

การรู้จำเสียงพูดภาษาไทย
THAI SPEECH RECOGNITION



โดย
นายยอดอรณพ จันทร
นายวศิน โพโสดี
นายวิชัย สมอ้าย

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เลขหมู่.....
เลขทะเบียน... 46482
วัน, เดือน, ปี - 2 เม.ย. 2546

b.....
i.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรู้จำเสียงพูดภาษาไทย
THAI SPEECH RECOGNITION



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2544

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การรู้จำเสียงพูดภาษาไทย

THAI SPEECH RECOGNITION

ผู้จัดทำ

1. นายยอดอรุณพ จันทร 41014343
2. นายวศิน โพโสดี 41014377
3. นายวิชัย สมอ้าย 41014383

(รศ.ดร. ยูทธพงษ์ รังสรรค์เสรี)

อาจารย์ที่ปรึกษา

การรู้จำเสียงพูดภาษาไทย
Thai Speech Recognition

โดย นายยอดธรรมพ จันทร 41014343

นายวสิน โปโสลี 41014377

นายวิชัย สมอ้าย 41014383

อาจารย์ที่ปรึกษา รศ.ดร.ยุทธพงษ์ รังสรรค์เสรี

บทคัดย่อ

ปริญญานิพนธ์นี้เป็นการศึกษาการรู้จำเสียงพูด โดยทำการวิเคราะห์สัญญาณเสียง ซึ่งขั้นตอนแรกต้องทำการหาขอบเขตของคำและเลือกมาเฉพาะส่วนที่พิจารณา จากนั้นนำสัญญาณดังกล่าวไปวิเคราะห์หาพารามิเตอร์โดยใช้วิธีการเชิงเส้น (Linear Predictive Coding : LPC) พารามิเตอร์ที่ได้จะถูกปรับปรุงให้เหมาะสมเพื่อนำไปใช้กับการเปรียบเทียบ โดยวิธีฮิดเดนมาร์คอฟ (Hidden Markov Model) และผ่านขั้นตอนการตัดสินใจ เพื่อจำแนกสัญญาณเสียงเป็นขั้นตอนสุดท้าย

ABSTRACT

This project is study of the "Speech Recognition" by analyze speech signal. First, we find bound of word, and select section that we consider. Then take the last result to analyze the parameters by LPC method. After that improve the parameters use to compare by Hidden Markov Model method, and pass decision rule. Finally, selecting word is shown.

สารบัญ

	หน้า
บทคัดย่อ.....	I
สารบัญ.....	II
สารบัญตาราง.....	IV
สารบัญรูป.....	V
บทที่ 1 บทนำ.....	1
บทที่ 2 ทฤษฎีหรือหลักการ.....	2
2.1 ทฤษฎีการสร้างเสียงพูด.....	2
2.1.1 อวัยวะที่ใช้ในการออกเสียงพูด.....	2
2.1.2 การเกิดของเสียง.....	3
2.2 ทฤษฎีการรู้จำเสียง.....	4
2.2.1 การเตรียมสัญญาณในการวิเคราะห์.....	4
2.2.2 การวิเคราะห์เสียงเบื้องต้น.....	6
2.2.3 การจัดระดับเวกเตอร์.....	12
2.2.4 การสร้างแบบจำลอง.....	16
บทที่ 3 การคำนวณและการสร้าง.....	21
3.1 ภาคการเตรียมสัญญาณในการวิเคราะห์.....	21
3.1.1 การวิเคราะห์หาค่าพลังงานของสัญญาณ.....	21
3.1.2 การวิเคราะห์หาค่าอัตราการตัดศูนย์.....	24
3.1.3 การนอร์มอลไลซ์.....	26
3.2 ภาคการวิเคราะห์สัญญาณเบื้องต้น.....	27
3.2.1 การหาค่าพารามิเตอร์.....	27
3.2.2 การสร้างไค้คบุค.....	30
3.3 ภาครูปแบบอ้างอิง.....	33
3.4 ภาคการเปรียบเทียบ.....	33
3.4.1 ฮิดเดนมาร์คอฟโมเดล.....	33
3.5 ภาคการตัดสินใจ.....	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
บทที่ 4 การทดลองและผลการทดลอง	36
4.1 การทดลอง.....	36
4.2 ขั้นตอนการเรียนรู้.....	58
4.3 ขั้นตอนการวิเคราะห์.....	58
4.4 การทดสอบการจดจำเสียงพูดในกรณีต่าง ๆ	58
4.5 ผลการเปรียบเทียบความถูกต้องของการทดสอบการรู้จำเสียงพูด	65
บทที่ 5 บทวิจารณ์และบทสรุป	66
เอกสารอ้างอิง	67
ภาคผนวก ก การใช้งานโปรแกรม	68
ภาคผนวก ข โปรแกรม.....	78



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
4.1 แสดงการจดจำเสียง 0-9 ของผู้ชายโดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 1	59
4.2 แสดงการจดจำเสียง 0-9 ของผู้ชายโดยใช้เสียงจากบุคคลอื่น (unknown) กรณีที่ 1	59
4.3 แสดงการจดจำเสียง 0-9 ของผู้หญิงโดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 2.....	60
4.4 แสดงการจดจำเสียง 0-9 ของผู้หญิงโดยใช้เสียงจากบุคคลอื่น (unknown) กรณีที่ 2.....	60
4.5 แสดงการจดจำเสียง 0-9 ของผู้ชายโดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 3	61
4.6 แสดงการจดจำเสียง 0-9 ของผู้ชายโดยใช้เสียงจากบุคคลอื่น (unknown) กรณีที่ 3	61
4.7 แสดงการจดจำเสียง 0-9 ของผู้หญิงโดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 4.....	62
4.8 แสดงการจดจำเสียง 0-9 ของผู้หญิงโดยใช้เสียงจากบุคคลอื่น (unknown) กรณีที่ 4.....	62
4.9 แสดงการจดจำเสียง 0-9 ของผู้ชายโดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 5	63
4.10 แสดงการจดจำเสียง 0-9 ของผู้ชายโดยใช้เสียงจากบุคคลอื่น (unknown) กรณีที่ 5	63
4.11 แสดงการจดจำเสียง 0-9 ของผู้หญิงโดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 6.....	64
4.12 แสดงการจดจำเสียง 0-9 ของผู้หญิงโดยใช้เสียงจากบุคคลอื่น (unknown) กรณีที่ 6.....	64
4.13 แสดงผลค่าความถูกต้องทุกกรณี.....	65

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 บล็อกไดอะแกรมของการรู้จำเสียง.....	4
2.2 บล็อกไดอะแกรมของการเตรียมสัญญาณเสียงในการวิเคราะห์.....	4
2.3 บล็อกไดอะแกรมของการวิเคราะห์เสียงเบื้องต้น.....	6
2.4 การแบ่งช่องของสัญญาณ.....	7
2.5 ขั้นตอนในการวิเคราะห์หาคุณลักษณะของเสียง.....	8
2.6 บล็อกไดอะแกรมแสดงโมเดลการสร้างสัญญาณเสียงพูดอย่างง่าย.....	8
2.7 การกระจายเฟรมของเสียงพูดแต่ละจุดแทนเฟรมของเสียง.....	13
2.8 การรวมกลุ่มของเฟรมเสียงเพื่อสร้างโค้ด X.....	13
2.9 บล็อกไดอะแกรมของเวกเตอร์ควอนไทซ์.....	14
2.10 ขั้นตอนของเวกเตอร์ควอนไทซ์เซชัน.....	15
3.1 บล็อกไดอะแกรมการจำแนกเสียงพูดโดยทั่วไป.....	21
3.2 ฟังก์ชันการหาจุดเริ่มต้นและจุดสุดท้ายของสัญญาณเสียงโดยการเปรียบเทียบพลังงาน.....	23
3.3 ฟังก์ชันการหาจุดเริ่มต้นและจุดสุดท้ายของสัญญาณเสียงโดยการหาอัตราการตัดศูนย์.....	25
3.4 ขั้นตอนการนอร์มอลไลซ์สัญญาณ.....	27
3.5 ส่วนสัญญาณที่ตัดมาวิเคราะห์.....	28
3.6 บล็อกไดอะแกรมการวิเคราะห์เสียงเบื้องต้น.....	29
3.7 ขั้นตอนการสร้างโค้ดบุค.....	31
3.8 ขั้นตอนการควอนไทซ์เซชัน.....	32
3.9 ขั้นตอนการสร้างแบบจำลองอ้างอิง.....	34
3.10 ขั้นตอนการรู้จำเสียงพูด.....	35
4.1 แสดงการหาขอบเขตของค่าและการนอร์มอลไลซ์ของเลขศูนย์.....	37
4.2 แสดงการหาขอบเขตของค่าและการนอร์มอลไลซ์ของเลขหนึ่ง.....	38
4.3 แสดงการหาขอบเขตของค่าและการนอร์มอลไลซ์ของเลขสอง.....	39
4.4 แสดงการหาขอบเขตของค่าและการนอร์มอลไลซ์ของเลขสาม.....	40
4.5 แสดงการหาขอบเขตของค่าและการนอร์มอลไลซ์ของเลขสี่.....	41
4.6 แสดงการหาขอบเขตของค่าและการนอร์มอลไลซ์ของเลขห้า.....	42
4.7 แสดงการหาขอบเขตของค่าและการนอร์มอลไลซ์ของเลขหก.....	43
4.8 แสดงการหาขอบเขตของค่าและการนอร์มอลไลซ์ของเลขเจ็ด.....	44
4.9 แสดงการหาขอบเขตของค่าและการนอร์มอลไลซ์ของเลขแปด.....	45
4.10 แสดงการหาขอบเขตของค่าและการนอร์มอลไลซ์ของเลขเก้า.....	46

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์เพื่อการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์อื่นใด

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.11 แสดงค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงศูนย์.....	47
4.12 แสดงค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงหนึ่ง.....	47
4.13 แสดงค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงสอง.....	48
4.14 แสดงค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงสาม.....	48
4.15 แสดงค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงสี่.....	49
4.16 แสดงค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงห้า.....	49
4.17 แสดงค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงหก.....	50
4.18 แสดงค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงเจ็ด.....	50
4.19 แสดงค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงแปด.....	51
4.20 แสดงค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงเก้า.....	51
4.21 แสดงโค้ดบุค 32 และจำนวนรอบของเสียงผู้ชายจากการคำนวณทั้งหมด 138 รอบ.....	52
4.22 แสดงโค้ดบุค 32 และจำนวนรอบของเสียงผู้หญิงจากการคำนวณทั้งหมด 87 รอบ.....	53
4.23 แสดงโค้ดบุค 64 และจำนวนรอบของเสียงผู้ชายจากการคำนวณทั้งหมด 141 รอบ.....	54
4.24 แสดงโค้ดบุค 64 และจำนวนรอบของเสียงผู้หญิงจากการคำนวณทั้งหมด 90 รอบ.....	55
4.25 แสดงโค้ดบุค 128 และจำนวนรอบของเสียงผู้ชายจากการคำนวณทั้งหมด 74 รอบ.....	56
4.26 แสดงโค้ดบุค 128 และจำนวนรอบของเสียงผู้หญิงจากการคำนวณทั้งหมด 56 รอบ.....	57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

โดยปกติแล้วการติดต่อสื่อสารระหว่างมนุษย์กับเครื่องคอมพิวเตอร์ทำได้โดยการป้อนคำสั่งผ่านทางคีย์บอร์ดและเมาส์ ในขณะที่มีความต้องการหาวิธีการอื่นที่เป็นไปอย่างสะดวกและเป็นธรรมชาติมากกว่า จึงมีการพัฒนาให้คอมพิวเตอร์สามารถรับรู้คำสั่งจากเสียงพูดของมนุษย์ได้ ซึ่งจะทำให้การติดต่อสื่อสารระหว่างมนุษย์และคอมพิวเตอร์เป็นไปอย่างสะดวกและง่ายขึ้น

สมัยก่อนการใช้คอมพิวเตอร์เพื่อวิเคราะห์เสียงพูดของมนุษย์และสามารถรับรู้เสียงพูดได้นั้นทำได้ยาก และใช้เวลานาน เนื่องจากการพูดของมนุษย์ มีความซับซ้อนและมีความแตกต่างในแต่ละบุคคลทำให้การพัฒนาเป็นไปอย่างล่าช้า แต่ในปัจจุบันได้มีการพัฒนาวิธีการต่างๆ ให้มีความสามารถในการที่จะหาตัวแทนของเสียงในรูปแบบต่างๆ รวมทั้งมีวิธีการเพิ่มมากขึ้น และมีความแม่นยำในการรับรู้เสียงที่ป้อนให้แก่คอมพิวเตอร์ ด้วยพื้นฐานความรู้ทางด้านคณิตศาสตร์เชิงเส้นที่สำคัญและพื้นฐานความรู้ทางด้านสถิติ จึงทำให้เราลดความยุ่งยากลงได้มาก

ปริญญาตินพนธ์นี้ได้มีการศึกษาและทดลองเอาส่วนของอัลกอริทึม (Algorithm) การรู้จำเสียงที่เป็นทฤษฎีนำมาใช้งานจริงบนเครื่องคอมพิวเตอร์ส่วนบุคคล โดยการบันทึกเสียงที่จะทำการทดลอง โดยใช้อุปกรณ์ได้แก่ ไมโครโฟน การ์ดเสียงและลำโพง รูปแบบของเสียงที่บันทึกจะอยู่ในรูปแบบที่เหมาะสม โดยจะจำแนกเสียงตัวเลข 0-9 ซึ่งใช้เฉพาะเสียงที่มีเสียงภาคกลางเท่านั้นและมีช่วงความถี่อยู่ระหว่าง 300-3400 เฮิรตซ์ และสุ่มโดยใช้ความถี่ 22050 เฮิรตซ์ (จะบันทึกเสียงให้อยู่ในรูปแบบชื่อไฟล์.wav) จากนั้นนำเสียงที่ได้ไปทำการตัดค่าเพื่อให้เหลือค่าที่เหมาะสมแก่การวิเคราะห์และนำค่าที่ได้ไปปรับแต่งค่าแซมปลิ่ง (Sampling) ของแต่ละค่าให้เท่ากันโดยวิธีการนอร์มอลไลซ์ (Normalize) และเพื่อให้ค่า ๆ นั้นมีค่าพารามิเตอร์ในการวิเคราะห์น้อยลง เราจะทำกาแทนค่าของสัญญาณเสียงในแต่ละช่วงด้วยพารามิเตอร์แอลพีซี (LPC parameter) เพื่อให้ค่าที่ได้มีเสถียรภาพที่ดีขึ้นโดยเรานำไปปรับปรุงอีกขั้นหนึ่ง และจะใช้หลักการของ มินิมัม ดิสเทนซ์ คลาสซิฟิเคชัน (Minimum Distance Classification) ซึ่งสามารถนำไปใช้ในการรู้จำเสียง เพื่อทราบว่าเสียงที่เราป้อนให้แก่เครื่องคอมพิวเตอร์นั้นเป็นเสียงอะไร จะเห็นได้ว่าถ้าเราเลือกวิธีการทางทฤษฎีที่เหมาะสมแล้วจะทำให้ เวลาในการคำนวณหาค่าต่างๆรวมไปถึงการรู้จำเสียงใช้เวลาที่น้อยลงตามไปด้วย

บทที่ 2

ทฤษฎีหรือหลักการ

2.1 ทฤษฎีการสร้างเสียงพูด

การพูดของมนุษย์มีใช้อาหารที่เกิดเฉพาะที่ปากเท่านั้น หากเริ่มจากลมหายใจเข้าของมนุษย์ที่นำลมเข้าสู่ปอด จากนั้นจะใช้ลมจากปอดซึ่งก็คือลมหายใจออก มาทำให้เกิดเสียงพูดโดยลมจะถูกบังคับให้ผ่านอวัยวะต่าง ๆ ที่สำคัญ เราสามารถจำเสียงและวิธีการพูดของคน ๆ หนึ่งได้ เนื่องจากเมื่อพิจารณาเสียงที่พูดออกมาในแต่ละครั้งอาจจะสังเกตลักษณะที่แตกต่างกันได้ เสียงพูดที่จะอธิบายด้วยหลักเกณฑ์ทางวิทยาศาสตร์แม้ว่าในภาษาหนึ่ง ๆ จะมีเสียงต่างกันมากบ้างน้อยบ้าง แต่ละเสียงก็สามารถนำมาพิจารณาและอธิบายให้รู้ลักษณะการออกเสียงและตำแหน่งที่เกิดเสียงได้ คำอธิบายนี้จะทำให้เข้าใจลักษณะเสียงทุกเสียง วิชาที่ว่าด้วยเสียงพูดเรียกว่า วิชาสัทศาสตร์ (Phonetics)

2.1.1 อวัยวะที่ใช้ในการออกเสียงพูด

อวัยวะที่ใช้ในการออกเสียงมีอยู่หลายส่วน แต่ละส่วนสามารถทำให้เสียงพูดแตกต่างกันไปได้ อวัยวะที่ใช้ในการเปล่งเสียงพูดแบ่งออกเป็น 2 ประเภท คือ

ก. อวัยวะที่ใช้ในการทำอาหาร (Articulator) คืออวัยวะที่เคลื่อนไหวเพื่อผลัดลมไปยังส่วนต่าง ๆ อวัยวะที่สำคัญ คือ ลิ้น ซึ่งเป็นส่วนที่เคลื่อนไหวได้มากที่สุด

ข. อวัยวะซึ่งเป็นตำแหน่งที่เกิดเสียงต่าง ๆ (Point Of Articulator) คือตำแหน่งที่เกิดเสียงต่าง ๆ เช่น ริมฝีปาก ฟัน เพดาน ส่วนต่าง ๆ เป็นต้น

อวัยวะส่วนที่มีหน้าที่ในการออกเสียงโดยตรงมีดังนี้

1. ริมฝีปาก เป็นอวัยวะส่วนที่เคลื่อนไหวได้มากและทำให้เสียงแตกต่างกันได้มาก เราอาจบังคับปากอยู่ชิด กัน ห่างกัน ยื่นออกมา หรือห่อกลม เป็นต้น ลักษณะริมฝีปากแบบต่าง ๆ นี้ล้วนมีอิทธิพลต่อการออกเสียงและทำให้เสียงแตกต่างกันไปทั้งสิ้น

2. ฟัน เป็นอวัยวะที่ทำให้เกิดเสียงหลายชนิด เช่น เมื่อฟันกดลงบนริมฝีปากล่าง หรือฟันล่าง ลมที่ผ่านออกมาโดยแรงจะลอดช่องที่พอดผ่านได้ออกมา ทำให้เกิดเป็นเสียงชนิดที่เรียกว่าเสียงเสียดแทรก เป็นต้น

3. ปุ่มเหงือก เป็นส่วนนูนออกมาอยู่หลังฟันด้านบน ถ้าเอาลิ้นแตะดูจะรู้สึกว่าเป็นคลื่น

4. เพดานแข็ง หรือเพดานปาก คือส่วนเฉพาะที่โค้งเป็นกระดูกแข็ง

5. เพดานอ่อน คือส่วนที่อยู่ต่อจากเพดานแข็งไปข้างใน มีลักษณะเป็นกระดูกอ่อนที่ขยับขึ้น-ลงได้ เวลาหายใจเพดานอ่อนและลิ้นไก่ซึ่งอยู่ปลายเพดานอ่อนลระดับลงมา เปิดช่องว่างให้ลมออกทางจมูก เวลาพูดส่วนใหญ่ปลายเพดานอ่อนและลิ้นไก่อจะถูกยกขึ้นไปจรดหลังคอ นอกจากเวลาออกเสียงนาสิกเท่านั้นที่เพดานอ่อนจะลระดับลงมา เพื่อให้ลมออกทางจมูก

6. ลิ้นไก่ เป็นก้อนเนื้อเล็ก ๆ อยู่ต่อปลายเพดานอ่อนตรงกลางปาก อวัยวะส่วนนี้สั้นรัวได้

7. ลิ้น เป็นส่วนที่เคลื่อนที่มากที่สุดในการออกเสียงพูด จึงต้องแบ่งออกเป็น 3 ส่วน ตามหน้าที่ในการออกเสียง คือ

7.1) ปลายลิ้น คือส่วนปลายลิ้นซึ่งสามารถยกขึ้นไปแตะกับอวัยวะส่วนต่าง ๆ ในปากตอนบนได้โดยง่าย

7.2) หน้าลิ้น คือส่วนที่อยู่ตรงข้ามกับเพดานแข็ง ถ้าวางลิ้นราบกับปาก

7.3) หลังลิ้น ถ้าวางลิ้นราบกับปาก ลิ้นส่วนนี้จะอยู่ตรงข้ามกับเพดานอ่อน

8. แผ่นเนื้อปากหลอดลม เป็นก้อนเนื้อเล็ก ๆ กล้ายลิ้นไก่ อยู่ต่อโคนลิ้นลงไปในคอ มีหน้าที่ปิดช่องลมในขณะที่รับประทานอาหาร และเปิดช่องลมเมื่อพูด

9. กรวยคอ หมายถึงโพรงคอที่อยู่ถัดจากช่องปากลงไปจนถึงเส้นเสียง

10. เส้นเสียง หรือสายเสียง เป็นอวัยวะสำคัญที่ทำให้เกิดเสียง เส้นเสียงมีลักษณะเป็นกล้ามเนื้อ 2 แผ่นปิดขวางอยู่บริเวณปากช่องหลอดลมจากด้านหลังมาด้านหน้า ระหว่างเส้นเสียงจะมีช่องว่างซึ่งเป็นช่องผ่านไหล่ลมไปถึงปอดและออกมาจากปอดได้ ช่องว่างนี้เรียกว่า ช่องว่างระหว่างเส้นเสียง (Glottis) เส้นเสียงทั้งสองสามารถดึงให้ห่างออกหรือดึงเข้าหากันได้ เส้นเสียงนี้เป็นส่วนสำคัญที่ทำให้เกิดเสียงพูดขึ้น

11. ช่องจมูก หมายถึง โพรงในช่องจมูกซึ่งอยู่เหนือลิ้นไก่ขึ้นไป เป็นช่องที่ลมซึ่งผ่านเส้นเสียงขึ้นมาจะผ่านออกไปทางจมูกได้เมื่อเวลาหายใจและเวลาออกเสียงนาสิก ในเวลาที่พูดเสียงอื่นลิ้นไก่จะถูกยกขึ้นไปปิดช่องจมูกเพื่อให้ลมออกมาทางปาก

เสียงที่เกิดขึ้นนั้นไม่ว่าจะเป็นเสียงประเภทใด จะผ่านไปตามทางเดินของเสียง โดยจะเริ่มตั้งแต่ช่องว่างระหว่างเส้นเสียงถึงริมฝีปาก ในทางวิทยาศาสตร์สรุปได้ว่าทางเดินของเสียงคือ ท่อนำเสียงที่มีรูปร่างไม่แน่นอน

12. เส้นเสียงปลอม เป็นอวัยวะที่มีลักษณะเหมือนเส้นเสียงแต่อยู่เหนือเส้นเสียงขึ้นไป เส้นเสียงปลอมนี้เข้าใจกันว่าจะดึงเข้าหากันเมื่อเวลาพูดเสียงกระซิบ

2.1.2 การเกิดของเสียง (Speech Production)

การเกิดของเสียงแบ่งออกเป็น 3 ขั้นตอนคือ

ขั้นตอนที่ 1 จุดเริ่มต้น เป็นขั้นตอนที่ลมเริ่มถูกขับออกจากปอด ผ่านเข้าไปสู่ขั้นตอนที่ 2

ขั้นตอนที่ 2 การคัดแปลงลมที่เส้นเสียง อวัยวะที่ใช้ในขั้นตอนนี้คือส่วนที่ต่อจากปอดขึ้นมาถึงกล่องเสียงและที่กล่องเสียง เส้นเสียงจะทำหน้าที่เป็นลิ้นปิดเปิดทำให้เกิดเสียง 2 ชนิดคือ

1. เสียงก้อง หรือเสียงโฆมะ (Voice) คือเสียงที่เกิดจากเส้นเสียงปิดกั้นลมไว้ ลมที่ผ่านออกมาจะเพิ่มแรงดันมากขึ้นจนเส้นเสียงปิด-เปิดสลับกันไป เป็นผลให้สัญญาณเสียงที่ได้ (speech waveform) มีลักษณะเป็นคาบ (quasi-periodic) ซึ่งสามารถเรียกความถี่ในการปิด-เปิดของเส้นเสียงว่า ความถี่มูลฐาน (Fundamental Frequency : F_0)

2. เสียงไม่ก้อง หรือเสียงอโฆมะ (Unvoice หรือ Voiceless) คือเสียงที่เกิดในขณะที่เส้นเสียงเปิด อากาศที่ไหลผ่านอย่างรวดเร็วทำให้เกิดเสียงที่เป็นเสียงของสัญญาณรบกวน (Noise) ซึ่งไม่เป็นคาบ

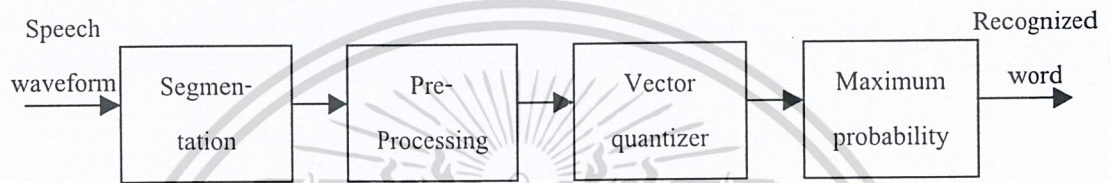
ขั้นตอนที่ 3 การเปลี่ยนแปลงลักษณะเส้นเสียง อวัยวะที่ใช้คือส่วนที่ต่อจากกล่องเสียงจนถึงริมฝีปาก โดยลมที่ผ่านออกจากกล่องเสียงจะทำให้เกิดเสียงในลักษณะต่าง ๆ ซึ่งจะเกิดจากการเปลี่ยนแปลงของอวัยวะที่กล่าวไว้ในหัวข้อที่แล้ว

2.2 ทฤษฎีการรู้จำเสียง

โครงสร้างพื้นฐานของระบบการรู้จำเสียงซึ่งมีรูปแบบการรู้จำเสียงพูดแบบคำเดี่ยว (Isolated word recognition) สามารถแบ่งออกได้เป็น 3 ส่วน คือ

1. การเตรียมสัญญาณในการวิเคราะห์ (Segmentation)
2. การเตรียมและวิเคราะห์เสียงเบื้องต้น (Pre-Processing)
3. การควอนไทเซชันแบบเวกเตอร์ (Vector Quantization)
4. การกำหนดแบบจำลองการรู้จำเสียง (Pattern Matching)

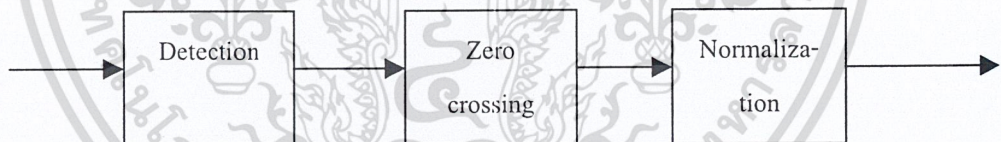
การรู้จำเสียง เมื่อสัญญาณเข้าเป็นคลื่นเสียง ส่วนสัญญาณออกเป็นคำที่วิเคราะห์จากสัญญาณเข้า สามารถเขียนเป็นบล็อกไดอะแกรม (Block diagram) ได้ดังนี้



รูปที่ 2.1 บล็อกไดอะแกรมของการรู้จำเสียง

2.2.1 การเตรียมสัญญาณเสียงในการวิเคราะห์

ก่อนที่จะทำการวิเคราะห์เสียง เราต้องทำการเตรียมสัญญาณเสียงซึ่งขั้นตอนในการเตรียมสัญญาณเพื่อการวิเคราะห์ สามารถเขียนเป็นบล็อกไดอะแกรมได้ดังนี้



รูปที่ 2.2 บล็อกไดอะแกรมของการเตรียมสัญญาณเสียงในการวิเคราะห์

การหาขอบเขตของคำ

การหาขอบเขตของคำแบ่งออกได้เป็น 2 ขั้นตอน คือ

ขั้นตอนที่ 1 การหาค่าพลังงานของเสียง (Energy Detection)

พลังงานของสัญญาณเป็นตัวแทนอันหนึ่งที่เรามักจะนำมาใช้ในการวิเคราะห์ลักษณะต่าง ๆ ของสัญญาณทั่ว ๆ ไป โดยพลังงานของสัญญาณ $s(n)$ ใด ๆ ที่แปรตามเวลาสามารถนิยามได้ว่า

$$E = \sum_{n=-\infty}^{\infty} S^2(n) \quad (2.1)$$

แต่สำหรับสัญญาณเสียงซึ่งเป็นสัญญาณที่แปรเปลี่ยนอยู่ตลอดเวลา ไม่มีเสถียรภาพตามเวลา เราจะต้องแบ่งสัญญาณออกมาพิจารณาเป็นช่วงเล็ก ๆ ตามแกนเวลาหรือเรียกว่าแบ่งออกเป็นเฟรม ดังนั้นก็จะสามารถหาพลังงานของเสียงในแต่ละเฟรมได้เป็น

$$E_l(m) = \sum_{n=0}^{N-1} s^2(n) \quad (2.2)$$

โดยที่ l แทนลำดับของเฟรมข้อมูลเสียง, $l = 0, 1, 2, 3, \dots, L$
 N แทนจำนวนข้อมูลเสียงในแต่ละเฟรม

การวัดค่าพลังงานดังในสมการ (2.2) นั้น มีข้อจำกัดตรงที่ว่ามันจะมีความไวต่อสัญญาณที่มีขนาดใหญ่ ๆ เนื่องจากเราใช้วิธียกกำลังสองค่าของสัญญาณอินพุต ดังนั้นการแก้ปัญหาอย่างหนึ่งก็คือวัดพลังงานของเสียงโดยใช้สมการดังนี้

$$E_l(m) = \sum_{n=0}^{N-1} |s(n)| \quad (2.3)$$

ขั้นตอนที่ 2 การหาค่าอัตราการตัดศูนย์ (Zero Crossing)

การเกิดอัตราการตัดศูนย์จะเกิดขึ้นจากการที่รูปคลื่นของสัญญาณมีการตัดกับแกนเวลานั้นคือค่าของสัญญาณจะมีการเปลี่ยนสัญลักษณ์ทางคณิตศาสตร์นั่นเอง อัตราการเกิดอัตราการตัดศูนย์เป็นเครื่องมืออย่างง่ายที่ใช้อธิบายการเปลี่ยนแปลงข้อมูลของสัญญาณค่าอัตราการตัดศูนย์นั้น สามารถนำมาใช้ในการตัดสินใจว่าสัญญาณเสียงนั้นเป็นเสียงก้อง หรือเสียงไม่ก้อง เนื่องจากเสียงก้องส่วนใหญ่จะมีค่าพลังงานอยู่ในช่วงความถี่ต่ำ ส่วนเสียงไม่ก้องจะมีพลังงานอยู่ในช่วงความถี่สูง และค่าอัตราการตัดศูนย์ก็มีความสัมพันธ์โดยตรงกับค่าความถี่ของสัญญาณ ดังนั้นจึงอาจสรุปได้ว่า สัญญาณเสียงที่มีค่าอัตราการตัดศูนย์สูงจะเป็นเสียงไม่ก้องและสัญญาณที่มีค่าอัตราการตัดศูนย์ต่ำจะเป็นเสียงก้อง แต่อย่างไรก็ตามการกำหนดขนาดของค่าอัตราการตัดศูนย์ที่แน่นอนเพื่อจำแนกชนิดของเสียงนั้น จะต้องอาศัยผลจากการทดลองเป็นหลัก

ในการหาค่าอัตราการตัดศูนย์สามารถทำได้โดยสมการดังนี้

$$Z = \frac{1}{2N} \sum_{n=1}^N |\text{Sign}(s(n)) - \text{Sign}(s(n-1))| \quad (2.4)$$

$$\text{Sign}(s(n)) = 1 \quad ; \quad s(n) > 0$$

$$0 \quad ; \quad \text{at other}$$

การนอลมอลไลเซชัน (Normalization)

เป็นการทำให้สัญญาณที่จะนำมาวิเคราะห์มีขนาดเท่ากันเพื่อให้ง่ายต่อการวิเคราะห์ โดยใช้หลักการ Interpolation จาก Numerical Mathematics มีสมการเป็นดังนี้

$$g(x) = \frac{b-x}{b-a} f(a) + \frac{x-a}{b-a} f(b) \quad (2.5)$$

2.2.2 การวิเคราะห์เสียงเบื้องต้น

ในการวิเคราะห์เสียงเพื่อนำไปทำแบบจำลอง (Model) จำเป็นต้องทำการวิเคราะห์เสียงเบื้องต้นก่อนเพื่อให้ได้ค่าตัวอย่างของแต่ละเสียงที่สามารถเป็นตัวแทนของเสียงนั้นได้ ทำให้การวิเคราะห์ขั้นต่อไปมีความสะดวก ขั้นตอนการวิเคราะห์เสียงเบื้องต้นสามารถเขียนเป็นบล็อกไดอะแกรมได้ดังนี้



รูปที่ 2.3 บล็อกไดอะแกรมของการวิเคราะห์เสียงเบื้องต้น

การพรีเอมฟาสีส (Pre-emphasis)

เนื่องจากองค์ประกอบส่วนใหญ่ของเสียงพูด จะอยู่ในบริเวณความถี่ต่ำ ดังนั้นเพื่อให้อัตราส่วนของสัญญาณต่อสัญญาณรบกวน (Signal to Noise Ratio : SNR) มีค่าคงที่ตลอดช่วงความถี่ของสัญญาณเสียงที่ใช้งาน จึงต้องมีการพรีเอมฟาสีส โดยเน้นความถี่สูงให้มีขนาดสูงขึ้น นั่นคือ การพรีเอมฟาสีสเป็นการกรองสัญญาณด้วยวงจรความถี่สูงผ่าน (High Pass Filter) นิยมใช้วงจรกรองอันดับหนึ่ง ซึ่งเป็นวงจรที่ง่ายที่สุด มีรูปแบบสมการดังนี้

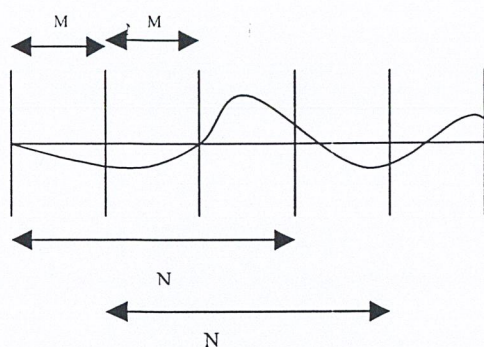
$$y(n) = x(n) - ax(n-1) \quad (2.6)$$

มีฟังก์ชันถ่ายโอนเป็น

$$H(z) = 1 - az^{-1} \quad ; \quad 0.9 < a < 1.0 \quad (2.7)$$

การแบ่งช่วงสัญญาณ (Frame Blocking)

สัญญาณที่ผ่านการพรีเอมฟาสีสแล้ว จะถูกแบ่งออกเป็นเฟรม (Frame) หรือช่วงสัญญาณ ซึ่งแต่ละช่วงสัญญาณจะประกอบด้วยสัญญาณเสียง N สัญญาณ และในแต่ละช่วงสัญญาณจะถูกวิเคราะห์โดยเลื่อนสัญญาณไปครั้งละ M สัญญาณ ดังแสดงในรูปที่ 2.4



รูปที่ 2.4 การแบ่งช่วงของสัญญาณ

จะเห็นได้ว่าถ้า ค่า M มีค่าน้อยกว่าค่า N มากเท่าใด จะทำให้การวิเคราะห์สัญญาณมีความแม่นยำมากขึ้น แต่ข้อเสียคือจะต้องใช้เวลาในการวิเคราะห์นานขึ้น และถ้าค่า M มีค่ามากกว่าค่า N แล้วจะทำให้สัญญาณบางส่วนไม่ถูกใช้ในการวิเคราะห์จะเกิดการผิดพลาดในการวิเคราะห์สัญญาณในส่วนต่อไป

การกำหนดขนาดของช่วงสัญญาณมีเงื่อนไขในการเลือกดังนี้

- 1.) ค่า M จะต้องสั้นพอที่ทำให้คุณสมบัติของเสียงไม่เปลี่ยนแปลง
- 2.) ค่า N จะต้องยาวพอที่จำนวนของตัวอย่างมีเพียงพอต่อการหาสัมประสิทธิ์
- 3.) การเลื่อนการวิเคราะห์ (ค่า M) ต้อง ไม่ข้ามข้อมูล

การวินโดว์ (Windowing)

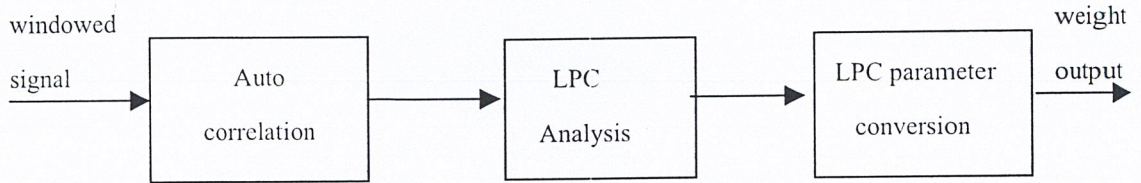
ในแต่ละช่วงสัญญาณที่ถูกนำมาวิเคราะห์นั้น ที่ขอบของช่วงสัญญาณนั้นจะมีความไม่ต่อเนื่องของสัญญาณเกิดขึ้นเมื่อมองใน โดเมนของความถี่สูง ดังนั้นเพื่อจะลดความไม่ต่อเนื่องดังกล่าวนี้ จะต้องคูณแต่ละช่วงสัญญาณนี้ด้วยฟังก์ชันวินโดว์ ในที่นี้จะใช้ฟังก์ชันวินโดว์แฮมมิง (Hamming window function) มีสมการดังนี้

$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right) ; n = 0, 1, 2, \dots, N-1 \quad (2.8)$$

การวิเคราะห์หาคุณลักษณะของเสียง (Feature extraction)

เป็นเทคนิคในการแปลงข้อมูลที่มีอยู่มากมาย ให้เป็นส่วนเล็ก ๆ ซึ่งส่วนเล็ก ๆ นี้จะแสดงคุณสมบัติของคลื่นเสียงนั้น ๆ โดยจะใช้วิธีการป้อนสัญญาณผ่านวงจรรอง (Filter) และการประมาณเชิงเส้น

หลักการพื้นฐานของการประมาณเชิงเส้นคือ การประมาณเสียงจากผลรวมเชิงเส้นของสัญญาณเสียงในอดีต โดยอาศัยหลักเกณฑ์กำลังสองของสัญญาณความคลาดเคลื่อนมีค่าต่ำสุดในการหาสัมประสิทธิ์การประมาณเชิงเส้น กล่าวคือหลังจากการทำพีเอชไอและวินโดว์ครบทุกตัวในหนึ่งเฟรมแล้ว จะทำการคำนวณออโตคอร์รีเลชัน ก็จะทำการหา สัมประสิทธิ์แอลพีซี และอัตราการขยาย G มีขั้นตอนการแสดงดังรูปที่ 2.5



รูปที่ 2.5 แสดงขั้นตอนในการวิเคราะห์หาคุณลักษณะของเสียง

ออโตคอร์รีเลชัน (Auto correlation)

ให้สัญญาณเดิมเป็น $S(n)$ การประมาณค่าสัญญาณเป็น $S'(n)$ ดังนั้นสามารถอธิบายการประมาณเชิงเส้นด้วยสมการดังนี้

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

เมื่อ α_k เป็นค่าคงที่ เรียกว่าวิธีการประมาณเชิงเส้นอันดับ p โดยมีเงื่อนไขว่า ค่า α_k ที่ใช้ในการประมาณที่จะต้องทำให้ผลรวมของกำลังสองของความคลาดเคลื่อน $\{S(n) - S'(n)\}^2$ มีค่าน้อยที่สุด ซึ่งใช้ในการประมาณเชิงเส้นวิธีออโตคอร์รีเลชัน (Autocorrelation Method) หรือวิธีออดิตัมพันธ์

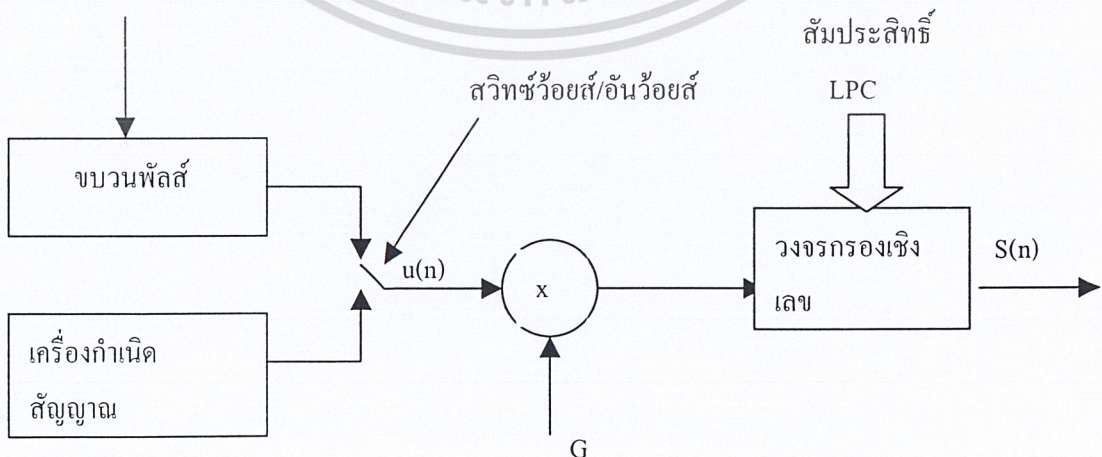
การคำนวณออโตคอร์รีเลชัน เป็นการเปรียบเทียบสัญญาณกับสัญญาณของตัวเองที่ถูกละเลี่ยนไปตามแกนเวลา ที่ใช้วิธีนี้เนื่องจากการคำนวณที่มีการแก้สมการที่น้อยกว่าวิธีอื่น ๆ และมีความแน่นอนในด้านเสถียรภาพ อีกทั้งมีการเก็บข้อมูลที่น้อยกว่า

คุณสมบัติของออโตคอร์รีเลชัน

1. เป็นฟังก์ชันคู่ $R(k) = R(-k)$
2. จะเป็นค่าสูงสุดเมื่อเปรียบเทียบสัญญาณของตัวเองที่ตำแหน่งทางแกนเวลาเดียวกัน คือ

$$R(0) = \max$$

จากหลักการพื้นฐานของการประมาณเชิงเส้นและแบบจำลองระบบสร้างสัญญาณเสียงสามารถเขียนบล็อกไดอะแกรมการทำารประมาณเชิงเส้นมาสร้างสัญญาณเสียงพูดได้ดังรูปที่ 2.6



รูปที่ 2.6 บล็อกไดอะแกรมแสดงโมเดลการสร้างสัญญาณเสียงพูดอย่างง่าย

จากรูปที่ 2.6 สามารถเขียนสมการได้เป็น

$$S(n) = G * u(n) + \sum_{k=1}^p a_k s(n-k) \quad (2.10)$$

การประมาณเชิงเส้นโดยใช้สัมประสิทธิ์ α_k คือ

$$S'(n) = \sum_{k=1}^p \alpha_k S(n-k) \quad (2.11)$$

ดังนั้น ความคลาดเคลื่อน คือ

$$\begin{aligned} e(n) &= S(n) - S'(n) \\ &= S(n) - \sum_{k=1}^p \alpha_k s(n-k) \end{aligned} \quad (2.12)$$

ฟังก์ชันถ่ายโอนระหว่าง $e(n)$ และ $S(n)$ คือ

$$\begin{aligned} A(z) &= E(z)/S(z) \\ &= 1 - \sum_{k=1}^p \alpha_k z^{-k} \end{aligned} \quad (2.13)$$

จากสมการที่(2.12)และ(2.10)จะเห็นได้ว่าถ้า $\{\alpha_k\} = \{a_k\}$ แล้ว

$$e(n) = G * u(n) \quad (2.14)$$

ดังนั้น ค่าผลรวมของกำลังสองของความคลาดเคลื่อน

$$\begin{aligned} E_n &= \sum_m e_n^2(m) \\ E_n &= \sum_m [S(m) - S'(m)]^2 \end{aligned} \quad (2.15)$$

โดยที่ n คือช่วงที่ n ของสัญญาณที่ใช้คำนวณ เพราะฉะนั้นเพื่อให้ได้ค่า E_n ต่ำสุดจะต้องมีเงื่อนไขว่า

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

จากสมการ (2.15)

$$\begin{aligned} \frac{\partial E_n}{\partial \alpha_i} &= -2s(m-i) \sum_m \left[s_n(m) - \sum_{k=1}^p \alpha_k s_n(m-k) \right] \quad \text{เมื่อ } i = 1, 2, 3, \dots, p \\ &= -2 \left[\sum_m s(m)s(m-i) \sum_{k=1}^p \sum_m \alpha_k s(m-k)s(m-i) \right] \end{aligned} \quad (2.16)$$

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

$$\sum_{k=1}^p \alpha_k \sum_m s_n(m-k)s_n(m-i) = \sum_m s_n(m)s_n(m-i) ; i = 1, 2, 3, \dots, p \quad (2.17)$$

ถ้าเรากำหนดให้ $\varphi_n(i, j) = \sum_m s_n(m-k)s_n(m-i)$ เพราะฉะนั้น

$$\sum_{k=1}^p \alpha_k \varphi_n(i, j) = \varphi_n(i, 0) \quad (2.18)$$

โดยสมการ(2.15) และ (2.16) 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 $\varphi_n(i, j) = \sum_m s_n(m-k)s_n(m-i)$

$$E_n = \varphi_n(0, 0) - \sum_{k=1}^p \alpha_k \varphi_n(0, k) \quad (2.19)$$

สมมติว่าใน 1 เฟรม ของสัญญาณที่ตัดมามีจำนวน N ตัวอย่าง คือ $s_n(0), s_n(1), s_n(2), \dots, s_n(N-1)$ ในที่นี้เราให้ $s_n(m) = 0$ เมื่อ $m < 0$ หรือ $m > N+1$ เพราะฉะนั้น

$$\begin{aligned} \varphi_n(i, j) &= \sum_m s_n(m-k)s_n(m-i) \\ &= \sum_{m=0}^{N-1-(i-k)} s_n(m)s_n(m+i-k) \quad 0 \leq k \leq p, 1 \leq i \leq p \end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\text{ให้ } R_n(k) = \sum_{m=0}^{N-1-k} s_n(m)s_n(m+k) \quad \text{เมื่อ } k = 0,1,2,\dots,p \quad (2.20)$$

ดังนั้นจากสมการ (2.19) และ (2.20) จะได้ว่า

$$R_n(k) = \sum_{m=0}^{N-1-k} x'(m)x'(m+k) \quad (2.21)$$

จากสมการที่ (2.20) จะได้ว่า

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

สามารถเขียนให้อยู่ในรูปเมทริกซ์ได้เป็น

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

หรือ

$$R_n \cdot \alpha = r_n \quad (2.22)$$

เมื่อ

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

การหาพารามิเตอร์แอลพีซี

พารามิเตอร์แอลพีซี ได้แก่ สัมประสิทธิ์ α และอัตราขยาย G

จากสมการ (2.22) สามารถหาสัมประสิทธิ์ α โดย

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

จากสมการ (2.14) และ (2.15) จะได้ว่า

$$e(n) = G * u(n) \quad (2.23)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$E_n = \sum_{m=0}^{N-1} e^2(m) = G \sum_{m=0}^{N-1} u^2(m) \quad (2.24)$$

และจากสมการ (2.19) จะได้ว่า

$$E_n = \varphi_n(0,0) - \sum_{k=1}^p \alpha_k R_n(k) \quad (2.25)$$

$$= R_n(0) - \sum_{k=1}^p \alpha_k R_n(k) \quad (2.26)$$

และจากสมการ 2.31 เราสามารถหาค่าอัตราขยาย G โดยตรงจาก

$$G^2 = \frac{R_n(0) - \sum_{k=1}^p \alpha_k R_n(k)}{\sum_{k=0}^{N-1} u^2(m)} \quad (2.27)$$

จากที่กล่าวมาในขั้นตอนนี้ ได้ว่าเมื่อวิเคราะห์สัญญาณไป 1 ช่วงสัญญาณแล้วจะได้ค่าสัมประสิทธิ์ α ทั้งหมด p ค่า: $\alpha_1, \alpha_2, \dots, \alpha_p$

การเปลี่ยนพารามิเตอร์แอลพีซีเป็นสัมประสิทธิ์เซปสตรีม

หลังจากการหาพารามิเตอร์แอลพีซีและอัตราขยายใน 1 เฟรมแล้วจะเปลี่ยนให้เป็นสัมประสิทธิ์เซปสตรีม เนื่องจากการรู้จำเสียงพูดนั้น สัมประสิทธิ์เซปสตรีมเป็นพารามิเตอร์ที่มีลักษณะน่าเชื่อถือได้ดีกว่าพารามิเตอร์แอลพีซีที่ยังมีความสัมพันธ์ใกล้ชิดในการรับรู้เสียงตามความรู้สึกของมนุษย์ โดยแท้จริง

การเวทค่าพารามิเตอร์ (Parameter Weighting)

เนื่องจากค่าสัมประสิทธิ์เซปสตรีมที่ได้นั้น ช่วงลำดับต้น ๆ และลำดับท้าย ๆ ของเฟรมที่นำมาวิเคราะห์จะเกิดความคลาดเคลื่อนมากกว่าบริเวณส่วนอื่น เพราะฉะนั้นจึงทำการถ่วงน้ำหนักเพื่อลดค่าความคลาดเคลื่อนดังกล่าวนี้ ด้วยฟังก์ชันเวทดัง ดังนี้คือ

$$W_m = \left[1 + \frac{Q}{2} \sin\left(\frac{\pi m}{Q}\right) \right], \quad 1 \leq m \leq Q \quad (2.38)$$

2.2.3 การจัดระดับเวกเตอร์ (Vector Quantization)

เวกเตอร์ควอนไทซ์เซชัน เป็นวิธีการลดโดเมนชั้น (Dimension) หรือจำนวนข้อมูล เวกเตอร์อินพุท หรือ เทรนนิ่งเซต (Training set) หรือพารามิเตอร์ที่ได้จากขั้นแอลพีซี จะถูกเลือกมากลุ่มหนึ่งซึ่งใช้เป็นตัวแทนของข้อมูลจำนวนหนึ่งหรือเรียกว่าการหาโค้ดบุค (Codebook) อินพุทที่เข้ามาจะถูกทำการเอกซารนี้เป็นเอกซารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

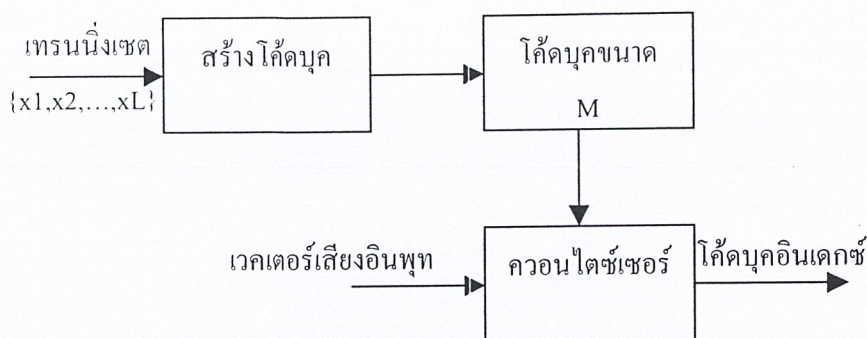
เปรียบเทียบกับ ไม้ค้อนที่มีอยู่ โดยจะพิจารณาว่าอินพุทที่เข้ามานั้นห่างจากไม้ค้อนใดน้อยที่สุด อินพุทดังกล่าวจะถูกแทนด้วยเวกเตอร์โค้ด (Index) นั้น อินพุททุกตัวเป็นสมาชิกของเวกเตอร์โค้ดใดๆจะถูกนำมาหาจุดศูนย์กลางร่วมใหม่ และนำจุดศูนย์กลางที่ได้นี้ไปทำการหาความคลาดเคลื่อนกับสมาชิกทุกตัว ถ้าค่าความคลาดเคลื่อนที่ได้มีค่ามากกว่าค่าที่กำหนดไว้ค่าหนึ่งหรือค่าที่ยอมรับได้ ก็จะนำศูนย์กลางใหม่นั้นไปเป็นไม้ค้อน แทน และจะทำการจัดกลุ่มอินพุทเข้ากับไม้ค้อนใหม่ที่ได้และหาค่าความคลาดเคลื่อนอีกครั้ง ทำอย่างนี้ซ้ำๆจนกระทั่งค่าความคลาดเคลื่อนมีค่าน้อยถึงค่าที่ยอมรับได้ ก็จะถือว่าไม้ค้อน ที่ดีที่สุดจะเป็นตัวแทนของอินพุททั้งหมด จะสังเกตได้ว่าทุกครั้งที่มีการหาไม้ค้อนใหม่นั้น ค่าความคลาดเคลื่อนที่ได้จะมีค่าลดลงทุกครั้งด้วย



รูปที่ 2.7 แสดงการกระจายเฟรมของเสียงพูด แต่ละจุดแทนเฟรมของเสียง

รูปที่ 2.8 การรวมกลุ่มของเฟรมเสียงเพื่อสร้างไม้ค้อน X

ตัวอย่างเวกเตอร์ควอนไทซ์แซนน์ สมมติให้เวกเตอร์แต่ละตัวมี 2 มิติ และทำการหาไม้ค้อนขนาด 8 เวกเตอร์ ทั้งหมดจะถูกจัดเข้ากลุ่มของไม้ค้อนต่างๆ แล้วทำการหาจุดศูนย์กลางใหม่โดยการเฉลี่ยค่าเวกเตอร์สมาชิกทุกตัวที่อยู่ในกลุ่มเดียวกัน ผลที่ได้คือไม้ค้อน 8 ตัว เป็นตัวแทนของเวกเตอร์ทั้งหมด



รูปที่ 2.9 บล็อกไดอะแกรมของเวกเตอร์ควอนไทซ์

การทำงานของควอนไทซ์แบบเวกเตอร์ แบ่งเป็น 2 ขั้นตอนดังนี้

1. การสร้างโค้ดบุคโดยวิธีเค-มีนส์ (K-mean)

จากขั้นตอนการประมาณเชิงเส้นของเสียงตัวอย่างจำนวนมากจะได้เทรนนิ่งเซตซึ่งประกอบด้วยเวกเตอร์สเปกตรัมจำนวน L เฟรม ; $X = \{X_i; 1 \leq i \leq L\}$ เฟรมละ P มิติ ; $X = [X_1, X_2, \dots, X_p]$ แล้วนำข้อมูลที่ได้ออกมาทำการสร้างกลุ่มของแบบอ้างอิง

ในระบบการรับรู้เสียงพูดแบบต่างๆ จะใช้อ้างอิงของคำหนึ่งๆ จากผู้พูดจำนวนมาก เพื่อที่จะได้ครอบคลุมถึงความแปรปรวนต่างๆ ที่เกิดระหว่างผู้พูดแต่ละคน เนื่องจากถ้าใช้แบบอ้างอิงจำนวนมากเวลาที่ใช้ในการตอบสนองจะมาก เนื้อที่หน่วยความจำสำรองที่ใช้เก็บแบบอ้างอิงจะเพิ่ม และเมื่อเพิ่มแบบอ้างอิงไประดับหนึ่ง ความถูกต้องในการรับรู้จะเริ่มคงที่ ดังนั้นการจัดกลุ่มแบบอ้างอิงใหม่ เพื่อให้ได้แบบอ้างอิงที่เหมาะสม และสามารถใช้เป็นตัวแทนของแบบอ้างอิงที่มีอยู่ทั้งหมดได้ อัลกอริทึมที่ใช้ได้แก่ เค-มีนส์ อัลกอริทึม (K-means Algorithm) ขั้นตอนการสร้างโค้ดบุคมีดังนี้

1) นำเทรนนิ่งเซตมาใช้ในการสร้างโค้ดบุค

ขนาดโค้ดบุคของการควอนไทซ์แบบเวกเตอร์คือ $M = 2^b$ เวกเตอร์ และเพื่อที่จะหาเซตของ M โค้ดบุคที่ดีที่สุด จำนวนเวกเตอร์อินพุตจะต้องมากกว่าขนาดโค้ดบุคมากๆ

2) การสุ่มค่าเริ่มต้น

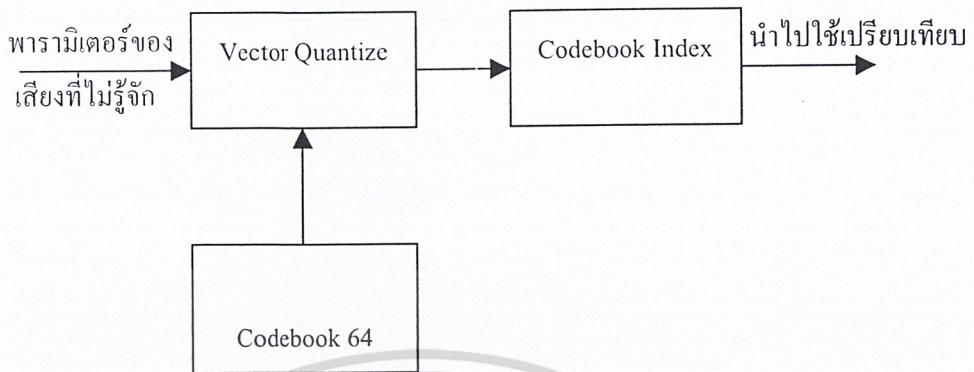
การสุ่มค่าเริ่มต้น เป็นวิธีหนึ่งในการออกแบบโค้ดบุค ซึ่งคือ การเลือกค่าเริ่มต้นของโค้ดบุค เรียกโค้ดบุคที่ได้จากการสุ่มค่าเริ่มต้นนี้ว่า แรนดอม โค้ดบุค (random codebook) ถึงแม้วิธีนี้จะไม่ใช่วิธีที่ดีที่สุด แต่โค้ดบุคที่ได้จากการสุ่มก็ได้ผลเป็นที่ยอมรับ

3) การหาความคลาดเคลื่อน

การหาความคลาดเคลื่อน เป็นส่วนที่จำเป็นและเป็นประโยชน์ต่อการออกแบบโค้ดบุค สมการทางพีชคณิตที่ใช้ในการคำนวณหาระยะทาง มีหลายวิธีแต่นำมาใช้คือ การหาความคลาดเคลื่อนกำลังสองรวม (Total Square Error) ซึ่งเป็นวิธีที่ง่ายและรวดเร็ว

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

$$d(v_1, v_2) = \|v_1 - v_2\|^2 = \sum_{i=0}^{k-1} (x_i - y_i)^2 \tag{2.39}$$



รูปที่ 2.10 ขั้นตอนของเวกเตอร์ควอนไทซ์เซชัน

4) การจัดกลุ่ม (classification) และการหาจุดศูนย์กลางของกลุ่ม (center cluster)

การจัดกลุ่มเป็นการแบ่งเวกเตอร์อินพุตเข้าไปตามกลุ่มต่างๆ ของเรนดอมโค้ดบุค โดยพิจารณา ระยะทางหรือความคลาดเคลื่อนน้อยที่สุด ของแต่ละเวกเตอร์อินพุต x กับเวกเตอร์โค้ดบุค y ซึ่งเป็นโค้ดบุคจากนั้นจะทำการหาค่าเฉลี่ยของแต่ละกลุ่ม เพื่อเป็นค่ากลางของกลุ่มนั้นๆจะได้

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

\bar{Y} เป็นจุดศูนย์กลางซึ่งเป็นเวกเตอร์ที่อยู่ตรงกลางของ $\{x_i\}_{i=1}^L$ ซึ่งแต่ละมิติจะไม่ขึ้นแก่กัน หมายความว่าแต่ละ y_k เป็นค่ากลางของ $\{x_i\}_{i=1}^L$ ทำ 2 ขั้นตอนนี้ซ้ำ จะเกิดการลู่เข้า (convergent) โดยความคลาดเคลื่อนรวมจะต่ำกว่าค่าหนึ่งๆ ซึ่งค่าความคลาดเคลื่อนรวมจะลดลงทุกครั้งที่มีการคำนวณซ้ำใหม่ จึงขึ้นกับค่าที่กำหนดว่าจะต้องการให้ความคลาดเคลื่อนรวมน้อยเท่าไร ค่ากลางดังกล่าวของแต่ละกลุ่มจะถูกเก็บเป็นเวกเตอร์โค้ดจะได้ว่า y เป็นควอนไทซ์ของค่า x

โดย $q(\cdot)$ เป็นโอเปอเรเตอร์ของควอนไทซ์ y ถูกเรียกว่าเอาท์พุทเวกเตอร์ของค่า x โดย y เป็นค่าใดค่าหนึ่งใน $y = \{y_i; 1 \leq i \leq M\}$ โดย $y_i = [y_{i1} \ y_{i2} \ \dots \ y_{in}]$ Y เป็นเซตของโค้ดบุค M เป็นขนาดของโค้ดบุค และ $\{y_i\}$ เป็นเซตของเวกเตอร์โค้ด y_i อาจเรียกได้ว่าโค้ดอ้างอิง และ M อาจเรียกว่าจำนวนระดับขั้น จะทำการแบ่งเวกเตอร์ x ไปใน M เซล $\{C_i; 1 \leq i \leq M\}$ เมื่อ x อยู่ในเซล C_i

$$q(x) = y_i \text{ ถ้า } x \in C_i$$

2. ขั้นตอนการเปรียบเทียบ

เวกเตอร์ควอนไทซ์เซชัน ที่ใช้เพื่อการออกแบบรับรู้เสียงพูดนั้น มีจำนวนควอนไทเซอร์ M ตัวซึ่งหมายถึงมี M ระดับเสียงเพื่อการรับรู้ แต่ละระดับเสียงที่พิจารณามาจากเซตของข้อมูลเทรนนิ่ง ซึ่ง M เป็นดัชนีระดับ แต่ละเซตของเทรนนิ่งในแต่ละระดับจะเก็บเสียงที่อยู่ในระดับเดียวกัน

เมื่อมีเสียงที่เราไม่ทราบ (Unknown) ; x , เข้ามา จะเป็นอินพุตเข้าไปยังทุกๆควอนไทเซอร์ค่าดัชนีระดับ (index) ที่ถูกเลือก จะเป็นระดับที่มีความคลาดเคลื่อนเฉลี่ยน้อยที่สุด $D(c)$ เมื่อ $i = 1, 2, \dots$

M ซึ่งความคลาดเคลื่อนเฉลี่ยนั้นหาได้จากการวัดระยะทางโดยใช้วิธีการหาความคลาดเคลื่อนกำลังสองรวม

2.2.4 การสร้างแบบจำลอง (Pattern Matching)

คือการทำหนดรูปแบบเพื่อเปรียบเทียบกับสิ่งที่เราไม่ทราบ โดยจัดกลุ่มของสัญญาณที่เราไม่รู้จักให้อยู่ในกลุ่มใดกลุ่มหนึ่ง วิธีการสร้างแบบจำลองมีอยู่ 2 วิธีคือ

- 1.) ไดนามิกไทม์วาร์ปิง (Dynamic Time Warping :DTW)
- 2.) ฮิดเดนมาร์คอฟโมเดล (Hidden Markov Model :HMM)

ในที่นี้จะเลือกใช้วิธีฮิดเดนมาร์คอฟโมเดล เพื่อสร้างแบบจำลองที่ใช้ในการรู้จำเสียงพูด

ฮิดเดนมาร์คอฟโมเดล

ฮิดเดนมาร์คอฟโมเดล เป็นแบบจำลองทางสถิติซึ่งพัฒนาเพื่อแบ่งกลุ่มของอนุกรมทางเวลาหรือสัญญาณที่ไม่คงที่ นั่นคือ ใช้สำหรับการจัดกลุ่มของสัญญาณที่ไม่รู้จัก (Unknown signal) ให้ไปอยู่ในกลุ่มใดกลุ่มหนึ่งของสัญญาณ ซึ่งฮิดเดนมาร์คอฟโมเดล ได้ถูกนำมาประยุกต์ใช้ในการรู้จำเสียงพูด

ฮิดเดนมาร์คอฟโมเดลแบ่งเป็น 2 ประเภท คือ แบบต่อเนื่อง ๆ (Continuous) และแบบไม่ต่อเนื่อง (Discrete-time) ในที่นี้จะเลือกใช้แบบไม่ต่อเนื่องเพราะเป็นวิธีที่ซับซ้อนน้อยกว่าและใช้ได้กับคำพูดสั้นๆ

ส่วนประกอบของ HMM

1 N คือจำนวนสเตต (state) ในแบบจำลอง โดยสามารถย้ายจากสเตตหนึ่งไปอีกสเตตหนึ่งได้ เราให้เซตของสเตตเป็น $\{1, 2, 3, \dots, N\}$ และสเตตที่เวลา t ใด ๆ เป็น q_t

2 M คือจำนวนของค่าปรากฏต่อ 1 สเตต แทนด้วยสัญลักษณ์

$$V = \{V_1, V_2, V_3, \dots, V_M\}$$

3 $A = \{a_{ij}\}$ คือความน่าจะเป็นในการเปลี่ยนสเตตที่

$$a_{ij} = P\{q_t = j \mid q_{t-1} = i\} \text{ เมื่อ } 1 \leq N$$

4 $B = \{b_j(k)\}$ คือความน่าจะเป็นของการเกิดค่าปรากฏที่

$$b_j(k) = P\{O_t = V_k \mid q_t = j\} \text{ เมื่อ } j = 1, 2, 3, \dots, N$$

5 π_i คือความน่าจะเป็นที่แต่ละสเตตจะเป็นสเตตเริ่มต้น เมื่อ

$$\pi_i = P\{q_1 = i\}$$

โครงสร้างของ HMM

แบ่งตามลักษณะการเปลี่ยนสเตต (transition) ของเมตริกซ์ A

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

1 แบบ อิกอร์ดิกโมเดล (Ergodic Model) หรือ ฟูลลี่คอนเนคเทดโมเดล (Fully Connected Model) แบบจำลองนี้ ทุกเสตท สามารถเปลี่ยนเสตทไปเสตทอื่น ๆ ได้ทุกเสตท

2 แบบ เลฟ-ไรท์โมเดล (Left – Right Model) หรือ แบกิสโมเดล (Bakis Model) แบบจำลองนี้การเปลี่ยนเสตทจะเปลี่ยนจากซ้ายไปขวา มีคุณสมบัติการเปลี่ยนเสตท ดังนี้

1. $a_{jj} = 0, j < 1$ หมายความว่าเมื่อผ่านเสตทใดไปแล้วจะไม่มีการย้อนกลับมายังเสตทนั้นอีก

2. $\pi_i = 1; i = 1$ หมายความว่า ลำดับของเสตทต้องเริ่มต้นที่เสตทที่ 1 เสตทที่เหลือจึงมีความน่าจะเป็นที่จะเป็นเสตทเริ่มต้นเท่ากับศูนย์

เลฟ-ไรท์โมเดล นี้มีกฎข้อบังคับการเปลี่ยนเสตทดังนี้

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

$a_{ij} = 0$ เมื่อ $i > i + \Delta i$ โดยค่าของ $\Delta i = 2$ หมายความว่า การเปลี่ยนเสตทจะสามารถกระโดดข้ามไปยังเสตทข้างหน้าได้ไม่เกิน 2 เสตท จะได้เมตริกซ์การเปลี่ยนเสตทเป็นดังนี้

จะเห็นว่าเสตทสุดท้ายมีสัมประสิทธิ์การเปลี่ยนเสตทเป็น

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

3 แบบ เลฟ-ไรท์โมเดลแบบขนาน (Parallel Left – Right Model) มีคุณสมบัติการเปลี่ยนเสตทคล้ายกับแบบที่ 2 แต่มีความยืดหยุ่นมากกว่า

ปัญหาของ HMM

ปัญหาของ HMM มี 3 ข้อซึ่งต้องใช้อัลกอริทึมวิธีต่าง ๆ ในการคำนวณเพื่อแก้ปัญหา

ปัญหาที่ 1 เมื่อลำดับของค่าปรากฏ $O = \{O_1, O_2, \dots, O_T\}$ และมีแบบจำลอง $\lambda = (A, B, \pi)$

เราจะคำนวณหาค่า $P(O | \lambda)$ ของลำดับของค่าปรากฏได้อย่างไร

ปัญหาที่ 2 เมื่อมีลำดับของค่าปรากฏ $O = \{O_1, O_2, \dots, O_T\}$ และมีแบบจำลอง $\lambda = (A, B, \pi)$

เราจะหาลำดับเสตท $q = \{q_1, q_2, \dots, q_T\}$ ที่เหมาะสมในการให้ค่าปรากฏนั้นได้อย่างไร

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

ลำดับของค่าปรากฏที่ใช้ปรับค่าพารามิเตอร์ A, B และ π เพื่อให้ได้แบบจำลองที่ดีที่สุดนั้นเรียกว่าลำดับเทรนนิ่ง (training sequence)

การคำนวณเพื่อแก้ปัญหาของ HMM

1 การแก้ปัญหาที่ 1 เป็นการคำนวณว่าแบบจำลองจะให้ความน่าจะเป็นที่จะได้ลำดับค่าปรากฏมากที่สุดเพียงใด มีวิธีการเพื่อช่วยแก้ปัญหาโดยใช้กระบวนการต่อไปนี้

1.1 กระบวนการไปข้างหน้า (Forward Procedure)

เมื่อกำหนดให้ตัวแปรไปข้างหน้า (forward variable) $\alpha_t(i) = P(O_1, O_2, \dots, O_T, q_t = i | \lambda)$ หมายถึง ความน่าจะเป็นของการเกิดลำดับค่าปรากฏ O_1, O_2, \dots, O_T ที่จะอยู่ในสเตต i ณ เวลา t โดยมีแบบจำลองเป็น λ โดยเราจะคำนวณหา $\alpha_t(i)$ ได้ดังนี้

1.1.1 การเริ่มต้น (initialization)

เมื่อกำหนด $\alpha_t(i) = \pi_i b_j(O_t)$ ที่เวลาเริ่มต้น $t=1$ และเหตุการณ์เริ่มต้น O_1 เมื่อ $1 \leq i \leq N$

1.1.2 การเหนี่ยวนำ (induction)

$$\alpha_t(i) = \left[\sum_{j=1}^N \alpha_{t-1}(j) a_{ij} \right] b_j(O_t) \text{ เมื่อ } 1 \leq j \leq N$$
 หมายถึง ความน่าจะเป็นของสเตต j ที่เวลา $t-1$ ได้มาจากสเตต i ที่เป็นไปได้ถึง N สเตตที่เวลา t

1.1.3 การสิ้นสุด (termination)

$$P(O | \lambda) = \sum_{i=1}^N \alpha_T(i)$$
 ความน่าจะเป็นของลำดับค่าปรากฏ O ได้จากผลรวมของ $\alpha_T(i)$ จากทุกๆ สเตตเมื่อ $1 \leq j \leq N$

1.2 กระบวนการย้อนกลับ (Backward Procedure)

เมื่อกำหนดให้ตัวแปรย้อนกลับ (backward variable) $\beta_T(i) = P(o_{t+1} o_{t+2} \dots o_T, q_t = i | \lambda)$ หมายถึงความน่าจะเป็นของลำดับค่าปรากฏส่วนหลังจากเวลา $t+1$ ไปจนจบ โดยกำหนดว่าต้องอยู่ที่สเตต i ที่เวลา t และมีแบบจำลองเป็น λ เราจะคำนวณหา $\beta_T(i)$ ได้ดังนี้

1.2.1 การเริ่มต้น

$$\beta_T(i) = 1; 1 \leq j \leq N$$

1.2.2 การเหนี่ยวนำ

$$\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(o_{t+1}) \beta_{t+1}(j); 1 \leq t \leq T-1 \text{ และ } 1 \leq i \leq N$$

2. การแก้ปัญหาที่ 2 เพื่อหาลำดับสเตตที่เหมาะสม

เราจะใช้วิธีวิเทอร์บี อัลกอริทึม (Viterbi algorithm) เพื่อหาลำดับสเตตที่ดีที่สุด ณ เวลา t หนึ่ง ๆ เมื่อกำหนดลำดับเหตุการณ์ $O = (o_1 o_2 \dots o_t)$ โดยนิยามให้

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

หมายถึง ความน่าจะเป็นสูงสุดของเส้นทาง (path) ณ เวลา t ซึ่งเริ่มนับจากเหตุการณ์ที่เวลาเริ่มต้นจนถึงเวลา t ที่สเตต i และโดยการอาศัยคุณสมบัติการเหนี่ยวนำ (induction) เราจะได้

$$\delta_{t+1}(i) = \left[\max_j \delta_t(j) a_{ij} \right] b_j(o_{t+1})$$
 เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราสามารถหาลำดับเซตที่ดีที่สุดได้โดยใช้กระบวนการต่อไปนี้ เมื่อกำหนดให้ $\psi_t(i)$ เป็น อาร์เรย์ (array)

2.1 การเริ่มต้น

$$\delta_t(i) = \pi_j b_j(o_t) \quad \text{เมื่อ } 1 \leq j \leq N$$

$$\psi_t(i) = 0$$

2.2 การย้อนกลับ

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

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

2.3 การสิ้นสุด

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

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

2.4 เส้นทางเดินย้อนกลับ (Path backtracking)

$$q_t^* = \psi_{t+1}(q_{t+1}^*) \quad ; \quad t = T-1, T-2, \dots, 2, 1$$

3. การแก้ปัญหาที่ 3 เพื่อหาโมเดลที่จะให้ผลตามลำดับค่าปรากฏหนึ่ง ๆ โดยเลือกค่าพารามิเตอร์ A, B, π ที่ดีที่สุดโดยใช้ กระบวนการทำซ้ำ (Iterative) วิธีที่เราเลือกใช้คือ วิธี Baum-Welch (Baum-Welch) หรือ อี-เอ็ม (EM : Expectation-Maximization)

เมื่อนิยามให้

$$1. \gamma(i) = P(q_t = i | O, \lambda)$$

หมายถึง ความน่าจะเป็นที่จะอยู่ที่เซต i ณ เวลา t โดยกำหนดลำดับเหตุการณ์ O และแบบจำลอง ให้สามารถแสดงค่า $\gamma_t(i)$ ได้ดังนี้

$$\begin{aligned} \gamma_t(i) &= \frac{P(O, q_t = i | \lambda)}{P(O | \lambda)} \\ &= \frac{P(O, q_t = i | \lambda)}{\sum_{i=1}^N P(O, q_t = i | \lambda)} \end{aligned}$$

เนื่องจาก $P(O, q_t = i | \lambda)$ มีค่าเท่ากับ $\alpha_t(i) \beta_t(i)$ จึงได้

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

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

หมายถึง ความน่าจะเป็นที่จะอยู่ที่เซต i ที่เวลา t และเซต j ที่เวลา $t+1$ เมื่อกำหนดแบบจำลองและลำดับค่าปรากฏให้ ซึ่งจากนิยามของตัวแปรไปข้างหน้าและตัวแปรย้อนกลับ สามารถนำมาสัมพันธ์กับ $\xi_t(i, j)$ ได้ดังนี้

$$\begin{aligned}\xi_t(i, j) &= \frac{P(q_t = i, q_{t+1} = j, O | \lambda)}{P(O | \lambda)} \\ &= \frac{\alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t-1}(j)}{P(O | \lambda)} \\ &= \frac{\alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^N \sum_{j=1}^N \alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}\end{aligned}$$

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

$$\gamma_t(i) = \sum_{j=1}^N \xi_t(i, j)$$

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

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

ดังนั้น สามารถหาค่าพารามิเตอร์ได้ดังนี้

$$\pi_i = \gamma_1(i) \quad ; 1 \leq i \leq N$$

$$a_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$$

$$b_j(k) = \frac{\sum_{t=1, o_t=v_k}^T \gamma_t(j)}{\sum_{t=1}^T \gamma_t(j)}$$

จากกระบวนการข้างต้น ถ้าเราจะคำนวณซ้ำ ๆ โดยให้ $\lambda' = (A', B', \pi')$ แทน $\lambda = (A, B, \pi)$ ซึ่งเป็นแบบจำลองเริ่มต้นแล้ว จะทำให้ความน่าจะเป็นของการเกิดลำดับค่าปรากฏ O ดีขึ้นจนกระทั่งถึงจุดวิกฤตจึงหยุด ซึ่งเราจะได้จุดวิกฤตของฟังก์ชันความน่าจะเป็นในกรณีที่ $\lambda' = \lambda$ หรือถ้า λ' มีความน่าจะเป็นมากกว่าแบบจำลอง λ

ในลักษณะที่ $P(O | \lambda') > P(O | \lambda)$ นั่นก็คือเราจะได้แบบจำลอง λ' ใหม่ที่น่าจะทำให้เกิดลำดับค่าปรากฏ O ได้ดีกว่า

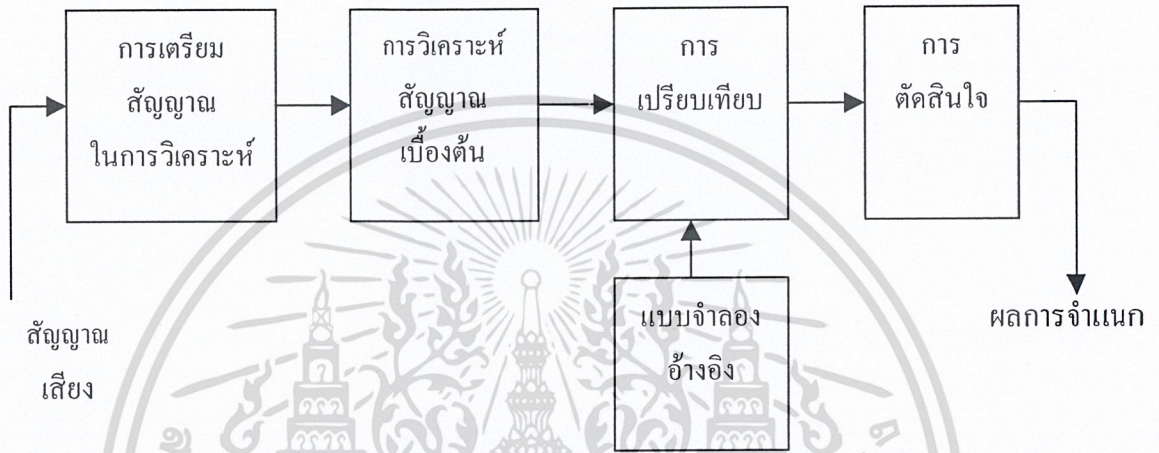
กฎการตัดสินใจ

การตัดสินใจว่าความน่าจะเป็นของเสียงที่เข้าไปเปรียบเทียบกับแบบจำลองอันไหนให้ความน่าจะเป็นมากที่สุด ก็จะตัดสินใจว่าเสียงที่เข้าไปเป็นเสียงที่อยู่ในแบบจำลองนั้น

บทที่ 3

การคำนวณและการสร้าง

ระบบการจำแนกเสียงพูดโดยทั่วไปสามารถแบ่งออกเป็นภาคต่าง ๆ ตามบล็อกไดอะแกรมจากรูปที่ 3.1 ซึ่งประกอบไปด้วยภาคการเตรียมสัญญาณในการวิเคราะห์ ภาควิเคราะห์สัญญาณเบื้องต้น ภาคเปรียบเทียบ ภาคตัดสินใจ และภาคการตัดสินใจ



รูปที่ 3.1 แสดงบล็อกไดอะแกรมการจำแนกเสียงพูดโดยทั่วไป

สำหรับปัญหานี้ การรู้จำเสียงจะแบ่งออกได้เป็นบล็อกไดอะแกรมดังรูปที่ 3.1 ซึ่งสามารถอธิบายได้เป็นส่วน ๆ ดังนี้

3.1 ภาคการเตรียมสัญญาณในการวิเคราะห์

เป็นขั้นตอนการเตรียมข้อมูลของสัญญาณเสียงที่เข้ามาให้เหมาะสม นั่นคือ การหาขอบเขตที่เป็นส่วนของคำที่แท้จริง ประกอบด้วย

3.1.1 การวิเคราะห์หาค่าพลังงานของสัญญาณ

จะใช้หลักการวิเคราะห์ค่าพลังงานของสัญญาณใน โดเมนของเวลา ซึ่งมีวิธีการดังต่อไปนี้

1. กำหนดหาค่าพลังงานของสัญญาณในช่วงเวลาที่ถูกบันทึกทั้งหมด โดยการคำนวณหาค่าพลังงานของสัญญาณซึ่งจะทำการคำนวณเป็นเฟรม กำหนดให้เฟรมหนึ่งมีจำนวนแซมเปิล 100 แซมเปิล และทำการคำนวณเป็นเฟรมต่อไปเรื่อย ๆ จนครบทุกเฟรม
2. พิจารณาค่าพลังงานสูงสุดของสัญญาณ (Maximum energy) และดูว่าที่ค่าพลังงานสูงสุดนี้อยู่ที่เฟรมใด แล้วกำหนดให้เป็นเฟรมที่ m
3. ทำการกำหนดพลังงานอ้างอิง (Reference energy) โดยที่ซึ่งพลังงานอ้างอิงสามารถคำนวณได้

เอกสารจาก a เท่าของพลังงานสูงสุด ($a * E_{max}$) ซึ่งในที่นี้ค่า a มีค่า $= 0.2$ ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

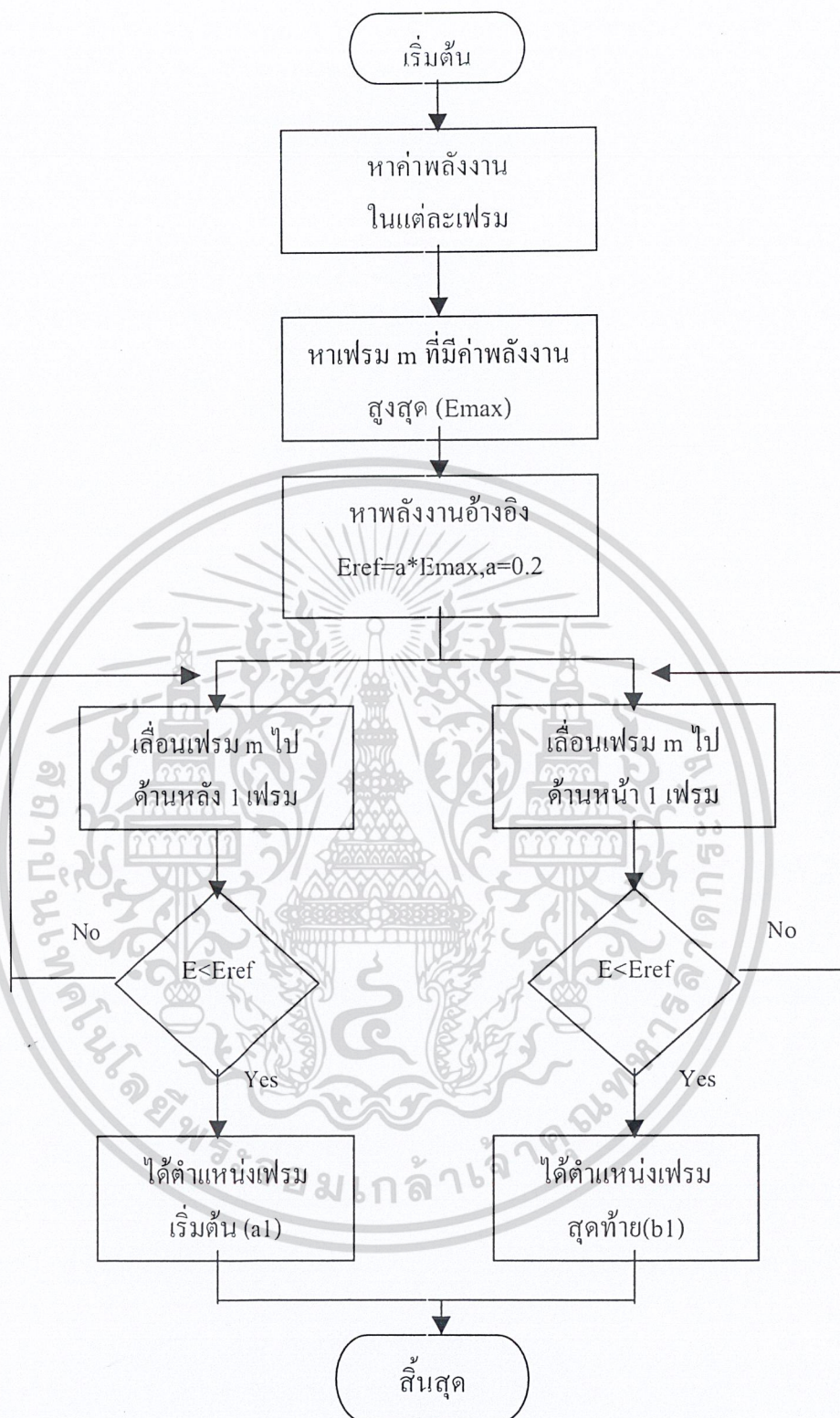
พลังงานอ้างอิงนี้จะเป็นตัวบอกว่า จุดเริ่มต้นและจุดสุดท้ายของค่าจะอยู่นอกช่วงนี้

4. ทำการหาจุดเริ่มต้นของค่า โดยจะพิจารณาเฟรมที่มีค่าพลังงานสูงสุดเป็นเฟรมแรก (เฟรมที่ m) ซึ่งนำค่าพลังงานของสัญญาณ เฟรมที่ m ไปเปรียบเทียบกับพลังงานอ้างอิง แล้วเลื่อนเฟรมที่จะเปรียบเทียบพลังงานกับพลังงานอ้างอิงไปด้านหลังเรื่อย ๆ (เฟรมที่ $m-1, m-2, \dots, 2, 1$) จนกว่าจะพบเฟรมที่มีค่าพลังงานน้อยกว่าค่าพลังงานอ้างอิงเป็นเฟรมแรก ก็จะได้จุดเริ่มต้นของสัญญาณ (a1)

5. การหาจุดสุดท้ายของค่า จะทำการพิจารณาเช่นเดียวกันกับการหาจุดเริ่มต้นของค่า คือเริ่มเปรียบเทียบจากเฟรมที่มีพลังงานสูงสุด (เฟรมที่ m) เช่นกัน แต่การเลื่อนเฟรมจะเลื่อนไปทางด้านหน้าแทน (เฟรมที่ $m+1, m+2, \dots$) จนกว่าจะพบเฟรมที่มีค่าพลังงานน้อยกว่าค่าพลังงานอ้างอิง ก็จะได้จุดสุดท้ายของสัญญาณ (b1)

จากขั้นตอนทั้งหมดที่กล่าวมานี้สามารถพิจารณาดังรูปที่ 3.2





รูปที่ 3.2 แสดงผังงานการหาจุดเริ่มต้นและจุดสุดท้ายของสัญญาณเสียง
โดยการเปรียบเทียบพลังงาน

3.1.2 การวิเคราะห์หาค่าอัตราการตัดศูนย์

เนื่องจากการใช้การวิเคราะห์โดยการหาค่าพลังงานของเสียงอาจเกิดความผิดพลาด จึงนำเอากการวิเคราะห์หาค่าอัตราการตัดศูนย์มาร่วมพิจารณา โดยมีวิธีการดังนี้

1. คำนวณหาค่าอัตราการตัดศูนย์ในช่วงเวลาที่ถูบบันทึกทั้งหมด โดยการคำนวณหาค่าอัตราการตัดศูนย์จะคำนวณเป็นเฟรม กำหนดให้แต่ละเฟรมมีตัวอย่าง 100 ตัวอย่าง และทำการคำนวณจนครบช่วงเวลาของเสียงที่ถูบบันทึกทั้งหมด

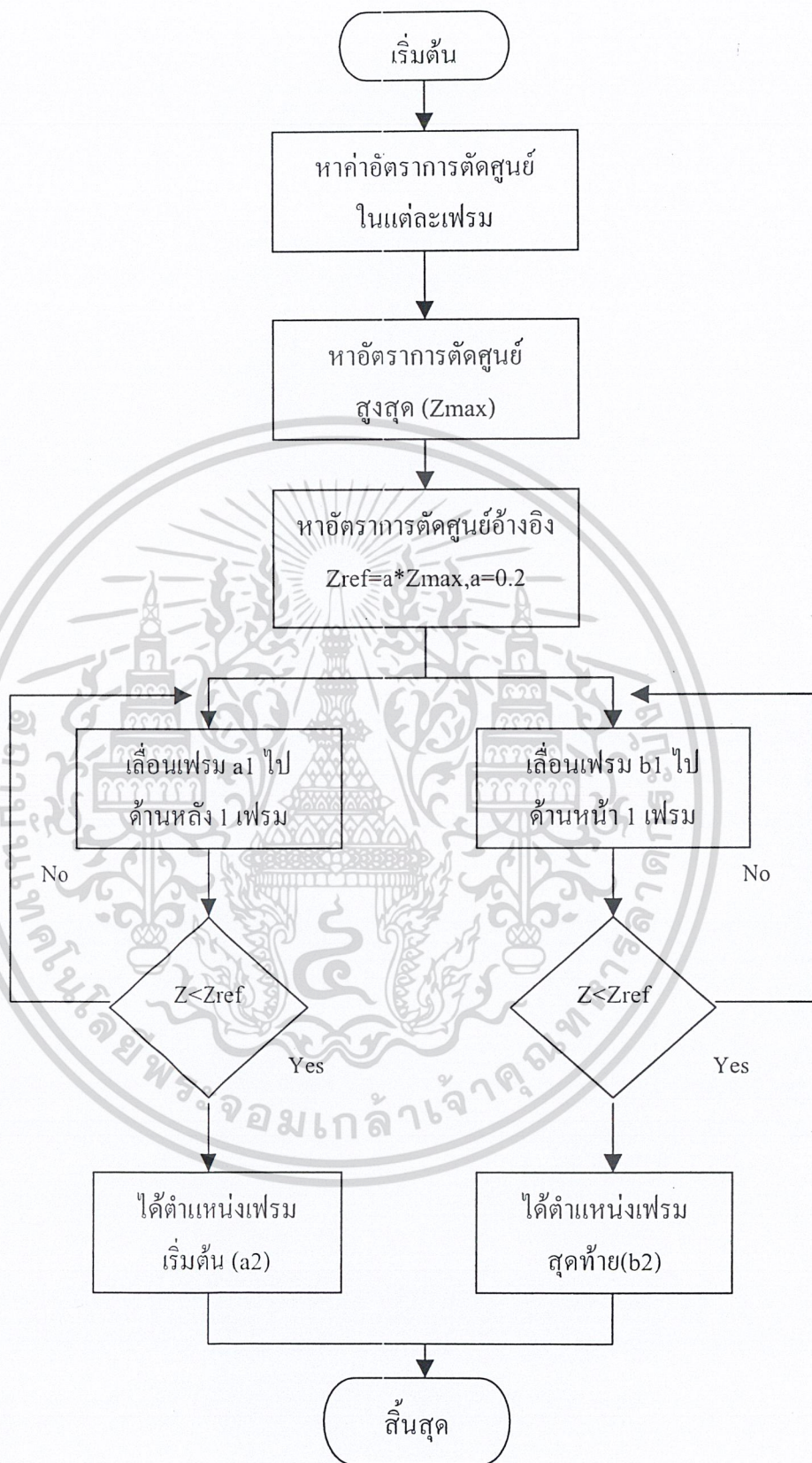
2. กำหนดระดับอ้างอิง (Z_{max}) กำหนดให้เป็นเฟรมที่ p

3. กำหนดค่าอัตราการตัดศูนย์ (Z_{ref}) ซึ่งมีค่าเป็น a เท่าของอัตราการตัดศูนย์สูงสุด ($a * Z_{ref}$) โดยค่า a จะเท่ากับ 0.2 เช่นกัน

4. จากจุดเริ่มต้น (a_1) ที่หาได้จากขั้นตอนที่ 1 จะเป็นเฟรมแรกที่จะพิจารณาโดยจะทำการเปรียบเทียบอัตราการตัดศูนย์ของเฟรมดังกล่าวกับอัตราการตัดศูนย์อ้างอิง ถ้ามีค่ามากกว่า จุดเริ่มต้นจะถูกเปลี่ยนตำแหน่ง โดยจะเลื่อนไปยังจุดแรกที่มีค่าน้อยกว่าอัตราการตัดศูนย์อ้างอิง (a_2) และจะถือว่าจุดนี้เป็นจุดเริ่มต้นที่แท้จริง ซึ่งการเลื่อนเฟรมจะเลื่อนไปด้านหลัง (เฟรมที่ $a_1-1, a_1-2, \dots, 2, 1$) แต่ในกรณีที่ a_1 น้อยกว่าอัตราการตัดศูนย์อ้างอิง จะถือว่า a_1 ที่หาได้จากขั้นตอนที่ 1 เป็นจุดเริ่มต้นที่ถูกต้องอยู่แล้ว

5. การหาจุดสุดท้าย จากจุดสุดท้าย (b_2) ที่ได้จากขั้นตอนที่ 1 จะพิจารณาเหมือนข้อ 3 คือจะเป็นเฟรมแรกที่จะเปรียบเทียบอัตราการตัดศูนย์อ้างอิง ถ้ามีค่ามากกว่า จุดสุดท้ายจะถูกเปลี่ยนตำแหน่งโดยจะเลื่อนเฟรมไปทางด้านหน้า (เฟรมที่ b_1+1, b_1+2, \dots) จนกว่าจะเจอจุดแรกที่มีค่าน้อยกว่าอัตราการตัดศูนย์อ้างอิง (b_2) และจะถือว่าจุดนี้เป็นจุดสุดท้ายที่แท้จริง แต่ถ้าอัตราการตัดศูนย์ที่ b_1 น้อยกว่าอัตราการตัดศูนย์อ้างอิงก็กล่าวได้ว่า b_1 เป็นจุดสุดท้ายที่ถูกต้องอยู่แล้วเช่นกัน

จากขั้นตอนทั้งหมดที่กล่าวมานี้สามารถพิจารณารูป 3.3

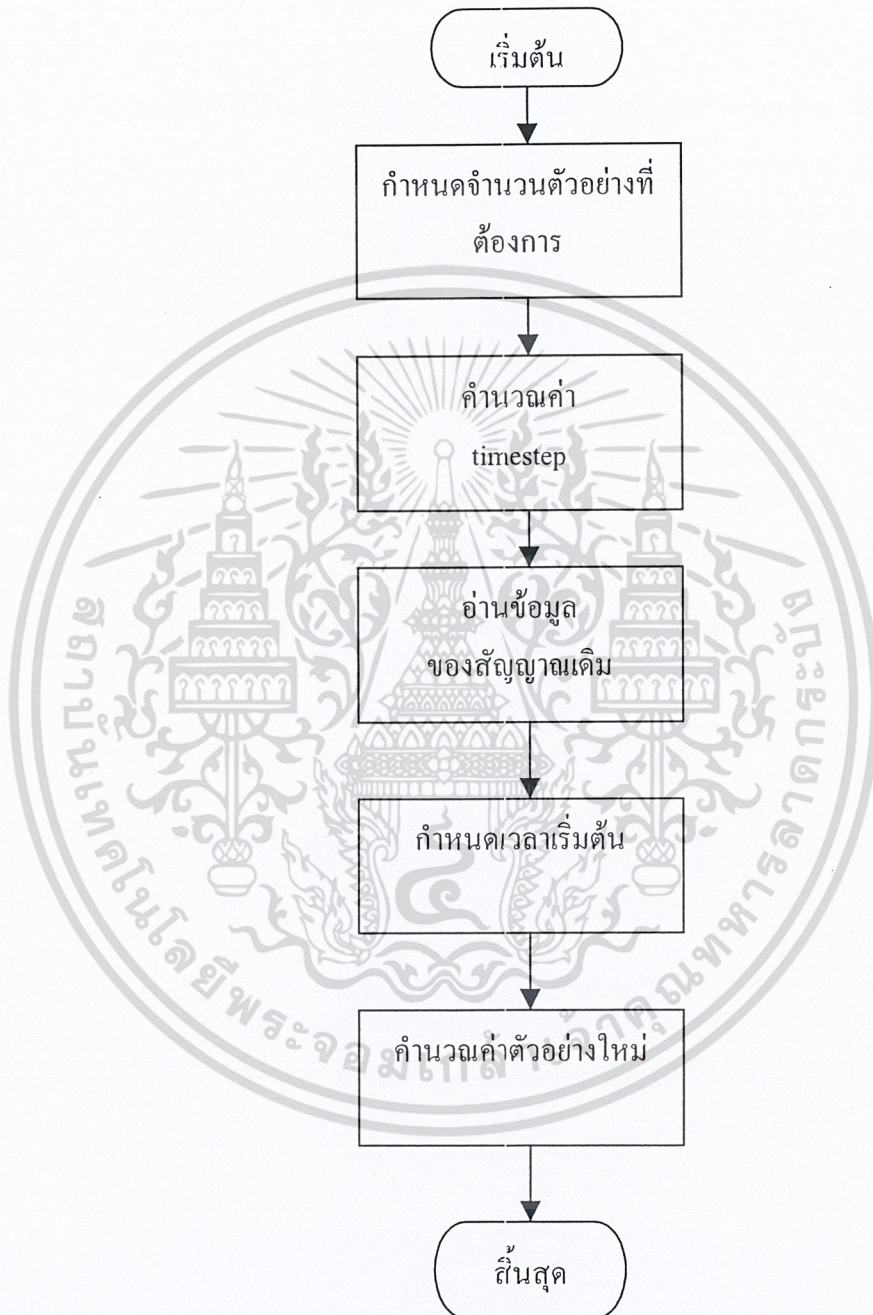


รูปที่ 3.3 แสดงผังงานการหาจุดเริ่มต้นและจุดสุดท้ายของสัญญาณเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3 การนอร์มอลไลซ์

เนื่องจากสัญญาณเสียงพูดแต่ละคำมีความยาวไม่เท่ากันจึงต้องมีการนอร์มอลไลซ์สัญญาณเสียงพูดให้ยาวเท่ากัน ขั้นตอนการนอร์มอลไลซ์สัญญาณเสียงพูดแสดงดังรูปที่ 3.4



รูปที่ 3.4 แสดงขั้นตอนการนอร์มอลไลซ์สัญญาณ

3.2 ภาคการวิเคราะห์สัญญาณเบื้องต้น

3.2.1 การหาค่าพารามิเตอร์

มีการกำหนดค่าต่าง ๆ ที่ใช้ในการคำนวณเพื่อทำการออกแบบ โปรแกรม ดังนี้

1. การพรีเอมฟาซิส

ใช้วงจรอันดับหนึ่ง มีฟังก์ชันถ่ายโอนเป็น

$$H(z) = 1 - az^{-1} \quad ; \quad 0.9 \leq a \leq 1.0 \quad (3.1)$$

ให้สัญญาณที่เข้ามาเป็น $s(n)$ และสัญญาณที่ผ่านการพรีเอมฟาซิสแล้วเป็น $s'(n)$ สามารถเขียนความสัมพันธ์ของทั้งสองสัญญาณโดยสมการ

$$s'(n) = s(n) - as(n-1) \quad (3.2)$$

ยิ่งค่า a เข้าใกล้ 1 เท่าใด ความถี่สูงจะถูกขยายมากขึ้นเท่านั้น ค่า a ที่นิยมสำหรับใช้ในการหาพารามิเตอร์แอลพีซี คือค่า $15/16 = 0.9375$

2. การแบ่งช่วงสัญญาณ

จากเงื่อนไขของการกำหนดขนาดของแต่ละช่วงสัญญาณนั้น ทำให้ค่า M ต้องมีค่าน้อยกว่าค่า N ดังนั้นจึงเลือกค่า N เป็น 240 ตัวอย่าง และ ค่า M เป็น 80 ตัวอย่าง โดยการแบ่งช่วงสัญญาณนี้กำหนดให้อันดับของช่วงเป็น l^{th} สัญญาณในแต่ละเฟรมเป็น $x_l(n)$ และมีจำนวนเฟรมทั้งหมดเป็น L เฟรมต่อสัญญาณเสียง 1 สัญญาณ เขียนเป็นสมการดังนี้

$$x_l(n) = s'(Ml + n) \quad n = 0, 1, \dots, N-1 \quad l = 0, 1, \dots, L-1 \quad (3.3)$$

ในการทดลองนี้ทำการกำหนดให้มีการสุ่มตัวอย่างของสัญญาณที่ 22050 เฮิรตซ์ ดังนั้นช่วงเวลาในการวิเคราะห์ของแต่ละเฟรม คือ $240/22050 = 10$ ms และเวลาที่ใช้ในการเลื่อนไปในแต่ละเฟรมคือ $80/22050 = 3.6$ ms

3. การวินโดว์

จากรูปที่ 3. 5 แสดงถึงความไม่ต่อเนื่องตรงขอบของเฟรมที่ผ่านการแบ่งช่วงสัญญาณแล้วดังนั้นจึงต้องทำการลดความไม่ต่อเนื่องดังกล่าวด้วยฟังก์ชันวินโดว์แฮมมิง ซึ่งนิยามโดยสมการ

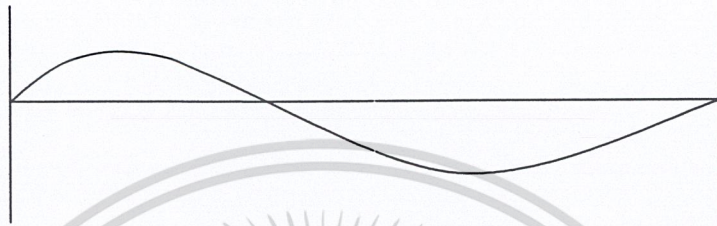
$$w(n) = 0.54 - 0.46\cos(2\pi n/N - 1) \quad 0 \leq n \leq N-1 \quad (3.4)$$

ซึ่งฟังก์ชันวินโดว์แฮมมิงนี้มีคุณลักษณะของสเปกตรัมที่เหมาะสมในการวิเคราะห์เสียงดังนี้

- ความถี่รีโซลูชันสูง (High frequency resolution) หมายถึงมีโลบหลัก (Main lobe) แคบ
- การลดทอน (Attenuation) นอกช่วงความถี่ผ่านต่ำ หมายถึง ไซด์โลบ (Side lobe) มีค่าน้อย

ถ้ากำหนดค้ำวินโดว์เป็น $w(n)$, $0 \leq n \leq N-1$ แล้วสัญญาณที่ผ่านการวินโดว์จะได้

$$x_1'(n) = x_1(n)w(n) \quad 0 \leq n \leq N-1 \quad (3.5)$$



รูปที่ 3.5 แสดงส่วนสัญญาณที่ตัดมาวิเคราะห์

4. การวิเคราะห์หาคคุณลักษณะของเสียง

การหาคคุณลักษณะของเสียง ซึ่งเป็นขั้นตอนการหาค่าพารามิเตอร์ โดยจะใช้วิธีลีนีเยร์พรีดิกทีฟ (Linear Predictive Coding) การวิเคราะห์แบบลีนีเยร์พรีดิกทีฟนี้เป็นการวิเคราะห์เสียงพูดที่วิธีหนึ่ง ในการใช้หาค่าพารามิเตอร์พื้นฐานของเสียง ซึ่งพารามิเตอร์ที่ได้เหล่านี้มีความถูกต้องและให้เวลาในการคำนวณไม่มากนัก โดยที่เราจะต้องเตรียมสัญญาณในการวิเคราะห์ก่อน แล้วจึงนำสัญญาณนี้ไปใช้ในวิธีลีนีเยร์พรีดิกทีฟ ขั้นตอนในการวิเคราะห์สามารถแบ่งได้ดังนี้

- การหาออโตคอร์รีเลชัน

สัญญาณในแต่ละเฟรมที่ผ่านวินโดว์แล้ว จะถูกผ่านการเข้าออโตคอร์รีเลชัน

$$r_m = \sum_{n=0}^{N-1-m} x'(n)x'(n+m) \quad m = 0,1,\dots,p \quad (3.6)$$

เรียกวิธีการนี้ว่าการประมาณเชิงเส้นอันดับ p โดยเรียกอันดับ p ว่า อันดับของการวิเคราะห์แอลพีซี โดยทั่วไปค่า p จะมีค่าระหว่าง 8 ถึง 16

เนื่องจากการวิเคราะห์โดยวิธีออโตคอร์รีเลชัน อันดับ p ที่มากจะทำให้การประมาณเสียงมีความใกล้เคียงมากยิ่งขึ้น แต่ถ้าอันดับ p มีค่ามากเกินไปจะทำให้การคำนวณมีความยุ่งยากและใช้เวลานาน ดังนั้นเพื่อความเหมาะสมค่าอันดับ p ที่ใช้คือ 12

- การหาพารามิเตอร์แอลพีซี (a)

วิธีการวิเคราะห์หาค่าพารามิเตอร์แอลพีซี ที่นิยมใช้กันได้รับการพัฒนาโดยเลวินสัน (Levinson) และ เดอร์บิน (Durbin) ซึ่งเรียกว่า วิธีการย้อนกลับของเลวินสัน-เดอร์บิน (Levinson-Durbin recursive

method) มีขั้นตอนดังนี้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$E^{(0)} = r^{(0)} \tag{3.7}$$

$$k_i = \left\{ r^{(i)} - \sum_{j=1}^{L-1} \alpha_j^{(i-1)} r^{(i-j)} \right\} / E^{(i-1)}, \quad 1 \leq i \leq p \tag{3.8}$$

$$\alpha_i^1 = k_i \tag{3.9}$$

$$\alpha_j^i = \alpha_j^{(i-1)} - k_i \alpha_{(i-j)}^{(i-1)} \quad 1 \leq j \leq i-1 \tag{3.10}$$

$$E^i = (1 - k_i^2) E^{(i-1)} \tag{3.11}$$

สมการที่ (3.7) – (3.11) จะถูกคำนวณแบบย้อนกลับ สำหรับ $i = 1, 2, \dots, p$ และผลลัพธ์สุดท้ายจะ
ได้พารามิเตอร์ $\{a_m(k), 1 \leq k \leq m\}$ ก็คือค่าพารามิเตอร์แอลพีซี หรือ α_j^i ที่อันดับ p ซึ่งผลลัพธ์สุดท้าย
คือ $\{a_m(k), 1 \leq k \leq p\}$ โดยที่ p คือ อันดับของพารามิเตอร์ที่ต้องการ

- สัมประสิทธิ์เซปสตรัมสามารถหาได้โดยตรงจากพารามิเตอร์แอลพีซี ดังนี้

$$C_0 = \ln G \tag{3.12}$$

$$C_m = a_m + \sum_{k=1}^{m-1} \left(\frac{k}{m} \right) C_k a_{m-k}, \quad 1 \leq m \leq p \tag{3.13}$$

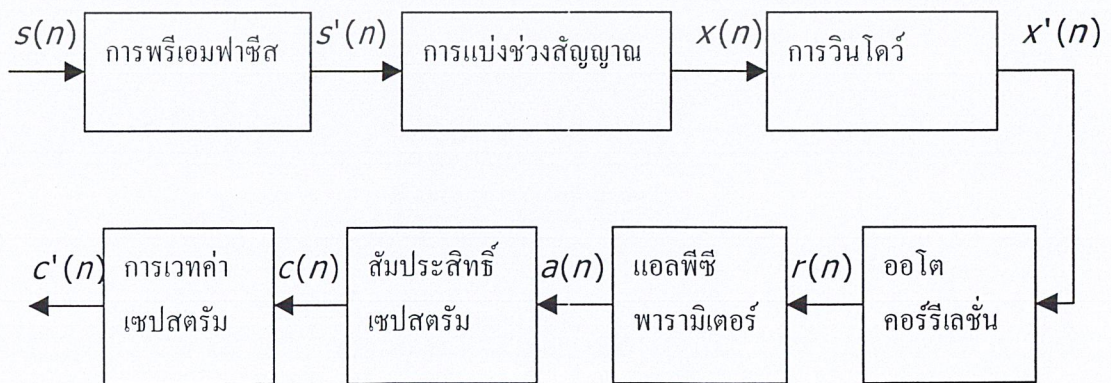
$$C_m = \sum_{k=1}^{m-1} \left(\frac{k}{m} \right) C_k a_{m-k}, \quad m > p \tag{3.14}$$

$$Q \approx \left(\frac{3}{2} \right)^P P \tag{3.15}$$

ทำการเวท (weight) หรือถ่วงน้ำหนักสัมประสิทธิ์เซปสตรัมที่ได้ เพื่อลดความคลาดเคลื่อนเหล่านี้นั้นด้วยฟังก์ชันเวทดัง จะได้พารามิเตอร์สุดท้ายคือ

$$C_m' = C_m * W_m \tag{3.16}$$

ขั้นตอนในการวิเคราะห์สัญญาณเสียงเบื้องต้นที่กล่าวมาทั้งหมด สามารถเขียนเป็นบล็อกไดอะแกรมได้ดังนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 3.6 บล็อกไดอะแกรมการวิเคราะห์เสียงเบื้องต้น นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 การสร้างโค้ดบุ๊ก

จากการทดสอบ (L.R.Rabiner.S.E.Levinson และ Sonthi.1982.) จะได้ว่าที่ขนาดโค้ดบุ๊กเท่ากับ n จะมีค่าความคลาดเคลื่อน โดยวิธีการคำนวณแบบสแควเออร์ดิสทอนซ์ (Square error distortion) ในการหาระยะทางเนื่องจากเป็นวิธีที่ง่ายและรวดเร็ว

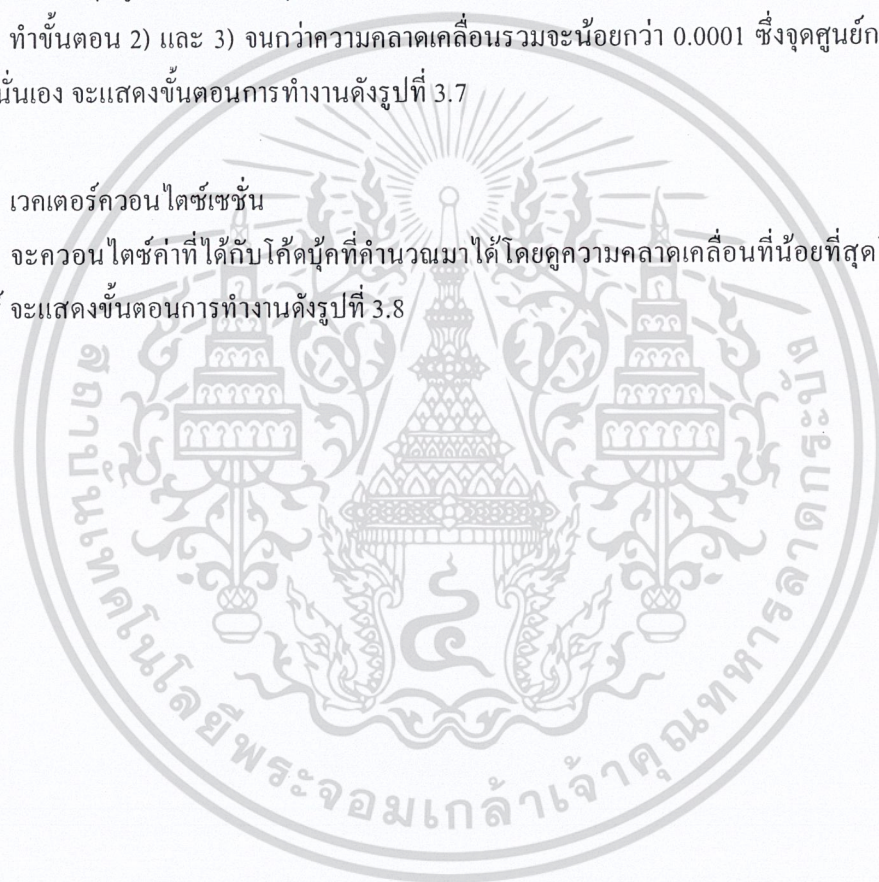
การสร้างโค้ดบุ๊กโดยนำเทรนนิงเซตที่ได้จากการประมาณเชิงเส้นมาผ่านกระบวนการดังนี้

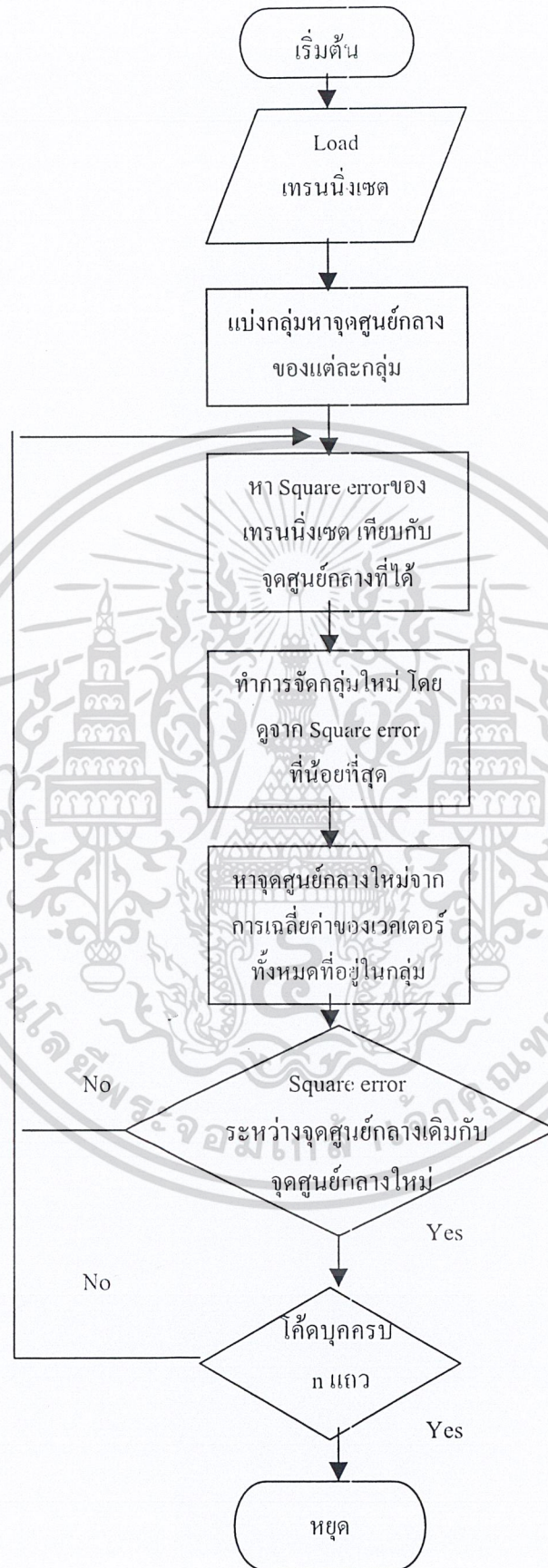
- 1) สุ่มค่าโค้ดบุ๊กเริ่มต้นมา n ตัว ตัวละ q มิติ
- 2) หาระยะทางระหว่างโค้ดบุ๊กกับเทรนนิงเซตแต่ละตัวโดยใช้ความคลาดเคลื่อนกำลังสอง
- 3) จัดกลุ่มของเวกเตอร์อินพุทโดยพิจารณาจากระยะทางที่น้อยที่สุด
- 4) หาจุดศูนย์กลางของกลุ่ม

ทำขั้นตอน 2) และ 3) จนกว่าความคลาดเคลื่อนรวมจะน้อยกว่า 0.0001 ซึ่งจุดศูนย์กลางที่ได้ก็คือโค้ดบุ๊กนั่นเอง จะแสดงขั้นตอนการทำงานดังรูปที่ 3.7

เวกเตอร์ควอนไทซ์เซชัน

จะควอนไทซ์ค่าที่ได้กับโค้ดบุ๊กที่คำนวณมาได้ โดยดูความคลาดเคลื่อนที่น้อยที่สุดโดยวิธีสแควเออร์ จะแสดงขั้นตอนการทำงานดังรูปที่ 3.8





รูปที่ 3.7 แสดงขั้นตอนการสร้างโค้ดบุค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานในเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ภาครูปแบบอ้างอิง

คือส่วนที่เก็บค่าพารามิเตอร์ของสัญญาณ (พารามิเตอร์แอลพีซี) ที่ใช้เป็นสัญญาณแบบอ้างอิง ในที่นี้คือสัญญาณเลข 0-9 กล่าวคือเมื่อมีสัญญาณเสียงทดสอบเข้ามาและผ่านภาคการหาขอบเขตของคำ และหาค่าพารามิเตอร์ของสัญญาณแล้ว ก็นำมาเปรียบเทียบความคล้ายคลึงโดยวิธีฮิดเดนมาร์คอฟโมเดล กับสัญญาณอ้างอิง 0-9 ในภาคการเปรียบเทียบต่อไป

3.4 ภาคการเปรียบเทียบ

คือขั้นตอนการเปรียบเทียบความคล้ายคลึง โดยการเปรียบเทียบระหว่างสัญญาณแบบทดสอบ โดยนำข้อมูลมาจากส่วน ฟูรีโพรเซสซิ่งกับพารามิเตอร์ของสัญญาณแบบอ้างอิง 0-9 ที่เก็บไว้ในภาคแบบอ้างอิงนั่นเอง ซึ่งในปริภูมิตฤษฎีนี้จะใช้วิธีของ ฮิดเดนมาร์คอฟโมเดลในขั้นตอนการเปรียบเทียบความคล้ายคลึงนี้

3.4.1 ฮิดเดนมาร์คอฟโมเดล

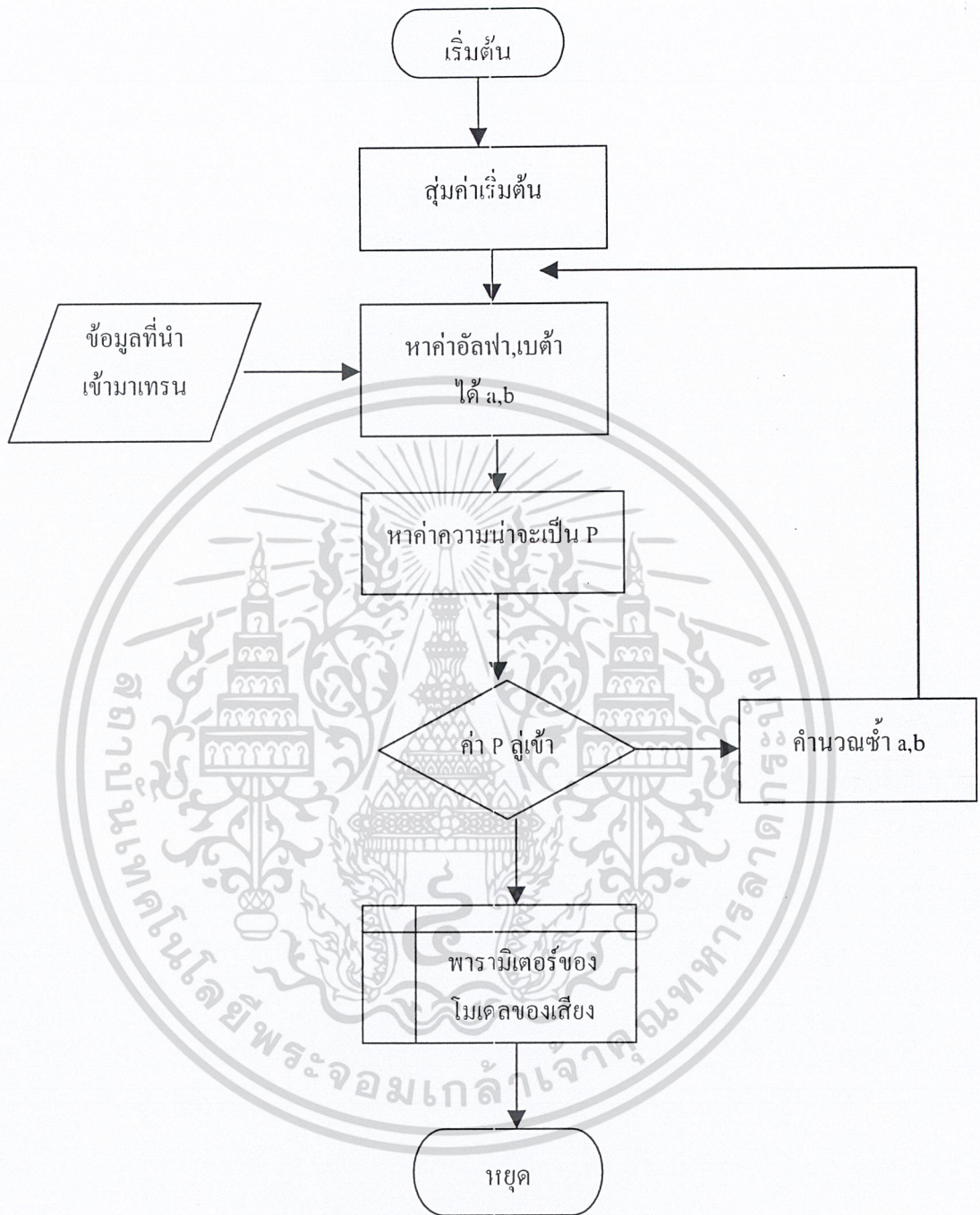
ในการสร้างแบบจำลอง และการรู้จำเสียงเพื่อเปรียบเทียบความคล้ายคลึงของสัญญาณเสียงแบบ HMM ของเสียงมีขั้นตอนดังต่อไปนี้

1. สุ่มค่าเริ่มต้น a, b และกำหนดให้ $\pi = [1 \ 0 \ 0 \ 0 \ 0]$ ตามเงื่อนไขแบบจำลองแบบ Left-Right
2. หาค่า α, β จากค่า a, b เริ่มต้น และลำดับค่าปรากฏ $O = \{O_1, O_2, \dots, O_T\}$ ซึ่งเรียกว่าลำดับเทรนนิ่ง ตามวิธีของ กระบวนการ ไปข้างหน้าและย้อนกลับ โดยใช้ลำดับของค่าปรากฏหลาย ๆ ลำดับเข้ามาเทรน เพื่อความถูกต้องมากขึ้น
3. หาค่าพารามิเตอร์ a, b, π ที่ให้ความน่าจะเป็นสูงสุดที่จะเป็นแบบจำลอง λ ที่เหมาะสม
4. ตรวจสอบค่าพารามิเตอร์แบบจำลองที่ได้ว่าลู่เข้าหรือยัง โดยใช้วิธีการคำนวณค่า a, b ซ้ำ โดยในที่นี้ตั้งไว้ว่าเมื่อไม่มีการเปลี่ยนแปลงหรือค่าเท่าเดิมให้หยุดการทำซ้ำ ก็จะได้ค่าพารามิเตอร์ a, b, π ของแบบจำลองที่ต้องการ
5. เก็บค่าพารามิเตอร์ a, b, π ที่ได้จากข้อ 4. เป็นพารามิเตอร์ของแบบจำลองไว้ และหลังจากที่ได้แบบจำลอง HMM ของแต่ละคำศัพท์แล้วเมื่อมีลำดับค่าปรากฏ $O = \{O_1, O_2, O_3, \dots, O_T\}$ ของเสียงที่ไม่ทราบ ซึ่งเป็นเสียงที่ต้องการทดสอบเข้ามา เราจะทำการคำนวณหาความน่าจะเป็น $P(O | \lambda)$ ทุกแบบจำลองของแต่ละคำศัพท์โดยใช้วิธี วิเทอริบี อัลกอริทึม แล้วเลือกเอาคำศัพท์ที่มีความน่าจะเป็นสูงสุดซึ่งก็คือ คำศัพท์ที่แบบจำลองจำได้นั่นเอง ดังแสดงในรูปที่ 3.9

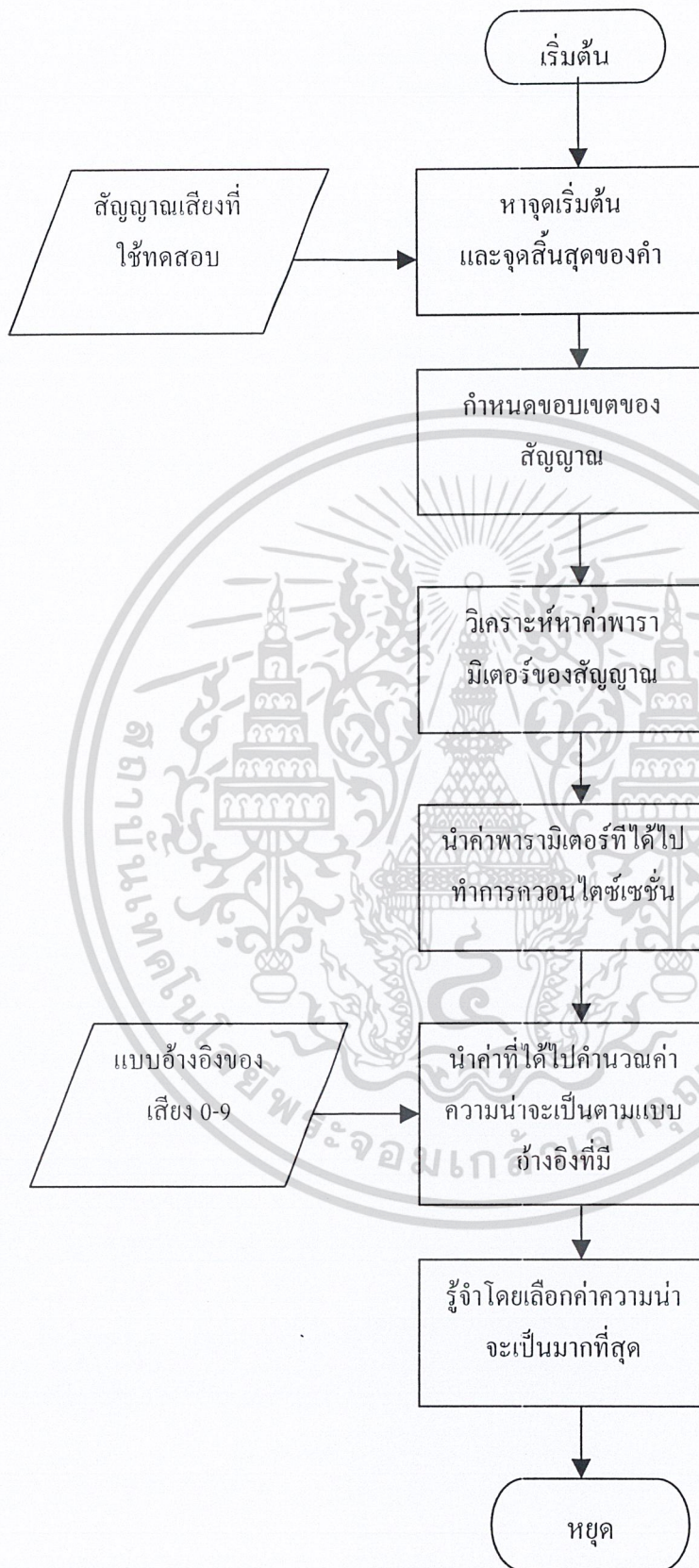
3.5 ภาคการตัดสินใจ

เป็นขั้นตอนการตัดสินใจว่าเสียงนั้นเป็นเสียงอะไร โดยเปรียบเทียบกับแบบอ้างอิง 0-9 หลังจากนั้นก็ทำการเลือกผลที่ได้จากการเปรียบเทียบในภาคที่แล้วที่เหมือนกันมากที่สุดตามกฎการตัดสินใจ นั่นคือการตัดสินใจในการจำแนกเสียงพูดว่าเสียงที่นำมาทดสอบเป็นเลขอะไรนั่นเอง ดังรูปที่ 3.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 แสดงขั้นตอนการสร้างแบบจำลองอ้างอิง



รูปที่ 3.10 แสดงขั้นตอนการรู้จำเสียงพูด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

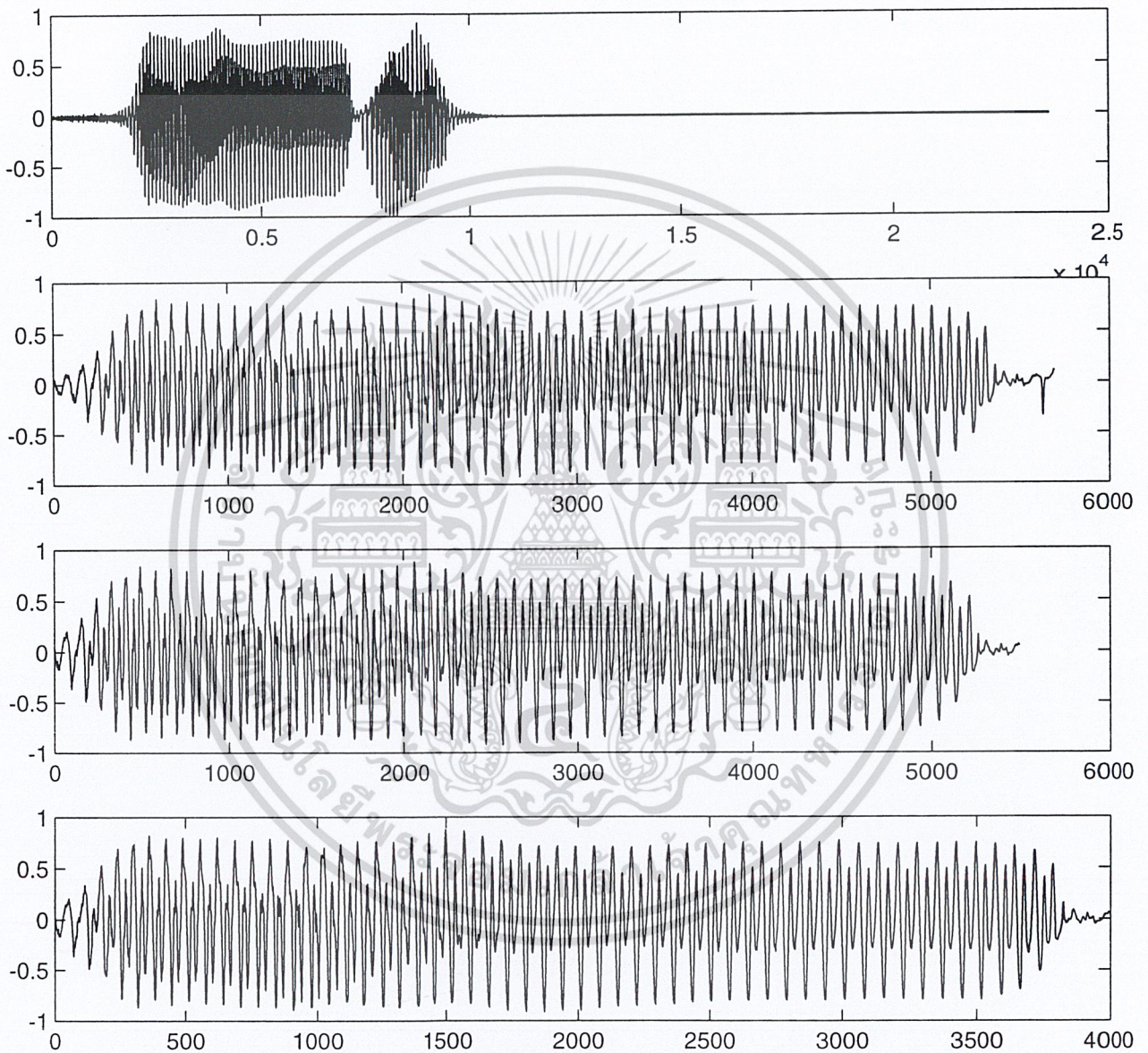
การทดลองและผลการทดลอง

4.1 การทดลอง

1. การหาขอบเขตของค่าโดยการหาค่าพลังงาน และการนอร์มอลไลซ์เสียง 0 – 9
2. การหาค่าพารามิเตอร์ ได้แก่ สัมประสิทธิ์แอลพีซี สัมประสิทธิ์เซปสตรัม และ สัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก
3. การหาโค้ดบุค 32 , 64 , 128 และแสดงผลรวมเวลาในการคำนวณ
4. การทดสอบการจดจำเสียงพูดในกรณีต่าง ๆ



4.1.1 การหาขอบเขตของคำ และการนอร์มอลไลซ์ ของเสียง 0 – 9



รูปที่ 4.1 แสดงตัวอย่างเสียง การหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขศูนย์ โดย

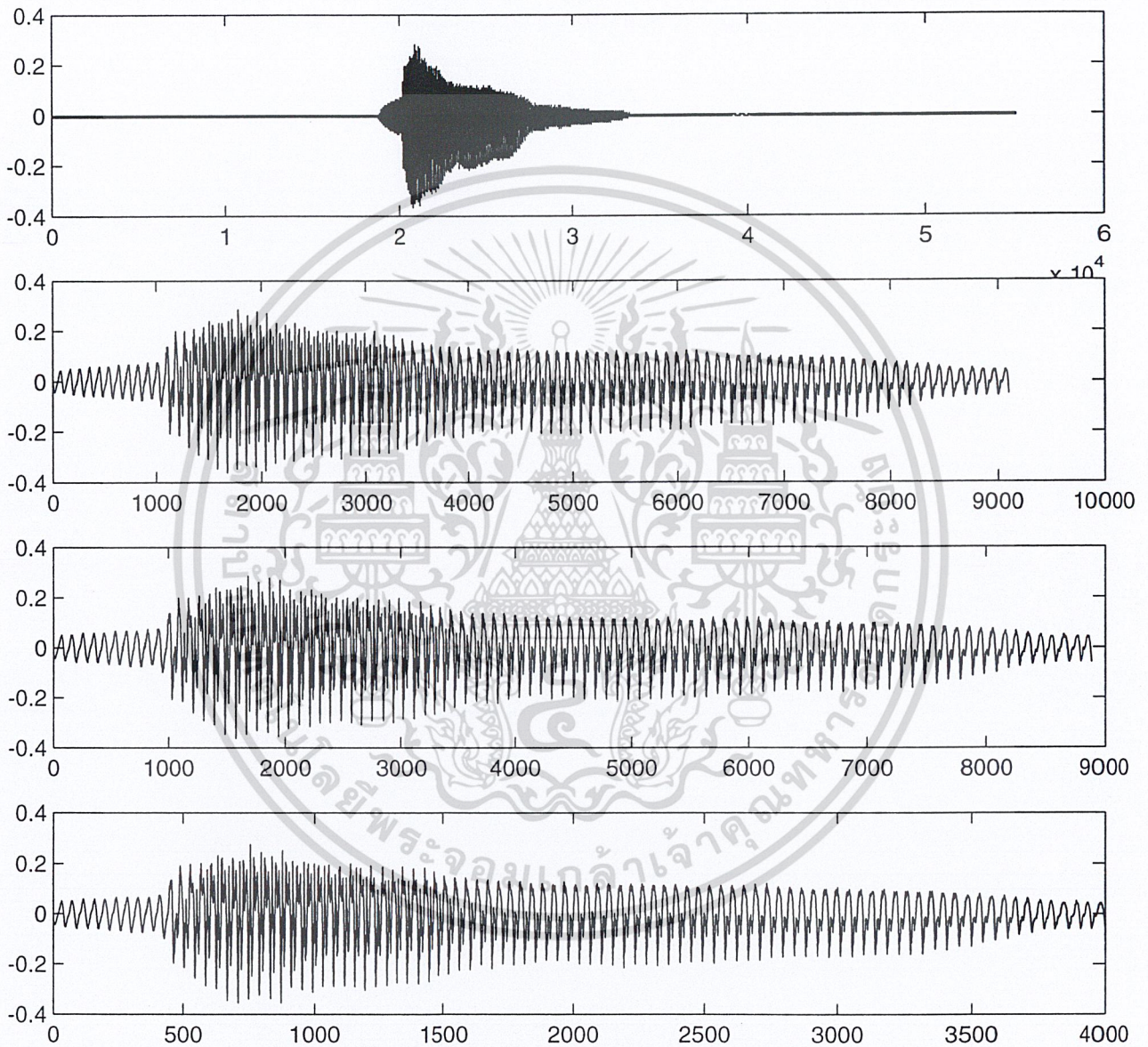
รูปที่ 1 แสดงตัวอย่างของเสียง

รูปที่ 2 แสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน

รูปที่ 3 แสดงการหาขอบเขตของคำโดยการหาค่าอัตราการตัดศูนย์

รูปที่ 4 แสดงการนอร์มอลไลซ์ให้มี 4000 แซมเปิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 แสดงตัวอย่างเสียง การหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขหนึ่ง โดย

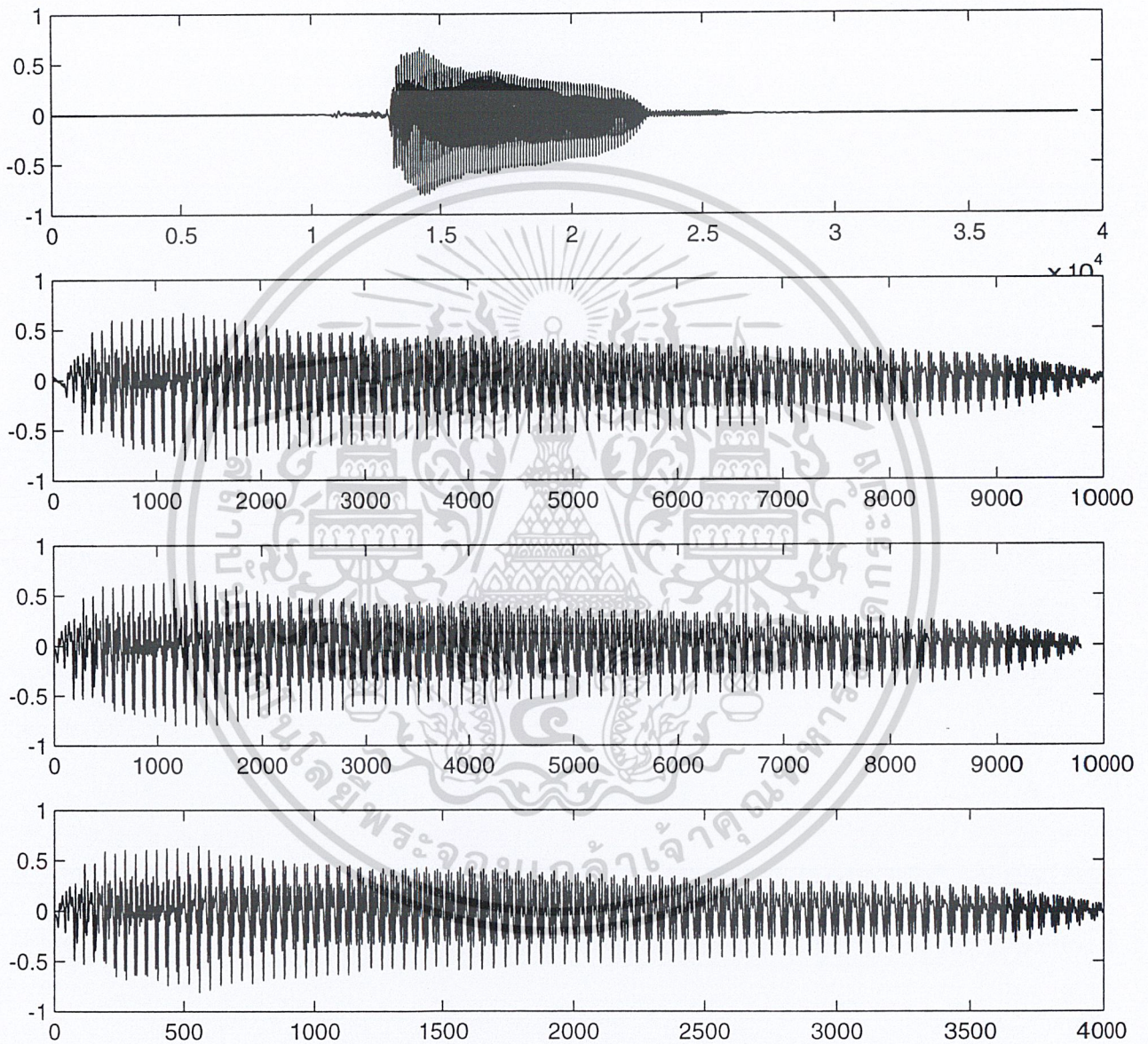
รูปที่ 1 แสดงตัวอย่างของเสียง

รูปที่ 2 แสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน

รูปที่ 3 แสดงการหาขอบเขตของคำโดยการหาค่าอัตราการตัดศูนย์

รูปที่ 4 แสดงการนอร์มอลไลซ์ให้มี 4000 แซมเปิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น มิใช่เพื่อให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 แสดงตัวอย่างเสียง การหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขสอง โดย

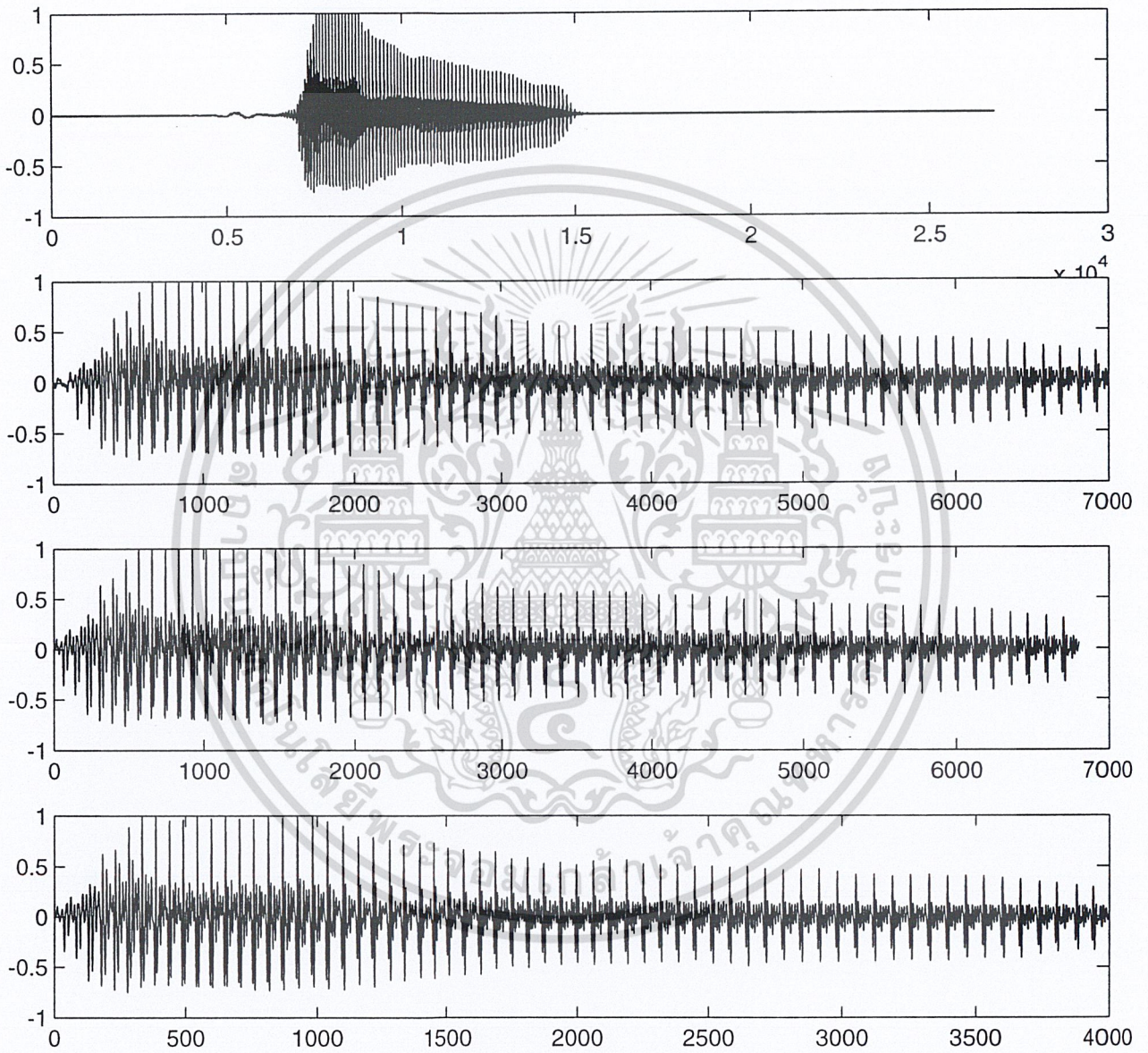
รูปที่ 1 แสดงตัวอย่างของเสียง

รูปที่ 1 แสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน

รูปที่ 2 แสดงการหาขอบเขตของคำโดยการหาค่าอัตราการตัดสินใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 3 แสดงการนอร์มอลไลซ์ให้มี 4000 แซมเปิล

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 แสดงตัวอย่างเสียง การหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขสาม โดย

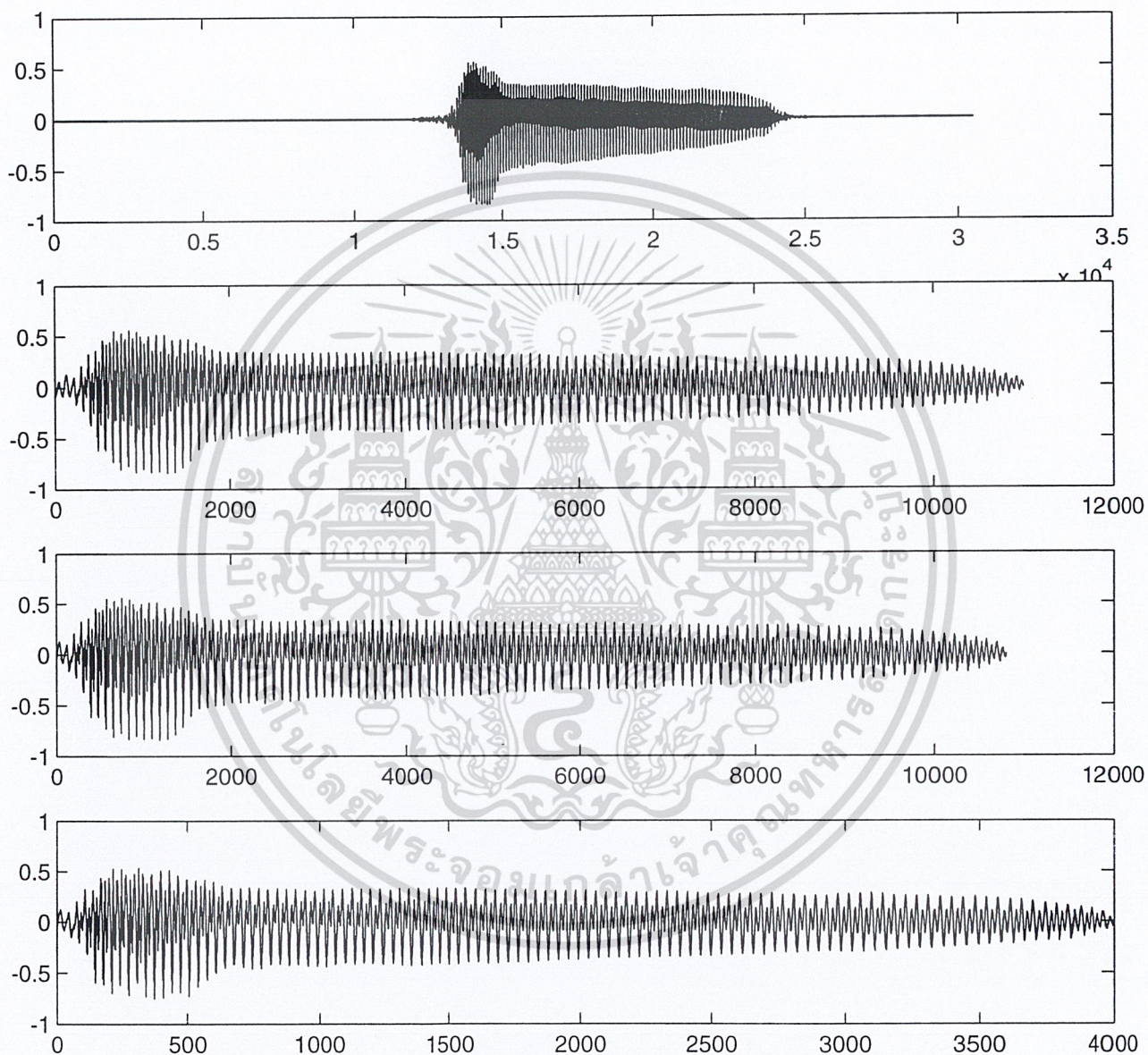
รูปที่ 1 แสดงตัวอย่างของเสียง

รูปที่ 2 แสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน

รูปที่ 3 แสดงการหาขอบเขตของคำโดยการหาค่าอัตราการตัดศูนย์

รูปที่ 4 แสดงการนอร์มอลไลซ์ให้มี 4000 แซมเปิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น มิใช่ผู้จัดทำให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 แสดงตัวอย่างเสียง การหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขสี่ โดย

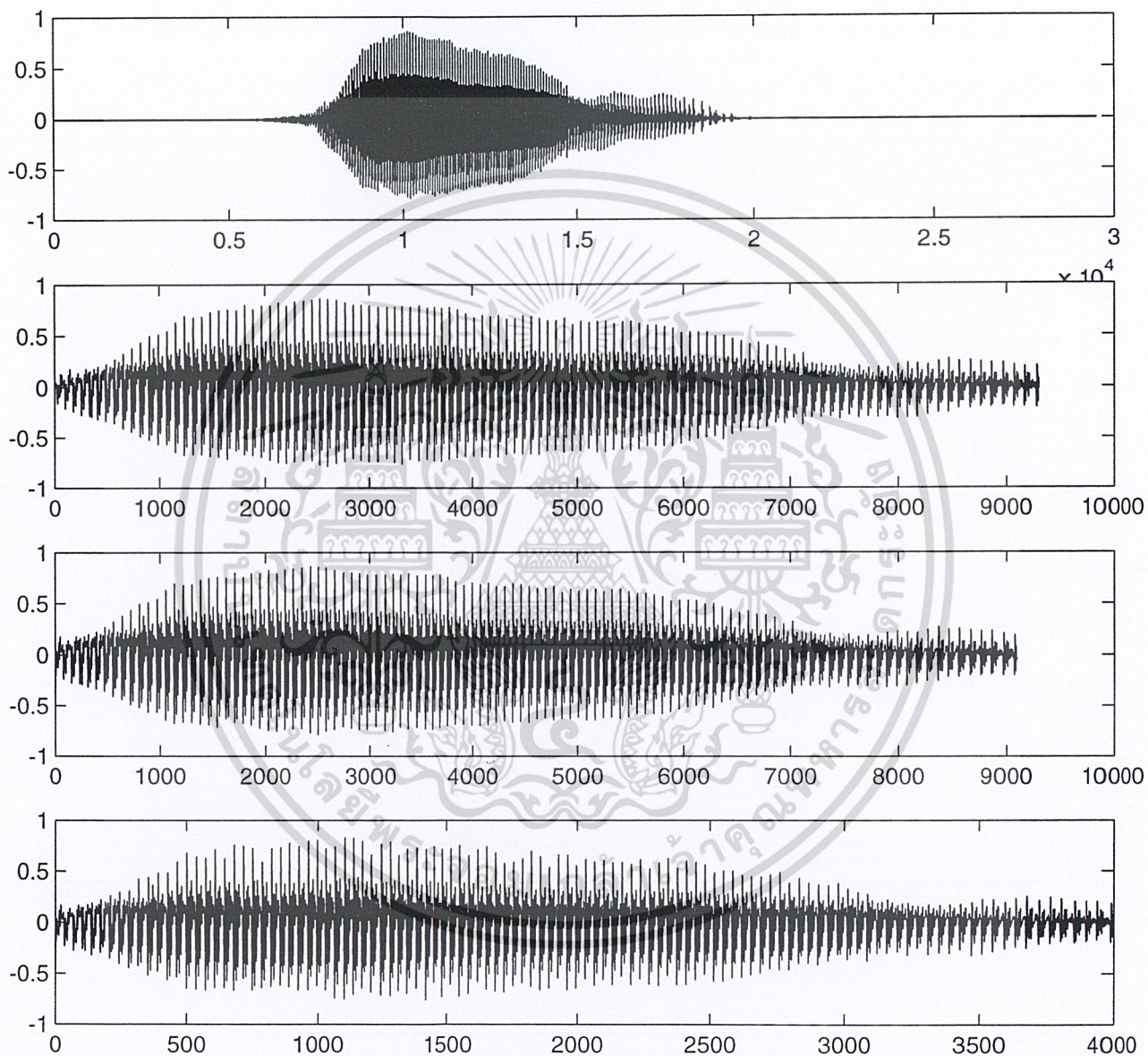
รูปที่ 1 แสดงตัวอย่างของเสียง

รูปที่ 2 แสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน

รูปที่ 3 แสดงการหาขอบเขตของคำโดยการหาค่าอัตราการตัดศูนย์

รูปที่ 4 แสดงการนอร์มอลไลซ์ให้มี 4000 แซมเปิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการวิจัยเท่านั้น เมื่อผู้ใดให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 แสดงตัวอย่างเสียง การหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขห้า โดย

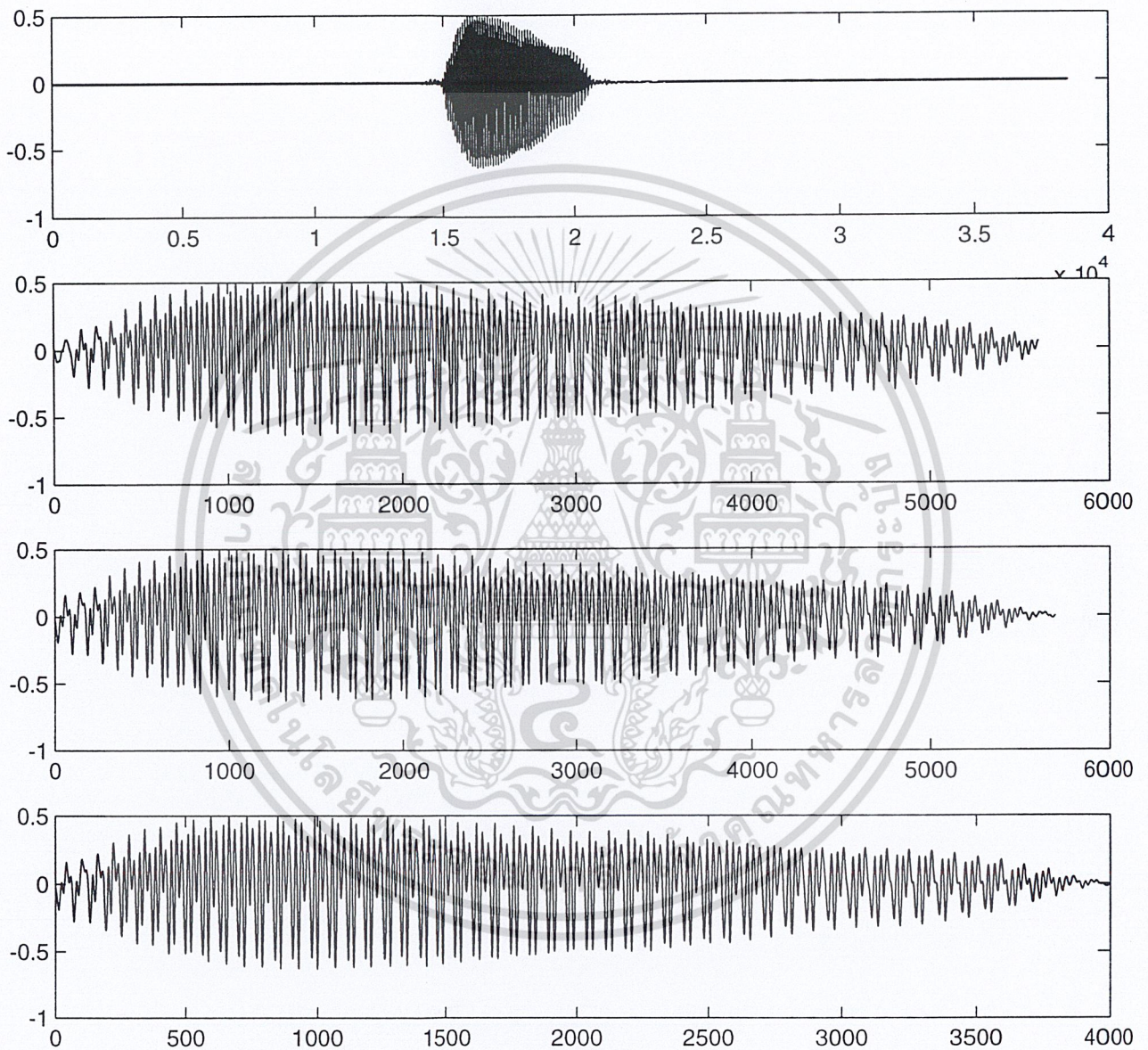
รูปที่ 1 แสดงตัวอย่างของเสียง

รูปที่ 2 แสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน

รูปที่ 3 แสดงการหาขอบเขตของคำโดยการหาค่าอัตราการตัดสินใจ

รูปที่ 4 แสดงการนอร์มอลไลซ์ให้มี 4000 แซมเปิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น เมื่อผู้ยืมได้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 แสดงตัวอย่างเสียง การหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขหกโดย

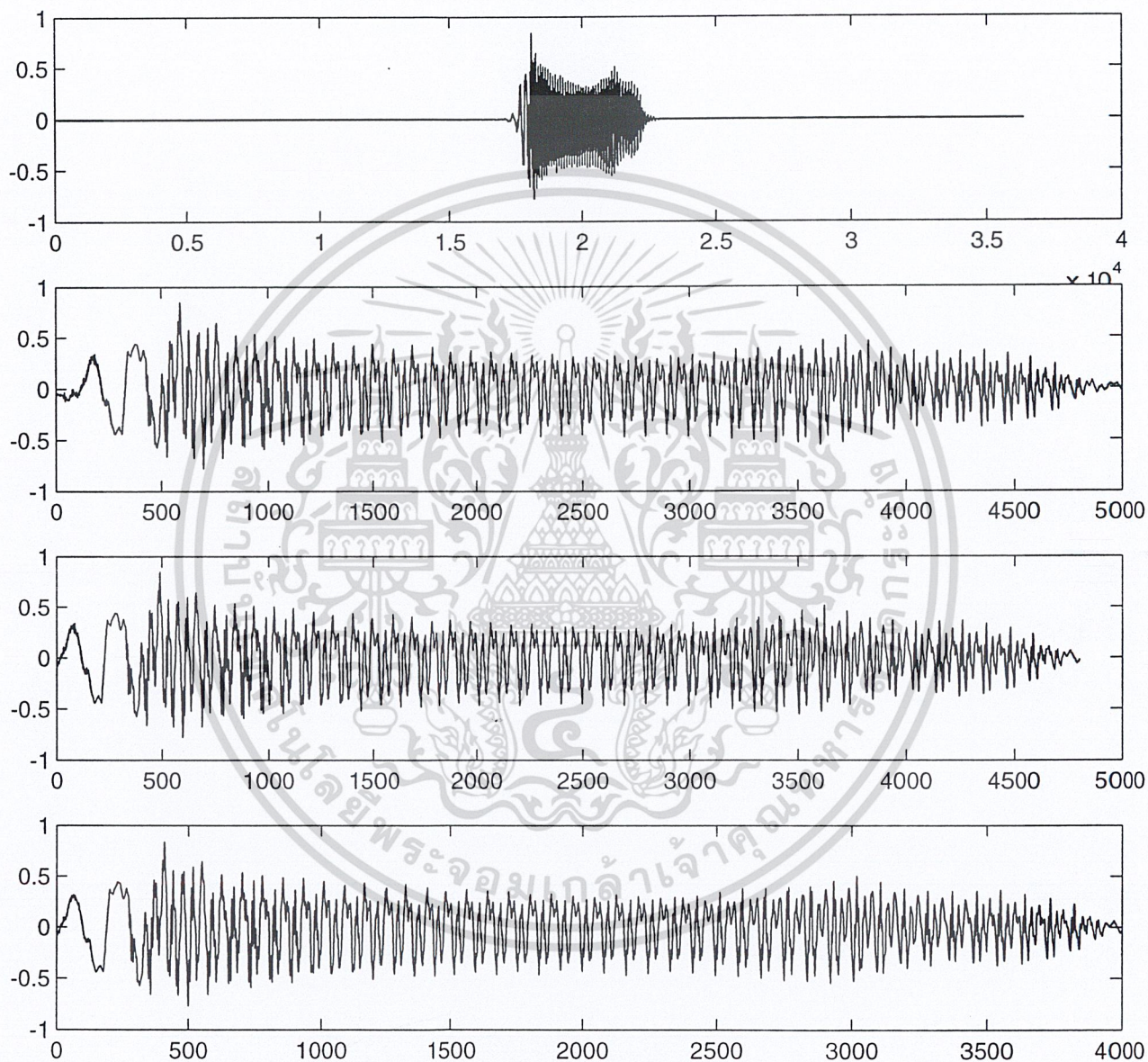
รูปที่ 1 แสดงตัวอย่างของเสียง

รูปที่ 2 แสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน

รูปที่ 3 แสดงการหาขอบเขตของคำโดยการหาค่าอัตราการตัดสินใจ

รูปที่ 4 แสดงการนอร์มอลไลซ์ให้มี 4000 แซมเปิลต่อนำไปใช้ประโยชน์ด้านการค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่ควรเผยแพร่หรือทำซ้ำโดยไม่ได้รับอนุญาต หากต้องการข้อมูลเพิ่มเติม กรุณาติดต่อฝ่ายประชาสัมพันธ์ โทร. 0-2253-8000 หรือ 0-2253-8001



รูปที่ 4.8 แสดงตัวอย่างเสียง การหาขอบเขตของค่าและการนอร์มอลไลซ์ของเลขเจ็ด โดย

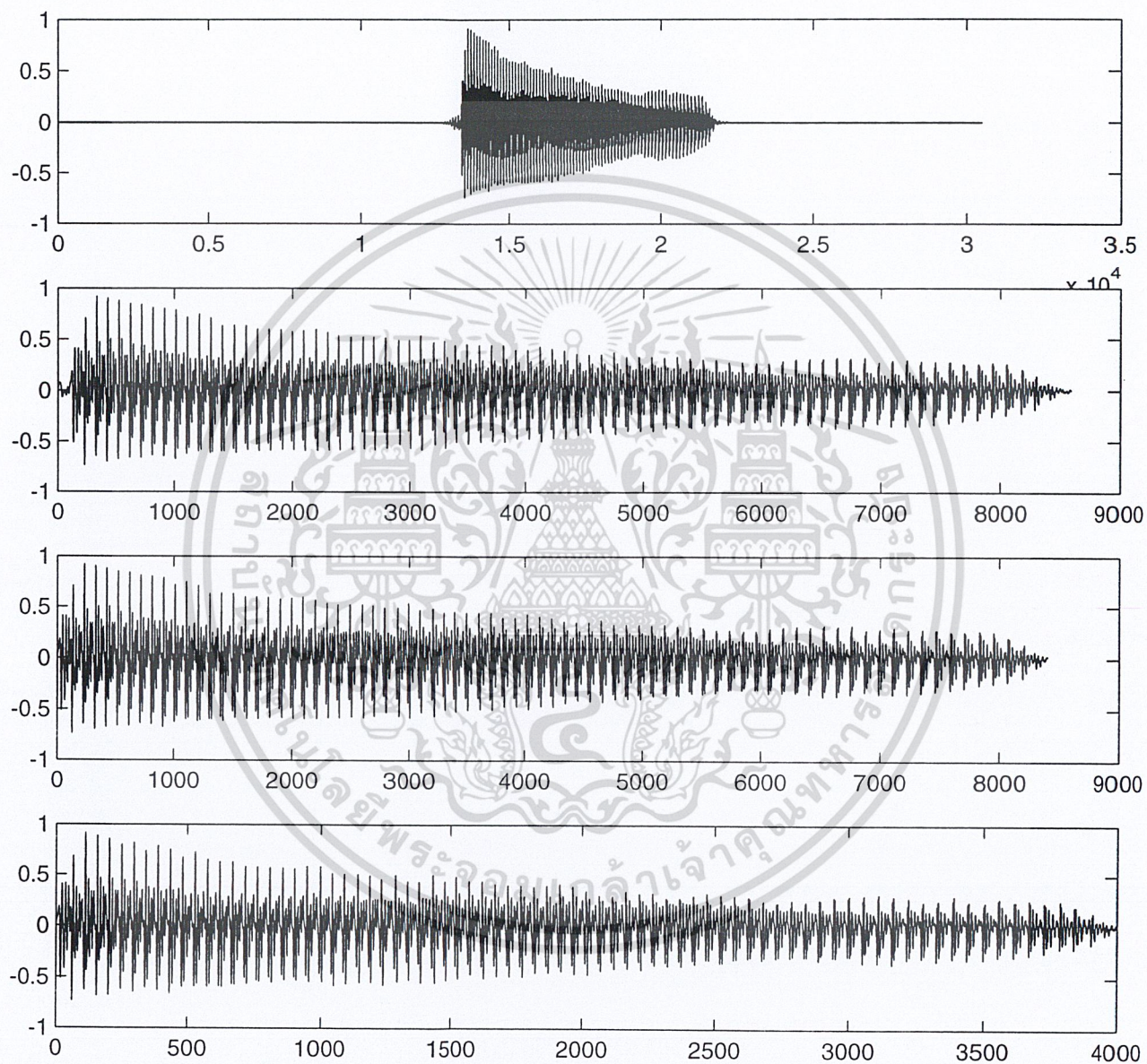
รูปที่ 1 แสดงตัวอย่างของเสียง

รูปที่ 2 แสดงการหาขอบเขตของค่าโดยการหาค่าพลังงาน

รูปที่ 3 แสดงการหาขอบเขตของค่าโดยการหาค่าอัตราการตัดสินใจ

รูปที่ 4 แสดงการนอร์มอลไลซ์ให้มี 4000 แซมเปิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 แสดงตัวอย่างเสียง การหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขแปด โดย

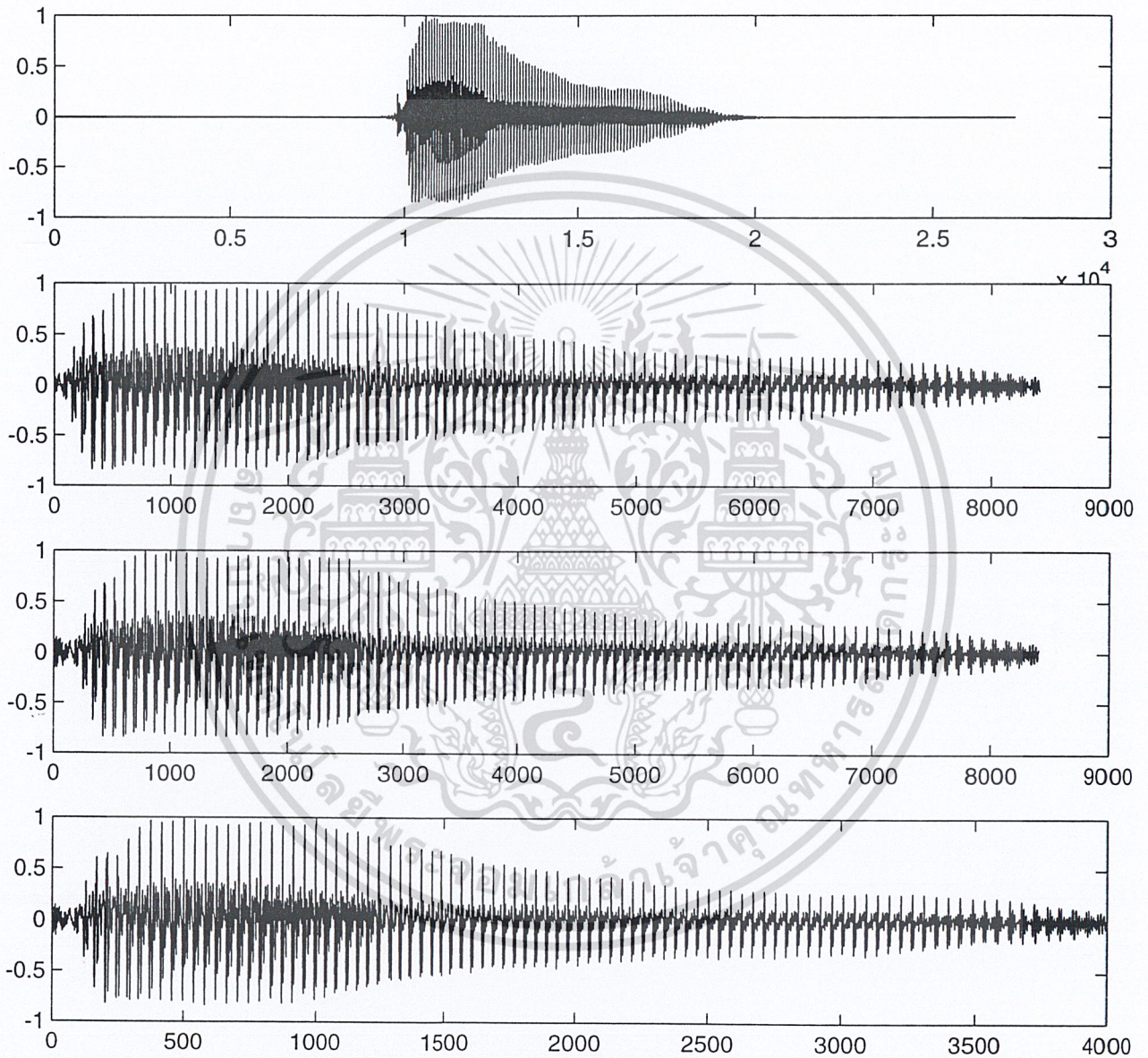
รูปที่ 1 แสดงตัวอย่างของเสียง

รูปที่ 1 แสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน

รูปที่ 2 แสดงการหาขอบเขตของคำโดยการหาค่าอัตราการตัดศูนย์

รูปที่ 3 แสดงการนอร์มอลไลซ์ให้มี 4000 แซมเปิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษารายบุคคลเท่านั้น เมื่อผู้ดูแลให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 แสดงตัวอย่างเสียง การหาขอบเขตของคำและการนอร์มอลไลซ์ของเลขเก้า โดย

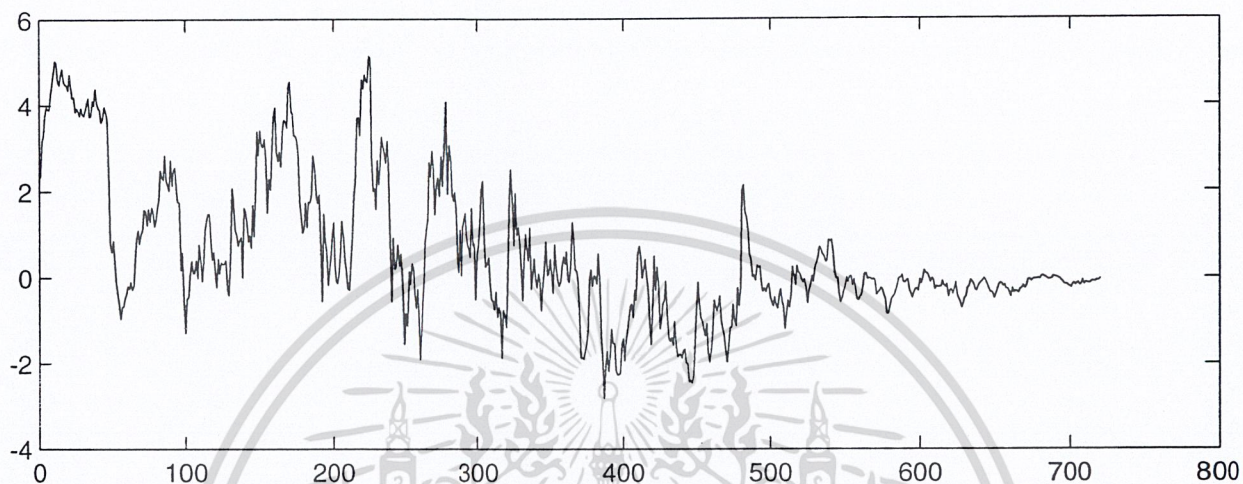
รูปที่ 1 แสดงตัวอย่างของเสียง

รูปที่ 2 แสดงการหาขอบเขตของคำโดยการหาค่าพลังงาน

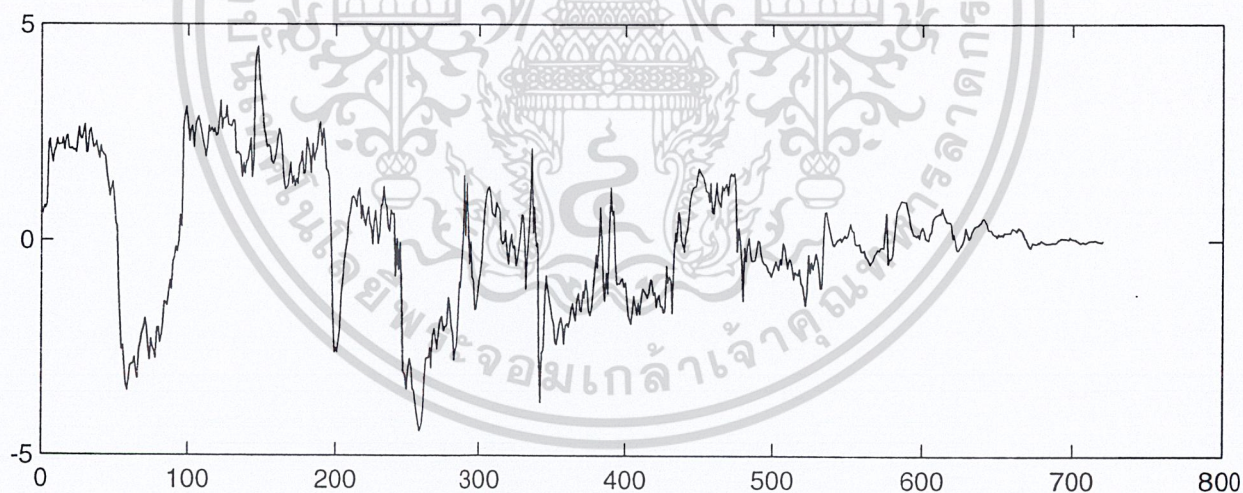
รูปที่ 3 แสดงการหาขอบเขตของคำโดยการหาค่าอัตราการตัดศูนย์

รูปที่ 4 แสดงการนอร์มอลไลซ์ให้มี 4000 แชมเปิล

4.1.2 การหาค่าพารามิเตอร์สัมประสิทธิ์เซปสตรีมถ่วงน้ำหนัก ของเลข 0-9

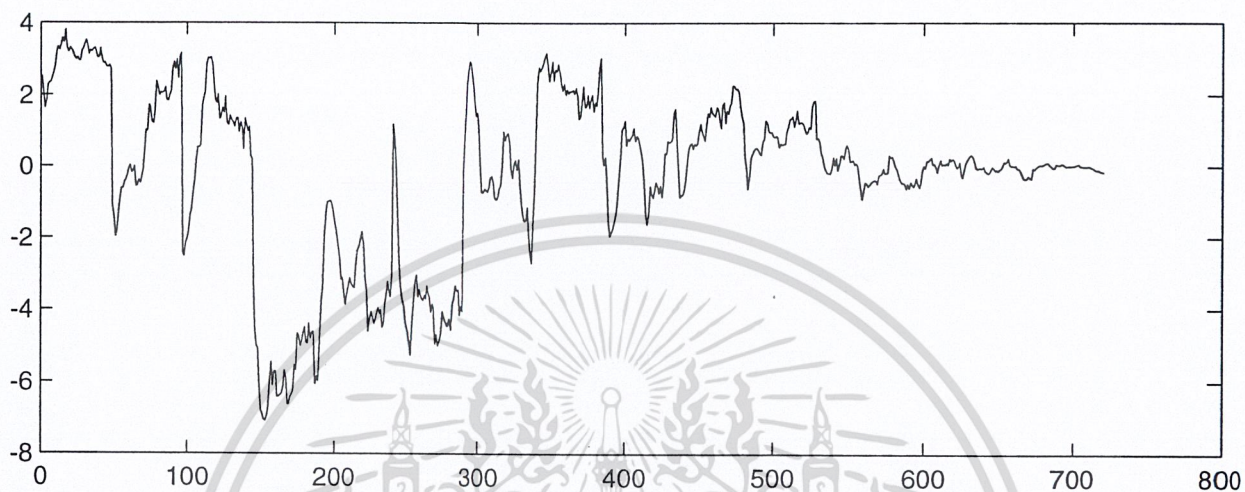


รูปที่ 4.11 แสดงค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียง ศูนย์

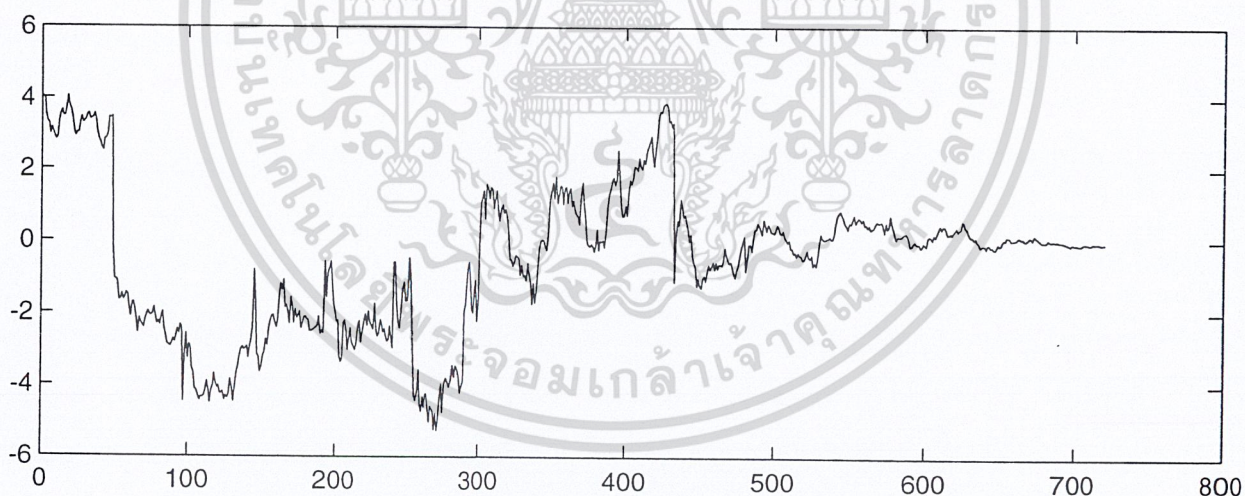


รูปที่ 4.12 แสดงค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียง หนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

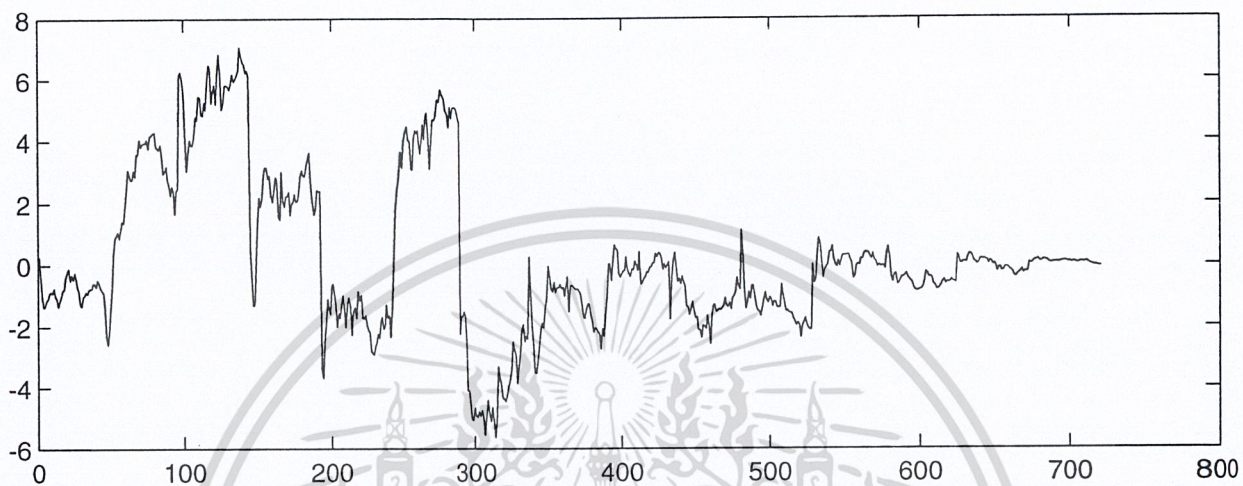


รูปที่ 4.13 แสดงค่าสัมประสิทธิ์สหสัมพันธ์ถ่วงน้ำหนักของสัญญาณเสียง สอง

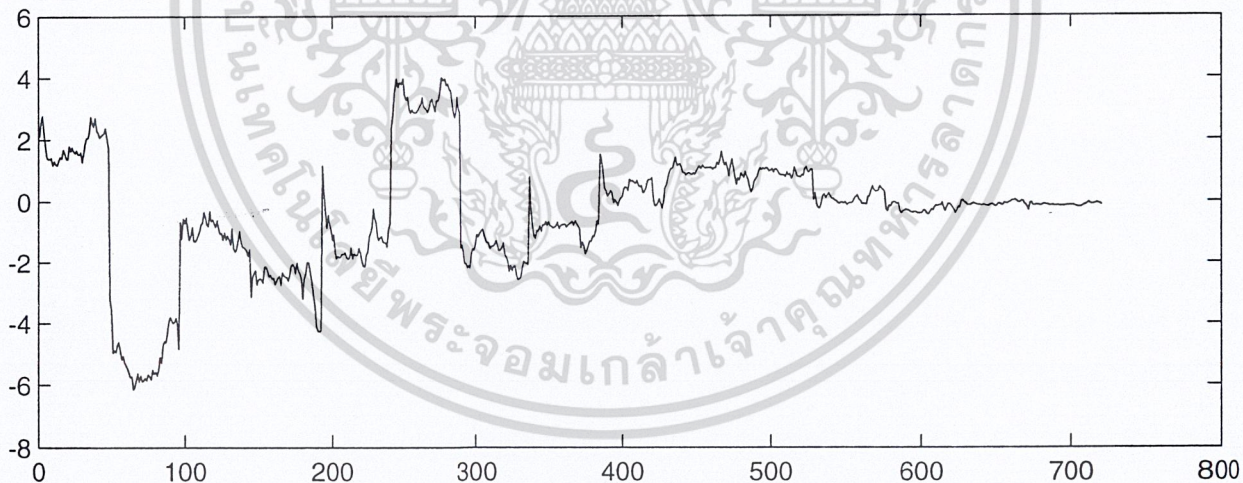


รูปที่ 4.14 แสดงค่าสัมประสิทธิ์สหสัมพันธ์ถ่วงน้ำหนักของสัญญาณเสียง สาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

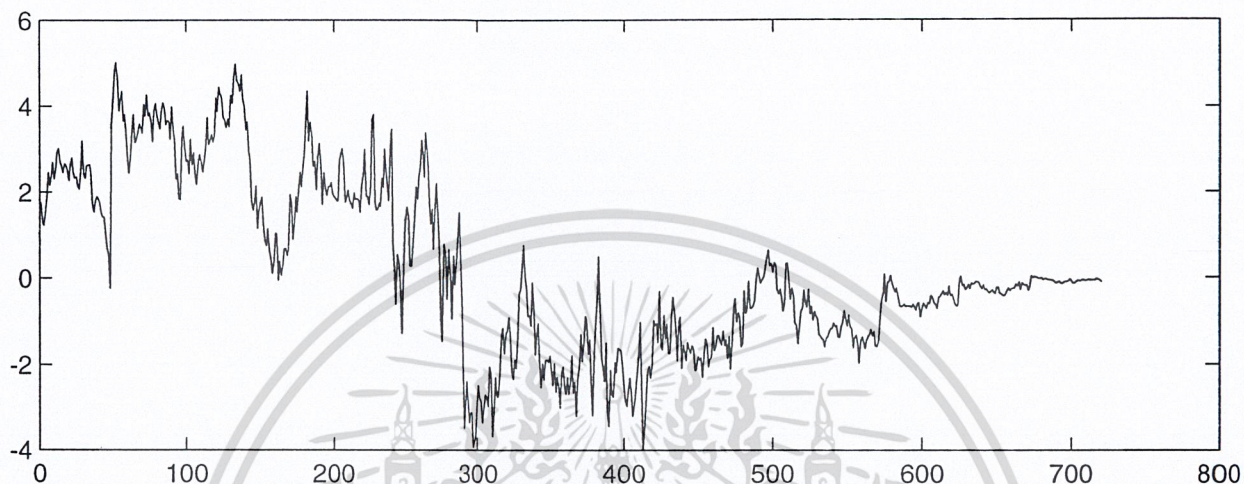


รูปที่ 4.15 แสดงค่าสัมประสิทธิ์สหสัมพันธ์ถ่วงน้ำหนักของสัญญาณเสียง สี

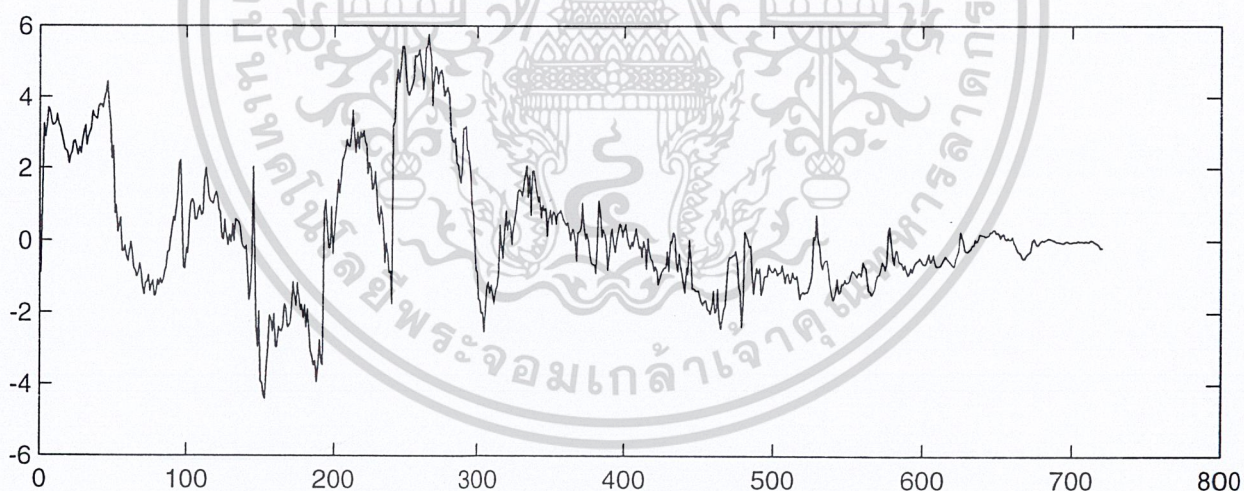


รูปที่ 4.16 แสดงค่าสัมประสิทธิ์สหสัมพันธ์ถ่วงน้ำหนักของสัญญาณเสียง ห้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

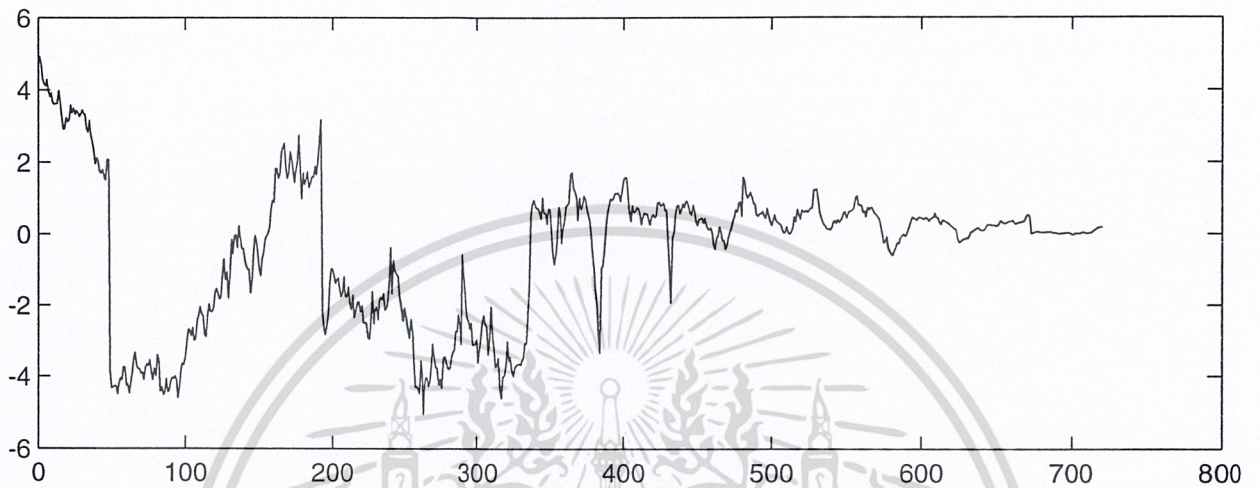


รูปที่ 4.17 แสดงค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียง หก

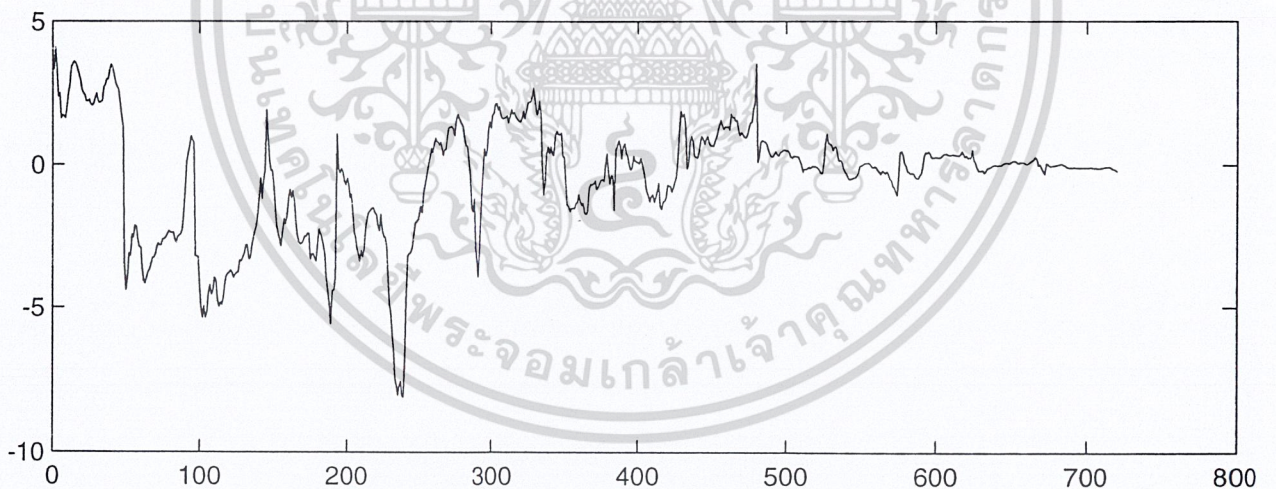


รูปที่ 4.18 แสดงค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียง เจ็ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

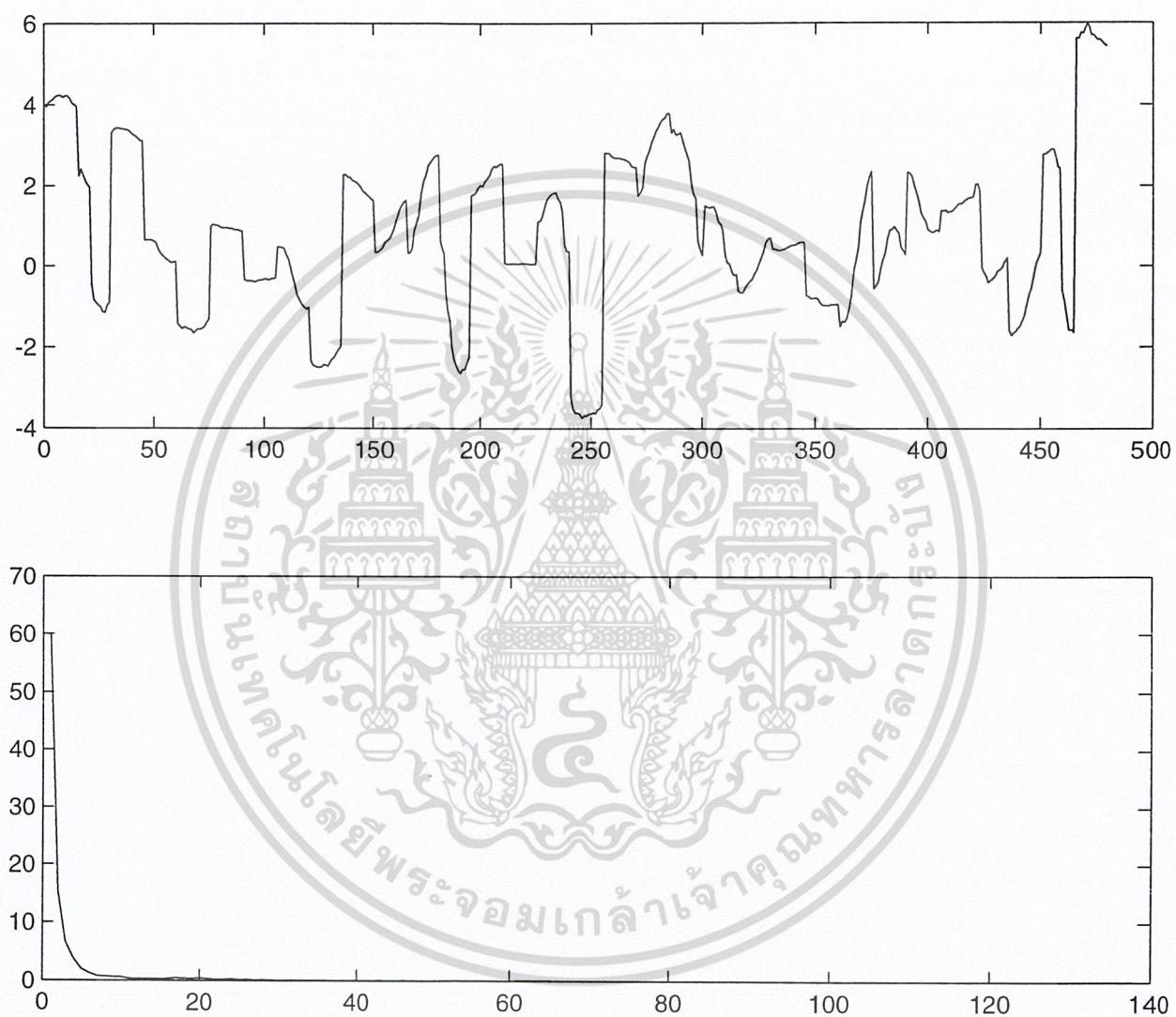


รูปที่ 4.19 แสดงค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียง เป็ด



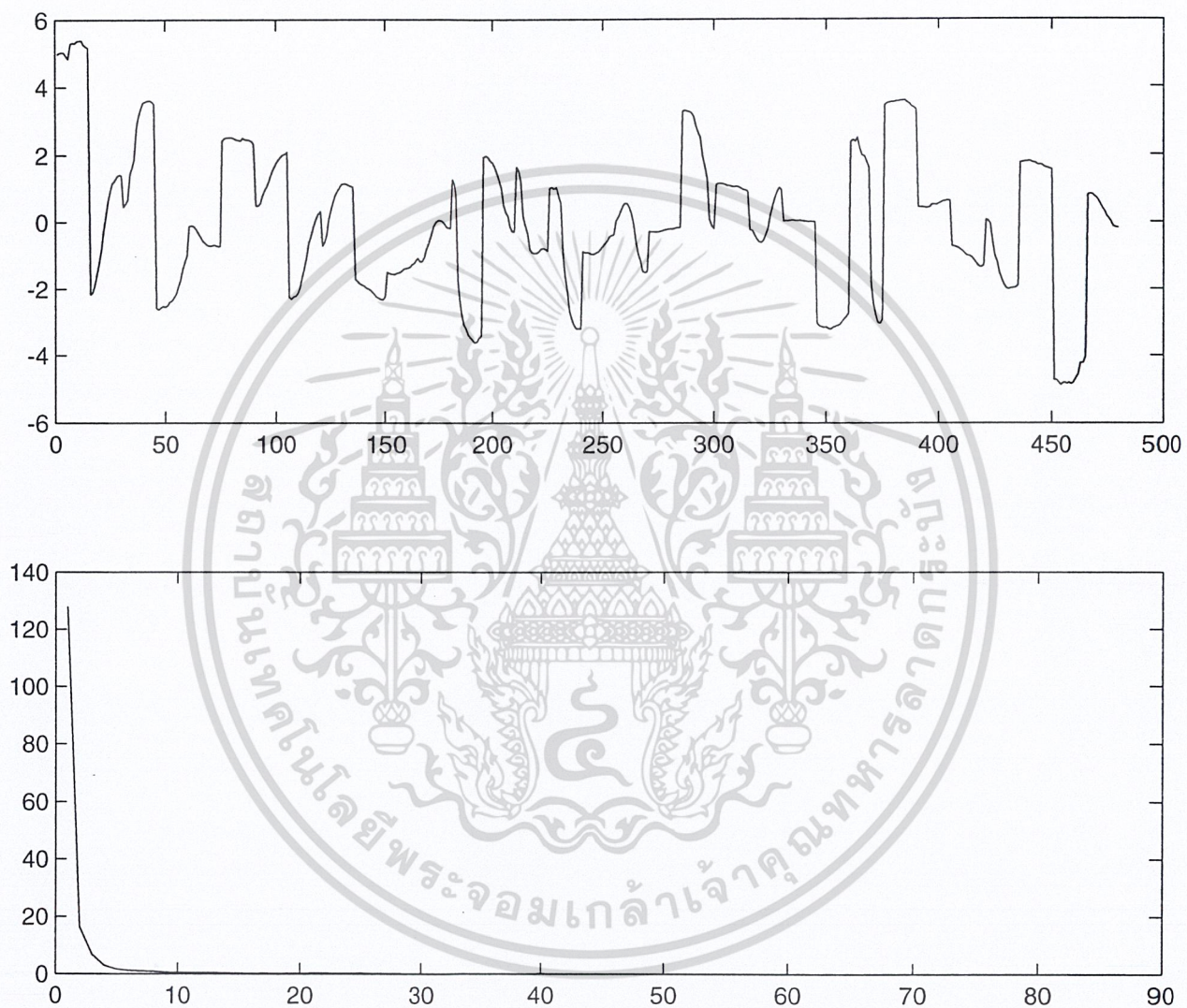
รูปที่ 4.20 แสดงค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียง เก้า

4.1.3 โค้ดบुक 32, 64, 128 ที่ได้และจำนวนรอบในการคำนวณ

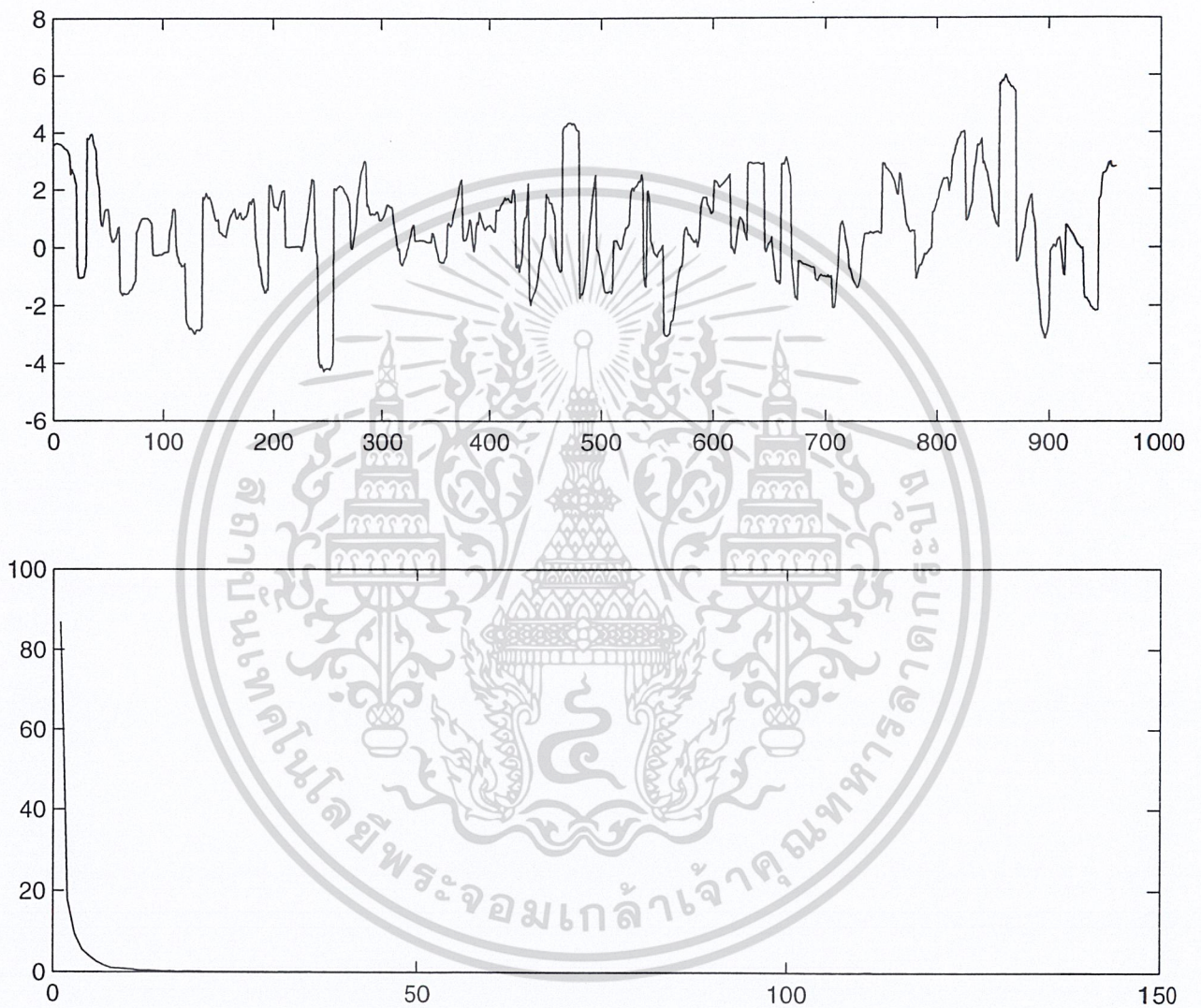


รูปที่ 4.21 แสดงโค้ดบुक 32 และจำนวนรอบของเสียงผู้ชายจากการคำนวณทั้งหมด 138 รอบ

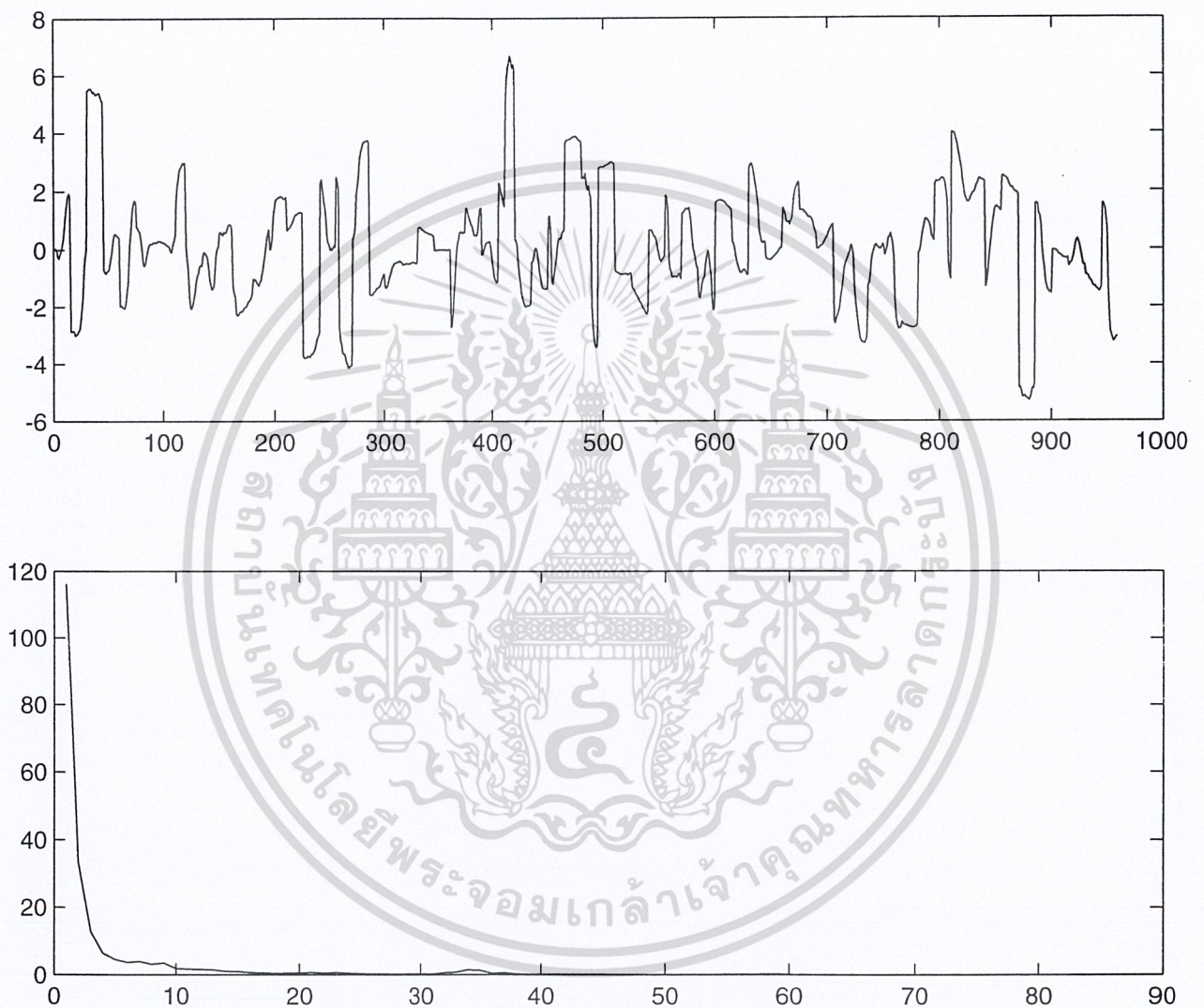
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.22 แสดงโค้ดบุค 32 และจำนวนรอบของเสียงผู้หญิงจากการคำนวณทั้งหมด 87 รอบ

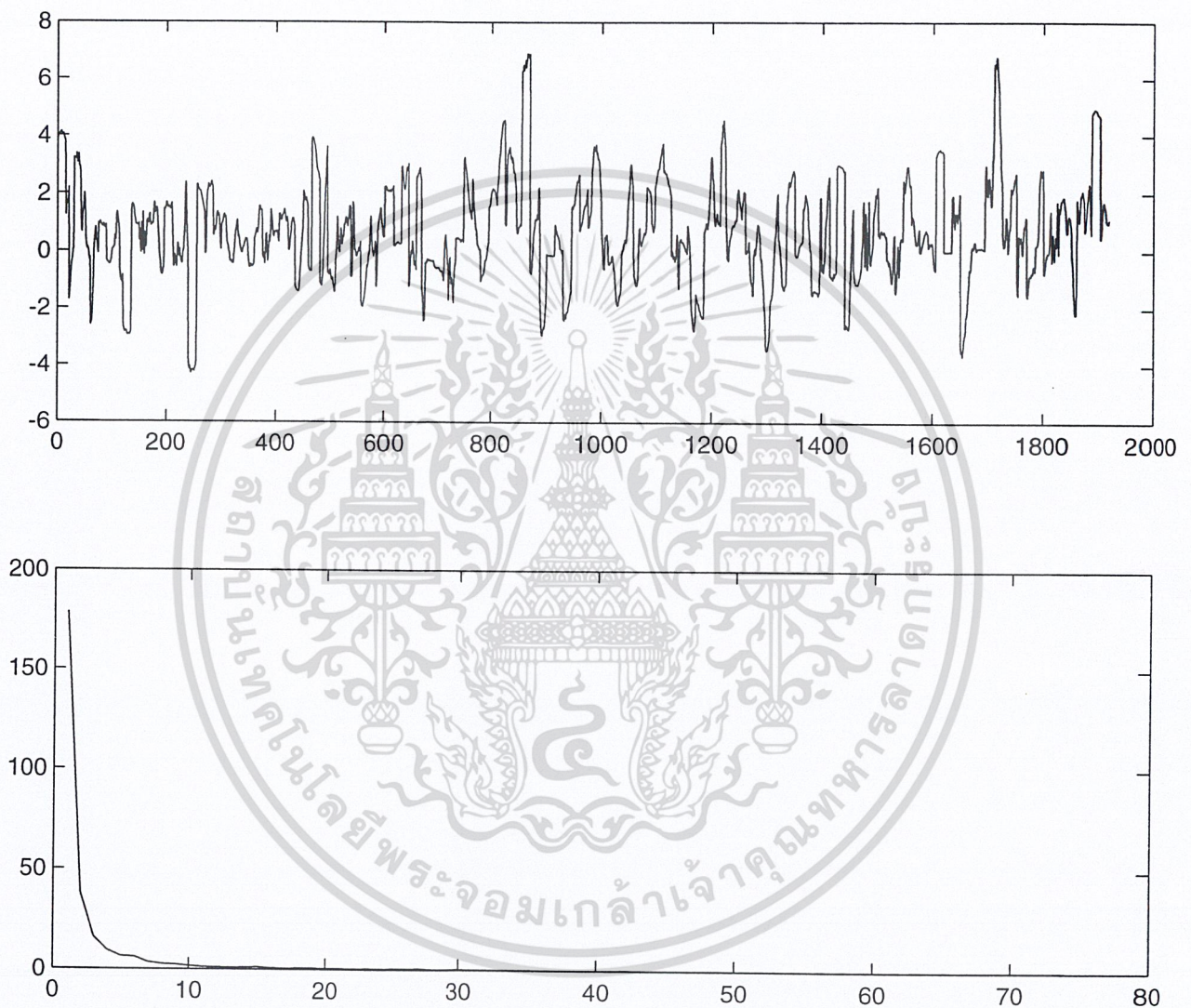


รูปที่ 4.23 แสดงโค้ดบุค 64 และจำนวนรอบของเสียงผู้ชายจากการคำนวณทั้งหมด 141 รอบ

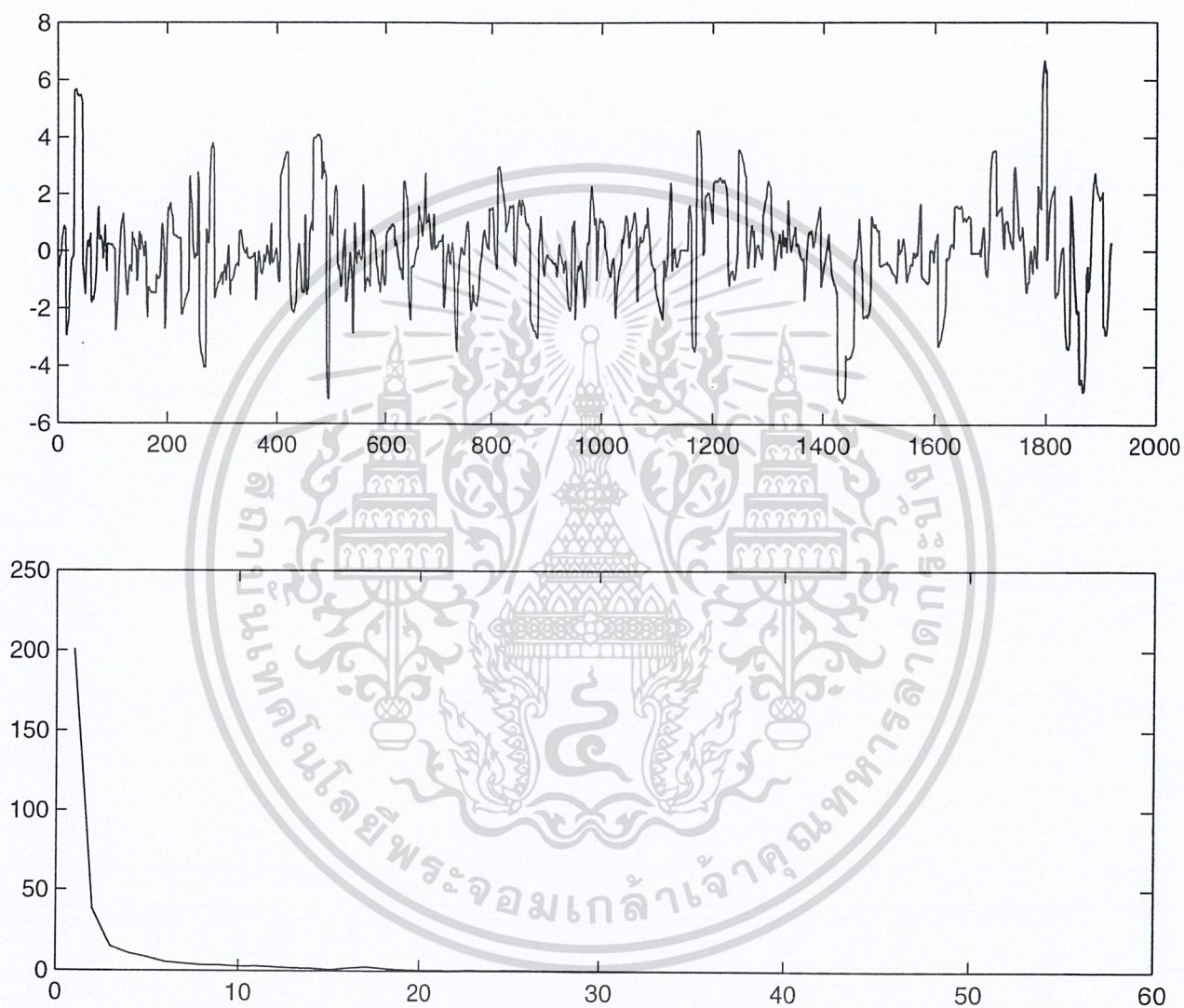


รูปที่ 4.24 แสดง ใค้ดบुक 64 และจำนวนรอบของเสียงผู้หญิงจากการคำนวณทั้งหมด 90 รอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.25 แสดงโค้ดบุค 128 และจำนวนรอบของเสียงผู้ชายจากการคำนวณทั้งหมด 74 รอบ



รูปที่ 4.26 แสดงโค้ดบुक 128 และจำนวนรอบของเสียงผู้หญิงจากการคำนวณทั้งหมด 56 รอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ขั้นตอนการเรียนรู้

1. เก็บเสียงที่ต้องการไว้เพื่อสร้าง โมเดล โดยจะเก็บเสียง 0 – 9 จากผู้ชาย 10 คน คนละ 3 ครั้ง และผู้หญิง 10 คน คนละ 3 ครั้ง รวมเป็น 600 เสียง
2. โดยทำการเก็บเสียงขนาด 8 บิต 22 kHz
3. นำข้อมูลเสียงที่ได้มาทำการหาค่าพารามิเตอร์ ที่เป็นตัวแทนของเสียงโดยใช้วิธี ลิเนียร์พรีดิคทีฟ และปรับปรุงอีกขั้นโดยใช้การเวทค่าพารามิเตอร์
4. นำพารามิเตอร์เสียงทั้งหมดมาทำการลดจำนวนข้อมูลโดยวิธีการเวคเตอร์ควอนไทซ์เซชัน ซึ่งจะได้เวคเตอร์ตัวแทนของเสียงที่ระบบสามารถรับรู้ได้ทั้งหมดออกมาเรียกว่า โค้ดบุค
5. ทำการหา เวคเตอร์โค้ด ของแต่ละเสียงโดยการนำเอาค่าเวท ที่ปรับปรุงมาจากแอลพีซี มาผ่านขั้นตอนอีก โดยแต่ละเสียงจะได้ค่าของ เวคเตอร์โค้ดออกมา
6. สร้างโมเดลของเสียงโดยการนำเวคเตอร์โค้ดของแต่ละกลุ่มเสียงมาผ่านขั้นตอนการสร้างโมเดลด้วยวิธีฮิดเดนมาร์คอฟโมเดล เมื่อทำงานครบทุกกลุ่มเสียง ก็จะได้โมเดลครบทุกเสียงและพร้อมที่จะนำไปใช้งานในขั้นตอนการวิเคราะห์ต่อไป

4.3 ขั้นตอนการวิเคราะห์

1. นำเสียงที่ต้องการทดสอบเพียง 1 เสียงมาผ่านขั้นตอน แอลพีซี และหาค่า เวท จะได้พารามิเตอร์ของเสียงนั้นและนำพารามิเตอร์มาเปรียบเทียบกับโค้ดบุค จะได้เวคเตอร์โค้ด
2. นำเวคเตอร์โค้ดมาหาค่าความน่าจะเป็นกับทุกโมเดลที่เก็บไว้ทั้งหมด โดยวิธี วิเทอร์บี อัลกอริทึม ก็จะได้ค่าความน่าจะเป็นของแต่ละโมเดลออกมา ค่าความน่าจะเป็นเมื่อเปรียบเทียบกับ โมเดลใดมีค่าสูงที่สุด ก็จะได้ผลว่าเสียงที่นำมาทดสอบตรงกับโมเดลใด
3. แสดงผลของเสียงที่ได้ว่าเป็นเสียงใด

4.4 การทดสอบการจดจำเสียงพูดในกรณีต่าง ๆ

เป็นการทดสอบการจดจำเสียงพูด โดยแต่ละกรณีจะทำการทดสอบ 2 แบบ คือ ใช้เสียงจากผู้พูดต้นแบบที่ใช้สร้างแบบอ้างอิง และใช้เสียงจากบุคคลอื่นหรือเรียกว่าเสียง unknown แสดง โดยตารางผลการทดลองในแต่ละกรณี ดังนี้

กรณีที่ 1 แสดงการทดสอบการจดจำเสียงพูดของผู้ชาย โดยใช้ไค์ดบुक 32

เสียงที่ต้องการ ทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	25	0	1	1	0	0	0	1	0	2	83.33%
1	0	26	1	2	0	0	0	1	0	0	86.67%
2	2	0	20	7	0	0	0	0	0	1	66.67%
3	1	0	0	25	0	0	1	1	1	1	83.33%
4	0	0	1	0	27	0	0	0	2	0	90%
5	0	1	1	3	0	24	0	0	0	1	80%
6	2	0	0	3	0	0	24	1	0	0	80%
7	0	0	0	1	0	0	2	27	0	0	90%
8	1	0	0	3	0	0	0	1	25	1	83.33%
9	0	0	1	2	0	0	0	0	0	27	90%

ตารางที่ 4.1 แสดงการจดจำเสียง 0-9 ของผู้ชายโดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 1

เสียงที่ต้องการ ทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	9	0	5	3	0	1	0	0	1	1	45%
1	0	10	1	5	2	0	1	1	0	0	50%
2	1	0	9	5	2	0	0	0	2	1	45%
3	0	0	3	11	0	0	0	2	3	1	55%
4	0	0	2	2	14	0	0	0	0	2	70%
5	0	1	1	7	1	8	0	1	0	1	40%
6	2	1	0	1	1	1	11	1	1	1	55%
7	0	0	0	2	0	1	0	16	1	0	80%
8	0	0	0	5	1	1	0	0	12	1	60%
9	0	0	0	6	1	0	0	0	0	13	65%

ตารางที่ 4.2 แสดงการจดจำเสียง 0-9 ของผู้ชายโดยใช้เสียงจากบุคคลอื่น (unknown) กรณีที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีที่ 2 แสดงการทดสอบการจดจำเสียงพูดของผู้หญิง โดยใช้ไค์ดบुक 32

เสียงที่ต้องการ ทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	29	1	0	0	0	0	0	0	0	0	96.67%
1	0	29	0	0	0	0	0	0	1	0	96.67%
2	2	0	19	4	0	0	0	0	1	4	63.33%
3	0	0	0	26	0	2	0	0	1	1	86.67%
4	0	2	0	0	28	0	0	0	0	0	93.33%
5	0	1	0	0	0	27	0	0	1	1	90%
6	4	2	0	0	0	0	23	0	0	1	76.67%
7	0	0	0	0	0	0	0	29	0	1	96.67%
8	0	0	0	0	0	2	0	0	27	1	90%
9	0	0	0	0	0	0	0	0	0	30	100%

ตารางที่ 4.3 แสดงการจดจำเสียง 0 – 9 ของผู้หญิงโดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 2

เสียงที่ต้องการ ทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	15	0	0	0	0	0	1	0	0	4	75%
1	0	14	1	0	0	0	1	0	0	1	70%
2	2	0	10	3	0	2	0	0	0	3	50%
3	0	1	1	13	0	2	0	1	1	1	65%
4	3	3	0	1	9	0	1	0	0	3	45%
5	0	1	0	2	0	13	0	2	1	1	65%
6	5	3	0	0	0	0	12	0	0	0	60%
7	0	2	0	0	0	3	0	14	0	1	70%
8	0	0	0	2	0	3	0	2	13	0	65%
9	0	0	2	2	0	0	0	0	2	14	70%

ตารางที่ 4.4 แสดงการจดจำเสียง 0 – 9 ของผู้หญิงโดยใช้เสียงจากบุคคลอื่น (unknown) กรณีที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีที่ 3 แสดงการทดสอบการจดจำเสียงพูดของผู้ชาย โดยใช้ไค์ดบुक 64

เสียงที่ต้องการ ทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	30	0	0	0	0	0	0	0	0	0	100%
1	0	29	0	0	0	0	1	0	0	0	96.67%
2	0	0	27	1	0	0	0	0	2	0	90%
3	1	1	1	24	0	1	0	0	2	0	80%
4	0	0	1	0	28	0	0	1	0	0	93.33%
5	0	0	0	0	0	26	0	1	1	2	86.67%
6	2	0	0	0	0	0	27	1	0	0	90%
7	0	0	0	0	0	0	0	29	0	1	96.67%
8	0	0	0	0	0	0	0	0	30	0	100%
9	0	0	0	0	0	0	0	0	0	30	100%

ตารางที่ 4.5 แสดงการจดจำเสียง 0 – 9 ของผู้ชาย โดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 3

เสียงที่ต้องการ ทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	13	0	2	1	1	1	0	0	0	2	65%
1	1	11	0	3	0	1	0	0	1	3	55%
2	0	0	11	2	2	0	1	0	2	2	55%
3	0	0	2	9	0	2	1	0	2	4	45%
4	0	0	0	1	13	4	0	0	0	2	65%
5	0	0	2	3	0	11	0	0	1	3	55%
6	2	0	0	0	1	0	13	1	0	3	65%
7	3	0	0	0	0	0	0	14	0	3	70%
8	1	0	0	2	1	3	0	0	11	2	55%
9	0	0	2	3	0	2	0	0	0	13	65%

ตารางที่ 4.6 แสดงการจดจำเสียง 0 – 9 ของผู้ชาย โดยใช้เสียงจากบุคคลอื่น (unknown) กรณีที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีที่ 4 แสดงการทดสอบการจดจำเสียงพูดของผู้หญิง โดยใช้ไค์คูป 64

เสียงที่ต้องการ ทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	29	0	0	0	0	0	2	0	0	0	96.67%
1	0	30	0	0	0	0	0	0	0	0	100%
2	1	0	27	0	0	1	0	0	1	0	90%
3	0	0	0	29	0	0	0	0	1	0	96.67%
4	1	0	0	0	29	0	0	0	1	0	96.67%
5	1	0	0	0	0	29	0	0	0	0	96.67%
6	1	0	0	0	0	0	29	0	0	0	96.67%
7	0	0	0	0	0	0	1	28	1	0	93.33%
8	0	0	0	0	0	0	0	0	30	0	100%
9	0	0	0	0	0	0	0	0	0	30	100%

ตารางที่ 4.7 แสดงการจดจำเสียง 0 – 9 ของผู้หญิง โดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 4

เสียงที่ต้องการ ทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	16	2	1	0	0	0	0	0	0	1	80%
1	4	11	1	1	0	0	0	0	1	2	55%
2	0	0	14	1	0	2	0	1	1	1	70%
3	0	0	2	13	0	2	0	0	1	2	65%
4	1	4	1	0	11	0	0	0	0	3	55%
5	0	0	1	1	0	17	0	0	0	1	85%
6	0	3	1	0	0	0	15	0	0	1	75%
7	0	0	0	3	0	0	2	12	0	3	60%
8	0	0	1	2	0	2	0	1	12	2	60%
9	0	0	1	3	0	0	0	1	3	12	60%

ตารางที่ 4.8 แสดงการจดจำเสียง 0 – 9 ของผู้หญิงโดยใช้เสียงจากบุคคลอื่น (unknown) กรณีที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีที่ 5 แสดงการทดสอบการจดจำเสียงพูดของผู้ชาย โดยใช้โค้ดบุค 128

เสียงที่ต้องการทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	30	0	0	0	0	0	0	0	0	0	100%
1	0	30	0	0	0	0	0	0	0	0	100%
2	0	0	29	1	0	0	0	0	0	0	96.67%
3	0	0	0	30	0	0	0	0	0	0	100%
4	0	0	0	0	30	0	0	0	0	0	100%
5	0	0	0	0	0	30	0	0	0	0	100%
6	0	0	0	0	0	0	30	0	0	0	100%
7	0	0	0	0	0	0	0	30	0	0	100%
8	0	0	0	0	0	0	0	0	30	0	100%
9	0	0	0	0	0	0	0	0	0	30	100%

ตารางที่ 4.9 แสดงการจดจำเสียง 0 - 9 ของผู้ชาย โดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 5

เสียงที่ต้องการทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	13	0	2	0	0	0	0	0	0	5	65%
1	0	11	0	1	1	1	0	0	1	5	55%
2	0	0	13	1	0	1	0	0	2	3	65%
3	0	0	0	11	0	1	0	0	1	6	55%
4	0	0	0	0	15	0	0	0	0	5	75%
5	0	0	1	2	0	14	0	0	0	3	70%
6	3	0	0	0	0	0	13	1	0	3	65%
7	0	0	1	0	0	1	0	12	0	6	60%
8	0	0	3	1	1	2	0	0	12	1	60%
9	0	0	0	1	0	2	0	0	1	16	80%

ตารางที่ 4.10 แสดงการจดจำเสียง 0 - 9 ของผู้ชาย โดยใช้เสียงจากบุคคลอื่น (unknown) กรณีที่ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีที่ 6 แสดงการทดสอบการจดจำเสียงพูดของผู้หญิง โดยใช้โค้ดบุค 128

เสียงที่ต้องการ ทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	30	0	0	0	0	0	0	0	0	0	100%
1	0	30	0	0	0	0	0	0	0	0	100%
2	0	0	30	0	0	0	0	0	0	0	100%
3	0	0	0	30	0	0	0	0	0	0	100%
4	0	0	0	0	30	0	0	0	0	0	100%
5	0	0	0	0	0	30	0	0	0	0	100%
6	0	0	0	0	0	0	30	0	0	0	100%
7	0	0	0	0	0	0	0	30	0	0	100%
8	0	0	0	0	0	0	0	0	30	0	100%
9	0	0	0	0	0	0	0	0	0	30	100%

ตารางที่ 4.11 แสดงการจดจำเสียง 0 – 9 ของผู้หญิง โดยใช้เสียงจากผู้พูดต้นแบบ กรณีที่ 6

เสียงที่ต้องการ ทดสอบ	ผลการจดจำเสียง										เปอร์เซ็นต์ ความถูกต้อง
	0	1	2	3	4	5	6	7	8	9	
0	14	0	0	0	0	0	0	0	0	6	70%
1	2	12	0	0	0	0	0	0	0	6	60%
2	0	1	11	0	0	0	0	0	0	8	55%
3	1	0	3	10	0	0	0	0	1	5	50%
4	0	2	0	1	10	0	0	1	1	5	50%
5	0	0	0	0	0	10	0	1	1	8	50%
6	1	2	1	0	2	1	10	1	0	2	50%
7	0	0	0	0	0	1	0	10	0	9	50%
8	0	0	0	1	0	1	0	1	12	5	60%
9	0	1	1	0	0	0	0	0	0	18	90%

ตารางที่ 4.12 แสดงการจดจำเสียง 0 – 9 ของผู้หญิง โดยใช้เสียงจากบุคคลอื่น (unknown) กรณีที่ 6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 ผลการเปรียบเทียบความถูกต้องของการทดสอบการรู้จำเสียงพูด

ผู้ทดสอบ	เปอร์เซ็นต์ต่ำสุด	เปอร์เซ็นต์สูงสุด	เปอร์เซ็นต์เฉลี่ย
ใช้ codebook 32			
กรณีที่ 1 การทดสอบเสียงผู้ชาย			
เสียงจากต้นแบบ	66.67%	90%	83.33%
เสียงจากบุคคลอื่น	40%	80%	56.5%
ผลโดยรวม			72.59%
กรณีที่ 2 การทดสอบเสียงผู้หญิง			
เสียงจากต้นแบบ	63.33%	100%	89%
เสียงจากบุคคลอื่น	45%	75%	63.5%
ผลโดยรวม			78.80%
ใช้ codebook 64			
กรณีที่ 3 การทดสอบเสียงผู้ชาย			
การทดสอบเสียงจากต้นแบบ	80%	100%	93.33%
การทดสอบเสียงจากบุคคลอื่น	45%	70%	59.5%
ผลโดยรวม			79.79%
กรณีที่ 4 การทดสอบเสียงผู้หญิง			
การทดสอบเสียงจากต้นแบบ	93.33%	100%	96.67%
การทดสอบเสียงจากบุคคลอื่น	55%	85%	66.5%
ผลโดยรวม			86.40%
ใช้ codebook 128			
กรณีที่ 5 การทดสอบเสียงผู้ชาย			
การทดสอบเสียงจากต้นแบบ	96.67%	100%	99.67%
การทดสอบเสียงจากบุคคลอื่น	55%	80%	65%
ผลโดยรวม			85.80%
กรณีที่ 6 การทดสอบเสียงผู้หญิง			
การทดสอบเสียงจากต้นแบบ	100%	100%	100%
การทดสอบเสียงจากบุคคลอื่น	50%	90%	58.5%
ผลโดยรวม			83.40%

ตารางที่ 4.13 แสดงผลค่าความถูกต้องทุกกรณี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5 บทวิจารณ์และบทสรุป

5.1 สรุปผลการทดลอง

จากการทดลองระบบการจดจำของเสียง 0 – 9 โดยการใช้สัญญาณเสียงที่สุ่มด้วยความถี่ 22 kHz ผลการทดลองสรุปได้ดังนี้

1. เสียงพูดของคน ๆ เดียวกัน พูดแต่ละครั้ง พบว่าเสียงที่พูดออกมาไม่เหมือนกัน
2. การหาขอบเขตของคำมีผลต่อความถูกต้องของการจดจำเสียง การหาขอบเขตของคำที่ถูกต้องแน่นอน จะเพิ่มอัตราการจดจำเสียงให้สูงขึ้น
3. เนื่องจากสัญญาณที่ผ่านการหาขอบเขตของคำมีความยาวเสียงที่ต่างกัน ทำให้การหาค่าเฉลี่ยของสัญญาณที่นำมาสร้างแบบอ้างอิงทำได้ไม่แน่นอน เพื่อความถูกต้องของการสร้างแบบอ้างอิงจึงทำการปรับสัญญาณก่อนที่จะมาทำการหาค่าพารามิเตอร์ จึงทำการนอร์มอลไลซ์ให้มีความยาวเท่ากันก่อน
4. จำนวนเสียงที่นำมาสร้างแบบอ้างอิงมีผลต่อความถูกต้องแม่นยำในการจดจำเสียง
5. การบันทึกเสียงเพื่อนำมาสร้างแบบอ้างอิง หรือเสียงที่ต้องการทดสอบ ควรจะบันทึกในสภาพห้องที่เงียบ มีเสียงรบกวนน้อย และมีอุปกรณ์บันทึกเสียงที่ดี เพื่อให้ได้สัญญาณเสียงที่มีคุณภาพ
6. ขนาดของโค้ดบุคมีผลต่อการหาค่าปรากฏที่เกิดขึ้น
7. ในการจำแนกเสียงโดยใช้วิธีฮิดเดนมาร์คอฟโมเดล สามารถจำได้ดีในเสียงต้นแบบที่ใช้สร้างแบบอ้างอิง ส่วนเสียงจากบุคคลอื่น สามารถจำได้ดีในระดับหนึ่ง

5.2 บทวิจารณ์และแนวทางพัฒนา

การที่ผลการทดลองออกมายังไม่สามารถรู้จำเสียงจากบุคคลอื่นได้ดี เนื่องจากเกิดสัญญาณรบกวนในขณะที่ทำการอัดเสียง และความแปรปรวนของความถี่และขนาดของเสียงจากบุคคลอื่นมีมากทำให้แบบจำลองอ้างอิงที่มีอยู่ไม่สามารถใช้เป็นแบบอ้างอิงของเสียงดังกล่าวได้ดีเท่าที่ควร ทำให้เกิดความคลาดเคลื่อนในการรู้จำเสียง

การพัฒนาและปรับปรุงในขั้นต่อไป ควรเลือกเสียงที่นำมาสร้างแบบอ้างอิงให้มีสถานะในการพูดที่แตกต่างกันมากขึ้น และสร้างแบบจำลองอ้างอิงที่สามารถจดจำเสียงได้ทั้งผู้ชายและผู้หญิง เพื่อเพิ่มประสิทธิภาพในการใช้งาน ควรปรับเปลี่ยนพารามิเตอร์ต่าง ๆ ที่ใช้ในการคำนวณให้เหมาะสม และอาจเปลี่ยนรูปแบบในการทำแบบจำลองอ้างอิงซึ่งคาดว่าจะช่วยในการรู้จำเสียงได้ดีขึ้น

เอกสารอ้างอิง

1. รศ.ดร. มনীศ สัจวรศิลป์, วรรัตน์ ภัทรอมรกุล. คู่มือการใช้งาน MATLAB ฉบับสมบูรณ์.
กรุงเทพฯ : อินโฟเพรส. 2543
2. Rabiner L.R., Juang B.H. **Fundamental of Speech Recognition**. New Jersey : Prentice Hall
International, Inc. 1993



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ก
การใช้งานโปรแกรม

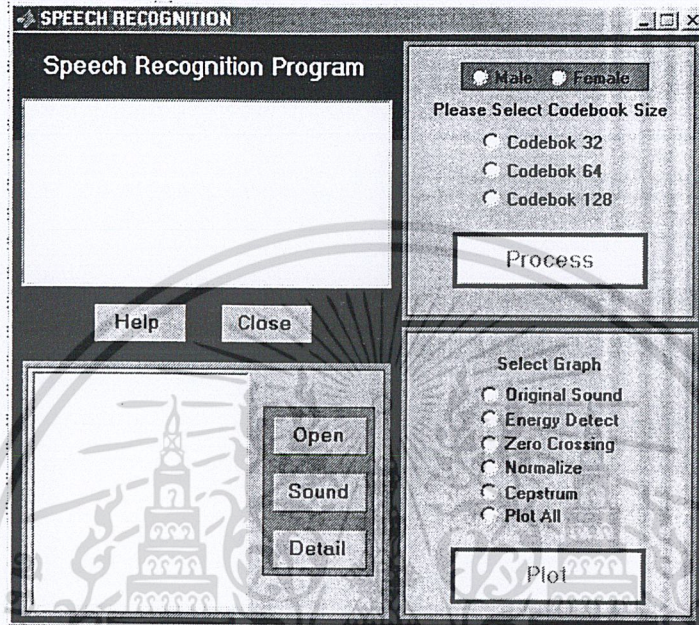
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



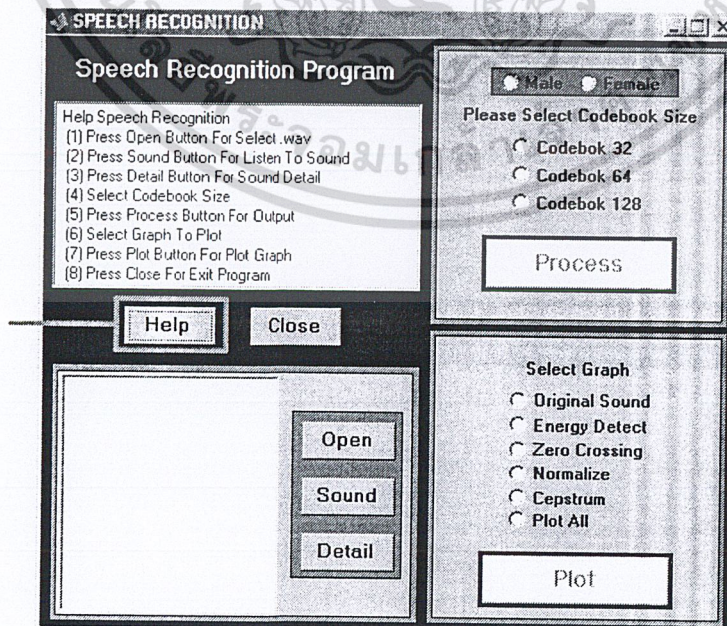
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานโปรแกรม SPEECH RECOGNITION

1. เมื่อทำการเปิดโปรแกรม Speech Recognition ขึ้นมาโปรแกรมจะมีลักษณะดังรูป

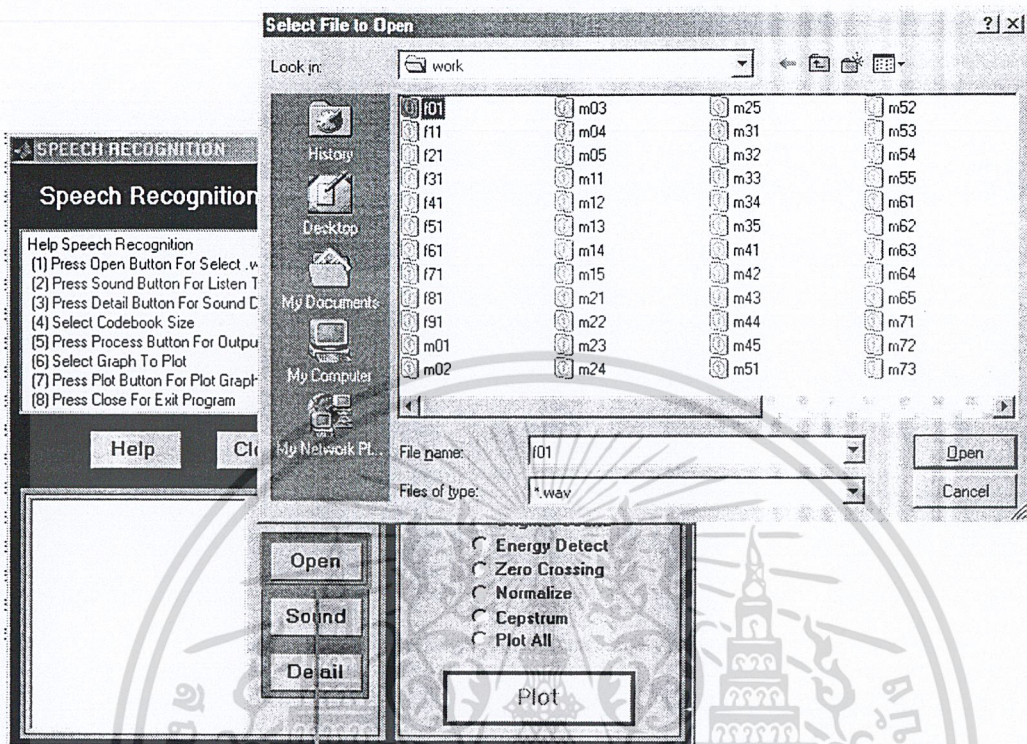


2. เมื่อทำการกดปุ่ม Help โปรแกรมจะแสดงวิธีการใช้งานขึ้นมาใน Edit Box ดังแสดงในรูป

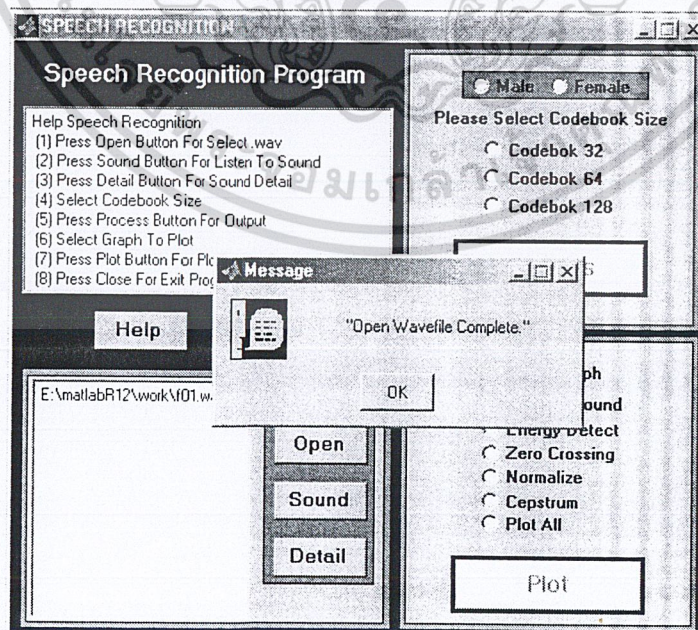


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. กดปุ่ม Open เพื่อเลือกเปิดไฟล์เสียงที่ต้องการนำมาทดสอบ โดยไฟล์เสียงจะอยู่ในรูป *.wav ดังรูป



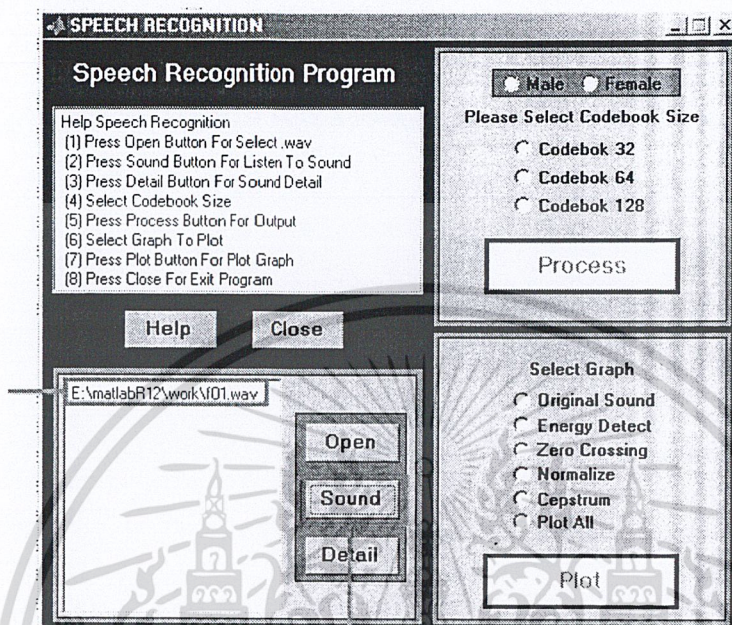
4. เมื่อทำการเปิดไฟล์เสียงเสร็จสิ้นแล้วจะมีข้อความแสดงว่า "Open Wave File Complete" ดังรูป



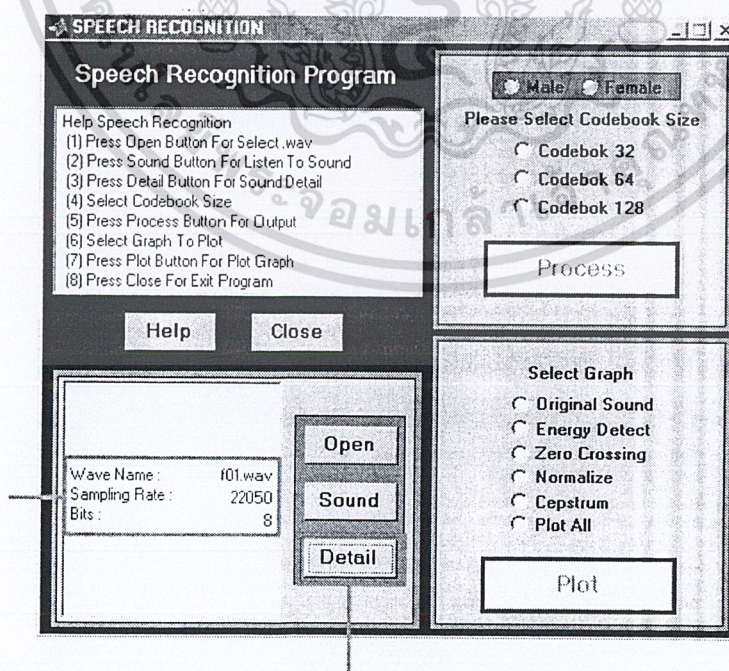
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เมื่อทำการเปิดไฟล์เสียงเรียบร้อยแล้วที่มาของเสียงจะแสดงใน Edit Box ดังรูป

6. เมื่อทำการกดปุ่ม Sound โปรแกรมจะแสดงเสียงที่นำมาทดสอบมาให้ฟัง

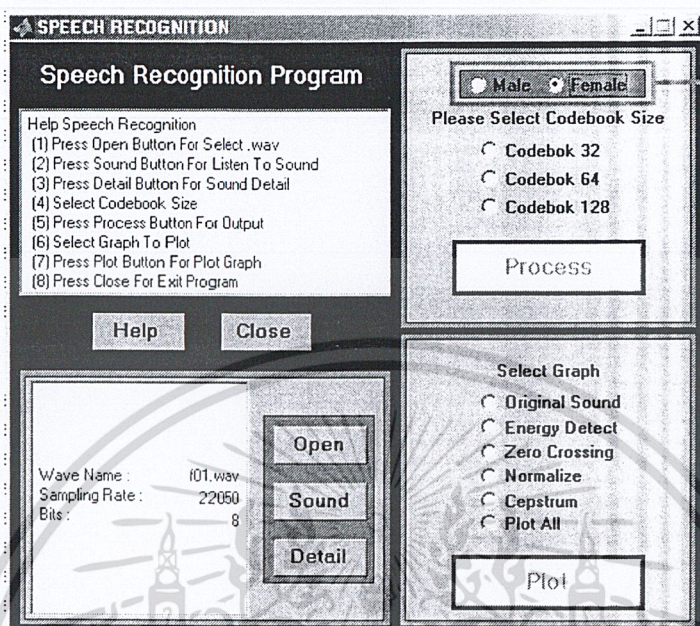


7. เมื่อทำการกดปุ่ม Detail โปรแกรมจะแสดงข้อมูลของเสียงที่นำมาทดสอบดังรูป

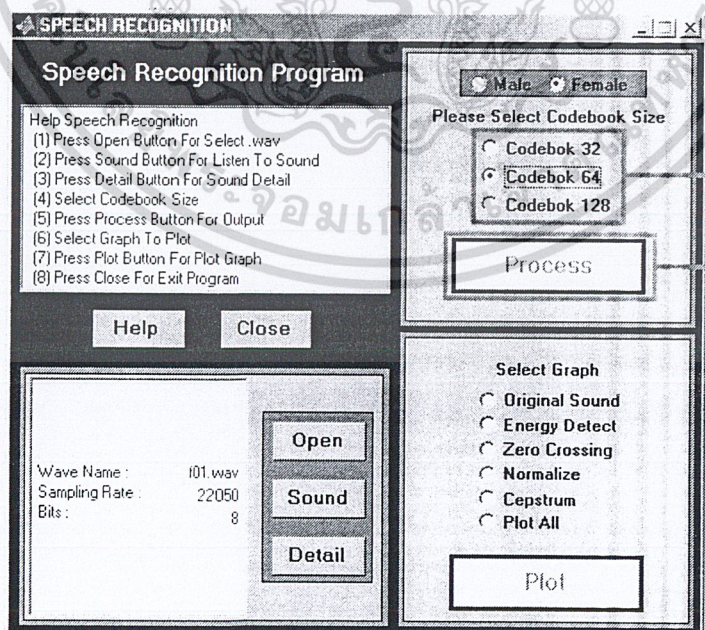


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. ทำการเลือกชายหรือหญิงเพื่อระบุเพศของเสียงที่นำมาทดสอบดังรูป

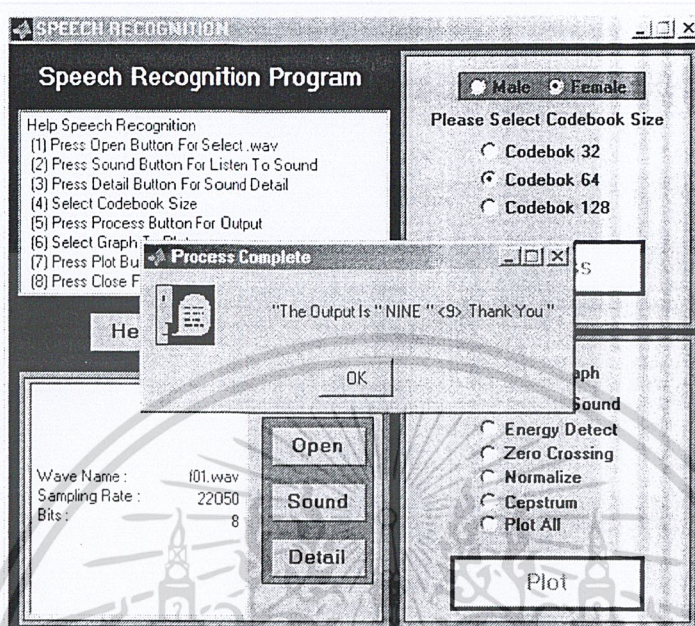


9. ทำการเลือกขนาดของโค้ดบุคที่จะใช้ในการทดสอบโดยมีให้เลือก 3 ขนาดคือ 32 , 64 และ 128 และกดปุ่ม Process เพื่อให้โปรแกรมทำงานดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

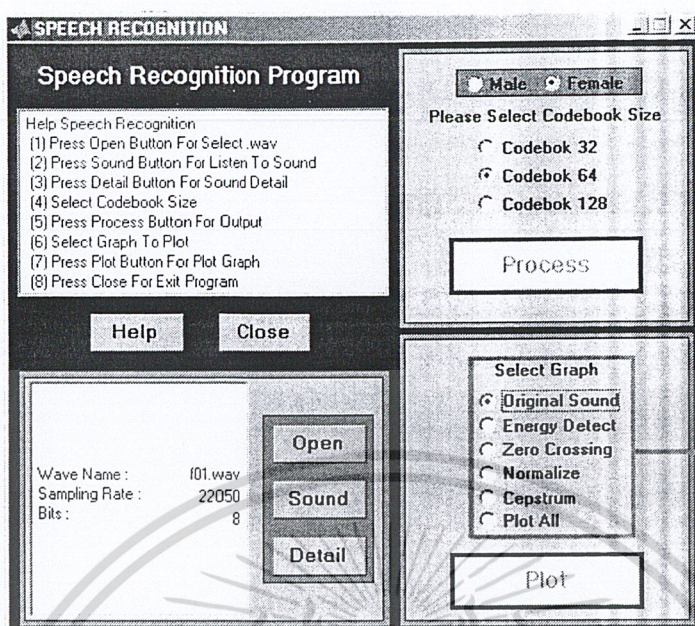
10. เมื่อโปรแกรมทำงานเสร็จสิ้นจะมีข้อความแสดงเอาท์พุทว่าเสียงที่นำทดสอบเป็นเสียงใด



11. เมื่อต้องการให้โปรแกรมแสดงกราฟ ทำการเลือกชนิดของกราฟที่ต้องการแสดง โดยมีให้เลือก 6 แบบ คือ

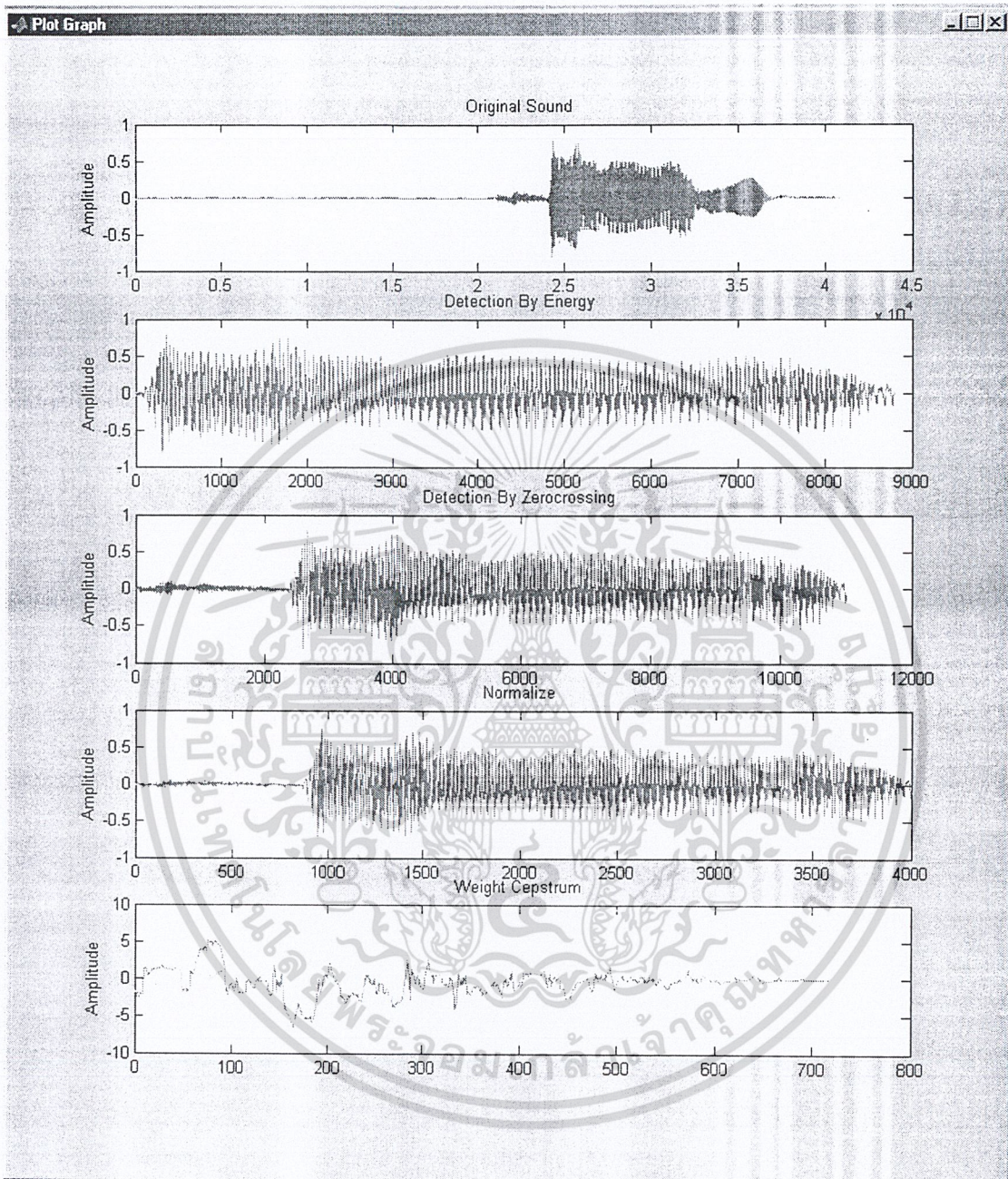
- Original Sound แสดงกราฟของเสียงต้นแบบ
- Energy Detect แสดงกราฟของเสียงที่ถูกคืเทค โดยพลังงาน
- Zero Crossing แสดงกราฟของเสียงที่ถูกคืเทค โดยการตัดศูนย์
- Normalize แสดงกราฟของเสียงที่ผ่านการนอร์มอลไลซ์
- Cepstrum แสดงกราฟของเสียงที่ผ่านการหาสัมประสิทธิ์เซปสตรัม
- Plot All แสดงกราฟทั้งหมดพร้อมกัน

ทำการกดปุ่ม Plot เพื่อพล็อตกราฟออกมาดังรูป



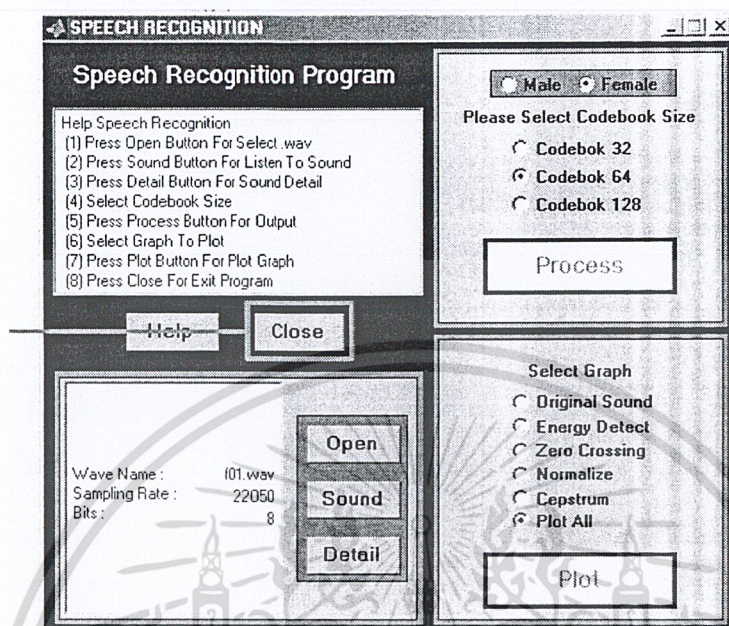
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กราฟที่ได้จะมีลักษณะดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12. กดปุ่ม Close เมื่อเสร็จสิ้นการใช้งาน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



โครงการหาขอบเขตของคำ

-ฟังก์ชัน energy1 : การเปรียบเทียบโดยการหาค่าพลังงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function [E,a1,b1,ref,s] = energy1(filename);
%
%
%
[inpt1,fs]=wavread(filename);
inpt=inpt1-mean(inpt1);
%
sample=100;
vnoloop=floor(length(inpt)/sample);
%
for k=1:vnoloop
    sum=0;
    for k1=1:sample
        sum=sum+abs(inpt((k-1)*sample+k1));
    end
    E(k)=sum;
end
%
Emax=max(E);
%
a=1/5;
%
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
            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
end
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



-ฟังก์ชัน zrocrs1 : การเปรียบเทียบโดยการหาอัตราการตัดศูนย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function [S,z,a2,b2,Zmax]=zrocrs1(filename,a1,b1);
%
%
%
[inpt1,fs]=wavread(filename);
inpt=inpt1-mean(inpt1);
%
sample=100;
vnoloop1=floor(length(inpt)/sample);
%
for k=1:vnoloop1
    sum=0;
    for k1=1:sample
        if k1==1
            sum = sum+abs(sg_n(inpt((k-1)*sample+k1)));
        else
            sum = sum+abs(sg_n(inpt((k-1)*sample+k1))-...
                sg_n(inpt((k-1)*sample+(k1-1))));
        end
    end
    z(k)=sum/(2*sample);
end
%
Zmax=max(z);
a=1/5;
%
for k=1:vnoloop1
    if z(k)==Zmax
        fixzero=k;
        break;
    end
end
refzro=a*Zmax;
%
for k1=a1:-1:1
    if z(k1)<=refzro
        a2=k1;
        break;
    end
end
for k2=b1:vnoloop1
    if z(k2)<=refzro
        b2=k2;
        break;
    end
end
end

%
n=0;
if isempty(b2)==1 | b2==vnoloop1 | isempty(a2)==1;
    if isempty(a2)==1;
        a2=a1-1;
        b2=vnoloop1-1;
        for k3=a2:b2
            for k4=1:sample
                n=n+1;
                S(n)=inpt((k3*sample)+k4);
            end
        end
    end
else

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

b2=vnoloop1-1;
for k3=a2:b2
    for k4=1:sample
        n=n+1;
        S(n)=inpt((k3*sample)+k4);
    end
end
end
else
for k5=a2:b2
    for k6=1:sample
        n=n+1;
        S(n)=inpt((k5*sample)+k6);
    end
end
end
end
function zerc=sg_n(s);
%
%
for i=1:length(s)
    if s(i)>0
        zerc=1;
    elseif s(i)==0
        zerc=0;
    else zerc=-1;
    end
end
end

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



โปรแกรมการนอร์มอลไลซ์สัญญาณ

-ฟังก์ชัน normalize : การนอร์มอลไลซ์สัญญาณให้ได้จำนวนตัวอย่าง 4000 ค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function normS = normalize(S,normsize);
%
%
%
%
%
%
%
%
normS(1)=S(1);
timestep=(length(S))/(normsize);

time=1;
j=2;
while j < normsize
    time = time + timestep;
    lowtime = floor(time);
    residue = time-lowtime;
    if (lowtime+1) > length(S)
        S(lowtime+1)=0;
    end
    normS(j) = S(lowtime)+residue*(S(lowtime+1)-S(lowtime));
    j=j+1;
end
normS(j)=S(length(S));
%

```





เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function cw=LPCprocess(ipt,order,Q);
% *****input*****
%   when ipt = Speech waveform
%   order= LPC order
%   Q = Order of cepstrum
% *****output*****
%   when cw = Weight cepstrum parameter
% *****
format long
S=ipt;
% *****
% Pre-emphasis
% *****
for i=1:length(S)
    if i==1
        s(i)=S(i);
    else s(i)=S(i)-0.9375*S(i-1);
    end
end
% *****
% Frame Blocking
% *****
flength=240;
fshift=80;
foverlap=160;
frame=1;
lastframe=floor(length(s)/fshift)-ceil(foverlap/fshift);
% *****
% Windowing
% *****
for frame=1:lastframe
    j=1;
    for i=((frame-1)*fshift)+1:(((frame-1)*fshift)+flength);
        w(frame,j)=s(i)*(0.54-0.46*cos(2*pi*(j-1)/(flength-1)));
        j=j+1;
    end
    % *****
    % Autocorrelation
    % *****
    for j=1:order+1
        r(frame,j)=0;
        for i=1:(flength+1-j)
            r(frame,j)=r(frame,j)+w(frame,i)*w(frame,(i-1+j));
        end
    end
end
% *****
% LPC Analysis
% *****
e(1)=r(frame,1);
K(1)=r(frame,2)/e(1);
A(1,1)=K(1);
for i=2:order
    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(frame,j);
        j=j-1;
    end
    K(i)=(r(frame,i+1)-sumAr)/e(i);
    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(frame)=sqrt(e(order));
for i=1:Q
    if i <=order
        a(frame,i)=A(order,i);
    else
        a(frame,i)=0;
    end
end
% *****
% LPC parameters converse to Cepstrum Coefficients
% *****
c(frame,1)=a(frame,1);
for i=2:Q
    sumca=0;
    for k=1:i-1
        sumca=sumca+(k*c(frame,k)*a(frame,i-k));
    end
    cep_sum(i)=(1/i)*sumca;
    if i <=order
        c(frame,i)=a(frame,i)+cep_sum(i);
    else
        c(frame,i)=cep_sum(i);
    end
end
% *****
% Parameter Weighting
% *****
for i=1:Q
    cweight(frame,i)=c(frame,i)*(1+(Q/2)*sin((pi*i)/Q));
end
cw=cweight;
cw=cw(:);
end

```



โครงการหาโค้ดบุค

-ฟังก์ชัน codebook32 : การหาโค้ดบุค 32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function [cb32,CVTD32] = codebook32(TrainingSetFile);
format long e;
VectorDimension = 15;
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 Codebook =====%
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
                NewCentroid(((nc-1)*VectorDimension)+nv) = ...
                    NewCentroid(((nc-1)*VectorDimension)+nv)+...
                    TrainingSetFile(((nf-1)*VectorDimension)+nv);
            end
        end
    end
end
end
if Cnum ~=0

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

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 Centroid To New Centroid =====%

OldCentroid = NewCentroid;

%-----%

end
cb32=NewCentroid;
save codebook32 cb32 CVTD32

```



-ฟังก์ชัน codebook64 : การหาโค้ดบุค 64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function [cb64,CVTD64] = codebook64(TrainingSetFile);
format long e;
VectorDimension = 15;
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 Codebook =====%
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
                NewCentroid(((nc-1)*VectorDimension)+nv) = ...
                    NewCentroid(((nc-1)*VectorDimension)+nv)+...
                    TrainingSetFile(((nf-1)*VectorDimension)+nv);
            end
        end
    end
end
end
if Cnum ~0

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าการณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

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 Centroid To New Centroid =====%

OldCentroid = NewCentroid;

%-----%

end
cb64=NewCentroid;
save codebook64 cb64 CVTD64

```



-ฟังก์ชัน codebook128 : การหาโค้ดบุค 128

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function [cb128,CVTD128] = codebook128(TrainingSetFile);
format long e;
VectorDimension = 15;
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 Codebook =====%
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
                NewCentroid(((nc-1).*VectorDimension)+nv) = ...
                    NewCentroid(((nc-1).*VectorDimension)+nv)+...
                    TrainingSetFile(((nf-1).*VectorDimension)+nv);
            end
        end
    end
end
end
end
if Cnum ==0

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

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 Centroid To New Centroid =====%
OldCentroid = NewCentroid;
%=====

end
cb128=NewCentroid;
save codebook128 cb128 CVTD128

```



-ฟังก์ชัน Observation : การหัดชี่นี้ได้บุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function Ob=Observation(weight,codebook,Number);
format long;
VectorDimension = 15;
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

```





โปรแกรมการหาแบบจำลองอ้างอิงโดยวิธีฮิดเดนมาร์คอฟ

- ฟังก์ชัน hmm32 : การหาแบบจำลองอ้างอิงของเสียงโดยใช้ค่าที่ผ่านโค้ดบุค 32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function [A,B]=hmm32(O);
format long e
%Initial model parameter
N=6;
M=32;
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
        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
                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
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

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

```

-ฟังก์ชัน **hmm64** : การหาแบบจำลองอ้างอิงของเสียงโดยใช้ค่าที่ผ่านโค้ดบุค 64



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function [A,B]=hmm64(O);
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
        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
                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
                end
                beta(n,t,i)=sumab;
            end
        end
    end
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

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

```

-ฟังก์ชัน hmm128 : การหาแบบจำลองอ้างอิงของเสียงโดยใช้ค่าที่ผ่านโค้ดบุค 128



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function [A,B]=hmm128(O);
format long e
%Initial model parameter
N=6;
M=128;
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
                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
end

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

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

```



โปรแกรมทดสอบสัญญาณเสียงกับแบบจำลองอ้างอิง

- ฟังก์ชัน `speechreg32` : การทดสอบเสียงที่เข้ามา กับแบบจำลองอ้างอิง `hmm32`

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function output=speechreg32(filename)
format long e
load codebook32
[E,a1,b1,ref,s]=detect1(filename);
[S,z,a2,b2,Zmax]=zrocrs1(filename,a1,b1);
z=normalize(S,4000);
w=LPCprocess(z,12,15);
O=observation(w,cb32,32);
%Initial model parameter
N=6;
M=32;
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 model32_00;
        case 1
            load model32_11;
        case 2
            load model32_22;
        case 3
            load model32_33;
        case 4
            load model32_44;
        case 5
            load model32_55;
        case 6
            load model32_66;
        case 7
            load model32_77;
        case 8
            load model32_88;
        case 9
            load model32_99;
        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
    end
end
%The viterbi algorithm
for i=1:N
    delta(1,i)=TT(i)*b(i,1);
    phi(1,i)=0;
end
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
end
output=numprobmax;

```

- ฟังก์ชัน `speechreg64` : การทดสอบเสียงที่เข้ามากับแบบจำลองอ้างอิง `hmm64`



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function output=speechreg64(filename)
format long e
load codebook64
[E,a1,b1,ref,s]=detect1(filename);
[S,z,a2,b2,Zmax]=zrocrs1(filename,a1,b1);
z=normalize(S,4000);
w=LPCprocess(z,12,15);
O=observation(w,cb64,64);
%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 model64_00;
    case 1
        load model64_11;
    case 2
        load model64_22;
    case 3
        load model64_33;
    case 4
        load model64_44;
    case 5
        load model64_55;
    case 6
        load model64_66;
    case 7
        load model64_77;
    case 8
        load model64_88;
    case 9
        load model64_99;
    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
end
%The viterbi algorithm
for i=1:N
    delta(1,i)=TT(i)*b(i,1);
    phi(1,i)=0;
end
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
output=numprobmax;

```

-ฟังค์ชั้น speechreg128 : การทดสอบเสียงที่เข้ามากับแบบจำลองอ้างอิง hmm128



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function output=speechreg128(filename)
format long e
load codebook128
[E,a1,b1,ref,s]=detect1(filename);
[S,z,a2,b2,Zmax]=zrocrs1(filename,a1,b1);
z=normalize(S,4000);
w=LPCprocess(z,12,15);
O=observation(w,cb128,128);
%Initial model parameter
N=6;
M=128;
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 model128_00;
    case 1
        load model128_11;
    case 2
        load model128_22;
    case 3
        load model128_33;
    case 4
        load model128_44;
    case 5
        load model128_55;
    case 6
        load model128_66;
    case 7
        load model128_77;
    case 8
        load model128_88;
    case 9
        load model128_99;
    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
end
%The viterbi algorithm
for i=1:N
    delta(1,i)=TT(i)*b(i,1);
    phi(1,i)=0;
end
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
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
end
output=numprobmax;

```



โปรแกรมการสร้างเมนู

Graphic User Interface : GUI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Graphic User Interface %%%%%%%%%%%%%%
fig1 = figure('Color',[0 0 0],...
             'MenuBar','none',...
             'Name','SPEECH RECOGNITION',...
             'NumberTitle','off',...
             'Position',[250 200 500 420],...
             'Resize','on',...
             'Tag','fig1');

uicontrol('Parent',fig1,'Units','Point','BackgroundColor',...
         [0.5 0.5 0.5],'Position',[212 162 160 150],...
         'Style','frame','Tag','frame1');
uicontrol('Parent',fig1,'Units','Point','BackgroundColor',...
         [0.5 0.5 0.5],'Position',[212 3 160 157],...
         'Style','frame','Tag','frame2');
uicontrol('Parent',fig1,'Units','Point','BackgroundColor',...
         [0.8 0.8 0.8],'Position',[215 165 154 144],...
         'Style','frame','Tag','frame3');
uicontrol('Parent',fig1,'Units','Point','BackgroundColor',...
         [0.8 0.8 0.8],'Position',[215 6 154 151],...
         'Style','frame','Tag','frame4');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Select Sex %%%%%%%%%%%%%%
male = 0; female = 0;

uicontrol('Parent',fig1,'Units','Point','BackgroundColor',...
         [0 1 0],'Position',[244 285 100 15],...
         'Style','frame','Tag','frame10');
sex1 = uicontrol('Parent',fig1,'Units','Point','BackgroundColor',...
               [0 1 0],'Position',[248 286 40 12],...
               'FontSize',8,'FontWeight','bold','String','Male',...
               'Style','radiobutton','Value',0,...
               'Callback',['set(sex1,'value',1),...
                           'set(sex2,'value',0),...
                           'male = 1;',...
                           'female = 0;']);
               %'Tag','RadioButton10');
sex2 = uicontrol('Parent',fig1,'Units','Point','BackgroundColor',...
               [0 1 0],'Position',[290 286 50 12],...
               'FontSize',8,'FontWeight','bold','String','Female',...
               'Style','radiobutton','Value',0,...
               'Callback',['set(sex1,'value',0),...
                           'set(sex2,'value',1),...
                           'male = 0;',...
                           'female = 1;']);
               %'Tag','RadioButton11');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Select Codebook Size %%%%%%%%%%%%%%
csize_1 = 0; csize_2 = 0; csize_3 = 0;

uicontrol('Parent',fig1,'Units','points','BackgroundColor',...
         [0.8 0.8 0.8],'FontSize',8,'FontWeight','bold',...
         'Position',[217 265 150 15],'String',...
         'Please Select Codebook Size',...
         'Style','text','Tag','StaticText1');
csize1 = uicontrol('Parent',fig1,'Units','Point','BackgroundColor',...
                 [0.8 0.8 0.8],'Position',[255 250 100 15],...
                 'FontSize',8,'FontWeight','bold','String','Codebook
32',...

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์โดยคณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่าการณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

'Style','radiobutton','Value',0,...
'CallBack',['set(csize1,'value',1),'...
            'set(csize2,'value',0),'...
            'set(csize3,'value',0),'...
            'csize_1 = 1;','...
            'csize_2 = 0;','...
            'csize_3 = 0;']);
    %'Tag','RadioButton1');
csize2 = uicontrol('Parent',fig1,'Units','Point','BackgroundColor',...
    [0.8 0.8 0.8],'Position',[255 235 100 15],...
    'FontSize',8,'FontWeight','bold','String','Codebok
64',...
    'Style','radiobutton','Value',0,...
    'CallBack',['set(csize1,'value',0),'...
                'set(csize2,'value',1),'...
                'set(csize3,'value',0),'...
                'csize_1 = 0;','...
                'csize_2 = 1;','...
                'csize_3 = 0;']);
    %'Tag','RadioButton2');
csize3 = uicontrol('Parent',fig1,'Units','Point','BackgroundColor',...
    [0.8 0.8 0.8],'Position',[255 220 100 15],...
    'FontSize',8,'FontWeight','bold','String','Codebok
128',...
    'Style','radiobutton','Value',0,...
    'CallBack',['set(csize1,'value',0),'...
                'set(csize2,'value',0),'...
                'set(csize3,'value',1),'...
                'csize_1 = 0;','...
                'csize_2 = 0;','...
                'csize_3 = 1;']);
    %'Tag','RadioButton3');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Process Button %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
uicontrol('Parent',fig1,'Units','Point','BackgroundColor',[0 0 0],...
    'Position',[240 180 104 30],...
    'Style','frame','Tag','frame5');
uicontrol('Parent',fig1,'Callback','selectcode','Units','Point',...
    'BackgroundColor',[1 1 1],'FontSize',12,...
    'ForegroundColor',[1 0 0],'FontWeight','bold',...
    'Position',[242 181.5 101 26.5],'String','Process',...
    'Style','PushButton',...
    'Tag','pushbutton1');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Select Graph %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
uicontrol('Parent',fig1,'Units','points','BackgroundColor',...
    [0.8 0.8 0.8],'FontSize',8,'FontWeight','bold',...
    'Position',[217 130 150 15],'String','Select
Graph',...
    'Style','text','Tag','StaticText2');
gplot1 = uicontrol('Parent',fig1,'Units','Point','BackgroundColor',...
    [0.8 0.8 0.8],'Position',[255 116 100 13],...
    'FontSize',8,'FontWeight','bold','String',...
    'Original Sound','Style','radiobutton','Value',0,...
    'CallBack',['set(gplot1,'value',1),'...
                'set(gplot2,'value',0),'...
                'set(gplot3,'value',0),'...
                'set(gplot4,'value',0),'...
                'set(gplot5,'value',0),'...

```

```

        'set(gplot6, 'value', 0), '...
        'gplot_1 = 1;', '...
        'gplot_2 = 0;', '...
        'gplot_3 = 0;', '...
        'gplot_4 = 0;', '...
        'gplot_5 = 0;', '...
        'gplot_6 = 0;']);
        %'Tag', 'RadioButton4');
gplot2 = uicontrol('Parent', fig1, 'Units', 'Point', 'BackgroundColor', ...
    [0.8 0.8 0.8], 'Position', [255 103 100 13], ...
    'FontSize', 8, 'FontWeight', 'bold', 'String', ...
    'Energy Detect', 'Style', 'radiobutton', 'Value', 0, ...
    'CallBack', ['set(gplot1, 'value', 0), '...
        'set(gplot2, 'value', 1), '...
        'set(gplot3, 'value', 0), '...
        'set(gplot4, 'value', 0), '...
        'set(gplot5, 'value', 0), '...
        'set(gplot6, 'value', 0), '...
        'gplot_1 = 0;', '...
        'gplot_2 = 1;', '...
        'gplot_3 = 0;', '...
        'gplot_4 = 0;', '...
        'gplot_5 = 0;', '...
        'gplot_6 = 0;']);
        %'Tag', 'RadioButton5');
gplot3 =
uicontrol('Parent', fig1, 'Units', 'Point', 'BackgroundColor', ...
    [0.8 0.8 0.8], 'Position', [255 90 100 13], ...
    'FontSize', 8, 'FontWeight', 'bold', 'String', ...
    'Zero Crossing', 'Style', 'radiobutton', 'Value', 0, ...
    'CallBack', ['set(gplot1, 'value', 0), '...
        'set(gplot2, 'value', 0), '...
        'set(gplot3, 'value', 1), '...
        'set(gplot4, 'value', 0), '...
        'set(gplot5, 'value', 0), '...
        'set(gplot6, 'value', 0), '...
        'gplot_1 = 0;', '...
        'gplot_2 = 0;', '...
        'gplot_3 = 1;', '...
        'gplot_4 = 0;', '...
        'gplot_5 = 0;', '...
        'gplot_6 = 0;']);
        %'Tag', 'RadioButton6');
gplot4 =
uicontrol('Parent', fig1, 'Units', 'Point', 'BackgroundColor', ...
    [0.8 0.8 0.8], 'Position', [255 77 100 13], ...
    'FontSize', 8, 'FontWeight', 'bold', 'String', 'Normalize', ...
    'Style', 'radiobutton', 'Value', 0, ...
    'CallBack', ['set(gplot1, 'value', 0), '...
        'set(gplot2, 'value', 0), '...
        'set(gplot3, 'value', 0), '...
        'set(gplot4, 'value', 1), '...
        'set(gplot5, 'value', 0), '...
        'set(gplot6, 'value', 0), '...
        'gplot_1 = 0;', '...
        'gplot_2 = 0;', '...
        'gplot_3 = 0;', '...
        'gplot_4 = 1;', '...
        'gplot_5 = 0;', '...
        'gplot_6 = 0;']);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานภายในบริษัทเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

                                %'Tag', 'RadioButton7');
gplot5 =
uicontrol('Parent',fig1,'Units','Point','BackgroundColor',...
          [0.8 0.8 0.8],'Position',[255 64 100 13],...
          'FontSize',8,'FontWeight','bold','String','Cepstrum',...
          'Style','radiobutton','Value',0,...
          'CallBack',['set(gplot1,'value',0),'...
                    'set(gplot2,'value',0),'...
                    'set(gplot3,'value',0),'...
                    'set(gplot4,'value',0),'...
                    'set(gplot5,'value',1),'...
                    'set(gplot6,'value',0),'...
                    'gplot_1 = 0;','...
                    'gplot_2 = 0;','...
                    'gplot_3 = 0;','...
                    'gplot_4 = 0;','...
                    'gplot_5 = 1;','...
                    'gplot_6 = 0;']]);
                                %'Tag', 'RadioButton8');
gplot6 = uicontrol('Parent',fig1,'Units','Point','BackgroundColor',...
                  [0.8 0.8 0.8],'Position',[255 51 100 15],...
                  'FontSize',8,'FontWeight','bold','String','Plot All',...
                  'Style','radiobutton','Value',0,...
                  'CallBack',['set(gplot1,'value',0),'...
                              'set(gplot2,'value',0),'...
                              'set(gplot3,'value',0),'...
                              'set(gplot4,'value',0),'...
                              'set(gplot5,'value',0),'...
                              'set(gplot6,'value',1),'...
                              'gplot_1 = 0;','...
                              'gplot_2 = 0;','...
                              'gplot_3 = 0;','...
                              'gplot_4 = 0;','...
                              'gplot_5 = 0;','...
                              'gplot_6 = 1;']]);
                                %'Tag', 'RadioButton9');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Plot Graph Button %%%%%%%%%%%%%%%
uicontrol('Parent',fig1,'Units','Point','BackgroundColor',[0 0 0],...
          'Position',[240 12 104 30],...
          'Style','frame','Tag','frame6');
uicontrol('Parent',fig1,'Callback','plotgraph','Units','Point',...
          'BackgroundColor',[1 1 1],'FontSize',12,...
          'ForegroundColor',[1 0 0],'FontWeight','bold',...
          'Position',[242 13.5 101 26.5],'String','Plot',...
          'Style','PushButton',...
          'Tag','pushbutton2');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Help Message %%%%%%%%%%%%%%%

uicontrol('Parent',fig1,'Units','points','BackgroundColor',...
          [0 0 0],'FontSize',11,'FontWeight','bold',...
          'ForegroundColor',[1 1 1],'Position',...
          [5 280 200 25],...
          'String','Speech Recognition Program',...
          'Style','text','Tag','StaticText3');
edit1 = uicontrol('Parent',fig1,'Units','points','BackgroundColor',...
                  [1 1 1],'HorizontalAlignment','left',...
                  'Max',15,'Position',[5 180 202 100],'Style',...
                  'edit','Tag','EditText1');

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานที่อนุญาตเท่านั้น โดยผู้ดูแลระบบขอสงวนสิทธิ์ในการดำเนินการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

uicontrol('Parent',fig1,'Callback','help1','Units','Point',...
    'BackgroundColor',[0.8 0.8 0.8],'FontSize',10,...
    'ForegroundColor',[0 0 0],'FontWeight','bold',...
    'Position',[45 150 50 20],'String','Help','Style',...
    'PushButton','Tag','pushbutton3');
uicontrol('Parent',fig1,'Callback','close','Units','Point',...
    'BackgroundColor',[0.8 0.8 0.8],'FontSize',10,...
    'ForegroundColor',[0 0 0],'FontWeight','bold',...
    'Position',[115 150 50 20],'String','Close','Style',...
    'PushButton','Tag','pushbutton4');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Sound %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

uicontrol('Parent',fig1,'Units','Point','BackgroundColor',...
    [0.5 0.5 0.5],'Position',[5 3 203 137],...
    'Style','frame','Tag','frame7');
uicontrol('Parent',fig1,'Units','Point','BackgroundColor',...
    [0.8 0.8 0.8],'Position',[8.5 6 195.5 130.5],...
    'Style','frame','Tag','frame8');
uicontrol('Parent',fig1,'Units','Point','BackgroundColor',...
    [0 0 1],'Position',[139 26 60 90],...
    'Style','frame','Tag','frame9');
edit2 = uicontrol('Parent',fig1,'Units','points','BackgroundColor',...
    [1 1 1],'HorizontalAlignment','left',...
    'Max',10,'Position',[11 8 120 126],'Style',...
    'edit','Tag','EditText1');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Detail %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

detail1 = uicontrol('Parent',fig1,'Units','points',...
    'BackgroundColor',[1 1
1],'HorizontalAlignment','right',...
    'Position',[75 79 50 11],'Style','text','Tag',...
    'StaticText4');
detail2 = uicontrol('Parent',fig1,'Units','points',...
    'BackgroundColor',[1 1
1],'HorizontalAlignment','right',...
    'Position',[75 67 50 11],'Style','text','Tag',...
    'StaticText5');
detail3 = uicontrol('Parent',fig1,'Units','points',...
    'BackgroundColor',[1 1
1],'HorizontalAlignment','right',...
    'Position',[75 55 50 11],'Style','text','Tag',...
    'StaticText6');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Open , Sound , Detail Button %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

uicontrol('Parent',fig1,'Callback','openwave','Units','Point',...
    'BackgroundColor',[0.8 0.8 0.8],'FontSize',10,...
    'ForegroundColor',[0 0
0],'FontWeight','bold','Position',...
    [145 90 50 20],'String','Open','Style','PushButton',...
    'Tag','pushbutton5');
uicontrol('Parent',fig1,'Callback','playsound','Units','Point',...
    'BackgroundColor',[0.8 0.8 0.8],'FontSize',10,...
    'ForegroundColor',[0 0
0],'FontWeight','bold','Position',...
    [145 60 50 20],'String','Sound','Style','PushButton',...
    'Tag','pushbutton6');
uicontrol('Parent',fig1,'Callback','sdetail','Units','Point',...
    'BackgroundColor',[0.8 0.8 0.8],'FontSize',10,...

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ภายใต้การสงวนลิขสิทธิ์ของสถาบันวิจัยและพัฒนาเทคโนโลยีสารสนเทศแห่งชาติ
 ไม่ว่ากรรมใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
'ForegroundColor',[0 0
0],'FontWeight','bold','Position',...
[145 30 50 20],'String','Detail','Style','PushButton',...
'Tag','pushbutton7');
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
***** Help *****
```

```
text=str2mat('Help Speech Recognition',...
    ' (1) Press Open Button For Select .wav',...
    ' (2) Press Sound Button For Listen To Sound',...
    ' (3) Press Detail Button For Sound Detail',...
    ' (4) Select Codebook Size',...
    ' (5) Press Process Button For Output',...
    ' (6) Select Graph To Plot',...
    ' (7) Press Plot Button For Plot Graph',...
    ' (8) Press Close For Exit Program');
set(edit1,'string',text);
```



```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Open Wave File %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
filename=[];
[filename pathname]=uigetfile('*.wav');
if filename ~=0
    filename1 = [pathname filename];
    [readwave,fs,bits,]=wavread(filename);
    numberwav = length(readwave);
    text=filename1;
    set(edit2,'string',text);
    msgbox('"Open Wavefile Complete." ','Message','help');
    cab = '';
    set(detail1,'string',cab);
    set(detail2,'string',cab);
    set(detail3,'string',cab);
else
    msgbox(' "Invalid Input" ','Error','error');
end

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Play Sound %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if filename ~=0
    if (bits==8)
        sound(readwave,fs);
    else
        msgbox('    "Not 8 Bits" ','Error','error');
    end
else
    msgbox(' "Please Open File.wav" ','Error','error');
end

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Show Detail %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
text2 = str2mat('',...
```

```

    ',...
    ',...
    ',...
    'Wave Name : ',...
    'Sampling Rate : ',...
    'Bits : ');
```

```

set(edit2,'string',text2);
set(detail1,'string',filename);
set(detail2,'string',fs);
set(detail3,'string',bits);
```



```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Process %%%%%%%%%%
if (male == 1)&(female == 0),
    if (csize_1 == 1)&(csize_2 == 0)&(csize_3 == 0),
        [output,s,S,z,w] = speechreg32m(filename);
    elseif (csize_1 == 0)&(csize_2 == 1)&(csize_3 == 0),
        [output,s,S,z,w] = speechreg64m(filename);
    elseif (csize_1 == 0)&(csize_2 == 0)&(csize_3 == 1),
        [output,s,S,z,w] = speechreg128m(filename);
    else
        msgbox('      " Please Select Codebook "', 'Error', 'error');
    end
elseif (male == 0)&(female == 1),
    if (csize_1 == 1)&(csize_2 == 0)&(csize_3 == 0),
        [output,s,S,z,w] = speechreg32f(filename);
    elseif (csize_1 == 0)&(csize_2 == 1)&(csize_3 == 0),
        [output,s,S,z,w] = speechreg64f(filename);
    elseif (csize_1 == 0)&(csize_2 == 0)&(csize_3 == 1),
        [output,s,S,z,w] = speechreg128f(filename);
    else
        msgbox('      " Please Select Codebook "', 'Error', 'error');
    end
else
    msgbox('      " Please Select Sex "', 'Error', 'error');
end

if output == 0,
    msgbox('"The Output Is " ZERO " <0> Thank You " ',...
        'Process Complete', 'help');
elseif output ==1,
    msgbox('"The Output Is " ONE " <1> Thank You " ',...
        'Process Complete', 'help');
elseif output ==2,
    msgbox('"The Output Is " TWO " <2> Thank You " ',...

```

```

        'Process Complete', 'help');
elseif output ==3,
    msgbox('"The Output Is " THREE " <3> Thank You " ',...
        'Process Complete', 'help');
elseif output ==4,
    msgbox('"The Output Is " FOUR " <4> Thank You " ',...
        'Process Complete', 'help');
elseif output ==5,
    msgbox('"The Output Is " FIVE " <5> Thank You " ',...
        'Process Complete', 'help');
elseif output ==6,
    msgbox('"The Output Is " SIX " <6> Thank You " ',...
        'Process Complete', 'help');
elseif output ==7,
    msgbox('"The Output Is " SEVEN " <7> Thank You " ',...
        'Process Complete', 'help');
elseif output ==8,
    msgbox('"The Output Is " EIGHT " <8> Thank You " ',...
        'Process Complete', 'help');
elseif output ==9,
    msgbox('"The Output Is " NINE " <9> Thank You " ',...
        'Process Complete', 'help');
else
    msgbox('
        " There Is No Output " ', 'Error', 'error');
end

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Plot Graph %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if (gplot_1 == 1)&(gplot_2 == 0)&(gplot_3 == 0)&(gplot_4 == 0)&...
    (gplot_5 == 0)&(gplot_6 == 0),

    fig2 = figure('Color',[0.8 0.8 0.8],...
                  'MenuBar','none',...
                  'Name','Plot Graph',...
                  'NumberTitle','off',...
                  'Position',[100 50 600 250],...
                  'Resize','on',...
                  'Tag','fig2');

    plot(readwave);
    title('Original Sound');
    ylabel('Amplitude');

elseif (gplot_1 == 0)&(gplot_2 == 1)&(gplot_3 == 0)&(gplot_4 == 0)&...
    (gplot_5 == 0)&(gplot_6 == 0),

    fig2 = figure('Color',[0.8 0.8 0.8],...
                  'MenuBar','none',...
                  'Name','Plot Graph',...
                  'NumberTitle','off',...
                  'Position',[100 50 600 250],...
                  'Resize','on',...
                  'Tag','fig2');

    plot(s);
    title('Detection By Energy');
    ylabel('Amplitude');

elseif (gplot_1 == 0)&(gplot_2 == 0)&(gplot_3 == 1)&(gplot_4 == 0)&...
    (gplot_5 == 0)&(gplot_6 == 0),

    fig2 = figure('Color',[0.8 0.8 0.8],...
                  'MenuBar','none',...
                  'Name','Plot Graph',...
                  'NumberTitle','off',...
                  'Position',[100 50 600 250],...
                  'Resize','on',...
                  'Tag','fig2');

    plot(S);
    title('Detection By Zerocrossing');
    ylabel('Amplitude');

elseif (gplot_1 == 0)&(gplot_2 == 0)&(gplot_3 == 0)&(gplot_4 == 1)&...
    (gplot_5 == 0)&(gplot_6 == 0),

    fig2 = figure('Color',[0.8 0.8 0.8],...
                  'MenuBar','none',...
                  'Name','Plot Graph',...
                  'NumberTitle','off',...
                  'Position',[100 50 600 250],...
                  'Resize','on',...
                  'Tag','fig2');

    plot(z);
    title('Normalize');
    ylabel('Amplitude');

```

```

elseif (gplot_1 == 0)&(gplot_2 == 0)&(gplot_3 == 0)&(gplot_4 == 0)&...
    (gplot_5 == 1)&(gplot_6 == 0),

    fig2 = figure('Color',[0.8 0.8 0.8],...
        'MenuBar','none',...
        'Name','Plot Graph',...
        'NumberTitle','off',...
        'Position',[100 50 600 250],...
        'Resize','on',...
        'Tag','fig2');

    plot(w);
    title('Weight Cepstrum');
    ylabel('Amplitude');

elseif (gplot_1 == 0)&(gplot_2 == 0)&(gplot_3 == 0)&(gplot_4 == 0)&...
    (gplot_5 == 0)&(gplot_6 == 1),

    fig2 = figure('Color',[0.8 0.8 0.8],...
        'MenuBar','none',...
        'Name','Plot Graph',...
        'NumberTitle','off',...
        'Position',[100 40 600 520],...
        'Resize','on',...
        'Tag','fig2');

    subplot(5,1,1);
    plot(readwave);
    title('Original Sound');
    ylabel('Amplitude');

    subplot(5,1,2);
    plot(s);
    title('Detection By Energy');
    ylabel('Amplitude');

    subplot(5,1,3);
    plot(S);
    title('Detection By Zerocrossing');
    ylabel('Amplitude');

    subplot(5,1,4);
    plot(z);
    title('Normalize');
    ylabel('Amplitude');

    subplot(5,1,5);
    plot(w);
    title('Weight Cepstrum');
    ylabel('Amplitude');

else msgbox('          Error Occur','Error','error');

end

```