บทที่ 2

ทฤษฎีแนวคิดและงานวิจัยที่เกี่ยวข้อง

2.1 คำนำ

การรู้จำเสียงพูด (Speech Recognition) การงานวิจัยหนึ่งในด้านคอมพิวเตอร์ที่ได้รับความสนใจ และพัฒนา มาเป็นเวลานาน สัญญาณเสียงเป็นสัญญาณที่เกิดจากการสั่นของวัตถุ และเคลื่อนที่ผ่านตัวกลาง ในรูปแบบต่างๆ และเสียงพูดก็เป็นการทำให้เส้นเสียงเกิดการเคลื่อนไหว ทำให้เกิดเป็นพลังงานส่งผ่านอากาศ ไปยัง หู หรืออุปกรณ์รับสัญญาณ โดยพื้นฐานของสัญญาณเสียงและเป็นพลังงานที่มีการเปลี่ยนแปลงขึ้นลงโดยมีความเร็วในการเปลี่ยนแปลง แตกต่างกันไปตามสัญญาณเสียง เราเรียกว่า ความถี่ของสัญญาณ (Frequency) และเสียงต่างๆ ต่างก็มีความถี่ที่แตกต่างกันไป เช่น เสียงดนตรี , เสียงนก และเสียงพูด เสียงส่วนใหญ่ มักจะประกอบด้วยความถี่หลายๆ ความถี่ผสมกัน และส่งออกมา เพื่อให้เกิดความแตกต่างของเสียงมากขึ้น นอกจากความถี่ที่ผสมกันออกมาแล้ว ระดับความแรงของความถี่ต่างๆ ก็มีระดับที่ไม่เท่ากัน และเกิดขึ้นในเวลาที่ต่างกันด้วย ทำให้เสียงที่มีการสร้างขึ้น มีความหลากหลาย โดยเฉพาะเสียงพูดของมนุษย์ โดยมนุษย์สามารถเสียงเสียงได้มากมาย

ดังนั้นในการศึกษาระบบรู้จำเสียงพูด จึงจำเป็นต้องรู้จักรูปแบบของเสียงพูด ต่างๆ ที่ถูกสร้างขึ้น และรูปแบบของการพูด โดยเราสามารถจำแนะรูปแบบการพูดเป็น 4 แบบคือ

2.1.1 ผู้พูดคนคนเดิม หรือผู้พูดคนเดียวกัน และวิธีการพูดจะต้องพูดแบบคำแยกจากกัน (discrete utterance) คือการพูดเป็นหน่วยเสียง หน่วยพยางค์ หน่วยคำ หรือหน่วยวลี สิ่งที่จะต้องทำก็คือ เมื่อฟังจากเสียงผู้พูดแล้วจะบันทึกรูปแบบของเสียงตามวิะของเสียงตามวิธีของเครื่องเอง และเครื่องจะรู้เองว่าหากพูดให้ฟังอีกครั้งจะเหมือนกับคำที่ผู้พูดคนเดิมไว้สอนไว้ด้วยหรือไม่ โดยอาศัยขั้นตอนดังนี้

เมื่อผู้ใช้ต้องการสอนให้เครื่องรู้จักคำ โดยพูดผ่านไมโครโฟน สัญญาณเสียงซึ่งขณะนี้อยู่ในรูปของสัญญาณไฟฟ้า จะถูกเปลี่ยนให้เป็นสัญญาณดิจิตอลก่อน จากนั้นก็จะนำข้อมูลดิจิตอลไปคำนวณหาค่าสัมประสิทธิ์ชุดหนึ่ง ซึ่งจะบอกถึงลักษณะเฉพาะของคำนั้น ๆ และเก็บข้อมูลต้นแบบนี้ไว้ในหน่วยความจำ หรือ ในแผ่นแม่เหล็กก็ได้ข้อมูลต้นแบบของคำ ๆ นี้เรียกว่า เทมเพลทต้นแบบ (templates) ขบวนการจนถึงขั้นนี้เรียกว่า การสอนเครื่องให้ให้รู้จักจำคำ (training) การทำงานของเครื่องจะมีความเชื่อถือได้มากน้อยแค่ไหนขึ้นอยู่กับยุทธวิธี (algorithm) ที่ใช้ในการวิเคราะห์สำหรับความจุของจำนวนคำ ขึ้นกับขนาดของหน่วยความจำ (memory) เครื่องในระดับไมโครคอมพิวเตอร์ 8 บิท สามารถจำได้นับ 100 คำ เวลาของการค้นหาคำที่รู้จักแล้ว จะเพิ่มขึ้นตามจำนวนคำที่สอนไว้ด้วย ถ้ามีจำนวนคำมากยุทธวิธีที่จะใช้ค้นหาคำให้ได้รวดเร็วก็จะยุ่งยากและซับซ้อนขึ้น

2.1.2 ผู้พูดเป็นใครก็ได้ วิธีการพูดยังเป็นแบบแยกจากกัน (discrete unterrance) สำหรับเครื่องไมโครคอมพิวเตอร์ขนาดเดียวกัน วิธีในแบบที่ 2 นี้ จะสามารถจำคำได้น้อยกว่าวิธีในแบบที่ 1 มาก วิธีนี้ไม่ต้องการเทมเพลทต้นแบบ จะถูกโปรแกรมไว้ล่วงหน้าแล้วใช้ยุทธวิธีจัดเข้ากลุ่มเพื่อที่จะฟังเสียงคนใดก็ได้

2.1.3 สอนให้เครื่องได้รู้จักคำในลักษณะของหน่วยเสียง หรืออย่างมากหน่วยพยางค์ เป็นเทคนิคที่เริ่มใช้มาก่อน ใช้กับระบบที่ไม่แพง ตัวเทมเพลทต้นแบบจะถูกเก็บไว้เป็นหน่วยเสียงหรือหน่วยพยางค์เท่านั้น ต่างกับ 2 แบบแรก ซึ่งจะเป็นหน่วยเสียง หน่วยพยางค์ หน่วยคำ หรือหน่วยวลีก็ได้

2.1.4 ผู้พูดสามารถพูดแบบต่อเนื่องเหมือนการพูดปกติ เครื่องจะต้องสามารถรู้จัก และจำแนกคำที่มีการออกเสียงเชื่อมต่อกัน การพูดแบบนี้สำหรับมนุษย์ รู้สึกเป็นของง่ายเพราะไม่ต้องตระเตรียมเป็นพิเศษ แต่สำหรับเครื่องเป็นสิ่งที่ยากมาก เพราะการพูดแบบต่อเนื่องทำความยุ่งยากให้กับเครื่องมาก เวลาที่มีเสียงควบกล้ำ (coarticulation) การพูดแบบนี้ไม่สามารถหาเส้นแบ่งเขตระหว่างคำ หรือระหว่างพยางค์ได้อย่างแม่นยำ ดังนั้นการรู้จักคำโดยวิธีการเอาเทมเพลทที่กำลังได้ยินไปเปรียบเทียบกับเทมเพลทต้นแบบจึงไม่ใช่ของง่าย วิธีการนี้ใช้ได้กับคอมพิวเตอร์ระดับมินิขึ้นไปเท่านั้น

งานด้าน การรู้จำเสียงพูด (Speech Recognition) อยู่บนพื้นฐานความรู้ด้าน เสียงพูดและการพูด (Speech and Spoken language) ความเข้าเกี่ยวกับการพูดมีการพัฒนาเป็นอย่างมากในช่วง 2 ทศวรรษที่ผ่านมา โดยในหัวข้อที่มีการศึกษาได้แก่ การสั่งด้วยเสียง (Speech Coding) และ การสังเคราะห์เสียงจากข้อความ (Text to Speech Synthesis) โดยมีจุดมุ่งหมายเพื่อสร้างระบบ "Human-Machine Communication by voice"

การรู้จำเสียงพูด (Speech Recognition) คือขบวนการที่อาศัยพื้นฐานของ การจัดกลุ่ม (Pattern Classification) โดยมีจุดมุ่งหมายเพื่อจัดการกับ รูปแบบข้อมูลเข้า (Input Pattern) ของ สัญญาณเสียง (Speech Signal) และการจัดหมวดหมู่ของสัญญาณและจัดเก็บซึ่งเป็นขบวนการเรียนรู้ (Learning) และถ้าเป็นรูปแบบ (Pattern) ที่ผ่านการเรียนรู้มาแล้ว ก็ทำการเปรียบเทียบรูปแบบของข้อมูล เพื่อหาข้อมูลที่ตรงกันมากที่สุดในฐานความรู้ ความยุ่งยากในระบบ การรู้จำเสียงพูด (Speech Recognition) คือสัญญาณเสียงพูด เป็นสัญญาณที่มีระดับของสัญญาณที่แตกต่างกันหลายระดับ ขึ้นอยู่กับผู้พูดแต่ละคนสัญญาณเสียงที่แตกต่างกันหรือเหมือนกัน ไม่ได้มีความหมายว่าเป็นสิ่งเดียวกันหรือแตกต่างกัน ความเร็วในการพูดก็มีผลต่อรูปแบบของสัญญาณเสียง นอกจากนั้นความแตกต่างของคำพูดจากเงื่อนไขสภาพแวดล้อมและเงื่อนไงของระบบเสียง ซึ่งระบบจะต้องเป็นผู้จัดการกับความหลากหลายเหล่านี้ของคำพูดเช่นการพูดคำว่า "father" กับ "farther" และสำเนียงของผู้คนในแต่ละท้องถิ่น

2.2 พื้นฐานของระบบรู้จำเสียง

ในการศึกษาในเรื่องของ การรู้จำเสียงพูด (Speech Recognition) จะอยู่บนพื้นฐาน 3 เรื่อง คือ

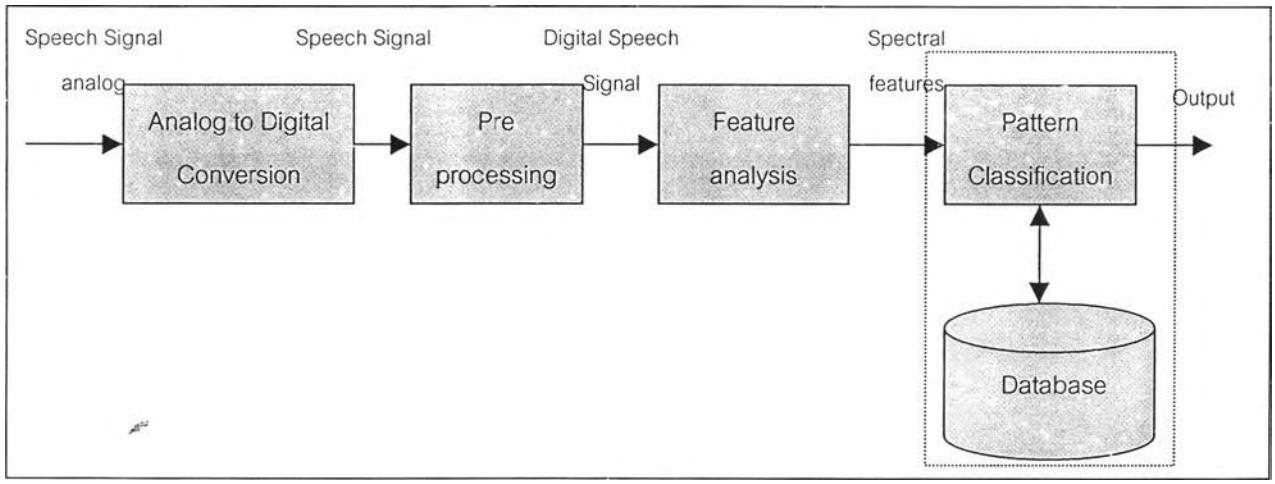
2.2.1 ข้อมูลในสัญญาณเสียงพูด (The information in the speech signal) ซึ่งเราสามารถจัดข้อมูลให้อยู่ในรูปแบบต่างๆ ที่สามารถแสดงถึงรายละเอียดสำคัญของสัญญาณเสียง โดยมากมักจะใช้เทคนิคที่เรียกว่า "Short time amplitude spectrum of the speech waveform" ซึ่งเทคนิคนี้จะสามารถหาข้อมูลสำคัญในข้อมูลสัญญาณเสียงเมื่อได้ข้อมูลสำคัญแล้วจึงนำไปสู่ขั้นตอน การเปรียบเทียบชุดข้อมูล (Pattern matching)

2.2.2 รายละเอียดของสัญญาณเสียง (The contents of the speech signal) คือขบวนการจัดการกับข้อมูลสำคัญที่ได้จากการวิเคราะห์ เพื่อการอธิบายรูปแบบของสัญญาณต้นฉบับ ซึ่งสามารถนำกลับมาเพื่ออธิบายสัญญาณได้ ซึ่งอาจใช้สัญญาลักษณะ หรือตัวอักษรต่างๆ เพื่อแทนข้อมูลและลำดับของสัญญาณเสียง เช่นในระบบ สังเคราะห์เสียงจากตัวอักษร (Text to speech Synthesis) ซึ่งสามารถเก็บเสียงที่ต้องสร้างโดยใช้สัญญาลักษณ์แทนเสียง และลักษณะการออกเสียง (Phonetic) หรือในพจนานุกรมใช้ตัวอักษรเพื่อแสดงลักษณะของคำต่างๆ

2.2.3 การรู้จำเสียงพูด (Speech Recognition) คือขบวนการจดจำและรู้จัก รูปแบบของข้อมูล (Pattern) ซึ่งในการทำความเข้าใจคำพูดของมนุษย์ นับได้ว่าเป็นเรื่องที่ยากมากเพราะต้องเข้าใจใน หลักไวยากรณ์ ความหมาย และความยุ่งยากในโครงสร้างภาษา เพราะถ้าใช้เพียงอย่างใดอย่างหนึ่งเพื่อให้เข้าใจคำพูดของมนุษย์นั้น อาจให้ความหมายที่ไม่ถูกต้องได้

2.3 โครงสร้างระบบรู้จำเสียงพูด (Speech Recognition Structure)

ระบบวิเคราะห์และรู้จำเสียงพูด เราสามารถแสดงโครงสร้างของระบบได้ดังรูป 2.1 ซึ่งเป็นรูปแบบพื้นฐานที่ได้รับความนิยมใช้กันในงานวิจัย ด้านการรู้จำเสียงพูด ทั่วไป



รูปที่ 2.1 โครงสร้างระบบวิเคราะห์และรู้จำเสียงพูด

ในการทำงานของระบบสามารถแบ่งเป็นส่วนๆ ได้ดังนี้

2.3.1 การแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิตอล (Analog to Digital Conversion)

2.3.2 การจัดรูปแบบสัญญาณก่อนการประมวลผล (Pre Processing)

2.3.3 การวิเคราะห์คุณสมบัติ (Feature analysis) ซึ่งมีอยู่หลายวิธี เช่น

2.3.3.1 การวิเคราะห์เสียงในด้านเวลา (Time Domain models for speech processing)

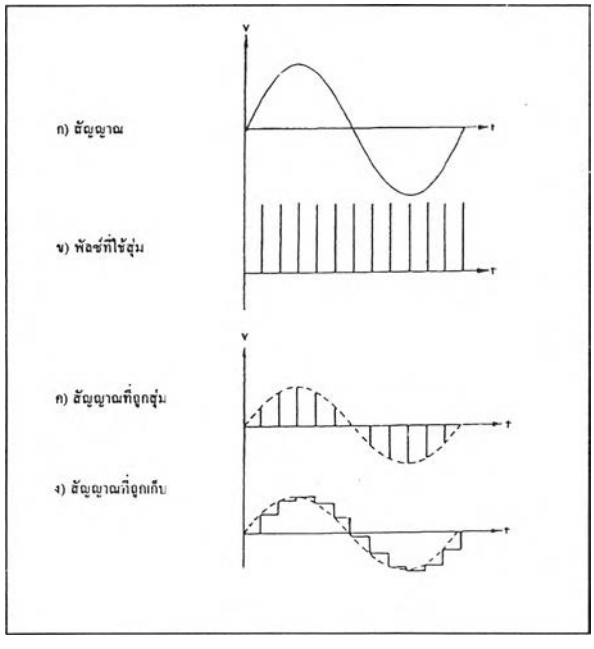
2.3.3.2 การวิเคราะห์โดยใช้ ฟูริเยอร์ Short-time Fourier analysis

2.3.3.3 การวิเคราะห์แบบการคาดเดาเชิงเส้น (Linear Predictive coding of speech)

2.3.4 การจัดกลุ่มรูปแบบที่เข้ากันได้ (Pattern classification) ทำหน้าที่รับข้อมูลคุณลักษณะของเสียงที่ได้จากการประมวลผลมาจัดการเรียนรู้ในรูปแบบของชุดข้อมูล (Pattern) และตรวจสอบข้อมูลว่าเป็นเสียงอะไร มีการสอนไว้หรือไม่ ถ้ามีการสอนไว้คำตอบคือเสียงอะไร ซึ่งในงานด้าน Speech Recognition ทฤษฎีที่ได้รับการยอมรับคือ Multi-layer feedforward neuron network

2.4 การแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิตอล (Analog to Digital Conversion)

รูปแบบสัญญาณไฟฟ้าที่เราพบเห็นและคุ้นเคยในชีวิตประจำวันจะอยู่ในรูปแบบของสัญญาณที่ต่อเนื่อง หรือที่เรียกว่าสัญญาณอนาล็อก ซึ่งแต่เดิมการจะนำเอาสัญญาณไฟฟ้าดังกล่าวมาประมวล (Processed) จะกระทำในแบบ อนาล็อกนั่นเอง แต่เมื่อเริ่มมีเทคนิคการประมวลสัญญาณทางดิจิตอลได้รับการพิจารณา เนื่องจากพบว่า ในรูปแบบของดิจิตอลการประมวลผล การสื่อสารและการแสดงผล สามารถกระทำได้ง่ายกว่า และมีประสิทธิภาพมากกว่า ดังนั้น การเปลี่ยนรูปของสัญญาณ (conversion) จึงได้มีความจำเป็นขึ้น จากสัญญาณอนาล็อกที่มีอยู่ตามธรรมชาติถูกเปลี่ยนมาเป็นสัญญาณดิจิตอล โดยวงจรแปลงสัญญาณ อนาล็อกเป็นสัญญาณดิจิตอล (Analog to Digital Cconverters) หรือ ADC และนำมาประมวลโดยตัวประมวลทางดิจิตอล (Digital processors) เช่น คอมพิวเตอร์ หรือ Digital Circuit



รูปที่ 2.2 การสุ่มสัญญาณ

ในระบบการสุ่ม สัญญาณอนาล็อกจะถูกสุ่มเป็นระยะคงที่ตามรูปที่ 2.2 กลุ่มของสัญญาณสุ่มจะแทน แบนด์วิทธ์ที่ทำงานด้วยความเร็วสูง ซึ่งจะทำการตัดต่อสัญญาณอนาล็อกในช่วงเวลาอันสั้น ผลของการสุ่มสัญญาณด้วยความเร็วจะเสมือนกับการคูณขบวนสัญญาณพัลซ์แคบๆ กับสัญญาณอนาล็อก ซึ่งจะได้เป็นสัญญาณที่เกิดการ มอดูเลท ระหว่างขบวนพัลซ์กับสัญญาณอนาล็อก ดังแสดงในรูป 2.2 (ค) โดยสัญญาณ

อนาล็อก จะขึ้นมาบนขบวนพัลซ์ ถ้าหากเอาสวิทช์และตัวเก็บประจุแทนสวิทช์แล้ว สัญญาณอนาล็อกที่ถูกสุ่มจะถูกเก็บไว้ในตัวเก็บประจุ จนกว่าสัญญาณค่าใหม่ถูกสุ่มเข้ามา ซึ่งลักษณะของเอาท์พุทที่แสดงในรูป 2.2 (ง) มีปัญหาที่ว่าอัตราการสุ่มสัญญาณนั้นควรมีขนาดเท่าใดนั้นจะไม่ทำให้ข้อมูลสูญเสียไปเมื่อสัญญาณนั้นถูกเปลี่ยนกลับมาเป็นเช่นเดิม คำตอบก็คือขึ้นอยู่กับความถี่ของสัญญาณอนาล็อกและทฤษฎีของการสุ่มกล่าวไว้ว่า "ถ้าสัญญาณต่อเนื่องซึ่งมีความถี่และอาร์โมนิคไม่เกิน f_c แล้วสัญญาณดังกล่าวจะสามารถเปลี่ยนกลับมาเป็นอย่างเดิมโดยไม่สูญเสียรายละเอียดหรือผิดเพี้ยนไป ถ้าอัตราการสุ่มไม่น้อยกว่า $2f_c$ ต่อวินาที"

2.5 การจัดรูปแบบสัญญาณก่อนการประมวลผล (Pre Processing)

สัญญาณเสียงที่จะส่งเข้าไปในขั้นตอน feature measurement ต้องผ่านการพรีโพรเซสซิ่งก่อน ซึ่งจะทำการเตรียมและปรับข้อมูลให้เหมาะสมกับระบบการรู้จำเสียงพูด การพรีโพรเซสซิ่งที่ใช้ประกอบด้วย การตัดหัวท้ายคำ (end point detection) และการนอร์แมลไลซ์ (normalization)

2.5.1 การตัดหัวท้ายคำ (End Point Detection)

การตัดหัวท้าคำ เป็นกระบวนการค้นหาช่วงที่เป็นเสียงพูดจากเสียงที่ได้จากการบันทึก นั่นคือ การแยกส่วนที่เป็นเสียงพูดจากส่วนที่เป็นเสียงพื้นหลัง (background sound) ขั้นตอนนี้เป็นขั้นตอนที่สำคัญ (Rabiner and Levinson ,1981) เพราะ

- ความผิดพลาดในการตัดหัวท้ายคำ จะทำให้ความน่าจะเป็นของความผิดพลาดในการรับรู้จำเสียงพูดเพิ่มขึ้น
- การตัดหัวท้ายคำที่ถูกต้อง ช่วยให้การคำนวณทั้งหมดของระบบต่ำสุด

การตัดหัวท้ายคำมีวิธีการหลัก ๆ ดังนี้

2.5.1.1 การตัดหัวท้ายคำ โดยใช้ แอมพลิจูด (เสาวลักษณ์ อารีย์พงศา ,2538) เมื่อสัญญาณมีค่า แอมพลิจูดมากกว่าค่าที่กำหนดไว้เท่ากับจำนวนครั้งที่กำหนด จะให้จุดนั้นเป็นจุดเริ่มต้นของเสียงพูด และทำเช่นเดียวกันในส่วนท้ายของเสียงที่บันทึกมาเพื่อกาจุดสิ้นสุด ข้อดีของวิธีนี้ คือ ใช้การคำนวณง่าย ๆ และใช้เวลาในการคำนวณน้อยมาก ข้อเสียของวิธีนี้ คือ จะตัดคำผิดพลาด เมื่อมีสัญญาณรบกวนที่มีแอมพิจูดสูงในบริเวณหัวหรือท้ายคำ เช่น เสียงหายใจ

2.5.1.2 การตัดหัวท้ายคำโดยใช้ค่าพลังงาน (Rabiner and Levinson , 1981) วิธีนี้ใช้คอนทัวร์ (contour) ของพลังงานในแต่ละส่วนย่อย เพื่อหาจุดที่มีพลังงานมากกว่าระดับที่กำหนดไว้ติดต่อกันนานกว่าคาบเวลาที่กำหนด จุดเริ่มต้นของเสียงพูดจะอยู่ก่อนจุดที่ตรวจพบด้วยระดับพลังงาน เท่ากับคาบเวลาคงที่ค่าหนึ่ง ข้อดีของวิธีนี้ คือ สามารถลดการตัดคำผิดพลาดเมื่อมีสัญญาณรบกวนที่มี แอมพลิจูดสูง ข้อเสียของวิธีนี้ คือ จุดเริ่มต้นที่คำนวณได้อาจคลาดเคลื่อนจากจุดเริ่มต้นที่แท้จริงของเสียงพูด

2.5.1.3 การตัดหัวท้ายคำโดยใช้ค่าพลังงานและอัตราการตัดค่าศูนย์ (zero – crossing rate) (Furui ,1989) เหมือนกับการตัดหัวท้ายคำ โดยใช้ค่าพลังงาน แต่มีการปรับปรุงการหาจุดเริ่มต้นของ

เสียงพูด โดยใช้อัตราการตัดค่าศูนย์แทนการใช้คาบเวลาคงที่ ทำให้สามารถหาจุดเริ่มต้นได้ถูกต้องมากขึ้น ซึ่งก็ต้องแลกเปลี่ยนด้วยเวลาที่ใช้ในการคำนวณอัตราการตัดค่าศูนย์

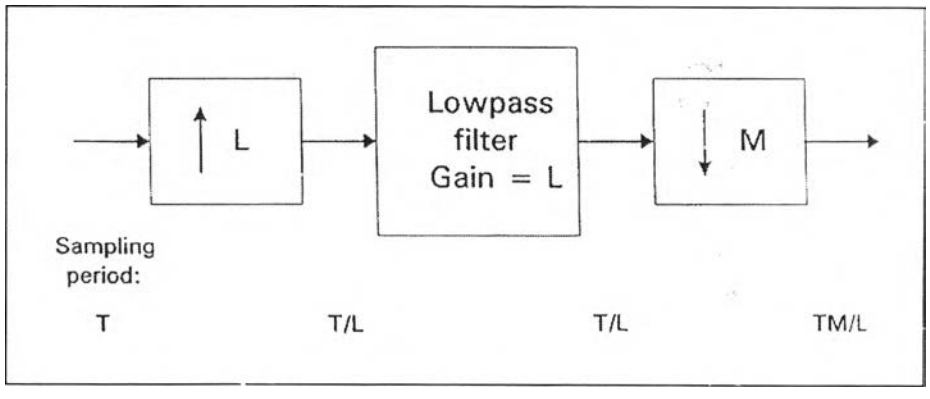
ในวิทยานิพนธ์ นี้เลือกใช้การตัดหัวท้ายคำโดยใช้ค่าพลังงาน เพราะวิธีนี้สามารถ แก้ปัญหาการตัดคำผิดพลาดได้ โดยที่ไม่ใช้เวลาในการคำนวณมากเกินไป ถึงแม้ว่าจุดเริ่มต้นของเสียงพูดที่คำนวณได้อาจคลาดเคลื่อนไปบ้าง แต่ก็สามารถแก้ไขได้โดยใช้การประมาณค่าคาบเวลาที่เหมาะสมกับกลุ่มคำที่ต้องการรู้จำ

2.5.2 การนอร์แมลไลซ์ (Normalization)

เป็นกระบวนการที่ทำให้สัญญาณเสียงพูดแต่ละคำมีจำนวนจุดสัญญาณในแกนเวลาเป็นจำนวนเท่ากัน กระบวนการนี้เป็นกระบวนการที่จำเป็นเพราะเสียงพูดแต่ละคำมีความยาวไม่เท่ากัน แต่นิวรอลเน็ตเวิร์กมีจำนวนโหนดในระดับข้อมูลเข้าคงที่ การนอร์แมลไลซ์มีวิธีการหลัก ๆ ดังนี้

2.5.2.1 การประมาณค่าในช่วงเชิงเส้น (linear interpolation) จะทำการประมาณค่าแอมพลิจูดของสัญญาณที่จุดไม่ทราบค่าจากความสัมพันธ์เชิงเส้นของจุดสัญญาณเดิมที่ได้จากการบันทึกเสียงพูดที่อยู่ล้อมรอบจุดที่ไม่ทราบค่า ข้อดีของวิธีนี้ คือคำนวณได้ง่าย ข้อเสียคือทำให้เกิดการเคลือบแฝง (aliasing) ในสัญญาณที่ประมาณค่า

2.5.2.2 การประมาณค่าโดยใช้การเปลี่ยนอัตราการชักตัวอย่าง (sampling rate) (Oppenheim, 1989) วิธีนี้มีขั้นตอนดังแสดงในรูปที่ 2.3 โดยนำสัญญาณเสียงพูดมาเพิ่มอัตราการชักตัวอย่างขึ้น L เท่า โดยการเพิ่มจุดศูนย์ (zero – packing) ระหว่างแต่ละจุดสัญญาณเดิม จากนั้นนำสัญญาณที่ได้ไปผ่านวงจรกรองแบบผ่านต่ำ แล้วลดอัตราการชักตัวอย่างลง M เท่า โดยที่ M เป็นจำนวนจุดสัญญาณเดิมที่ได้จากการบันทึกและ L เป็นจำนวนจุดสัญญาณที่ต้องการ ข้อดีของวิธีนี้คือ ป้องกันการเกิดการเคลือบแฝงได้ ข้อเสียของวิธีนี้คือใช้หน่วยความจำและเวลาในการคำนวณมาก



รูปที่ 2.3 การนอร์แมลไลซ์โดยใช้การเปลี่ยนอัตราการชักตัวอย่าง

2.6 การวิเคราะห์คุณลักษณะของสัญญาณเสียง

ในระบบวิเคราะห์และรู้จำเสียงพูด การวิเคราะห์หารูปแบบที่เป็นลักษณะเฉพาะของเสียง ต่างๆ นับว่าเป็นขั้นตอนสำคัญอย่างยิ่ง เนื่องจากการที่เราให้ให้คอมพิวเตอร์เปรียบเทียบ ชุดข้อมูลนั้น เราจำเป็นต้องอธิบายลักษณะของข้อมูล ให้ชัดเจน ซึ่งในขั้นตอนนี้เราใช้แขนงวิชา การประมวลผลสัญญาณเชิงเลข (Digital Signal Processing : DSP) และในการประมวลผลสัญญาณ ก็มีการศึกษาไว้หลายวิธี ซึ่งแต่ละวิธีก็ มีจุดประสงค์เพื่ออธิบาย และ แยกแยะความแตกต่างของสัญญาณ ที่ประมวลผล โดยแต่ละวิธี ก็มีวิธีการที่แตกต่างกัน ความสามารถ และจุดเด่น จุดด้อย แตกต่างกัน เวลาที่ใช้ในการประมวลผลก็แตกต่างกัน วิธีการวิเคราะห์สัญญาณ (Signal analysis methods) มีหลายวิธีใหญ่ ๆ ดังนี้

วิธีที่ 1 การวิเคราะห์เสียงโดยวิธีโดเมนเวลา (Time Domain Analysis Methods)

วิธีที่ 2 การวิเคราะห์จุดตัดศูนย์ของสัญญาณ (Zero crossing Analysis Methods)

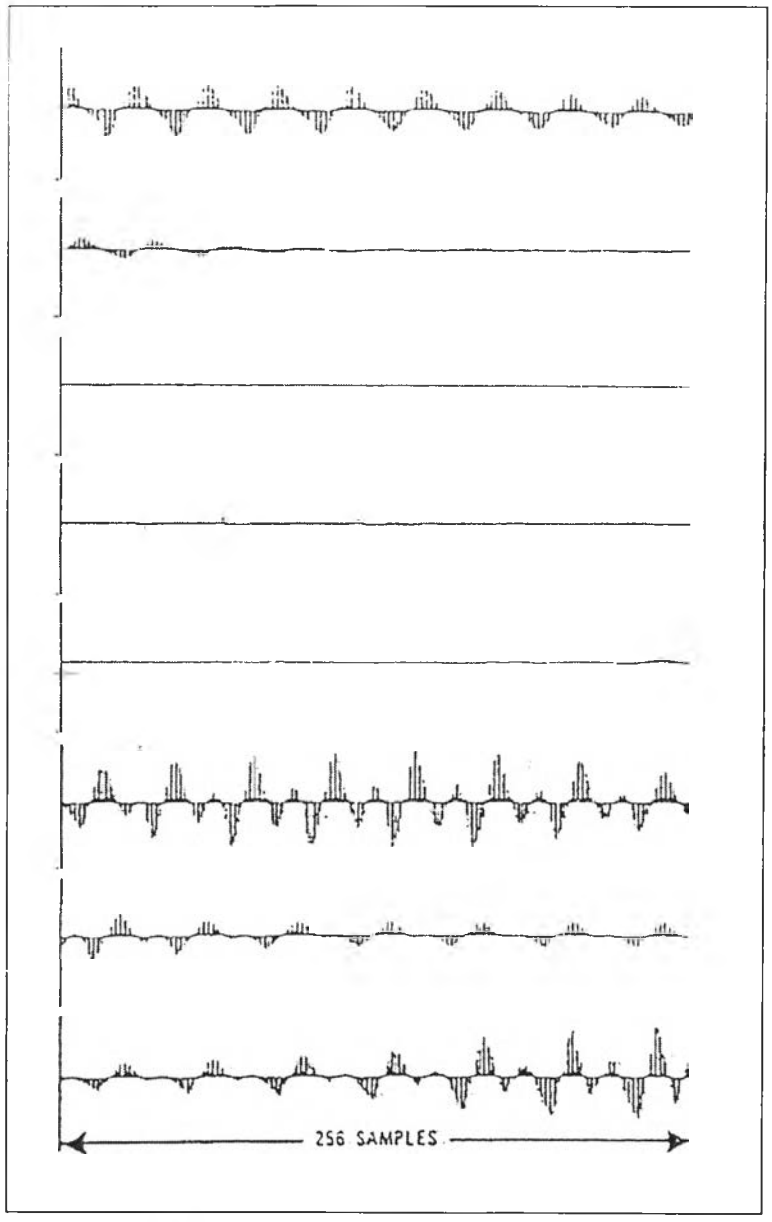
วิธีที่ 3 การวิเคราะห์โดยวิธีชอร์ตไทม์สเปคตรัม (Short-Time Spectrum Analysis Methods)

วิธีที่ 4 การวิเคราะห์แบบโฮโมมอร์ฟิค (Homomophic Speech Processing)

วิธีที่ 5 การวิเคราะห์แบบการคาดเดาเชิงเส้น (Linear Predictive Analysis)

2.6.1 การวิเคราะห์เสียงโดยวิธีโดเมนเวลา (Time Domain Analysis)

การวิเคราะห์สัญญาณเสียงที่แปรตามเวลา สัญญาณเสียงจะถูกแทนด้วยลำดับการแซมปลิ้ง ประมาณ 8000 ครั้ง/วินาที ดังรูป 2.4 ซึ่งแสดงให้เห็นถึงคุณสมบัติของเสียงที่เปลี่ยนตามเวลา ดังตัวอย่าง มีการเปลี่ยนเสียงระหว่างมีสัญญาณและไม่มีสัญญาณเสียง การเปลี่ยนแปลงของแอมปลิจูดสูงสุด (peak Amplitude) และการเปลี่ยนแแปลงของความถี่พื้นฐานการเปลี่ยนแปลงทั้งหมดจะเห็นอย่างชัดเจนเมื่อพล็อตรูปคลื่น การแซมปิ้งทางด้านเวลา ซึ่งแทนลักษณะของสัญญาณเสียงได้



รูปที่ 2.4 รูปคลื่นของการสุ่มสัญญาณเสียง

โดยมากการวิเคราะห์สัญญาณมักจะใช้วิธี short-time Fourier ซึ่งสามารถแทนด้วยสมการดังนี้

$$Q_n = \sum_{m=-\infty}^{\infty} T[x(m) \bullet w(n-m)] \qquad (2.1)$$

สัญญาณเสียงจะถูกแปลงรูปโดย ซึ่งอาจจะเป็นแบบเชิงเส้นหรือไม่เชิงเส้นขึ้นอยู่กับการตั้งค่าของพารามิเตอร์ ซึ่งผลคือ การคูณตามลำดับการแซมปลิ้งที่เวลา n ใด ๆ

short-time energy ของสัญญาณเสียงสามารถหาได้โดย

$$E = \sum_{n=-\infty}^{\infty} x^2(m) \qquad (2.2)$$

ถ้าเป็นสัญญาณเสียงที่ถูกแซมปลิ้ง

$$E(n) = \sum_{m=n-N+1}^{\infty} x^2(m) \qquad (2.3)$$

ซึ่งมันจะหมายถึง short-time energy ที่ n จะเป็นการรวมของ N แซมปลิ้ง ตั้งแต่ n-N+1 จนถึง n

short-time energy และขนาดเฉลี่ย เราจะสังเกตเห็นว่าแอมปลิจูดของสัญญาณเสียงจะเปลี่ยนตามเวลา และแอมปลิจูด ขณะไม่มีเสียงจะต่ำกว่าขณะมีเสียงแต่ short-time energy จะให้ผลของการเปลี่ยนแปลงแอมปลิจูดไว้แล้ว ซึ่งสามารถเขียนได้

$$E(n) = \sum_{n=-\infty}^{\infty} [x(m) \bullet w(n-m)]^2$$

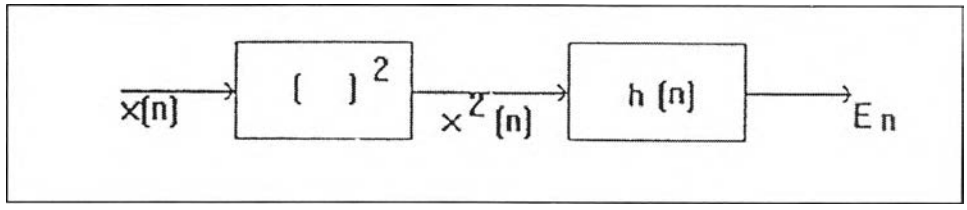
หรือ

$$E(n) = \sum_{n=-\infty}^{\infty} x^2(m) \cdot h(n-m) \qquad (2.4)$$

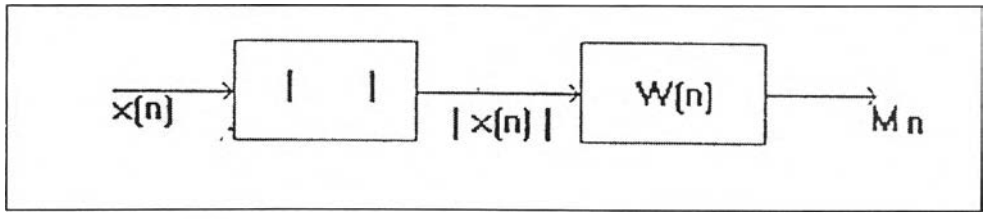
โดยที่

$$h(n) = w^2(n)$$

ซึ่งสามารถเขียนเป็นบล็อกไดอะแกรมได้ดังรูป



รูปที่ 2.5 บล็อกไดอะแกรมของ Short Time Energy



รูปที่ 2.6 บล็อกไดอะแกรมของ Short Time Average Energy

จากสมการข้างบน จะมีความไวสูงในกรณีสัญญาณมีขนาดใหญ่มาก เนื่องจากสัญญาณจะถูกยกกำลัง 2 ซึ่งมีวิธีแก้ คือ

$$M(n) = \sum_{m=-\infty}^{\infty} |x(m)| \cdot w(n-m) \qquad (2.5)$$

2.6.2 การวิเคราะห์จุดตัดศูนย์ของสัญญาณ

ในสัญญาณดิสครีท จุดตัดศูนย์จะเกิดขึ้นเมื่อเกิดการเปลี่ยนเครื่องหมาย ซึ่งอัตราการเกิดจุดตัดศูนย์จะขึ้นอยู่กับความถี่ของสัญญาณเสียงตัวอย่างเช่น สัญญาณไซน์ความถี่ Fo มีอัตราการแซมปลิ้งเท่ากับ FS เพราะฉะนั้น

จุดตัดศูนย์ = 2Fo/FS จุดตัดต่อวินาที

ดังนั้นอัตราเฉลี่ยของจุดตัดศูนย์จะเป็นหลักการหนึ่งในการคาดเดาความถี่ของรูปปคลื่นไซน์ สัญญาณเสียงเป็นสัญญาณที่เป็นแถบความถี่ การแสดงถึงอัตราจุดตัดศูนย์เฉลี่ยจะมีความแน่นอนน้อยมาก แต่อย่างไรก็ตามก็สามารถคาดเดาคุณสมบัติคร่าว ๆ ได้ ซึ่งก่อนอื่นเราควรศึกษาหลักการของจุดตัดศูนย์ก่อน ซึ่งมีสมการดังนี้

$$Z(n) = \sum_{m=-\infty}^{\infty} |\mathrm{sgn}[x(m)] - \mathrm{sgn}[x(n-1)]| \cdot w(n-m) \qquad (2.6)$$

เมื่อ

$$\mathrm{sgn}[X(n)] = \begin{cases} 1 & ; X(n) \geq 0 \\ -1 & ; X(n) < 0 \end{cases}$$

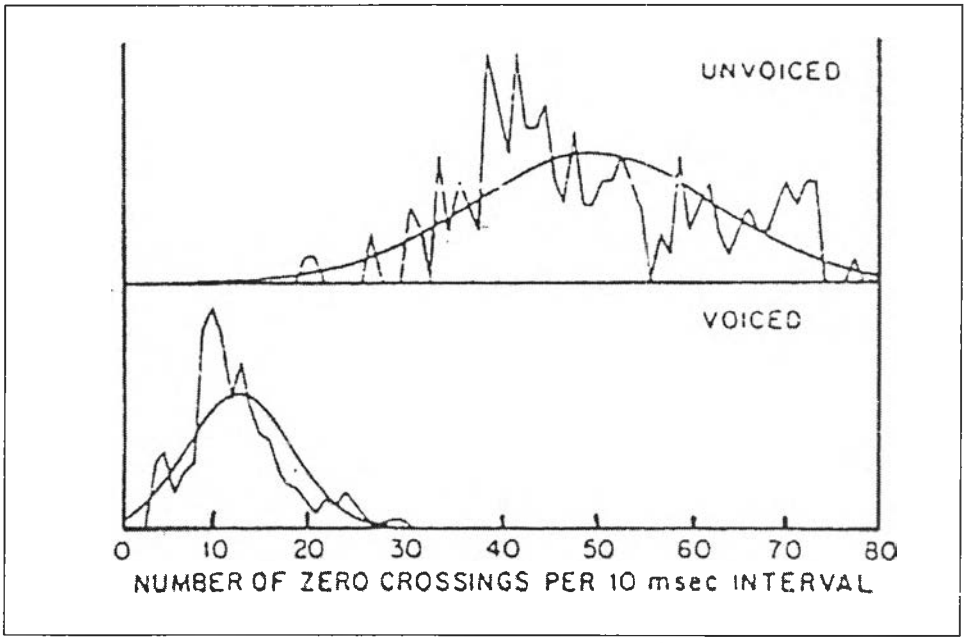
และ

$$W(n) = \begin{cases} 1/2N & ; 0 \leq n \leq N-1 \\ 0 & ; \textit{กรณีอื่นๆ} \end{cases}$$

จากรูปแสดงการหาอัตราจุดตัดศูนย์เฉลี่ย ซึ่งประกอบด้วย การหาพลังงาน การหาค่าขนาดเฉลี่ย โดยค่าที่ได้อยู่ใน Z(n) ซึ่งจะต้องหารด้วย N ซึ่งจะกลายเป็นอัตราเฉลี่ย

การนำเอาอัตราจุดตัดศูนย์เฉลี่ยมาประยุกต์ใช้งานกับสัญญาณเสียงนั้น เราทราบว่าสัญญาณเสียงนั้นจะมีความถี่ต่ำกว่า 3KHz ขณะที่ไม่มีสัญญาณเสียงเราจะพบว่าจะมีสัญญาณรบกวนที่ความถี่ที่สูงกว่า ดังนั้น สรุปว่าถ้าอัตราจุดตัดศูนย์มีค่ามาก ๆ แสดงว่าไม่ใช่เสียงพูด ถ้าจุดตัดศูนย์น้อยแสดงว่าอยู่ในช่วงเสียงพูด แต่อย่างไรก็ตาม ก็ยังคงไม่มีความแน่นอนมากนัก

จากรูป 2.7 แสดงฮิสโตแกรมของอัตราจุดตัดศูนย์เฉลี่ยของทั้งขณะที่มีแสดงและขณะที่ไม่มีเสียง ซึ่งแสดงให้เห็นว่าขณะไม่มีเสียงมีค่าอัตราจุดตัดศูนย์เฉลี่ยที่ประมาณ 49 ครั้ง ต่อ 10 วินาที และขณะมีเสียงพูดจะมีอัตราจุดตัดศูนย์เฉลี่ยประมาณ 14 ครั้ง ต่อ 10 วินาที และจะมีบางส่วนที่ซ้อนกัน ดังนั้นจึงเป็นไปไม่ได้ที่จะวิเคราะห์เสียงโดยใช้วิธีอัตราจุดตัดศูนย์เฉลี่ยวิธีนี้เพียงอย่างเดียว



รูปที่ 2.7 ค่าของ Average Zero Crossings ขณะมีสัญญาณเสียงและไม่มีสัญญาณเสียง

2.6.3 การวิเคราะห์โดยวิธี ชอร์ตไทม์สเปคตรัม (Short-time Spectrum analysis)

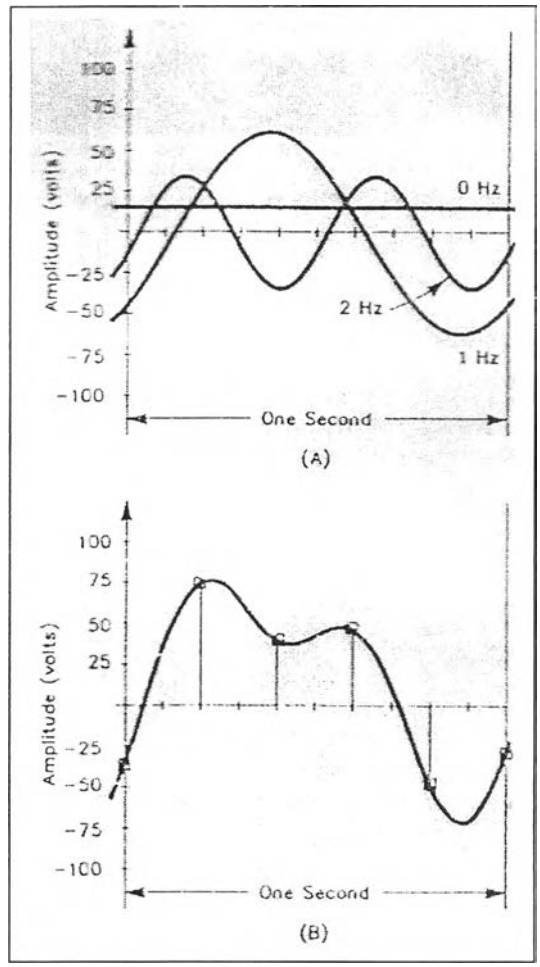
ในทางวิทยาศาสตร์และทางวิศวกรรม จะแทนสัญญาณต่างๆ ให้อยู่ในรูปของผลรวมของสัญญาณรูปไซน์ (Sinusoid) หรือสัญญาณเอกโปเนนเชียล (exponential) เพื่อในการแก้ปัญหาและเพื่อจะเข้าใจอย่างลึกซึ้งเกี่ยวกับลักษณะทางฟิสิกส์มากขึ้นกว่าเดิม เช่น การแทนด้วยฟูริเยอร์ ซึ่งใช้ในการประมวลสัญญาณ ซึ่งมีเหตุผล 2 ประการ

ประการที่ 1 ใช้กับระบบที่เป็นเชิงเส้น เพื่อสะดวกในการหาผลตอบสนองโดยใช้ทฤษฎีทับซ้อน (superposition) ของสัญญาณรูปไซน์หรือสัญญาณเอ็กซ์โปเนนเชียล

ประการที่ 2 การแทนด้วยฟูริเยอร์จะช่วยให้มองเห็นคุณสมบัติของสัญญาณได้ชัดเจนมากกว่าสัญญาณเดิม

การวิจัยด้านการสื่อสารทางเสียง จะใช้หลักการของฟูริเยอร์ในการแก้ไข เพราะฟูริเยอร์จะช่วยในการสร้างรูปแบบสำหรับสัญญาณเสียงของระบบเชิงเส้น ที่เป็นคาบเวลาหรือการสุ่มของสัญญาณที่แปรตามเวลา โดยทั่วไป สเปกตรัม (Spectrum) ของสัญญาณที่ออกมาจะอยู่ในรูปของผลตอบสนองทางด้านความถี่ ดังนี้มันจึงสามารถคาดเดาได้ว่าสเปคตรัมของเอาท์พุทจะสะท้อนให้เห็นคุณสมบัติของความถี่ของเสียง แต่อย่างไรก็ตาม รูปแบบของเสียงนี้จะยุ่งยากมากกว่าในเรื่องสระของเสียงและการออกเสียง ดังนั้น การแทนด้วยฟูริเยอร์จะเหมาะสมสำหรับสัญญาณคาบ, ทรานเซียนต์ (transient) หรือสัญญาณสุ่มที่ไม่ใช้สำหรับเป็นเสียงพูดที่ต่อเนื่อง ซึ่งคุณสมบัติเปลี่ยนแปลงเป็นฟังก์ชันของเวลา อย่างไรก็ตาม เราจะสามารถเห็นคุณสมบัติของเสียงได้มากกว่าการวิเคราะห์โดยวิธีชอร์ตไทม์ ยกตัวอย่างเช่น คุณสมบัติของพลังงาน จุดตัดศูนย์ ซึ่งสามารถประมาณเวลาประมาณ 10 ถึง 30 มิลลิวินาที

ในการศึกษาคุณสมบัติของสัญญาณเสียง เราสามารถศึกษาเกี่ยวกับรูปแบบ หลักการของฟูริเยอร์ของสัญญาณที่เปลี่ยนตามเวลา เราจะกำหนดการแปลงฟูริเยอร์ (Fourier Transform) และการกระทำของการวิเคราะห์ของการแปลงฟูริเยอร์ เราสามารถใช้เทคนิคการคำนวณพื้นฐานโดยใช้อัลกอริทึมที่เร็วกว่าดิสครีทฟูริเยอร์ ทรานสฟอร์ม (Discrete Fourier Transform) การใช้งานและวิเคราะห์เสียงพูดการแสดงสเปคตรัม



รูปที่ 2.8 แสดงองค์ประกอบของสัญญาณ

2.6.3.1 ข้อกำหนดและคุณสมบัติ

การแปลงฟูริเยอร์ ที่แปรผันตามเวลา เนื่องมาจากมีความต้องการที่จะสะท้อนถึงคุณสมบัติแปรผันตามเวลาของรูปคลื่นเสียง การกำหนดรูปแบบการแปลงฟูริเยอร์ของสัญญาณเสียง ซึ่งฟูริเยอร์ที่แปรผันตามเวลา คือ

$$Xn\left(e^{jw}\right) = \sum_{m=-\alpha}^{\alpha} w\left(n-m\right)x\left(m\right)e^{-jwm} \qquad (2.7)$$

ในสมการ W(n-m) จะเป็นค่าจริงของลำดับของสัญญาณซึ่งใช้กำหนดกลุ่มของสัญญาณอินพุทที่รับเข้ามาที่เวลาใดๆ n ฟูริเยอร์ทรานสฟอร์มที่ขึ้นอยู่กับเวลาจะเป็นฟังก์ชันของสองตัวแปร คือ ตัวแปร เวลาใดๆ (n) และตัวแปรทางความถี่ (W) ซึ่งจะต่อเนื่องกันไป การเปลี่ยนรูปแบบของสมการ โดยเปลี่ยนดัชนีการรวมตัวกัน (summation) ดังสมการ

$$Xn\left(e^{jw}\right) = \sum_{m=-\alpha}^{\alpha} w(n-m)x(m)e^{-jw(n-m)}$$

$$= e^{jwn} \sum_{m=-\alpha}^{\alpha} x(n-m)w(m)e^{-jwm} \qquad (2.8)$$

ถ้าเรากำหนดให้

$$Xn\left(e^{jw}\right) = \sum_{m=-\alpha}^{\alpha} w(n-m)x(m)e^{-jwm}$$

จะได้ว่า

$$Xn\left(e^{jw}\right) = e^{-jwn} X(n)e^{-jw} \qquad (2.9)$$

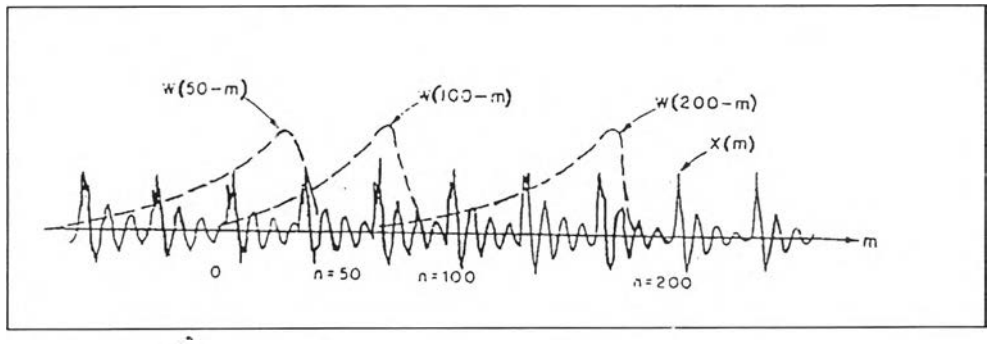
จากสมการนี้ สามารถแปลไปได้ 2 ทาง คือ

ทางที่ 1 ถ้า n คงที่ เรากำหนดให้เป็นการแปลงฟูริเยอร์ของ W(n-x)X(m) ดังนั้น การกำหนดให้ n,Xn (e^{jw}) คงที่จะทำให้มีคุณสมบัติเหมือนการแปลงฟูริเยอร์ธรรมดา

ทางที่ 2 ถ้าพิจารณา Xn(e^{jw}) เป็นฟังก์ชันของเวลาใดๆ (n) และ W คงที่ ในกรณีนี้เราจะลดรูปของสมการให้ อยู่ในรูปของการคอนโวลูชัน

2.6.3.2 การแปลงฟูริเยอร์

พิจารณา Xn(e^{jw}) ซึ่งเป็นการแปลงฟูริเยอร์ของ W (n-m) X(m) เมื่อ -∞<m<∞ การแปลงฟูริเยอร์จะเป็นฟังก์ชันของเวลา ณ เวลา n ซึ่งสามารถทำเป็นตัวเลขได้ตาม รูปที่ 2.9 ซึ่งแสดง X(m) และ W(n-m) ซึ่งเป็นฟังก์ชันของ m สำหรับ n ค่าต่างๆ



รูปที่ 2.9 แสดงค่าของ X(m) w(n-m) ที่ n ค่าต่างๆ

สภาพการมีอยู่ของการแทนการแปลงฟูริเยอร์ (Fourier transform repesentation) จะยังคงเป็นจริง ถ้าเราสามารถนำเอาสภาวะที่เพียงพอสำหรับการแปลงฟูริเยอร์ซึ่งมันก็คือ การบวกของค่าสัมบูรณ์ ในกรณีนี้ เราต้องการ X(m) W(n-m) ที่เป็นผลรวมทั้งหมดของทุกๆ ค่าของ n ซึ่ง W(n-m) จะอยู่ในช่วงจำกัด ซึ่งเป็นสภาพที่เหมาะสม

สมการของการแปลงฟูริเยอร์ ของ W(n-m) X(m) สามารถเขียนได้ดังนี้

$$w(n-m)x(m) = \frac{1}{2\pi}\int_{-\pi}^{\pi} x_n(e^{jw})e^{jwn}dw \qquad 2.10$$

และจากคุณสมบัติของวินโดว์แบบต่างๆ ซึ่งมีผลทำให้ความกว้างของโหลปหลักจะเป็นสัดส่วนกลับกับความกว้างของหน้าต่าง และระดับของโหลปข้างจะไม่ขึ้นอยู่กับความกว้างของวินโดว์

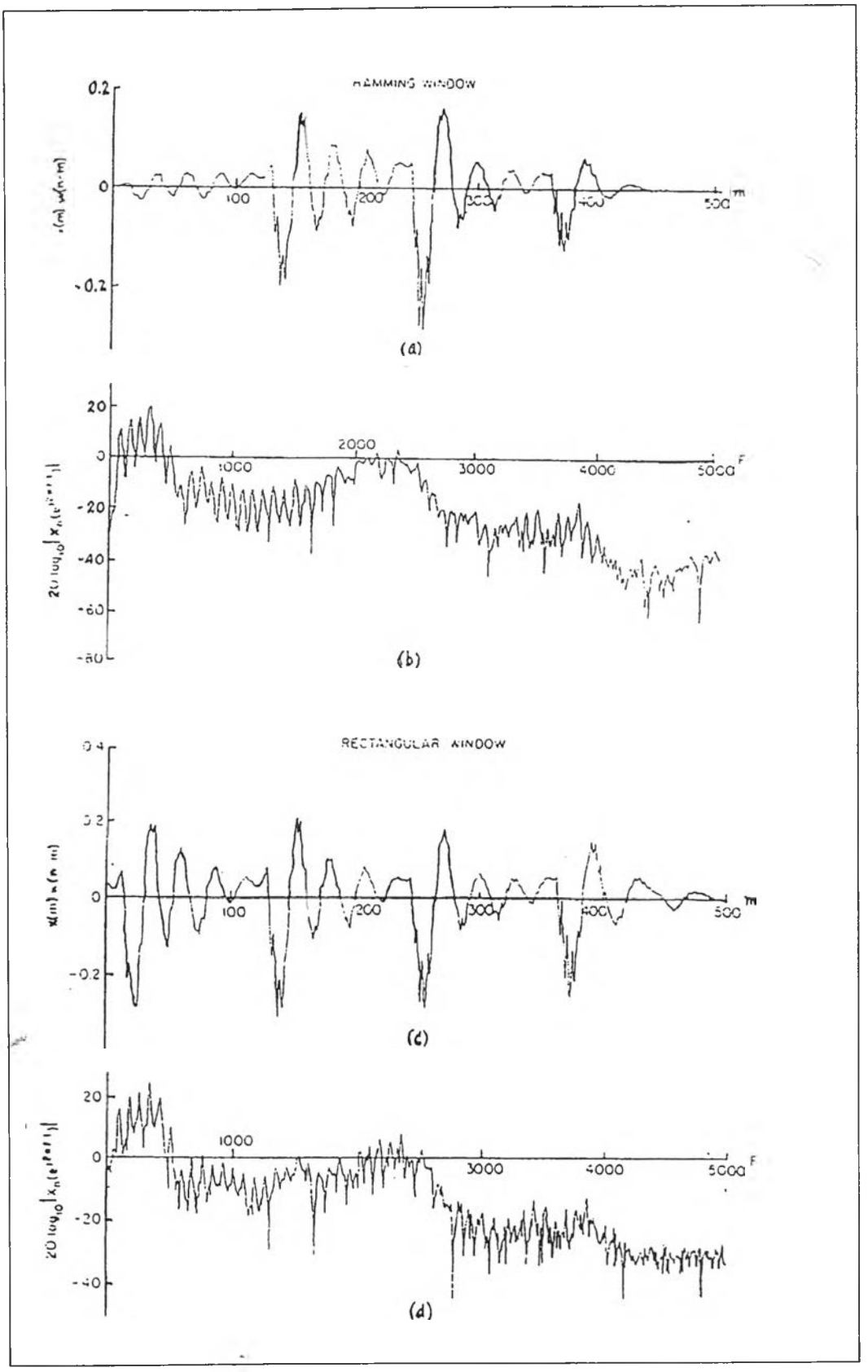
ผลการใช้วินโดว์ในการหาสเปคตรัม ดังแสดงในรูป 2.10 ถึง 2.11 รูป a ของแต่ละรูปจะแสดงวินโดว์ของสัญญาณ X(n) W(n-m) รูป b จะแสดง log magnitude ของ xn(e^{jw}) รูป c จะแสดงการใช้หน้าต่างแบบสี่เหลี่ยม และรูป d จะแสดงสเปคตรัมของ log magnitude Spectrum รูป 2.10 จะแสดงผลของวินโดว์ ที่ 500 แซมเปิล (50mS ที่ความถี่ การแซมปลิ้ง 10 kHz) ซึ่งสัญญาณรายคาบสามารถดูได้จาก รูป a รูป b จะเห็นความถี่พื้นฐานและความถี่ ฮาร์โมนิค ที่แสดงเป็นยอดแหลมเล็กๆ นอกจากนี้ยังเห็นยอดสูงสุดที่ 300-400 Hz และยอดเรียบที่ประมาณ 2200 Hz และที่ 3800 Hz สุดท้ายสเปคตรัมที่แสดงก็จะตกไปที่ความถี่สูงๆ

เปรียบเทียบ รูป 2.10 a และ 2.10 d ซึ่งเป็นหน้าต่าง แบบแฮมมิ่ง และแบบสี่เหลี่ยม ตามลำดับโดยพิจารณากลุ่มของฮาร์โมนิค โครงสร้างและความเด่นชัดของยอดแหลมซึ่งแตกต่างกัน ที่เห็นเด่นชัดคือ การเพิ่มของยอดแหลมของฮาร์โมนิค ในรูป 2.10 d ความแตกต่างอันอื่นก็คือ ความกว้างของโหลปข้าง และอีกสิ่งหนึ่งคือ วินโดว์แบบสี่เหลี่ยมจะมีปลายยอดแหลมมากกว่า หรือมีสเปคตรัมของสัญญาณรบกวนมากกว่านั่นเอง

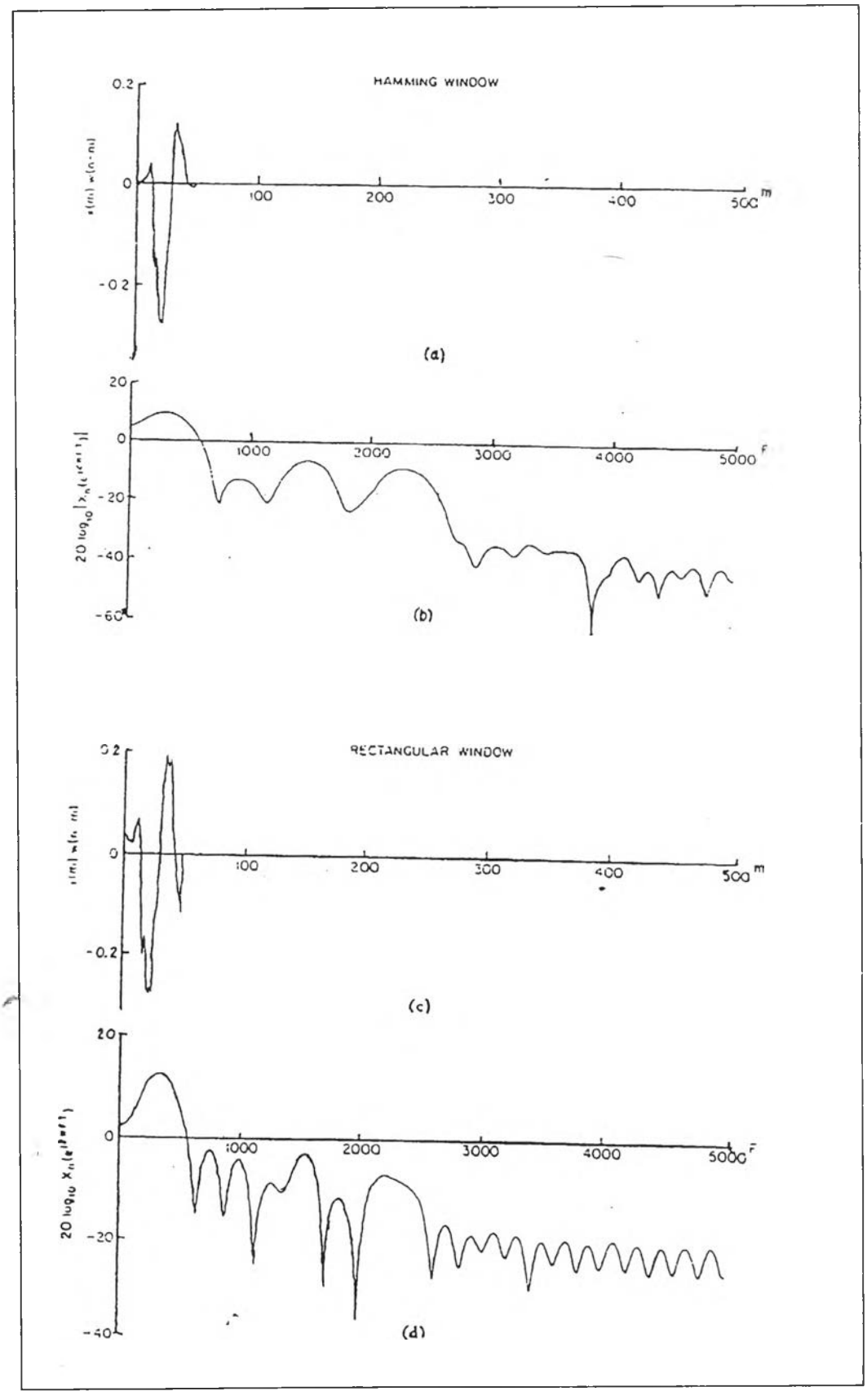
รูป 2.11 แสดงการเปรียบเทียบที่ 50 สัญญาณแซมปลิ้ง ของสัญญาณเสียงพิจารณาสเปคตรัม รูป 2.11 b และ 2.11 d เทียบกับ รูป 2.10 รูป 2.11 จะแสดงเพียงค่าสูงสุดกว้างๆ ที่ประมาณ 400 Hz, 1400 HZ และ 2200 Hz

รูป 2.12 และ 2.13 ก็แสดงผลของวินโดว์ขณะไม่มีเสียงพูดที่ 500 สัญญาณสุ่มและ 50 สัญญาณสุ่มตามลำดับ จากรูปจะเห็นสเปคตรัมแสดงการเปลี่ยนทิศทางของยอดแหลมอย่างช้าๆ ความกว้างของยอดแหลมและสเปคตรัมของทั้งคู่จะเป็นการสุ่มสัญญาณแบบธรรมชาติของเสียงรบกวน และการใช้วินโดว์แบบแฮมมิงจะดูเรียบกว่าแบบสี่เหลี่ยม

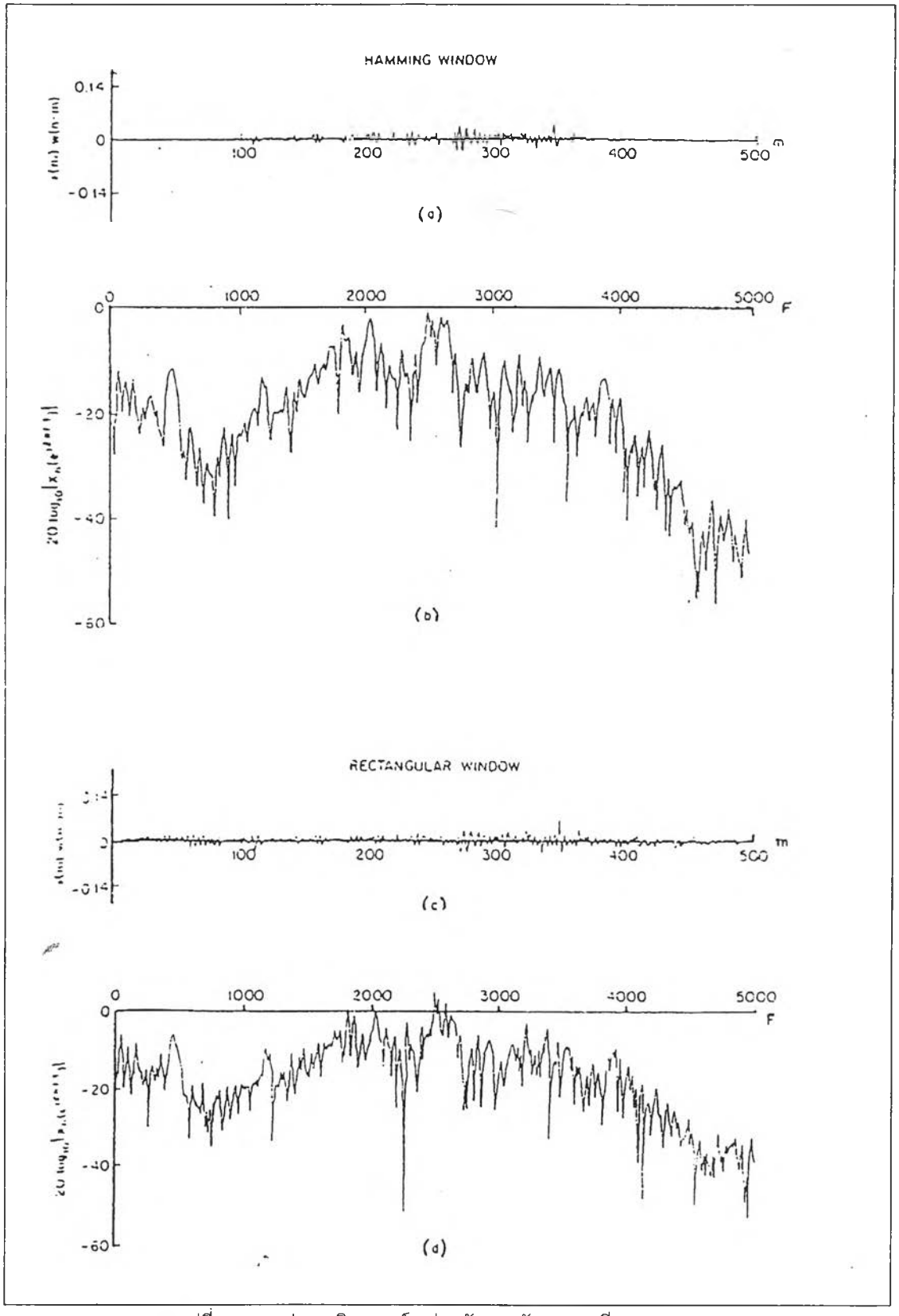
จากตัวอย่างรูป 2.10 ถึงรูป 2.13 แสดงความสัมพันธ์ระหว่างระยะเวลาของวินโดว์ 2 และคุณสมบัติของการแปลงฟูริเยอร์ นั่นคือ การกระจายความถี่จะแปรปรวนกลับกับความยาวของวินโดว์ และจุดประสงค์ของวินโดว์ เพื่อจำกัดเวลาระหว่างการวิเคราะห์เพื่อคุณสมบัติของรูปคลื่นไม่เปลี่ยนแปลงไป



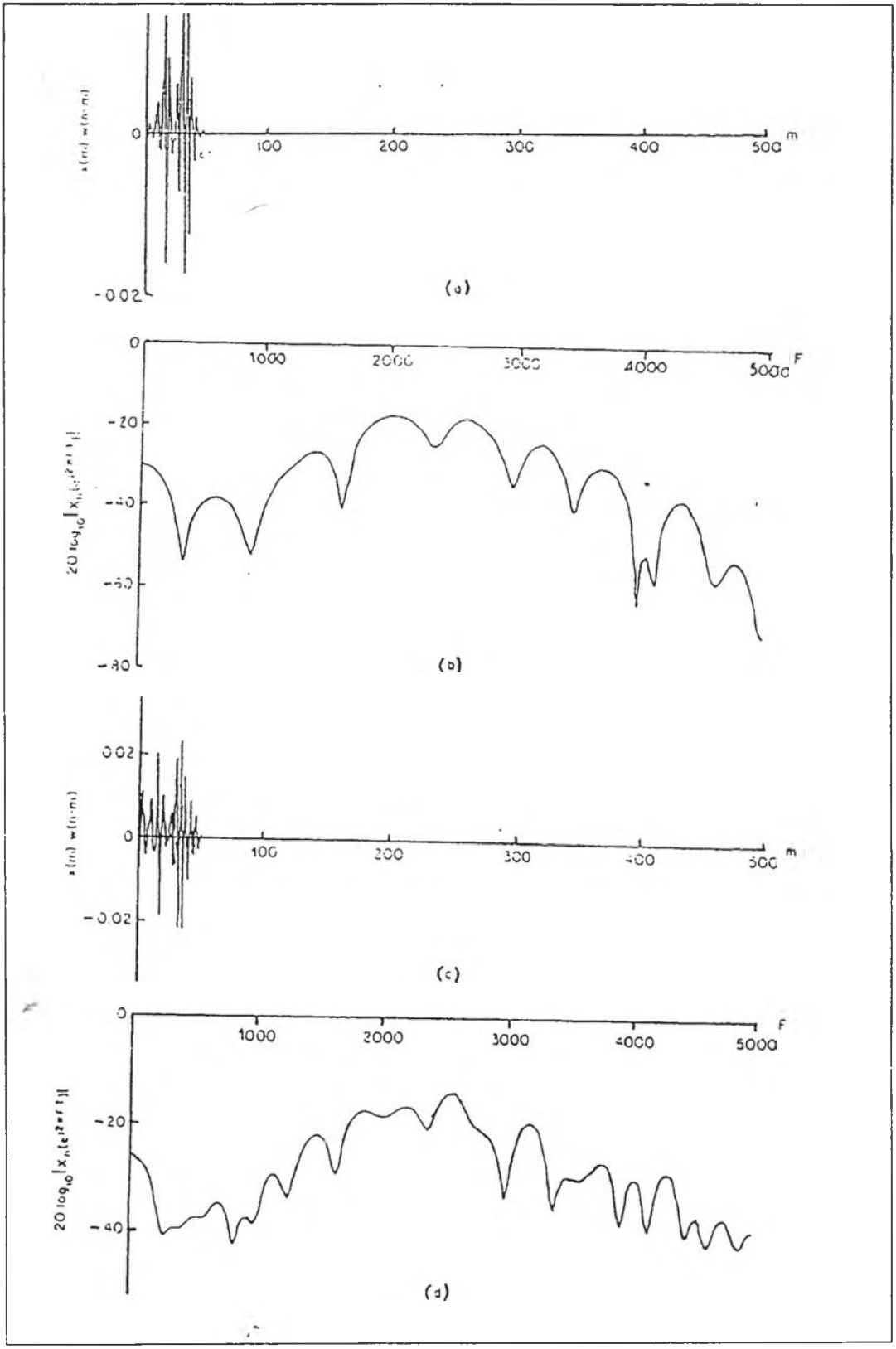
รูปที่ 5 แสดงค่าการวิเคราะห์สเปคตรัมของสัญญาณเสียง



รูปที่ 6 แสดงค่าการวิเคราะห์สเปคตรัมของสัญญาณเสียง



รูปที่ 7 แสดงค่าการวิเคราะห์สเปคตรัมของสัญญาณเสียง



รูปที่ 8 แสดงค่าการวิเคราะห์สเปคตรัมของสัญญาณเสียง

2.6.4 วิธีการวิเคราะห์แบบโฮโมมอร์ฟิค

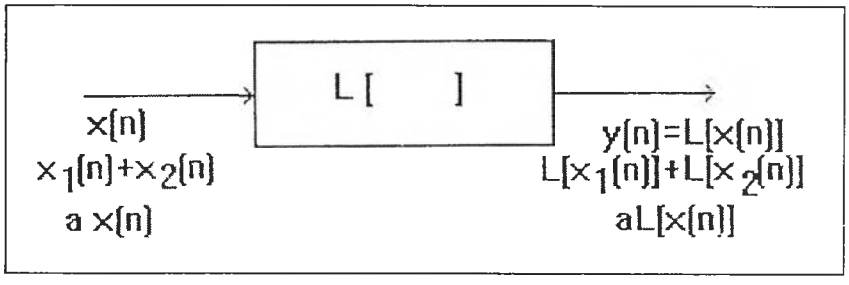
ระบบโฮโมมอร์ฟิคสำหรับการคอนโวลูชัน ในส่วนของระบบโฮโมมอร์ฟิคที่ใช้สำหรับการคอนโวลูชันจะใช้หลักการพื้นฐานของทฤษฎีทับซ้อนหลักการ ซึ่งทฤษฎีทับซ้อนนี้จะใช้ได้เฉพาะระบบที่เป็นเชิงเส้นดังสมการ

$$L[X(n)] = L[X_1(n) + X_2(n)]$$
$$= L[X_1(n)] + L[X_2(n)]$$
$$= Y_1(n) + Y_2(n) \qquad = Y(n) \qquad (2.11)$$

และ $$L[X(n)] = aL[X(n)] \qquad = aY(n) \qquad (2.12)$$

เมื่อ L แทนโอปะเรเตอร์ที่เป็นเชิงเส้น

หลักการของทฤษฎีซับซ้อน คือ ถ้าสัญญาณอินพุต ประกอบด้วยสัญญาณหลาย ๆ สัญญาณรวมกันแล้ว สัญญาณเอาต์พุตทั้งหมดจะรวมกันอย่างเป็นเชิงเส้นกลายเป็นผลตอบสนองเอาต์พุต ดังรูปที่ 2.14 เมื่อเครื่องหมาย "+" ที่อินพุตและเอาต์พุต จะทำการรวมสัญญาณที่อินพุต แล้วสร้างผลรวมของเอาต์พุต



รูปที่ 2.14 การแทนระบบด้วยหลักการของทฤษฎีทับซ้อน

จากสมการ การคอนโวลูชัน

$$Y(n) = \sum_{k=-\infty}^{\infty} h(n-k)x(k) = h(n) \times X(n) \qquad (2.13)$$

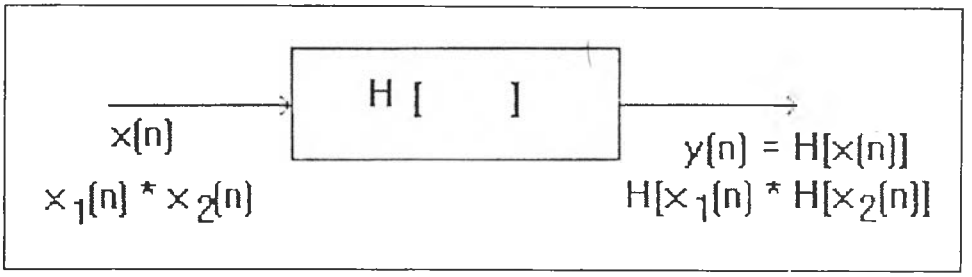
โดย * แทนสัญลักษณ์ของการคอนโวลูชัน

จากหลักการ ทฤษฎีทับซ้อน สำหรับระบบที่เป็นเชิงเส้น เราสามารถกำหนดระดับชั้นของระบบ ซึ่งปฏิบัติตามหลักการของ ทฤษฎีทับซ้อน โดยการเปลี่ยนการคอนโวลูชันดังนี้

$$H[X(n)] = H[X_1(n) + X_2(n)]$$
$$= H[X_1(n)] + H[X_2(n)]$$
$$= Y_1(n) + Y_2(n) \qquad = Y(n) \qquad (2.14)$$

สมการนี้จะเหมือนกับสมการ 2.12 ซึ่งแสดงการคูณค่าคงที่ลงไป แต่อย่างไรก็ตาม การคูณค่าคงที่ไม่เป็นที่ต้องการสำหรับใช้งานที่เราจะพิจารณา ระบบที่มีคุณสมบัติตามสมการ 2.14 จะเป็นเทอมของ ระบบ

โฮโมมอร์ฟิค สำหรับการคอนโวลูชันในระบบนี้มาจากความจริงที่ว่าวิธีการแปลงต่าง ๆ สามารถแสดงเป็นการแปลงแบบ โฮโมมอร์ฟิค ของเวกเตอร์เชิงเส้นดังระบบรูป 2.15 ซึ่งการกระทำของการคอนโวลูชันจะอธิบายที่อินพุตและเอาต์พุตของระบบตัวกรองโฮโมมอร์ฟิคจะเป็นตัวอย่างของระบบโฮโมมอร์ฟิคที่มี 1 อุปกรณ์ตลอดจนระบบพิเศษที่ไม่เปลี่ยนแปลง ขณะที่อุปกรณ์ที่ไม่ต้องการถูกเอาออกไป



รูปที่ 2.15 การแทนระบบโฮโมมอร์ฟิคสำหรับการคอนโวลูชัน

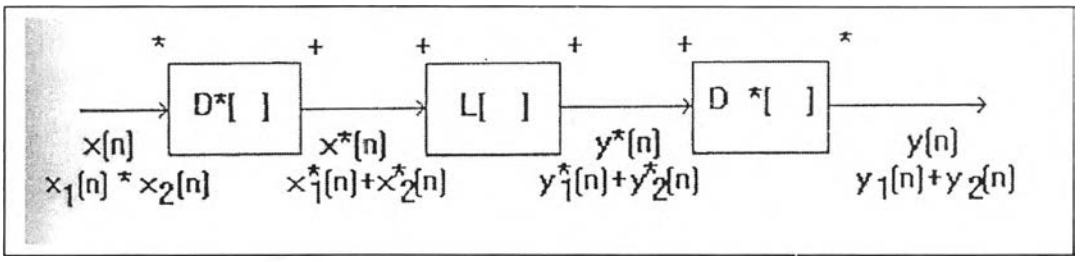
หลักการสำคัญของระบบโฮโมมอร์ฟิค คือ ระบบโฮโมมอร์ฟิคทุกระบบ สามารถแทนด้วยระบบโฮโมมอร์ฟิค 3 ตัว ต่อแบบคาสเคดกัน ดังรูป 2.16

ระบบที่ 1 จะนำอินพุตเข้ามาและรวมโดยคอนโวลูชันและการแปลงอินพุตไปที่เอาท์พุต

ระบบที่ 2 จะเป็นระบบที่เป็นเชิงเส้นโดยอาศัยหลักการทฤษฎีทับซ้อน ดังสมการ (2.16)

ระบบที่ 3 จะเป็นการอินเวอร์สกับระบบแรก คือมันจะแปลงกลับสัญญาณรวม โดยบวกสัญญาณกลับโดยวิธีการคอนโวลูชันส่วนสำคัญของการมีอยู่ของรูปแบบโฮโมมอร์ฟิค ในความเป็นจริง คือ การออกแบบให้ลดปัญหาในการออกแบบระบบเชิงเส้นดังรูป 2.16 คุณสมบัติของการแปลงกลับคอนโวลูชัน (Deconvolution) นี้จะใช้หลักการทฤษฎีทับซ้อน ซึ่งเมื่ออินพุตเป็นการทำคอนโวลูชันและเอาต์พุตเป็นของเดิม ให้ D*[ ] เป็นคุณสมบัติของการแปลงกลับคอนโวลูชันของระบบโฮโมมอร์ฟิคคุณสมบัติของระบบ คือ

$$\begin{aligned} D^*[X(n)] &= D^*[X_1(n) + X_2(n)] \\ &= D^*[X_1(n)] + D^*[X_2(n)] \\ &= \overline{X}_1(n) - \overline{X}_2(n) \qquad = \overline{X}(n) \end{aligned} \tag{2.15}$$



รูปที่ 2.16 รูปแบบของระบบสำหรับการการแปลงกลับคอนโวลูชันของระบบโฮโมมอร์ฟิค

ลักษณะเดียวกันคุณสมบัติการแปรกลับ กำหนดโดย

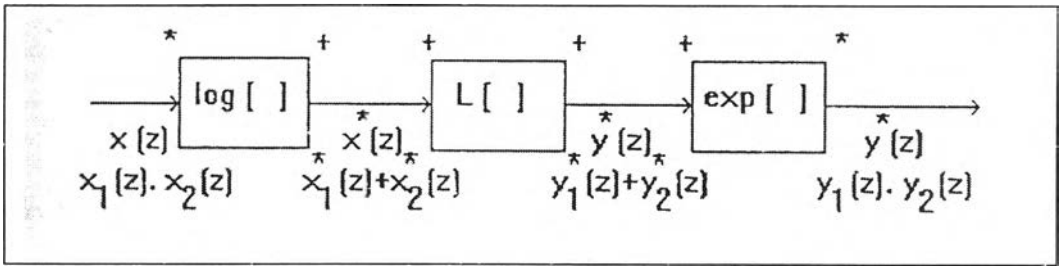
$$D^{-1} * [X(n)] = D^{-1} * [Y_1(n) + Y_2(n)]$$
$$= D^{-1} * [Y_1(n)] + D^{-1} * [Y_2(n)]$$
$$= Y_1(n) + Y_2(n) \qquad = Y(n) \qquad (2.16)$$

การแทนด้วยคณิตศาสตร์ของระบบ จะขึ้นอยู่กับว่าถ้าอินพุตเป็นการคอนโวลูชัน

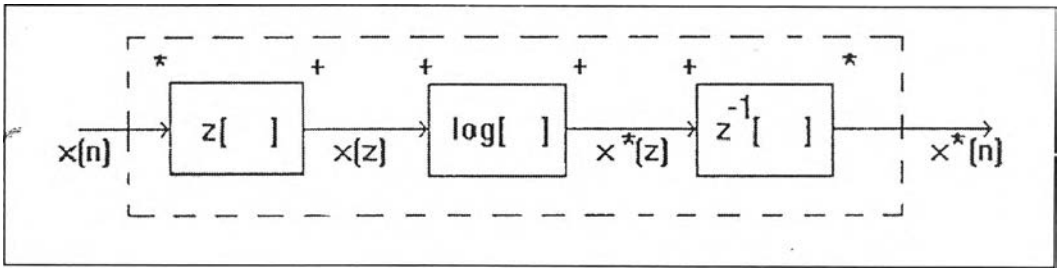
$$X(n) = X_1(n) * X_2(n) \qquad (2.17)$$

แล้วการแปลง Z ของอินพุตจะเป็นผลของการแปลง Z จุดนั้น ๆ

$$X(z) = X_1(z) * X_2(z) \qquad (2.18)$$



รูปที่ 2.17 การแทนในโดเมนความถี่โดยการคอนโวลูชันของระบบโฮโมมอร์ฟิค

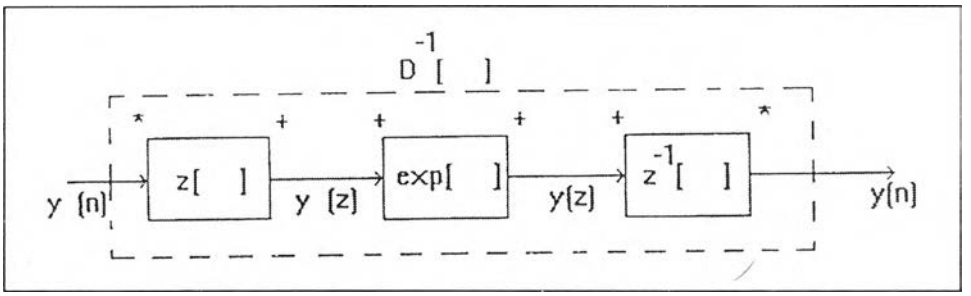


รูปที่ 2.18 การแทนคุณลักษณะการแปลงกลับคอนโวลูชันของระบบโฮโมมอร์ฟิค

สมการ 2.15 จะหมดไป เมื่อการแปลง Z ของเอาต์พุตของระบบก็จะต้องนำมารวมกับการแปลง Z ตอนแรก ดังนั้น โดเมนทางความถี่จะปฏิบัติตัวตามลักษณะของระบบ สำหรับการคอนโวลูชันจะต้องมีคุณสมบัติ ซึ่งถ้าสัญญาณถูกแทนด้วยผลของการแปลง Z ที่อินพุตแล้วเอาต์พุต จะต้องเป็นผลรวมของผลตอบสนองของการแปลง Z ทาง เอาต์พุตดังรูป 2.17

$$X(z) = \log [X(z)]$$
$$= \log [X_1(n) * X_2(n)]$$
$$= \log [X_1(z) + \log [X_2(z)] \quad (2.19)$$

ถ้าเราแทนสัญญาณด้วยลำดับสัญญาณระบบสามารถแทนได้ดังรูป 2.18 และการแปลงกลับของระบบสามารถแทนได้โดย รูป 2.18



รูปที่ 2.19 การแทนกลับของระบบด้วยการแปลงกลับคอนโวลูชันของระบบโฮโมมอร์ฟิค

การแทนด้วยระบบและการแปลงกลับดังรูปที่ 2.18 และรูปที่ 2.19 นั้นจะขึ้นอยู่กับสมการ 2.19 นั่นคือ ลอการิทึมของผลลัพธ์ จะเท่ากับผลบวกของลอการิทึมซึ่งจะเป็นจริงเมื่อเป็นค่าจำนวนเต็มบวก อย่างไรก็ตาม การแปลง Z จะเป็นจำนวนเชิงซ้อนและมีความสำคัญในการพิจารณาลักษณะเด่น เมื่อทำกับลอการิทึมของจำนวนเชิงซ้อน ในการคำนวณเราจะสนใจเกี่ยวกับความแน่นอนของสมการ (2.19) ซึ่งจะถูกต้องเมื่อค่าอยู่ในวงกลม 1 หน่วย จากคอมเพล็กซ์ลอการิทึม

$$X(e^{jw}) = \log X(e^{jw}) + \arg [X(e^{jw})] \quad (2.20)$$

จากสมการส่วนของจำนวนจริงจะไม่มีปัญหา ปัญหาจะอยู่ที่ส่วนจินตภาพ ซึ่งค่าของมุมเฟสของการแปลง Z ต้องอยู่บนวงกลม 1 หน่วย

เพื่อให้สามารถคำนวณค่าคอมเพล็กซ์ ลอการิทึมของสมการ 2.19 ให้เป็นที่น่าพอใจการแปลงกลับของ คอมเพล็กซ์ ลอการิทึม โดยการแปลงฟูริเยอร์ของอินพุตจะเป็นเอาต์พุตของระบบที่คอนโวลูชันกัน

$$X(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \bar{x}(e^{jw}) e^{jwn} dw \quad (2.21)$$

เอาต์พุตของระบบ (X(n)) จะถูกเรียกว่า คอมเพล็กซ์ เซ็พสตรัม (Complex Cepstrum) เราจะใช้เทอมเซ็พสตรัมสำหรับจำนวน

$$c(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log|x(e^{jw})| e^{jwn} dw \quad (2.22)$$

จากการอธิบายเราสามารถจะกำหนดระบบที่มีคุณสมบัติสำหรับโฮโมมอร์ฟิค คอนโวลูชันและกำหนดรูปแบบสำหรับส่วนที่เป็นเชิงเส้นของระบบตัวเลือกของระบบเชิงเส้นจำเป็นในการออกแบบระบบ

การพิจารณาการคำนวณ การแทนด้วยคณิตศาสตร์ของระบบและการแปลงกลับระบบในรูป 2.17 และ 2.18 ตามลำดับนั้น มีคำแนะนำสำหรับระบบโฮโมมอร์ฟิคที่คอนโวลูชัน คือถ้าเราจำกัดจำนวนลำดับอินพุตที่จะทำการรวมกันแล้วจะทำให้การแปลง Z ของสัญญาณอินพุตจะอยู่ในย่านวงกลม 1 หน่วยนั่น คือ ลำดับสัญญาณจากการ แปลงฟูริเยอร์ในกรณีเดียวกันมันจะแทนโอปะเรเตอร์การแปลง Z ในรูป 2.17 และ 2.18 ด้วยสำหรับกรณีพิเศษของลำดับอินพุตที่มีความยาวจำกัด การแทนด้วยคณิตศาสตร์ของระบบโดยการคอนโวลูชันจะเป็น

$$X\left(e^{jw}\right) = \sum_{n=0}^{N-11} x(n)e^{-jwn} \tag{2.23}$$

$$\begin{aligned} X\left(e^{jw}\right) &= \log\left[x\left(e^{jw}\right)\right] \\ &= \log X\left(e^{jw}\right) + j\arg\left[X\left(e^{jw}\right)\right] \end{aligned} \tag{2.24}$$

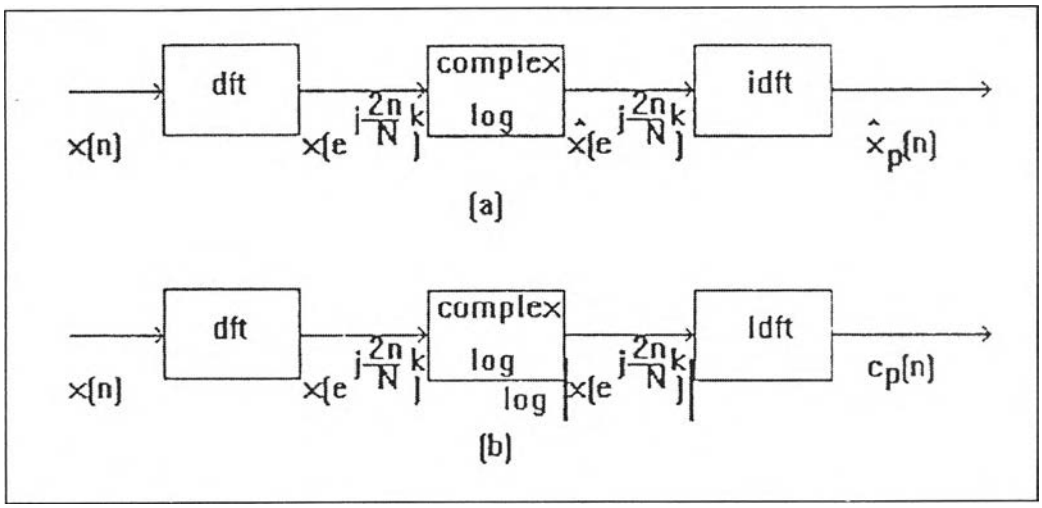
$$X(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \bar{x}\left(e^{jw}\right)e^{jwn}\,dw \tag{2.25}$$

สมการ 2.23 จะเป็นการแปลงฟูริเยอร์ของลำดับอินพุตสมการ 2.24 จะเป็นคอมเพล็กซ์ลอการิทึมของการแปลงฟูริเยอร์ของอินพุต และสมการ 2.25 จะเป็นการแปลงกลับฟูริเยอร์ของคอมเพล็กซ์ ลอการิทึม เราสังเกตว่าในการหาคอมเพล็กซ์ ลอการิทึมจากสมการ 2.23 เราจะต้องกำหนดคอมเพล็กซ์ ลอการิทึมของการแปลงฟูริเยอร์ ซึ่งจะช่วยกำหนดข้อบังคับของ คอมเพล็กซ์ เซ็พสตรัม ของลำดับอินพุตที่เป็นแต่จำนวนจริง การแปลงฟูริเยอร์ของลำดับจำนวนจริงจะเป็นฟังก์ชันคู่ ส่วนของจำนวนจินตภาพจะเป็นฟังก์ชันคี่ ดังนั้น ถ้าคอม เพล็กซ์ เซ็พสตรัม เป็นลำดับของจำนวนจริง เราจะต้องกำหนดให้ฟังก์ชันของลอก แมกนีจูด เป็นฟังก์ชันคู่ของ W และเฟสจะต้องกำหนดเป็นฟังก์ชันคี่ของ W ซึ่งสามารถแสดงให้เห็นสภาพของคอมเพล็กซ์ ลอการิทึม ซึ่งเฟสจะเป็นฟังก์ชันคาบต่อเนื่องของ W กับคาบของ 2π สมการ 2.23 ยังไม่สมการที่ใช้คำนวณ เราสามารถประมาณสมการ 2.23 โดยใช้ การแปลงดิสครีทฟูริเยอร์ (DFT) ที่ขณะจำกัดเพื่อให้เหมือนกับการสุ่มสัญญาณของ การแปลงฟูริเยอร์ที่ลำดับเดียวกัน เนื่องจาก DFT สามารถคำนวณโดย FFT ดังนี้ การทำให้สำเร็จก็โดยการแทนตัวกระทำของการแปลงฟูริเยอร์ โดยจะตรงกับตัวกระทำของ DFT ซึ่งผลของสมการจะเป็น

$$Xp(k) = \sum_{n=0}^{N-1} x(n)\,e^{-\frac{j2\pi kn}{N}} \tag{2.26}$$

$$Xp(k) = \log[Xp(k)] \qquad ;N < k < k-1 \tag{2.27}$$

$$Xp(k) = \sum_{k=0}^{N-1} \bar{x}_p(k)\,e^{-\frac{j2\pi kn}{N}} \tag{2.28}$$



รูปที่ 2.20 (a) คอมเพล็กซ์ เซ็พสตรัม (b) เซ็พสตรัม

สมการ 2.28 จะแทนการแปลงกลับของดิสครีทฟูริเยอร์ (IDFT) ของคอมเพล็กซ์ลอการิทึมของดิสครีทฟูริเยอร์ ที่จำกัดความยาวของลำดับอินพุต ตัวห้อย p เป็นผลของลำดับ ซึ่งไม่แน่นอนเท่ากับคอมเพล็กซ์เซ็พ สตรัมจากสมการ 2.23 นี้ จะทำให้เกิดความจริง ซึ่งคอมเพล็กซ์ลอการิทึมใช้ใน DFT ซึ่งเป็นเวอร์ชันที่ใช้การสุ่ม ของ $x(e^{jw})$ และผลของแปลงกลับ จะเป็นอไลส์เวอร์ชันของค่าจริงของคอมเพล็กซ์ เซ็พสตรัม นั่นคือ คอมเพล็กซ์ เซ็พสตรัม คำนวณโดยสมการ 2.26 จะอ้างอิงไปยังคอมเพล็กซ์ เซ็พสตรัมโดย เราสามารถสังเกตว่าคอมเพล็กซ์ เซ็พสตรัม ประกอบด้วยการใช้คอมเพล็กซ์ ลอการิทึม และเซ็พสตรัม โอปะเรเตอร์การคำนวณสำหรับระบบการคอนโวลูชัน ดังแสดงในรูป 2.20

$$Xp(k) = \sum_{r=-\infty}^{\infty} \overline{x}_p(n + rN) \qquad (2.29)$$

2.6.5 การวิเคราะห์แบบการคาดเดาเชิงเส้น

การวิเคราะห์แบบการคาดเดาเชิงเส้น (Linear Predictive Coding) หรือ LPC นี้เป็นเทคนิคอย่างหนึ่งในการวิเคราะห์เกี่ยวกับเสียง วิธีนี้เป็นเทคนิคที่ดีกว่าในการตัดพารามิเตอร์พื้นฐานออกไป เช่น การออกเสียง ระดับความดังของเสียง ส่วนสำคัญของวิธีนี้ขึ้นอยู่กับความสามารถที่ให้ความแม่นยำในการตัดพารามิเตอร์ต่าง ๆ และขึ้นอยู่กับความสัมพันธ์ของความเร็วในการคำนวณ

ความคิดพื้นฐานของการวิเคราะห์แบบ LPC คือ การสุ่มสัญญาณซึ่งสามารถประมาณว่าเป็นการรวมเชิงเส้นของสัญญาณสุ่มเดิม โดยลดผลบวกของความแตกต่างระหว่างสัญญาณสุ่มจริงกับสัญญาณที่คาดว่าเป็นเชิงเส้น

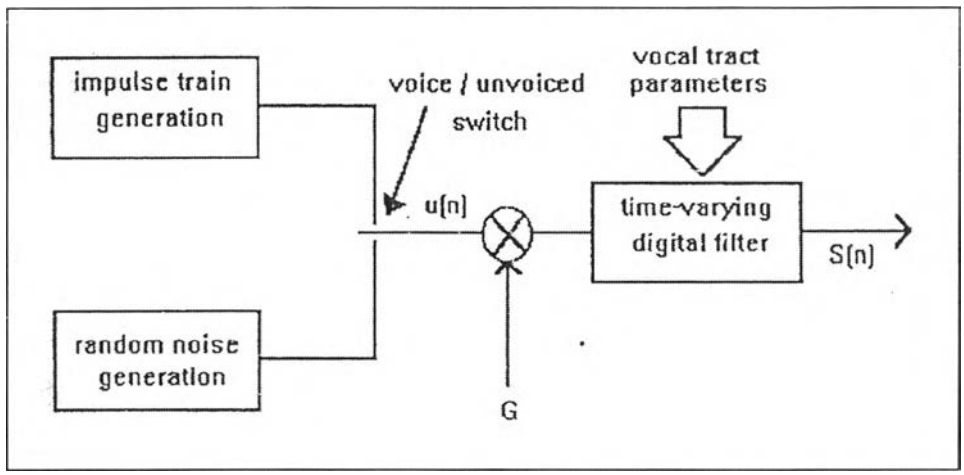
ปรัชญา ของการคาดเดาแบบเชิงเส้น จะเกี่ยวข้องกับรูปแบบในการสังเคราะห์เสียงพื้นฐาน เช่น ความเป็นเชิงเส้น ระบบที่แปรผันตามเวลาทั้งขณะมีเสียง และขณะไม่มีเสียง วิธี LPC นี้จะให้ความแม่นยำ ความเชื่อมมั่นในการตัดพารามิเตอร์

หลักการของ LPC พิจารณาการวิเคราะห์การคาดเดาเชิงเส้น ดังแสดงในรูป 2.21

จากรูป ผลของสเปกตรัมของการแพร่กระจาย (radiation) กลุ่มการออกเสียงจะถูกแทนด้วยตัวกรองที่แปรผันตามเวลา ซึ่งฟังก์ชันของระบบที่ภาวะคงที่จะอยู่ในรูปที่ 2.21

$$H(Z) = \frac{S(z)}{U(z)} = \frac{G}{1 - \sum_{k=1}^{p} akz^{-k}} \qquad (2.30)$$

ระบบนี้จะถูกกระตุ้นโดยขบวน อิมพัลส์ ของเสียงพูดหรือสัญญาณรบกวน ในกรณีที่ไม่มีเสียงพูด ดังนั้น ค่าพารามิเตอร์ของรูปแบบนี้คือ การแยกระหว่าง มีเสียงกับไม่มีเสียง คาบเวลาของระดับเสียง อัตราขยายเสียง (G) และค่าสัมประสิทธิ์ (ak) ของดิจิตอลฟิลเตอร์ ค่าพารามิเตอร์จะเปลี่ยนแปลงช้า ๆ ตามกาลเวลา



รูปที่ 2.21 บล็อกไดอะแกรมของการสุ่มสัญญาณเสียง

จากรูปที่ 2.21 สัญญาณสุ่มของเสียงพูด S(n) จะสัมพันธ์กับ U(n) โดยสมการดังนี้

$$S(n) = \sum_{k=1}^{p} a_k\, s(n-k) + Gu(n) \qquad (2.31)$$

สัมประสิทธิ์การคาดเดา จะกำหนดเมื่อเอาต์พุตเป็น

$$S(n) = \sum_{k=1}^{p} \alpha_k\, s(n-k) \qquad (2.32)$$

เพื่อที่จะลดการเปลี่ยนของสัญญาณของระบบที่อันดับ P ซึ่งเป็นโพลิโนเมียล

$$\overline{S}(n) = \sum_{k=1}^{p} \alpha_k\, z^{-k} \qquad (2.33)$$

ค่าความผิดพลาดจากการคาดเดา e(n) หาได้จาก

$$e(n) = S(n) - S(n) = S(n) - \sum_{k=11}^{p} \alpha_k \, s(n-k) \qquad (2.34)$$

$$A(z) = 1 - \sum_{k=1}^{p} \alpha k z^{-k} \qquad (2.35)$$

จากสมการ 3.36 สามารถเห็นได้ว่า ลำดับการผิดพลาดของการทำนายจะเป็นเอาต์พุตของระบบ ซึ่งทรานสเฟอร์ฟังก์ชัน คือ

เปรียบเทียบสมการ (2.31) และ (2.34) ถ้าสัญญาณเสียงเปลี่ยนแปลงตามสมการ (2.31) และ ถ้า αk = ak แล้ว e(n) = GU(n) ดังนั้น ตัวกรองความผิดพลาดการคาดเดา (Predicpion error filter) และ A(z) จะเป็นการกรองกลับ (inverse filter) ของระบบ H(z) นั่นคือ

$$H(z) = \frac{G}{A(Z)} \qquad (2.35)$$

ปัญหาพื้นฐานพื้นฐานของการวิเคราะห์แบบนี้ คือ ในการกำหนดชุดของสัมประสิทธิ์การคาดเดา (αk) โดยตรงจากสัญญาณเสียง โดยการประมาณที่ดีของคุณสมบัติของสัญญาณเสียงตามสมการ 2.36 เพราะธรรมชาติของการแปรผันตามเวลาของสัมประสิทธิ์การคาดเดา จะต้องประมาณค่าจากส่วนเล็ก ๆ ของเสียง หลักพื้นฐานก็ คือการหาชุดสัมประสิทธิ์การคาดเดาจะลดการผิดพลาด การคาดเดากำลังสอง (mean-squared prediction error) ในส่วนสั้น ๆ ของเสียง ผลของพารามิเตอร์จะถือเป็นพารามิเตอร์ของฟังก์ชันระบบ H(z) ในโมดูล ของการคาดเดาเสียง

ถ้า αk = ak แล้ว e(n) = GU(n) หมายความว่า e(n) จะประกอบด้วยขบวนของอิมพัลซ์ คือ e(n) จะมีเวลาสั้น ๆ ดังนั้น การหา αk จะลดความผิดพลาดการคาดเดา

กระตุ้นตามความจริง คือถ้าสัญญาณถูกสร้างมาจากสมการที่ 2.31 ด้วยสัมประสิทธิ์ ไม่แปรตามเวลา และกระตุ้นทั้งสัญญาณอิมพัลซ์ และสัญญาณรบกวนแล้ว มันจะแสดงให้เห็นว่า สัมประสิทธิ์การคาดเดาเป็นผลมาจากการลดการผิดพลาดการคาดเดากำลังสองทุก ๆ เวลา

เหตุผลที่ยุ่งยากสำหรับการใช้การผิดพลาดการคาดเดากำลังสองเป็นเกณฑ์สำหรับประมาณพารามิเตอร์ ซึ่งจะเซ็ทของสมการเชิงเส้นให้สามารถวิเคราะห์ค่าพารามิเตอร์

การผิดพลาดการคาดเดากำลังสองจะหาค่าได้จาก

$$S(n) = \sum_{m} e_n^2(m) \qquad (2.37)$$

$$= \sum_{m} \left(s_n(m) - \bar{s}_n(m) \right) \qquad (2.38)$$

$$= \sum_{m} \left[s_n(m) - \sum_{k=1}^{p} \alpha k s_n(m-k) \right]^2 \qquad (2.39)$$

เมื่อ $S_n(m)$ เป็นส่วนของเสียงที่ถูกเลือกจากความใกล้เคียงของการแซมปลิ้ง n เช่น

$$S_n(m) = S(M + n) \qquad (2.40)$$

ในการรวม ในสมการ 2.38 ถึงสมการ 2.40 ไม่เจาะจงว่าจะต้องเป็นด้านซ้ายเสมอไป แต่เนื่องจากเราใช้วิธีวิเคราะห์ร่วมกับแบบซอร์ตไทม์ ซึ่งอาจทำให้ผลบวกเกินจุดที่เรากำหนดไว้ อย่างไรก็ตามค่านี้ไม่ได้อยู่ในสมการเชิงเส้น เราสามารถที่จะละทิ้งไปได้ จากนั้นเราก็จะสามารถหาค่าของสัมประสิทธิ์การคาดเดา αk โดยการลดค่า E_n ในสมการที่ 2.40 โดยให้ $dE_r/d\alpha l = 0$ โดย $l = 1, 2, 3, \ldots p$ ตามสมการ

$$\sum_{k=1}^{p} s_n(m-i)\, s_n(m) = \sum_{k=1}^{p} \overline{\alpha k} \sum_{m} s_n(m-i)\, s_n(m-k) \qquad (2.41)$$

เมื่อ αk เป็นค่าของ αk ที่ลดค่าของ E_n ถ้าเราหา

$$\Phi_n(i,k) = \sum_{m} s_n(m-i)\, s_n(m-k) \qquad (2.42)$$

แล้วสมการ 2.42 สามารถเขียนเป็น

$$\sum_{k=1}^{p} \alpha k \Phi_n(i,k) = \Phi_n(i,k) \qquad ;i = 12 \ldots p \qquad (2.43)$$

ถ้าเรารู้ค่า P ก็สามารถจะวิเคราะห์หาค่า ของสัมประสิทธิ์การคาดเดา (αk) และสมการผิดพลาดการคาดเดากำลังสองของ $S_n(m)$ โดยใช้สมการ 2.40 ถึง 2.42 คือ การผิดพลาดการคาดเดากำลังสอง จะเป็น

$$E_n = \sum_{m} s_n^2(m) = \sum_{k=1}^{p} \alpha k \sum_{m} s_{n(m-k)} \qquad (2.44)$$

และใช้สมการที่ 2.44 เราสามารถหา โดย

$$E_n = \Phi_{n(0,0)} - \sum_{k=1}^{p} \alpha_k \Phi_n(0,k) \qquad (2.45)$$

2.7 การหาความคล้ายคลึงกันของรูปแบบ (Pattern Similarity Determination)

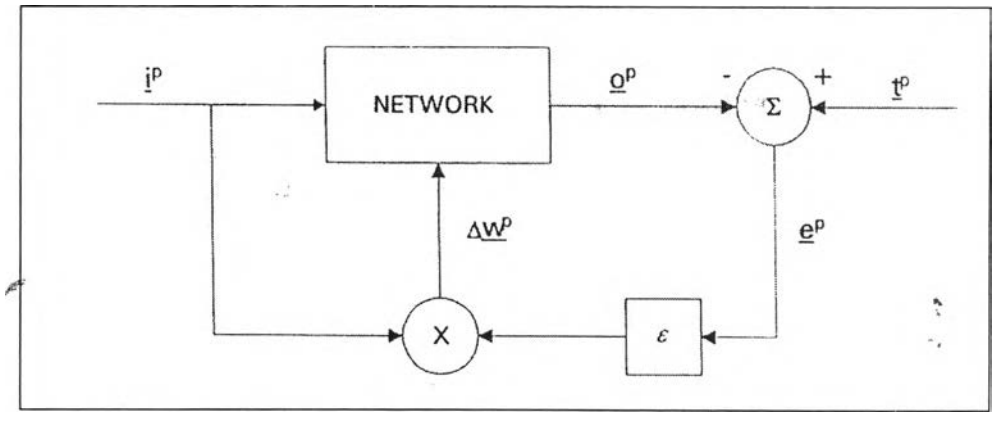
เป็นการหาค่าความใกล้เคียงของคำที่เราไม่ทราบเทียบกับคำอ้างอิงแต่ละคำ วิธีที่ใช้ในขั้นตอนนี้มีหลายวิธี เช่น DTW, HMM, นิวรอลเน็ตเวิร์ค DTW ใช้เทคนิคการปรับยืดขยายหรือหด รูปคลื่นสัญญาณตามแกนเวลาแบบไดนามิค วิธีนี้ถูกนำมาใช้ในการรู้จำเสียงสระภาษาไทย โดย ธีระ ภัทราพรนันท์ (2538) วิธี HMM ถูกนำมาใช้ในการรู้จำเสียงตัวเลขภาษาไทย โดย เสาวลักษณ์ อารีย์พงศา (2538) ขีดจำกัดของวิธี DTW และวิธี HMM คือ เมื่อเพิ่มจำนวนคำที่ต้องการรู้จำเสียงมากขึ้นทำให้เสียเวลาในการทดสอบมากขึ้น เพราะต้องทดสอบกับแบบอ้างอิงของเสียงทุกคำ ส่วนวิธีนิวรอลเน็ตเวิร์คยังคงใช้เวลาในการทดสอบเท่าเดิม เพราะนิวรอลเน็ตเวิร์ค เก็บความรู้เกี่ยวกับลักษณะของเสียงทุก ๆ คำ รวมอยู่ในน้ำหนักการเชื่อมต่อ ไม่ได้แยกเก็บเป็นแบบอ้างอิงสำหรับแต่ละคำ อย่างไรก็ตามถ้าเพิ่มจำนวนคำขึ้นมาก ๆ ต้องเพิ่มขนาดของนิวรอลเน็ตเวิร์ค ขึ้นด้วย เพื่อให้นิวรอลเน็ตเวิร์ค มีความสามารถเพียงพอในการรู้จำเสียง

2.7.1 นิวรอลเน็ตเวิร์ค (Neural Networks)

นิวรอลเน็ตเวิร์ค แบ่งแยกตามลักษณะของการเรียนรู้ได้เป็น 2 ชนิด คือ unsupervised learning และ supervised learning ในวิทยานิพนธ์นี้เลือกใช้นิวรอลเน็ตเวิร์ค แบบ multi-layer perceptron ซึ่งอยู่ในประเภท supervised learning

2.7.1.1 ขั้นตอนการฝึก (training) นิวรอลเน็ตเวิร์ค

Multi-layer perceptron neural network ใช้การฝึก (training) แบบ error backpropagation หรือ generalized delta rule ดังแสดงในรูปที่ 2.22



รูปที่ 2.22 โครงสร้างของการฝึก

โดยที่ $\underline{i}^p$ แทนค่าเวกเตอร์ input pattern ลำดับที่ p

$\underline{o}^p$ แทนค่าเวกเตอร์ output pattern ลำดับที่ p ที่ได้จากเน็ตเวิร์ค

$\underline{\omega}^p$ แทนค่าเวกเตอร์ network weights เมื่อใส่ค่าอินพุตลำดับที่ p เข้าสู่เน็ตเวิร์ค

$\underline{t}^p$ แทนค่าเวกเตอร์ output pattern ลำดับที่ p ที่ต้องการ

ε แทนค่า learning rate

$\underline{e}^p$ แทนค่าเวกเตอร์ความผิดพลาดของเอาต์พุตลำดับที่ p

นิวรอลเน็ตเวิร์คจะเรียนรู้จาก แต่ละตัวอย่างคู่ข้อมูลอินพุตเอาต์พุต ($\underline{i}^p$, $\underline{t}^p$) ที่อยู่ในชุดฝึก (training set) ซึ่งมีขั้นตอนพื้นฐานดังนี้

2.7.1.1.1 ป้อนค่าเวกเตอร์อินพุต (input vector) ให้กับระดับข้อมูลเข้า (input layer) ของนิวรอลเน็ตเวิร์ค

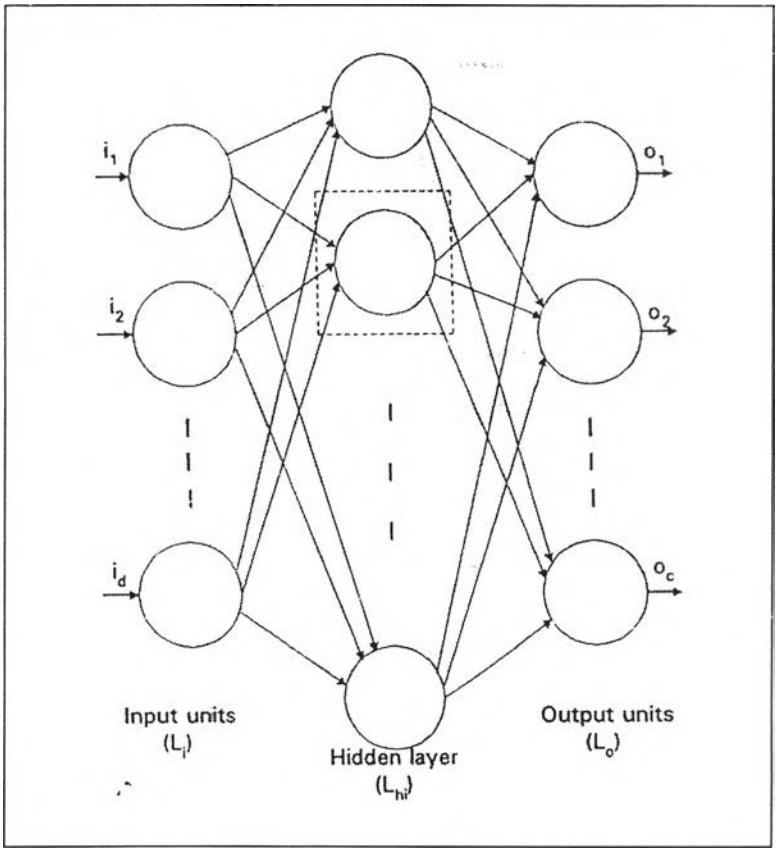
2.7.1.1.2 'Feed forward' หรือแพร่กระจายค่าอินพุต (input) เพื่อหาค่าเอาต์พุต (output) ของทุกโหนด

2.7.1.1.3 เปรียบเทียบค่าเอาต์พุต $\underline{O}^p$ ในระดับข้อมูลออก (output layer) กับค่าเอาต์พุตที่ต้องการ $\underline{t}^p$

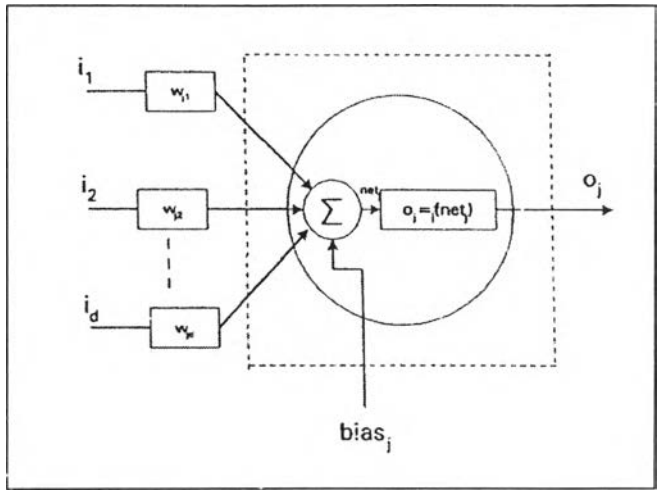
2.7.1.1.4 คำนวณและแพร่กระจายค่าความผิดพลาด ในทิศทางย้อนกลับ (เริ่มจากระดับข้อมูลออก (output layer)) ตลอดทั้งเน็ตเวิร์ค

2.7.1.15 ลดค่าความผิดพลาดที่แต่ละระดับ (layer) โดยการปรับค่าน้ำหนัก (weight) ที่เชื่อมต่อระหว่างโหนดของแต่ละระดับ

การฝึกนี้จะถูกทำซ้ำไปเรื่อย ๆ จนกว่าค่าความผิดพลาดจะอยู่ในระดับที่ยอมรับได้จึงจะยุติการฝึก ค่าน้ำหนักการเชื่อมต่อ (connection weight) ของนิวรอลเน็ตเวิร์คที่ผ่านการฝึกแล้วเปรียบเทียบได้กับความรู้ที่ได้รับจากการฝึกจากตัวอย่างคู่ข้อมูลอินพุตเอาต์พุต ดังนั้นถ้ามีตัวอย่างคู่ข้อมูลอินพุตเอาต์พุตที่หลากหลาย จะทำให้ นิวรอลเน็ตเวิร์ค มีความรู้เพียงพอที่จะใช้ในการเปรียบเทียบเสียง



รูปที่ 2.23 โครงสร้างของ multi-layer perceptron neural network



รูปที่ 2.24 รายละเอียดของโหนดในนิวรอลเน็ตเวิร์ค

รูปที่ 2.23 แสดงโครงสร้างของ multi-layer perceptron neural network ซึ่งประกอบด้วย ระดับข้อมูลเข้า (input layer), ระดับซ่อนตัว (hidden layer) ซึ่งอาจมีมากกว่า 1 ระดับ และระดับข้อมูลออก (output layer) ระดับข้อมูลเข้า (input layer) มีจำนวนโหนดเท่ากับ d โหนด ระดับข้อมูลออก (output layer) มีจำนวนโหนดเท่ากับ c โหนด ที่แต่ละโหนดในชั้นใด ๆ จะมีค่าน้ำหนักการเชื่อมต่อที่เชื่อมต่อไปยังโหนดที่อยู่ในชั้นถัดไปที่อยู่ติดกันเท่านั้น รูปที่ 2.24 แสดงรายละเอียดของโหนดในนิวรอลเน็ตเวิร์ค โดยที่ ค่า net input ที่โหนด j แสดงได้ดังนี้

$$net_j^p = \sum \omega_{ji} \tilde{o}_i^p + bias_j \qquad (2.46)$$

เมื่อ $\tilde{o}_i^p = o_i^p$ ถ้าอินพุตเป็น ค่าเอาต์พุตของโหนด ในระดับ (layer) ที่อยู่ข้างหน้า (เมื่อ j เป็นโหนดในระดับซ่อนตัว (hidden layer) และระดับข้อมูลออก (output layer)

$= i_i^p$ ถ้าอินพุตเป็นค่าข้อมูลอินพุตที่ป้อนเข้าสู่เน็ตเวิร์ค

ω_{ji} เป็นค่าน้ำหนักการเชื่อมต่อที่เชื่อมต่อจากโหนด i ไปยังโหนด j ที่อยู่ในระดับถัดไป

$bias_j$ เป็นค่าที่ใช้ปรับให้ net_j มีค่าไม่เท่ากับ $\tilde{o}_i^p$ ศูนย์ ในกรณีที่ ทุกโหนดมีค่าเป็นศูนย์หมด

$$O_j^p = f_j(net_j) \qquad (2.47)$$

ค่าเอาต์พุต o_j ของโหนด j สามารถคำนวณได้จาก net_j ดังนี้

โดยที่ฟังก์ชันกระตุ้น (activation function) เป็นฟังก์ชันเพิ่มและฟังก์ชันที่สามารถหาอนุพันธ์ได้ (differentiable) ในวิทยานิพนธ์นี้เลือกใช้ฟังก์ชัน sigmoid

$$f_j(net_j) = \frac{1}{1 + e^{-net_j}} \qquad (2.48)$$

2.7.1.2 การปรับค่าน้ำหนักการเชื่อมต่อ

เวกเตอร์ค่าความผิดพลาดของเอาต์พุต สำหรับตัวอย่างคู่ข้อมูลอินพุตเอาต์พุตที่ p กำหนดโดย

$$\underline{e^p} = \underline{t^p} - \underline{o^p} \qquad (2.49)$$

$$E_p = \frac{1}{2}\sum_j \left(t_j^p - o_j^p\right)^2 \qquad (2.50)$$

E_p แทนค่าความผิดพลาดของเอาต์พุต สำหรับตัวอย่างคู่ข้อมูลอินพุตเอาต์พุตที่ p หาได้จากหลักการของการปรับค่าน้ำหนักการเชื่อมต่อใน backpropagation training เริ่มต้นจากการคำนวณพื้นผิวของค่าความผิดพลาด E และคำนวณค่าเกรเดียนต์ (gradient) ของ E เทียบกับค่าน้ำหนักการเชื่อมต่อ $\partial E/\partial \omega_{ji}$ การปรับค่าน้ำหนักการเชื่อมต่อ $\Delta\omega_{ji}$ จะปรับค่าเป็นสัดส่วนกับ $-\partial E/\partial \omega_{ji}$ เพื่อให้การปรับน้ำหนักการเชื่อมต่อเป็นไปในทิศทางที่ลดค่าผิดพลาดลง สมการสำหรับการปรับค่าน้ำหนักการเชื่อมต่อแสดง ได้ดังนี้

$$\Delta^p \omega_{ji} = \varepsilon \delta_j^p \tilde{o}_i^p \qquad (2.51)$$

เมื่อ มีค่าตามที่แสดงในสมการ 2.46

ε คือค่า learning rate ซึ่งเป็นค่าคงที่และมีค่าเป็นบวก

$$\delta_j^p = -\frac{\partial E_p}{\partial net_j^p} \qquad (2.52)$$

สำหรับค่า δ_j^p คือ ค่าความไว (sensitivity) ของค่าความผิดพลาดเทียบกับค่า net input ที่โหนด j

สำหรับโหนดในระดับข้อมูลออก (output layer) $\delta_j^p = (t_j^p - o_j^p) f_j'(net_j^p)$

สำหรับโหนดใน internal layer $\delta_j^p = f_j'(net_j^p)\Sigma_n \delta_n^p \omega_{nj}$ เมื่อ δ_n^p เป็นค่าความไว (sensitivity) ในชั้นถัดออกไป

อนุพันธ์ของฟังก์ชันกระตุ้นชนิด sigmoid อยู่ในรูปที่คำนวณได้ง่าย ซึ่งนับเป็นข้อดีของฟังก์ชันชนิดนี้ กำหนดโดย

$$f_j'(net_j^p) = o_j^p(1 - o_j^p) \qquad (2.53)$$

2.7.1.3 กฎเกณฑ์การตัดสินใจ (Decision Rule)

การใช้กฎเกณฑ์ใดตัดสินว่าคำที่เราไม่ทราบ (Unknow word) คือเสียงใด เราจะต้องคำนึงถึงวิธีการที่ใช้ในขั้นตอน Pattern similarity determination เพราะต้องมีความสอดคล้องกัน เนื่องจาก multi-layer

perceptron neural network ที่ผ่านการฝึกแล้ว จะให้ค่าเอาต์พุตที่คล้ายคลึงกับค่าเอาต์พุตของตัวอย่างคู่ข้อมูลอินพุตเอาต์พุตที่มีค่าอินพุตของข้อมูลฝึกคล้ายกับค่าอินพุตที่ป้อนเข้ามา ดังนั้นเกณฑ์การตัดสินใจที่เลือกใช้คือการเลือกเสียงที่ตรงกับโหนดเอาต์พุตที่มีค่าเอาต์พุตสูงสุด อีกเหตุผลหนึ่งที่ใช้สนับสนุนเกณฑ์การตัดสินใจนี้ คือ การทำงานของ นิวรอลเน็ตเวิร์ค ขณะฝึกมีการปรับน้ำหนัก การเชื่อมต่อในทิศทางที่ลดความผิดพลาดให้เหลือน้อยที่สุด ดังนั้นนิวรอลเน็ตเวิร์คที่ผ่านการฝึกแล้วจะให้ค่าเอาต์พุตที่มีความผิดพลาดน้อยที่สุด บนพื้นฐานความรู้ที่นิวรอลเน็ตเวิร์คได้รับจากการฝึก โหนดเอาต์พุตที่มีค่าเอาต์พุตสูงสุดคำนวณได้จาก

$$\text{nodeoutput} = i \text{ เมื่อ } o_i = \text{Max}\,(o_1, o_2, \ldots, o_c) \qquad (2.54)$$

โดยที่ C คือ จำนวนกลุ่มของเสียงที่ต้องการรู้จำ