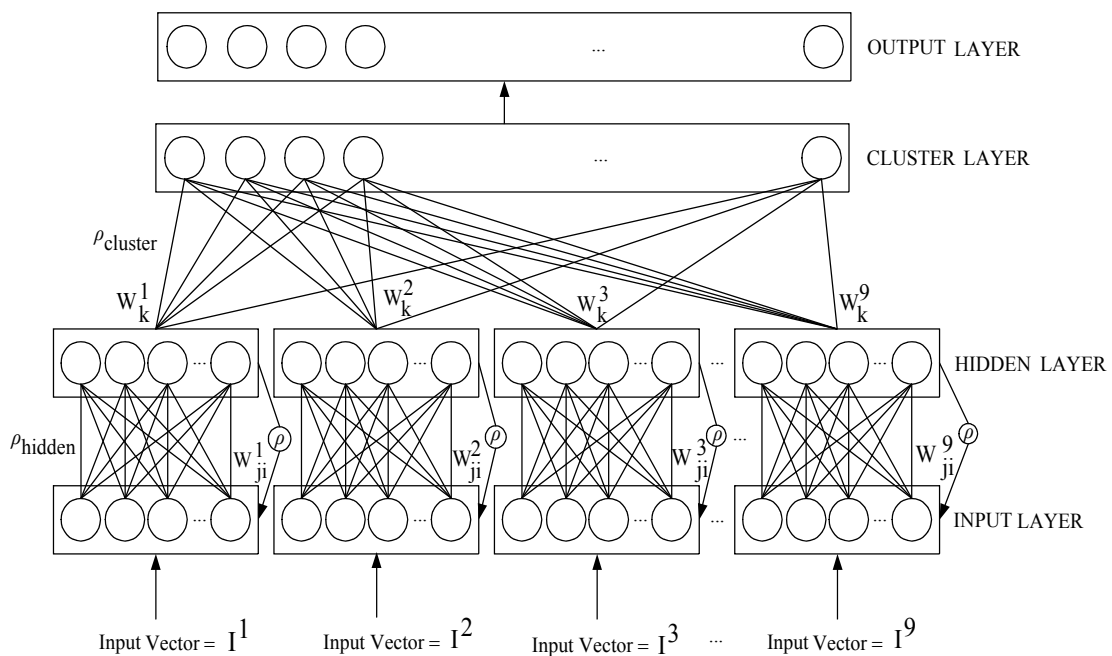
ตารางที่ 3.4 (ต่อ)

กรอบย่อยที่	ค่าข้อมูล	Input Vector มีขนาดเท่ากับ 10
6 (I^6)	0.5 0.5 0.5 0.625 0.5 0.5 0.5	0.5 0.5 0.5 0.625 0.5 0.5 0.5 0.5 0.5 0.5
7 (I^7)	0.625 0.5 0.5 0.625 0.625 0.75 0.75	0.625 0.5 0.5 0.625 0.625 0.75 0.75 0.75 0.75 0.75
8 (I^8)	0.625 0.75 0.875 0.75 0.875 0.875 1 0.875 1 1	0.625 0.75 0.875 0.75 0.875 0.875 1 0.875 1 1
9 (I^9)	0.125	0.125 0.125 0.125 0.125 0.125 0.125 0.125 0.125 0.125 0.125

**รูปที่ 3.23** โครงสร้างของระบบการรู้จำตัวอักษรที่นำเสนอในงานวิจัยนี้โดยอาศัย Hierarchical Normalized Cross-Correlation (HNCC) Neural Network

3.2 กระบวนการรู้จำ

ระบบในการรู้จำที่นำเสนอใหม่นี้เป็นแนวทางการรู้จำโดยมีการออกแบบสถาปัตยกรรมของโครงข่ายประสาทเทียมที่ประกอบด้วยโครงข่ายประสาทเทียมตามวิธีการของฟีดฟอร์เวิร์ดแบบสี่ชั้น (Four Layers Feedforward Neural Network) ตามที่แสดงในรูปที่ 3.24 ชั้นเลเยอร์แรกเรียกว่าชั้นอินพุตประกอบไปด้วย 9 ส่วน และแต่ละส่วนจะรับข้อมูลอินพุตเข้ามาเป็นอินพุตเวกเตอร์ที่มีขนาด N โหนดเท่า ๆ กันในแต่ละส่วนตามค่าข้อมูลอินพุตเวกเตอร์สำหรับแต่ละกรอบย่อยที่มีอยู่ทั้งหมด 9 กรอบย่อยที่ถูกแบ่งออก และจากนั้นก็ทำการกำหนดด้วยรหัสแทนเส้นตัวอักษรนั้น ๆ ตามขั้นตอนที่ 3 ในกระบวนการก่อนการรู้จำที่กล่าวถึงขั้นต้น ต่อมาเป็นชั้นเลเยอร์ที่สองเรียกว่าชั้นฮิดเดน (Hidden Layer) ทุกโหนดใน Hidden Layer จะเชื่อมต่อกับทุกโหนดเฉพาะในแต่ละส่วนของชั้นอินพุตเดียวกันตามที่แบ่งออกไว้ 9 ส่วน ในการเชื่อมต่อจะอาศัยเวกเตอร์ค่าน้ำหนัก (Weight Vector) ที่เชื่อมต่อผ่านทาง w_{ji}^h ระหว่างชั้น i กับชั้น j หรือ Hidden Layer สำหรับแต่ละส่วนย่อย h จากนั้นชั้นเลเยอร์ที่สามเรียกว่าชั้นคลัสเตอร์ (Cluster Layer) แต่ละโหนดในชั้นนี้จะเชื่อมต่อกับโหนดใน Hidden Layer ที่สร้างขึ้นมาในช่วงของการเรียนรู้ และจะมีค่า Weight Vector ในชั้นนี้เป็นตัวเชื่อมต่อกับโหนดที่เชื่อมกันคือ w_k^h สำหรับชั้นเลเยอร์ที่สี่เรียกว่าชั้นเอาต์พุต (Output Layer) แต่ละโหนดในชั้นนี้จะนำเสนอประเภทของตัวอักษรที่เป็น Target Output Vector สำหรับในแต่ละอินพุตเวกเตอร์ในช่วงของการเรียนรู้แบบ Supervised Learning



รูปที่ 3.24 สถาปัตยกรรมของโครงข่ายประสาทเทียมสี่ชั้น

อัลกอริทึมในการเรียนรู้ประกอบไปด้วย 7 ขั้นตอนดังต่อไปนี้
ขั้นตอนที่ 1

กำหนดค่า Input vector (I): ในแต่ละ Input vector ได้มีการแบ่งช่วงย่อยของ Input vector ออกเป็น 9 ส่วนย่อยตามทิศทางของเส้นตัวอักษรที่อยู่ในแต่ละกรอบย่อยที่ถูกแบ่งออกเป็น 9 กรอบย่อย (ตามตัวอย่างที่แสดงในตารางที่ 3.2, 3.3 และ 3.4 เป็นต้น)

$$I = \{I^1 \ I^2 \ I^3 \ I^4 \ I^5 \ I^6 \ I^7 \ I^8 \ I^9\} \quad (3.2)$$

$$I^1 = \{I_1^1 \ I_2^1 \ I_3^1 \ \dots \ I_N^1\} \quad (3.3)$$

$$I^2 = \{I_1^2 \ I_2^2 \ I_3^2 \ \dots \ I_N^2\} \quad (3.4)$$

$$I^3 = \{I_1^3 \ I_2^3 \ I_3^3 \ \dots \ I_N^3\} \quad (3.5)$$

$$I^4 = \{I_1^4 \ I_2^4 \ I_3^4 \ \dots \ I_N^4\} \quad (3.6)$$

$$I^5 = \{I_1^5 \ I_2^5 \ I_3^5 \ \dots \ I_N^5\} \quad (3.7)$$

$$I^6 = \{I_1^6 \ I_2^6 \ I_3^6 \ \dots \ I_N^6\} \quad (3.8)$$

$$I^7 = \{I_1^7 \ I_2^7 \ I_3^7 \ \dots \ I_N^7\} \quad (3.9)$$

$$I^8 = \{I_1^8 \ I_2^8 \ I_3^8 \ \dots \ I_N^8\} \quad (3.10)$$

$$I^9 = \{I_1^9 \ I_2^9 \ I_3^9 \ \dots \ I_N^9\} \quad (3.11)$$

กำหนดให้	I	แทน Input vector ของกระบวนการรู้จำ
	$I^1 \ \dots \ I^h \ \dots \ I^9$	แทน Input vector ที่มีการกำหนดในขอบเขตย่อยในแต่ละกรอบย่อยที่ 1 ถึง 9 ของกระบวนการรู้จำ
	N	คือจำนวนโหนดอินพุตในแต่ละส่วนย่อย (Feature Components)
	h	คือลำดับกรอบย่อยที่ส่งค่า Input Vector เข้ามาในระบบการรู้จำ

ขั้นตอนที่ 2

ครั้งแรกของกระบวนการเรียนรู้ จะกำหนดค่าเริ่มต้นให้กับ Weight Vector ที่ชั้น Hidden Layer และ Cluster Layer โดยที่ Weight vector (W_{ji}^h): ในชั้นของ Hidden Layer จะมีค่าเท่ากับค่าของ Input Vector แรกที่เริ่มต้นส่งเข้ามาในกระบวนการรู้จำ สำหรับ Weight Vector (W_{lh}): ในชั้นของ Cluster Layer ก็จะมีการกำหนดค่าเริ่มต้นให้เท่ากับค่า Weight Vector (W_1^h) ตามสมการต่อไปนี้

$$W_{li}^h = I_i^h \quad (3.12)$$

$$W_{lh} = W_l^h \quad (3.13)$$

กำหนดให้	W_{li}^h	คือค่า Weight vector อ้างอิงของโหนด $j=1$ ซึ่งเป็นโหนดแรก ที่กำหนดขึ้นมาในชั้น Hidden Layer สำหรับการเรียนรู้ครั้งแรก
	W_{lh}	คือค่า Weight vector อ้างอิงของโหนด $k=1$ ซึ่งเป็นโหนดแรก ที่สร้างขึ้นในชั้น Cluster Layer สำหรับการเรียนรู้ครั้งแรก
	I_i^h	แทน Input vector แรกที่ส่งเข้ามาใน Input Layer ในแต่ละ กรอบย่อยที่ h
	i	คือ จำนวน element ใน Input vector หรือจำนวน Input Nodes ที่ส่งเข้ามา

จำนวน neurons ในชั้น Hidden Layer = กรอบย่อยของแต่ละส่วนของตัวอักษร

ขั้นตอนที่ 3

จากนั้นเมื่อรับเอา Input Pattern ของภาพตัวอักษรตัวต่อไปเข้ามาจะมีการคำนวณหาค่า Choice Function ($T_j^h(I^h)$) ตามสมการที่ 3.14 โดยสามารถหาได้จากค่า Maximum of The Normalized Cross-correlation ระหว่าง Input Vector (I^h) ของกรอบย่อยที่ h กับค่า Weight Vector (W_j^h) ที่อ้างอิงของกรอบย่อยที่ h เดียวกัน ในทุก ๆ ค่า p ที่กำหนดตามสมการที่ 3.15 แล้วจะได้ โหนด J ที่มีค่า $T_j^h(I^h)$ มากที่สุดออกมา ค่าที่ได้จะมีค่า $0 \leq f(I^h, W_j^h) \leq 1$ ซึ่งถ้าค่าที่ได้ออกมามีค่า เข้าใกล้ 0 นั้นแสดงว่าความคล้ายคลึงกันระหว่าง Input Vector กับ Weight Vector นั้นมีความ คล้ายคลึงกันน้อยที่สุดหรือแทบจะไม่คล้ายคลึงกันเลย แต่หากค่าที่ได้มีค่าเข้าใกล้ 1 จะหมายถึงค่า Input Vector และค่า Weight Vector มีความคล้ายคลึงกันมาก โดยที่จะพิจารณาคำนวณหา ค่า $T_j^h(I^h)$ ออกมาในทุก ๆ กรอบย่อย h และสำหรับแต่ละโหนด J ใน Hidden Layer ที่มีค่า $T_j^h(I^h)$ สูงสุดออกมาจะถูกเลือกให้เป็นโหนดที่ชนะ (Winning Node) ตามสมการที่ 3.16 ในกรณีที่หากมีค่า มากที่สุดมากกว่า 1 ค่า จะเลือกค่าที่มากที่สุดจากลำดับโหนด J ที่น้อยที่สุดออกมา

$$T_j^h(I^h) = f(I^h, W_j^h) \quad (3.14)$$

$$f(x, y) = \max_p \left(\frac{\sum_{i=1}^N x_i y_{i-p}}{\sqrt{\sum_{i=1}^N x_i^2 \sum_{i=1}^N y_i^2}} \right) \quad (3.15)$$

$$J = \arg \max_j \{T_j^h(I^h)\} \quad (3.16)$$

กำหนดให้	$T_j^h(I^h)$	แทนค่า Choice Function ที่ใช้วัดความคล้ายคลึงกันระหว่าง Input Vector กับ Weight Vector ของโหนด j ใน Hidden Layer
$f(x, y)$		แทนค่า Maximum of The Normalized Cross-correlation at any shifted period p
p		คือค่า Lag Variable ที่มีค่าระหว่าง $-(N-1)$ ถึง $(N-1)$
N		คือจำนวน Feature compoments ของ Input vector ในแต่ละ h^{th} segment
I		คือ Input Vector
W		คือ Weight Vector
h		คือลำดับกรอบย่อยที่ส่งค่า Input Vector เข้ามาในการรู้จำ มีค่าเท่ากับ 1, 2, 3, ..., 9
i		มีค่าเท่ากับ 1, 2, 3, ..., N
j		มีค่าเท่ากับ 1, 2, 3, ..., M^h
J		เป็น Winning Node

ขั้นตอนที่ 4

ตรวจสอบค่า Vigilance Parameter (ρ) ซึ่งเป็นค่าความคล้ายคลึงระหว่าง Input Vector และ Weight Vector ของโหนดที่ชนะ J นั้นว่ามีความคล้ายคลึงกันมากพอที่จะจัดให้ Input นั้นไปรวมอยู่ในกลุ่ม Category ของโหนด J นั้นหรือไม่ จาก Vigilance Criterion โดยจะนำเอาค่า $T_j^h(I^h)$ มาเปรียบเทียบกับค่า ρ_{hidden} ตามสมการที่ 3.17 หากค่าที่ได้จากสมการนั้นเป็นจริงแสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะ J นั้นได้ ต่อจากนั้นจะทำการปรับค่า Weight Vector (W_j^h) สำหรับโหนดที่ชนะ J ตามสมการที่ 3.18 จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$T_j^h(I^h) \geq \rho_{hidden} \quad (3.17)$$

$$W_J^h(t+1) = \beta(I^h \wedge W_J^h(t)) + (1-\beta)W_J^h(t) \quad (3.18)$$

กำหนดให้	$T_J^h(I^h)$	แทนค่า Choice Function ที่ใช้วัดความคล้ายคลึงกันระหว่าง Input Vector กับ Weight Vector ของโหนด J ใน Hidden Layer
	ρ_{hidden}	คือค่า Vigilance Parameter ที่ Hidden Layer มีค่าระหว่าง 0 - 1
	$W_J^h(t+1)$	คือค่า Weight Vector ใหม่ที่ปรับเปลี่ยนสำหรับโหนด J
	$W_J^h(t)$	คือค่า Weight Vector เก่าของโหนด J
	I^h	คือค่า Input Vector ที่เป็น Pattern นำมาเปรียบเทียบกับค่า Weight Vector ของโหนด J ใน Hidden Layer
	β	คือค่า Learning Parameter จะมีค่าได้ 3 ค่าตามความเร็วของการเรียนรู้ดังนี้ ○ $\beta=1$ เรียกว่า fast learning ส่งผลให้ค่าของ Weight Vector มีการลดลงเปลี่ยนแปลงไปอย่างรวดเร็วขึ้นอยู่กับ Input และ Weight Vector ○ $0 < \beta < 1$ แสดงว่ามีการเรียนรู้แบบค่อยเป็นค่อยไป ส่งผลให้ค่าของ Weight Vector มีการลดลงอย่างเรื่อย ๆ เป็นสัดส่วนกับค่า β ที่เลือกใช้ ○ $\beta=0$ แสดงว่าไม่มีการเรียนรู้เลย ส่งผลให้ค่าของ Weight Vector ไม่มีการเปลี่ยนแปลงตลอดการเรียนรู้

แต่ในกรณีที่ค่าที่ได้จากสมการที่ 3.17 นั้นเป็นเท็จ แสดงว่า Input นั้นมีความคล้ายคลึงกันไม่มากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะ J นั้นได้ ต่อจากนั้นจะทำการสร้างโหนดใหม่ใน Hidden Layer และกำหนดให้มี Weight Vector ของโหนดใหม่ที่สร้างขึ้นมามีค่าเท่ากับค่า Input Vector ของช่วงกรอบส่วนย่อย h นั้นตามสมการที่ 3.19 จะเรียกกรณีนี้ว่าการเกิด “Reset”

$$W_J^h = I^h \quad (3.19)$$

กำหนดให้	W_J^h	คือค่า Weight Vector ใหม่ที่กำหนดให้กับโหนด J ใน Hidden Layer ใหม่ที่สร้างขึ้นมา
	I^h	คือค่า Input Vector ที่เป็น Pattern ที่กำหนดให้กับ Hidden Node ใหม่ที่สร้างขึ้นมา

ขั้นตอนที่ 5

จากนั้นจะส่งต่อค่า Weight Vector ของโหนดที่ชนะ J (กรณีการเกิด Resonance) หรือค่า Weight Vector ของโหนดใหม่ที่สร้างขึ้นมาจากโหนด J (กรณีการเกิด Reset) ใน Hidden Layer ไปยัง Cluster Layer แล้วมีการคำนวณค่า Choice Function กับแต่ละโหนด k^{th} ใน Cluster Layer ซึ่งจะคำนวณตามสมการที่ 3.20

$$C_k = f(W_J, W_k) \quad (3.20)$$

กำหนดให้	W_J	คือค่า Weight Vector ที่ส่งต่อขึ้นมาจาก Hidden Layer โดยจะมีค่า $W_J = \{W_J^1, W_J^2, W_J^3, \dots, W_J^9\}$
	W_k	คือค่า Reference Weight Vector ใน Cluster Layer โดยจะมีค่า $W_k = \{W_{k1}, W_{k2}, W_{k3}, \dots, W_{k9}\}$
	$f(W_J, W_k)$	เป็นฟังก์ชันที่คำนวณได้ตามสมการที่ 3.14
	C_k	คือค่า Choice Function ที่ใช้วัดความคล้ายคลึงกันระหว่าง Weight Vector ของ โหนด J ใน Hidden Layer กับ Weight Vector ของ โหนด k ใน Cluster Layer

ขั้นตอนที่ 6

ต่อจากนั้นจะทำการเลือกโหนดที่ชนะ (K) ใน Cluster Layer ที่มีค่า Maximum Choice Function เมื่อเปรียบเทียบกับทุกโหนดใน Cluster Layer ตามสมการที่ 3.21 โดยที่หากมีโหนดมากกว่า 1 โหนดที่มีค่า Maximum Choice Function จะทำการเลือกลำดับโหนดที่ชนะ K ที่มีลำดับน้อยที่สุดออกมา

$$K = \arg \max_k \{C_k\} \quad (3.21)$$

กำหนดให้	C_k	คือค่า Choice Function ที่ใช้วัดความคล้ายคลึงกันระหว่าง Weight Vector ของ โหนด J ใน Hidden Layer กับ Weight Vector ของ โหนด k ใน Cluster Layer
	K	คือโหนดที่ถูกเลือกให้เป็นโหนดที่ชนะใน Cluster Layer ที่มีค่า Maximum Choice Function

ขั้นตอนที่ 7

ทำการเปรียบเทียบ C_K กับ Vigilance Criterion ที่มีการกำหนดค่าไว้ใน Cluster Layer โดยพิจารณา Vigilance Parameter ($\rho_{cluster}$) ที่ Cluster Layer และทำการเปรียบเทียบกันตามสมการที่ 3.22 เพื่อตรวจสอบว่าโหนด K ที่เลือกนั้นจัดเป็นกลุ่ม Output Class ที่ถูกต้องที่สามารถจัด Input Pattern นั้นเข้ากลุ่มได้หรือไม่ หากสมการที่ 3.22 เป็นจริงหรือเรียกว่าเกิดกรณี “Resonance” แสดงว่า Weight Vector (W_J) นั้นมีความคล้ายคลึงกับ Weight Vector (W_k) ของโหนดที่เลือกให้เป็นโหนดที่ชนะคือโหนด K ใน Cluster Layer และตรงกับกลุ่มผลลัพธ์ของ Input Pattern นอกจากนี้จะพิจารณาต่อว่าหาก W_J นั้นเป็นค่า Weight Vector ที่ส่งค่ามาจากโหนด J ที่มีการสร้างขึ้นใหม่ใน Hidden Layer ก็จะมีการปรับค่า Weight Vector ($W_{kh}(t+1)$) ตามสมการที่ 3.23 แต่หาก W_J นั้นเป็นค่า Weight Vector ที่ส่งค่ามาจากโหนด J ที่เป็นโหนดเดิมอยู่แล้วใน Hidden Layer ก็จะมีการกำหนดค่า Weight Vector ($W_{kh}(t+1)$) ตามสมการที่ 3.24

$$C_K \geq \rho_{cluster} \quad (3.22)$$

$$W_{kh}(t+1) = \beta(W_J^h \wedge W_{kh}(t)) + (1 - \beta)W_{kh}(t) \quad (3.23)$$

$$W_{kh}(t+1) = W_{kh}(t) \quad (3.24)$$

กำหนดให้	C_K	คือค่า Maximum Choice Function ของโหนดที่เลือกให้ชนะ K
	$\rho_{cluster}$	คือค่า Vigilance Parameter ที่ Cluster Layer โดยจะมีค่าระหว่าง 0 และ 1
	W_J^h	คือค่า Weight Vector ที่ส่งต่อมาจากโหนด J ที่สร้างขึ้นใหม่ใน Hidden Layer
	$W_{kh}(t+1)$	คือค่า Weight Vector ใหม่ที่ปรับเปลี่ยนสำหรับโหนด k ที่เชื่อมกับกรอบย่อยที่ h
	$W_{kh}(t)$	คือค่า Weight Vector เก่าของโหนด k ที่เชื่อมกับกรอบย่อยที่ h

ทั้งนี้หากสมการที่ 3.22 เป็นเท็จหรือเรียกว่าเป็นกรณี “Reset” แสดงว่าโหนดที่เลือกให้เป็นโหนดชนะ K ที่ Cluster Layer นั้นมีการจัดเข้ากลุ่มที่ไม่ถูกต้องตรงตาม กลุ่ม Output Class ของ Input Pattern จะมีการสร้างโหนดใหม่ขึ้นมาใน Cluster Layer และทำการกำหนด Weight Vector ของโหนดใหม่ k (W_k) ให้มีค่าเท่ากับค่า Weight Vector (W_J) ที่ส่งต่อขึ้นมาจาก Hidden Layer ตามสมการที่ 3.25 หลังจากนั้นจะมีการสร้างการเชื่อมต่อจากโหนดใหม่ใน Cluster Layer ไปยังโหนดใน Output Layer ที่ตรงตาม Target Output ของ Input Pattern ที่ทำการเรียนรู้ครั้งนั้น

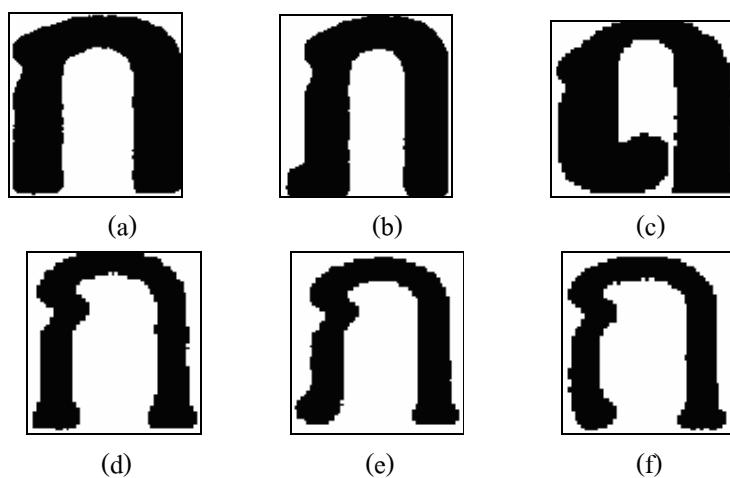
$$W_k = W_J \quad (3.25)$$

กำหนดให้	k	คือโหนดใหม่ที่ทำกรสร้างขึ้นมาใน Cluster Layer
	W_k	คือค่า Weight Vector ที่กำหนดขึ้นมาใหม่ให้กับโหนด k ที่สร้างขึ้นมาใหม่ใน Cluster Layer
	W_J	คือค่า Weight Vector ที่ส่งต่อมาจากโหนด J ใน Hidden Layer

ช่วงของการทดสอบระบบ รูปแบบอินพุตที่เข้ามาจะเริ่มจากการแบ่งกลุ่มในการแยกแยะกลุ่ม 3 กลุ่มได้แก่ระดับบน กลาง และล่างตามลำดับ จากนั้นจะทำการทดสอบส่งผ่านค่าเข้ามาแล้วคาดคะเนตาม Cluster Node ที่ได้ออกมาจากค่าความคล้ายคลึงกันมากที่สุดที่ระบบได้ทำการคำนวณออกมาว่ามีการจัดเข้าประเภท Class ใดใน Output Node ก็จะได้คำตอบของแต่ละอินพุตที่ใส่เข้าไปออกมาจากกระบวนการรู้จำแบบใหม่นี้

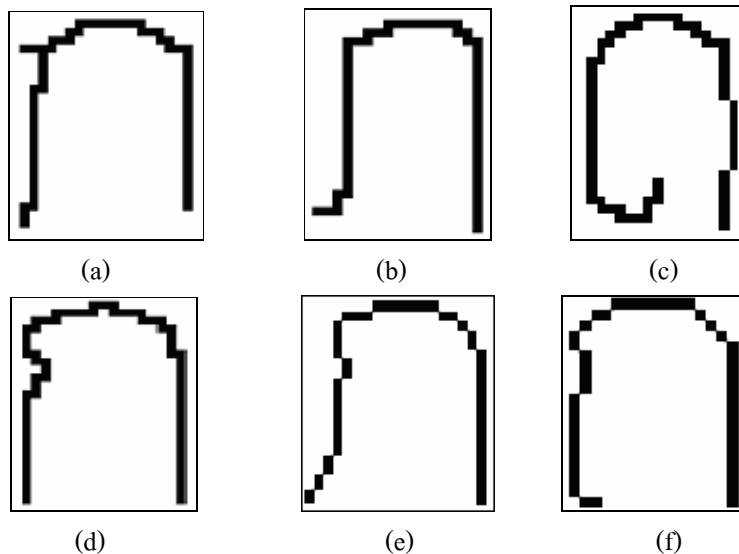
3.3 ตัวอย่างการคำนวณ

3.3.1 ข้อมูลใน Training Set มีทั้งหมด 6 รูปแบบได้แก่ตัวอักษร “ก” “ภ” “ด” “ค” “ภ” และ “ด” ตามลำดับดังรูปที่ 3.25



รูปที่ 3.25 รูปแบบข้อมูล Training Set (a) ตัวอักษรตัวที่ 1 คือ “ก” (b) ตัวอักษรตัวที่ 2 คือ “ภ” (c) ตัวอักษรตัวที่ 3 คือ “ด” (d) ตัวอักษรตัวที่ 4 คือ “ก” (e) ตัวอักษรตัวที่ 5 คือ “ภ” (f) ตัวอักษรตัวที่ 6 คือ “ด”

ต่อจากนั้นเข้าสู่กระบวนการก่อนการรู้จำเริ่มจากการแบ่งออกเป็นกรอบตัวอักษรแต่ละตัว ดังรูปที่ 3.25 แล้วทำการลดขนาดให้เท่ากับ 25x29 Pixels จากนั้นทำการ Skeleton ตัวอักษรแต่ละตัว ที่รับเข้ามา หลังจากนั้นจะได้กรอบตัวอักษรที่มีการ Skeleton แล้วดังรูปที่ 3.26



รูปที่ 3.26 ข้อมูลภาพตัวอักษรใน Training Set ที่มีการ Resize [29, 25] และทำการ Skeleton แล้ว

(a) ตัวอักษรตัวที่ 1 คือ “ก” (b) ตัวอักษรตัวที่ 2 คือ “ข” (c) ตัวอักษรตัวที่ 3 คือ “ค”
(d) ตัวอักษรตัวที่ 4 คือ “ก” (e) ตัวอักษรตัวที่ 5 คือ “ข” (f) ตัวอักษรตัวที่ 6 คือ “ค”

ขั้นตอนต่อไปคือการแทนค่ารหัสแทนตัวอักษรของภาพตัวอักษรใน Training Set ตัวที่ 1, 2, 3, 4, 5, และ 6 ตามตารางที่ 3.5, 3.6, 3.7, 3.8, 3.9, และ 3.10 ตามลำดับ

ตารางที่ 3.5 การแทนค่ารหัสแทนเส้นทางของภาพตัวอักษรใน Training Set ตัวที่ 1

กรอบย่อยที่	ค่าข้อมูล	Input Vector มีขนาดเท่ากับ 12
1 (I^1)	0.25 0.25 0 0.25 0.25 0.25 0.25 0.25	0.25 0.25 0 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25
2 (I^2)	0.25 0.25 0.25 0.25 0.25 0.25 0 0.25	0.25 0.25 0.25 0.25 0.25 0.25 0 0.25 0.25 0.25 0.25 0.25
3 (I^3)	0.25 0.25 0.25 0.5 0.5 0 0.25 0 0 0.25 0	0.25 0.25 0.25 0.5 0.5 0 0.25 0 0 0.25 0 0
4 (I^4)	0.25 0 0 0 0 0 0 0	0.25 0 1 1 1 1 1 1 1 1 1 0

ตารางที่ 3.5 (ต่อ)

กรอบย่อยที่	ค่าข้อมูล	Input Vector มีขนาดเท่ากับ 12
5 (I^5)	0.75 1 1 0.75 1 0.75 1 1 0.75 0.75 0.75 0.75	0.75 1 1 0.75 1 0.75 1 1 0.75 0.75 0.75 0.75
6 (I^6)	0.75 0.75 0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
7 (I^7)	0.75 0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
8 (I^8)	-	0 0 0 0 0 0 0 0 0 0 0 0
9 (I^9)	-	0 0 0 0 0 0 0 0 0 0 0 0

ตารางที่ 3.6 การแทนค่ารหัสแทนเส้นทางของภาพตัวอักษรใน Training Set ตัวที่ 2

กรอบย่อยที่	ค่าข้อมูล	Input Vector มีขนาดเท่ากับ 11
1 (I^1)	0 0 0 0.25 0.25 0 0.25 0.25 0.25 0.25	0 1 0 0.25 0.25 0 0.25 0.25 0.25 0.25 0.25
2 (I^2)	0.25 0.25 0.25 0.25 0.25 0.25	0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25
3 (I^3)	0.25 0.25 0.25 0.25 0 0	0.25 0.25 0.25 0.25 0 1 1 1 1 1 0
4 (I^4)	0.25 0 0 0.25 0 0 0 0	0.25 0 0 0.25 0 1 1 1 1 1 0
5 (I^5)	0 0 0.75 1 0.75 1 0.75 0.75 0.75 0.75 0.75	0 0 0.75 1 0.75 1 0.75 0.75 0.75 0.75 0.75
6 (I^6)	0.75 0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
7 (I^7)	0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
8 (I^8)	-	0 0 0 0 0 0 0 0 0 0 0
9 (I^9)	-	0 0 0 0 0 0 0 0 0 0 0

ตารางที่ 3.7 การแทนค่ารหัสแทนเส้นทางของภาพตัวอักษรใน Training Set ตัวที่ 3

กรอบย่อยที่	ค่าข้อมูล	Input Vector มีขนาดเท่ากับ 12
1 (I^1)	0.5 0.5 0.625 0.5 0.5 0.5	0.5 0.5 0.625 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
2 (I^2)	0.5 0.5 0.375 0.375 0.25 0.375 0.25 0.25 0.25	0.5 0.5 0.375 0.375 0.25 0.375 0.25 0.25 0.25 0.25 0.25 0.25
3 (I^3)	0.25 0.125 0.25 0.25 0.25 0.375 0.25	0.25 0.125 0.25 0.25 0.25 0.375 0.25 0.25 0.25 0.25 0.25 0.25
4 (I^4)	0.25 0.125 0.125 0.125 0.125 0.125 0	0.25 0.125 0.125 0.125 0.125 0.125 0 1 1 1 1 0
5 (I^5)	0 0 0 0 0 0.875	0 1 1 1 0 0.875 0.875 0.875 0.875 0.875 0.875 0.875
6 (I^6)	1 0.875 1 0.875 0.75 0.75 0.75 0.75	1 0.875 1 0.875 0.75 0.75 0.75 0.75 0.75
7 (I^7)	0.75 0.75 0.875 0.625 0.75 0.75 0.875	0.75 0.75 0.875 0.625 0.75 0.75 0.875 0.875 0.875
8 (I^8)	0.625 0.75 0.75 0.75 0.75 0.75	0.625 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
9 (I^9)	-	0 0 0 0 0 0 0 0 0 0

ตารางที่ 3.8 การแทนค่ารหัสแทนเส้นทางของภาพตัวอักษรใน Training Set ตัวที่ 4

กรอบย่อยที่	ค่าข้อมูล	Input Vector มีขนาดเท่ากับ 13
1 (I^1)	0.25 0.25 0.25 0.25 0.25 0.25 0.25	0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25
2 (I^2)	0.25 0.25 0.25 0.25 0 0.25 0.25 0 0.25 0.25	0.25 0.25 0.25 0.25 0 0.25 0.25 0 0.25 0.25 0.25 0.25 0.25
3 (I^3)	0.5 0.25 0.5 0.25 0.25 0.25 0 0.25 0 0 0.25 0 0	0.5 0.25 0.5 0.25 0.25 0.25 0 0.25 0 0 0.25 0 0
4 (I^4)	0 0.25 0 0 0.75 1 1 1	0 0.25 0 0 0.75 1 1 1 1 1 1 1 1
5 (I^5)	0.75 1 1 0.75 1 0.75 0.75 0.75 1 0.75 0.75	0.75 1 1 0.75 1 0.75 0.75 0.75 1 0.75 0.75 0.75 0.75

ตารางที่ 3.8 (ต่อ)

กรอบย่อยที่	ค่าข้อมูล	Input Vector มีขนาดเท่ากับ 13
6 (I^6)	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
7 (I^7)	0.75 0.75 0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
8 (I^8)	-	0 0 0 0 0 0 0 0 0 0 0 0 0
9 (I^9)	-	0 0 0 0 0 0 0 0 0 0 0 0 0

ตารางที่ 3.9 การแทนค่ารหัสแทนเส้นทางทิศทางของภาพตัวอักษรใน Training Set ตัวที่ 5

กรอบย่อยที่	ค่าข้อมูล	Input Vector มีขนาดเท่ากับ 8
1 (I^1)	0.125 0.25 0.125 0.25 0.125 0.25 0.25 0.25	0.125 0.25 0.125 0.25 0.125 0.25 0.25 0.25
2 (I^2)	0.25 0.25 0.25 0.25 0.25 0.125 0.25 0.25	0.25 0.25 0.25 0.25 0.25 0.125 0.25 0.25
3 (I^3)	0.375 0.25 0.25 0.25 0.125 0 0 0	0.375 0.25 0.25 0.25 0.125 0 1 0
4 (I^4)	0.125 0 0 0 0 0 0 0	0.125 0 1 1 1 1 1 0
5 (I^5)	0.875 1 0.875 0.875 0.75 0.875 0.75 0.75	0.875 1 0.875 0.875 0.75 0.875 0.75 0.75
6 (I^6)	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
7 (I^7)	0.75 0.75 0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
8 (I^8)	-	0 0 0 0 0 0 0 0
9 (I^9)	-	0 0 0 0 0 0 0 0

ตารางที่ 3.10 การแทนค่ารหัสแทนเส้นทางทิศทางของภาพตัวอักษรใน Training Set ตัวที่ 6

กรอบย่อยที่	ค่าข้อมูล	Input Vector มีขนาดเท่ากับ 9
1 (I^1)	0.5 0.375 0.25 0.25 0.25 0.25 0.25 0.25	0.5 0.375 0.25 0.25 0.25 0.25 0.25 0.25 0.25
2 (I^2)	0.25 0.25 0.25 0.25 0.125 0.25 0.25 0.25	0.25 0.25 0.25 0.25 0.125 0.25 0.25 0.25 0.25

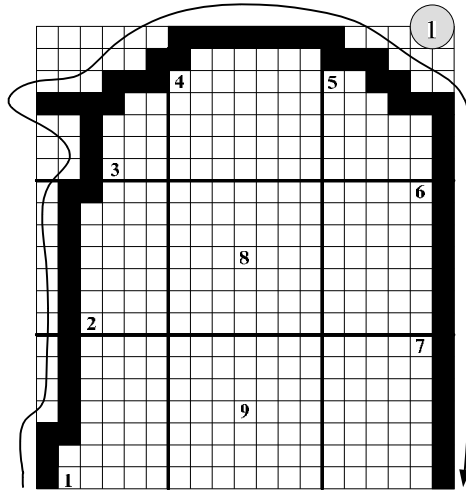
ตารางที่ 3.10 (ต่อ)

กรอบย่อยที่	ค่าข้อมูล	Input Vector มีขนาดเท่ากับ 9
3 (I^3)	0.25 0.375 0.25 0.125 0.125 0 0 0.125 0	0.25 0.375 0.25 0.125 0.125 0 0 0.125 0
4 (I^4)	0 0 0 0 0 0	0 1 1 1 1 1 1 1 0
5 (I^5)	0 0.875 0.875 0.875 0.875 0.75 0.75 0.75	0 0.875 0.875 0.875 0.875 0.75 0.75 0.75 0.75
6 (I^6)	0.75 0.75 0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
7 (I^7)	0.75 0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
8 (I^8)	-	0 0 0 0 0 0 0 0 0
9 (I^9)	-	0 0 0 0 0 0 0 0 0

จะสังเกตเห็นว่าทั้ง 6 รูปแบบภาพตัวอักษรใน Training Set ที่รับเข้ามาเมื่อหาจุดศูนย์กลางของกรอบตัวอักษรแล้วอยู่ในกลุ่มระดับกลาง จากนั้นจะส่งรูปแบบตัวอักษรแต่ละตัวเข้าสู่ระบบการรู้จำแนวทางใหม่ในกระบวนการรู้จำตัวอักษรภาษาไทยด้วยวิธีใหม่ที่น่าสนใจ

กระบวนการรู้จำในขั้นตอนของการ Training

การเรียนรู้ครั้งที่ 1 รับค่ารูปแบบตัวอักษรที่มีการกำหนด Input Vector ตามตารางที่ 3.5 โดยรูปแบบของตัวอักษรมีการแบ่งออกเป็น 9 กรอบย่อยแสดงดังรูปที่ 3.27 แล้วทำการกำหนดค่า Vigilance Parameter (ρ) ได้แก่ ค่า $\rho_{\text{hidden}} = 0.75$ และค่า $\rho_{\text{cluster}} = 0.85$ สำหรับค่า Learning Rate (β) กำหนดให้เท่ากับ 0.05 จากนั้นกำหนดค่าเริ่มต้นให้กับ Weight Vector โดยที่ Weight vector (W_{ji}^h): ในชั้นของ Hidden Layer มีค่าเท่ากับ Input Vector แรกที่เริ่มต้นส่งเข้ามาในกระบวนการรู้จำ ค่า Weight Vector ที่ชั้น Hidden Layer ในการเรียนรู้ครั้งที่ 1 แสดงตามตารางที่ 3.11



รูปที่ 3.27 แสดงทิศทางของเส้นตัวอักษรในการ Training ตัวที่ 1 ตามการจัดเรียงกรอบย่อยของ ภาพตัวอักษรตามลำดับของทิศทางเส้นตัวอักษรเพื่อกำหนดรหัสแทนเส้นตัวอักษร

ตารางที่ 3.11 ค่า Weight Vector ที่ชั้น Hidden Layer ในการเรียนรู้ครั้งที่ 1

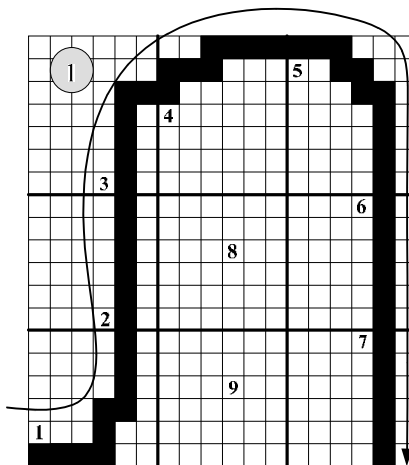
กรอบย่อยที่	Weight Vector ที่ชั้น Hidden Layer
1 (W_{li}^1)	0.25 0.25 0 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25
2 (W_{li}^2)	0.25 0.25 0.25 0.25 0.25 0.25 0 0.25 0.25 0.25 0.25 0.25
3 (W_{li}^3)	0.25 0.25 0.25 0.5 0.5 0 0.25 0 0 0.25 0 0
4 (W_{li}^4)	0.25 0 1 1 1 1 1 1 1 1 1 0
5 (W_{li}^5)	0.75 1 1 0.75 1 0.75 1 1 0.75 0.75 0.75 0.75
6 (W_{li}^6)	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
7 (W_{li}^7)	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
8 (W_{li}^8)	0 0 0 0 0 0 0 0 0 0 0 0
9 (W_{li}^9)	0 0 0 0 0 0 0 0 0 0 0 0

จากนั้นจะส่งต่อค่า Weight Vector ของโหนด J ใน Hidden Layer ต่อไปให้เป็นค่า Input Vector ของ Cluster Layer ส่วน Weight Vector (W_{lh}): ในชั้นของ Cluster Layer ก็จะมีการกำหนดค่าเริ่มต้นให้เท่ากับค่า Weight Vector (W_l^h) ที่ส่งขึ้นมาเป็น Input ตามสมการที่ 3.19 ดังนั้นค่า Weight Vector ที่ชั้น Cluster Layer ในการเรียนรู้ครั้งที่ 1 แสดงตามตารางที่ 3.12 จากนั้นเปรียบเทียบกับคำตอบที่แท้จริงจะได้ Output Node ที่ 1 คือ Target Output = “ก”

ตารางที่ 3.12 ค่า Weight Vector ที่ชั้น Cluster Layer ในการเรียนรู้ครั้งที่ 1

กรอบย่อยที่	Weight Vector ที่ชั้น Cluster Layer
1 (W_{11})	0.25 0.25 0 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25
2 (W_{12})	0.25 0.25 0.25 0.25 0.25 0.25 0 0.25 0.25 0.25 0.25 0.25
3 (W_{13})	0.25 0.25 0.25 0.5 0.5 0 0.25 0 0 0.25 0 0
4 (W_{14})	0.25 0 1 1 1 1 1 1 1 1 1 0
5 (W_{15})	0.75 1 1 0.75 1 0.75 1 1 0.75 0.75 0.75 0.75
6 (W_{16})	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
7 (W_{17})	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
8 (W_{18})	0 0 0 0 0 0 0 0 0 0 0 0
9 (W_{19})	0 0 0 0 0 0 0 0 0 0 0 0

การเรียนรู้ครั้งที่ 2 รับค่ารูปแบบตัวอักษรตัวที่ 2 ที่มีการกำหนด Input Vector ตามตารางที่ 3.6 โดยรูปแบบของตัวอักษรที่มีการแบ่งออกเป็น 9 กรอบย่อยแสดงดังรูปที่ 3.28



รูปที่ 3.28 แสดงทิศทางของเส้นตัวอักษรในการ Training ตัวที่ 2 ตามการจัดเรียงกรอบย่อยของภาพตัวอักษรตามลำดับของทิศทางเส้นตัวอักษรเพื่อกำหนดรหัสแทนเส้นตัวอักษร

ทำการคำนวณหาค่า Choice Function ($T_j^h(I^h)$) ตามสมการที่ 3.14 แล้วนำเอาค่า $T_j^h(I^h)$ มาเปรียบเทียบกับค่า $\rho_{\text{hidden}} = 0.75$ ที่กำหนดไว้แล้วทำการพิจารณาตรวจสอบค่า Vigilance Parameter ที่ Hidden Layer (ρ_{hidden}) ตามสมการที่ 3.17 ดังนี้

กรอบย่อยที่ 1: คำนวณหาค่า Choice Function ($T_1^1(I^1)$) = 0.7385

$$W_{li}^1 = 0.2500 \ 0.2500 \ 0 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500$$

เนื่องจาก $T_1^1(I^1) < \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันไม่มากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการสร้างโหนดใหม่ใน Hidden Layer และกำหนดให้มี Weight Vector ของโหนดใหม่ที่สร้างขึ้นมาเท่ากับค่า Input Vector ของช่วงกรอบส่วนย่อย h นั้น จะเรียกกรณีนี้ว่าการเกิด “Reset”

$$W_{2i}^1 = 0 \ 1.0000 \ 0 \ 0.2500 \ 0.2500 \ 0 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500$$

กรอบย่อยที่ 2: คำนวณหาค่า Choice Function ($T_1^2(I^2)$) = 0.9574

$$W_{li}^2 = 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500$$

เนื่องจาก $T_1^2(I^2) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^2(t+1) = 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500$$

กรอบย่อยที่ 3: คำนวณหาค่า Choice Function ($T_1^3(I^3)$) = 0.8473

$$W_{li}^3 = 0.2500 \ 0.2500 \ 0.2500 \ 0.5000 \ 0.5000 \ 0 \ 0.2500 \ 0 \ 0 \ 0.2500 \ 0 \ 0$$

เนื่องจาก $T_1^3(I^3) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^3(t+1) = 0.2500 \ 0.2500 \ 0.2500 \ 0.4875 \ 0.4750 \ 0 \ 0.2500 \ 0 \ 0 \ 0.2500 \ 0 \ 0$$

กรอบย่อยที่ 4: คำนวณหาค่า Choice Function ($T_1^4(I^4)$) = 0.7795

$$W_{li}^4 = 0.2500 \ 0 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 0$$

เนื่องจาก $T_1^4(I^4) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^4(t+1) = 0.2500 \ 0 \ 0.9500 \ 0.9625 \ 0.9500 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 0.9500 \ 0$$

กรอบย่อยที่ 5: คำนวณหาค่า Choice Function ($T_1^5(I^5)$) = 0.9266

$$W_{li}^5 = 0.7500 \ 1.0000 \ 1.0000 \ 0.7500 \ 1.0000 \ 0.7500 \ 1.0000 \ 1.0000 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

เนื่องจาก $T_1^5(I^5) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^5(t+1) = 0.7125 \ 0.9500 \ 0.9875 \ 0.7500 \ 0.9875 \ 0.7500 \ 0.9875 \ 0.9875 \ 0.7500 \ 0.7500 \ 0.7500 \\ 0.7500$$

กรอบย่อยที่ 6: คำนวณหาค่า Choice Function ($T_1^6(I^6)$) = 1

$$W_{li}^6 = 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

เนื่องจาก $T_1^6(I^6) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^6(t+1) = 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \\ 0.7500$$

กรอบย่อยที่ 7: คำนวณหาค่า Choice Function ($T_1^7(I^7)$) = 1

$$W_{li}^7 = 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

เนื่องจาก $T_1^7(I^7) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^7(t+1) = 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \\ 0.7500$$

กรอบย่อยที่ 8: คำนวณหาค่า Choice Function ($T_1^8(I^8)$) = 0

$$W_{li}^8 = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

เนื่องจากขนาดของ $W_{li}^8 = 0$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

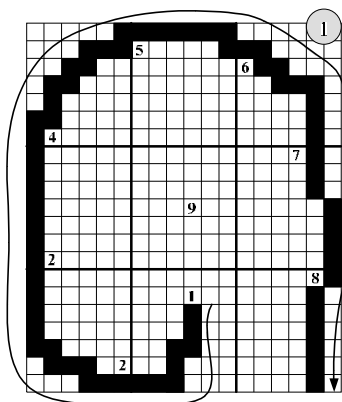
$$W_{li}^8(t+1) = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

ตารางที่ 3.14 (ต่อ)

กรอบย่อยที่	Weight Vector ที่ชั้น Cluster Layer
8 (W_{28})	0 0 0 0 0 0 0 0 0 0 0 0 0
9 (W_{29})	0 0 0 0 0 0 0 0 0 0 0 0 0

เมื่อทำการพิจารณา Choice Function ตามสมการที่ 3.20 จะได้ค่า C_K ออกมา จากนั้นทำการเปรียบเทียบ $C_K \geq \rho_{\text{cluster}}$ หรือไม่ ตามสมการที่ 3.22 ทั้งนี้ $C_1 = 0.7896$ ซึ่งทำให้สมการที่ 3.22 เป็นเท็จหรือเรียกว่าเป็นกรณี “Reset” แสดงว่าโหนดที่เลือกให้เป็นโหนดชนะ $K=1$ ที่ Cluster Layer นั้นมีการจัดเข้ากลุ่มที่ไม่ถูกต้องตรงตาม กลุ่ม Output Class ของ Input Pattern จะมีการสร้างโหนดใหม่ขึ้นมาใน Cluster Layer โดยที่ $K = 2$ และทำการกำหนด Weight Vector ของโหนดใหม่ (W_{kh}) ให้มีค่าเท่ากับค่า Weight Vector (W_j) ที่ส่งต่อขึ้นมาจาก Hidden Layer ตามสมการที่ 3.25 หลังจากนั้นจะมีการสร้างการเชื่อมต่อจากโหนดใหม่ใน Cluster Layer ไปยังโหนดใน Output Layer ที่ตรงตาม Target Output ของ Input Pattern ที่ทำการเรียนรู้ครั้งนั้นคือ Target Output = “ภ”

การเรียนรู้ครั้งที่ 3 รับค่ารูปแบบตัวอักษรที่มีการกำหนด Input Vector ตามตารางที่ 3.7 โดยรูปแบบของตัวอักษรที่มีการแบ่งออกเป็น 9 กรอบย่อยแสดงดังรูปที่ 3.29



รูปที่ 3.29 แสดงทิศทางของเส้นตัวอักษรในการ Training ตัวที่ 3 ตามการจัดเรียงกรอบย่อยของภาพตัวอักษรตามลำดับของทิศทางเส้นตัวอักษรเพื่อกำหนดรหัสแทนเส้นตัวอักษร

ทำการคำนวณหาค่า Choice Function ($T_j^h(I^h)$) ตามสมการที่ 3.14 แล้วนำเอาค่า $T_j^h(I^h)$ มาเปรียบเทียบกับค่า $\rho_{\text{hidden}} = 0.75$ ที่กำหนดไว้แล้วทำการพิจารณาตรวจสอบค่า Vigilance Parameter ที่ Hidden Layer (ρ_{hidden}) ตามสมการที่ 3.17 ดังนี้

กรอบย่อยที่ 1: คำนวณหาค่า Choice Function ($T_2^1(I^1)$) = 0.7055

$$W_{2i}^1 = 0 \ 1.0000 \ 0 \ 0.2500 \ 0.2500 \ 0 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500$$

เนื่องจาก $T_2^1(I^1) < \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันไม่มากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ขณะนั้นได้ ต่อจากนั้นจะทำการสร้างโหนดใหม่ใน Hidden Layer และกำหนดให้มี Weight Vector ของโหนดใหม่ที่สร้างขึ้นมามีค่าเท่ากับค่า Input Vector ของช่วงกรอบส่วนย่อย h นั้น จะเรียกกรณีนี้ว่าการเกิด “Reset”

$$W_{3i}^1 = 0.5000 \ 0.5000 \ 0.6250 \ 0.5000 \ 0.5000 \ 0.5000 \ 0.5000 \ 0.5000 \ 0.5000 \ 0.5000 \ 0.5000 \ 0.5000$$

กรอบย่อยที่ 2: คำนวณหาค่า Choice Function ($T_1^2(I^2)$) = 0.9374

$$W_{1i}^2 = 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500$$

เนื่องจาก $T_1^2(I^2) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ขณะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ขณะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{1i}^2(t+1) = 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500$$

0.2500

กรอบย่อยที่ 3: คำนวณหาค่า Choice Function ($T_1^3(I^3)$) = 0.7888

$$W_{1i}^3 = 0.2500 \ 0.2500 \ 0.2500 \ 0.4875 \ 0.4750 \ 0 \ 0.2500 \ 0 \ 0 \ 0.2500 \ 0 \ 0$$

เนื่องจาก $T_1^3(I^3) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ขณะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ขณะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{1i}^3(t+1) = 0.2500 \ 0.2437 \ 0.2500 \ 0.4756 \ 0.4638 \ 0 \ 0.2500 \ 0 \ 0 \ 0.2500 \ 0 \ 0$$

กรอบย่อยที่ 4: คำนวณหาค่า Choice Function ($T_1^4(I^4)$) = 0.7491

$$W_{1i}^4 = 0.2500 \ 0 \ 0.9500 \ 0.9625 \ 0.9500 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 0.9500 \ 0$$

เนื่องจาก $T_1^4(I^4) < \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันไม่มากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ขณะนั้นได้ ต่อจากนั้นจะทำการสร้างโหนดใหม่ใน Hidden Layer และ

กำหนดให้มี Weight Vector ของโหนดใหม่ที่สร้างขึ้นมาเท่ากับค่า Input Vector ของช่วงกรอบ ส่วนย่อย h นั้น จะเรียกกรณีนี้ว่าการเกิด “Reset”

$$W_{2i}^4 = 0.2500 \ 0.1250 \ 0.1250 \ 0.1250 \ 0.1250 \ 0.1250 \ 0 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 0$$

กรอบย่อยที่ 5: คำนวณหาค่า Choice Function ($T_1^5(I^5)$) = 0.9233

$$W_{li}^5 = 0.7125 \ 0.9500 \ 0.9875 \ 0.7500 \ 0.9875 \ 0.7500 \ 0.9875 \ 0.9875 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

เนื่องจาก $T_1^5(I^5) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^5(t+1) = 0.6769 \ 0.9500 \ 0.9875 \ 0.7500 \ 0.9381 \ 0.7500 \ 0.9819 \ 0.9819 \ 0.7500 \ 0.7500 \ 0.7500 \\ 0.7500$$

กรอบย่อยที่ 6: คำนวณหาค่า Choice Function ($T_1^6(I^6)$) = 0.9932

$$W_{li}^6 = 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

เนื่องจาก $T_1^6(I^6) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^6(t+1) = 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \\ 0.7500$$

กรอบย่อยที่ 7: คำนวณหาค่า Choice Function ($T_1^7(I^7)$) = 0.9951

$$W_{li}^7 = 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

เนื่องจาก $T_1^7(I^7) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^7(t+1) = 0.7500 \ 0.7500 \ 0.7500 \ 0.7437 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \\ 0.7500$$

กรอบย่อยที่ 8: คำนวณหาค่า Choice Function ($T_1^8(I^8)$) = 0

$$W_{li}^8 = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

เนื่องจาก $W_{li}^8 = 0$ แต่ค่า Input Vector มีค่าไม่เท่ากับ 0 ดังนั้นแสดงว่า Input นั้นจะต้องทำการสร้างโหนดใหม่ใน Hidden Layer และกำหนดให้มี Weight Vector ของโหนดใหม่ที่สร้างขึ้นมามีค่าเท่ากับค่า Input Vector ของช่วงกรอบส่วนย่อย h นั้น จะเรียกกรณีนี้ว่าการเกิด “Reset”

$$W_{2i}^8 = 0.6250 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

กรอบย่อยที่ 9: คำนวณหาค่า Choice Function ($T_1^9(I^9) = 0$)

$$W_{li}^9 = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

เนื่องจากขนาดของ $W_{li}^9 = 0$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^9(t+1) = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

จากตารางที่ 3.15 แสดงค่า Weight Vector ที่ชั้น Hidden Layer ในการเรียนรู้ครั้งที่ 3 และตารางที่ 3.16 แสดงค่า Weight Vector ที่ชั้น Cluster Layer ในการเรียนรู้ครั้งที่ 3

ตารางที่ 3.15 ค่า Weight Vector ที่ชั้น Hidden Layer ในการเรียนรู้ครั้งที่ 3

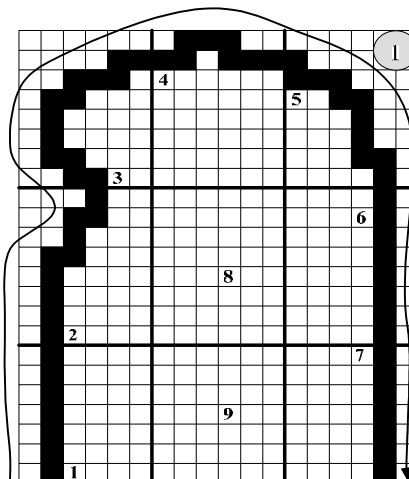
กรอบย่อยที่	Weight Vector ที่ชั้น Hidden Layer
1 (W_{3i}^1)	0.5000 0.5000 0.6250 0.5000 0.5000 0.5000 0.5000 0.5000 0.5000 0.5000 0.5000 0.5000
2 (W_{li}^2)	0.2500 0.2500 0.2500 0.2500 0.2500 0.2500 0 0.2500 0.2500 0.2500 0.2500 0.2500
3 (W_{li}^3)	0.2500 0.2437 0.2500 0.4756 0.4638 0 0.2500 0 0 0.2500 0 0
4 (W_{2i}^4)	0.2500 0.1250 0.1250 0.1250 0.1250 0.1250 0 1.0000 1.0000 1.0000 1.0000 0
5 (W_{li}^5)	0.6769 0.9500 0.9875 0.7500 0.9381 0.7500 0.9819 0.9819 0.7500 0.7500 0.7500 0.7500
6 (W_{li}^6)	0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500
7 (W_{li}^7)	0.7500 0.7500 0.7500 0.7437 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500
8 (W_{2i}^8)	0.6250 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.75 0.75 0.75
9 (W_{li}^9)	0 0 0 0 0 0 0 0 0 0 0 0

ตารางที่ 3.16 ค่า Weight Vector ที่ชั้น Cluster Layer ในการเรียนรู้ครั้งที่ 3

กรอบย่อยที่	Weight Vector ที่ชั้น Cluster Layer
1 (W_{31})	0.5000 0.5000 0.6250 0.5000 0.5000 0.5000 0.5000 0.5000 0.5000 0.5000 0.5000 0.5000
2 (W_{32})	0.2500 0.2500 0.2500 0.2500 0.2500 0.2500 0 0.2500 0.2500 0.2500 0.2500 0.2500
3 (W_{33})	0.2500 0.2437 0.2500 0.4756 0.4638 0 0.2500 0 0 0.2500 0 0
4 (W_{34})	0.2500 0.1250 0.1250 0.1250 0.1250 0.1250 0 1.0000 1.0000 1.0000 1.0000 0
5 (W_{35})	0.6769 0.9500 0.9875 0.7500 0.9381 0.7500 0.9819 0.9819 0.7500 0.7500 0.7500 0.7500
6 (W_{36})	0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500
7 (W_{37})	0.7500 0.7500 0.7500 0.7437 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500
8 (W_{38})	0.6250 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500
9 (W_{39})	0 0 0 0 0 0 0 0 0 0 0 0

เมื่อทำการพิจารณา Choice Function ตามสมการที่ 3.20 จะได้ค่า C_K ออกมา จากนั้นทำการเปรียบเทียบ $C_K \geq \rho_{\text{cluster}}$ หรือไม่ ตามสมการที่ 3.22 ทั้งนี้ $C_1 = 0.8134$ ซึ่งทำให้สมการที่ 3.22 เป็นเท็จหรือเรียกว่าเป็นกรณี “Reset” แสดงว่าโหนดที่เลือกให้เป็นโหนดชนะ $K=1$ ที่ Cluster Layer นั้นมีการจัดเข้ากลุ่มที่ไม่ถูกต้องตรงตาม กลุ่ม Output Class ของ Input Pattern จะมีการสร้างโหนดใหม่ขึ้นมาใน Cluster Layer โดยที่ $K = 3$ และทำการกำหนด Weight Vector ของโหนดใหม่ (W_{kh}) ให้มีค่าเท่ากับค่า Weight Vector (W_j) ที่ส่งต่อขึ้นมาจาก Hidden Layer ตามสมการที่ 3.25 หลังจากนั้นจะมีการสร้างการเชื่อมต่อจากโหนดใหม่ใน Cluster Layer ไปยังโหนดใน Output Layer ที่ตรงตาม Target Output ของ Input Pattern ที่ทำการเรียนรู้ครั้งนั้นคือ Target Output = “ถ”

การเรียนรู้ครั้งที่ 4 รับค่ารูปแบบตัวอักษรที่มีการกำหนด Input Vector ตามตารางที่ 3.8 โดยรูปแบบของตัวอักษรที่มีการแบ่งออกเป็น 9 กรอบย่อยแสดงดังรูปที่ 3.30



รูปที่ 3.30 แสดงทิศทางของเส้นตัวอักษรในการ Training ตัวที่ 4 ตามการจัดเรียงกรอบย่อยของภาพตัวอักษรตามลำดับของทิศทางเส้นตัวอักษรเพื่อกำหนดรหัสแทนเส้นตัวอักษร

ทำการคำนวณหาค่า Choice Function ($T_j^h(I^h)$) ตามสมการที่ 3.14 แล้วนำเอาค่า $T_j^h(I^h)$ มาเปรียบเทียบกับค่า $\rho_{\text{hidden}} = 0.75$ ที่กำหนดไว้แล้วทำการพิจารณาตรวจสอบค่า Vigilance Parameter ที่ Hidden Layer (ρ_{hidden}) ตามสมการที่ 3.17 ดังนี้

กรอบย่อยที่ 1: คำนวณหาค่า Choice Function ($T_1^1(I^1)$) = 0.9608

$$W_{li}^1 = 0.2500 \ 0.2500 \ 0 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500$$

เนื่องจาก $T_1^1(I^1) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^1(t+1) = 0.2500 \ 0.2500 \ 0 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.25 \ 0.25 \ 0.25$$

กรอบย่อยที่ 2: คำนวณหาค่า Choice Function ($T_1^2(I^2)$) = 0.8704

$$W_{li}^2 = 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500$$

เนื่องจาก $T_1^2(I^2) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^2(t+1) = \begin{bmatrix} 0.2500 & 0.2500 & 0.2500 & 0.2500 & 0.2375 & 0.2500 & 0 & 0.2375 & 0.2500 & 0.2500 & 0.2500 \\ 0.2500 & 0.2500 & & & & & & & & & \end{bmatrix}$$

กรอบย่อยที่ 3: คำนวณหาค่า Choice Function ($T_1^3(I^3)$) = 0.7486

$$W_{li}^3 = 0.2500 \ 0.2437 \ 0.2500 \ 0.4756 \ 0.4638 \ 0 \ 0.2500 \ 0 \ 0 \ 0.2500 \ 0 \ 1 \ 0$$

เนื่องจาก $T_1^3(I^3) < \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันไม่มากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการสร้างโหนดใหม่ใน Hidden Layer และกำหนดให้มี Weight Vector ของโหนดใหม่ที่สร้างขึ้นมาเท่ากับค่า Input Vector ของช่วงกรอบส่วนย่อย h นั้น จะเรียกกรณีนี้ว่าการเกิด “Reset”

$$W_{2i}^3 = 0.5000 \ 0.2500 \ 0.5000 \ 0.2500 \ 0.2500 \ 0.2500 \ 0 \ 0.2500 \ 0 \ 0 \ 0.2500 \ 0 \ 0$$

กรอบย่อยที่ 4: คำนวณค่า Choice Function ($T_1^4(I^4)$) = 0.9901

$$W_{ij}^4 = 0.2500 \ 0 \ 0.9500 \ 0.9625 \ 0.9500 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 0.9500 \ 0 \ 0$$

เนื่องจาก $T_1^4(I^4) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{ji}^4(t+1) = 0.2375 \ 0 \ 0.9025 \ 0.9144 \ 0.9400 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 0.9500 \ 0 \ 0$$

กรอบย่อยที่ 5: คำนวณค่า Choice Function ($T_1^5(I^5)$) = 0.9900

$$\mathbf{W}_{\text{li}}^5 = \begin{bmatrix} 0.6769 & 0.9500 & 0.9875 & 0.7500 & 0.9381 & 0.7500 & 0.9819 & 0.9819 & 0.7500 & 0.7500 & 0.7500 & 0.7500 \\ 0.7500 & & & & & & & & & & & \end{bmatrix}$$

เนื่องจาก $T_1^5(I^5) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้า
 รมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียก
 กรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^5(t+1) = 0.6769 \ 0.9500 \ 0.9875 \ 0.7500 \ 0.9381 \ 0.7500 \ 0.9703 \ 0.9703 \ 0.7500 \ 0.7500 \ 0.7500$$

กรอปปอยที่ 6: คำนวณหาค่า Choice Function $(T_1^6(I^6)) = 1$

$$\mathbf{W}_{\text{li}}^6 = \begin{bmatrix} 0.7500 & 0.7500 & 0.7500 & 0.7500 & 0.7500 & 0.7500 & 0.7500 & 0.7500 & 0.7500 & 0.7500 & 0.7500 & 0.7500 \\ 0.7500 & & & & & & & & & & & \end{bmatrix}$$

0.7500

เนื่องจาก $T_1^6(I^6) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^6(t+1) = 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

กรอบย่อยที่ 7: คำนวณหาค่า Choice Function ($T_1^7(I^7) = 1$)

$$W_{li}^7 = 0.7500 \ 0.7500 \ 0.7500 \ 0.7437 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

เนื่องจาก $T_1^7(I^7) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^7(t+1) = 0.7500 \ 0.7500 \ 0.7500 \ 0.7437 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

กรอบย่อยที่ 8: คำนวณหาค่า Choice Function ($T_1^8(I^8) = 0$)

$$W_{li}^8 = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

เนื่องจากขนาดของ $W_{li}^8 = 0$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^8(t+1) = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

กรอบย่อยที่ 9: คำนวณหาค่า Choice Function ($T_1^9(I^9) = 0$)

$$W_{li}^9 = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

เนื่องจากขนาดของ $W_{li}^9 = 0$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^9(t+1) = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

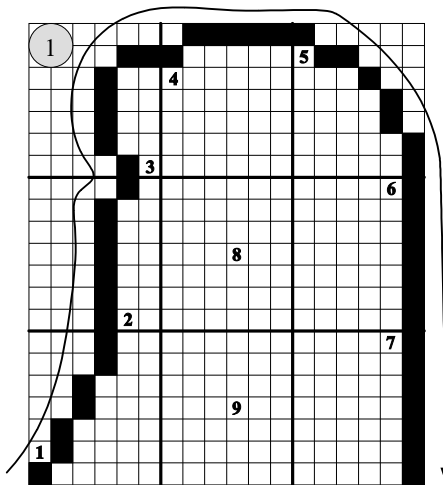
จากตารางที่ 3.17 แสดงค่า Weight Vector ที่ชั้น Hidden Layer ในการเรียนรู้ครั้งที่ 4 และ ตารางที่ 3.18 แสดงค่า Weight Vector ที่ชั้น Cluster Layer ในการเรียนรู้ครั้งที่ 4

ตารางที่ 3.18 (ต่อ)

กรอบย่อยที่	Weight Vector ที่ชั้น Cluster Layer
8 (W_{18})	0 0 0 0 0 0 0 0 0 0 0 0 0
9 (W_{19})	0 0 0 0 0 0 0 0 0 0 0 0 0

เมื่อทำการพิจารณา Choice Function ตามสมการที่ 3.20 จะได้ค่า C_K ออกมา จากนั้นทำการเปรียบเทียบ $C_K \geq \rho_{\text{cluster}}$ หรือไม่ ตามสมการที่ 3.22 ทั้งนี้ $C_1 = 0.9448$ ซึ่งทำให้สมการที่ 3.22 เป็นจริงหรือเรียกว่าเกิดกรณี “Resonance” แสดงว่า Weight Vector (W_J) นั้นมีความคล้ายคลึงกับ Weight Vector (W_k) ของโหนดที่เลือกให้เป็นโหนดที่ชนะคือโหนด $K=1$ ใน Cluster Layer และตรงกับกลุ่มผลลัพธ์ของ Input Pattern นอกจากนี้จะพิจารณาว่า W_{2i}^3 นั้นเป็นค่า Weight Vector ที่ส่งค่ามาจากโหนด $J=2$ ที่มีการสร้างขึ้นใหม่ใน Hidden Layer ก็จะมีการปรับค่า Weight Vector ($W_{13}(t+1)$) ตามสมการที่ 3.23 หลังจากนั้นจะมีการสร้างการเชื่อมต่อจากโหนด $K=1$ ใน Cluster Layer ไปยังโหนดใน Output Layer ที่ตรงตาม Target Output ของ Input Pattern ที่ทำการเรียนรู้ครั้งนั้นคือ Target Output = “ก”

การเรียนรู้ครั้งที่ 5 รับค่ารูปแบบตัวอักษรที่มีการกำหนด Input Vector ตามตารางที่ 3.9 โดยรูปแบบของตัวอักษรที่มีการแบ่งออกเป็น 9 กรอบย่อยแสดงดังรูปที่ 3.31



รูปที่ 3.31 แสดงทิศทางของเส้นตัวอักษรในการ Training ตัวที่ 5 ตามการจัดเรียงกรอบย่อยของภาพตัวอักษรตามลำดับของทิศทางเส้นตัวอักษรเพื่อกำหนดรหัสแทนเส้นตัวอักษร

ทำการคำนวณหาค่า Choice Function ($T_j^h(I^h)$) ตามสมการที่ 3.14 แล้วนำเอาค่า $T_j^h(I^h)$ มาเปรียบเทียบกับค่า $\rho_{\text{hidden}} = 0.75$ ที่กำหนดไว้ แล้วทำการพิจารณาตรวจสอบค่า Vigilance Parameter ที่ Hidden Layer (ρ_{hidden}) ตามสมการที่ 3.17 ดังนี้

กรอบย่อยที่ 1: คำนวณหาค่า Choice Function ($T_2^1(I^1)$) = 0.9685

$$W_{2i}^1 = 0 \ 1.0000 \ 0 \ 0.2500 \ 0.2500 \ 0 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500$$

เนื่องจาก $T_2^1(I^1) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{2i}^1(t+1) = 0 \ 0.9625 \ 0 \ 0.2500 \ 0.2437 \ 0 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500$$

กรอบย่อยที่ 2: คำนวณหาค่า Choice Function ($T_1^2(I^2)$) = 0.9480

$$W_{1i}^2 = 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2375 \ 0.2500 \ 0 \ 0.2375 \ 0.2500 \ 0.2500 \ 0.2500$$

เนื่องจาก $T_1^2(I^2) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{1i}^2(t+1) = 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2375 \ 0.2437 \ 0 \ 0.2375 \ 0.2500 \ 0.2500 \ 0.2500$$

กรอบย่อยที่ 3: คำนวณหาค่า Choice Function ($T_2^3(I^3)$) = 0.8093

$$W_{2i}^3 = 0.5000 \ 0.2500 \ 0.5000 \ 0.2500 \ 0.2500 \ 0.2500 \ 0 \ 0.2500 \ 0 \ 0 \ 0.2500 \ 0 \ 0$$

เนื่องจาก $T_2^3(I^3) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{2i}^3(t+1) = 0.4937 \ 0.2500 \ 0.4875 \ 0.2500 \ 0.2437 \ 0.2375 \ 0 \ 0.2375 \ 0 \ 0 \ 0.2500 \ 0 \ 0$$

กรอบย่อยที่ 4: คำนวณหาค่า Choice Function ($T_1^4(I^4)$) = 0.9473

$$W_{1i}^4 = 0.2375 \ 0 \ 0.9025 \ 0.9144 \ 0.9400 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 0.9500 \ 0 \ 0$$

เนื่องจาก $T_1^4(I^4) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^4(t+1) = 0.2319 \ 0 \ 0.9025 \ 0.9144 \ 0.9400 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 0.9500 \ 0 \ 0$$

กรอบย่อยที่ 5: คำนวณหาค่า Choice Function ($T_1^5(I^5)$) = 0.9880

$$W_{li}^5 = 0.6769 \ 0.9500 \ 0.9875 \ 0.7500 \ 0.9381 \ 0.7500 \ 0.9703 \ 0.9703 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

เนื่องจาก $T_1^5(I^5) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^5(t+1) = 0.6769 \ 0.9500 \ 0.9819 \ 0.7500 \ 0.9287 \ 0.7500 \ 0.9593 \ 0.9593 \ 0.7500 \ 0.7500 \ 0.7500$$

กรอบย่อยที่ 6: คำนวณหาค่า Choice Function ($T_1^6(I^6)$) = 1

$$W_{li}^6 = 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

เนื่องจาก $T_1^6(I^6) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^6(t+1) = 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

กรอบย่อยที่ 7: คำนวณหาค่า Choice Function ($T_1^7(I^7)$) = 1

$$W_{li}^7 = 0.7500 \ 0.7500 \ 0.7500 \ 0.7437 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

เนื่องจาก $T_1^7(I^7) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^7(t+1) = 0.7500 \ 0.7500 \ 0.7500 \ 0.7437 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

กรอบย่อยที่ 8: คำนวณหาค่า Choice Function ($T_1^8(I^8) = 0$)

$$W_{li}^8 = 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0$$

เนื่องจากขนาดของ $W_{li}^8 = 0$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^8(t+1) = 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0$$

กรอบย่อยที่ 9: คำนวณหาค่า Choice Function ($T_1^9(I^9) = 0$)

$$W_{li}^9 = 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0$$

เนื่องจากขนาดของ $W_{li}^9 = 0$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^9(t+1) = 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0$$

จากตารางที่ 3.19 แสดงค่า Weight Vector ที่ชั้น Hidden Layer ในการเรียนรู้ครั้งที่ 5 และตารางที่ 3.20 แสดงค่า Weight Vector ที่ชั้น Cluster Layer ในการเรียนรู้ครั้งที่ 5

ตารางที่ 3.19 ค่า Weight Vector ที่ชั้น Hidden Layer ในการเรียนรู้ครั้งที่ 5

กรอบย่อยที่	Weight Vector ที่ชั้น Hidden Layer
1 (W_{2i}^1)	0 0.9625 0 0.2500 0.2437 0 0.2500 0.2500 0.2500 0.2500 0.2500
2 (W_{li}^2)	0.2500 0.2500 0.2500 0.2500 0.2375 0.2437 0 0.2375 0.2500 0.2500 0.2500 0.2500 0.2500
3 (W_{2i}^3)	0.4937 0.2500 0.4875 0.2500 0.2437 0.2375 0 0.2375 0 0 0.2500 0 0
4 (W_{li}^4)	0.2319 0 0.9025 0.9144 0.9400 1.0000 1.0000 1.0000 1.0000 1.0000 0.9500 0 0
5 (W_{li}^5)	0.6769 0.9500 0.9819 0.7500 0.9287 0.7500 0.9593 0.9593 0.7500 0.7500 0.7500 0.7500 0.7500
6 (W_{li}^6)	0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500
7 (W_{li}^7)	0.7500 0.7500 0.7500 0.7437 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500
8 (W_{li}^8)	0 0 0 0 0 0 0 0 0 0 0 0 0

ตารางที่ 3.19 (ต่อ)

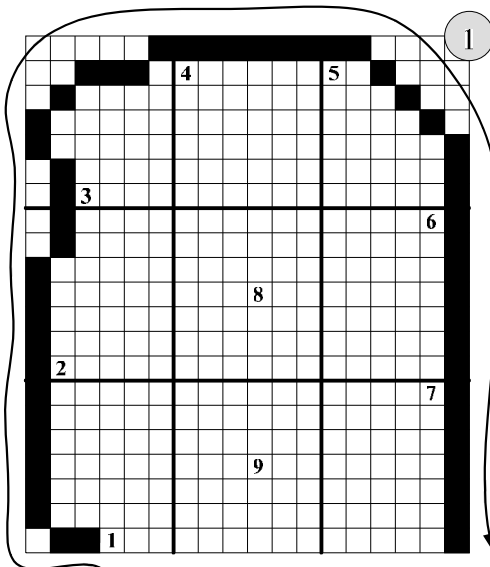
กรอบย่อยที่	Weight Vector ที่ชั้น Hidden Layer
9 (W_{li}^9)	0 0 0 0 0 0 0 0 0 0 0 0 0 0

ตารางที่ 3.20 ค่า Weight Vector ที่ชั้น Cluster Layer ในการเรียนรู้ครั้งที่ 5

กรอบย่อยที่	Weight Vector ที่ชั้น Cluster Layer
1 (W_{41})	0 0.9625 0 0.2500 0.2437 0 0.2500 0.2500 0.2500 0.2500 0.2500 0.2500
2 (W_{42})	0.2500 0.2500 0.2500 0.2500 0.2375 0.2437 0 0.2375 0.2500 0.2500 0.2500 0.2500 0.2500
3 (W_{43})	0.4937 0.2500 0.4875 0.2500 0.2437 0.2375 0 0.2375 0 0 0.2500 0 0
4 (W_{44})	0.2319 0 0.9025 0.9144 0.9400 1.0000 1.0000 1.0000 1.0000 1.0000 0.9500 0 0
5 (W_{45})	0.6769 0.9500 0.9819 0.7500 0.9287 0.7500 0.9593 0.9593 0.7500 0.7500 0.7500 0.7500 0.7500
6 (W_{46})	0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500
7 (W_{47})	0.7500 0.7500 0.7500 0.7437 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500
8 (W_{48})	0 0 0 0 0 0 0 0 0 0 0 0 0 0
9 (W_{49})	0 0 0 0 0 0 0 0 0 0 0 0 0 0

เมื่อทำการพิจารณา Choice Function ตามสมการที่ 3.20 จะได้ค่า C_K ออกมา จากนั้นทำการเปรียบเทียบ $C_K \geq \rho_{cluster}$ หรือไม่ ตามสมการที่ 3.22 ทั้งนี้ $C_1 = 0.8319$ ซึ่งทำให้สมการที่ 3.22 เป็นเท็จหรือเรียกว่าเป็นกรณี “Reset” แสดงว่าโหนดที่เลือกให้เป็นโหนดชนะ $K=1$ ที่ Cluster Layer นั้นมีการจัดเข้ากลุ่มที่ไม่ถูกต้องตรงตาม กลุ่ม Output Class ของ Input Pattern จะมีการสร้างโหนดใหม่ขึ้นมาใน Cluster Layer โดยที่ $K = 4$ และทำการกำหนด Weight Vector ของโหนดใหม่ (W_{kh}) ให้มีค่าเท่ากับค่า Weight Vector (W_j) ที่ส่งต่อขึ้นมาจาก Hidden Layer ตามสมการที่ 3.25 หลังจากนั้นจะมีการสร้างการเชื่อมต่อจากโหนดใหม่ใน Cluster Layer ไปยังโหนดใน Output Layer ที่ตรงตาม Target Output ของ Input Pattern ที่ทำการเรียนรู้ครั้งนั้นคือ Target Output = “ภ”

การเรียนรู้ครั้งที่ 6 รับค่ารูปแบบตัวอักษรที่มีการกำหนด Input Vector ตามตารางที่ 3.10 โดยรูปแบบของตัวอักษรที่มีการแบ่งออกเป็น 9 กรอบย่อยแสดงดังรูปที่ 3.32



รูปที่ 3.32 แสดงทิศทางของเส้นตัวอักษรในการ Training ตัวที่ 6 ตามการจัดเรียงกรอบย่อยของ ภาพตัวอักษรตามลำดับของทิศทางเส้นตัวอักษรเพื่อกำหนดรหัสแทนเส้นตัวอักษร

ทำการคำนวณหาค่า Choice Function ($T_j^h(I^h)$) ตามสมการที่ 3.14 แล้วนำเอาค่า $T_j^h(I^h)$ มาเปรียบเทียบกับค่า $\rho_{\text{hidden}} = 0.75$ ที่กำหนดไว้แล้วทำการพิจารณาตรวจสอบค่า Vigilance Parameter ที่ Hidden Layer (ρ_{hidden}) ตามสมการที่ 3.17 ดังนี้

กรอบย่อยที่ 1: คำนวณหาค่า Choice Function ($T_3^1(I^1)$) = 0.9624

$$W_{3i}^1 = 0.5000 \ 0.5000 \ 0.6250 \ 0.5000 \ 0.5000 \ 0.5000 \ 0.5000 \ 0.5000 \ 0.5000 \ 0.5000 \ 0.5000 \ 0.5000$$

เนื่องจาก $T_3^1(I^1) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{3i}^1(t+1) = 0.5000 \ 0.4937 \ 0.6062 \ 0.4875 \ 0.4875 \ 0.4875 \ 0.4875 \ 0.4875 \ 0.4875 \ 0.4875 \ 0.4875 \ 0.4875$$

กรอบย่อยที่ 2: คำนวณหาค่า Choice Function ($T_1^2(I^2)$) = 0.9500

$$W_{1i}^2 = 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2375 \ 0.2437 \ 0 \ 0.2375 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2500$$

$$0.2500$$

เนื่องจาก $T_1^2(I^2) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^2(t+1) = 0.2500 \ 0.2500 \ 0.2500 \ 0.2500 \ 0.2319 \ 0.2437 \ 0 \ 0.2375 \ 0.2500 \ 0.2500 \ 0.2500 \\ 0.2500 \ 0.2500$$

กรอบย่อยที่ 3: คำนวณหาค่า Choice Function ($T_2^3(I^3)$) = 0.8396

$$W_{2i}^3 = 0.4937 \ 0.2500 \ 0.4875 \ 0.2500 \ 0.2437 \ 0.2375 \ 0 \ 0.2375 \ 0 \ 0 \ 0.2500 \ 0 \ 0$$

เนื่องจาก $T_2^3(I^3) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{2i}^3(t+1) = 0.4816 \ 0.2500 \ 0.4756 \ 0.2437 \ 0.2378 \ 0.2256 \ 0 \ 0.2319 \ 0 \ 0 \ 0.2375 \ 0 \ 0$$

กรอบย่อยที่ 4: คำนวณหาค่า Choice Function ($T_1^4(I^4)$) = 0.8936

$$W_{li}^4 = 0.2375 \ 0 \ 0.9025 \ 0.9144 \ 0.9400 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 1.0000 \ 0.9500 \ 0 \ 0$$

เนื่องจาก $T_1^4(I^4) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^4(t+1) = 0.2256 \ 0 \ 0.9025 \ 0.9144 \ 0.9400 \ 1.0000 \ 1.0000 \ 1.0000 \ 0.9500 \ 0.9500 \ 0.9025 \ 0 \ 0$$

กรอบย่อยที่ 5: คำนวณหาค่า Choice Function ($T_1^5(I^5)$) = 0.9683

$$W_{li}^5 = 0.6769 \ 0.9500 \ 0.9819 \ 0.7500 \ 0.9287 \ 0.7500 \ 0.9593 \ 0.9593 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \\ 0.7500$$

เนื่องจาก $T_1^5(I^5) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^5(t+1) = 0.6430 \ 0.9462 \ 0.9765 \ 0.7500 \ 0.9260 \ 0.7500 \ 0.9488 \ 0.9488 \ 0.7500 \ 0.7500 \ 0.7500 \\ 0.7500 \ 0.7500$$

กรอบย่อยที่ 6: คำนวณหาค่า Choice Function ($T_1^6(I^6)$) = 1

$$W_{li}^6 = 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

เนื่องจาก $T_1^6(I^6) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^6(t+1) = 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

กรอบย่อยที่ 7: คำนวณหาค่า Choice Function ($T_1^7(I^7) = 1$)

$$W_{li}^7 = 0.7500 \ 0.7500 \ 0.7500 \ 0.7437 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

เนื่องจาก $T_1^7(I^7) > \rho_{\text{hidden}}$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^7(t+1) = 0.7500 \ 0.7500 \ 0.7500 \ 0.7437 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500 \ 0.7500$$

กรอบย่อยที่ 8: คำนวณหาค่า Choice Function ($T_1^8(I^8) = 0$)

$$W_{li}^8 = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

เนื่องจากขนาดของ $W_{li}^8 = 0$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^8(t+1) = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

กรอบย่อยที่ 9: คำนวณหาค่า Choice Function ($T_1^9(I^9) = 0$)

$$W_{li}^9 = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

เนื่องจากขนาดของ $W_{li}^9 = 0$ แสดงว่า Input นั้นมีความคล้ายคลึงกันมากพอที่จะจัดเข้ารวมอยู่กับโหนดที่ชนะนั้นได้ ต่อจากนั้นจะทำการปรับ Weight Vector ของโหนดที่ชนะ จะเรียกกรณีนี้ว่าการเกิด “Resonance”

$$W_{li}^9(t+1) = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

จากตารางที่ 3.21 แสดงค่า Weight Vector ที่ชั้น Hidden Layer ในการเรียนรู้ครั้งที่ 6 และ ตารางที่ 3.22 แสดงค่า Weight Vector ที่ชั้น Cluster Layer ในการเรียนรู้ครั้งที่ 6

ตารางที่ 3.21 ค่า Weight Vector ที่ชั้น Hidden Layer ในการเรียนรู้ครั้งที่ 6

กรอบย่อยที่	Weight Vector ที่ชั้น Hidden Layer
1 (W_{3i}^1)	0.5000 0.4937 0.6062 0.4875 0.4875 0.4875 0.4875 0.4875 0.4875 0.4875 0.4875 0.4875
2 (W_{li}^2)	0.2500 0.2500 0.2500 0.2500 0.2319 0.2437 0 0.2375 0.2500 0.2500 0.2500 0.2500 0.2500
3 (W_{2i}^3)	0.4816 0.2500 0.4756 0.2437 0.2378 0.2256 0 0.2319 0 0 0.2375 0 0
4 (W_{li}^4)	0.2256 0 0.9025 0.9144 0.9400 1.0000 1.0000 1.0000 0.9500 0.9500 0.9025 0 0
5 (W_{li}^5)	0.6430 0.9462 0.9765 0.7500 0.9260 0.7500 0.9488 0.9488 0.7500 0.7500 0.7500 0.7500 0.7500
6 (W_{li}^6)	0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500
7 (W_{li}^7)	0.7500 0.7500 0.7500 0.7437 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500
8 (W_{li}^8)	0 0 0 0 0 0 0 0 0 0 0 0 0
9 (W_{li}^9)	0 0 0 0 0 0 0 0 0 0 0 0 0

ตารางที่ 3.22 ค่า Weight Vector ที่ชั้น Cluster Layer ในการเรียนรู้ครั้งที่ 6

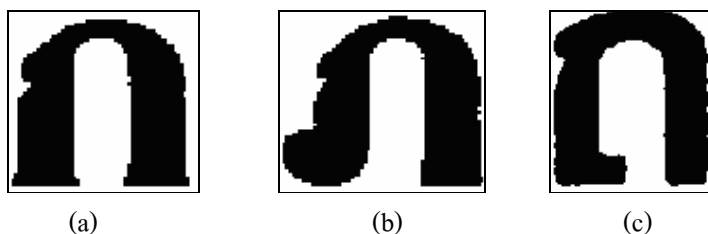
กรอบย่อยที่	Weight Vector ที่ชั้น Cluster Layer
1 (W_{31})	0.5000 0.4937 0.6062 0.4875 0.4875 0.4875 0.4875 0.4875 0.4875 0.4875 0.4875 0.4875
2 (W_{32})	0.2500 0.2500 0.2500 0.2500 0.2319 0.2437 0 0.2375 0.2500 0.2500 0.2500 0.2500 0.2500
3 (W_{33})	0.2500 0.2437 0.2500 0.4640 0.4525 0 0.2375 0 0 0.2375 0 0 0
4 (W_{34})	0.2488 0.1187 0.1250 0.1250 0.1250 0.1250 0 1.0000 0.9975 0.9975 0.9951 0 0
5 (W_{35})	0.6430 0.9462 0.9765 0.7500 0.9260 0.7500 0.9488 0.9488 0.7500 0.7500 0.7500 0.7500 0.7500

ตารางที่ 3.22 (ต่อ)

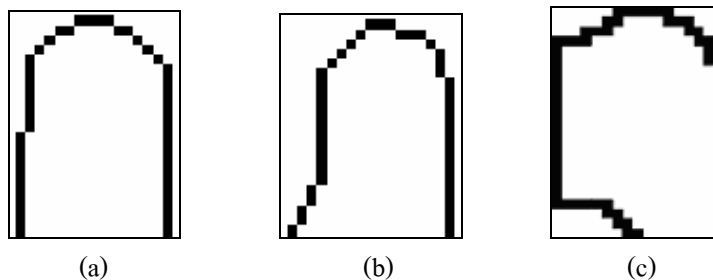
กรอบย่อยที่	Weight Vector ที่ชั้น Cluster Layer
6 (W_{36})	0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500
7 (W_{37})	0.7500 0.7500 0.7500 0.7437 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500 0.7500
8 (W_{38})	0 0 0 0 0 0 0 0 0 0 0 0 0 0
9 (W_{39})	0 0 0 0 0 0 0 0 0 0 0 0 0 0

เมื่อทำการพิจารณา Choice Function ตามสมการที่ 3.20 จะได้ค่า C_K ออกมา จากนั้นทำการเปรียบเทียบ $C_K \geq \rho_{cluster}$ หรือไม่ ตามสมการที่ 3.22 ทั้งนี้ $C_3 = 0.9148$ ซึ่งทำให้สมการที่ 3.22 เป็นจริงหรือเรียกว่าเกิดกรณี “Resonance” แสดงว่า Weight Vector (W_j) นั้นมีความคล้ายคลึงกับ Weight Vector (W_k) ของโหนดที่เลือกให้เป็นโหนดที่ชนะคือโหนด $K=3$ ใน Cluster Layer และตรงกับกลุ่มผลลัพธ์ของ Input Pattern นอกจากนี้จะพิจารณาว่า W_{2i}^3 , W_{li}^4 นั้นเป็นค่า Weight Vector ที่ส่งค่ามาจากโหนดใหม่ใน Hidden Layer ก็จะมีการปรับค่า Weight Vector ($W_{33}(t+1)$ และ $W_{34}(t+1)$) ตามสมการที่ 3.23 หลังจากนั้นจะมีการสร้างการเชื่อมต่อจากโหนด $K=3$ ใน Cluster Layer ไปยังโหนดใน Output Layer ที่ตรงตาม Target Output ของ Input Pattern ที่ทำการเรียนรู้ครั้งนั้นคือ Target Output = “ถ”

3.3.2 ข้อมูลใน Testing Set มีทั้งหมด 3 รูปแบบได้แก่ตัวอักษร “ก” “ภ” และ “ถ” ตามลำดับดังรูปที่ 3.33



รูปที่ 3.33 รูปแบบข้อมูล Testing Set (a) ตัวอักษรตัวที่ 1 คือ “ก” (b) ตัวอักษรตัวที่ 2 คือ “ภ”
(c) ตัวอักษรตัวที่ 3 คือ “ถ”



รูปที่ 3.34 ข้อมูลภาพตัวอักษรใน Testing Set ที่มีการ Resize [29, 25] และทำการ Skeleton แล้ว

(a) ตัวอักษรตัวที่ 1 คือ “ก” (b) ตัวอักษรตัวที่ 2 คือ “ภ” (c) ตัวอักษรตัวที่ 3 คือ “ถ”

ต่อจากนั้นเข้าสู่กระบวนการก่อนการรู้จำเริ่มจากการแบ่งออกเป็นรอบตัวอักษรแต่ละตัว ดังรูปที่ 3.33 แล้วทำการลดขนาดให้เท่ากับ 25x29 Pixels จากนั้นทำการ Skeleton ตัวอักษรแต่ละตัว ที่รับเข้ามา หลังจากนั้นจะได้รอบตัวอักษรที่มีการ Skeleton แล้วดังรูปที่ 3.34

ขั้นตอนต่อไปคือการแทนค่ารหัสแทนเส้นทางของภาพตัวอักษรใน Testing Set ตัวที่ 1, 2 และ 3 ตามตารางที่ 3.23, 3.24 และ 3.25 ตามลำดับ

ตารางที่ 3.23 การแทนค่ารหัสแทนเส้นทางของภาพตัวอักษรใน Testing Set ตัวที่ 1

กรอบย่อยที่	ค่าข้อมูล	Input Vector มีขนาดเท่ากับ 7
1 (I^1)	0.25 0.25 0.25 0.25 0.25 0.25 0.25	0.25 0.25 0.25 0.25 0.25 0.25 0.25
2 (I^2)	0.25 0.25 0.25 0.125 0.25 0.25 0.25	0.25 0.25 0.25 0.125 0.25 0.25 0.25
3 (I^3)	0.25 0.25 0.125 0.125 0.125 0 0.125	0.25 0.25 0.125 0.125 0.125 0 0.125
4 (I^4)	0 0 0 0.875 1	0 1 0 0.875 1 1 1
5 (I^5)	0.875 0.875 0.875 0.875 0.75 0.75	0.875 0.875 0.875 0.875 0.75 0.75 0.75
6 (I^6)	0.75 0.75 0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75
7 (I^7)	0.75 0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75
8 (I^8)	-	0 0 0 0 0 0 0
9 (I^9)	-	0 0 0 0 0 0 0

ตารางที่ 3.24 การแทนค่ารหัสแทนเส้นทางทิศทางของภาพตัวอักษรใน Testing Set ตัวที่ 2

กรอบย่อยที่	ค่าข้อมูล	Input Vector มีขนาดเท่ากับ 8
1 (I^1)	0.125 0.25 0.125 0.25 0.125 0.25	0.125 0.25 0.125 0.25 0.125 0.25 0.25 0.25
2 (I^2)	0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25	0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25
3 (I^3)	0.25 0.125 0.125 0.125	0.25 0.125 0.125 0.125 0.125 0.125 0.125 0.125
4 (I^4)	0.125 0.125 0 0 0.875 1	0.125 0.125 0 0 0.875 1 1 1
5 (I^5)	1 1 0.875 0.875 0.75 0.75 0.75 0.875	1 1 0.875 0.875 0.75 0.75 0.75 0.875
6 (I^6)	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
7 (I^7)	0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
8 (I^8)	-	0 0 0 0 0 0 0 0
9 (I^9)	-	0 0 0 0 0 0 0 0

ตารางที่ 3.25 การแทนค่ารหัสแทนเส้นทางทิศทางของภาพตัวอักษรใน Testing Set ตัวที่ 3

กรอบย่อยที่	ค่าข้อมูล	Input Vector มีขนาดเท่ากับ 13
1 (I^1)	0.5 0.25 0.5 0.25 0.5 0.25 0.5	0.5 0.25 0.5 0.25 0.5 0.25 0.5 0.5 0.5 0.5 0.5 0.5 0.5
2 (I^2)	0.5 0.5 0.5 0.5 0.25 0.25 0.25 0.25 0.25	0.5 0.5 0.5 0.5 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25
3 (I^3)	0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25	0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25 0.25
4 (I^4)	0.25 0.25 0.25 0.25 0 0 0 0.25 0 0	0.25 0.25 0.25 0.25 0 1 0 0.25 0 1 1 1 0
5 (I^5)	0.25 0 0.25 0 0 0 0 0	0.25 0 0.25 0 1 1 1 1 1 1 1 1 0

ตารางที่ 3.25 (ต่อ)

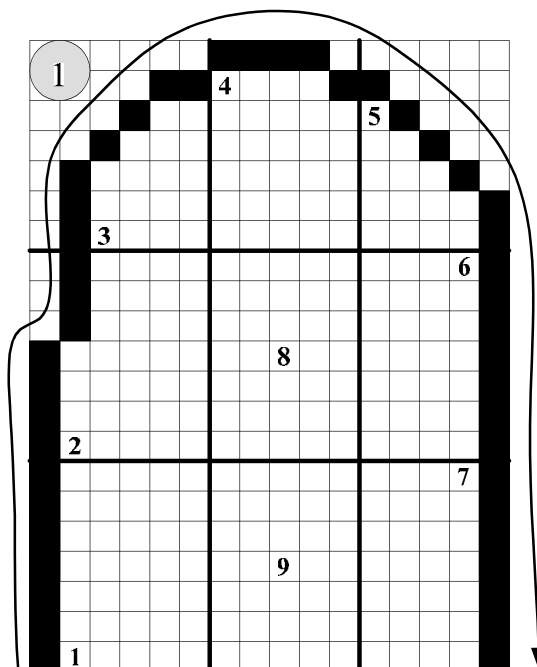
กรอบย่อยที่	ค่าข้อมูล	Input Vector มีขนาดเท่ากับ 13
6 (I^6)	0.75 1 1 0.75 1 0.75 1 0.75 0.75 1 0.75 0.75 0.75	0.75 1 1 0.75 1 0.75 1 0.75 0.75 1 0.75 0.75 0.75
7 (I^7)	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
8 (I^8)	0.75 0.75 0.75 0.75 0.75 0.75 0.75	0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75 0.75
9 (I^9)	-	0 0 0 0 0 0 0 0 0 0 0 0 0

จะสังเกตเห็นว่าทั้ง 3 รูปแบบภาพตัวอักษรใน Testing Set ที่รับเข้ามาเมื่อหาจุดศูนย์กลางของกรอบตัวอักษรแล้วอยู่ในกลุ่มระดับกลาง จากนั้นจะส่งรูปแบบตัวอักษรแต่ละตัวเข้าสู่ระบบการรู้จำแนวทางใหม่ในกระบวนการรู้จำตัวอักษรภาษาไทยด้วยวิธีใหม่ที่น่าสนใจเพื่อทดสอบระบบ

กระบวนการรู้จำในขั้นตอนของการ Testing

ในกระบวนการทดสอบระบบการรู้จำตัวอักษรที่ผ่านกระบวนการเรียนรู้ข้างต้นนี้เป็นการตรวจสอบว่าระบบที่ผ่านการเรียนรู้มาแล้วนั้นสามารถที่จะรู้จำผลลัพธ์ออกมาได้มากน้อยเพียงใด ซึ่งในการหาคำตอบของระบบทำได้โดยการคำนวณหาค่า Choice Function ($T_j^h(I^h)$) ตามสมการที่ 3.14 แล้วนำเอาค่า $T_j^h(I^h)$ ของโหนด J ที่ให้ค่ามากที่สุดออกมา และจะเลือกให้โหนด J นั้นเป็นโหนดที่ชนะ จากนั้นจะส่งต่อค่า Weight Vector ของโหนดที่ชนะใน Hidden Layer ต่อไปให้เป็นค่า Input Vector สำหรับ Cluster Layer แล้วก็ทำการคำนวณหาค่า Choice Function (C_k) ตามสมการที่ 3.20 แล้วนำเอาค่า C_k ของโหนด K ที่ให้ค่ามากที่สุดออกมา และจะเลือกให้โหนด K นั้นเป็นโหนดที่ชนะ แล้วเปรียบเทียบกับคำตอบที่แท้จริงกับผลลัพธ์ที่ได้จากระบบ

การทดสอบครั้งที่ 1 รับค่ารูปแบบตัวอักษรที่มีการกำหนด Input Vector ตามตารางที่ 3.23 โดยรูปแบบของตัวอักษรที่มีการแบ่งออกเป็น 9 กรอบย่อยแสดงดังรูปที่ 3.35



รูปที่ 3.35 แสดงทิศทางของเส้นตัวอักษรในการ Testing ตัวที่ 1 ตามการจัดเรียงกรอบย่อยของภาพ
ตัวอักษรตามลำดับของทิศทางเส้นตัวอักษรเพื่อกำหนดรหัสแทนเส้นตัวอักษร

กรอบย่อยที่ 1: มีค่า Maximum Choice Function = 0.9979 ที่ $J = 1$

กรอบย่อยที่ 2: มีค่า Maximum Choice Function = 0.9477 ที่ $J = 1$

กรอบย่อยที่ 3: มีค่า Maximum Choice Function = 0.7626 ที่ $J = 2$

กรอบย่อยที่ 4: มีค่า Maximum Choice Function = 0.7464 ที่ $J = 1$

กรอบย่อยที่ 5: มีค่า Maximum Choice Function = 0.9894 ที่ $J = 1$

กรอบย่อยที่ 6: มีค่า Maximum Choice Function = 1 ที่ $J = 1$

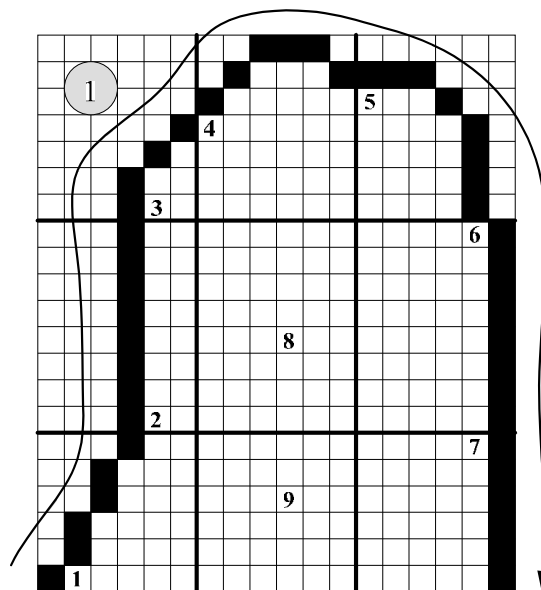
กรอบย่อยที่ 7: มีค่า Maximum Choice Function = 1 ที่ $J = 1$

กรอบย่อยที่ 8: มีค่า Maximum Choice Function = 0 ที่ $J = 1$

กรอบย่อยที่ 9: มีค่า Maximum Choice Function = 0 ที่ $J = 1$

เมื่อทำการหาค่า Choice Function ที่ Cluster Layer จะพบว่า มีค่ามากที่สุดที่ได้คือ 0.9246 ที่ $K = 1$ ซึ่งทำให้ผลจากการรู้จำคือตัวอักษร “ก” ซึ่งระบบสามารถทำการรู้จำตัวอักษรได้ถูกต้อง

การทดสอบครั้งที่ 2 รับค่ารูปแบบตัวอักษรที่มีการกำหนด Input Vector ตามตารางที่ 3.24 โดยรูปแบบของตัวอักษรที่มีการแบ่งออกเป็น 9 กรอบย่อยแสดงดังรูปที่ 3.36



รูปที่ 3.36 แสดงทิศทางของเส้นตัวอักษรในการ Testing ตัวที่ 2 ตามการจัดเรียงกรอบย่อยของภาพ
ตัวอักษรตามลำดับของทิศทางเส้นตัวอักษรเพื่อกำหนดรหัสแทนเส้นตัวอักษร

กรอบย่อยที่ 1: มีค่า Maximum Choice Function = 0.9685 ที่ $J = 3$

กรอบย่อยที่ 2: มีค่า Maximum Choice Function = 0.9605 ที่ $J = 1$

กรอบย่อยที่ 3: มีค่า Maximum Choice Function = 0.8020 ที่ $J = 2$

กรอบย่อยที่ 4: มีค่า Maximum Choice Function = 0.7745 ที่ $J = 2$

กรอบย่อยที่ 5: มีค่า Maximum Choice Function = 0.9847 ที่ $J = 1$

กรอบย่อยที่ 6: มีค่า Maximum Choice Function = 1 ที่ $J = 1$

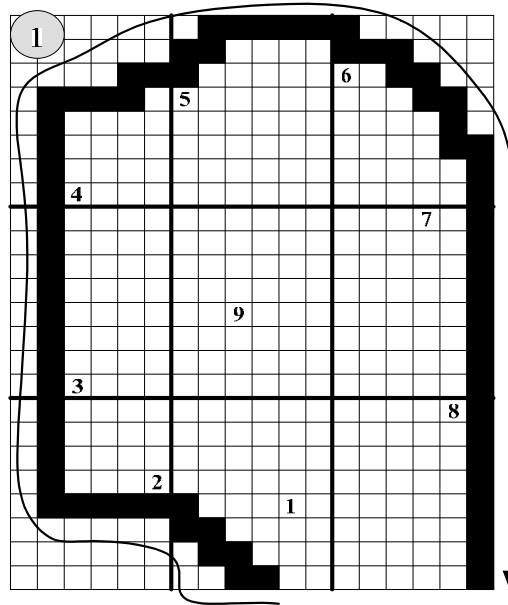
กรอบย่อยที่ 7: มีค่า Maximum Choice Function = 1 ที่ $J = 1$

กรอบย่อยที่ 8: มีค่า Maximum Choice Function = 0 ที่ $J = 1$

กรอบย่อยที่ 9: มีค่า Maximum Choice Function = 0 ที่ $J = 1$

เมื่อทำการหาค่า Choice Function ที่ Cluster Layer จะพบว่า มีค่ามากที่สุดที่ได้คือ 0.9190 ที่ $K = 4$ ซึ่งทำให้ผลจากการรู้จำคือตัวอักษร “ภ” ซึ่งระบบสามารถทำการรู้จำตัวอักษรได้ถูกต้อง

การทดสอบครั้งที่ 3 รับค่ารูปแบบตัวอักษรที่มีการกำหนด Input Vector ตามตารางที่ 3.25
โดยรูปแบบของตัวอักษรที่มีการแบ่งออกเป็น 9 กรอบย่อยแสดงดังรูปที่ 3.37



รูปที่ 3.37 แสดงทิศทางของเส้นตัวอักษรในการ Testing ตัวที่ 3 ตามการจัดเรียงกรอบย่อยของภาพ
ตัวอักษรตามลำดับของทิศทางเส้นตัวอักษรเพื่อกำหนดรหัสแทนเส้นตัวอักษร

กรอบย่อยที่ 1: มีค่า Maximum Choice Function = 0.9730 ที่ $J = 3$

กรอบย่อยที่ 2: มีค่า Maximum Choice Function = 0.4439 ที่ $J = 1$

กรอบย่อยที่ 3: มีค่า Maximum Choice Function = 0.7402 ที่ $J = 2$

กรอบย่อยที่ 4: มีค่า Maximum Choice Function = 0.7731 ที่ $J = 1$

กรอบย่อยที่ 5: มีค่า Maximum Choice Function = 0.8247 ที่ $J = 1$

กรอบย่อยที่ 6: มีค่า Maximum Choice Function = 0.9898 ที่ $J = 1$

กรอบย่อยที่ 7: มีค่า Maximum Choice Function = 1 ที่ $J = 1$

กรอบย่อยที่ 8: มีค่า Maximum Choice Function = 0.9990 ที่ $J = 2$

กรอบย่อยที่ 9: มีค่า Maximum Choice Function = 0 ที่ $J = 1$

เมื่อทำการหาค่า Choice Function ที่ Cluster Layer จะพบว่า มีค่ามากที่สุดที่ได้คือ 0.9639 ที่ $K = 3$ ซึ่งทำให้ผลจากการรู้จำคือตัวอักษร “ถ” ซึ่งระบบสามารถทำการรู้จำตัวอักษรได้ถูกต้อง