การจดจำเสียงพูด

SPEECH RECOGNITION

โดย

นายขวัญชัย คงสุข

นายพงศ์สิริ จุฑาจรัส

เลขเรียกหนังสือ ปพ ขว๔๑ ๒๕๔๒

เลขทะเบียน 041815

วัน เดือน ปี 22/11/43

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2542

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

041815

การจดจำเสียงพูด

SPEECH RECOGNITION

โดย

นายขวัญชัย คงสุข

นายพงศ์สิริ จุฑาจรัส

อาจารย์ที่ปรึกษา

อาจารย์กฤษณ์ วงรุจิระ

ดร. สุทธิชัย นพนาคีพงษ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2542

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2542

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง **การจดจำเสียงพูด**

SPEECH RECOGNITION

ผู้จัดทำ

1.นายขวัญชัย คงสุข **40013002**

2.นายพงศ์สิริ จุฑาจรัส **40013062**

อาจารย์ที่ปรึกษา

(อาจารย์กฤษณ์ วงรุจิระ)

อาจารย์ที่ปรึกษา

(ดร. สุทธิชัย นพนาคีพงษ์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การจดจำเสียงพูด

SPEECH RECOGNITION

โดย นายขวัญชัย คงสุข 40013002
นายพงศ์สิริ จุฑาจรัส 40013062

อาจารย์ที่ปรึกษา อาจารย์กฤษณ์ วงรุจิระ
ดร.สุทธิชัย นพนาคีพงษ์

บทคัดย่อ

ปัจจุบัน วิธีการติดต่อสั่งงานของมนุษย์ต่อ อุปกรณ์ เครื่องอำนวยความสะดวก ถูกพัฒนาไปอย่างต่อเนื่อง หลักการของการรู้จำเสียงพูด (Speech Recognition) ถูกนำมาใช้อย่างแพร่หลายเป็นลำดับ โครงงานนี้เป็นการศึกษา ระบบการจดจำเสียงพูด เป็นคำไม่จำกัดผู้พูดเฉพาะผู้ชาย ซึ่งประกอบด้วย ขั้นตอนการสุ่มสัญญาณเสียง แล้วเปลี่ยนเป็นสัญญาณแบบดิจิตอล ทำการตัดคำเพื่อหาขอบเขตของสัญญาณ จากนั้นนำสัญญาณดังกล่าวไปวิเคราะห์หาพารามิเตอร์ที่สำคัญ โดยใช้วิธีการทำนายเชิงเส้น(Linear Predictive Coding : LPC) พารามิเตอร์ที่ได้จะถูกปรับปรุงให้เหมาะสม เพื่อนำไปใช้กับการเปรียบเทียบค่าความน่าจะเป็น โดยวิธีฮิดเดนมาคอฟ(Hidden Markov Model) แล้วเลือกค่าความน่าจะเป็นที่มากที่สุดเป็นเสียงจากการจดจำ

ABSTRACT

In the present , way that people control the device is develop continuously. The "SPEECH RECOGNITION" is widely used for this purpose. This project is the study of recognized system in isolated independent word by man. Procedure of system compose sampling speech signal convert to digital signal. Signal, which is received, was cut the signal in order to create the range of signal by used average energy of signal technique. Then take the last result to analyze the parameters by LPC method. After that improve the parameters use to probability compare by Hidden Markov Model method choose maximum probability in recognition.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

		หน้า
บทที่ 1	บทนำ	1
บทที่ 2	ทฤษฎีหรือหลักการ	2
	2.1 ลักษณะของเสียงพูด	2
	2.2 ทฤษฎีการรู้จักจำ	3
	2.2.1 การวิเคราะห์เสียงเบื้องต้น	3
	2.2.2 การจัดระดับเวคเตอร์	11
	2.2.3 การสร้างแบบจำลอง	14
บทที่ 3	การคำนวณและการสร้าง	25
	3.1 การหาขอบเขตของคำ	25
	3.2 การนอร์มอลไลซ์	26
	3.3 การหาค่าพารามิเตอร์	26
	3.4 การสร้างโค้ดบุค	27
	3.5 การสร้างแบบจำลอง HMM	27
	3.6 การตัดสินใจ	28
บทที่ 4	การทดลองและผลการทดลอง	29
	4.1 การทดลอง	29
	4.2 ขั้นตอนการเรียนรู้	29
	4.3 ขั้นตอนการวิเคราะห์	29
	4.4 การทดสอบการจดจำเสียงพูด	53
บทที่ 5	บทวิจารณ์และบทสรุป	59
ภาคผนวก		
กิตติกรรมประกาศ		
หนังสืออ้างอิง		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

หน้า

รูปที่ 2.1 บล็อคไดอะแกรมของการรู้จำเสียง 3
รูปที่ 2.2 แสดงขั้นตอนการเตรียมสัญญาณในการวิเคราะห์ 3
รูปที่ 2.3 วงจรกรองความถี่สูงผ่าน 4
รูปที่ 2.4 การแบ่งช่วงของสัญญาณ 4
รูปที่ 2.5 แสดงวินโดว์แบบแฮมมิ่ง 5
รูปที่ 2.6 แสดงขั้นตอนในการวิเคราะห์หาคุณลักษณะของเสียง 6
รูปที่ 2.7 บล็อคไดอะแกรมแสดงโมเดลการสร้างสัญญาณเสียงพูดอย่างง่าย 7
รูปที่ 2.8 แสดงการกระจายเฟรมของเสียงพูดแต่ละจุดแทนเฟรมของเสียง 11
รูปที่ 2.9 การรวมกลุ่มของเฟรมเสียงเพื่อสร้างโค้ด X 11
รูปที่ 2.10 บล็อกไดอะแกรมของเวกเตอร์ควอนไตซ์ 12
รูปที่ 2.11 ขั้นตอนของเวกเตอร์ควอนไตเซชั่น 13
รูปที่ 2.12 แสดงแบบจำลองต่าง ๆ ของ HMM 15
รูปที่ 2.13 กระบวนการไปข้างหน้า 17
รูปที่ 2.14 กระบวนการถอยหลัง 17
รูปที่ 2.15 แสดงลำดับการคำนวณการเกิดค่าปรากฏร่วมซึ่งจะอยู่สเตท I ที่เวลา t และอยู่ที่สเตท j ที่เวลา t+1 19
รูปที่ 2.16 บล็อกไดอะแกรมการรู้จำคำโดดด้วยแบบจำลองของมาร์คอฟ 24
รูปที่ 3.1 แสดงบล็อกไดอะแกรมการจำแนกเสียงพูดโดยทั่วไป 25
รูปที่ 4.1 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขศูนย์ 30
รูปที่ 4.2 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขหนึ่ง 31
รูปที่ 4.3 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขสอง 32
รูปที่ 4.4 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขสาม 33
รูปที่ 4.5 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขสี่ 34
รูปที่ 4.6 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขห้า 35
รูปที่ 4.7 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขหก 36
รูปที่ 4.8 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขเจ็ด 37
รูปที่ 4.9 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขแปด 38
รูปที่ 4.10 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขเก้า 39
รูปที่ 4.11 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขศูนย์ 40
รูปที่ 4.12 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขหนึ่ง 41
รูปที่ 4.13 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขสอง 42

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

รูปที่ 4.14 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขสาม 43
รูปที่ 4.15 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขสี่ 44
รูปที่ 4.16 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขห้า 45
รูปที่ 4.17 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขหก 46
รูปที่ 4.18 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขเจ็ด 47
รูปที่ 4.19 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขแปด 48
รูปที่ 4.20 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขเก้า 49
รูปที่ 4.21 แสดงโค้ดบุค 32 และจำนวนรอบจากการคำนวณ 50
รูปที่ 4.22 แสดงโค้ดบุค 64 และจำนวนรอบจากการคำนวณ 51
รูปที่ 4.23 แสดงโค้ดบุค 128 และจำนวนรอบจากการคำนวณ 52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 4.1 แสดงการจดจำเสียง 0 ถึง 9 โดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 1	53
ตารางที่ 4.2 แสดงการจดจำเสียง 0 ถึง 9 โดยใช้เสียงจากบุคคลอื่น(unknown) กรณีที่ 1	54
ตารางที่ 4.3 แสดงการจดจำเสียง 0 ถึง 9 โดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 2	54
ตารางที่ 4.4 แสดงการจดจำเสียง 0 ถึง 9 โดยใช้เสียงจากบุคคลอื่น(unknown) กรณีที่ 2	55
ตารางที่ 4.5 แสดงการจดจำเสียง 0 ถึง 9 โดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 3	55
ตารางที่ 4.6 แสดงการจดจำเสียง 0 ถึง 9 โดยใช้เสียงจากบุคคลอื่น(unknown) กรณีที่ 3	56
ตารางที่ 4.7 แสดงการจดจำเสียง 0 ถึง 9 โดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 4	56
ตารางที่ 4.8 แสดงการจดจำเสียง 0 ถึง 9 โดยใช้เสียงจากบุคคลอื่น(unknown) กรณีที่ 4	57
ตารางที่ 4.9 แสดงผลค่าความถูกต้องทุกกรณี	58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

การวิเคราะห์ว่าให้คอมพิวเตอร์สามารถรู้จำเสียงพูดของมนุษย์ในสมัยก่อนนั้นทำได้ยาก และใช้เวลานาน เนื่องจากการพูดของมนุษย์ มีความซับซ้อนและมีความแตกต่างกันในแต่ละบุคคล ความเร็วในการพูดรวมถึงเสียงสูงต่ำของแต่ละคน ยากที่จะใช้เครื่องมือในการวิเคราะห์ แต่ในปัจจุบันได้มีการพัฒนาวิธีการต่าง ๆ ให้มีความสามารถในการที่จะหาตัวแทนของเสียงในรูปแบบต่าง ๆ รวมทั้งมีวิธีการเพิ่มมากขึ้น และมีความแม่นยำในการรับรู้เสียงที่ป้อนให้แก่คอมพิวเตอร์ ด้วยพื้นฐานความรู้ทางด้านคณิตศาสตร์เชิงเส้นที่สำคัญ และ พื้นฐานความรู้ทางด้านสถิติ จึงทำให้เราลดความยุ่งยากลงได้มาก

โดยธรรมชาติของเสียงนั้น เกิดจากต้นกำเนิดที่ได้จากการสั่นของอากาศ การเคลื่อนที่ของเสียงจำเป็นต้องมีตัวกลางที่จะให้เสียงอาศัยในการเคลื่อนที่เสมอ ในการเปล่งเสียงของคนเรานั้น การเกิดเสียงเกิดจากกล่องเสียงมีการสั่น ซึ่งคลื่นเสียงที่เราเปล่งออกมานั้นผ่าน ลิ้น,ฟัน,ริมฝีปาก ซึ่งถ้าเราทำปากในลักษณะที่แตกต่างกันทำให้เกิดเสียงที่แตกต่างกันด้วย และเป็นผลทำให้รูปสัญญาณเสียงเชิงเวลาก็แตกต่างกันไป

ปริญญานิพนธ์นี้ได้มีการศึกษาและทดลองนำเอาส่วนของการรู้จำเสียงที่เป็นทฤษฎีมาใช้งานจริง โดยทำในรูปการบันทึกเสียงที่จะทำการเปรียบเทียบ ผ่านไมโครโฟนซึ่งจะบันทึกรูปแบบของเสียงให้อยู่ในรูปแบบที่เหมาะสม โดยจะจำแนกเสียงตัวเลข 0-9 มีช่วงของความถี่อยู่ระหว่าง 300-3400 เฮริตซ์ และสุ่มโดยใช้ความถี่ 8000 เฮริตซ์ ซึ่งเป็นไปตามกฎของไนควิสต์ (จะบันทึกเสียงให้อยู่ในรูป ชื่อไฟล์.wav) จากนั้นนำเสียงที่ได้ไปทำการตัดคำเพื่อให้เหลือคำที่เหมาะสมแก่การวิเคราะห์ และ นำคำที่ได้ไปปรับแต่งค่าแซมปลิ้งของแต่ละคำให้เท่ากันโดยวิธีการนอร์มอลไลซ์ และเพื่อให้คำๆ นั้นมีค่าสัมประสิทธิ์ในการวิเคราะห์น้อยลง เราจะทำการแทนค่าของสัญญาณเสียงในแต่ละช่วง ด้วยสัมประสิทธิ์ แอล พี ซี (LPC) เพื่อให้ค่าที่ได้มีเสถียรภาพที่ดีขึ้นโดยเรานำไปปรับปรุงอีกขั้นหนึ่ง และจะใช้หลักการของ Hidden Markov Model (HMM) ซึ่งสามารถนำไปใช้ในการรู้จำเสียง เพื่อรับรู้ว่าเสียงที่เราป้อนให้แก่เครื่องคอมพิวเตอร์นั้นเป็นเสียงอะไร จะเห็นได้ว่าถ้าเราเลือกวิธีการทางทฤษฎีที่เหมาะสมแล้วจะทำให้ เวลาในการคำนวณหาค่าต่างๆ รวมไปถึงการรู้จำเสียงใช้เวลาที่น้อยลงตามไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีหรือหลักการ

2.1 ลักษณะของเสียงพูด

คนเราเปล่งเสียงพูดด้วยอวัยวะที่ใช้ในการออกเสียง ทำเสียงตามที่มีในระบบภาษาของตน แม้ว่าคนที่อยู่ในสังคมเดียวกันจะใช้ภาษาเดียวกันแต่ถ้าพิจารณาเสียงที่เปล่งออกมาจริงๆ แล้วแต่ละ ครั้งก็อาจจะสังเกตลักษณะที่แตกต่างกันได้ เราจึงสามารถจำเสียง จำวิธีพูดของคนที่เราคุ้นเคยได้ เสียงพูดที่จะอธิบายด้วยหลักเกณฑ์ทางวิทยาศาสตร์แม้ว่าในภาษาหนึ่ง ๆ จะมีเสียงต่างกันมากบ้างน้อยบ้าง แต่ละเสียงก็สามารถนำมาพิจารณาและอธิบายให้รู้ลักษณะการออกเสียงและตำแหน่งที่เกิดเสียงได้คำอธิบายนี้จะทำให้เข้าใจลักษณะเสียงทุกเสียง วิชาที่ว่าด้วยเสียงพูดเรียกว่า วิชาสัทศาสตร์ (Phonetics)

ในการศึกษาเสียงพูดแบ่งได้เป็น 2 ลักษณะ คือ

ก. สรีรสัทศาสตร์ (Articulatory) เป็นการศึกษาเสียงพูดจากอวัยวะและการเคลื่อนไหวอวัยวะที่ทำให้เกิดเสียงพูด การอธิบายนี้ก็จะอธิบายโดยอาศัยลักษณะและอาการเคลื่อนไหวของอวัยวะที่เกี่ยวข้องในการเปล่งเสียงพูดนั้น

ข. กลสัทศาสตร์ (Acoustic Phonetics) เป็นการศึกษาเสียงพูดจากลักษณะคลื่นเสียงที่ผู้พูดเปล่งออกมาแล้ว และผู้ฟังได้ยินว่ามีลักษณะทางกลศาสตร์อย่างไร การศึกษาตามแนวนี้ต้องอาศัยความรู้ทางฟิสิกส์และคณิตศาสตร์ช่วยอธิบายลักษณะของคลื่นเสียง

2.1.1 การเกิดของเสียง (Speech Production)

การเกิดของเสียงแบ่งออกเป็น 3 ขั้นตอน คือ

ขั้นตอนที่ 1 จุดเริ่มต้น เป็นขั้นตอนที่ลมเริ่มถูกขับออกจากปอด ผ่านเข้าไปสู่ขั้นตอนที่ 2

ขั้นตอนที่ 2 การดัดแปลงลมที่เส้นเสียงอวัยวะ ที่ใช้ในขั้นตอนนี้คือส่วนที่ต่อ จากปอดขึ้นมาถึง กล่องเสียง และที่กล่องเสียง เส้นเสียงจะทำหน้าที่ เป็นลิ้นปิดเปิดทำให้เกิดเสียง 2 ชนิด คือ

1. เสียงก้อง (Voiced) เกิดจากเส้นเสียงปิดกั้นลมไว้ ลมที่ผ่านออกมาจะเพิ่มแรงดันมากขึ้นจนเส้นเสียงปิดเปิดสลับกันไป ทำให้เกิดเสียงก้องขึ้นมา ซึ่งสามารถเรียกความถี่ในการปิดเปิดของเส้นเสียงว่าความถี่มูลฐาน

2. เสียงไม่ก้อง (Unvoiced) เสียงชนิดนี้เส้นเสียงจะเปิดตลอดเวลาที่ลมผ่าน ลมจึงผ่านออกมาได้สะดวกทำให้เกิดเสียงไม่ก้องขึ้น

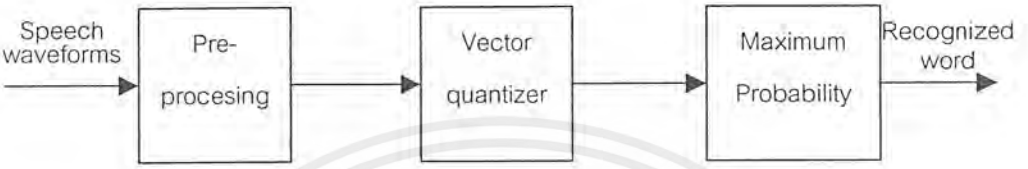
ขั้นตอนที่ 3 การเปลี่ยนแปลงลักษณะเส้นเสียง อวัยวะที่ใช้คือส่วนที่ต่อจากกล่องเสียงจนถึงริมฝีปาก โดยลมที่ผ่านออกจากกล่องเสียงจะทำให้เกิดเสียงในลักษณะต่าง ๆ ซึ่งจะเกิดจากการเปลี่ยนแปลงของอวัยวะที่กล่าวไว้ในหัวข้อที่แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ทฤษฎีการรู้จำเสียง

จากการศึกษารูปแบบการจำเสียงพูดแบบคำเดี่ยว (Isolated word recognition) โครงสร้างพื้นฐานของระบบการจดจำเสียงพูด แบ่งออกได้ 3 ส่วนได้แก่

1. การวิเคราะห์เสียงเบื้องต้น (Speech Pre-Processing)
2. การควอนไตเซชันแบบเวคเตอร์ (Vector Quantization)
3. การกำหนดแบบจำลองการรู้จำเสียง (Pattern matching)

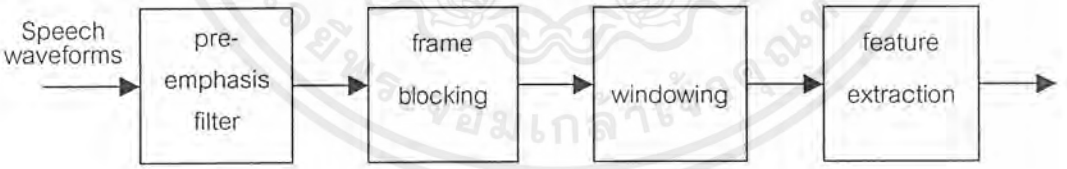


รูปที่ 2.1 บล็อคไดอะแกรมของการรู้จำเสียง

สัญญาณเข้าเป็นคลื่นเสียง ส่วนสัญญาณออกเป็นคำที่วิเคราะห์ได้จากสัญญาณเข้า ซึ่งผ่านขั้นตอนต่าง ๆ ดังต่อไปนี้

2.2.1 การวิเคราะห์เสียงเบื้องต้น (Speech Pre-Processing)

ในการวิเคราะห์สัญญาณเสียง เราจำต้องมีการเตรียม ข้อมูลการวิเคราะห์ก่อน ซึ่งขั้นตอนในการเตรียมสัญญาณเข้าเป็นดังรูปที่ 2.2



รูปที่ 2.2 แสดงขั้นตอนการเตรียมสัญญาณในการวิเคราะห์

การเตรียมสัญญาณเสียงในการวิเคราะห์ มีขั้นตอนดังต่อไปนี้

การพรีเอมฟาซีส (Preemphasis)

เนื่องจากสัญญาณเสียงพูดของมนุษย์ จะมีองค์ประกอบส่วนใหญ่อยู่บริเวณความถี่ต่ำ เมื่อเทียบกับความถี่ปฏิบัติงานไม่เกิน 5 kHz ดังนี้ เพื่อทำให้อัตราส่วนของสัญญาณเสียงต่อสัญญาณรบกวน (SNR) มีค่าคงที่ตลอดช่วงความถี่ปฏิบัติงานจึงทำการพรีเอมฟาซีส โดยเน้นให้ความถี่สูงมีขนาดสูงขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

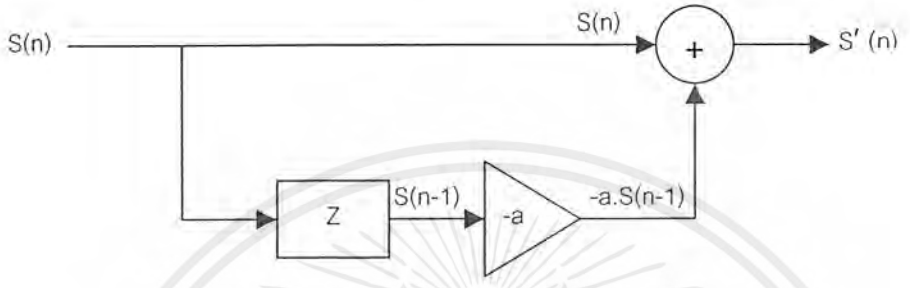
การพรีเอมฟาซีส คือ การกรอง สัญญาณด้วยวงจรกรองความถี่สูงผ่าน (High Pass Filter) ซึ่งนิยมใช้วงจรอันดับหนึ่ง ซึ่งตัวกรองเชิงเลขแบบง่ายที่สุด มีรูปแบบสมการดังนี้

$$y(n)=x(n)-ax(n-1) \qquad (2.1)$$

มีฟังก์ชันถ่ายโอนเป็น

$$H(z)=1-az^{-1} \quad ; \quad 0.9<a<1.0 \qquad (2.2)$$

สมมติว่าสัญญาณเดิมเป็น S(n) เมื่อผ่านวงจรกรองแล้วจะเป็น S′ (n)



รูปที่ 2.3 วงจรกรองความถี่สูงผ่าน

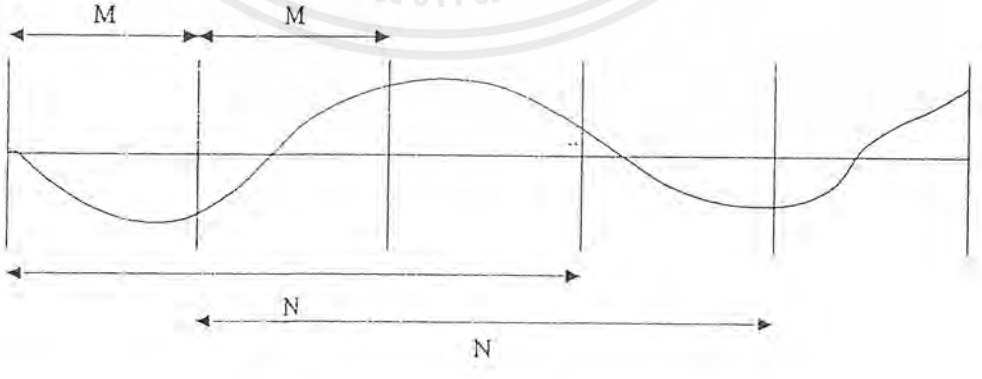
จะได้ว่า

$$S'(n)=S(n)-a.S(n-1) \qquad (2.3)$$

ยิ่งค่า a เข้าใกล้ 1 เท่าใด ความถี่สูงจะถูกขยายมากขึ้นเท่านั้น ค่า a ที่นิยมสำหรับใช้ในการหาพารามิเตอร์ของ LPC คือค่า 15/16= 0.9375

การแบ่งช่วงสัญญาณ (Frame Blocking)

สัญญาณที่ผ่านการพรีเอมฟาซีสแล้ว ถูกตัดมาวิเคราะห์ทีละเฟรม เฟรมละ N ตัวอย่าง สัญญาณในการวิเคราะห์ที่แต่ละช่วงของ N ตัวอย่าง สัญญาณนั้น จะวิเคราะห์ โดยเลื่อนเป็นระยะ M ช่วงสัญญาณ จะหมดสัญญาณเสียงที่นำมาวิเคราะห์ ดังรูปที่ 2.4 ซึ่งระยะ N ที่เหมาะสมจะต้องมีค่ามากกว่า M จะทำให้วิเคราะห์สัญญาณได้แม่นยำ แต่ถ้าค่า N น้อยเกินไปการวิเคราะห์จะล่าช้า



รูปที่ 2.4 การแบ่งช่วงของสัญญาณ

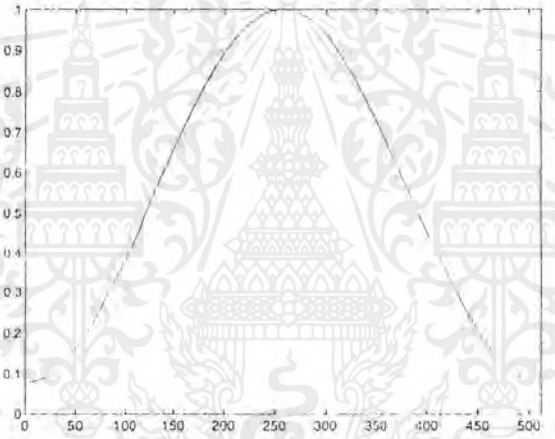
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากสัญญาณเสียงมีการเปลี่ยนแปลงคุณสมบัติตามเวลา แต่ถ้าแบ่งเสียงออกเป็นเฟรมสั้น ๆ เสียงจะมีการเปลี่ยนแปลงคุณสมบัตินิ้อยมาก ทำให้การวิเคราะห์ทำได้ง่ายขึ้น ในการทดลองนี้จึงทำการแบ่งสัญญาณเสียงออกเป็นเฟรม ๆ ละ M แซมเปิ้ล (Sample) และทำการเลื่อนเฟรมไปเรื่อย ๆ จนหมด

การวินโดว์ (Windowing)

เนื่องจากที่เราตัดสัญญาณเสียงมาวิเคราะห์ทีละเฟรมทำให้ที่ขอบของเฟรมเกิดความไม่ต่อเนื่องของสัญญาณ ถ้ามองในโดเมนความถี่สูง ดังนั้นเพื่อที่จะลดองค์ประกอบทางความสูงเหล่านี้ เราจะต้องคูณด้วยฟังก์ชันวินโดว์เพื่อลดความไม่ต่อเนื่องของสัญญาณที่ขอบ และไม่ทำให้สเปคตรัมของสัญญาณที่ช่วงความถี่ต่ำเปลี่ยนไปมากนัก ในที่นี้จะใช้ฟังก์ชันวินโดว์แฮมมิ่ง (Hamming window function) ซึ่งนิยมโดยสมการ

$$w(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right) \text{ เมื่อ } \quad n = 0, 1, 2, \ldots, N-1 \qquad (2.4)$$



รูปที่ 2.5 แสดงวินโดว์แบบแฮมมิ่ง

ในการวิเคราะห์เสียงโดยใช้ฟังก์ชันวินโดว์ จะพบว่าสัญญาณเสียงที่ผ่านการกรองโดยวินโดว์นั้นจะมีการแกว่งขึ้นลงมากน้อยขึ้นกับช่วงเวลาของวินโดว์ (ความกว้างของวินโดว์) คือถ้าช่วงของวินโดว์สั้นจะมีการแกว่งขึ้นลงอย่างรวดเร็ว และถ้าช่วงเวลาของการวินโดว์มากจะมีการแกว่งขึ้นลงช้า ๆ ดังนั้นการเลือกช่วงเวลาของการวินโดว์ ต้องให้อยู่ในช่วงที่เหมาะสม คือไม่ให้เอาท์พุทของสัญญาณแกว่งช้าหรือเร็วเกินไป อยู่ในช่วงระหว่าง 10-30 ms

เมื่อคูณกับฟังก์ชันแล้ว ก็จะได้

$$x'(n) = w(n).x(n) \qquad (2.5)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การวิเคราะห์หาคุณลักษณะเสียง (Feature extraction)

การวิเคราะห์หาคุณลักษณะของเสียง เป็นเทคนิคในการแปลงข้อมูลที่มีอยู่มากมาย ให้เป็นส่วนเล็ก ๆ ซึ่งส่วนเล็ก ๆ นี้จะแสดงคุณสมบัติของคลื่นเสียงนั้น ๆ โดยจะใช้วิธีการป้อนสัญญาณผ่านวงจรกรอง (Filter) และการประเมินเชิงเส้น (Linear Predictive Coding)

หลักการพื้นฐานของการประเมินเชิงเส้นคือ การประเมินเสียงจากผลรวมเชิงเส้นของสัญญาณเสียงในอดีต โดยอาศัยเกณฑ์กำลังสองของสัญญาณความคลาดเคลื่อน มีค่าต่ำสุดในการหาสัมประสิทธิ์การประเมินเชิงเส้น กล่าวคือหลังจากการทำพรีเอมฟาซีสและวินโดว์ครบทุกตัวในหนึ่งเฟรมแล้ว จะทำการคำนวณออโตคอร์รีเลชั่น คือที่จะทำการหาค่า สัมประสิทธิ์ LPC และอัตราขยาย G มีขั้นตอนแสดงดังรูปที่ 2.6

รูปที่ 2.6 แสดงขั้นตอนในการวิเคราะห์หาคุณลักษณะของเสียง

การคำนวณออโตคอร์รีเลชั่น (Autocorrelation)

สมมติว่าสัญญาณเดิมเป็น S(n) การประมาณ ค่าสัญญาณเป็น S′(n) ดังนั้นสามารถอธิบายการประมาณเชิงเส้นด้วยสมการดังนี้

$$S'(n) = \sum_{k=1}^{p} \alpha_k S(n-k) \qquad (2.6)$$

เมื่อ α_k เป็นค่าคงที่ เรียกวิธีการประมาณเชิงเส้นอันดับ P โดยมีเงื่อนไขว่า ค่า α_k ที่ใช้ในการประมาณที่จะต้องทำให้ ผลรวมของกำลังสองของความคลาดเคลื่อน $\{S(n)-S'(n)\}^2$ มีค่าน้อยที่สุด ซึ่งใช้ในการประมาณเชิงเส้น (Autocorrelation Method) หรือวิธีอัตตสัมพันธ์

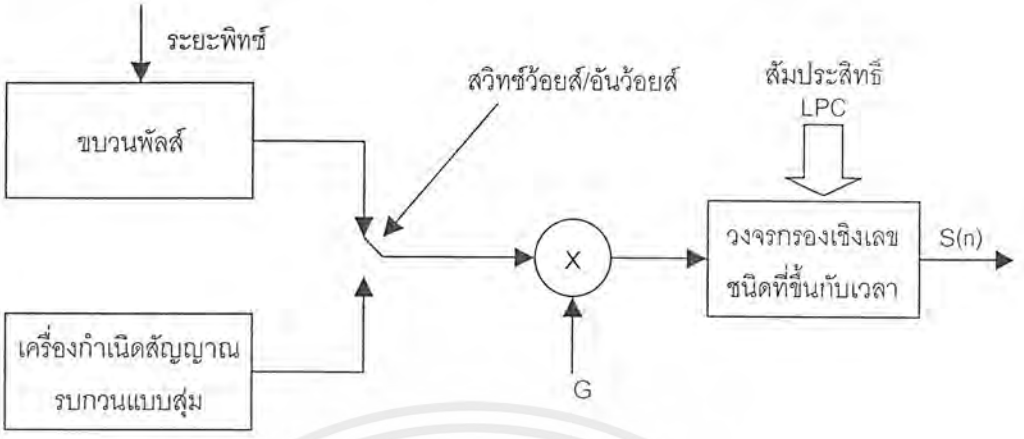
การคำนวณออโตคอร์รีเลชั่น เป็นวิธีการหาสัมประสิทธิ์ LPC โดยฟังก์ชันออโตคอร์รีเลชั่น ซึ่งเป็นการเปรียบเทียบสัญญาณกับสัญญาณของตัวมันเองที่ถูกเปลี่ยนไปตามแกนเวลา ที่ใช้วิธีนี้ เนื่องจากเป็นการคำนวณที่มีการแก้สมการทีละน้อยกว่าวิธีอื่นๆ และมีความแน่นอนในด้านเสถียรภาพ อีกทั้งมีการเก็บข้อมูลที่น้อยกว่า

คุณสมบัติของฟังก์ชันออโตคอร์รีเลชั่น

1. เป็นฟังก์ชันคู่ R(k) = R(-k)
2. จะเป็นค่าสูงสุดเมื่อเปรียบเทียบสัญญาณของตัวมันเองที่ตำแหน่งทางแกนเวลาเดียวกัน คือ R(0)=max

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากหลักการพื้นฐานของการประมาณเชิงเส้นและแบบจำลองระบบสร้างสัญญาณเสียงสามารถเขียนบล๊อกไดอะแกรมการทำประมาณเชิงเส้นมาสร้างสัญญาณเสียงพูดได้ดังรูปที่ 2.7



รูปที่ 2.7 บล๊อคไดอะแกรมแสดงโมเดลการสร้างสัญญาณเสียงพูดอย่างง่าย

จากรูปที่ 2.7 สามารถเขียนสมการได้เป็น

$$S(n) = G * u(n) + \sum_{k=1}^{p} \alpha_k S(n-k) \tag{2.7}$$

การประมาณเชิงเส้นโดยใช้สัมประสิทธิ์ $\{\alpha_k\}$ คือ

$$S'(n) = \sum_{k=1}^{p} \alpha_k S(n-k) \tag{2.8}$$

ดังนั้น ความคลาดเคลื่อน คือ

$$e(n) = S(n) - S'(n)$$

$$e(n) = S(n) - \sum_{k=1}^{p} \alpha_k S(n-k) \tag{2.9}$$

ฟังก์ชันถ่ายโอนระหว่าง e(n) และ S(n) คือ

$$A(z) = E(z)/S(z)$$

$$= 1 - \sum_{k=1}^{p} \alpha_k z^{-k} \tag{2.10}$$

จากสมการที่ (2.7) และ (2.9) จะเห็นได้ว่าถ้า $\{\alpha_k\} = \{a_k\}$ แล้ว

$$e(n) = G * u(n) \tag{2.11}$$

ดังนั้น ค่าผลรวมของกำลังสองของความคลาดเคลื่อน

$$E_n = \sum_m e_n^2(m)$$

$$E_n = \sum_m [S(m) - S'(m)]^2 \tag{2.12}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ n คือ ช่วงที่ n ของสัญญาณที่ใช้คำนวณ เพราะฉะนั้นเพื่อให้ได้ค่า E_n ต่ำที่สุดจะต้องมีเงื่อนไขว่า

$$\frac{\partial E_n}{\partial \alpha_i}=0 \qquad \text{เมื่อ} \quad i = 1, 2, 3, \ldots, p$$

จากสมการ (2.12)

$$\frac{\partial E_n}{\partial \alpha_i}=-2s(m-i)\sum_m\left[s_n(m)-\sum_{k=1}^{p}\alpha_k s_n(m-k)\right] \qquad \text{เมื่อ} \quad i = 1, 2, 3, \ldots, p$$

$$=-2\left[\sum_m s(m)s(m-i)-\sum_{k=1}^{p}\sum_m \alpha_k s(m-k)s(m-i)\right]$$

$$\frac{\partial E_n}{\partial \alpha_i}=0 \qquad \text{ก็ต่อเมื่อ}$$

$$\sum_{k=1}^{p}\alpha_k\sum_m s_n(m-k)s_n(m-i)=\sum_m s_n(m)s_n(m-i) \qquad (2.13)$$

เมื่อ $\quad i = 1, 2, 3, \ldots, p$

ถ้าเรากำหนดให้ $\phi_n(i,j)=\sum s_n(m-k)s_n(m-i)$ เพราะฉะนั้น

$$\sum_{k=1}^{p}\alpha_k\phi_n(i,j)=\phi_n(i,0) \qquad (2.14)$$

โดยสมการ (2.12)-(2.13) จะได้ว่า

$$E_n=\sum_m s_n^2(m)-\sum_{k=1}^{p}\alpha_k\sum_m s_n(m)s_n(m-k)$$

และจาก $\quad \phi_n(i,j)=\sum s_n(m-k)s_n(m-i)$

$$E_n=\phi_n(0,0)-\sum_{k=1}^{p}\alpha_k\phi_n(0,k) \qquad (2.15)$$

สมมติว่าใน 1 เฟรม ของสัญญาณ ที่ตัดมามีจำนวน N ตัวอย่าง คือ $s_n(0)$, $s_n(1)$, $s_n(2)$, ..., $s_n(N-1)$ ในที่นี้เราให้ $s_n(m)=0$ เมื่อ m < 0 หรือ m > N-1 เพราะฉะนั้น

$$\phi_n(i,j)=\sum_m s_n(m-k)s_n(m-i)$$

$$=\sum_{m=0}^{N-1-(i-k)} s_n(m)s_n(m+i-k) \qquad 0\le k\le p, 1\le i\le p$$

ให้ $\quad R_n(k)=\sum_m^{N-1-k} s_n(m)s_n(m+k) \qquad$ เมื่อ $\quad k = 0, 1, 2, \ldots, p \qquad (2.16)$

ดังนั้น จากสมการ (2.5) และ (2.16) จะได้ว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$R_n(k) = \sum_{m=0}^{N-1-k} x'(m)x'(m+k) \qquad (2.17)$$

จากสมการที่ (2.14) จะได้ว่า

$$\sum_{k=1}^{p} \alpha_k R_n(|i-k|) = R_n(i) \qquad \text{เมื่อ} \qquad i = 1, 2, 3, \ldots, p$$

สามารถเขียนให้อยู่ในรูปเมตริกได้เป็น

$$\begin{bmatrix} R_n(0) & R_n(1) & \cdots & R_n(p-1) \\ R_n(1) & R_n(0) & \cdots & R_n(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ R_n(p-1) & R_n(p-2) & \cdots & R_n(0) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} R_n(1) \\ R_n(2) \\ \vdots \\ R_n(p) \end{bmatrix}$$

หรือ

$$R_n \cdot \alpha = r_n \qquad (2.18)$$

เมื่อ

$$R_n = \begin{bmatrix} R_n(0) & R_n(1) & \cdots & R_n(p-1) \\ R_n(1) & R_n(0) & \cdots & R_n(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ R_n(p-1) & R_n(p-2) & \cdots & R_n(0) \end{bmatrix}, \quad \alpha = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_p \end{bmatrix}, \quad r_n = \begin{bmatrix} R_n(1) \\ R_n(2) \\ \vdots \\ R_n(p) \end{bmatrix}$$

การหาพารามิเตอร์ LPC

พารามิเตอร์ LPC ได้แก่ สัมประสิทธิ์ α และอัตราขยาย G

เมื่อได้ค่า $R_n(0), R_n(1), R_n(2), \ldots, R_n(p)$ จากสมการ (2.18) แล้วก็สามารถหาค่า α นั่นคือ

$$\alpha = R_n^{-1} \cdot r_n$$

และจากสมการ (2.7) และ (2.9) จะได้ว่า

$$e(n) = G \cdot u(n)$$

$$\therefore \qquad E_n = \sum_{m=0}^{N-1} e^2(m) = G \sum_{m=0}^{N-1} u^2(n) \qquad (2.19)$$

จากสมการ (2.15) จะได้ว่า

$$E_n = \phi_n(0,0) - \sum_{k=1}^{p} \alpha_k \phi_n(0,k)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$= R_n(0) - \sum_{k=1}^{p} \alpha_k R_n(k) \qquad (2.20)$$

และจากสมการ (2.19) เราสามารถหาค่า G โดยตรงจาก

$$G^2 = \frac{R_n(0) - \sum_{k=1}^{p} \alpha_k R_n(k)}{\sum_{m=0}^{N-1} u^2(m)} \qquad (2.21)$$

ขั้นตอนนี้จะได้วิเคราะห์สัญญาณไปหนึ่งเฟรมแล้ว โดยในหนึ่งเฟรมนี้จะได้ค่าสัมประสิทธิ์ทั้งหมด 12 ค่า (P=12) ; $\alpha_1, \alpha_2, \alpha_3, \ldots, \alpha_p$

การเปลี่ยนพารามิเตอร์ LPC เป็นสัมประสิทธิ์เซปสตรัม

หลังจากการหาสัมประสิทธิ์ LPC และ Gain ใน 1 เฟรมแล้ว จะเปลี่ยนให้เป็น สัมประสิทธิ์เซปสตรัม เนื่องจากการรู้จำเสียงพูดนั้น สัมประสิทธิ์เซปสตรัมเป็นพารามิเตอร์ที่มีลักษณะน่าเชื่อถือได้ดีกว่าสัมประสิทธิ์ LPC ทั้งยังมีความสัมพันธ์ใกล้ชิดกับการรับรู้เสียง ตามความรู้สึกของมนุษย์โดยแท้จริง สัมประสิทธิ์เซปสตรัมสามารถหาได้โดยตรงจากสัมประสิทธิ์ LPC ดังนี้

$C_0 = \ln G$ เป็นสัมประสิทธิ์ตัวแรกซึ่งเป็นเกน

$Q \approx \frac{3}{2} p$ โดย p = 10 ดังนั้น Q = 12 ไม่รวมกับเกน (C_0) จะได้ว่า สัมประสิทธิ์เซปสตรัมใน 1 เฟรม = 12 ตัว

$$C_m = a_m + \sum_{k=1}^{m-1} \left(\frac{k}{m} \right) C_k a_{m-k} \quad ; \quad 1 \le m \le p$$

$$C_m = \sum_{k=1}^{m-1} \left(\frac{k}{m} \right) C_k a_{m-k} \quad ; \quad m > p$$

การเวทค่าพารามิเตอร์ (Parameter Weighting)

เนื่องจากสัมประสิทธิ์เซปสตรัมที่ได้นั้น ช่วงลำดับต้น ๆ และลำดับท้าย ๆ ของเฟรมที่นำมาวิเคราะห์จะเกิด ความคลาดเคลื่อนมากกว่าบริเวณส่วนอื่น เพราะฉะนั้นจึงทำการถ่วงน้ำหนักเพื่อลดค่าความคลาดเคลื่อนดังกล่าวนี้ ด้วยฟังก์ชันเวทติ้ง ดังนี้คือ

$$w_m = \left[1 + \frac{Q}{2} \sin\left(\frac{\pi m}{Q} \right) \right] \quad ; \quad 1 \le m \le Q \qquad (2.22)$$

จะได้พารามิเตอร์สุดท้ายคือ

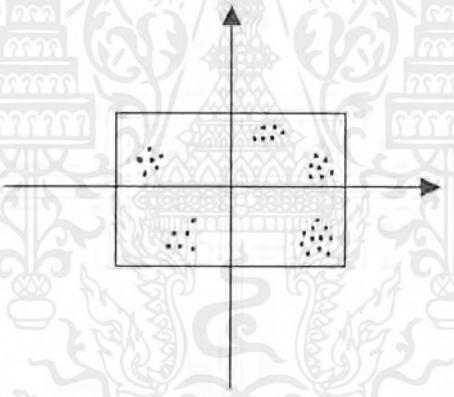
$$C_m' = C_m \cdot w_m \qquad (2.23)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

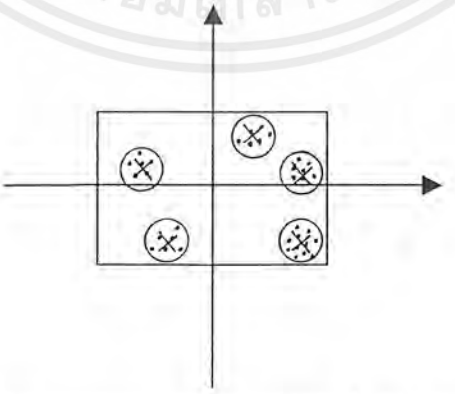
จากนั้นก็จะพิจารณาให้ครบทุกเฟรมของข้อมูล เมื่อพิจารณาเรียบร้อยแล้วก็จะนำไปจัดกลุ่มเสียงและสร้างแบบจำลองเสียงเพื่อใช้ในการเปรียบเทียบต่อไป

2.2.2 การจัดระดับเวคเตอร์ (Vector Quantization)

เวคเตอร์ควอนไตซ์เซชั่น เป็นวิธีการลดไดเมนชั่น (Dimension) หรือจำนวนข้อมูล เวคเตอร์อินพุท หรือ เซตเทรนนิ่ง หรือ พารามิเตอร์ที่ได้จากขั้น LPC จะถูกเลือกมากลุ่มหนึ่งซึ่งใช้เป็นตัวแทนของข้อมูลจำนวนหนึ่งหรือเรียกว่าการหา Codebook อินพุทที่เข้ามาจะถูกทำการเปรียบเทียบกับ codebook ที่มีอยู่ โดยจะพิจารณาว่าอินพุทที่เข้ามานั้นห่างจาก codebook ใดน้อยที่สุด อินพุตดังกล่าวจะถูกแทนด้วยเวคเตอร์โค้ด (index) นั้น อินพุททุกตัวเป็นสมาชิกของเวคเตอร์โค้ดใด ๆ จะถูกนำมาหาจุดศูนย์กลางร่วมใหม่ และนำจุดศูนย์กลางที่ได้นี้ไปทำการหาความคลาดเคลื่อนกับสมาชิกทุกตัว ถ้าค่าความคลาดเคลื่อนที่ได้มีค่ามากกว่าค่าที่กำหนดไว้ค่าหนึ่งหรือค่าที่ยอมรับได้ ก็จะนำศูนย์กลางใหม่นั้นไปเป็น codebook แทน และจะทำการจัดกลุ่มอินพุทเข้ากับ codebook ใหม่ที่ได้และหาค่าความคลาดเคลื่อนอีกครั้งทำอย่างนี้ซ้ำ ๆ จนกระทั่งค่าความคลาดเคลื่อนมีค่าน้อยถึงค่าที่ยอมรับได้ ก็จะถือได้ว่า codebook ที่ดีที่สุดจะเป็นตัวแทนของอินพุททั้งหมด จะสังเกตได้ว่าทุกครั้งที่มีการหา codebook ใหม่นั้น ค่าความคลาดเคลื่อนที่ได้จะมีค่าลดลงทุกครั้งด้วย



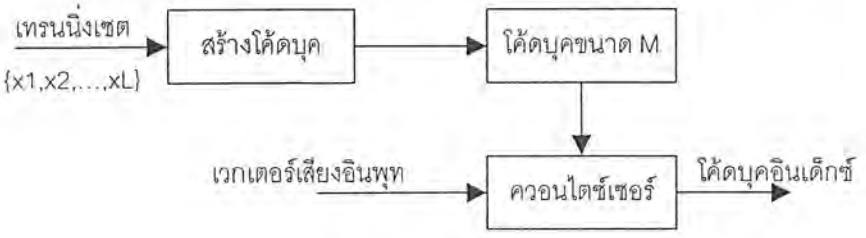
รูปที่ 2.8 แสดงการกระจายเฟรมของเสียงพูดแต่ละจุดแทนเฟรมของเสียง



รูปที่ 2.9 การรวมกลุ่มของเฟรมเสียงเพื่อสร้างโค้ด X

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างเวกเตอร์ควอนไตซ์เซชั่น สมมติให้เวกเตอร์แต่ละตัวมี 2 มิติ และทำการหา Codebook ขนาด 8 เวกเตอร์ ทั้งหมดจะถูกจัดเข้ากลุ่มของ Codebook ต่าง ๆ แล้วทำการหาจุดศูนย์ใหม่โดยการเฉลี่ยค่าเวกเตอร์สมาชิกทุกตัวที่อยู่ในกลุ่มเดียวกัน ผลที่ได้คือ Codebook 8 ตัว เป็นตัวแทนของเวกเตอร์ทั้งหมด



รูปที่ 2.10 บล๊อกไดอะแกรมของเวกเตอร์ควอนไตซ์

การทำงานของการควอนไตซ์แบบเวกเตอร์ แบ่งเป็น 2 ขั้นตอนดังนี้

1. การสร้างโค้ดบุค (codebook) โดยวิธี K-means

จากขั้นตอนการประมาณเชิงเส้นของเสียงตัวอย่างจำนวนมากจะได้เทรนนิ่งเซตซึ่งประกอบด้วยเวกเตอร์สเปคตรัมจำนวน L เฟรม ; $x=\{x_i;1\le i\le L\}$ เฟรมละ P มิติ ; $x=\lfloor x_1,x_2,\ldots,x_p \rfloor$ แล้วนำข้อมูลที่ได้มาทำการสร้างกลุ่มของแบบอ้างอิง

ในระบบการรับรู้เสียงพูดแบบต่าง ๆ จะใช้อ้างอิงของคำหนึ่ง ๆ จากผู้พูดจำนวนมาก เพื่อที่จะได้ครอบคลุมถึงความแปรปรวนต่าง ๆ ที่เกิดระหว่างผู้พูดแต่ละคน เนื่องจากถ้าใช้แบบอ้างอิงจำนวนมากเวลาที่ใช้ในการตอบสนองจะมาก เนื้อที่หน่วยความจำสำรองที่ใช้เก็บแบบอ้างอิงจะเพิ่ม และเมื่อเพิ่มแบบอ้างอิงไประดับหนึ่ง ความถูกต้องในการรับรู้จะเริ่มคงที่ ดังนั้นการจัดกลุ่มแบบอ้างอิงใหม่ เพื่อให้ได้แบบอ้างอิงที่เหมาะสม และสามารถใช้เป็นตัวแทนของแบบอ้างอิงที่มีอยู่ทั้งหมดได้ อัลกอริทึมที่ใช้ได้แก่ K-means Algorithm ขั้นตอนการสร้างโค้ดบุคมีดังนี้

1) นำเทรนนิ่งเซตมาใช้ในการสร้างโค้ดบุค

ขนาดโค้ดบุคของการควอนไตซ์ แบบเวกเตอร์ คือ $M=2^B$ เวกเตอร์ และ เพื่อที่จะหาเซตของ M โค้ดบุคที่ดีที่สุด จำนวนเวกเตอร์อินพุทจะต้องมากกว่าขนาดโค้ดบุคมากๆ

2) การสุ่มค่าเริ่มต้น

การสุ่มค่าเริ่มต้น เป็นวิธีหนึ่งในการออกแบบโค้ดบุค ซึ่งคือ การเลือกค่าเริ่มต้นของโค้ดบุคเรียกโค้ดบุค ที่ได้จากการสุ่มค่าเริ่มต้นนี้ว่า แรนดอมโค้ดบุค (random codebook) ถึงแม้วิธีนี้จะไม่ใช่วิธีที่ดีนัก แต่โค้ดบุคที่ได้จากการสุ่มก็ได้ผลเป็นที่ยอมรับ

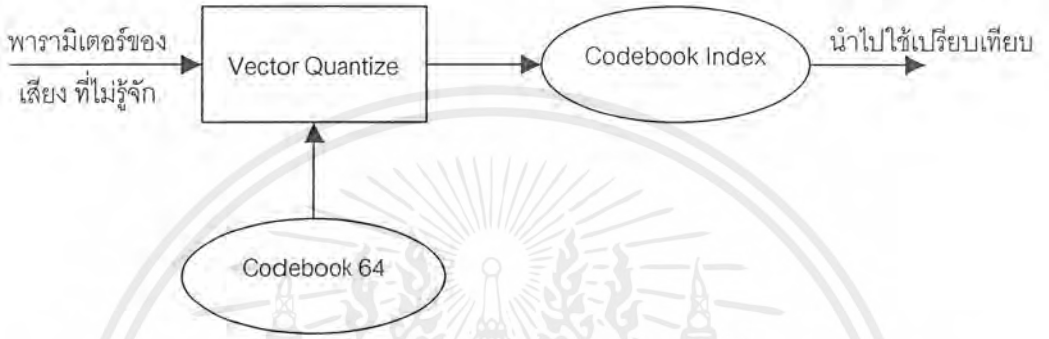
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) การหาความคลาดเคลื่อน

การหาความคลาดเคลื่อน เป็นส่วนที่จำเป็นและเป็นประโยชน์ต่อการออกแบบโค้ดบุค สมการทางพีชคณิตที่ใช้ในการหาระยะทาง มีหลายวิธี แต่ที่นำมาใช้คือ การหาความคลาดเคลื่อนกำลังสองรวม (Totel square error) ซึ่งมีวิธีที่ง่ายและรวดเร็ว

ถ้าสัญญาณมี P มิติ สามารถหาระยะห่างระหว่างสัญญาณอินพุท (x) กับเวกเตอร์โค้ด(y) โดยสมการ

$$d(v_1, v_2) = \| v_1 - v_2 \|^2 = \sum_{i=0}^{k-1} (x_i - y_i)^2 \qquad (2.24)$$



รูปที่ 2.11 ขั้นตอนของเวกเตอร์ควอนไตเซชั่น

4) การจัดกลุ่ม (classification) และการหาจุดศูนย์กลางของกลุ่ม (center cluster)

การจัดกลุ่มเป็นการแบ่งเวกเตอร์อินพุทเข้าไปตามกลุ่มต่าง ๆ ของแรนดอมโค้ดบุค โดยพิจารณาระยะทางหรือความคลาดเคลื่อนน้อยที่สุด ของแต่ละเวกเตอร์อินพุท x กับเวกเตอร์โค้ดบุค y ซึ่งเป็นโค้ดบุคจากนั้นจะทำการหาค่าเฉลี่ยของแต่ละกลุ่ม เพื่อเป็นค่ากลางของกลุ่มนั้น ๆ จะได้

$$\bar{Y} = \frac{1}{L} \sum_{i=1}^{L} x_i$$

$\bar{Y}$ เป็นจุดศูนย์กลางซึ่งเป็นเวกเตอร์ที่อยู่ตรงกลางของ $\{x_i\}_{i=1}^{L}$ ซึ่งแต่ละมิติจะไม่ขึ้นแก่กันหมายความว่าแต่ละ y_k เป็นค่ากลางของ $\{x_i\}_{i=1}^{L}$ ทำ 2 ขั้นตอนนี้ซ้ำ จะเกิดการลู่เข้า (convergent) โดยความคลาดเคลื่อนรวมจะต่ำกว่าค่าหนึ่ง ๆ ซึ่งค่าความคลาดเคลื่อนรวมจะลดลงทุกครั้งที่มีการคำนวณซ้ำใหม่ จึงขึ้นกับค่าที่กำหนดว่าจะต้องการให้ความคลาดเคลื่อนรวมน้อยกว่า ค่ากลางดังกล่าวของแต่ละกลุ่มจะถูกเก็บเป็นเวกเตอร์โค้ดจะได้ว่า y เป็นควอนไตซ์ของค่า x

โดย q(.) เป็นโอเปอร์เรเตอร์ของควอนไตซ์ y ถูกเรียกว่าเอาท์พุทเวกเตอร์ของค่า x โดย y เป็นค่าใดค่าหนึ่งใน $Y = \{y_i; 1 \le i \le M\}$ โดย $y_i = \lfloor y_{i1} \quad y_{i2} \quad \cdots \quad y_{ip} \rfloor$ Y เป็นเซ็ตของโค้ดบุค M เป็นขนาดของโค้ดบุค และ $\{y_i\}$ เป็นเซ็ตของเวกเตอร์โค้ด y_i อาจเรียกว่าโค้ดอ้างอิง และ M อาจเรียกว่าจำนวนระดับขั้น จะทำการแบ่งเวกเตอร์ X ไปใน M เซล $\{C_i; 1 \le i \le M\}$ เมื่อ X อยู่ในเซล C_i

$$q(x) = y_i \quad \text{ถ้า} \quad x \in C_i$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ขั้นตอนการเปรียบเทียบ

เวกเตอร์ควอนไตเซชั่น ที่ใช้เพื่อการออกแบบรับรู้เสียงพูดนั้น มีจำนวนควอนไตเซอร์ M ตัวซึ่งหมายถึงมี M ระดับเสียงเพื่อการรับรู้ แต่ละระดับเสียงพิจารณามาจากเซตของข้อมูลเทรนนิ่ง ซึ่ง M เป็นดัชนีระดับ แต่ละเซตของเทรนนิ่งในแต่ละระดับจะเก็บเสียงที่อยู่ในระดับเดียวกัน

เมื่อมีเสียงที่เราไม่ทราบ (unknown) ; x_t เข้ามา จะเป็นอินพุทเข้าไปยังทุกๆ ควอนไตเซอร์ ค่าดัชนีระดับ (index) ที่ถูกเลือก จะเป็นระดับที่มีความคลาดเคลื่อนเฉลี่ยน้อยที่สุด D(c) เมื่อ i = 1, 2, …, M ซึ่งความคลาดเคลื่อนเฉลี่ยนั้นหาได้จากการวัดระยะทางโดยใช้วิธีการหาความคลาดเคลื่อนกำลังสองรวม

2.2.3 การสร้างแบบจำลอง (Pattern Matching)

หมายถึง การกำหนดรูปแบบขึ้นนำมาเปรียบเทียบกับสิ่งที่เราไม่ทราบ โดยการจัดกลุ่มของสัญญาณที่ไม่รู้จัก (unknown signal) ซึ่งเป็นสัญญาณที่ไม่คงที่ให้อยู่ในกลุ่มใดกลุ่มหนึ่ง วิธีที่เลือกใช้ในการสร้างและหารูปแบบของคำพูดที่เหมือนอยู่สองแนวทาง

-Dynamic Time Warping (DTW)

-Hidden Markov Models (HMM)

วิธีที่เลือกใช้ในการสร้างแบบจำลองคือ Hidden Markov Models (HMM) ซึ่งเป็นวิธีการสร้างแบบจำลองที่นำมาประยุกต์ใช้ในการจำเสียงพูด แบบจำลองมาร์คอฟ แบ่งออกเป็น 2 ประเภทคือ แบบต่อเนื่อง (Continuous) และแบบไม่ต่อเนื่อง (Discrete-time) ในที่นี้จะเลือกแบบไม่ต่อเนื่องเพราะเป็นวิธีการที่ซับซ้อนน้อยกว่าและใช้กับคำพูดสั้น ๆ

Hidden Markov Models (HMM)

HMM เป็นวิธีการที่ใช้ทฤษฎีความน่าจะเป็นมาอธิบายการเกิดเหตุการณ์ของลำดับการณ์ 2 ลำดับ คือ สเตท (state) และปรากฏ (observation) โดยเหตุการณ์ที่สังเกตเห็นจะเห็นได้เพียงเอาท์พุทของแต่ละสเตทในลักษณะที่เป็นค่าปรากฏเท่านั้น จะไม่ทราบแน่ชัดว่าอยู่ที่สเตทใด

ส่วนประกอบของแบบจำลอง HMM

1. N คือจำนวนสเตทในแบบจำลอง โดยสามารถย้ายจากสเตทหนึ่งไปยังอีกสเตทหนึ่งได้ เราให้เซตของสเตทเป็น {1,2,….,N} และสเตทที่เวลา t ใด ๆ เป็น q_t

2. M คือจำนวนของค่าต่อหนึ่งสเตท แทนด้วยสัญลักษณ์ $V = \{v_1, v_2, ..., v_M\}$

3. A= $\{a_{ij}\}$ คือความน่าจะเป็นในการเป็นสเตทที่ $a_{ij} = P\{q_t = j \mid q_{t-1} i\}$ เมื่อ $1 \le i, j \le N$

4. $B = \{b_i(k)\}$ คือความน่าจะเป็นของการเกิดค่าปรากฏที่สเตท $b_i(k) = P\{O_t = v_k \mid q_t = i\}$ เมื่อ i = 1, 2, 3, …, N

5. $\pi = \{\pi_i\}$ คือค่าความน่าจะเป็นที่แต่ละสเตทจะเป็นสเตทเริ่มต้นเมื่อ $\pi_i = P(q_1 = i)$

โครงสร้างแบบจำลอง HMM

แบ่งตามลักษณะการเปลี่ยนสเตท (transition) ของเมตริกซ์ A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$A=\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}$$

1. แบบ Egordic Model หรือ Fully Connected Model แบบจำลองนี้ทุกสเตทสามารถเปลี่ยนสเตทไปยังสเตทอื่น ๆ ได้ทุก ๆ สเตท ดังรูปที่ 2.10 (a) เป็นตัวอย่างของแบบจำลองที่มี N=4

2. แบบ Left-Right Model หรือ Bakis Model แบบจำลองนี้การเปลี่ยนสเตทจะเปลี่ยนจากซ้ายไปขวามีคุณสมบัติการเปลี่ยนดังนี้

2.1 $a_{ij}=0; j<i$ หมายความว่าเมื่อผ่านสเตทใดไปแล้วจะไม่มีการย้อนกลับมายังสเตทนั้น

2.2 $\pi_i=1; i=1$ หมายความว่า ลำดับของสเตทต้องเริ่มต้นที่สเตทที่ 1 สเตทที่เหลือจึงมีความน่าจะเป็นที่จะเป็นสเตทเริ่มต้นเท่ากับศูนย์

Lift-Right Model นี้มีกฎข้อบังคับการเปลี่ยนแปลงสเตทดังนี้

$a_{ij}=0$ เมื่อ $j>i+\Delta i$ โดยค่าของ $\Delta i=2$ หมายความว่าการเปลี่ยนสเตทจะสามารถกระโดดข้ามเปลี่ยนไปยังสเตทข้างหน้าได้ไม่เกิน 2 สเตท

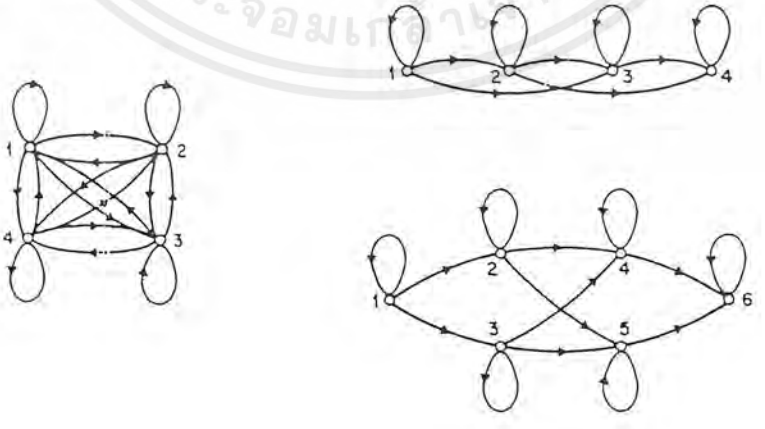
จะได้เมตริกซ์การเปลี่ยนสเตทเป็น

$$A=\begin{bmatrix} a_{11} & a_{12} & a_{13} & 0 \\ 0 & a_{22} & a_{23} & a_{24} \\ 0 & 0 & a_{33} & a_{34} \\ 0 & 0 & 0 & a_{44} \end{bmatrix}$$

จะเห็นว่าสเตทสุดท้ายมีสัมประสิทธิการเปลี่ยนสเตทเป็น $a_{NN}=1, a_{Ni}=0$ เมื่อ $i<N$

แบบจำลองนี้เหมาะกับสัญญาณที่มีการเปลี่ยนแปลงอย่างต่อเนื่องเช่น คำพูด

3. แบบ Parallel Lift-Righ Model มีคุณสมบัติการเปลี่ยนสเตทคล้ายแบบที่ 2 แต่มีความยืดหยุ่นมากกว่า ดังรูปที่ 2.13 (c)



รูปที่ 2.12 แสดงแบบจำลองต่าง ๆ ของ HMM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปัญหาของ HMM

ปัญหาของ HMM มี 3 ข้อ ซึ่งต้องใช้อัลกอริทึมวิธีต่าง ๆ ในการคำนวณเพื่อแก้ปัญหานี้

ปัญหาที่ 1 เมื่อลำดับของค่าปรากฏ $O=\{o_1,o_2,...,o_T\}$ และมีโมเดล $\lambda=(A,B,\pi)$ เราจะคำนวณหาค่า $P(O|\lambda)$ ของลำดับของค่าปรากฏได้อย่างไร

ปัญหาที่ 2 เมื่อลำดับของค่าปรากฏ $O=\{o_1,o_2,...,o_T\}$ และแบบจำลอง $\lambda=(A,B,\pi)$ เราจะหาลำดับสเตท $q=\{q_1q_2\ldots q_T\}$ ที่เหมาะสมในการให้ค่าปรากฏนั้นได้อย่างไร

ปัญหาที่ 3 จะหาแบบจำลอง $\lambda=(A,B,\pi)$ ที่ให้ค่า $P(O|\lambda)$ มากที่สุดได้อย่างไร

ลำดับของค่าปรากฏที่ใช้ปรับค่าพารามิเตอร์ A, B และ π เพื่อให้ได้แบบจำลองที่ดีที่สุดนั้นเรียกว่าลำดับเทรนนิ่ง (training sequence)

การคำนวณเพื่อแก้ปัญหาของ HMM

1. การแก้ปัญหาข้อที่ 1 เป็นการคำนวณว่าแบบจำลอง λ ให้ได้ความน่าจะเป็นที่จะได้ลำดับค่าปรากฏมากน้อยเพียงใด มีวิธีการเพื่อช่วยแก้ปัญหาโดยกระบวนการต่อไปนี้

1.1 กระบวนการไปข้างหน้า (Forward Procedure)

เมื่อกำหนดให้ตัวแปรไปข้างหน้า (Forward variable)

$$\alpha_t(i)=P(o_1,o_2,\ldots,o_T,q_t=i|\lambda)$$

หมายถึงความน่าจะเป็นของการเกิดลำดับค่าปรากฏ $o_1,o_2,\ldots o_T$ ที่อยู่ในสเตท i ณ เวลา t โดยมีแบบจำลองเป็น λ โดยสามารถหา $\alpha_t(i)$ ได้ดังนี้

1.1.1 การเริ่มต้น (initialization) เมื่อกำหนด $\alpha_t(i)=\pi_i b_i(o_t)$ ที่เวลาเริ่มต้น t=1 และเหตุการณ์ เริ่มต้น o_1 เมื่อ $1\le i\le N$

1.1.2 การเหนี่ยวนำ (induction)

$$\alpha_{t+1}(j)=\left[\sum_{i=1}^{N}\alpha_t(i)a_{ij}\right]b_j(o_{t+1}) \qquad (2.25)$$

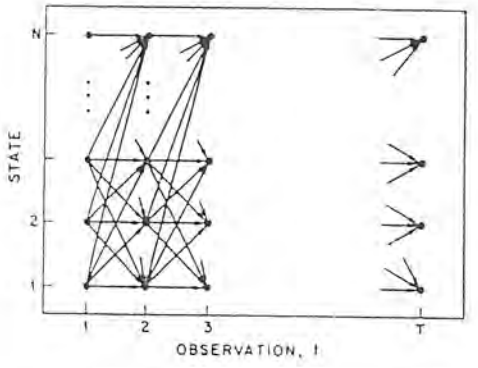
เมื่อ $1\le t\le T-1$ หมายถึง ความน่าจะเป็นสเตท j ที่เวลา t+1 ได้มาจากสเตท i ที่เป็นไปได้ ถึง N สเตทที่เวลา t ดังรูปที่ 2.14

1.1.3 การสิ้นสุด (termination)

$$P(o|\lambda)=\sum_{i=1}^{N}\alpha_t(i) \qquad (2.26)$$

ความน่าจะเป็นของลำดับค่าปรากฏ O ได้จากผลรวม $\alpha_t(i)$ จากทุก ๆ สเตท เมื่อ $1\le j\le N$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 กระบวนการไปข้างหน้า

1.2 กระบวนการย้อนกลับ (Backward Procedure)

เมื่อกำหนดให้ตัวแปรย้อนกลับ(Backward variable) $\beta_T(i)=P(o_{t+1}o_{t+2}\cdots o_T, q_t=i\,|\,\lambda)$ หมายถึงความน่าจะเป็นของลำดับค่าปรากฏส่วนหลัง จากเวลา t+1 ไปจนจบ โดยกำหนดว่าต้องอยู่ที่สเตท i ความน่าจะเป็นของลำดับค่าปรากฏส่วนหลัง จากเวลา ที่เวลา t และมีการจำลองเป็น λ เราจะคำนวณหา $\beta_T(i)$ ได้ดังนี้

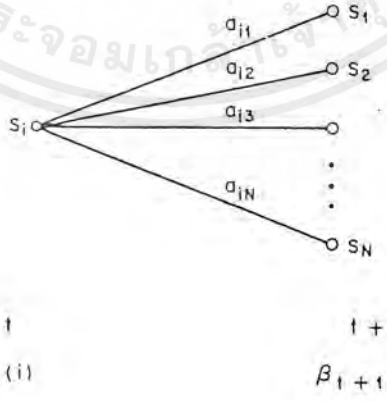
1. การเริ่มต้น (initialization)

$$\beta_T(i)=1 \qquad \text{เมื่อ} \qquad 1\le j\le N$$

2. การเหนี่ยวนำ (induction)

$$\beta_t(i)=\sum_{j=1}^{N} a_{ij}b_j(o_{t+1})\beta_{t+1}(i) \tag{2.27}$$

เมื่อ $t=T-1,T-2,\cdots,1$ และ $1\le i\le N$



รูปที่ 2.14 กระบวนการถอยหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.14 แสดงถึงค่าปรากฏที่จะอยู่ที่สเตท i ที่เวลา t โดยคำนึงถึงลำดับค่าปรากฏจากเวลา t+1 ซึ่งต้องพิจารณาสเตท j ที่จะเป็นไปได้ทั้งหมด ณ เวลา t +1 โดยจะขึ้นอยู่กับค่า a_{ij} และ $b_j(o_{t+1})$

2. การแก้ปัญหาที่ 2 เพื่อหาลำดับสเตทที่เหมาะสม

เราจะใช้วิธี วิทเทอร์บี อัลกอริทึม (Viterbi Algorithm) เพื่อหาลำดับสเตทที่ดีที่สุด ณ เวลา t หนึ่ง ๆ เมื่อกำหนดลำดับเหตุการณ์ $O=(o_1 o_2 \ldots o_T)$ โดยนิยามให้

$$\delta_t(i)= \max_{q_1,q_2,\ldots,q_T} P[q_1,q_2,\ldots,q_{t-1},q_t=i,o_1 o_2 \ldots o_t \mid \lambda]$$

หมายถึง ความน่าจะเป็นสูงสุดของเส้นทาง (path) ณ เวลา t ซึ่งเริ่มนับจากเหตุการณ์ที่เวลาเริ่มต้น จนถึงเวลา t สเตท i และโดยการอาศัยคุณสมบัติการเหนี่ยวนำ (induction) เราจะได้

$$\delta_{t+1}(i)=[\max_i \delta_t(i)a_{ij}]b_j(o_{t+1})$$

เราสามารถหาลำดับสเตทที่ดีที่สุดได้โดยใช้กระบวนการต่อไปนี้ เมื่อกำหนดให้ $\psi_t(i)$ เป็นอาเรย์ (array)

1. การเริ่มต้น (initialization)

$$\delta_1(i)=\pi_i b_i(o_1) \qquad \text{เมื่อ} \qquad 1\le i\le N$$

$$\psi_1(i)=0$$

2. การย้อนกลับ (recursion)

$$\delta_t(j)= \max_{1\le i\le N} [\delta_{t-1}(i)a_{ij}]b_j(o_t) \qquad \text{เมื่อ} \qquad 2\le t\le T, 1\le j\le N$$

$$\psi_t(j)=\arg \max_{1\le i\le N} [\delta_{t-1}(i)a_{ij}] \qquad \text{เมื่อ} \qquad 2\le t\le T, 1\le j\le N$$

3. การสิ้นสุด (termination)

$$P^* = \max_{1\le i\le N} [\delta_T(i)]$$

$$q_T^* =\arg \max_{1\le i\le N} [\delta_T(i)]$$

4. เส้นทางเดินย้อนกลับ(Path backtracking)

$$q_t^* =\psi_{t+1}(q_{t+1}^*) \qquad \text{เมื่อ} \qquad t = T\text{-}1, T\text{-}2, \ldots, 2, 1$$

3. การแก้ปัญหาที่ 3 เพื่อหาโมเดลที่จะให้ผลตามลำดับค่าปรากฏหนึ่ง ๆ โดยเลือกค่าพารามิเตอร์ A, B, π ที่ดีที่สุด โดยใช้ กระบวนการทำซ้ำ (Iterative) วิธีที่เราเลือกใช้ คือวิธี บาม-เวลช์ (Baum-Welch) หรือ EM (Expectation – Maximization)

เมื่อนิยามให้

1. $\gamma_t(i)=P(q_t=i \mid O,\lambda)$

หมายถึงความน่าจะเป็นที่จะอยู่สเตท i ณ เวลา t โดยกำหนดลำดับเหตุการณ์ O และแบบจำลอง λ ให้สามารถแสดงค่า $\gamma_t(i)$ ได้ดังนี้

$$\gamma_t(i)=\frac{P(O,q_t=i \mid \lambda)}{P(O \mid \lambda)}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

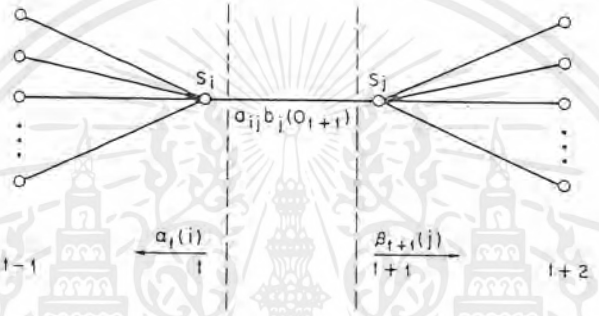
$$= \frac{P(O, q_t = i \mid \lambda)}{\sum_{i=1}^{N} P(O, q_t = i \mid \lambda)}$$

เนื่องจาก $P(O, q_t = i \mid \lambda)$ มีค่าเท่ากับ $\alpha_t(i)\beta_t(i)$ จึงทำให้

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^{N} \alpha_t(i)\beta_t(i)} \tag{2.28}$$

2. $\varepsilon_t(i,j) = P(q_t = i, q_{t+1} j \mid O, \lambda)$

หมายถึง ความน่าจะเป็นที่จะอยู่ที่สเตท i ที่เวลา t และ สเตท j ที่เวลา t+1 เมื่อกำหนดแบบจำลอง และลำดับค่าปรากฏให้



รูปที่ 2.15 แสดงลำดับการคำนวณการเกิดค่าปรากฏร่วมซึ่งจะอยู่สเตท i ที่เวลา t และอยู่ที่สเตท j ที่เวลา t+1

ซึ่งจากนิยามของตัวแปรไปข้างหน้า และตัวแปรย้อนกลับ สามารถนำมาสัมพันธ์กับ $\varepsilon_t(i,j)$ ได้ดังนี้

$$\varepsilon_t(i,j) = \frac{P(q_t = i, q_{t+1} = j, O \mid \lambda)}{P(O \mid \lambda)}$$

$$= \frac{\alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)} \tag{2.29}$$

และจะได้ความสัมพันธ์ของ $\gamma_t(i)$ กับ $\varepsilon_t(i,j)$ ดังนี้

$$\gamma_t(i) = \sum_{j=1}^{N} \varepsilon_t(i,j) \tag{2.30}$$

และ

$\sum_{t=1}^{T-1} \gamma_t(i)$ = จำนวนของการเปลี่ยนสเตทออกจากสเตท i ในลำดับค่าปรากฏ O

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$\sum_{t=1}^{T-1}\varepsilon_t(i,j)=$ จำนวนของการเปลี่ยนสเตทจากสเตท i ไป j ในลำดับค่าปรากฏ O

ดังนั้นสามารถหาค่าพารามิเตอร์ได้ดังนี้

$$\pi_i^{'}=\gamma_1(i) \quad \text{เมื่อ} \quad 1\le i\le N \tag{2.31}$$

$$a_{ij}^{'}=\frac{\sum_{t=1}^{T}\varepsilon_{t-1}(i,j)}{\sum_{t=1}^{T}\gamma_{t-1}(i)} \tag{2.32}$$

$$b_j^{'}(k)=\frac{\sum_{t=1,o_t=v_k}^{T}\gamma_t(i)}{\sum_{t=1}^{T}\gamma_t(i)} \tag{2.33}$$

จากกระบวนการข้างต้น ถ้าเราคำนวณซ้ำ ๆ โดยให้ $\lambda'=(A',B',\pi')$ แทน $\lambda=(A,B,\pi)$ ซึ่งเป็นแบบจำลองเริ่มต้นแล้ว จะทำให้ความน่าจะเป็นของการเกิดลำดับค่าปรากฏ O ดีขึ้น จนกระทั่งถึงจุดวิกฤต ซึ่งเราจะได้จุดวิกฤตของฟังก์ชันความน่าจะเป็นในกรณีที่ $\lambda'=\lambda$ หรือถ้า λ' มีความน่าจะเป็นมากกว่าแบบจำลอง λ ในลักษณะที่ $P(O|\lambda')>P(O|\lambda)$ นั่นก็คือ เราก็จะได้ แบบจำลอง λ' ใหม่ ที่น่าจะทำให้เกิดลำดับค่าปรากฏ O ได้ดีกว่า

การปรับค่าพารามิเตอร์ของ Hidden Markov Model

1.การสเกลลิ่ง(Scaling)

เนื่องจาก α_t(i) จะประกอบไปด้วยผลรวมของเทอมจำนวนมาก ซึ่งก็คือ

$$\left(\prod_{s=1}^{t-1}a_{q_sq_{s+1}}\prod_{s=1}^{t}b_{q_t}(O_s)\right)$$

และเนื่องจากแต่ละเทอมของ a และ b มีค่าน้อยกว่า 1 อยู่แล้วเมื่อพิจารณาผลรวมของการคูณค่าจะทำให้ค่าที่ได้มีค่าน้อยลงไปเรื่อยๆ แสดงว่าเมื่อ t มากขึ้น แต่ละเทอมของ α_t(i) จะเข้าสู่ศูนย์ทำให้ Dynamic Range ของการคำนวณ α_t(i) เกิน Range ของคอมพิวเตอร์ ทำให้ค่าที่ได้ไม่ถูกต้อง จึงได้มีการสเกลลิ่งขึ้นเพื่อทำให้ α_t(i) อยู่ภายใน Dynamic Range ของคอมพิวเตอร์ การสเกลลิ่งทำได้โดยการคูณ α_t(i) โดยสัมประสิทธิ์การสเกลลิ่ง ซึ่งสัมประสิทธิ์นี้ไม่ขึ้นอยู่กับ i การสเกลลิ่ง B_t(i) ก็เช่นเดียวกัน หลังการคำนวณค่า การสเกลลิ่งก็จะตัดกันหมดไปเอง พิจารณาจากสมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\bar{a}_{ij}=\frac{\sum_{t=1}^{T-1}\alpha_t(i)\,a_{ij}\,b_j(O_{t+1})\,\beta_{t+1}(j)}{\sum_{t=1}^{T}\sum_{j=1}^{N}\alpha_t(i)\,a_{ij}\,b_j(O_{t+1})\,\beta_{t+1}(j)} \qquad (2.34)$$

เมื่อเราให้ $\alpha_t(i)$ แทน α ที่ยังไม่ได้สเกลลิ่ง

$\alpha_t(i)$ เป็น α ที่สเกลลิ่งแล้ว

$\alpha_t(i)$ แทนเวอร์ชั่นของ α ก่อนการสเกลลิ่ง

เมื่อ t=1 จะได้ $\alpha_1(i) = c_1\alpha_1(i)$

เมื่อ
$$c_1=\frac{1}{\sum_{i=1}^{N}\alpha_t(i)}$$

เมื่อ $2 \le t \le T$ คำนวณ $\alpha_t(i)$ จากสมการ (2.25) ในเทอมของ $\alpha_t(i)$ ค่าก่อน

$$\hat{\alpha}_t(i)=\sum_{j=1}^{N}\hat{\alpha}_{t-1}(j)\,a_{ji}\,b_i(o_t) \qquad (2.35)$$

เมื่อ สัมประสิทธิ์การสเกลลิ่ง เป็น
$$c_1=\frac{1}{\sum_{i=1}^{N}\hat{\hat{\alpha}}_t(i)}$$

เมื่อให้ $\alpha_t(i) = c_t\,\alpha_t(i)$

จากสมการ (2.35) จะเขียนได้ว่า

$$\alpha_t(i)=\frac{\sum_{j=1}^{N}\hat{\alpha}_{t-1}(j)\,a_{ji}\,b_j(o_t)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\hat{\alpha}_{t-1}(j)\,a_{ji}\,b_j(o_t)} \qquad (2.36)$$

และโดยการเหนี่ยวนำจะได้

$$\hat{\alpha}_{t-1}(j)=\left(\prod_{\tau=1}^{t-1}c_\tau\right)\alpha_{t-1}(j)$$

จะได้ว่า

$$\hat{\alpha}_t(i)=\frac{\sum_{j=1}^{N}\alpha_{t-1}(j)\left(\prod_{\tau=1}^{t-1}c_\tau\right)a_{ji}\,b_i(o_t)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_{t-1}(j)\left(\prod_{\tau=1}^{t-1}c_\tau\right)a_{ji}\,b_i(o_t)}=\frac{\alpha_t(i)}{\sum_{i=1}^{N}\alpha_t(i)} \qquad (2.37)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นั่นคือจะสเกล $\alpha_t(i)$ ได้โดยหารด้วยผลรวมของ $\alpha_t(i)$ ทั้งหมดและสเกล $B_t(i)$ ด้วยค่าเดียวกันนี้ในเทอมของการสเกลนี้สมการ (2.34) จะเป็น

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \hat{\alpha}_t(i)\, a_{ij}\, b_j(O_{t+1})\, \hat{\beta}_{t+1}(j)}{\sum_{t=1}^{T-1}\sum_{j=1}^{N} \hat{\alpha}_t(i)\, a_{ij}\, b_j(O_{t+1})\, \hat{\beta}_{t+1}(j)} \tag{2.38}$$

โดยแต่ละ $\alpha_t(i),\beta_{\tau+1}(i)$ จะได้เป็น

$$\hat{\alpha}_t(i) = \left[\prod_{s=1}^{t} c_s\right] \alpha_t(i) = c_t\, \alpha_t(i) \tag{2.39}$$

$$\hat{\beta}_{t+1}(j) = \left[\prod_{s=t+1}^{T} c_s\right] \beta_{t+1}(j) = D_{t+1}\, \beta_{t+1}(j) \tag{2.40}$$

ดังนั้นสมการ (2.38) จะเขียนได้เป็น

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} C_t\, \alpha_t(i)\, a_{ij}\, b_j(o_{t+1})\, D_{t+1}\, \beta_{t+1}(j)}{\sum_{t=1}^{T-1}\sum_{j=1}^{N} C_t\, \alpha_t(i)\, a_{ij}\, b_j(o_{t+1})\, D_{t+1}\, \beta_{t+1}(j)} \tag{2.41}$$

ซึ่งเทอม C_tD_{t+1} จะเขียนในเทอม

$$C_t D_{t+1} = \prod_{s=1}^{t} c_s \prod_{s=t+1}^{T} c_s = \prod_{s=1}^{T} c_s = C_t \tag{2.42}$$

ซึ่งไม่ขึ้นกับเวลา t ดังนั้น C_tD_{t+1} จะตัดทิ้ง ทั้งเศษและส่วนของสมการ(2.41) ซึ่งทำให้สูตรการคำนวณซ้ำๆ (reestimate) เดิมกลับคืนมา กระบวนการสเกลลิ่ง ดังกล่าวนี้สามารถใช้ได้กับสัมประสิทธิ์ β และ π ในการเสเกลลิ่งนี้จะทำให้การคำนวณค่า $P(O|\lambda)$ เปลี่ยนไปเราจะไม่สามารถหาได้จากการรวมกันของเทอม $\alpha_T(i)$ แต่จะหาได้จากคุณสมบัติ

$$\prod_{t=1}^{T} c_t \sum_{i=1}^{N} \alpha_T(i) = C_T \sum_{i=1}^{N} \alpha_T(i) = 1 \tag{2.43}$$

ดังนั้นจะได้

$$\prod_{t=1}^{T} c_t\, P(O|\lambda) = 1$$

$$P(O|\lambda) = \frac{1}{\prod_{t=1}^{T} c_t} \tag{2.44}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำให้อยู่ในรูปของ log ของ P เพื่อไม่ให้เกิน dynamic range ของคอมพิวเตอร์

$$\log\left[P(O|\lambda\right]=\sum_{t=1}^{T}\log c_t \tag{2.45}$$

2.ลำดับของค่าปรากฏหลายเหตุการณ์(Multiple Observation Sequence)

ในการใช้แบบจำลอง Left-Right นั้น การเทรนแบบจำลองต้องใช้หลายๆ เหตุการณ์ของลำดับค่าปรากฏเข้ามาเทรน เพื่อให้ได้ค่าพารามิเตอร์ที่ถูกต้องมากขึ้น

ถ้าให้เซตของ v ลำดับค่าปรากฏเป็น

$$O=[O^{(1)},O^{(2)},...,O^{(K)}]$$

เมื่อ $O^{(k)}=(O_1^{(k)},O_2^{(k)},...,O_{Tk}^{(k)})$ เป็นลำดับค่าปรากฏของเหตุการณ์ที่ v โดยให้แต่ละเหตุการณ์เป็นอิสระต่อกันจะได้

$$P(O|\lambda)=\prod_{k=1}^{K}P(O^{(k)}|\lambda)$$

$$=\prod_{k=1}^{K}P_k$$

นำเอาจำนวนเหตุการณ์ของการเกิดค่าปรากฏแต่ละเหตุการณ์มารวมกันจะได้สูตรหา a_{ij} ,$b_j(k)$ เป็น

$$\bar{a}_{ij}=\frac{\sum_{k=1}^{K}\frac{1}{P_k}\sum_{t=1}^{Tk-1}\alpha_t^k(i)a_{ij}b_j(o_{t+1}^{(k)})\beta_{t+1}^k(j)}{\sum_{k=1}^{K}\frac{1}{P_k}\sum_{t=1}^{Tk-1}\alpha_t^k(i)\beta_t^k(i)} \tag{2.46}$$

$$\bar{b}_j(\ell)=\frac{\sum_{k=1}^{K}\frac{1}{P_k}\sum_{t=1}^{Tk-1}\alpha_t^k(i)\beta_t^k(i)}{\sum_{k=1}^{K}\frac{1}{P_k}\sum_{t=1}^{Tk-1}\alpha_t^k(i)\beta_t^k(i)} \quad s.t.\ O_t=V\ell \tag{2.47}$$

ส่วน π_i ไม่ต้องคำนวณเนื่องจาก $\pi_1=1$, $\pi_i=0, i\neq 1$

จะได้การสเกลลิ่งที่เหมาะสมของสมการ (2.46)-(2.47)

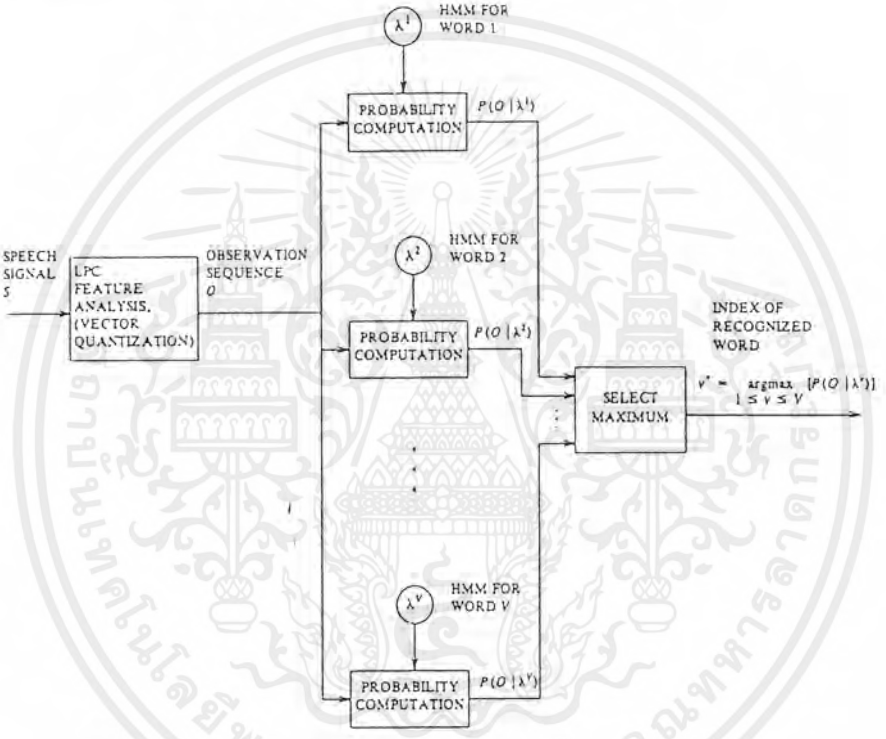
$$\bar{a}_{ij}=\frac{\sum_{k=1}^{K}\frac{1}{P_k}\sum_{t=1}^{Tk-1}\hat{\alpha}_t^k(i)\,a_{ij}\,b_j(o_{t+1}^{(k)})\,\hat{\beta}_{t+1}^k(j)}{\sum_{k=1}^{K}\frac{1}{P_k}\sum_{t=1}^{Tk-1}\hat{\alpha}_t^k(i)\,\hat{\beta}_t^k(i)} \tag{2.48}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\bar{b}_j(\ell) = \frac{\sum_{k=1}^{K} \frac{1}{P_k} \sum_{t=1}^{Tk-1} \alpha_t^k(i)\,\beta_t^k(i)}{s.t.\ O_t = V_\ell} \qquad (2.49)$$
$$\sum_{k=1}^{K} \frac{1}{P_k} \sum_{t=1}^{Tk-1} \alpha_t^k(i)\,\beta_t^k(i)$$

ระบบการจำลองมาร์คอฟ

เมื่อเรามีคำศัพท์อยู่ V คำ ในการทำการรู้จำได้ เราจะต้องสร้างแบบจำลองของคำ แต่ละคำที่แตกต่างกัน คำแต่ละคำจะมีลำดับเทรนนิ่งที่ได้จากคุณลักษณะเฉพาะของคำศัพท์นั้น ๆ การที่เราจะรู้จำคำพูดได้ต้องทำได้ตามบล๊อกไดอะแกรมดังต่อไปนี้



รูปที่ 2.16 บล๊อกไดอะแกรมการรู้จำคำโดดด้วยแบบจำลองของมาร์คอฟ

1. เมื่อมีคำศัพท์อยู่ V คำ เราต้องสร้างแบบจำลองมาร์คอฟ : λ_V ของแต่ละคำนั้น นั่นคือการหาค่า (A,B,π) ที่เหมาะสมกับลำดับเทรนนิ่งของคำนั้น ๆ

2. ในการจะรู้จำคำศัพท์แต่ละคำ เราจะทำการหาค่า $P(O|\lambda)$ ของทุก ๆ แบบจำลอง แล้วเลือกแบบจำลองที่มีค่าความน่าจะเป็นในการเกิดค่าปรากฏสูงสุด คือ

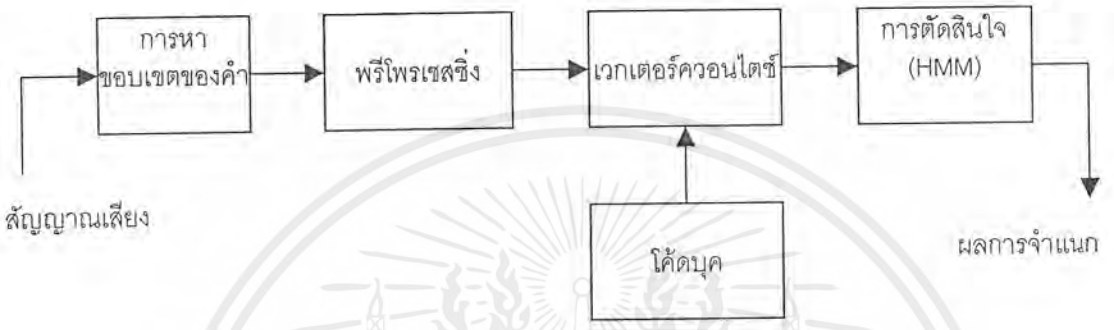
$$V^* = \arg\max P(o|\lambda_V)$$

คำศัพท์ที่สอดคล้องกับแบบจำลองดังกล่าวนี้ จะเป็นคำเดียวกับคำศัพท์ที่เราต้องการจะรู้จำนั่นเองโดยขั้นตอนการคำนวณหาค่าความน่าจะเป็น จะใช้วิธีวิทเทอร์บี ดังที่ได้กล่าวแล้วในตอนต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การคำนวณและการสร้าง

ระบบการจำแนกเสียงพูดโดยทั่วไปสามารถแบ่งออกเป็นภาคต่างๆตามบล็อกไดอะแกรมจากรูปที่ 3.1 ซึ่งประกอบไปด้วยภาคการหาขอบเขตของคำ ภาคพรีโพรเซสซิ่ง ภาคเวกเตอร์ควอนไตซ์ ภาคโค้ดบุค ภาคการตัดสินใจ สำหรับในส่วนของการหาขอบเขตของคำ เป็นขั้นตอนการปรับข้อมูลเสียงจากอินพุทให้เหมาะสม นั่นก็คือ การหาขอบเขตที่เป็นส่วนของคำที่แท้จริง ก่อนที่จะส่งไปภาคอื่นๆต่อไป



รูปที่ 3.1 แสดงบล็อกไดอะแกรมการจำแนกเสียงพูดโดยทั่วไป

สำหรับปริญญานิพนธ์นี้ แต่ละภาคจะมีวิธีการทำดังแสดงในบล็อกไดอะแกรมในรูปที่ 3.1 ซึ่งสามารถอธิบายเป็นส่วน ๆ ได้ดังนี้

3.1 การหาขอบเขตของคำ

ขั้นตอนในภาคนี้จะใช้ หลักการวิเคราะห์ค่าพลังงานของสัญญาณเสียง ในโดเมนเวลา ซึ่งมีวิธีการดังต่อไปนี้

1. คำนวณหาค่าพลังงานของสัญญาณในช่วงเวลาที่ถูกบันทึกทั้งหมด โดยการคำนวณหาค่าพลังงานของสัญญาณซึ่งจะทำการคำนวณเป็นเฟรม กำหนดให้เฟรมหนึ่งมีขนาด 100 แซมเปิ้ล และทำการคำนวณต่อไปเรื่อย ๆ จนครบทุกเฟรม

2. พิจารณาหาค่าพลังงานสูงสุดของสัญญาณ(Maximum Energy)และดูว่าที่ค่าพลังงานสูงสุดนี้อยู่ที่เฟรมใด แล้วกำหนดให้เฟรมนั้นเป็นเฟรมอ้างอิง

3. กำหนดระดับอ้างอิง (Reference) โดยที่ระดับอ้างอิงนี้จะเป็นตัวบอกว่า จุดเริ่มต้นและจุดสุดท้ายของคำจะอยู่นอกช่วงนี้สามารถคำนวณได้จาก(Cutfactor*Emax) ซึ่งในที่นี้จะใช้ค่า Cutfactor=0.2

4. การหาจุดเริ่มต้นของคำ เมื่อกำหนดระดับอ้างอิง เป็นที่เรียบร้อยแล้ว พิจารณาโดยการเปรียบเทียบค่าพลังงานของสัญญาณกับระดับอ้างอิง โดยเฟรมแรกจะเริ่มที่เฟรมที่มีค่าพลังงานสูงสุดก่อน แล้วเลื่อนเฟรมที่จะเปรียบเทียบ พลังงานกับพลังงานอ้างอิงไปด้านหลังเรื่อยๆ จนกว่าจะพบเฟรมที่มีค่าพลังงานน้อยกว่าเฟรมอ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. การหาจุดสุดท้ายของคำ จะทำการพิจารณาค่าพลังงานไปข้างหน้า เช่นเดียวกับการหาจุดเริ่มต้นของคำ จนกว่าจะพบเฟรมที่มีค่าน้อยกว่า เฟรมที่มีค่าพลังงานอ้างอิง ก็จะได้จุดสุดท้ายของสัญญาณ

3.2 การนอร์มอลไลซ์ (Normalization)

เนื่องจากสัญญาณเสียงพูดแต่ละคำมีความยาวไม่เท่ากันจึงต้องมีการนอร์มอลไลซ์สัญญาณเสียงพูดที่ได้จากการตัดคำให้มีความยาวเท่ากันเสียก่อนโดยเราจะใช้หลักการ Interpolation จาก Numerical Mathematics คำนวณจากสมการ

$$g(x)=\frac{b-x}{b-a}f(a)+\frac{x-a}{b-a}f(b)$$

3.3 การหาค่าพารามิเตอร์(pre-processing)

3.3.1 การวิเคราะห์สัญญาณเสียงเบื้องต้น

มีการเลือกใช้ค่าต่าง ๆ ในการคำนวณ และออกแบบโปรแกรมดังนี้

1) การพรีเอมฟาซีส

ใช้วงจรอันดับหนึ่ง ซึ่งมีฟังก์ชันถ่ายโอน คือ

$$H(z)=1-az^{-1}$$

ค่า ที่ใช้คือ 15/16 = 0.9375

2) การแบ่งช่วงสัญญาณ

ขนาดของช่วงสัญญาณมีเงื่อนไขในการเลือก คือ

- ค่า M ต้องสั้นพอที่คุณสมบัติของเสียงไม่เปลี่ยนแปลง
- ค่า N ต้องยาวพอที่จำนวนของตัวอย่างมีเพียงพอสำหรับการหาสัมประสิทธิ์
- การเลื่อนในการวิเคราะห์ (ค่า M) ต้องไม่ข้ามข้อมูล

ดังนั้นค่า M จะต้องน้อยกว่าค่า N แต่ถ้าค่า M มีขนาดเล็กเกินไปจะทำให้การคำนวณช้าลง จึงเลือกค่า M=80 แซมเปิ้ล และค่า N = 240 แซมเปิ้ล

3) ความถี่ที่ใช้ในการสุ่มสัญญาณ

เนื่องจาก ความถี่ที่ใช้ในการแซมปลิ้ง มากกว่าสองเท่าของความถี่เสียง $f_s > 2f_{max}$ เนื่องจากความถี่ของเสียงพูดมีความถี่อยู่ในช่วง 300 Hz – 3400 Hz จึงใช้ความถี่แซมปลิ้ง 8kHz

ดังนั้น ช่วงเวลาในการวิเคราะห์แต่ละเฟรม คือ 240/8000=30 ms และระยะที่ใช้ในการเลื่อนแต่ละเฟรมคือ 80/8000 = 10 ms

4) การเลือกวินโดว์ที่เหมาะสมสำหรับการวิเคราะห์เสียง

โดยพิจารณาลักษณะสเปคตรัม คือ

- ความถี่รีโซลูชั่นสูง (high frequency resolution) คือมีโลบหลักแคบและแหลม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-การลดทอน (Attenuation) นอกช่วงความถี่ที่ผ่านได้ต่ำ คือไซด์โลบมีค่าน้อยฟังก์ชันวินโดว์มีหลายชนิด แต่ละชนิดเหมาะสมที่สุดที่จะนำมาใช้ได้แก่ แฮมมิ่งวินโดว์(Hamming Window) ซึ่งมีค่าโลบต่ำ (-40 dB) และเมนโลบแคบพอใช้มีสมการดังนี้

$$w(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right) \quad \text{เมื่อ} \quad n = 0, 1, 2, \ldots, N\text{-}1$$

5) การหาคุณลักษณะของเสียง

ใช้ในการประมาณพันธะเชิงเส้นในการวิเคราะห์หาค่าสัมประสิทธิ์ LPC ซึ่งการประมาณเชิงเส้นที่เลือกใช้คือ วิธีอัตตสัมพัทธ์ (autocorrelation method) ซึ่งวิธีนี้มีการคำนวณที่ง่ายกว่าวิธีอื่น ๆ และมีความแน่นอนด้านเสถียรภาพ อีกทั้งมีการเก็บข้อมูลที่น้อยกว่า

เนื่องจากการวิเคราะห์โดยวิธีอัตตสัมพัทธ์ อันดับการประมาณเชิงเส้น (P) ที่มากจะทำให้การประมาณเสียงมีความใกล้เคียงมากยิ่งขึ้น แต่ถ้าอันดับ P มีค่ามากเกินไปจะทำให้การคำนวณมีความยุ่งยากและใช้เวลานาน ดังนั้นเพื่อความเหมาะสมค่าอันดับ P ที่ใช้คือ 12

3.4 การสร้างโค้ดบุค(codebook)

จากการทดสอบ (L.R. Rabiner.S.E. Levinson และ Sondhi. 1982.) จะได้ว่าที่ขนาดโค้ดบุคเท่ากับ 128 จะมีค่าความคลาดเคลื่อน ใช้วิธีการคำนวณแบบ Square error distortion ในการหาระยะทางเนื่องจากเป็นวิธีง่าย และรวดเร็ว

การสร้างโค้ดบุค โดยนำเทรนนิ่งเซตที่ได้จากการประมาณเชิงเส้นมาผ่านกระบวนการดังนี้

1) สุ่มค่าโค้ดบุคเริ่มต้นมา 128 ตัว ตัวละ 19 มิติ

2) หาระยะทางระหว่างโค้ดบุคกับเทรนนิ่งเซต แต่ละตัวโดยใช้ความคลาดเคลื่อนกำลังสอง

3) จัดกลุ่มของเวกเตอร์อินพุตโดยพิจารณาจากระยะทางที่น้อยที่สุด

4) หาจุดศูนย์กลางของกลุ่ม

5) ทำขั้นตอน 3 และ 4 จนกว่าความคลาดเคลื่อนรวมจะน้อยกว่า 0.0001 ซึ่งจุดศูนย์กลางที่ได้ก็คือโค้ดบุคนั่นเอง

6)ทำการหาโค้ดบุคเพิ่มเติม โดยใช้จำนวน โค้ดบุค 32 และ 64

3.5 การสร้างแบบจำลอง HMM ของเสียงมีขั้นตอนดังนี้

1) สุ่มค่าเริ่มต้น a,b และกำหนดให้ π=[100000] ตามเงื่อนไขในการใช้แบบจำลองแบบ Left-Right Model

2) หาค่า α,β จากค่า a , b เริ่มต้น และลำดับค่าปรากฏ $O = \{o_1 o_2 o_3 \cdots o_T\}$ ซึ่งเรียกว่าลำดับเทรนนิ่ง ตามวิธีของ Forword – Backword Procedure โดยใช้ลำดับของค่าปรากฏหลาย ๆ ลำดับเข้ามาเทรนเพื่อความถูกต้องมากขึ้น

3) การทำสเกลลิ่ง เพื่อให้ค่าอยู่ในย่านที่คอมพิวเตอร์สามารถคำนวณได้อย่างถูกต้อง

4) หาค่าพารามิเตอร์ λ ที่ให้ค่าความน่าจะเป็นสูงสุด ที่จะเป็นแบบจำลอง λ ที่เหมาะสมของคำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) ตรวจสอบค่าพารามิเตอร์ a,b,ของแบบจำลองที่ได้ ว่าลู่เข้าหรือยัง โดยใช้วิธีการคำนวณค่า a,b ซ้ำ ประมาณ 50 รอบ เมื่อมีการเปลี่ยนแปลงน้อยมากจนเป็นที่พอใจตามระดับค่าที่ตั้งในที่นี้ให้ใช้เท่า 10^{-5} ก็จะหยุด และจะได้ค่าพารามิเตอร์ a,b และ π แบบจำลองที่ต้องการ

6) เก็บค่าพารามิเตอร์ a,b,π ที่ได้จากข้อ 5 เป็นพารามิเตอร์ของแบบจำลองไว้

3.6 ขั้นตอนการตัดสินใจ

3.6.1 การหาดัชนีโค้ดบุค

โดยการนำเวกเตอร์เสียงจากการประมาณเชิงเส้นทีละเสียง แล้วเปรียบเทียบกับโค้ดบุคที่ได้จากการสร้างในขั้นตอนการเรียนรู้ ทีละเฟรม โดยวิธีความคลาดเคลื่อนกำลังสอง เวกเตอร์เสียงห่างจากโค้ดบุคใดน้อยที่สุด จะได้ว่าเป็นดัชนีโค้ดบุคของเฟรมเสียงนั้น และเก็บดัชนีโค้ดบุคของแต่ละเฟรมในแต่ละเสียงไว้เป็นลำดับค่าปรากฏ (Observation sequence) สำหรับการสร้างแบบจำลองต่อไป

3.6.2 การรู้จำเสียง

หลังจากที่ได้ แบบจำลอง HMM ของแต่ละคำแล้ว เมื่อมีลำดับของ ค่าปรากฏ $O = \{o_1 o_2 o_3 \cdots o_T\}$ ของเสียง Unknown ซึ่งเป็นเสียงที่ต้องการทดสอบเข้ามา เราจะทำการคำนวณหาความน่าจะเป็น $P(o \mid \lambda)$ ทุกแบบจำลองของแต่ละศัพท์โดยใช้วิธี Viterbi Algorithm แล้วเลือกเอาคำที่มีความน่าจะเป็นสูงสุด ซึ่งก็คือ คำศัพท์ที่แบบจำลองการจำได้นั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

4.1 การทดลอง

1) เป็นการหาขอบเขตของคำโดยการหาค่าพลังงาน และการนอร์มอลไลซ์ เสียง 0-9
2) เป็นการหาค่าพารามิเตอร์ ซึ่งได้แก่ สัมประสิทธิ์แอลพีซี สัมประสิทธิ์เซปสตรัม และสัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก
3) ทำการหาโค้ดบุค 32,64,128 และแสดงผลรวมทั้งเวลาในการคำนวณ
4) เป็นการทดสอบการจดจำเสียงพูด ในกรณีต่างๆ

4.2 ขั้นตอนการเรียนรู้

1. เก็บเสียงที่ต้องการไว้เพื่อสร้างโมเดล โดยเราจะเก็บเสียง 0-9 ผู้ชาย 5 คนคนละ 5 ครั้ง รวมเป็น 250 เสียง ทำการเก็บเสียงต่างเป็น m(xx)t(y)s(z).wav ในโฟเดอร์ Sound
2. โดยทำการเก็บเสียง ขนาด 8 บิต 8 kHz
3. นำข้อมูลเสียงที่ได้มาทำการหาค่าพารามิเตอร์ ที่เป็นตัวแทนของเสียงโดยใช้วิธี Linear Predictive Coding(LPC) และปรับปรุงอีกขั้น
4. นำพารามิเตอร์เสียง ทั้งหมดจากทุกกลุ่มมาทำการลดจำนวนข้อมูล โดยวิธีการ Vector Quantization (VQ) ซึ่งจะได้เวคเตอร์ตัวแทนของเสียงที่ระบบสามารถรับรู้ได้ทั้งหมดออกมาเรียกว่า โค้ดบุค(Codebook)
5. ทำการหา Codebook Index ของแต่ละเสียง โดยการนำเอาค่า Weight ที่ปรับปรุงมาจาก LPC มาผ่านขั้นตอนอีก โดยแต่ละเสียงจะได้ค่าของ Index ออกมา
6. สร้างโมเดลของเสียง โดยการนำ Codebook Index ของแต่ละกลุ่มเสียงมาผ่านขั้นตอนการสร้างโมเดลด้วย วิธี Hidden Markov Model (HMM) เมื่อทำจนครบทุกกลุ่มเสียง ก็จะได้โมเดลครบทุกเสียงและพร้อมที่จะนำไปใช้งานในขั้นตอนการวิเคราะห์ต่อไป

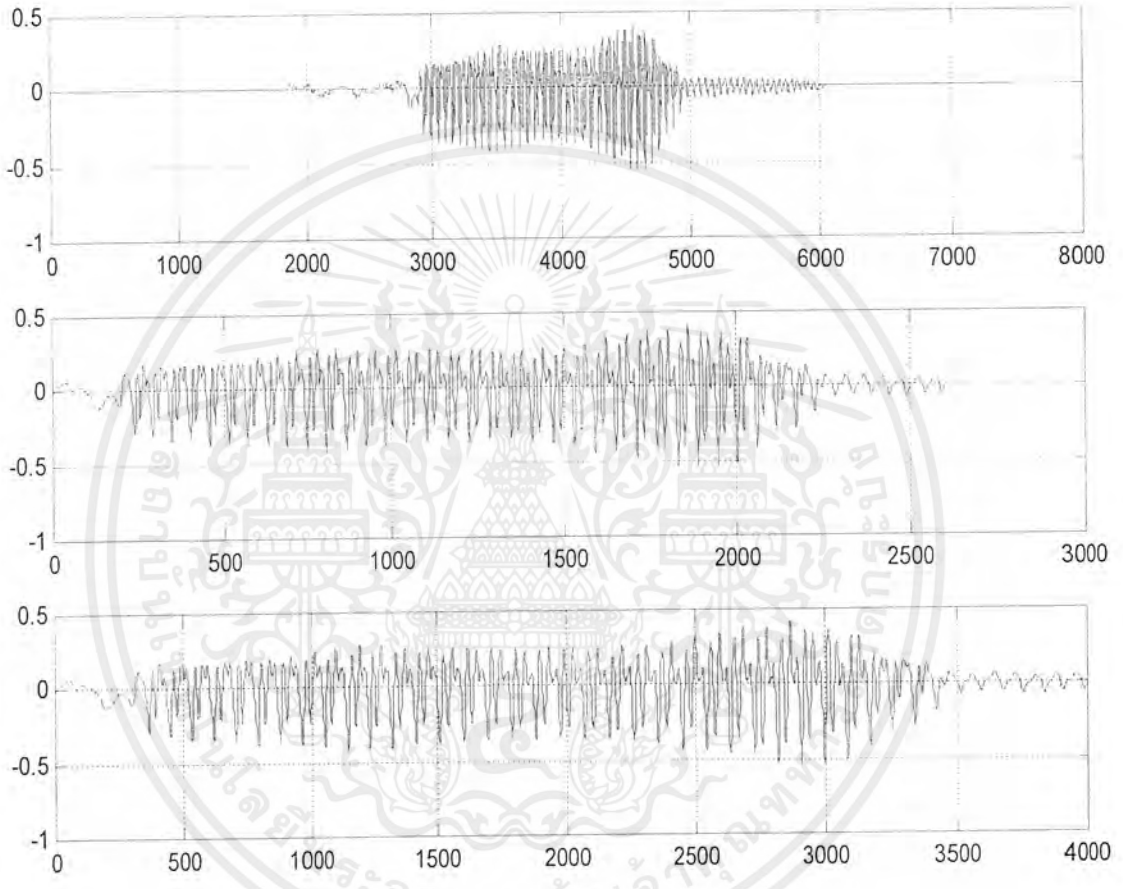
4.3 ขั้นตอนการวิเคราะห์

เป็นขั้นตอนที่จะวิเคราะห์เสียงที่ไม่ทราบค่าว่าเป็นคำใด ว่ามีความเหมือนกับโมเดลของเสียงใดที่เก็บไว้มากที่สุด ขั้นตอนในช่วงแรกจะคล้ายกับขั้นการเรียนรู้ ดังนี้

1. นำเสียงที่ต้องการทดสอบ เพียง 1 เสียง มาผ่านขั้นตอน LPC และหาค่า Weight จะได้พารามิเตอร์ของเสียงนั้น และ นำพารามิเตอร์มาเปรียบเทียบ Codebook จะได้ Codebook Index
2. นำ Codebook Index มาหาค่าความน่าจะเป็นกับทุกโมเดลที่เก็บไว้ทั้งหมด โดยวิธี Viterbi Algorithm ก็จะได้ค่าความน่าจะเป็นของแต่ละโมเดลออกมา ค่าความน่าจะเป็นเมื่อเปรียบเทียบกับโมเดลใดมีค่าสูงที่สุด ผลจะได้ว่าเสียงที่นำมาทดสอบตรงกับโมเดลใด
3. แสดงผลของเสียงที่ได้ว่าเป็นเสียงใด

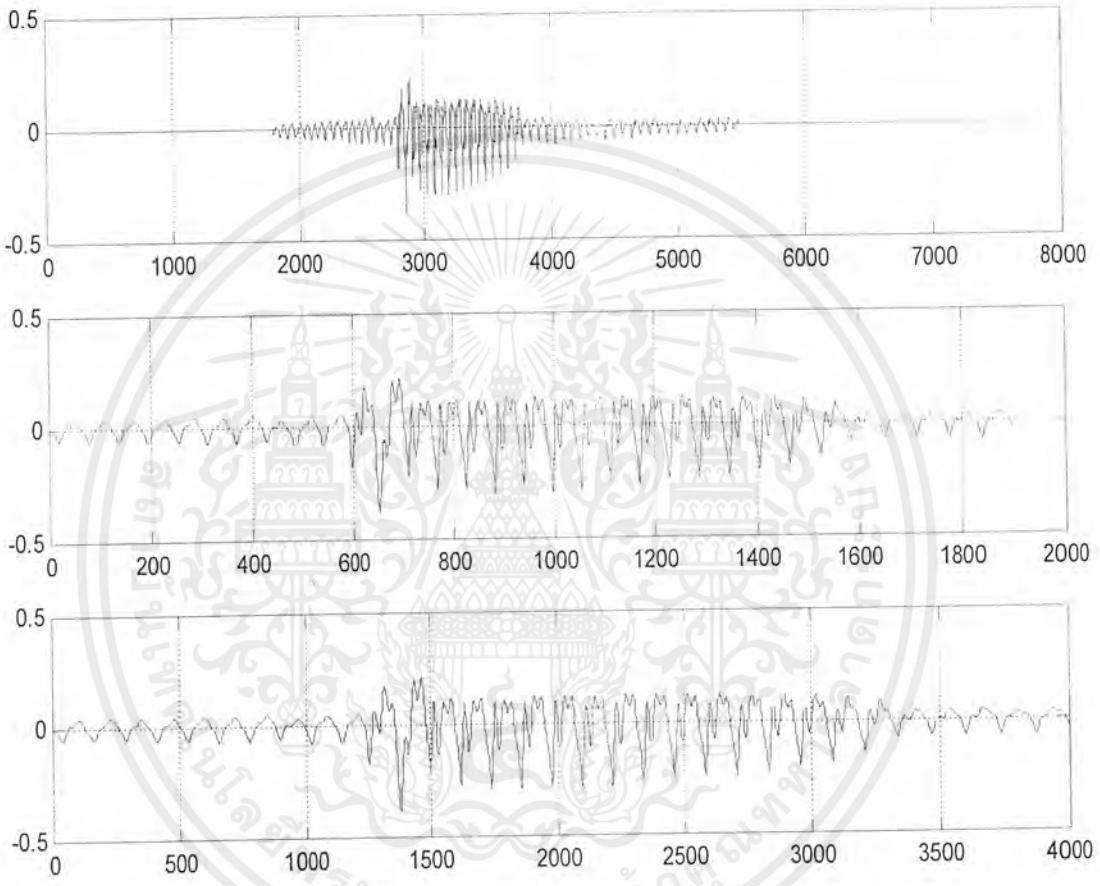
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.1 การหาขอบเขตของคำ และ การนอร์มอลไลซ์ ของเสียง 0- 9



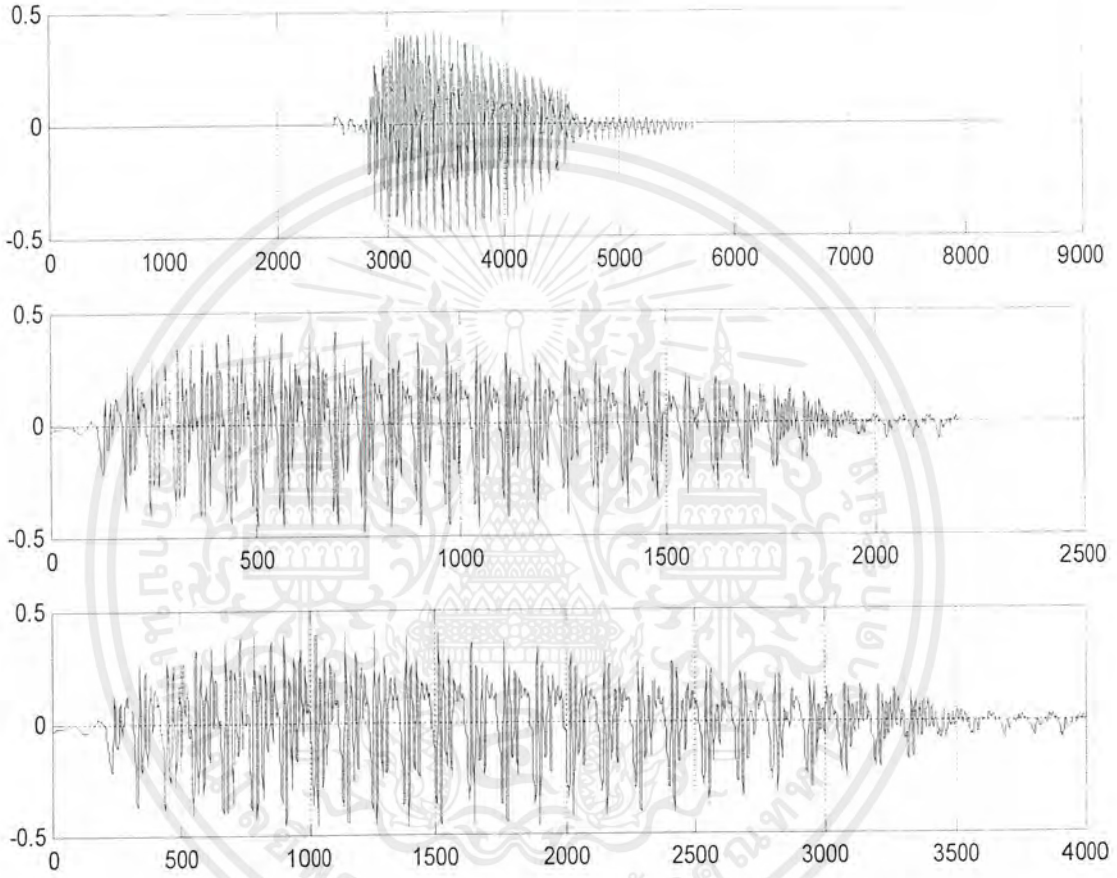
รูปที่ 4.1 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขศูนย์ โดย
รูปที่หนึ่งแสดงสัญญาณก่อนการหาขอบเขต
รูปที่สองแสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน
รูปที่สามแสดงการนอร์มอลไลซ์ให้มี 4000 Sample

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขหนึ่ง โดย
รูปที่หนึ่งแสดงสัญญาณก่อนการหาขอบเขต
รูปที่สองแสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน
รูปที่สามแสดงการนอร์มอลไลซ์ให้มี 4000 Sample

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



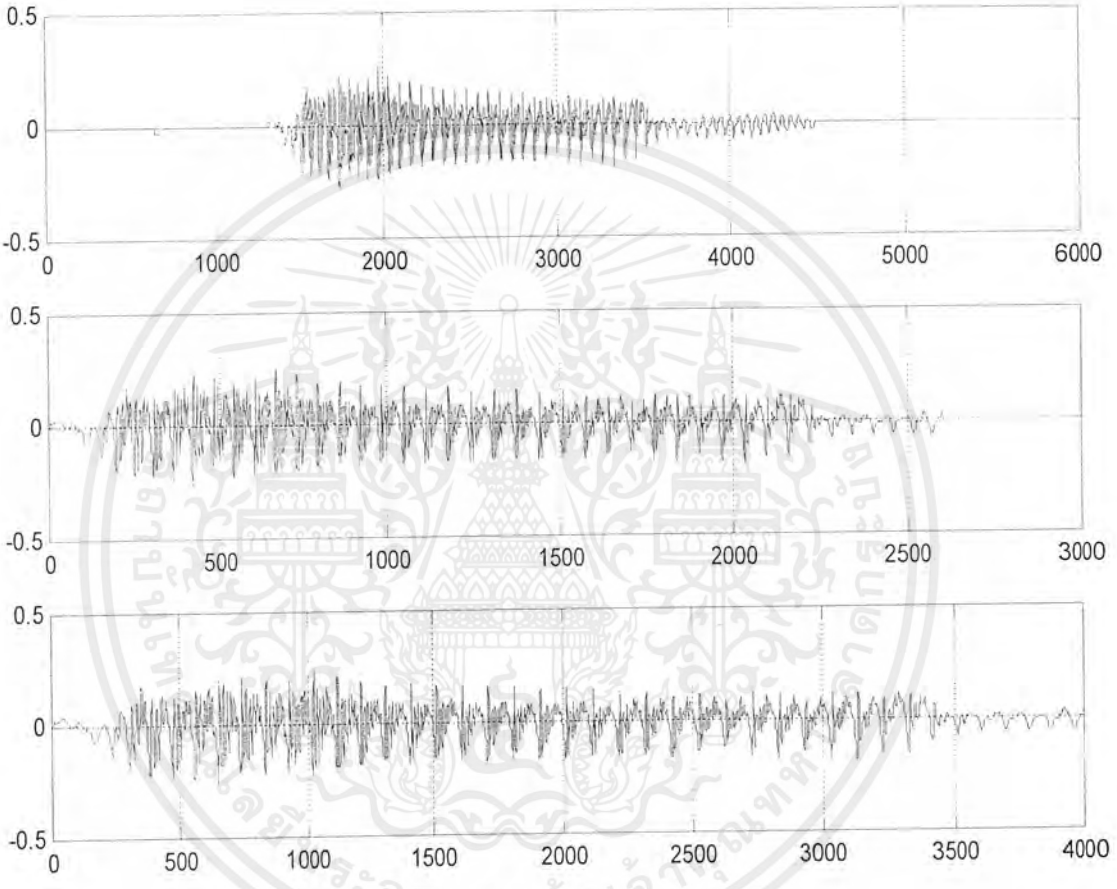
รูปที่ 4.3 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขสอง โดย

รูปที่หนึ่งแสดงสัญญาณก่อนการหาขอบเขต

รูปที่สองแสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน

รูปที่สามแสดงการนอร์มอลไลซ์ให้มี 4000 Sample

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



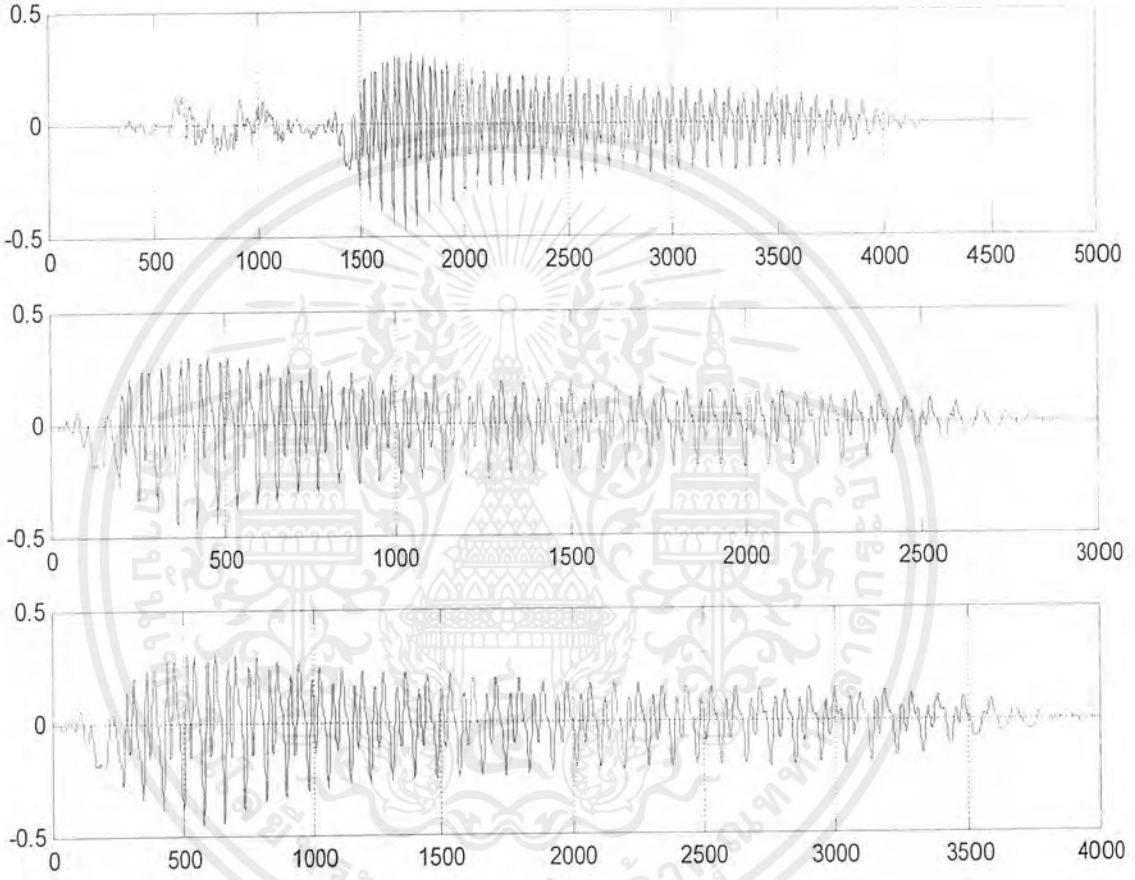
รูปที่ 4.4 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขสาม โดย

รูปที่หนึ่งแสดงสัญญาณก่อนการหาขอบเขต

รูปที่สองแสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน

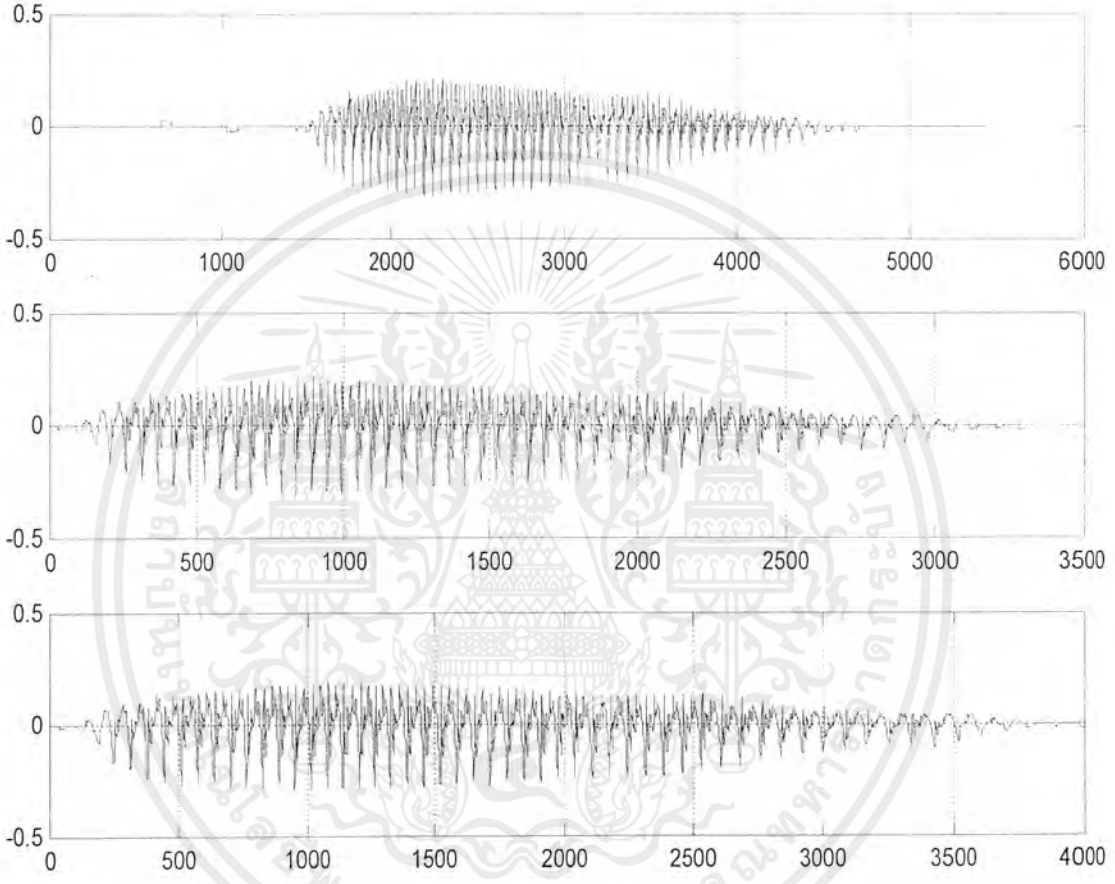
รูปที่สามแสดงการนอร์มอลไลซ์ให้มี 4000 Sample

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



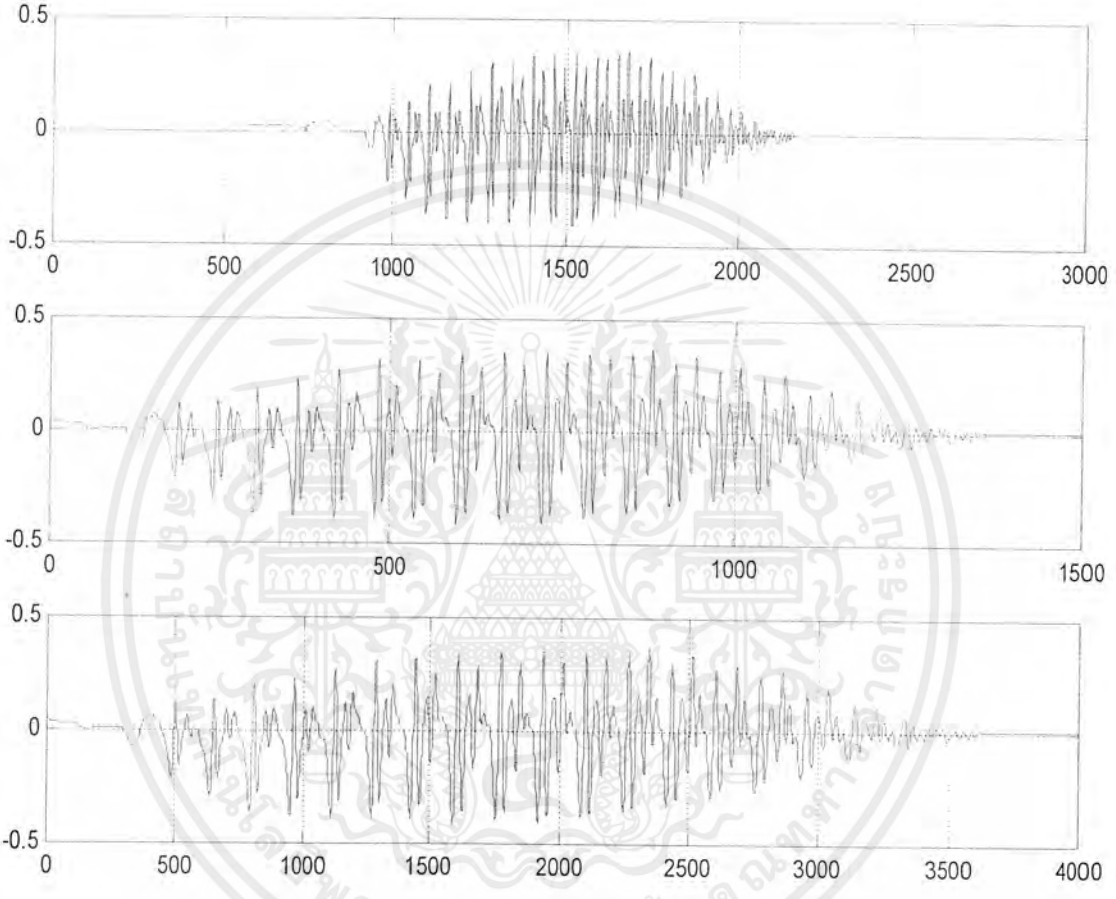
รูปที่ 4.5 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขสี่ โดย
รูปที่หนึ่งแสดงสัญญาณก่อนการหาขอบเขต
รูปที่สองแสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน
รูปที่สามแสดงการนอร์มอลไลซ์ให้มี 4000 Sample

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



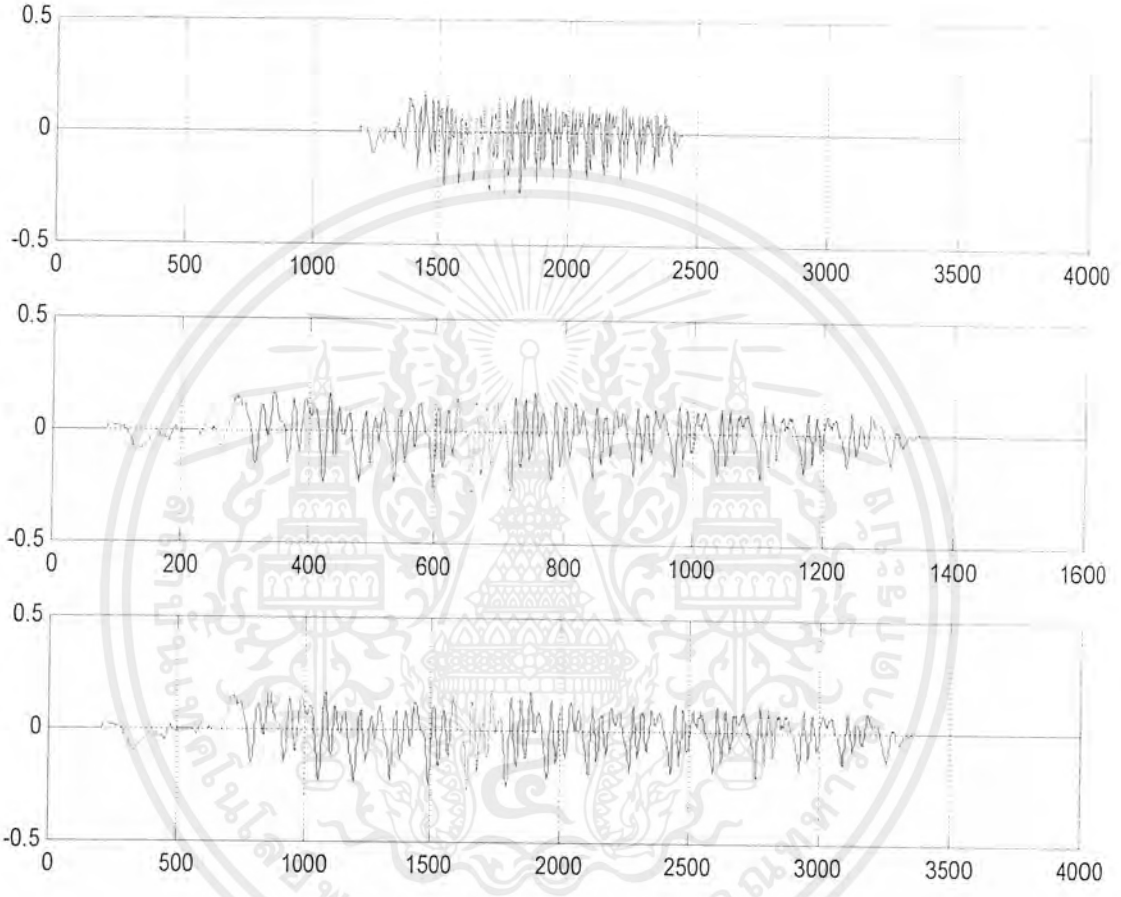
รูปที่ 4.6 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขห้า โดย
รูปที่หนึ่งแสดงสัญญาณก่อนการหาขอบเขต
รูปที่สองแสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน
รูปที่สามแสดงการนอร์มอลไลซ์ให้มี 4000 Sample

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขหก โดย
รูปที่หนึ่งแสดงสัญญาณก่อนการหาขอบเขต
รูปที่สองแสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน
รูปที่สามแสดงการนอร์มอลไลซ์ให้มี 4000 Sample

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



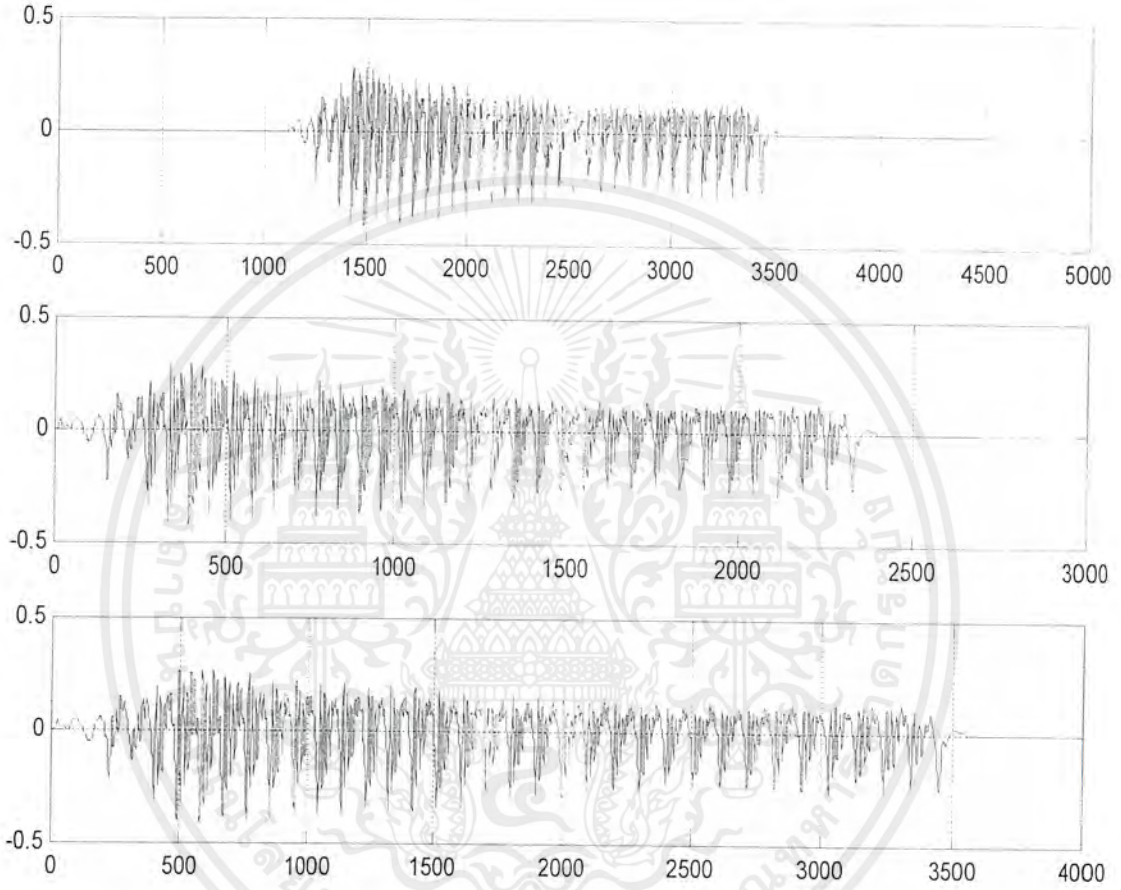
รูปที่ 4.8 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขเจ็ด โดย

รูปที่หนึ่งแสดงสัญญาณก่อนการหาขอบเขต

รูปที่สองแสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน

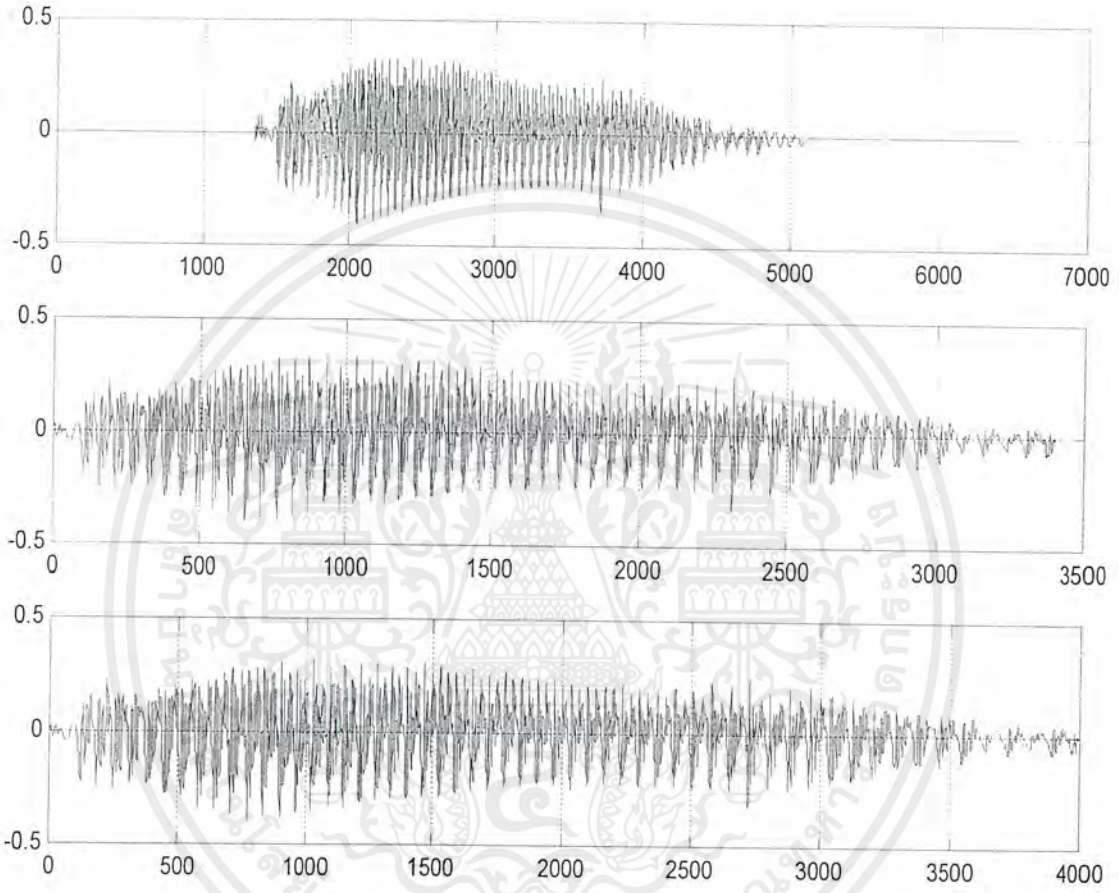
รูปที่สามแสดงการนอร์มอลไลซ์ให้มี 4000 Sample

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขแปด โดย
รูปที่หนึ่งแสดงสัญญาณก่อนการหาขอบเขต
รูปที่สองแสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน
รูปที่สามแสดงการนอร์มอลไลซ์ให้มี 4000 Sample

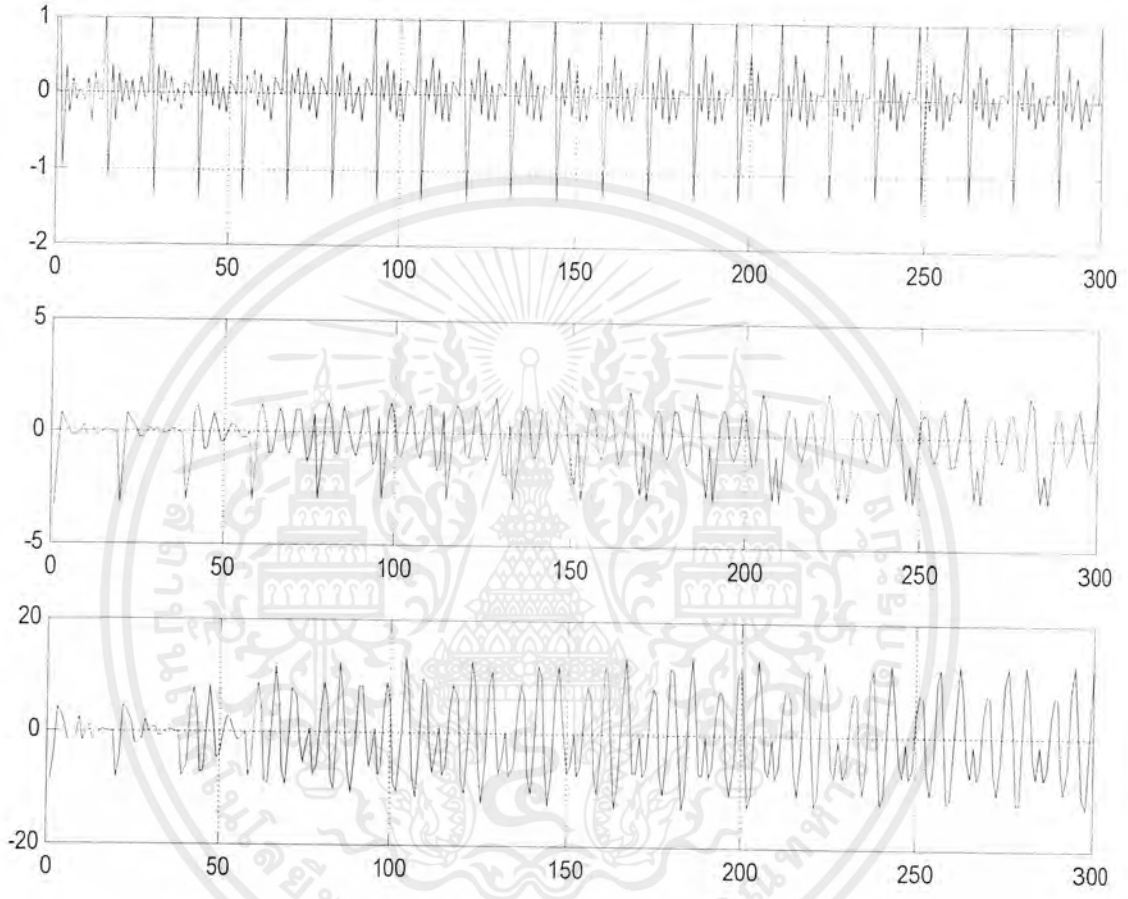
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 แสดงการหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขเก้า โดย
รูปที่หนึ่งแสดงสัญญาณก่อนการหาขอบเขต
รูปที่สองแสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน
รูปที่สามแสดงการนอร์มอลไลซ์ให้มี 4000 Sample

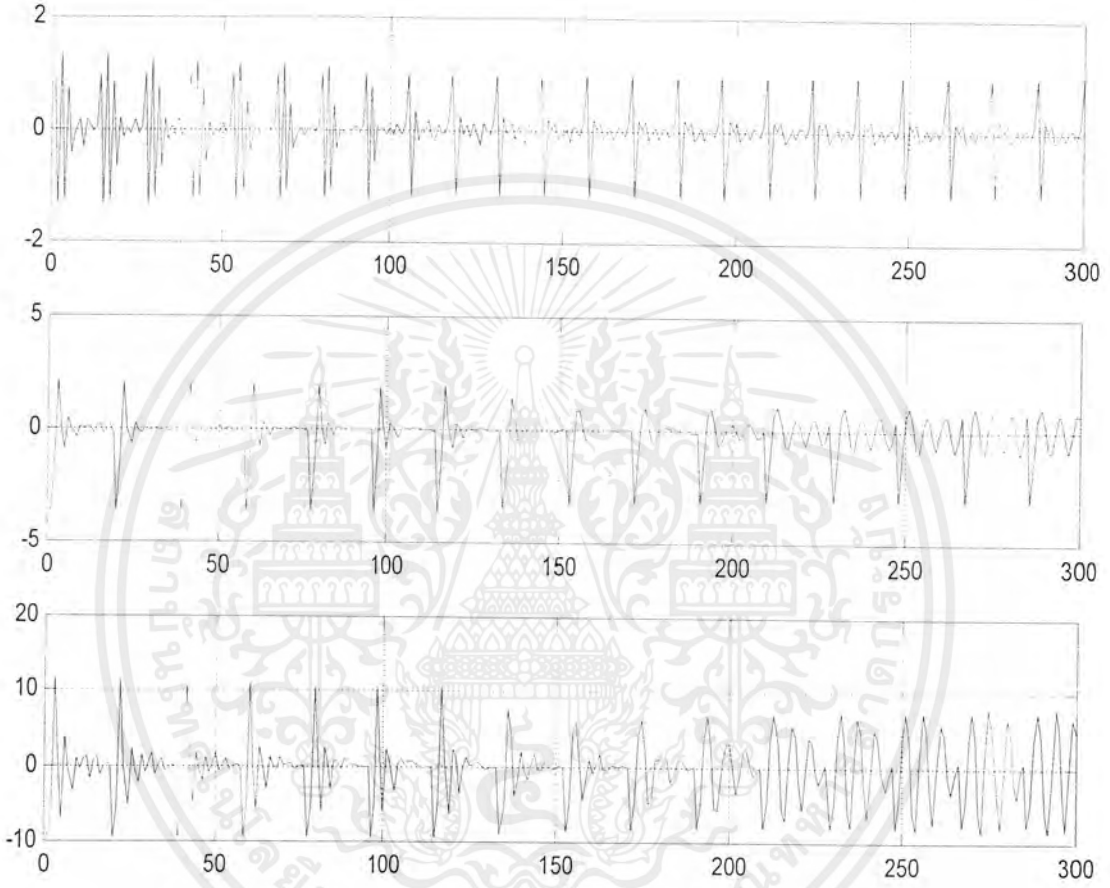
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 การหาค่าพารามิเตอร์สัมประสิทธิ์แอลพีซี สัมประสิทธิ์เซปสตรัม และ สัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก ของเลข 0-9



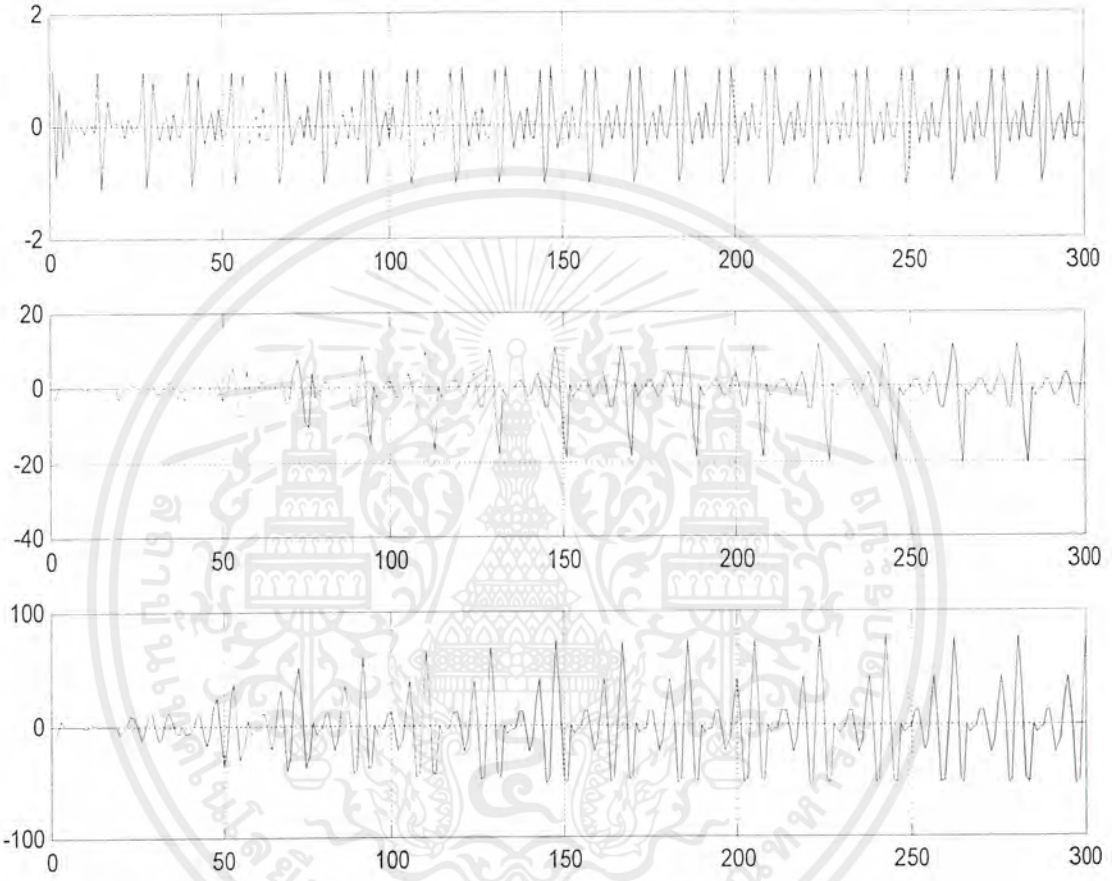
รูปที่ 4.11 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขศูนย์ โดย
รูปที่หนึ่งแสดงค่าสัมประสิทธิ์แอลพีซี
รูปที่สองแสดงค่าสัมประสิทธิ์เซปสตรัม
รูปที่สามแสดงค่าสัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



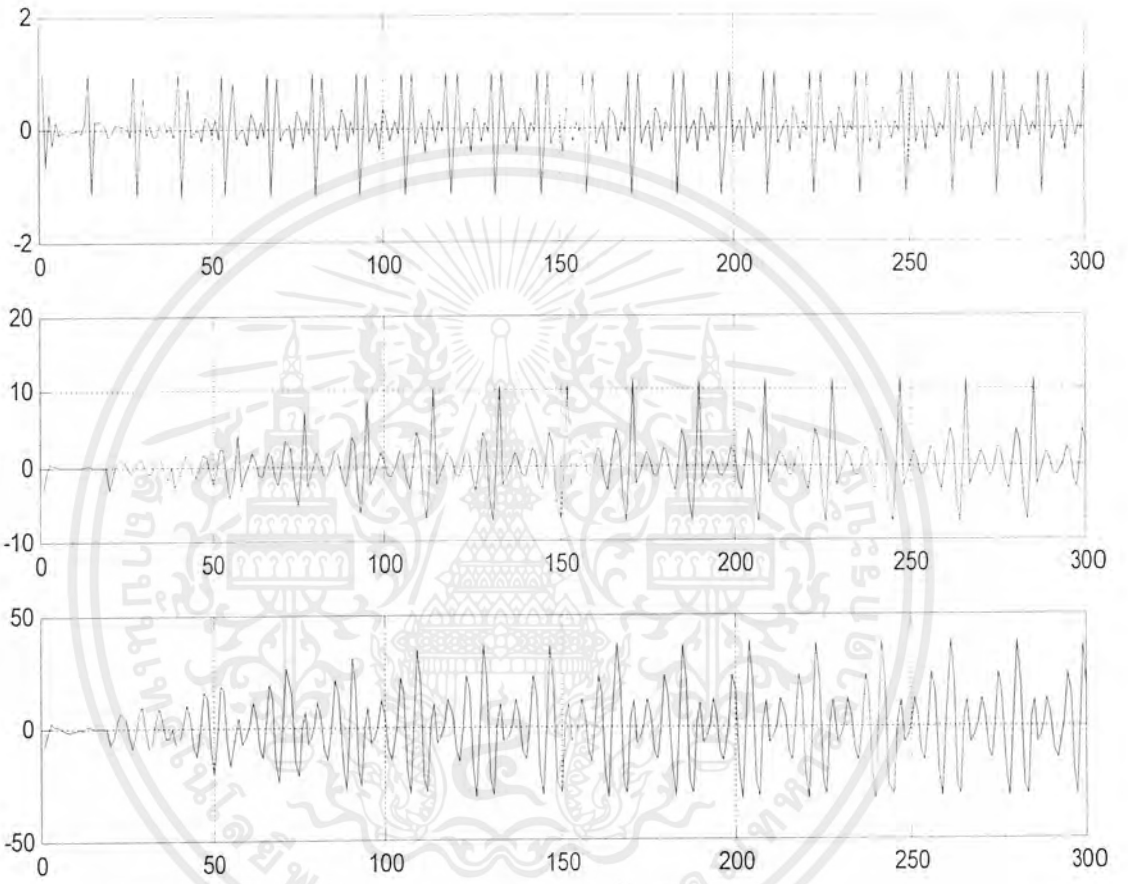
รูปที่ 4.12 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขหนึ่ง โดย
รูปที่หนึ่งแสดงค่าสัมประสิทธิ์แอลพีซี
รูปที่สองแสดงค่าสัมประสิทธิ์เซปสตรัม
รูปที่สามแสดงค่าสัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



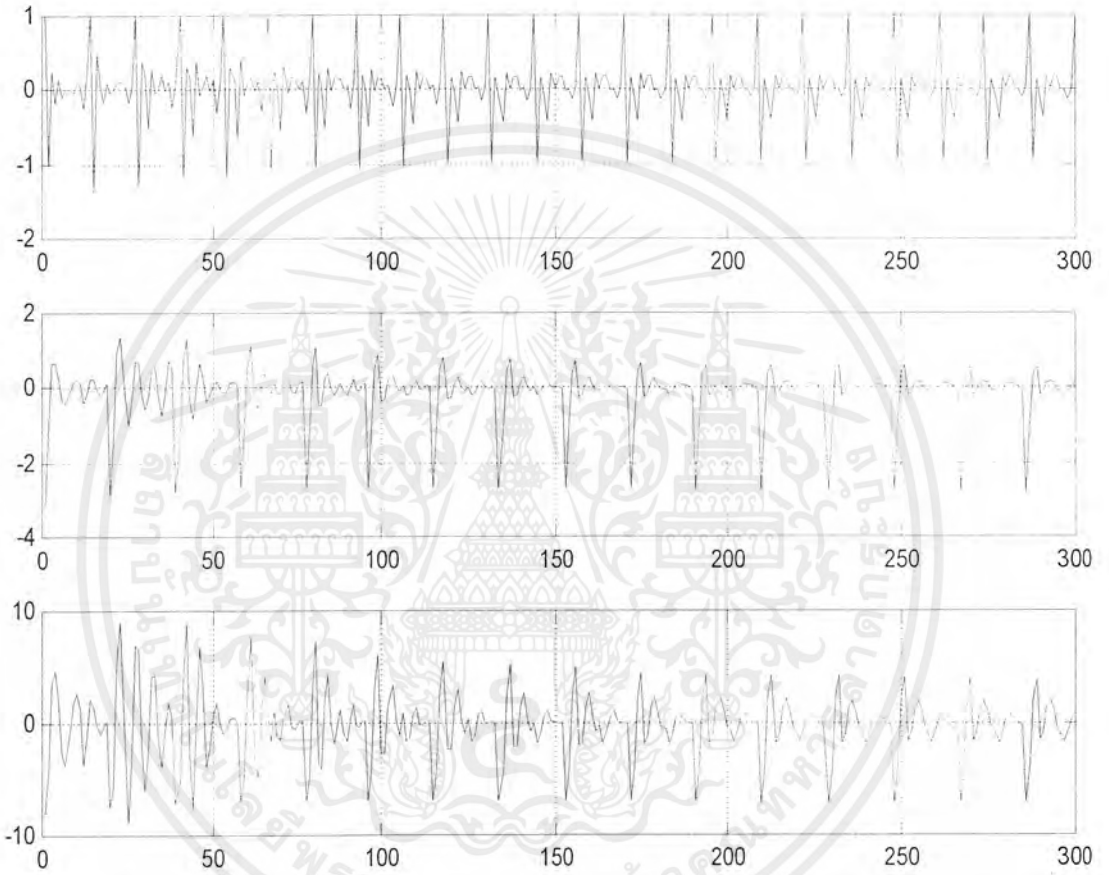
รูปที่ 4.13 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขสอง โดย
รูปที่หนึ่งแสดงค่าสัมประสิทธิ์แอลพีซี
รูปที่สองแสดงค่าสัมประสิทธิ์เซปสตรัม
รูปที่สามแสดงค่าสัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



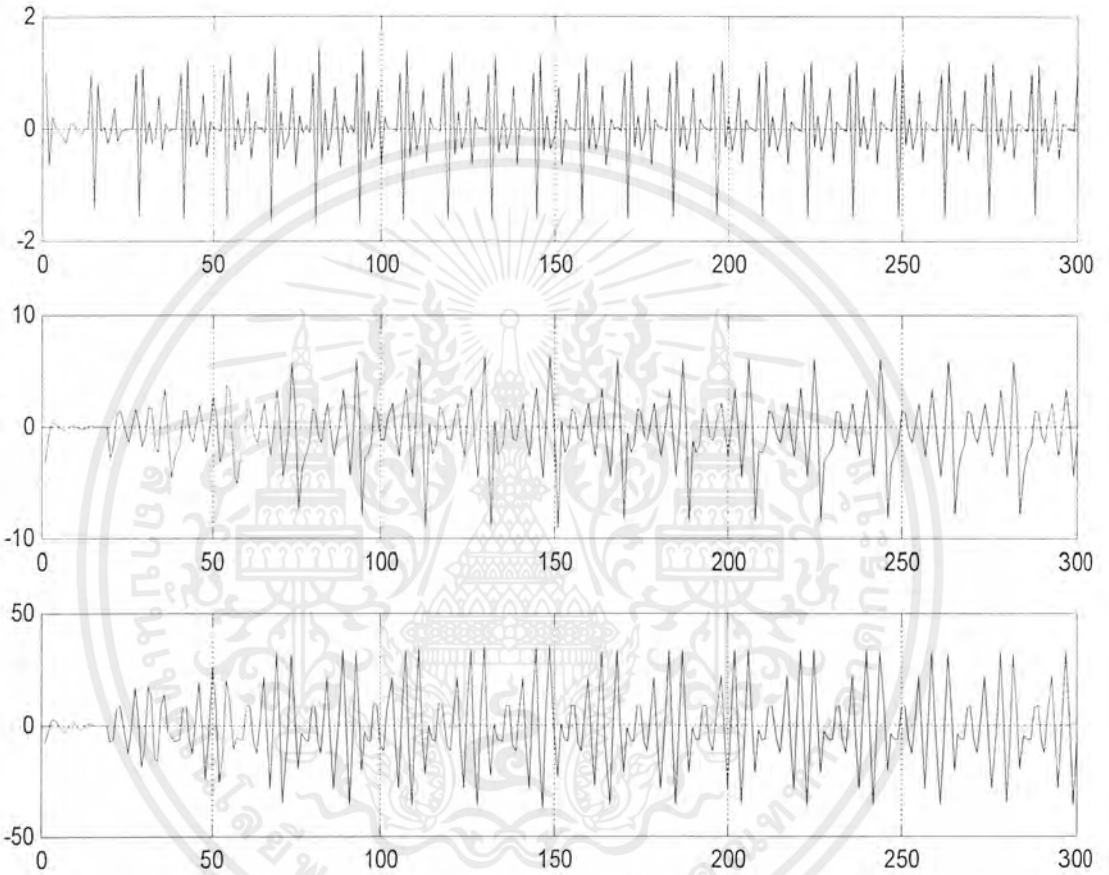
รูปที่ 4.14 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขสาม โดย
รูปที่หนึ่งแสดงค่าสัมประสิทธิ์แอลพีซี
รูปที่สองแสดงค่าสัมประสิทธิ์เซปสตรัม
รูปที่สามแสดงค่าสัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



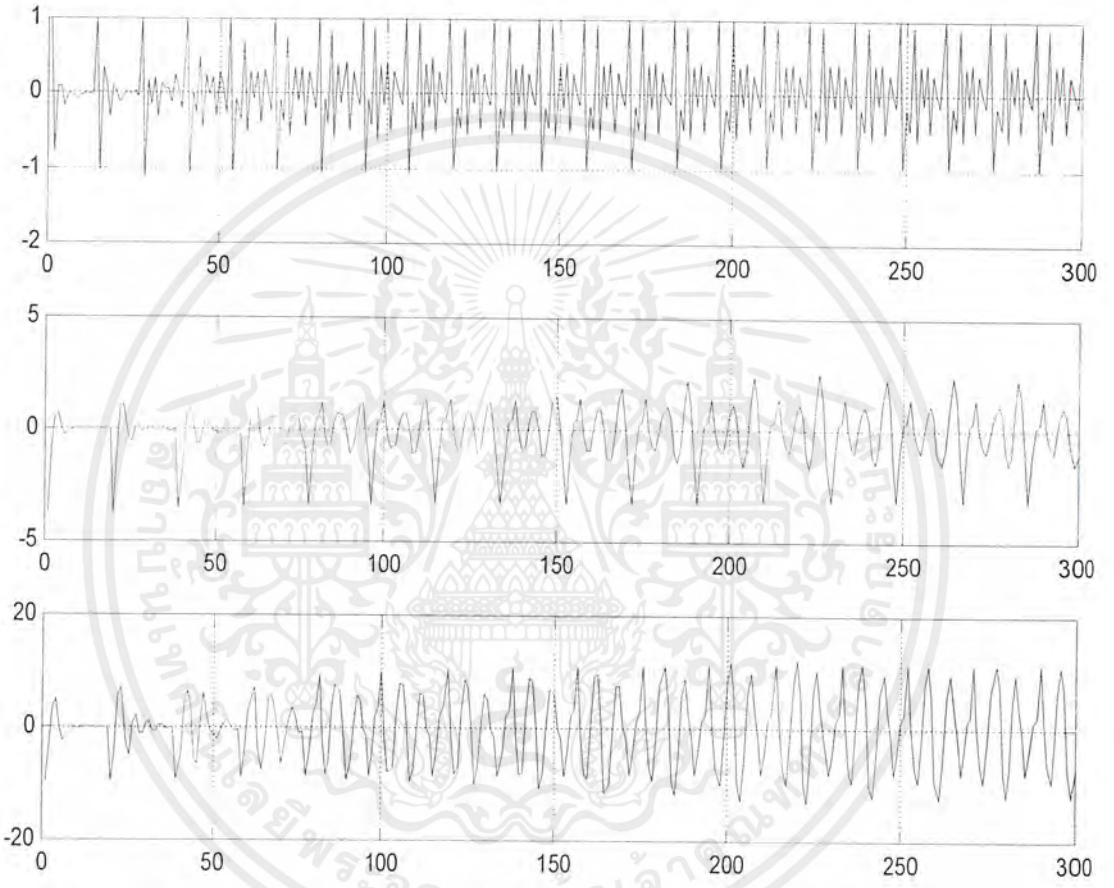
รูปที่ 4.15 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขสี่ โดย
รูปที่หนึ่งแสดงค่าสัมประสิทธิ์แอลพีซี
รูปที่สองแสดงค่าสัมประสิทธิ์เซปสตรัม
รูปที่สามแสดงค่าสัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



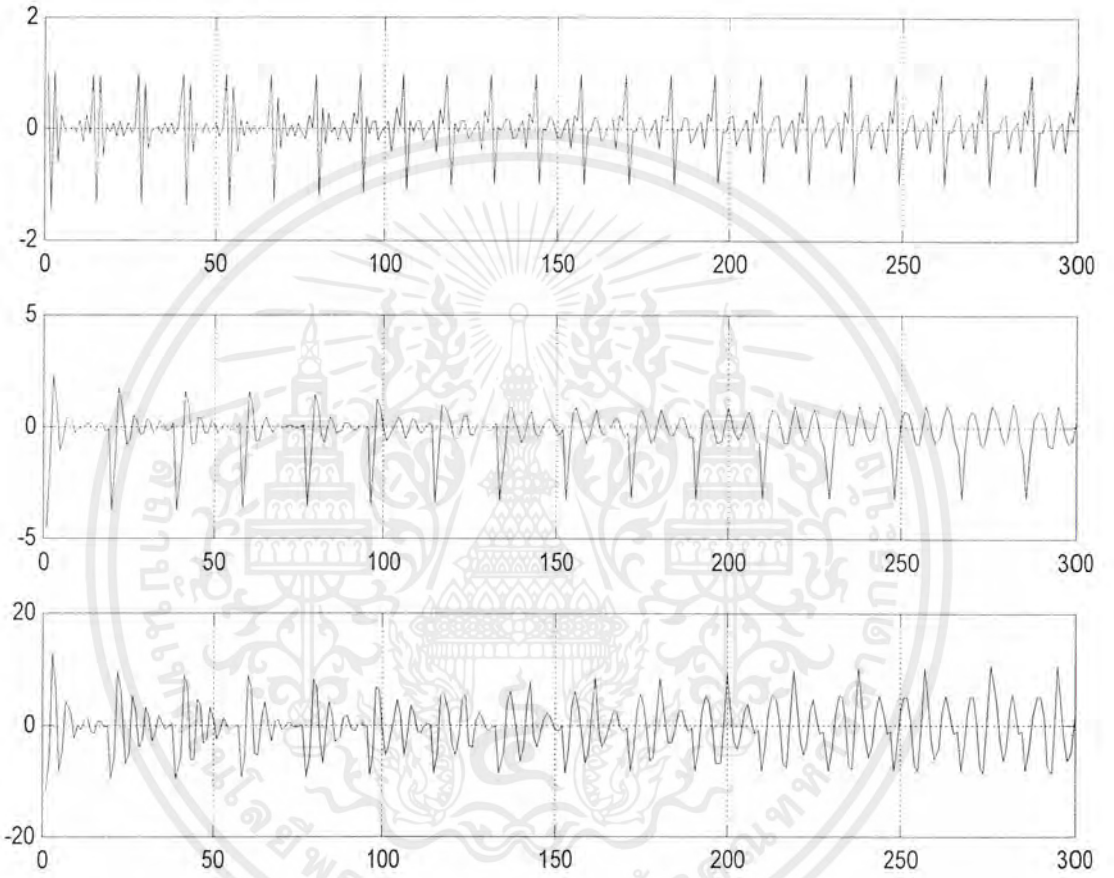
รูปที่ 4.16 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขห้า โดย
รูปที่หนึ่งแสดงค่าสัมประสิทธิ์แอลพีซี
รูปที่สองแสดงค่าสัมประสิทธิ์เซปสตรัม
รูปที่สามแสดงค่าสัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



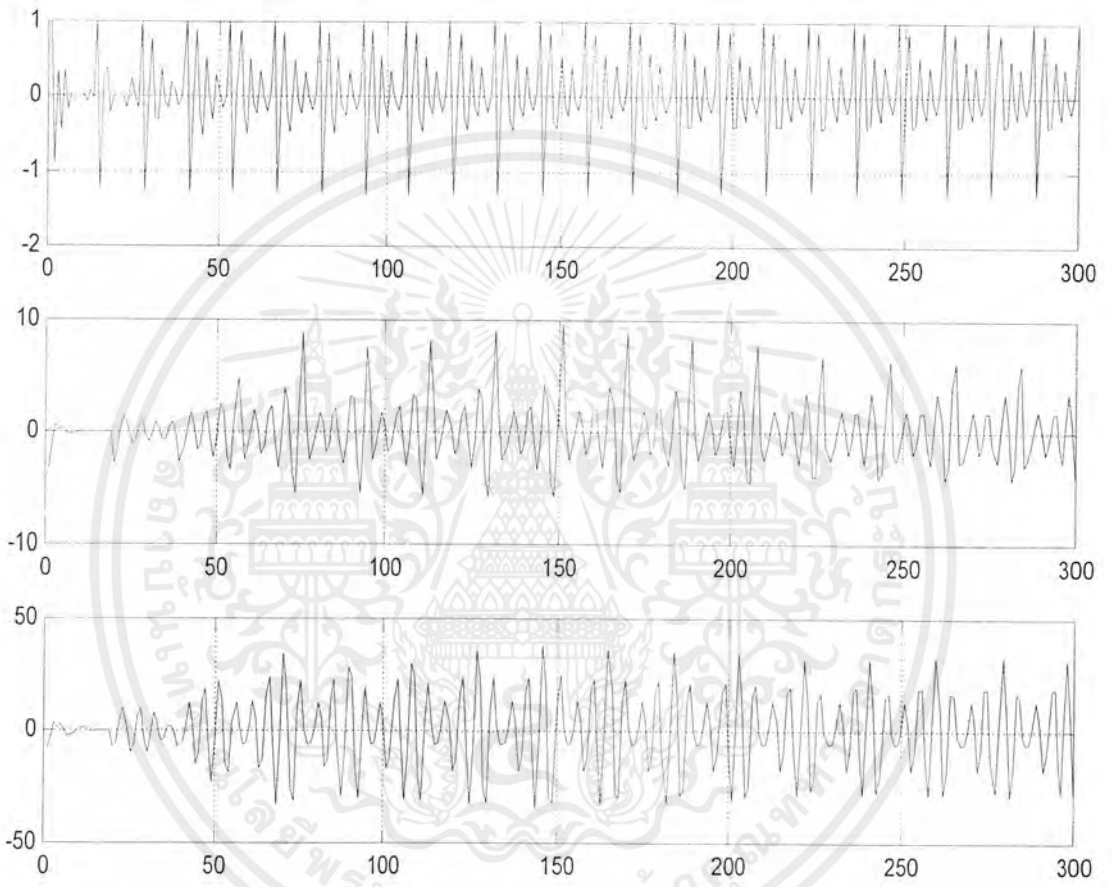
รูปที่ 4.17 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขหก โดย
รูปที่หนึ่งแสดงค่าสัมประสิทธิ์แอลพีซี
รูปที่สองแสดงค่าสัมประสิทธิ์เซปสตรัม
รูปที่สามแสดงค่าสัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



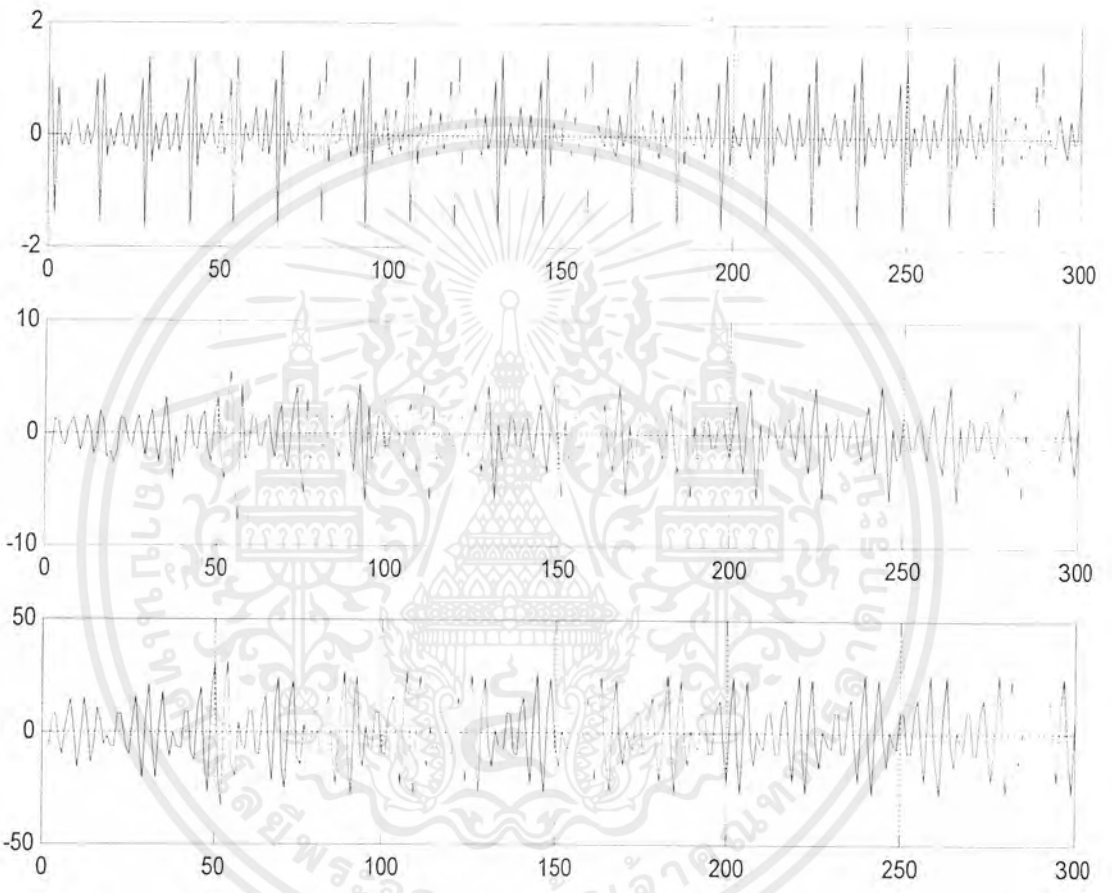
รูปที่ 4.18 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขเจ็ด โดย
รูปที่หนึ่งแสดงค่าสัมประสิทธิ์แอลพีซี
รูปที่สองแสดงค่าสัมประสิทธิ์เซปสตรัม
รูปที่สามแสดงค่าสัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.19 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขแปด โดย
รูปที่หนึ่งแสดงค่าสัมประสิทธิ์แอลพีซี
รูปที่สองแสดงค่าสัมประสิทธิ์เซปสตรัม
รูปที่สามแสดงค่าสัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก

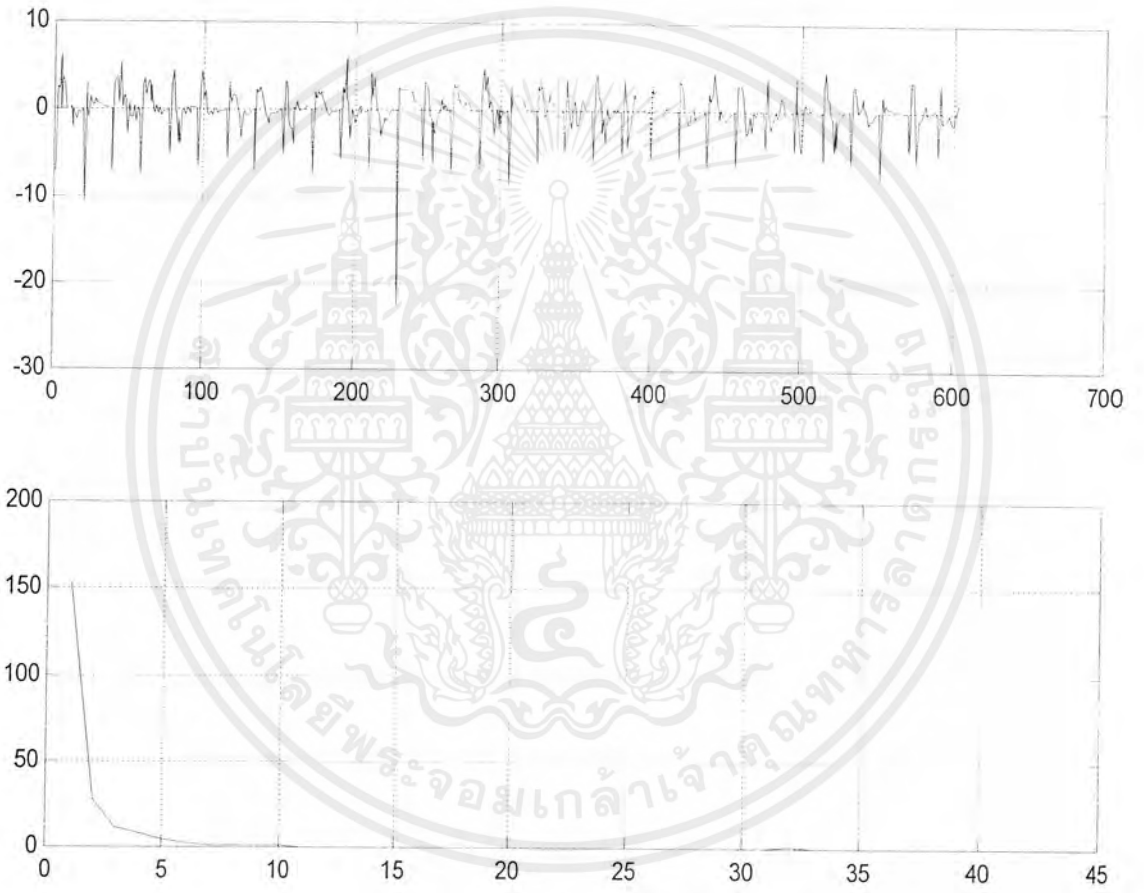
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.20 แสดงค่าสัมประสิทธิ์แบบต่างๆ ของสัญญาณเสียง เลขเก้า โดย
รูปที่หนึ่งแสดงค่าสัมประสิทธิ์แอลพีซี
รูปที่สองแสดงค่าสัมประสิทธิ์เซปสตรัม
รูปที่สามแสดงค่าสัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก

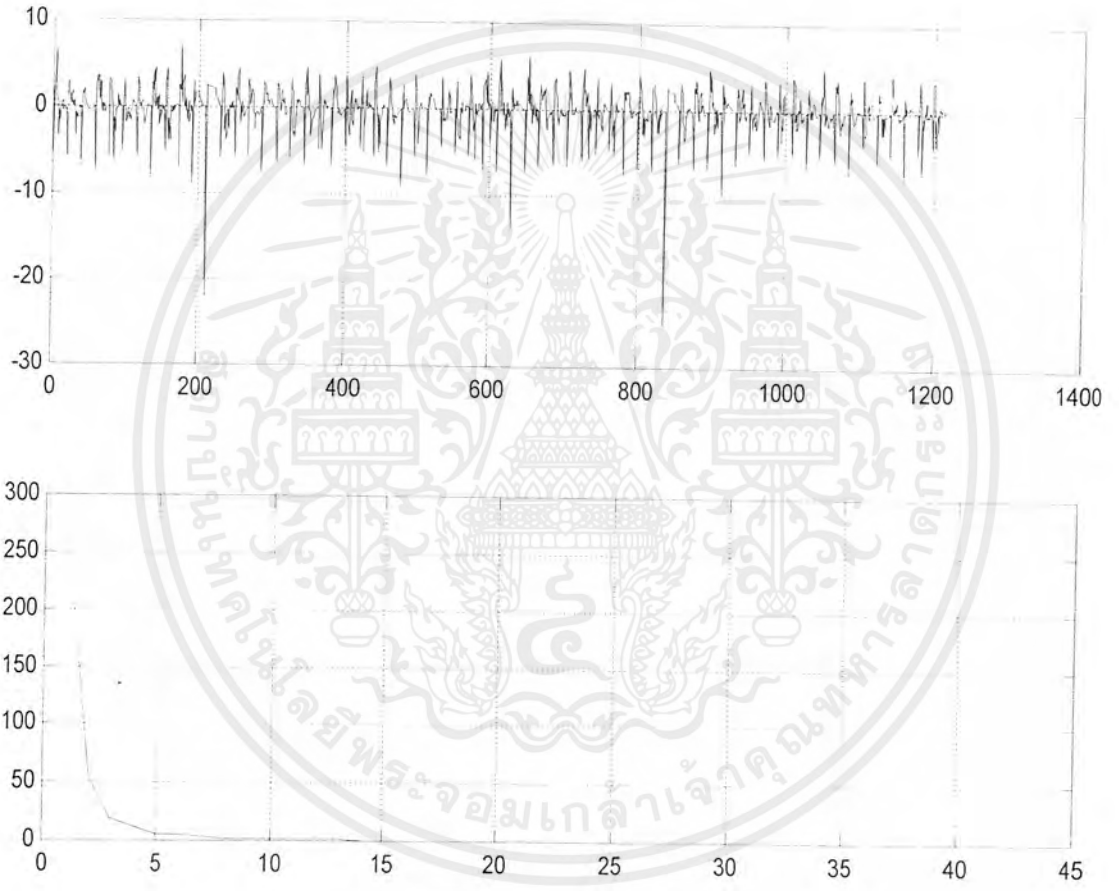
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.3 โค้ดบุค 32,64,128 ที่ได้และจำนวนรอบในการคำนวณ



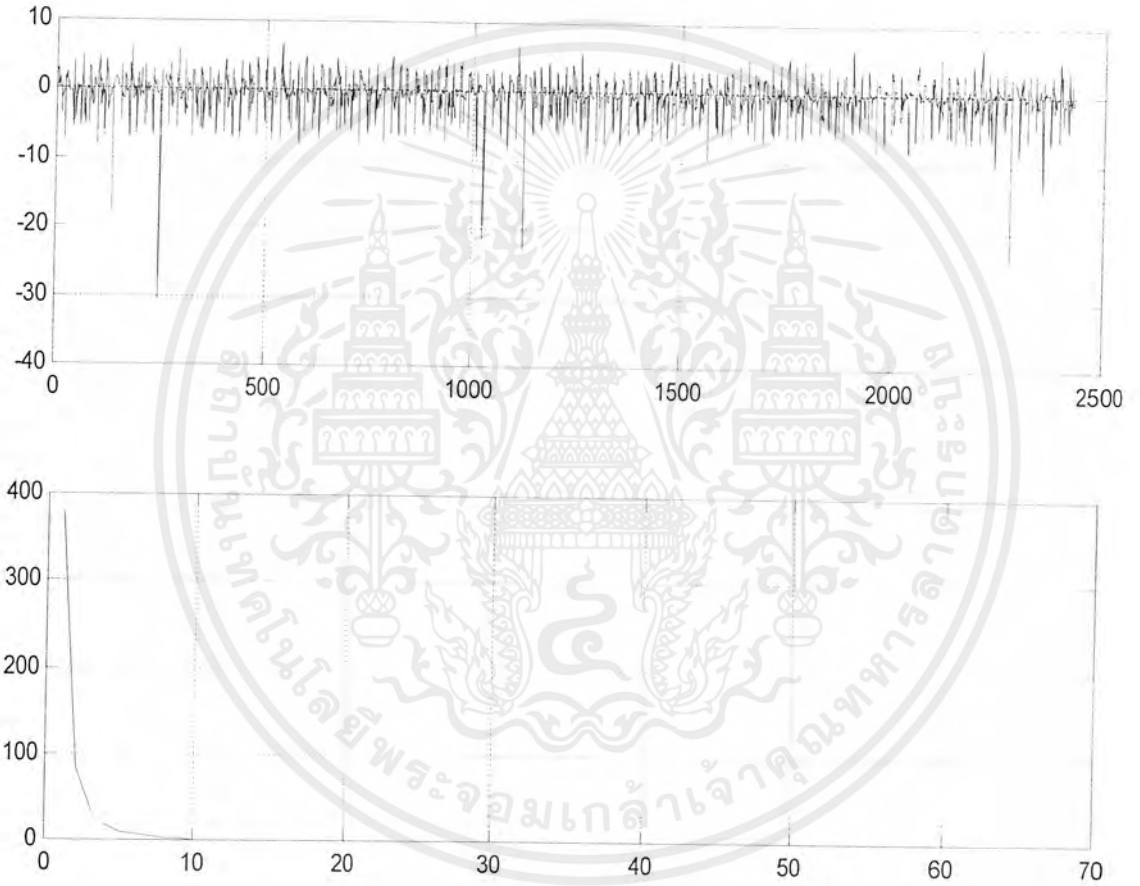
รูปที่ 4.21 แสดงโค้ดบุค 32 และจำนวนรอบจากการคำนวณทั้งหมด 41 รอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.22 แสดงโค้ดบุค 64 และจำนวนรอบจากการคำนวณทั้งหมด 43 รอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.23 แสดงโค้ดบุค 128 และจำนวนรอบจากการคำนวณทั้งหมด 68 รอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การทดสอบการจดจำเสียงพูดในกรณีต่างๆ

เป็นการทดสอบการจดจำเสียงพูด โดยแต่ละกรณีจะทำการทดสอบ 2 แบบ คือ ใช้เสียงจากผู้พูดต้นแบบที่ใช้สร้างแบบจำลอง และใช้เสียงจากบุคคลอื่นหรือเรียกว่าเสียง unknown แสดงโดยตารางผลการทดลองในแต่ละกรณี ดังนี้

กรณีที่ 1 แบบจำลองจากเสียงต้นแบบพูด 0 ถึง 9 เสียงละ 20 ครั้ง รวมเป็น 200 เสียง โดยใช้ codebook 32

เสียงที่ต้องการทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	13	3	2	0	1	0	1	0	0	0	65
1	2	15	0	0	3	0	0	0	0	0	75
2	0	0	15	5	0	0	0	0	0	0	75
3	0	0	5	10	0	5	0	0	0	0	50
4	1	0	0	0	19	0	0	0	0	0	95
5	0	0	0	14	0	6	0	0	0	0	30
6	0	0	0	0	0	0	20	0	0	0	100
7	0	15	0	0	4	0	1	0	0	0	0
8	2	0	5	10	0	3	0	0	0	0	0
9	0	0	0	20	0	0	0	0	0	0	0

ตารางที่ 4.1 แสดงการจดจำเสียง 0 ถึง 9 โดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เสียงที่ต้องการทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	13	6	0	0	1	0	0	0	0	0	65
1	0	16	0	0	2	0	1	1	0	0	80
2	0	0	17	3	0	0	0	0	0	0	85
3	0	0	4	13	0	3	0	0	0	0	65
4	2	1	0	0	17	0	0	0	0	0	85
5	0	0	3	10	0	7	0	0	0	0	35
6	4	0	0	1	0	0	15	0	0	0	75
7	4	11	0	0	1	0	4	0	0	0	0
8	0	0	10	5	1	4	0	0	0	0	0
9	0	0	5	13	0	2	0	0	0	0	0

ตารางที่ 4.2 แสดงการจดจำเสียง 0 ถึง 9 โดยใช้เสียงจากบุคคลอื่น(unknown) กรณีที่ 1

กรณีที่ 2 แบบจำลองจากเสียงต้นแบบพูด 0 ถึง 9 เสียงละ 20 ครั้ง รวมเป็น 200 เสียง โดยใช้ codebook 64

เสียงที่ต้องการทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	18	0	2	0	0	0	0	0	0	0	90
1	2	11	0	0	1	0	0	6	0	0	55
2	0	0	15	1	0	0	0	0	2	2	75
3	0	0	0	11	0	6	0	0	2	1	55
4	1	0	0	0	19	0	0	0	0	0	95
5	0	0	0	2	1	15	0	0	2	0	75
6	0	0	1	0	0	0	19	0	0	0	95
7	0	1	0	0	1	0	0	18	0	0	90
8	0	0	0	0	0	0	0	0	19	1	95
9	0	0	0	0	0	0	0	0	0	20	100

ตารางที่ 4.3 แสดงการจดจำเสียง 0 ถึง 9 โดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เสียงที่ต้องการทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	15	1	2	0	0	0	0	1	!	0	75
1	2	13	0	0	3	0	1	1	0	0	65
2	0	0	14	2	0	0	0	0	0	4	70
3	0	0	7	7	0	2	0	0	1	3	35
4	2	0	0	0	18	0	0	0	0	0	90
5	0	0	1	3	1	7	0	0	2	6	35
6	0	2	1	0	0	0	17	0	0	0	85
7	1	1	0	0	1	0	2	15	0	0	75
8	0	0	3	0	0	0	0	0	16	1	80
9	0	0	2	1	0	1	0	0	1	15	75

ตารางที่ 4.4 แสดงการจดจำเสียง 0 ถึง 9 โดยใช้เสียงจากบุคคลอื่น(unknown) กรณีที่ 2

กรณีที่ 3 แบบจำลองจากเสียงต้นแบบพูด 0 ถึง 9 เสียงละ 20 ครั้ง รวมเป็น 200 เสียง โดยใช้ codebook 128

เสียงที่ต้องการทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	15	3	0	0	0	0	1	0	1	0	75
1	2	13	0	0	2	0	0	3	0	0	65
2	0	0	16	0	0	0	0	0	2	2	80
3	0	0	0	17	0	1	0	0	2	0	85
4	0	0	0	0	20	0	0	0	0	0	100
5	0	0	0	4	0	15	0	0	0	1	75
6	0	0	1	0	0	0	19	0	0	0	95
7	0	2	0	0	1	0	0	17	0	0	85
8	0	0	1	1	0	0	0	0	17	1	85
9	0	0	0	0	0	0	0	0	0	20	100

ตารางที่ 4.5 แสดงการจดจำเสียง 0 ถึง 9 โดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เสียงที่ต้องการทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	14	3	2	0	0	0	0	1	0	0	70
1	2	15	0	0	1	0	2	0	0	0	75
2	0	0	15	3	0	0	0	0	0	2	75
3	0	0	7	7	0	1	0	0	1	4	35
4	0	1	0	0	19	0	0	0	0	0	95
5	0	0	0	7	0	8	0	0	2	3	40
6	0	1	1	0	0	0	18	0	0	0	90
7	0	4	0	0	0	0	2	14	0	0	70
8	0	0	2	0	0	1	0	0	16	1	80
9	0	0	0	1	0	3	0	0	1	15	75

ตารางที่ 4.6 แสดงการจดจำเสียง 0 ถึง 9 โดยใช้เสียงจากบุคคลอื่น(unknown) กรณีที่ 3

<u>กรณีที่ 4</u> แบบจำลองจากเสียงต้นแบบพูด 0 ถึง 9 เสียงละ 20 ครั้ง รวมเป็น 200 เสียง โดยใช้ codebook 128 และมีการข้ามสเตทเกิดขึ้นในโมเดลเสียงไม่เกิน 1 สเตท

เสียงที่ต้องการทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	17	2	0	0	0	0	0	0	1	0	85
1	2	13	0	0	2	0	0	3	0	0	65
2	0	0	16	0	0	0	0	0	2	2	80
3	0	0	2	15	0	1	0	0	2	0	75
4	0	0	0	0	20	0	0	0	0	0	100
5	0	0	0	5	0	14	0	0	1	0	70
6	0	0	1	0	0	0	19	0	0	0	95
7	0	2	0	0	1	0	0	17	0	0	85
8	0	0	1	0	0	0	0	0	14	5	70
9	0	0	0	0	0	0	0	0	0	20	100

ตารางที่ 4.7 แสดงการจดจำเสียง 0 ถึง 9 โดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เสียงที่ต้องการทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	14	4	1	0	0	0	0	1	0	0	70
1	2	15	0	0	1	0	1	1	0	0	75
2	0	0	16	1	0	0	0	0	0	3	80
3	0	0	7	6	0	2	0	0	1	4	30
4	0	0	0	0	20	0	0	0	0	0	100
5	0	0	0	8	0	7	0	0	1	4	35
6	3	0	0	0	1	0	16	0	0	0	80
7	0	4	0	0	0	0	2	14	0	0	70
8	0	0	1	0	0	2	0	0	15	2	75
9	0	0	3	1	0	3	0	0	0	13	65

ตารางที่ 4.8 แสดงการจดจำเสียง 0 ถึง 9 โดยใช้เสียงจากบุคคลอื่น(unknown) กรณีที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 ผลการเปรียบเทียบความถูกต้องการทดสอบการรู้จำเสียงพูด

ผู้ทดสอบ	ต่ำสุด	สูงสุด	เฉลี่ย
• กรณีที่ 1 ใช้ codebook 32			
1.การทดสอบเสียงจากต้นแบบ	0%	100%	49%
2.การทดสอบเสียงจากบุคคลอื่น	0%	85%	49%
ผลโดยรวม			49%
• กรณีที่ 2 ใช้ codebook 64			
1.การทดสอบเสียงจากต้นแบบ	55%	100%	82.5%
2.การทดสอบเสียงจากบุคคลอื่น	35%	90%	68.5%
ผลโดยรวม			75.5%
• กรณีที่ 3 ใช้ codebook 128			
1.การทดสอบเสียงจากต้นแบบ	65%	100%	84.5%
2.การทดสอบเสียงจากบุคคลอื่น	35%	95%	70.5%
ผลโดยรวม			77.5%
• กรณีที่ 4 ใช้ codebook 128 ที่มีการข้ามสเตท			
1.การทดสอบเสียงจากต้นแบบ	65%	100%	82.5%
2.การทดสอบเสียงจากบุคคลอื่น	30%	100%	68%
ผลโดยรวม			75.25%

ตารางที่ 4.9 แสดงผลค่าความถูกต้องทุกกรณี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และบทสรุป

5.1 สรุปผลการทดลอง

หลังจากที่ได้เสียงมาเพื่อทำการวิเคราะห์ โดยใช้วิธีการประมาณเชิงเส้น (Linear Predictive Coding : LPC) จะเป็นการวิเคราะห์พารามิเตอร์ของเสียง และสามารถลดขนาดข้อมูลได้อย่างมี ประสิทธิภาพ และในการประมาณเชิงเส้น เราจะเลือกใช้วิธีออโตคอร์รีเลชั่น เพราะเป็นวิธีที่มีความน่าเชื่อถือที่สูง

จากการทดลองระบบการจดจำของเสียง 0-9 โดยการใช้สัญญาณเสียงที่สุ่มด้วยความถี่ 8 kHz ผลการทดลองสรุปได้ดังนี้

1. เสียงพูดของคนๆ เดียวกัน พูดคนละครั้งกัน พบว่าเสียงที่พูดออกมาไม่เหมือนกัน
2. การหาขอบเขตของคำ มีผลต่อความถูกต้องของการจดจำเสียง การหาขอบเขตของคำที่ถูกต้องแน่นอน จะเพิ่มอัตราการจดจำเสียงให้สูงขึ้น
3. เนื่องจากสัญญาณที่ผ่านการหาขอบเขตของคำ มีความยาวของเสียงที่ต่างกัน ทำให้การหาค่าเฉลี่ยของสัญญาณที่นำมาสร้างแบบอ้างอิงทำได้ไม่แน่นอน เพื่อความถูกต้องของการสร้างแบบอ้างอิงจึงทำการปรับสัญญาณก่อนที่จะมาทำการหาค่าพารามิเตอร์ แล้วทำการนอร์มอลไลซ์ ให้มีความยาวเท่ากันเสียก่อน
4. การหาค่าพารามิเตอร์แบ่งๆคือ หาค่าพรามิเตอร์ แอลพีซี, เซปตรัม, การถ่วงน้ำหนักเซปตรัม, หาค่าปรากฏ ,สร้างโมเดลเสียง ซึ่งเรียกว่าการจดจำเสียงพูดแบบ HMM/VQ
5. ความถูกต้อง แม่นยำในการจดจำเสียง ขึ้นอยู่กับจำนวนสียงที่นำมาสร้างโมเดลอ้างอิง
6. จากวิธีการ ฮิดเดนมาร์คอฟ นั้นเมื่อมีค่าความน่าจะเป็นของค่าปรากฏที่ออกมาเป็น ศูนย์จะทำให้ค่า ความน่าจะเป็นรวมออกมาเป็น ศูนย์ ซึ่งทำให้การหาโมเดลเสียงผิดพลาด
7. การบันทึกเสียงเพื่อนำมาสร้างโมเดลอ้างอิง หรือเสียงที่ต้องการตรวจสอบ จะต้องบันทึกในสภาพห้องที่เงียบ มีเสียงรบกวนน้อย เพื่อให้ได้สัญญาณเสียงที่มีคุณภาพ
8. ขนาดของโค้ดบุคมีผลต่อการหาค่าปรากฏที่เกิดขึ้น
9. การใช้เสียงจากผู้พูดจำนวนมาก และโค้ดบุคที่มีขนาดใหญ่ เพื่อนำมาสร้างโมเดลอ้างอิง จะทำให้มีฐานข้อมูลขนาดใหญ่ และที่สำคัญ ใช้เวลานานในการหาโมเดลเสียง
10. ความเร็วในการรู้จำเสียงดีกว่า การใช้วิธี Dynamic Time Warping มาก และจำนวนข้อมูลของแบบอ้างอิงมีน้อยกว่า

บทวิจารณ์ และ แนวทางพัฒนา

1. การทดสอบนี้ได้ทดสอบเฉพาะผู้ชายเท่านั้น เนื่องจากช่วงความถี่ ของเสียงผู้ชายและเสียงผู้หญิง จะมีพารามิเตอร์ที่ต่างกัน จึงควรพัฒนาต่อไปให้สามารถจดจำได้ ทั้งเสียงผู้ชายและเสียงผู้หญิงรวมถึงช่วงอายุของผู้ที่เราสร้างแบบอ้างอิงด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ในการหาพารามิเตอร์ของเสียงนั้นจะทำการหาในโดเมนเวลา จะทำให้วิธีการหามีความยุ่งยากน้อยกว่า การหาพารามิเตอร์ในโดเมนความถี่
3. วิธีการฮิดเดนมาร์คอฟนี้ยังสามารถนำไปใช้กับ การรู้จำคำที่ต่อเนื่องได้ โดยจะมีแนวทางและวิธีการที่ต้องพัฒนาต่อได้โดยใช้พื้นฐาน จากความเข้าใจในการรู้จำคำที่เป็นคำโดด
4. ควรนำไปพัฒนาต่อโดยใช้ Hardware เพื่อที่จะสามารถนำไปใช้ในทางปฏิบัติได้ ซึ่งปัจจุบันได้มี Hardware ที่มีการบีบอัดข้อมูลที่มีประสิทธิภาพสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการทำงานและโปรแกรมต่างๆ ที่จะใช้ในการจดจำเสียง แสดงดังต่อไปนี้

1)โปรแกรมการตัดคำและนอร์มอลไลซ์

- รูปที่ 1 แสดงขั้นตอนการทำงานของโปรแกรม detect
- รูปที่ 2 แสดงขั้นตอนการทำงานของโปรแกรม normalize
- ฟังก์ชัน detect : การเปรียบเทียบโดยการหาค่าพลังงาน
- ฟังก์ชัน normalize : การปรับค่า ของจำนวนตัวอย่างให้ได้ 4000 ค่า

2)โปรแกรมการหาค่าพารามิเตอร์

- รูปที่ 3 แสดงขั้นตอนการทำงานของโปรแกรม gsp
- ฟังก์ชัน gsp : การหาค่าสัมประสิทธิ์ LPC , สัมประสิทธิ์ Cepstrum , Weight Cepstrum

3)โปรแกรมสร้าง codebook

- รูปที่ 4 แสดงขั้นตอนการทำงานของโปรแกรม codebook32,64,128
- ฟังก์ชัน codebook32 : การหาโค้ดบุคที่เหมาะสม 32 ระดับ
- ฟังก์ชัน codebook64 : การหาโค้ดบุคที่เหมาะสม 64 ระดับ
- ฟังก์ชัน codebook128 : การหาโค้ดบุคที่เหมาะสม 128 ระดับ

4)โปรแกรมผลิต Outcome จาก ชื่อไฟล์.wav

- รูปที่ 5 แสดงขั้นตอนการทำงานของโปรแกรม Observation
- ฟังก์ชัน Observation : การผลิตค่าโค้ดบุคอินเด็กซ์เทียบจากโค้ดบุค

5)โปรแกรม การสร้างโมเดล HMM

- รูปที่ 6 แสดงขั้นตอนการทำงานของโปรแกรม hmm6
- ฟังก์ชัน hmm6 : การสร้างโมเดลของ HMM
- รูปที่ 7 แสดงขั้นตอนการทำงานของโปรแกรม reg5
- ฟังก์ชัน reg5 : เป็นการตัดสินใจ เพื่อ บอกผลการจดจำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 1 แสดงขั้นตอนการทำงานของโปรแกรม detect

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2 แสดงขั้นตอนการทำงานของโปรแกรม normalize

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชัน detect

```
function s=detect(ipt)
%++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+% Detect speech length
%  function s=detect(ipt)
%++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+
inpt=ipt-mean(ipt);

%number of frame
sample = 100;
vnoloop = floor(length(inpt)/sample);

%find energy each of frame
for k=1:vnoloop
   sum=0;
   for k1=1:sample
      sum=sum+abs(inpt((k-1)*sample+k1));
   end
   E(k)=sum;
end

% find maximum energy
Emax=max(E);

%set value
a=0.2;

%begin process
ref=a*Emax;
for k=1:vnoloop
   if E(k)==Emax
      fixenergy=k;
      break;
   end
end

for k=fixenergy:-1:1
   if E(k)<=ref
      a1=k;
      break
   end
end
for k1=fixenergy:vnoloop
   if E(k1)<=ref
      b1=k1;
      break
   end
end
num=0;
if isempty(b1)==1|b1==vnoloop
   b1=vnoloop-1;
   for i=(a1-1):b1
      for i1=1:sample
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            num=num+1;
            s(num)=inpt(i*sample+i1);
         end
      end
   else
      for i=(a1-1):(b1+1)
         for i1 = 1:sample
            num=num+1;
            s(num)=inpt(i*sample+i1);
         end
      end
   end
```

- ฟังก์ชัน normalize

```
function y=normalize(x)
%++++++++++++++++++++++++++++++++++++++++++++++++++
%  Normalize the number of y to size newsize
%  function y=normalize(x,newsize)
%++++++++++++++++++++++++++++++++++++++++++++++++++

newsize=4000;
y(1)=x(1);
step=(length(x)-1)/(newsize-1);
for i=2:newsize
   point=(1+(i-1)*step);
   fixlow=floor(point);
   fixhigh=ceil(point);
   if fixlow==fixhigh
      y(i)=x(point);
   else
      y(i)=x(fixlow)+(x(fixhigh)-x(fixlow))*(point-fixlow)/(fixhigh-
fixlow);
   end
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3 แสดงขั้นตอนการทำงานของโปรแกรม gsp

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

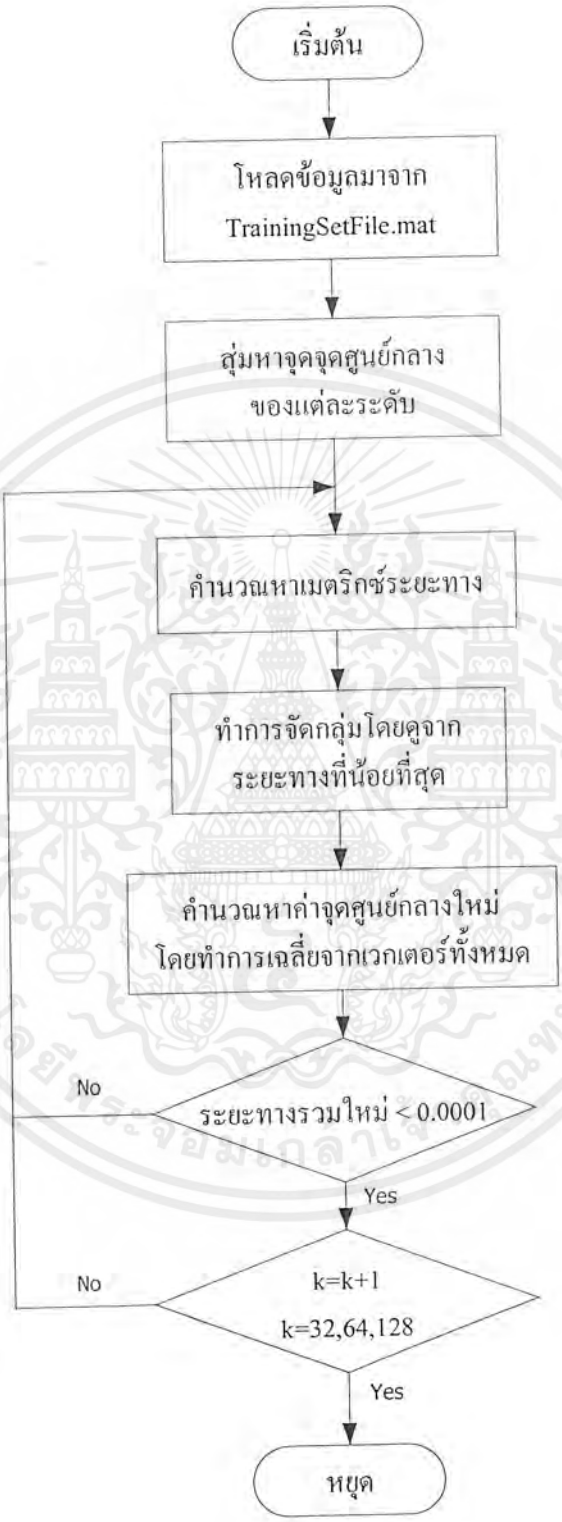
- ฟังก์ชัน gsp

```
function cw=gsp(ipt)
%++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
%  Generate speech parameter
%  function cw=gsp(z)
%++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
z=preemphasis(ipt);
%+++++++++++++++++
%  Frame blocking
%+++++++++++++++++
lpcorder=12;
Qfactor=18;
framelength=240;
frameshift=80;
framenumber=1;
frameoverlap=framelength-frameshift;
lastframe=floor(length(z)/frameshift)-ceil(frameoverlap/frameshift);
for framenumber=1:lastframe

   %++++++++++++
   %  Windowing
   %++++++++++++
   j=1;
   for i=(((framenumber-1)*frameshift)+1):(((framenumber-
1)*frameshift)+framelength)
      w(framenumber,j)=z(i)*(0.54-0.46*cos(2*pi*(j-1)/(framelength-1)));
      j=j+1;
   end

   %++++++++++++++++++++++++++
   %  Autocorrelation analysis
   %++++++++++++++++++++++++++
   for j=1:lpcorder+1
      r(framenumber,j)=0;
      for i=1:(framelength+1-j)
         r(framenumber,j)=r(framenumber,j)+w(framenumber,i)*w
(framenumber,(i-1+j));
      end
   end

   %+++++++++++++++
   %  LCP analysis
   %+++++++++++++++
   e(1)=r(framenumber,1);
   K(1)=r(framenumber,2)/e(1);
   A(1,1)=K(1);
   for i=2:lpcorder
      e(i)=(1-K(i-1)^2)*e(i-1);
      j=i;
      sumAr=0;
      for k=1:i-1
         sumAr=sumAr+A(i-1,k)*r(framenumber,j);
         j=j-1;
      end
      K(i)=(r(framenumber,i+1)-sumAr)/e(i);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        for j=i:-1:1
            if j==i
               A(i,j)=K(i);
            else
               A(i,j)=A(i-1,j)-K(i)*A(i-1,i-j);
            end
        end
    end
    gains(framenumber)=sqrt(e(lpcorder));
    for i=1:lpcorder
       a(framenumber,i)=A(lpcorder,i);
    end
    for i=lpcorder+1:Qfactor
       a(framenumber,i)=0;
    end

    %++++++++++++++++++++++++++++++++++++++++++++++++++
    %  LPC parameter conversion to cepstal coefficients
    %++++++++++++++++++++++++++++++++++++++++++++++++++
    c(framenumber,1)=a(framenumber,1);
    for i=2:Qfactor
       sumca=0;
       for k=1:i-1
          sumca=sumca+(k/i)*c(framenumber,k)*a(framenumber,i-k);
       end
       c(framenumber,i)=a(framenumber,i)+sumca;
    end

    %++++++++++++++++++++
    %  Parameter weighting
    %++++++++++++++++++++
    for i=1:Qfactor
       cweight(framenumber,i)=c(framenumber,i)*(1+(Qfactor/2)*sin
((pi*i)/Qfactor));
    end
    for k=Qfactor+1:-1:2
       cweight(framenumber,k)=cweight(framenumber,k-1);
    end
    cweight(framenumber,1)=2*log(gains(framenumber));
end

%++++++++++++++++++++++++++++++++++++++++
%  Converse parameter to 1 dimension matrix
%++++++++++++++++++++++++++++++++++++++++
cweight=cweight';
cw=cweight(:);


function y=preemphasis(x)
%++++++++++++++++++++++++++++++++
%  Preemphasis
%  function y=preemphasis(x)
%++++++++++++++++++++++++++++++++
y(1)=x(1);
for i=2:length(x)
   y(i)=x(i)-0.9375*x(i-1);
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4 แสดงขั้นตอนการทำงานของโปรแกรม codebook 32,64,128

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชัน codebook32

```
function cb32=codebook32(TrainingSetFile)
format long e;
VectorDimension = 19;
CodeBookNumber = 32;
RejectValue=0.0001;
VTD=2*RejectValue;

%============== Find Number of Frame ====================
NumberOfFrame = ((length(TrainingSetFile)/VectorDimension));

%==========  Random Centroid From TrainingSetFile ========

   x = floor(NumberOfFrame.*rand(1,CodeBookNumber))+1;
for nc=1:CodeBookNumber
   for nv=1:VectorDimension
      OldCentroid(((nc-1).*VectorDimension)+nv)=...
         TrainingSetFile(((x(nc)-1).*VectorDimension)+nv);
   end
end
k=0;

%======================== Loop ============================

while VTD>RejectValue

%================== Find Code Book ======================

for nf=1:NumberOfFrame
   for nc=1:CodeBookNumber
      TotalDistance=0;
      distance=0;
      for nv=1:VectorDimension
         distance=(TrainingSetFile(((nf-1).*VectorDimension)+nv)-...
            OldCentroid(((nc-1).*VectorDimension)+nv)).^2;
         TotalDistance =TotalDistance+distance;
      end
      VectorDistance(nc)=TotalDistance;
   end
   [MinDistance(nf),MinIndex(nf)]=min(VectorDistance);

end

%=============== Find New Centroid ================

for nc=1:CodeBookNumber
   Cnum=0;
   for nv=1:VectorDimension
   NewCentroid(((nc-1)*VectorDimension)+nv)=0;
   end
   for nf=1:NumberOfFrame
      if nc==MinIndex(nf)
         Cnum=Cnum+1;
         for nv=1:VectorDimension
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            NewCentroid(((nc-1)*VectorDimension)+nv)=...
               NewCentroid(((nc-1)*VectorDimension)+nv)+...
               TrainingSetFile(((nf-1)*VectorDimension)+nv);
         end
      end
   end

   if Cnum~=0
      for nv=1:VectorDimension
         NewCentroid(((nc-1)*VectorDimension)+nv) =...
            NewCentroid(((nc-1)*VectorDimension)+nv)./Cnum;
      end
   else
      for nv=1:VectorDimension
         NewCentroid(((nc-1)*VectorDimension)+nv)=...
            OldCentroid(((nc-1)*VectorDimension)+nv);
      end
   end
end

%============== Check Distance ============================

d=NewCentroid-OldCentroid;
e=d.^2;
VTD=sum(e)%./CodeBookNumber
k=k+1;
CVTD32(k)=VTD;

%===========Copy Old Centroie to New Centroid=============

OldCentroid=NewCentroid;

%=========================================================
end
cb32=NewCentroid;
save codebook32 cb32 CVTD32
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชัน codebook64

```
function cb64=codebook64(TrainingSetFile)
format long;
VectorDimension = 19;
CodeBookNumber = 64;
RejectValue=0.0001;
VTD=2*RejectValue;

%============== Find Number of Frame ====================

NumberOfFrame = ((length(TrainingSetFile)/VectorDimension));

%==========  Random Centroid From TrainingSetFile ========

   x = floor(NumberOfFrame.*rand(1,CodeBookNumber))+1;

for nc=1:CodeBookNumber
   for nv=1:VectorDimension
      OldCentroid(((nc-1).*VectorDimension)+nv)=...
         TrainingSetFile(((x(nc)-1).*VectorDimension)+nv);
   end
end

k=0;
%========================= Loop ============================

while VTD>RejectValue

%================== Find Code Book ========================

for nf=1:NumberOfFrame
   for nc=1:CodeBookNumber
      TotalDistance=0;
      distance=0;
      for nv=1:VectorDimension
         distance=(TrainingSetFile(((nf-1).*VectorDimension)+nv)-...
            OldCentroid(((nc-1).*VectorDimension)+nv)).^2;
         TotalDistance =TotalDistance+distance;
      end
      VectorDistance(nc)=TotalDistance;
   end

   [MinDistance(nf),MinIndex(nf)]=min(VectorDistance);

end

   %============== Find New Centroid ================

for nc=1:CodeBookNumber
   Cnum=0;
   for nv=1:VectorDimension
   NewCentroid(((nc-1)*VectorDimension)+nv)=0;
   end
   for nf=1:NumberOfFrame
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      if nc==MinIndex(nf)
         Cnum=Cnum+1;
         for nv=1:VectorDimension
            NewCentroid(((nc-1)*VectorDimension)+nv)=...
               NewCentroid(((nc-1)*VectorDimension)+nv)+...
               TrainingSetFile(((nf-1)*VectorDimension)+nv);
         end
      end
   end

   if Cnum~=0
      for nv=1:VectorDimension
         NewCentroid(((nc-1)*VectorDimension)+nv) =...
            NewCentroid(((nc-1)*VectorDimension)+nv)./Cnum;
      end
   else
      for nv=1:VectorDimension
         NewCentroid(((nc-1)*VectorDimension)+nv)=...
            OldCentroid(((nc-1)*VectorDimension)+nv);
      end
   end
end

%============== Check Distance ============================

d=NewCentroid-OldCentroid;
e=d.^2;
VTD=sum(e)%./CodeBookNumber
k=k+1;
CVTD64(k)=VTD;

%===========Copy Old Centroie to New Centroid=============

OldCentroid=NewCentroid;
end
cb64=NewCentroid;
save codebook64 cb64 CVTD64
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชัน codebook128

```
function cb128=codebook128(TrainingSetFile)
format long;
load TrainingSetFile
VectorDimension = 19;
CodeBookNumber = 128;
RejectValue=0.0001;
VTD=2*RejectValue;

%============== Find Number of Frame ====================

NumberOfFrame = ((length(TrainingSetFile)/VectorDimension));

%==========  Random Centroid From TrainingSetFile ========

   x = floor(NumberOfFrame.*rand(1,CodeBookNumber))+1;

for nc=1:CodeBookNumber
   for nv=1:VectorDimension
      OldCentroid(((nc-1).*VectorDimension)+nv)=...
         TrainingSetFile(((x(nc)-1).*VectorDimension)+nv);
   end
end

k=0;
%========================= Loop ============================

while VTD>RejectValue

%================== Find Code Book ======================

for nf=1:NumberOfFrame
   for nc=1:CodeBookNumber
      TotalDistance=0;
      distance=0;
      for nv=1:VectorDimension
         distance=(TrainingSetFile(((nf-1).*VectorDimension)+nv)-...
            OldCentroid(((nc-1).*VectorDimension)+nv)).^2;
         TotalDistance =TotalDistance+distance;
      end
      VectorDistance(nc)=TotalDistance;
   end

   [MinDistance(nf),MinIndex(nf)]=min(VectorDistance);

end

   %=============== Find New Centroid ================

for nc=1:CodeBookNumber
   Cnum=0;
   for nv=1:VectorDimension
   NewCentroid(((nc-1)*VectorDimension)+nv)=0;
   end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    for nf=1:NumberOfFrame
       if nc==MinIndex(nf)
          Cnum=Cnum+1;
          for nv=1:VectorDimension
             NewCentroid(((nc-1)*VectorDimension)+nv)=...
                NewCentroid(((nc-1)*VectorDimension)+nv)+...
                TrainingSetFile(((nf-1)*VectorDimension)+nv);
          end
       end
    end

    if Cnum~=0
       for nv=1:VectorDimension
          NewCentroid(((nc-1)*VectorDimension)+nv) =...
             NewCentroid(((nc-1)*VectorDimension)+nv)./Cnum;
       end
    else
       for nv=1:VectorDimension
          NewCentroid(((nc-1)*VectorDimension)+nv)=...
             OldCentroid(((nc-1)*VectorDimension)+nv);
       end
    end
end

%============== Check Distance ============================

d=NewCentroid-OldCentroid;
e=d.^2;
VTD=sum(e)%./CodeBookNumber
k=k+1;
CVTD128(k)=VTD;

%===========Copy Old Centroie to New Centroid============

OldCentroid=NewCentroid;
end
cb128=NewCentroid;
save codebook128 cb128 CVTD128
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

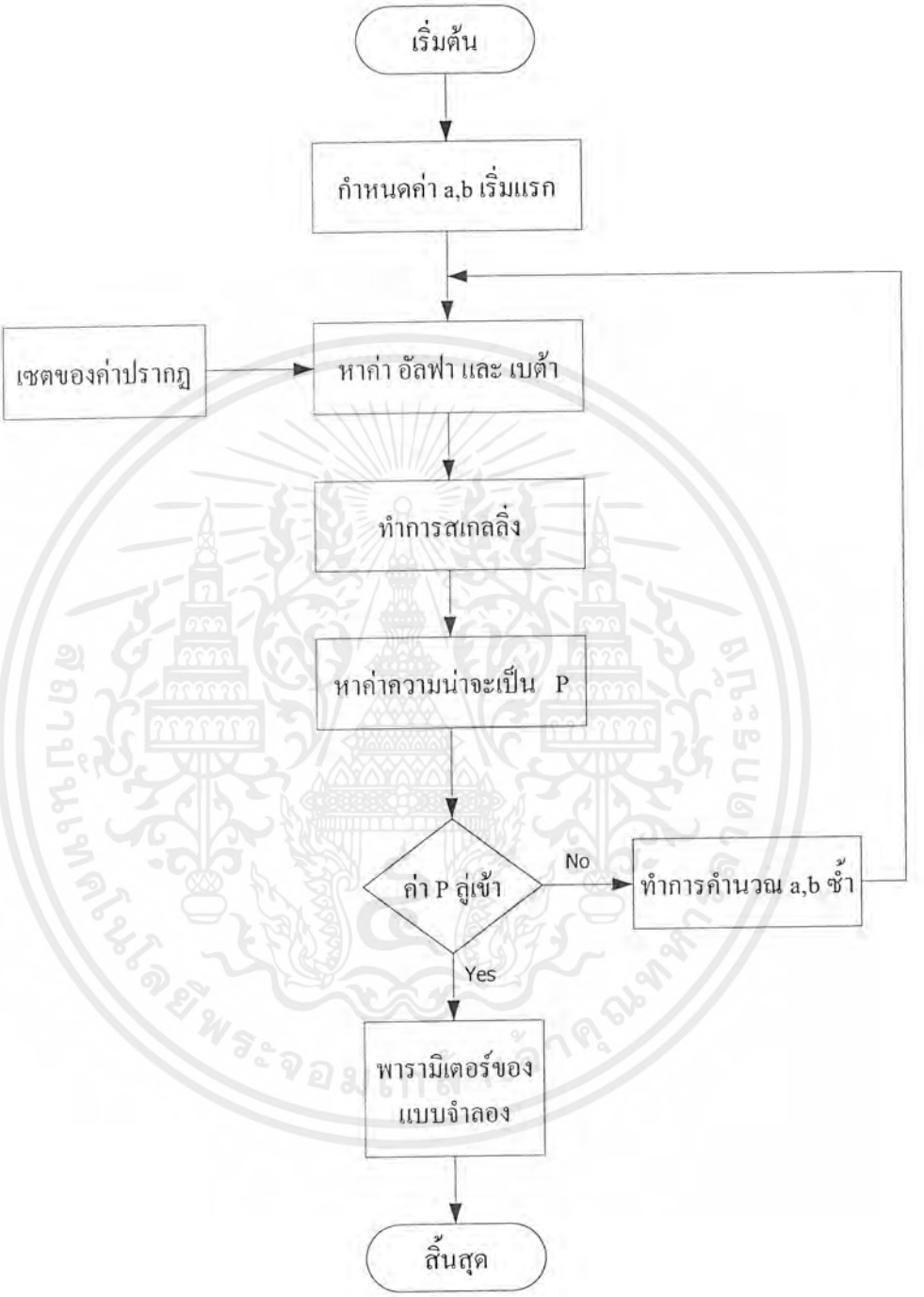
รูปที่ 5 แสดงขั้นตอนการทำงานของโปรแกรม Observation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชัน Observation

```
function Ob=Observation(weight,codebook,Number)

format long;
VectorDimension = 19;
CodeBookNumber=Number;
NumberOfFrame = (length(weight)/VectorDimension);

UnknowFile=weight;

%============ Compare Distance ===============

for nf=1:NumberOfFrame
   for nc=1:CodeBookNumber
      TD=0;
      dis=0;
      for nv=1:VectorDimension
         dis=(codebook(((nc-1).*VectorDimension)+nv)-...
            UnknowFile(((nf-1).*VectorDimension)+nv)).^2;
         TD=TD+dis;
      end
      VD(nc)=TD;
   end


   [MinDis(nf),Ob(nf)]=min(VD);
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6 แสดงขั้นตอนการทำงานของโปรแกรม hmm6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชัน hmm6

```
format long e
%Initial model parameter
N=6;
M=64;
V=1:M;
[K,T]=size(O);
%Defind TT
TT=[1 0 0 0 0 0];
%Defind matrix A
A=[0.5 0.5 0 0 0 0
   0 0.5 0.5 0 0 0
   0 0 0.5 0.5 0 0
   0 0 0 0.5 0.5 0
   0 0 0 0 0.5 0.5
   0 0 0 0 0 1];
%Defind  matrix B
for i=1:N
   for k=1:M
      B(i,k)=1/M;
   end
end
%Start HMM
%Iterative for new A and B
for round=1:50
   for n=1:K
      for t=1:T
         for i=1:N
            for k=1:M
               if O(n,t)==V(k)
                  b(n,i,t)=B(i,k);
               end
            end
         end
      end
      %The forward procedure
      for i=1:N
         alpha(n,1,i)=TT(i)*b(n,i,1);
      end
      for t=1:T-1
         for j=1:N
            sumaa=alpha(n,t,1)*A(1,j);
            for i=2:N
               sumaa=sumaa+alpha(n,t,i)*A(i,j);
            end
            alpha(n,t+1,j)=sumaa*b(n,j,t+1);
         end
      end
      P(n)=sum(alpha(n,T,1:N));
      %The backward procedure
      for i=1:N
         beta(n,T,i)=1;
      end
      for t=T-1:-1:1
         for i=1:N
```

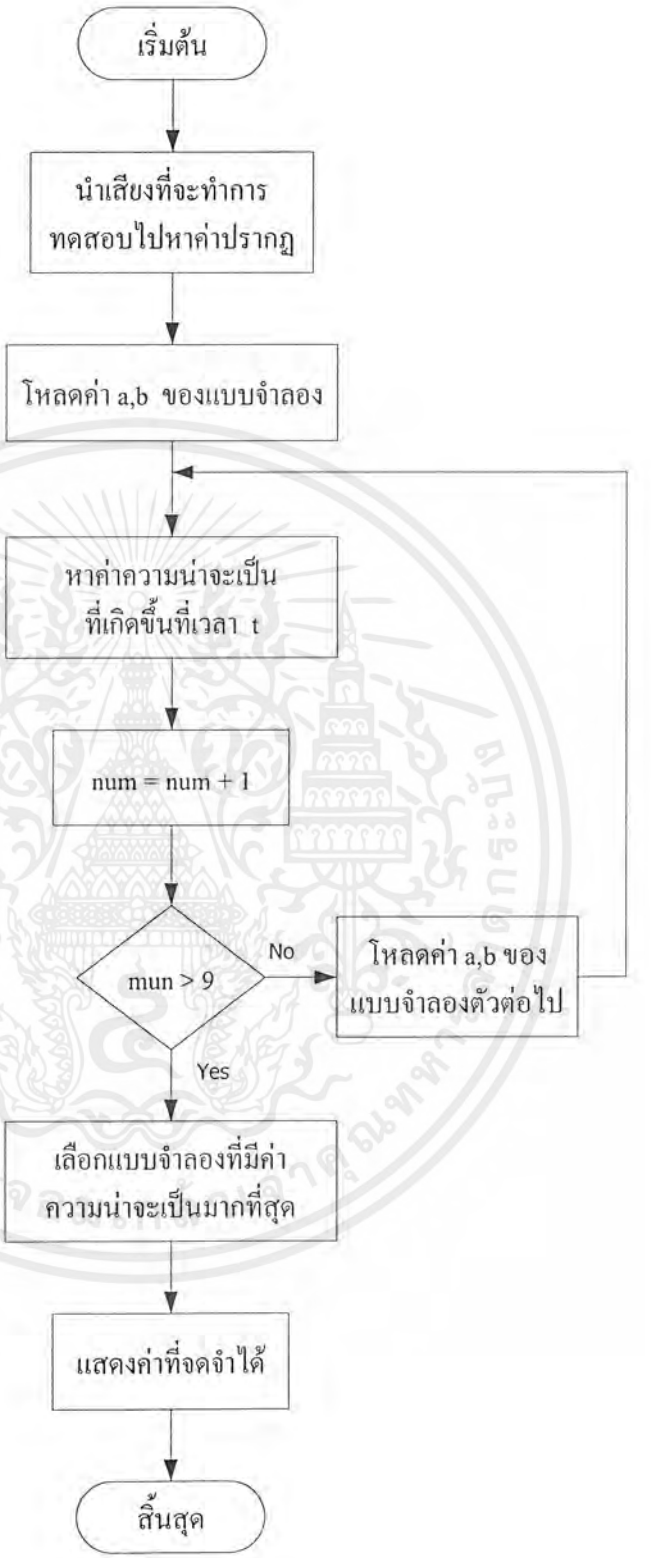
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            sumab=A(i,1)*b(n,1,t+1)*beta(n,t+1,1);
            for j=2:N
               sumab=sumab+A(i,j)*b(n,j,t+1)*beta(n,t+1,j);
            end
            beta(n,t,i)=sumab;
         end
      end
   end
   %Parameter estimation
   %Find matrix A
   for i=1:N
      for j=1:N
         nom=0;
         for n=1:K
            kvt=0;
            for t=1:T-1
               kvt=kvt+alpha(n,t,i)*A(i,j)*b(n,j,t+1)*beta(n,t+1,j);
            end
            nom=nom+kvt/P(n);
         end
         denom=0;
         for n=1:K
            kvt=0;
            for t=1:T-1
               for k=1:N
                  kvt=kvt+alpha(n,t,i)*A(i,k)*b(n,k,t+1)*beta(n,t+1,k);
               end
            end
            denom=denom+kvt/P(n);
         end
         A(i,j)=nom/denom;
      end
   end
   %Find matrix B
   for i=1:N
      for k=1:M
         nom=0;
         for n=1:K
            kvt=0;
            for t=1:T
               if O(n,t)==V(k)
                  ptk=1;
               else
                  ptk=0;
               end
               kvt=kvt+alpha(n,t,i)*beta(n,t,i)*ptk;
            end
            nom=nom+kvt/P(n);
         end
         denom=0;
         for n=1:K
            kvt=0;
            for t=1:T
               kvt=kvt+alpha(n,t,i)*beta(n,t,i);
            end
            denom=denom+kvt/P(n);
         end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            B(i,k)=nom/denom;
        end
    end
end
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7 แสดงขั้นตอนการทำงานของโปรแกรม reg5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชัน reg5

```
function result=reg5(filename)
format long e
load codebook64
[x fs b]=wavread(filename);
y=detect(x);
z=normalize(y);
w=gsp(z);
O=Observation(w,cb64);
%Initial model parameter
N=6;
M=64;
V=1:M;
T=length(O);
%Defind TT
TT(1)=1;
for i=2:N
   TT(i)=0;
end
for num=0:9
   switch num
   case 0
      load model00;
   case 1
      load model11;
   case 2
      load model22;
   case 3
      load model33;
   case 4
      load model44;
   case 5
      load model55;
   case 6
      load model66;
   case 7
      load model77;
   case 8
      load model88;
   case 9
      load model99;
   end
   for t=1:T
      for i=1:N
         for k=1:M
            if O(t)==V(k)
               b(i,t)=B(i,k);
            end
         end
      end
   end
   %The viterbi algorithm
   for i=1:N
      delta(1,i)=TT(i)*b(i,1);
      phi(1,i)=0;
   end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    for t=2:T
       for j=1:N
          damax=delta(t-1,1)*A(1,j)*b(j,t);
          dastatemax=1;
          for i=2:N
             if delta(t-1,i)*A(i,j)*b(j,t)>=damax
                damax=delta(t-1,i)*A(i,j)*b(j,t);
                dastatemax=i;
             end
          end
          delta(t,j)=damax;
          phi(t,j)=dastatemax;
       end
    end
    deltamax=delta(T,1);
    deltastatemax=1;
    for i=2:N
       if delta(T,i)>=deltamax
          deltamax=delta(T,i);
          deltastatemax=i;
       end
    end
    qsequence(T)=deltastatemax;
    for t=T-1:-1:1
       qsequence(t)=phi(t+1,(qsequence(t+1)));
    end
    %Total probability of this path
    pathprob(num+1)=TT(qsequence(1))*b(qsequence(1),1);
    for t=2:T
       pathprob(num+1)=pathprob(num+1)*A(qsequence(t-1),qsequence(t))*b
(qsequence(t),t);
    end
end
probmax=pathprob(1);
numprobmax=0;
for i=1:9
   if pathprob(i+1)>=probmax
      probmax=pathprob(i+1);
      numprobmax=i;
   end
end
result=numprobmax;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ชิ้นนี้ สำเร็จลุล่วงได้ด้วยดี ก็ด้วยความช่วยเหลือจากบุคคลหลายๆ ท่านในด้านความรู้ เครื่องมืออุปกรณ์ต่างๆ ตลอดจนการให้คำแนะนำ อีกทั้งยังคอยเป็นกำลังใจในโครงงานนี้ ซึ่งต้องขอกล่าวถึงบุคคลเหล่านี้

บิดา มารดา ที่คอยเป็นกำลังใจในการทำโครงงานชิ้นนี้สำเร็จ

อาจารย์กฤษณ์ วงรุจิระ และ ดร.สุทธิชัย นพนาคีพงษ์ อาจารย์ที่ปรึกษาโครงงาน ที่คอยให้ทั้ง คำแนะนำ คำปรึกษา และช่วยเหลือง่านวยความสะดวกในการใช้เครื่องมือ และ อุปกรณ์

คุณปวีณา จุฑาจรัส,คุณกิตติชัย สวัสดิ์อรุณวงศ์,คุณอรุณี ทวีอมรรัตนา และ คุณฐิติวรรณ ปัญญาพุฒิกุล ที่คอยช่วยเหลือด้านคอมพิวเตอร์ การขนส่ง ตรวจเอกสาร

และขอขอบคุณบุคคลต่างๆ อีกหลายๆ คนที่ไม่สามารถกล่าว ณ. ที่นี้ได้หมด และรวมถึงเพื่อนๆ ทุกคน ที่ให้การช่วยเหลือในการให้ตัวอย่างเสียง และด้านอื่นๆ

คณะผู้จัดทำ

27 มีนาคม 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1]. Lawrene Rabiner and Biing-Hwang Jung , "Fundamental of Speech Recognition ,New Jersey : Prentic Hall, 1993.

[2]. Duane Hanselman and Bruce Littlefield , "Mastering MATLAB 5" , Maine :Prentice Hall ,1998.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้