ระบบระบุตัวตนด้วยปัญญาประดิษฐ์

PERSON IDENTIFICATION SYSTEM WITH ARTIFICIAL INTELLIGENCE

โดย

นางสาวไอรดา ไชยเม็ง

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2561

ระบบระบุตัวตนด้วยปัญญาประดิษฐ์

PERSON IDENTIFICATION SYSTEM WITH ARTIFICIAL INTELLIGENCE

โดย

นางสาวไอรดา ไชยเม็ง

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2561

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบระบุตัวตนด้วยปัญญาประดิษฐ์

PERSON IDENTIFICATION SYSTEM WITH ARTIFICIAL INTELLIGENCE

โดย

นางสาวไอรดา ไชยเม็ง 58011473

อาจารย์ที่ปรึกษา

ผศ.ดร.ธเนศ พัฒนธาดาพงษ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2561

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2561

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบระบุตัวตนด้วยปัญญาประดิษฐ์

PERSON IDENTIFICATION SYSTEM WITH ARTIFICIAL INTELLIGENCE

ผู้จัดทำ

1. นางสาวไอรดา ไชยเม็ง 58011473

..... อาจารย์ที่ปรึกษา

(ผศ.ดร.ธเนศ พัฒนธาดาพงษ์)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์เรื่อง “ระบบระบุตัวตนด้วยปัญญาประดิษฐ์” สามารถประสบความสำเร็จลุล่วงไปได้ด้วยดี เนื่องจากได้รับความอนุเคราะห์ ความกรุณา และการสนับสนุนจาก ผศ.ดร.ธเนศ พัฒนธาดาพงษ์ ผศ.ดร.นภัทร สระเอี่ยม และ ดร.สมปอง วิเศษพานิชกิจ ที่ถ่ายทอดวิชาความรู้แก่ข้าพเจ้า โดยให้คำปรึกษา และข้อชี้แนะ ซึ่งเป็นคำแนะนำที่เป็นประโยชน์ในการศึกษาเพื่อทำปริญญานิพนธ์นี้ ตลอดทั้งความเสียสละเวลาแก่ข้าพเจ้ามาโดยตลอด จนทำให้ปริญญานิพนธ์นี้สำเร็จลุล่วงด้วยดี

ขอขอบพระคุณคณาจารย์และเจ้าหน้าที่ประจำภาควิชาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ได้อบรมสั่งสอนประสิทธิ์ประสาทวิชา ความรู้ และอำนวยความสะดวกในการจัดหาอุปกรณ์สำหรับการทำปริญญานิพนธ์ในครั้งนี้

ขอขอบพระคุณบิดา มารดา และครอบครัวที่มีส่วนช่วยในการให้กำลังใจ และให้โอกาสในการศึกษาแก่ข้าพเจ้าด้วย

ขอบคุณพี่ ๆ และเพื่อน ๆ ที่คอยให้กำลังใจ และช่วยเหลือข้าพเจ้าในเรื่องต่าง ๆ จนสามารถสำเร็จลุล่วงผ่านมาได้

นางสาวไอรดา ไชยเม็ง
ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบระบุตัวตนด้วยปัญญาประดิษฐ์

PERSON IDENTIFICATION SYSTEM WITH ARTIFICIAL INTELLIGENCE

โดย นางสาวไอรดา ไชยเม็ง 58011473

อาจารย์ที่ปรึกษา ผศ.ดร.ธเนศ พัฒนธาดาพงษ์

บทคัดย่อ

ปริญญานิพนธ์นี้ได้นำเทคโนโลยี “ปัญญาประดิษฐ์” มาประยุกต์ใช้กับระบบรู้จำ ซึ่งเป็นนวัตกรรมที่น่าสนใจและมีการใช้งานกันอย่างแพร่หลาย โดยจัดสร้างระบบรู้จำทั้งส่วนเสียงและใบหน้า เป็นการนำลักษณะทางกายภาพของมนุษย์มาระบุตัวตนเพื่อรักษาความปลอดภัยเบื้องต้น การออกแบบระบบแบ่งออกเป็น 3 ส่วน คือ ส่วนรู้จำด้วยเสียง ส่วนรู้จำด้วยใบหน้าและส่วนติดต่อผู้ใช้งาน การใช้งานระบบผู้ใช้งานสามารถเลือกการระบุตัวตนด้วยระบบรู้จำอย่างใดอย่างหนึ่ง เมื่อระบบรู้จำสามารถระบุตัวตนผู้ใช้งานได้ก็จะปลดล็อคประตูเพื่อให้ผู้ใช้งานเข้าไปยังพื้นที่ส่วนนั้น

ABSTRACT

This thesis has brought technology "Artificial Intelligence" applied to the recognition system which is an interesting innovation and is widely used, by creating a sound and face recognition system is the identification of human physical characteristics to identify the initial security. The system design is divided into 3 parts: voice recognition, face recognition and user interface. Using the system, users can choose to identify with one of the recognition systems. When the recognition system identifies the user, the user will unlock the door for the user to enter that area.

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

		หน้า
กิตติกรรมประกาศ		I
บทคัดย่อ		II
สารบัญ		III
สารบัญรูป		VI
สารบัญตาราง		VIII
บทที่ 1	**บทนำ**	**1**
	1.1 ความเป็นมาและความสำคัญของปัญหา	1
	1.2 วัตถุประสงค์	1
	1.3 ขอบเขตของปริญญานิพนธ์	1
บทที่ 2	**ทฤษฎีและหลักการที่เกี่ยวข้อง**	**2**
	2.1 ระบบรู้จำเสียงบุคคล (SPEAKER RECOGNITION)	2
	2.1.1 การดึงลักษณะเด่นของข้อมูล	3
	2.1.2 การจำแนกประเภทบุคคลด้วยเสียง	8
	2.2 ระบบรู้จำใบหน้าบุคคล(FACE RECOGNITION	9
	2.2.1 โครงข่ายประสาทเทียมแบบคอนโวลูชัน (CONVOLUTION NEURON NETWORK)	9
บทที่ 3	**การออกแบบและการจัดทำปริญญานิพนธ์**	**16**
	3.1 ระบบระบุตัวตนด้วยปัญญาประดิษฐ์ สำหรับระบุตัวตนด้วยเสียง	16
	3.1.1 การออกแบบระบบระบุตัวตนด้วยปัญญาประดิษฐ์ สำหรับระบุตัวตนด้วยเสียง	16
	3.1.2 การออกแบบการดึงลักษณะเด่น สำหรับระบุตัวตนด้วยเสียง	17

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

		หน้า
	3.1.3 การออกแบบการจำแนกบุคคลด้วยเสียง	22
	3.1.4 การหาค่าขีดแบ่งสำหรับตัดสินใจ	22
	3.2 ระบบระบุตัวตนด้วยปัญญาประดิษฐ์ สำหรับระบุตัวตนด้วยใบหน้า	23
	3.2.1 การตรวจจับและบันทึกใบหน้าบุคคล	24
	3.2.2 การเตรียมรูปภาพใบหน้าบุคคล	25
	3.2.3 การดึงคุณลักษณะเด่นของรูปภาพใบหน้าบุคคล	25
	3.2.4 การออกแบบระบบรู้จำบุคคลด้วยใบหน้า	27
	3.3 เครื่องมือที่ใช้ในการทดลอง	28
	3.3.1 RASPBERRY PI CAMERA MODULE	28
	3.3.2 KINOBO – USB MINI MICROPHONE	28
	3.3 การจัดเก็บผลการทดลอง	29
บทที่ 4	**ผลการทดลอง**	**31**
	4.1 ระบบรู้จำเสียงบุคคล	31
	4.1.1 การดึงคุณลักษณะเด่นของสัญญาณ	31
	4.1.2 ระบบรู้จำเสียงบุคคล	35
	4.1.3 การหาค่าขีดแบ่งสำหรับตัดสินใจ	35
	4.1.4 การทดลองเพื่อวัดประสิทธิภาพระบบรู้จำเสียงบุคคล	36
	4.2 ระบบรู้จำใบหน้าบุคคล	37
	4.2.1 การตรวจจับและบันทึกรูปภาพใบหน้าบุคคล	37
	4.2.2 การเตรียมรูปภาพใบหน้าบุคคล	38
	4.2.3 การดึงคุณลักษณะเด่นของรูปภาพใบหน้าบุคคล	38
	4.2.4 ระบบรู้จำ	38
	4.2.5 การทดลองเพื่อวัดประสิทธิภาพระบบรู้จำใบหน้าบุคคล	39

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

		หน้า
บทที่ 5	**สรุปผลและข้อเสนอแนะ**	**42**
	5.1 สรุปผล	42
	5.2 ข้อเสนอแนะ	42
บรรณานุกรม		**43**
ภาคผนวก ก	SOURCE CODE SPEAKER RECOGNITION	44
ภาคผนวก ข	SOURCE CODE FACIAL RECOGNITION	53

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่		หน้า
2.1	การรู้จำเสียงบุคคล	2
2.2	กระบวนการดึงคุณลักษณะเด่นของสัญญาณเสียงด้วยวิธีหาค่าสัมประสิทธิ์เซพสตรัม	3
2.3	ระบบรู้จำภาพใบหน้าโดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน (CNN)	9
2.4	โครงสร้างทั่วไปของโครงข่ายประสาทเทียมแบบคอนโวลูชัน (CNN)	9
2.5	ตัวอย่างการคอนโวลูชันระหว่างอินพุตกับตัวกรอง	10
2.6	MAX POOLING	11
2.7	AVERAGE POOLING	11
2.8	L_2 - NORM POOLING	12
2.9	ผลการทำงานของฟังก์ชัน SIGMOID	13
2.10	ผลการทำงานของฟังก์ชัน TANH	14
2.11	ผลการทำงานของฟังก์ชัน RELU	14
3.1	กระบวนการทำงานของระบบระบุตัวตนด้วยเสียง	16
3.2	การดึงลักษณะเด่นของเสียง	18
3.3	การแบ่งเฟรมของสัญญาณเสียง	18
3.4	ระยะห่างเชิงเส้นบนสเกลเมล	19
3.5	ระยะห่างเชิงเส้นบนสเกลเฮิร์ตซ์	20
3.6	ชุดตัวกรองความถี่บนสเกลทั้ง 26 ตัว	20
3.7	การยกระดับค่าสัมประสิทธิ์ด้วยสัญญาณไซนูซอยดอล (SINUSOIDAL LIFTER)	21
3.8	โฟวชาร์ตการทำงานของระบบระบุตัวตนด้วยใบหน้า	24
3.9	โฟวชาร์ตของกระบวนการตรวจจับและบันทึกรูปภาพใบหน้า	24
3.10	โฟวชาร์ตของกระบวนการเตรียมรูปภาพใบหน้าบุคคล	25

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่		หน้า
3.11	โฟวชาร์ตของกระบวนการดึงคุณลักษณะเด่นของรูปภาพใบหน้าบุคคล	26
3.12	การดึงคุณลักษณะของรูปภาพ	27
3.13	การเรียนรู้แบบโครงข่ายประสาทเทียม	27
3.14	RASPBERRY PI CAMERA MODULE	28
3.15	KINOBO – USB MINI MICROPHONE	29
4.1	การทำพรีเอมฟาซิส	31
4.2	การแบ่งเฟรมสัญญาณเสียง	32
4.3	การทำฟังก์ชันวินโดว์ในเฟรมที่ 11, 12 และ 13	32
4.4	สเปกตรัมเชิงกำลังของสัญญาณเสียงในโดเมนความถี่ในเฟรมที่ 11, 12 และ 13	33
4.5	สเปกตรัมของเฟรมที่ 11 ที่คูณกับตัวกรองตัวที่ 10 และ 20	33
4.6	การหาค่าสัมประสิทธิ์เซพสตรัลบนสเกลเมล 26 ค่าของเฟรมที่ 11, 12 และ 13	34
4.7	ค่าสัมประสิทธิ์เดลต้าของเฟรมที่ 11, 12 และ 13	35
4.8	การหาค่าขีดแบ่งสำหรับตัดสินใจ จากกราฟฮิสโทกราฟ	36
4.9	รูปภาพใบหน้าขนาด 100X100X3 พิกเซล สำหรับฝึกสอนโมเดล	38
4.10	การดึงคุณลักษณะเด่นของรูปภาพ	38
4.11	โมเดลสำหรับจำแนกบุคคลด้วยใบหน้า ชื่อ “MODEL_01.CKPT.META”	39

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่

3.1 การออกแบบตารางคอนฟูชันแมททริก สำหรับวัดประสิทธิภาพโมเดล 29

4.1 ผลการทดลองระบบรู้จำเสียงบุคคล 37

4.2 ผลการทดลองระบบรู้จำใบหน้าบุคคล 40

4.3 ผลการทดลองการวัดประสิทธิภาพข้อมูลของตาราง CONFUSION MATRIX (คอนฟูชั่น แมททริก) 41

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันการสร้างสิ่งอำนวยความสะดวกต่าง ๆ ให้กับมนุษย์จำเป็นต้องพึ่งพาเทคโนโลยี ซึ่งเทคโนโลยีที่เรียกว่า “ปัญญาประดิษฐ์” (Artificial Intelligence) หรือรู้จักในชื่อว่า “AI” หมายถึง ความฉลาดเทียมที่สร้างขึ้นให้กับสิ่งไม่มีชีวิต ซึ่งเป็นการเรียนรู้เกี่ยวกับกระบวนการการคิด การกระทำ การให้เหตุผล การปรับตัว หรือการอนุมานและการทำงานของสมอง ปัญญาประดิษฐ์เป็นสาขาหนึ่งใน ด้านวิทยาการคอมพิวเตอร์และวิศวกรรม และสามารถนำไปประยุกต์ใช้กับศาสตร์อื่น ๆ ได้หลากหลาย ซึ่งระบบรู้จำนั้น ก็อยู่ในส่วนหนึ่งของศาสตร์ปัญญาประดิษฐ์เช่นกัน

ระบบรู้จำ (Recognition System) เป็นนวัตกรรมที่กำลังเป็นที่น่าสนใจ และเริ่มใช้งาน กันอย่างแพร่หลาย ซึ่งในประเทศไทยนั้นได้มีงานวิจัยในด้านนี้มากกว่า 20 ปีแล้ว[5] ทางผู้จัดทำจึงได้ นำระบบรู้จำนี้ไปประยุกต์ใช้งานจริง ในการจัดสร้างระบบระบุตัวตนทั้งส่วนของเสียงและใบหน้า

1.2 วัตถุประสงค์

ปริญญานิพนธ์ที่ได้จัดทำขึ้นเพื่อศึกษาและสร้างระบบระบุตัวตนด้วยปัญญาประดิษฐ์ มี วัตถุประสงค์ดังนี้

1) เพื่อศึกษาและพัฒนาระบบรู้จำพร้อมนำไปประยุกต์ใช้งาน
2) เพื่อศึกษาและพัฒนาระบบรู้จำเสียงโดยใช้วิธีการ MFCC
3) เพื่อศึกษาและพัฒนาระบบรู้จำใบหน้าโดยใช้วิธีการ CNN
4) เพื่อศึกษาและพัฒนาการเรียนรู้แบบปัญญาประดิษฐ์พร้อมนำไปประยุกต์ใช้งาน

1.3 ขอบเขตของปริญญานิพนธ์

ปริญญานิพนธ์เรื่อง “ระบบระบุตัวตนด้วยปัญญาประดิษฐ์” สำหรับระบุตัวตนด้วยเสียง และใบหน้า เป็นการประยุกต์การใช้งานระบบรู้จำเสียงและใบหน้าร่วมกับการเรียนรู้แบบ ปัญญาประดิษฐ์ (Artificial Intelligence) โดยระบบจะแบ่งการรู้จำออกเป็น 2 ส่วนคือ ส่วนรู้จำเสียง ซึ่งจะรู้จำเสียงผู้พูดทั้งหมด 5 คน และส่วนรู้จำใบหน้าจะรู้จำใบหน้าบุคคลทั้งหมด 10 คน ผลลัพธ์จาก การระบุตัวตนของระบบรู้จำเสียงและใบหน้านี้นำไปประยุกต์ใช้งานในการเข้าพื้นที่ส่วนบุคคล

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

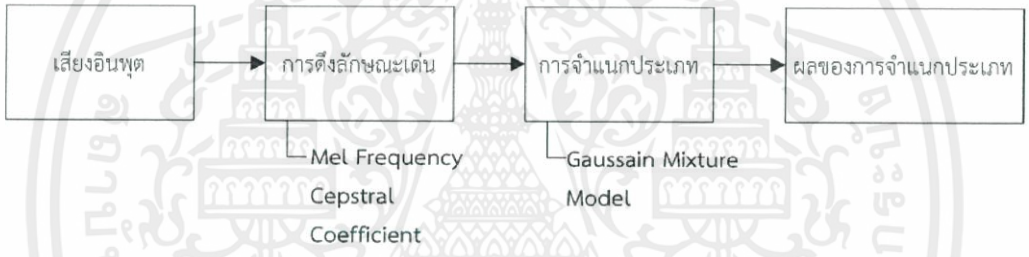
บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

ปริญญานิพนธ์เรื่อง “ระบบระบุตัวตนด้วยปัญญาประดิษฐ์” ได้มีการอธิบายทฤษฎีและหลักการที่เกี่ยวข้อง ซึ่งมีเนื้อหาดังต่อไปนี้

2.1 ระบบรู้จำเสียงบุคคล (Speaker Recognition)

การรู้จำบุคคล คือการระบุตัวตนบุคลลจากลักษณะของเสียงเพื่อตอบคำถาม “ใครกำลังพูด” ซึ่งสามารถลดความยุ่งยากในการระบุตัวตนจากการจดจำบุคคลด้วยคำพูดเป็นการจดจำบุคคลด้วยเสียงแทน และสามารถรับรองความถูกต้องหรือตรวจสอบตัวตนของบุคคลเพื่อรักษาความปลอดภัย การรู้จำเสียงมีองค์ประกอบหลักที่สำคัญ แสดงดังรูปที่ 2.1



รูปที่ 2.1 การรู้จำเสียงบุคคล

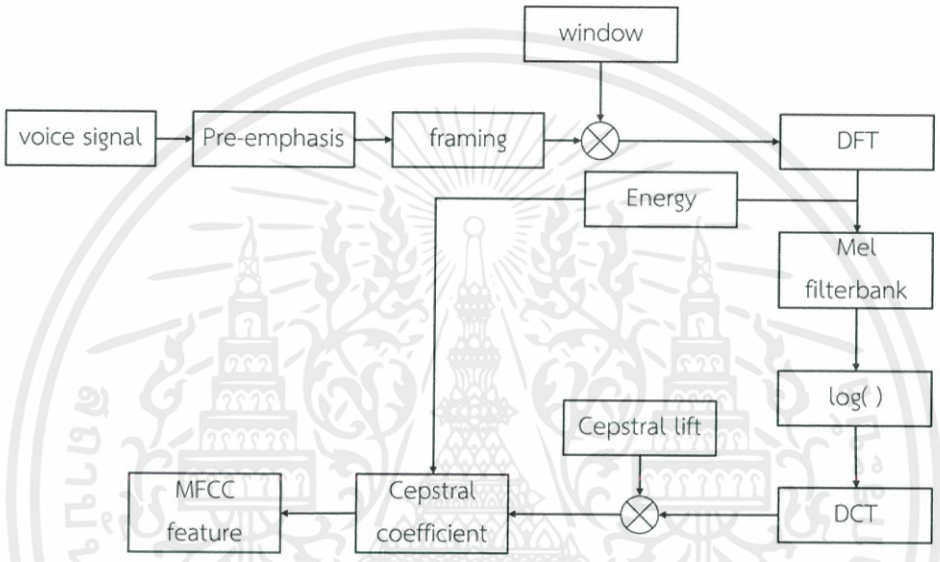
จากรูปที่ 2.1 การทำงานของการรู้จำแบบจะเริ่มต้นเมื่อมีอินพุตเข้ามา ระบบจะทำการดึงลักษณะเด่นของอินพุตด้วยวิธีหาค่าสัมประสิทธิ์เซพสตรัมบนสเกลเมล (Mel Frequency Cepstral Coefficient: MFCC) เพื่อลดปริมาณของข้อมูลให้เหลือเฉพาะข้อมูลที่จำเป็นต่อระบบและส่งผลให้การจำแนกบุคคลมีประสิทธิภาพมากยิ่งขึ้น จากนั้นนำค่าสัมประสิทธิ์เซพสตรัมเหล่านี้เข้าสู่กระบวนการจำแนกบุคคลโดยใช้วิธีการหารูปแบบจำลองผสมของเกาส์ (Gaussian Mixture Model: GMM) เพื่อหารูปแบบของแบบจำลอง ผลการจำแนกที่ได้จะเป็นรูปแบบจำลองผสมของสัญญาณเสียงอินพุตที่เข้ามา

การรู้จำเสียงบุคคลจะมี 2 ขั้นตอนหลักที่สำคัญในการดำเนินการได้แก่ การดึงลักษณะเด่นด้วยวิธี MFCC และการจำแนกบุคคลด้วยเสียงด้วยวิธี GMM ซึ่งจะกล่าวถึงในบทนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1 การดึงลักษณะเด่นของข้อมูล

วิธีการหาค่าสัมประสิทธิ์เซพสตรัมบนสเกลเมลเป็นวิธีการดึงคุณลักษณะเด่นของเสียงเพื่อสร้างค่าสัมประสิทธิ์เซพสตรัมซึ่งวิธีการนี้เป็นที่นิยมในปัจจุบัน เพราะจะได้ขนาดข้อมูลจะมีขนาดเล็ก สามารถให้รายละเอียดที่ดีสำหรับการวิเคราะห์สัญญาณเสียงช่วงความถี่ต่ำและยังมีความทนทานต่อสัญญาณรบกวน (Noise) แสดงดังรูปที่ 2.2



รูปที่ 2.2 กระบวนการดึงคุณลักษณะเด่นของสัญญาณเสียงด้วยวิธีหาค่าสัมประสิทธิ์เซพสตรัม

จากรูปที่ 2.2 ขั้นตอนการดึงคุณลักษณะเด่นของสัญญาณเสียง มีดังนี้

1) การทำพรีเอมฟาซิส (Pre-emphasis)

การทำพรีเอมฟาซิสเป็นการนำสัญญาณมาผ่านวงจรกรองความถี่สูงผ่าน ทำให้อัตราส่วนของสัญญาณเสียงต่อสัญญาณรบกวนมีค่าคงที่ ตลอดทุกช่วงความถี่ เพื่อลดสัญญาณรบกวนก่อนนำสัญญาณไปประมวลผลในขั้นตอนต่อไป ดังสมการที่ 2.1

$$H(z) = 1 - \alpha z^{-1} \tag{2.1}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) การแบ่งช่วงสัญญาณ (Framing)

สัญญาณที่ผ่านการทำพรีเอมฟาซิสจะถูกแบ่งออกเป็นเฟรม ๆ เนื่องจากสัญญาณเสียงพูดเป็นสัญญาณไม่หยุดนิ่งมีการเปลี่ยนแปลงไปตามเวลา การแบ่งข้อมูลออกเป็นเฟรมทำให้สามารถวิเคราะห์รูปร่างสัญญาณเสียง โดยอนุมานว่าสัญญาณเสียงในเฟรมนั้นเป็นสัญญาณหยุดนิ่งและไม่เปลี่ยนแปลงไปตามเวลา จากนั้นนำแต่ละเฟรมมาซ้อนทับกัน เพื่อให้ข้อมูลในการวิเคราะห์มีความต่อเนื่องกัน

3) การทำวินโดว์ (Windowing)

การแบ่งสัญญาณเสียงออกเป็นเฟรมทำให้เกิดความไม่ต่อเนื่องที่ขอบของเฟรมทั้งสองด้าน ส่งผลต่อการวิเคราะห์รูปร่างสัญญาณอาจทำให้เกิดความผิดพลาดได้ การคูณวินโดว์กับเฟรมของสัญญาณเสียงจะค่อย ๆ ลดแอมพลิจูดที่ปลายทั้งสองข้างของเฟรม เพื่อลดปัญหาการเปลี่ยนแปลงแอมพลิจูดของสัญญาณอย่างกระทันหัน ช่วยให้รูปร่างสัญญาณในโดเมนความถี่มีความเรียบมากยิ่งขึ้น

4) การแปลงดิสครีตฟูเรียร์ทรานฟอร์ม (DFT)

การแปลงสัญญาณจากโดเมนเวลาเป็นโดเมนความถี่ จะใช้เครื่องมือที่ชื่อว่า “ดิสครีตฟูเรียร์ทรานฟอร์ม (Discrete Fourier Transform: DFT)” ดังสมการที่ 2.2

$$X[k] = \sum_{n=1}^{N} x(k) e^{\frac{-j2\pi m}{N}} \tag{2.2}$$

โดยที่ $x(k)$ คือ เฟรมสัญญาณเสียงที่คูณกับแฮมมิ่งวินโดว์

$X[k]$ คือ สเปกตรัมของเฟรมสัญญาณเสียง $x(k)$

N คือ ขนาดของเฟรมสัญญาณเสียง $x(k)$

m คือ ตำแหน่งสัญญาณเสียงอินพุต ซึ่งมีค่าตั้งแต่ $m = 0$ ถึง $N-1$

จากนั้นทำการหาสเปกตรัมเชิงกำลังของสัญญาณเสียงในตัวแปร $P(k)$ ดังสมการที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$P[k]=|X[k]|^2 \tag{2.3}$$

5) ชุดตัวกรองความถี่บนสเกลเมล (Mel-scale filterbank)

ชุดตัวกรองความถี่บนสเกลเมล เป็นชุดตัวกรองที่ช่วยแปลงค่าสเปกตรัมของสัญญาณเสียงที่มีลักษณะเป็นเชิงเส้นให้เป็นมีค่าสเปกตรัมที่ไม่เป็นลักษณะเชิงเส้น โดยชุดตัวกรองจะมีรูปร่างเป็นสามเหลี่ยมที่มีความสมมาตรกันทั้งสองด้าน ซึ่งการสร้างชุดตัวกรองความถี่บนพื้นฐานของสเกลเมลมีขั้นตอนดังนี้

1. ทำการเปลี่ยนความถี่ต่ำสุด และความถี่สูงสุดบนสเกลทางความถี่เฮิร์ตซ์ให้เป็นความถี่สเกลเมล ดังสมการที่ 2.4 ดังนี้

$$M(f)=2595\ln\left(1+\frac{f}{700}\right) \tag{2.4}$$

2. ทำการแบ่งระยะทั้งหมด N จุด โดยค่า N คือ จำนวนจุดที่ต้องการบนสเกล สุดท้ายสเกลจะมีทั้งหมด $N+2$ จุด

3. จากนั้นทำการแปลงกลับเป็นสเกลทางความถี่เฮิร์ตซ์เช่นเดิม ดังสมการที่ 2.5 ดังนี้

$$M^{-1}(m)=700\left(\exp\left(\frac{M(f)}{2595}\right)-1\right) \tag{2.5}$$

4. ทำการสร้างชุดตัวกรองความถี่ โดยตัวกรองตัวแรกจะเริ่มที่จุดแรกของกลุ่มความถี่จากขั้นตอนที่ 3 ไปถึงจุดสูงสุดที่จุดที่สอง จากนั้นกลับไปสู่ศูนย์ที่จุดที่สาม และตัวกรองที่สองจะเริ่มที่จุดที่สองไปถึงจุดสูงสุดที่จุดที่สาม จากนั้นกลับไปสู่ศูนย์ที่จุดที่สี่ เป็นต้น โดยสร้างดังสมการที่ 2.6 ดังนี้

$$H_m[k]=\begin{cases}0, & k<f(m-1)\\ \dfrac{2(k-f(m-1))}{(f(m+1)-f(m-1))(f(m)-f(m-1))}, & f(m-1)\le k\le f(m)\\ \dfrac{2(f(m+1)-k)}{(f(m+1)-f(m-1))(f(m+1)-f(m))}, & f(m)\le k\le f(m+1)\\ 0, & k>f(m+1)\end{cases} \tag{2.6}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สุดท้ายจะได้ชุดกรองความถี่บนสเกลเมลสำหรับนำไปผ่านสเปกตรัมเชิงกำลังของแต่ละเฟรม

6) สเปกตรัมของสเกลเมล

เมื่อได้ชุดกรองความถี่บนสเกลเมลจากขั้นตอนที่ 5 แล้ว นำสเปกตรัมเชิงกำลังของแต่ละเฟรมไปผ่านชุดตัวกรอง ผลลัพธ์ที่ได้เป็นการคูณกันของชุดตัวกรองความถี่แต่ละตัวและสเปกตรัมเชิงกำลังแล้วหาค่าเฉลี่ยออกมา ซึ่งแต่ละเฟรมจะได้ผลลัพธ์จากการคูณกันนี้ทั้งหมด m ค่าตามจำนวนตัวกรองความถี่สเกลเมล โดยตัวกรองรูปสามเหลี่ยมนี้เป็นตัวกรองแบบแบรนด์พาส (Band pass) ดังสมการที่ 2.7 ดังนี้

$$S[k] = P[k] \cdot H_m[k] \tag{2.7}$$

โดยที่ $P[k]$ คือ สเปกตรัมเชิงกำลังของเฟรมสัญญาณเสียง

$H_m[k]$ คือ ผลตอบสนองทางขนาดของตัวกรอง

m คือ จำนวนตัวกรองความถี่สเกลเมล

$S[k]$ คือ ผลลัพธ์จากการนำสเปกตรัมเชิงกำลังของสัญญาณเสียงไปผ่านชุดตัวกรองแต่ละตัว

7) ลอการิทึม (Logarithm)

เฟรมสัญญาณเสียงที่ผ่านตัวกรองรูปสามเหลี่ยมแต่ละตัวจะนำไปผ่านฟังก์ชันลอการิทึม เพื่อให้สเปกตรัมเชิงกำลังของเฟรมสัญญาณเสียงอยู่บนสเกลลอการิทึม ดังสมการที่ 2.8

$$S_{mel}(k) = \log[S[k]] \tag{2.8}$$

8) ค่าสัมประสิทธิ์เซพสตรัมบนสเกลเมล (Mel Frequency Cepstral Coefficient)

นำผลลัพธ์จากขั้นตอนที่ 6 จะนำมาหาค่าสัมประสิทธิ์เซพสตรัมสเกลเมล ด้วยการแปลงโคซายน์แบบไม่ต่อเนื่อง (Discrete Cosine Transform) ดังสมการที่ 2.9

$$C(m) = \sqrt{\frac{2}{K}} \sum_{k=1}^{K} S_{mel}(k) \frac{1}{\sqrt{1+\delta_{m1}}} \cos\left(\frac{\pi}{2K}(2k-1)(m-1)\right) \tag{2.9}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผ่านการแปลงโคไซน์แบบไม่ต่อเนื่องแล้วจะได้ค่าสัมประสิทธิ์เซพสตรัมบนสเกลเมลที่มีสัมประสิทธิ์เท่ากับจำนวนตัวกรองในแต่ละเฟรม

9) การยกระดับค่าสัมประสิทธิ์เซพสตรัล (Cepstral Lifting)

การยกระดับค่าสัมประสิทธิ์เซพสตรัลจะทำให้ขนาดของค่าสัมประสิทธิ์มีความแตกต่างกันด้วยวิธีการนำค่าสัมประสิทธิ์เซพสตรัลคูณกับน้ำหนัก ส่งผลให้ระบบการรู้จำเสียงบุคคลดียิ่งขึ้น โดยขนาดของค่าสัมประสิทธิ์เซพสตรัลที่มีค่ามากจะมีค่าแอมพลิจูดแตกต่างน้อยกว่าขนาดของค่าสัมประสิทธิ์เซพสตรัลที่มีค่าน้อย

10) การคำนวณพลังงานรวมของแต่ละเฟรม (Total Energy in each frame)

ทำการรวมค่าสเปกตรัมเชิงกำลังจากขั้นตอนที่ 4 ให้เป็นค่าพลังงานรวมของแต่ละเฟรมในรูปแบบค่าสเกลาร์ จากนั้นนำค่าพลังงานรวมไปแทนที่ค่าสัมประสิทธิ์เซพสตรัมบนสเกลเมลตำแหน่งแรก

11) ค่าสัมประสิทธิ์เดลต้าของเซพสตรัมบนสเกลเมล (Delta MFCC)

การเพิ่มค่าสัมประสิทธิ์เดลต้าของเซพสตรัมบนสเกลเมล เพื่อเพิ่มประสิทธิภาพให้ระบบหรือสร้างความแตกต่างให้คุณลักษณะเด่นของสัญญาณเสียงให้เป็นคุณสมบัติที่ดี สามารถแสดงการหาค่าสัมประสิทธิ์เดลต้าของเซพสตรัมบนสเกลเมล ดังสมการที่ 2.10 ดังนี้

$$d_t = \frac{\sum_{n=1}^{N} n(c_{t+n} - c_{t-n})}{2\sum_{n=1}^{N} n^2} \qquad (2.10)$$

โดยที่ d_t คือ ค่าสัมประสิทธิ์เดลต้าของเฟรมตำแหน่ง t

t คือ ตำแหน่งของเฟรม โดยนับจากเฟรมที่ 0 ถึง T

N คือ จำนวนเดลต้าในที่นี่กำหนดเป็น 2

ผลลัพธ์สุดท้ายของการดึงคุณลักษณะเด่นในแต่ละเฟรมของสัญญาณเสียงจะได้ค่าสัมประสิทธิ์เซพสตรัมบนสเกล m ค่า รวมกับค่าสัมประสิทธิ์เดลต้าที่มีจำนวนเท่ากับค่าสัมประสิทธิ์เซพสตรัมบนสเกลเมลอีก m ค่า ดังนั้นตัวแทนของข้อมูลในแต่ละเฟรมจะมีทั้งหมด $2m$ ค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 การจำแนกประเภทบุคคลด้วยเสียง

เมื่อสัญญาณเสียงผ่านกระบวนการขั้นตอนต่าง ๆ ตามที่ได้กล่าวมาขั้นต้น สัญญาณเสียงนั้น ๆ จะได้ตัวแทนสัญญาณเสียงเพื่อใช้ในการจำแนกรู้จำบุคคลจากการหาค่าสัมประสิทธิ์ทั้ง 3 ชนิด ซึ่งค่าสัมประสิทธิ์เหล่านี้จะถูกนำไปเรียนรู้ฝึกฝนเพื่อสร้างรูปแบบสำหรับจำแนกบุคคลด้วยเสียง วิธีการที่ใช้ในการสร้างรูปแบบจำแนกรู้จำในปริญญานิพนธ์นี้คือ “แบบจำลองส่วนผสมของเกาส์” (Gaussian Mixture Model: GMM) ซึ่งเป็นรูปแบบการจัดกลุ่มความน่าจะเป็น เพื่อเป็นตัวแทนของการแสดงตัวของประชากรย่อยในกลุ่มประชากรรวม โดยมีแนวคิดคือ การประมาณการกระจายความน่าจะเป็นของคลาส ด้วยการรวมกันแบบเชิงเส้นของการแจกแจงแบบเกาส์ ความน่าจะเป็นของเวกเตอร์คุณลักษณะของโมเดล สามารถคำนวณได้ดังสมการที่ 2.11

$$P(X|\lambda)=\sum_{k=1}^{K} w_k P_K\left(X\middle|\mu_k,\sigma_k^2\right) \tag{2.11}$$

โดยที่ $P_k\left(X\middle|\mu_k,\sigma_k^2\right)$ คือ การกระจายแบบเกาส์ ดังสมการที่ 2.12

$$P_k\left(X\middle|\mu_k,\sigma_k^2\right)=\frac{1}{\sqrt{2\pi\sigma_k^2}}e^{\frac{\left(X-\mu_k\right)^2}{2\sigma_k^2}} \tag{2.12}$$

โดยที่ X_i คือข้อมูลสำหรับฝึกสอนของคลาส λ

μ คือ ค่าเฉลี่ย

σ คือ ความแปรปรวน

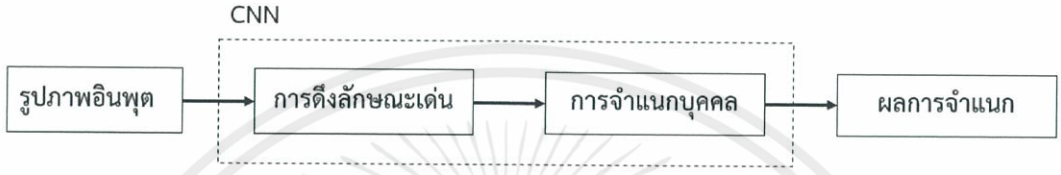
w คือ ค่าน้ำหนักของกลุ่มที่ k

ขั้นตอนแรกจะระบุกลุ่มให้ข้อมูลอินพุตด้วยการหาค่า K-mean และระบุค่าน้ำหนักให้แต่ละกลุ่ม โดยค่าน้ำหนักจะเท่ากับ $w=\frac{1}{k}$ จากนั้นคำนวณการกระจายแบบเกาส์ของแต่ละกลุ่ม k ด้วยพารามิเตอร์ μ, σและ wกระทำวนซ้ำด้วยค่าอับเดตของพารามิเตอร์ทั้ง 3 ตัว จนกระทั่งเสร็จสิ้นด้วยอัลกอริทึมความคาดหวังสูงสุด (Expectation Maximization algorithm: EM)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ระบบรู้จำใบหน้าบุคคล (Face Recognition)

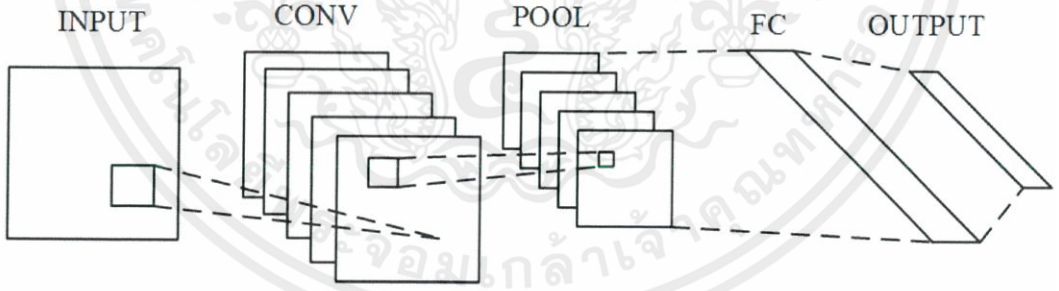
การรู้จำภาพใบหน้าด้วยโครงข่ายประสาทเทียมแบบคอนโวลูชันนั้นจะรวมการดึงลักษณะเด่นและการจำแนกบุคคลเข้าไว้ในวิธีการเดียวกัน ทำให้การรู้จำภาพใบหน้ามีกระบวนการการทำงานที่เร็วขึ้นและไม่จำเป็นต้องหาวิธีการอื่น ๆ มาใช้ ส่งผลให้โครงข่ายประสาทเทียมแบบคอนโวลูชันได้รับความนิยมในการใช้งานในปัจจุบัน แสดงดังรูปที่ 2.3



รูปที่ 2.3 ระบบรู้จำภาพใบหน้าโดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน (CNN)

2.2.1 โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolution Neuron Network)

โครงข่ายประสาทเทียมแบบคอนโวลูชัน หรือ Convolution Neuron Network (CNN) เป็นโครงข่ายประสาทแบบหลายชั้น (Multilayer perceptron) นิยมใช้ในงานวิเคราะห์รูปภาพหรือข้อมูลที่มีความต่อเนื่อง เนื่องจาก CNN สามารถดึงลักษณะเด่นและจำแนกได้โดยตรง โครงสร้างทั่วไปของ CNN ประกอบด้วย ชั้นอินพุต (INPUT), ชั้นคอนโวลูชัน (CONV), ชั้นพูลลิ่ง (POOL), ชั้นฟูลลี่ (FC) และชั้นเอาต์พุต (OUTPUT) แสดงดังรูปที่ 2.4



รูปที่ 2.4 โครงสร้างทั่วไปของโครงข่ายประสาทเทียมแบบคอนโวลูชัน (CNN)

การทำงานในแต่ละชั้นสามารถอธิบายได้ดังต่อไปนี้

1) ชั้นอินพุต (Input layer)

ชั้นนี้เป็นการนำข้อมูลอินพุตเข้าสู่ระบบ

2) ชั้นคอนโวลูชัน (Convolution layer)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชั้นนี้เป็นการนำรูปภาพอินพุตมาคอนโวลูชันกับตัวกรอง (Filter) โดยค่าสัมประสิทธิ์ของตัวกรองได้จากการสุ่ม ซึ่งเอาต์พุตของการทำคอนโวลูชันจะมีขนาดเท่ากับ $W_2 \times H_2 \times D_2$ สามารถคำนวณได้จากสมการที่ 2.13

$$W_2 = (W_1 - F + 2P)/S + 1 \tag{2.13}$$

โดยที่ W_1 คือ ขนาดความกว้างของอินพุต

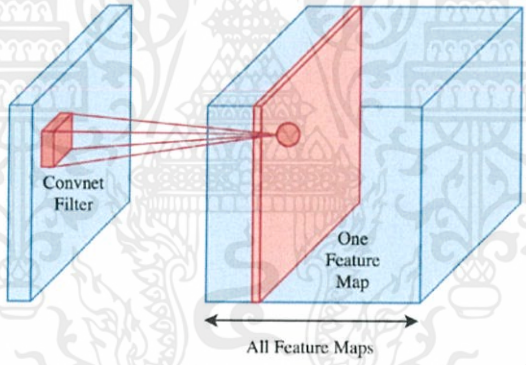
F คือ ขนาดของตัวกรอง

P คือ ขนาดของ Zero-Padding

S คือ ระยะการเลื่อนของตัวกรอง

ซึ่งการหาขนาดความยาวของอินพุต H_2 สามารถคำนวณจากสมการที่ 2.13 ได้เช่นกัน โดยเปลี่ยนตัวแปรจาก W เป็น H และ D_2 คือ จำนวนของตัวกรอง

ตัวอย่างการทำคอนโวลูชันระหว่างอินพุตกับตัวกรอง แสดงดังรูปที่ 2.5



รูปที่ 2.5 ตัวอย่างการคอนโวลูชันระหว่างอินพุตกับตัวกรอง

จากตัวอย่างเอาต์พุตที่ได้เกิดจากการนำค่าสัมประสิทธิ์ของตัวกรอง คูณกับค่าสัมประสิทธิ์ของอินพุต จำนวนค่าที่ได้จะเท่ากับค่าสัมประสิทธิ์ของตัวกรอง จากนั้นนำมาบวกกันจะได้เอาต์พุตเท่ากับ 1 ตัว ต่อการเลื่อนตัวกรอง 1 ครั้ง ทำซ้ำตามกระบวนการเดิมจนครบขนาดของอินพุต ซึ่งเอาต์พุตจะมีขนาดเท่าเดิมแต่จำนวนจะเท่ากับจำนวนของตัวกรอง

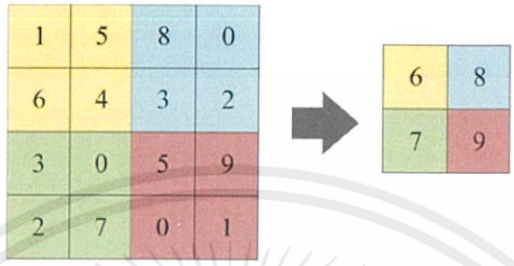
3) ชั้นพูลลิ่ง (Pooling layer)

ในชั้นนี้ทำหน้าที่ลดปริมาณข้อมูลในเชิงพื้นที่จากชั้นคอนโวลูชัน เพื่อแก้ปัญหา Overfitting และลดจำนวนพารามิเตอร์ โดยวิธีการทำพูลลิ่งที่นิยมมีอยู่ 3 รูปแบบ คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. Max pooling

การทำพูลลิ่งด้วยค่าสูงสุดนั้นจะเลือกค่าสูงสุดในแต่ละบล็อคที่ตัวกรองครอบข้อมูลอินพุตอยู่เพื่อเป็นตัวแทนข้อมูลแต่ละบล็อคนั้น ๆ แสดงดังรูปที่ 2.6



รูปที่ 2.6 Max pooling

จากรูปที่ 2.6 การทำพูลลิ่งด้วยค่าสูงสุดระหว่างอินพุตขนาด 4x4 กับตัวกรองขนาด 2x2 และมีการเลื่อนตัวกรองไปครั้งละ 2 ช่อง เอาต์พุตที่ได้คือตัวแทนที่มีค่าสูงสุดของแต่ละบล็อค และมีขนาดลดลงครึ่งหนึ่งของขนาดอินพุต

2. Average pooling

การทำพูลลิ่งด้วยค่าเฉลี่ยนั้น จะคำนวณค่าเฉลี่ยภายในแต่ละบล็อคที่ตัวกรองครอบข้อมูลอินพุตอยู่เพื่อเป็นตัวแทนข้อมูลแต่ละบล็อคนั้น ๆ แสดงดังรูปที่ 2.7

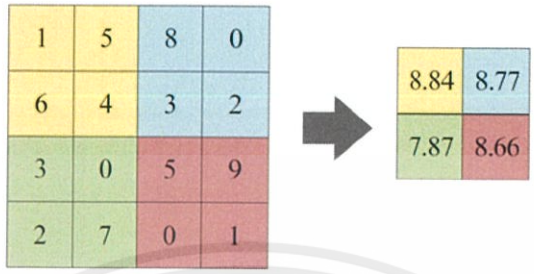
รูปที่ 2.7 Average pooling

จากรูปที่ 2.7 การทำพูลลิ่งด้วยค่าสูงสุดระหว่างอินพุตขนาด 4x4 กับตัวกรองขนาด 2x2 และมีการเลื่อนตัวกรองไปครั้งละ 2 ช่อง เอาต์พุตที่ได้คือตัวแทนที่มีค่าเฉลี่ยของแต่ละบล็อค และมีขนาดลดลงครึ่งหนึ่งของขนาดอินพุต

3. $L_2 - norm$ pooling

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำพูลลิ่งด้วย $L_2 - norm$ นั้น จะคำนวณหาค่า $L_2 - norm$ ของแต่ละบล็อคที่ตัวกรองครอบข้อมูลอินพุตอยู่เพื่อเป็นตัวแทนข้อมูลแต่ละบล็อคนั้น ๆ แสดงดังรูปที่ 2.8



รูปที่ 2.8 $L_2 - norm$ pooling

จากรูปที่ 2.8 การทำพูลลิ่งด้วยค่าสูงสุดระหว่างอินพุตขนาด 4x4 กับตัวกรองขนาด 2x2 และมีการเลื่อนตัวกรองไปครั้งละ 2 ช่อง เอาต์พุตที่ได้คือตัวแทนที่มีค่า $L_2 - norm$ ของแต่ละบล็อค และมีขนาดลดลงครึ่งหนึ่งของขนาดอินพุต

วิธีการทำพูลลิ่งทั้ง 3 รูปแบบกับอินพุตรูปแบบเดียวกัน ค่าเอาต์พุตแต่ละบล็อคของการทำพูลลิ่งด้วยค่าเฉลี่ยและการทำพูลลิ่งด้วย $L_2 - norm$ ให้ค่าข้อมูลที่ใกล้เคียงกัน แต่การทำพูลลิ่งด้วยค่าสูงสุดให้ค่าเอาต์พุตที่แตกต่างจากการทำพูลลิ่งใน 2 รูปแบบแรก ทำให้การทำพูลลิ่งด้วยค่าสูงสุดเป็นที่นิยมใช้งาน เนื่องจากได้ค่าที่แตกต่างมากกว่าวิธีการทำพูลลิ่งแบบอื่น ซึ่งส่งผลต่อลักษณะข้อมูลที่ได้ และมีขั้นตอนทำที่ง่ายกว่า

3) ชั้นลักษณะเด่น (Fully Connected layer)

เมื่อเซลล์ประสาทที่ผ่านการทำงานจากชั้นก่อนหน้า จะเข้าสู่การเรียนรู้เพื่อจำแนกต่อไปนั้น จะต้องเตรียมข้อมูลให้เหมาะสมก่อนที่จะนำไปใช้งาน ชั้นลักษณะเด่นจะทำการเชื่อมโยงเซลล์ประสาทที่ผ่านการเรียนรู้จากชั้นคอนโวลูชัน และชั้นพูลลิ่งมาเชื่อมโยงกันทุกเซลล์ประสาทโดยเชื่อมต่อกันเป็น 1 มิติ ซึ่งชั้นนี้จะเชื่อมต่อไปยังชั้นเอาต์พุต

4) ชั้นเอาต์พุต (Output layer)

การเรียนรู้เพื่อจำแนกข้อมูลจะเกิดขึ้นในชั้นนี้ จำนวนเซลล์ประสาทที่ได้จะเท่ากับจำนวนคลาสที่ใช้ในการเรียนรู้ สามารถคำนวณได้จากสมการที่ 2.14

$$y_i = \frac{\exp\left(y_i^{'}\right)}{\sum_{j=1}^{n} \exp\left(y_j^{'}\right)} \tag{2.14}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ $y_j^{'} = f\sum_{i=1}^{m}(x_i \cdot w_{i,j} + b_j)$ โดยที่ $w_{i,j}$ คือ ตัวกรองที่ใช้ในการคอนโวลูชัน

x_i คือ อินพุต

b_j คือ ค่าไบแอส

f คือ ฟังก์ชันกระตุ้น

n เป็นจำนวนคลาสที่ต้องการจำแนก

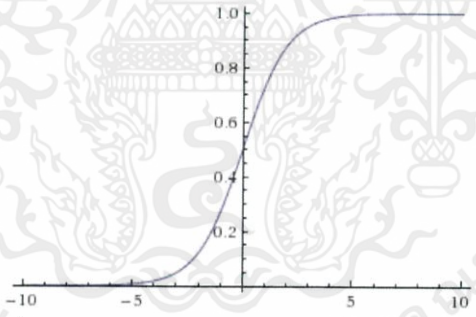
โดยฟังก์ชันกระตุ้นที่นิยมใช้มีอยู่ 3 ชนิดได้แก่

1. Sigmoid function

ฟังก์ชัน Sigmoid เป็นฟังก์ชันทางคณิตศาสตร์แบบไม่เชิงเส้น ที่เปลี่ยนค่าทั้งหมดบนเส้นจำนวนให้อยู่ในช่วง (0,1) แสดงดังสมการที่ 2.15

$$f(x) = \frac{1}{1+e^{-x}} \tag{2.15}$$

ผลการทำงานของฟังก์ชัน Sigmoid แสดงดังรูปที่ 2.9



รูปที่ 2.9 ผลการทำงานของฟังก์ชัน Sigmoid

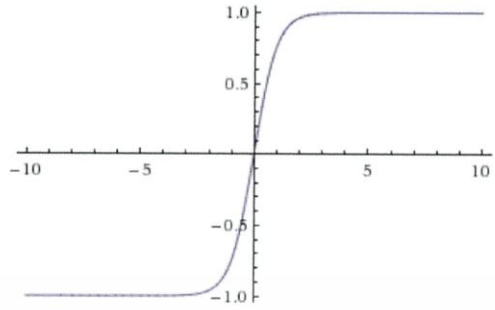
2. Tan hyperbolic function (tanh function)

ฟังก์ชัน tanh เป็นฟังก์ชันทางคณิตศาสตร์แบบไม่เชิงเส้น ที่เปลี่ยนค่าทั้งหมดบนเส้นจำนวนให้อยู่ในช่วง (-1,1) แสดงดังสมการที่ 2.16

$$f(x) = \tanh x = \frac{\sinh x}{\cosh x} = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{2.16}$$

ผลการทำงานของฟังก์ชัน tanh แสดงดังรูปที่ 2.10

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



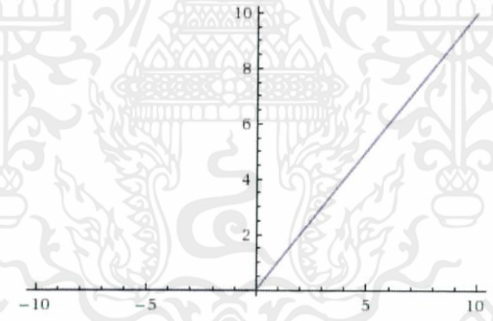
รูปที่ 2.10 ผลการทำงานของฟังก์ชัน tanh

3. Rectified Linear Unit (ReLU function)

ฟังก์ชัน ReLU เป็นฟังก์ชันที่ได้รับความนิยมและถูกนำมาใช้งานกับ CNN มากกว่าฟังก์ชันอื่น เนื่องจากลดโอกาสที่บางข้อมูลจะสูญหาย โดยหากข้อมูลมีค่ามากกว่า 0 จะเลือกมาเป็นเอาต์พุต แต่หากข้อมูลมีค่าน้อยกว่า 0 จะให้ค่าเอาต์พุตเป็น 0 แสดงดังสมการที่ 2.17

$$f(x) = \max(0, x) \tag{2.17}$$

ผลการทำงานของฟังก์ชัน ReLU แสดงดังรูปที่ 2.11



รูปที่ 2.11 ผลการทำงานของฟังก์ชัน ReLU

เนื่องจากโครงข่ายประสาทเทียมแบบคอนโวลูชัน (CNN) เป็นโครงข่ายประสาทเทียมที่ใช้กระบวนการเรียนรู้แบบแพร่กลับ (Back Propagation) ซึ่งมีจุดเด่นคือ การหาค่าความผิดพลาดย้อนกลับเพื่อปรับค่าน้ำหนักในชั้นก่อนหน้าให้ดีกว่าเดิม การหาค่า gradient ของเซลล์ประสาท k แสดงดังสมการที่ 2.18

$$\delta_k(p) = y_k(p) \times [1 - y_k(p)] \times e_k(p) \tag{2.18}$$

โดยที่ $\delta_k(p)$ คือ ค่า gradient ของเซลล์ประสาท k

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$y_k(p)$ คือ ข้อมูลอินพุตของเซลล์ประสาท k

$e_k(p)$ คือ ค่าความผิดพลาดที่เซลล์ประสาท k

จากนั้นนำค่า gradient ที่ได้คำนวณน้ำหนักที่เปลี่ยนแปลงไป ดังสมการที่ 2.19

$$\Delta w_{kj}(p) = \alpha \times y_k(p) \times \delta_k(p) \tag{2.19}$$

โดยที่ $\Delta w_{kl}(p)$ คือ ค่าน้ำหนักที่เปลี่ยนแปลงของชั้นเอาต์พุต

α คือ อัตราการเรียนรู้

จากนั้นคำนวณหาค่า gradient ของเซลล์ประสาท j ดังสมการที่ 2.20

$$\delta_j(p) = y_j(p) \times \left[1 - y_j(p)\right] \times \sum_{k=1}^{l} \delta_k(p) \times w_{kj}(p) \tag{2.20}$$

ขั้นตอนการเรียนรู้เป็นการเรียนรู้แบบคำนวณซ้ำ ในแต่ละรอบของการเรียนรู้จะปรับค่าน้ำหนักดังสมการที่ 2.21

$$w_{kj}(p+1) = w_{kj}(p) + w_{kj} \tag{2.21}$$

โดยที่ $w_{kj}(p)$ คือ ค่าน้ำหนักจากเซลล์ประสาท j ถึงเซลล์ประสาท k ในชั้นเอาต์พุต รอบการคำนวณที่ p

$w_{kj}(p+1)$ คือ ค่าน้ำหนักจากเซลล์ประสาท j ถึงเซลล์ประสาท k ในชั้นเอาต์พุต รอบการคำนวณที่ $p+1$

เพิ่มค่า p ขึ้นมาและคำนวณซ้ำ ๆ ไปจนกว่าค่า Sum of Squared Errors จะน้อยกว่าที่กำหนด จากนั้นระบบจะหยุดการทำงาน ผลลัพธ์สุดท้ายจะได้ค่าน้ำหนัก ซึ่งนำไปใช้ในกระบวนการทดสอบต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและการจัดทำปริญญานิพนธ์

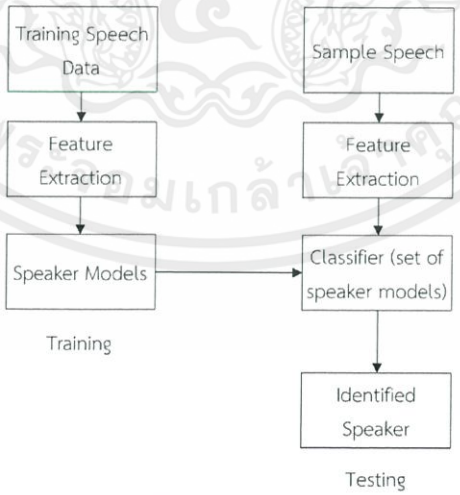
ระบบระบุตัวตนด้วยปัญญาประดิษฐ์ ถูกออกแบบการทำงานของระบบออกเป็น 2 ส่วนหลักคือ สำหรับระบุตัวตนด้วยเสียง และสำหรับระบุตัวตนด้วยใบหน้า ซึ่งการออกแบบและการจัดทำระบบจะกล่าวในบทนี้

3.1 ระบบระบุตัวตนด้วยปัญญาประดิษฐ์ สำหรับระบุตัวตนด้วยเสียง

การทำงานของระบบระบุตัวตนด้วยปัญญาประดิษฐ์ สำหรับระบุตัวตนด้วยเสียงนั้น เริ่มต้นด้วยการบันทึกสัญญาณเสียงผู้พูด ทำการดึงลักษณะเด่นของสัญญาณเสียงนั้น ๆ เพื่อเป็นตัวแทนของสัญญาณเสียง จากนั้นนำลักษณะเด่นมาจำแนกเพื่อสร้างรูปแบบการจัดหมวดหมู่ในระบบ

3.1.1 การออกแบบระบบระบุตัวตนด้วยปัญญาประดิษฐ์ สำหรับระบุตัวตนด้วยเสียง

การออกแบบระบบระบุตัวตนด้วยปัญญาประดิษฐ์ สำหรับระบุตัวตนด้วยเสียงมีโครงสร้างพื้นฐานมาจากเว็บบล็อกในหัวข้อ “Spoken Speaker Identification based on Gaussian Mixture Models : Python Implementation”ของ Abhijeet Kumar [1] เนื่องจากระบบที่ใช้อ้างอิงนี้ให้ค่าความถูกต้องสูง ซึ่งกระบวนการทำงานของระบบระบุตัวตนด้วยเสียงนั้นแสดงดังรูปที่ 3.1



รูปที่ 3.1 กระบวนการทำงานของระบบระบุตัวตนด้วยเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.1 กระบวนการทำงานของระบบระบุตัวตนด้วยเสียงแบ่งออกเป็น 2 กระบวนการคือ การสร้างโมเดลจำแนกเสียงบุคคลและการทดสอบโมเดลจำแนกเสียงบุคคล สามารถอธิบายได้ดังนี้

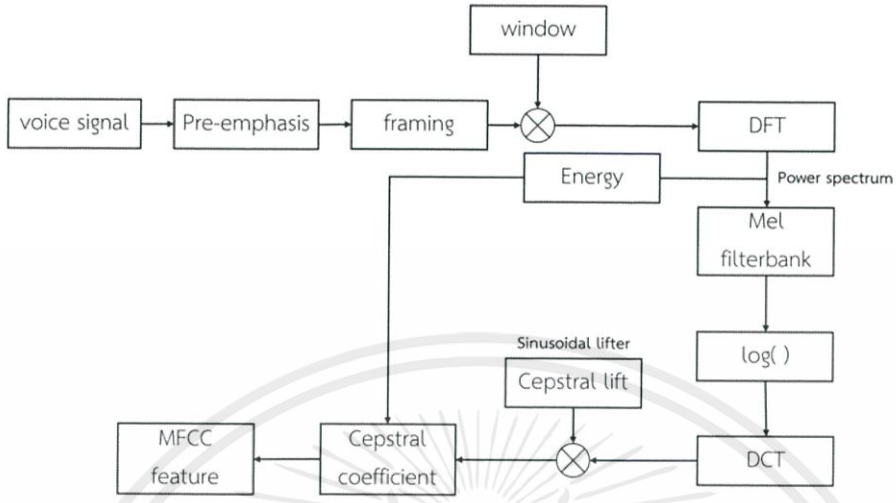
การสร้างโมเดลจำแนกเสียงบุคคล เริ่มจากการดึงลักษณะเด่นของเสียงแต่ละบุคคลด้วยวิธีหาค่าสัมประสิทธิ์เซพสตรัลบนสเกลเมล (Mel Frequency Cepstral Coefficient) หรือ MFCC จากนั้นนำลักษณะเด่นที่ได้เข้าสู่กระบวนการจำแนกบุคคลด้วยวิธีการหารูปแบบจำลองผสมของเกาส์ (Gaussian Mixture Model) หรือ GMM เพื่อหารูปแบบของแบบจำลอง ผลลัพธ์ที่ได้จะเป็นรูปแบบจำลองผสมของเสียงแต่ละบุคคล และถูกจัดเก็บเป็นชุดข้อมูลสำหรับการจำแนกบุคคลด้วยเสียง

การทดสอบโมเดลจำแนกเสียงบุคคล เริ่มจากสัญญาณเสียงอินพุตจะผ่านกระบวนการดึงลักษณะเด่นซึ่งใช้วิธีเดียวกันกับการสร้างโมเดลจำแนกเสียงบุคคล จากนั้นนำลักษณะเด่นที่ได้มาหาค่าความน่าจะเป็นจากชุดข้อมูลสำหรับการจำแนกบุคคลด้วยเสียง ระบบจะทำนายว่าเสียงอินพุตเป็นใคร จากค่าความน่าจะเป็นมากที่สุดของรูปแบบจำลองผสมของเสียงบุคคล

3.1.2 การออกแบบการดึงลักษณะเด่น สำหรับระบุตัวตนด้วยเสียง

การออกแบบการดึงลักษณะเด่น สำหรับระบุตัวตนด้วยเสียงที่ใช้ปริญญานิพนธ์นี้แสดงดังรูปที่ 3.2

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

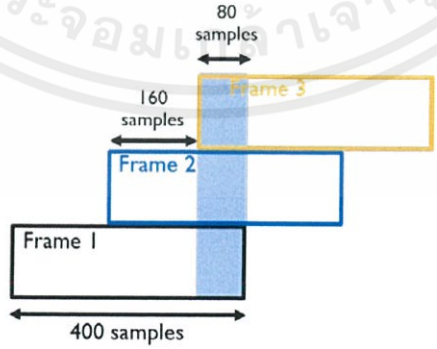


รูปที่ 3.2 กระบวนการดึงลักษณะเด่นของเสียง

1) สัญญาณเสียงอินพุตจะถูกบันทึกด้วยไมโครโฟนที่ Sampling ด้วยความถี่ 48 kHz ขนาด 16 bits / sample และถูกจัดเก็บเป็นไฟล์นามสกุล .wav

2) นำสัญญาณไปผ่านตัวกรองความถี่สูงเพื่อเน้นการองค์ประกอบความถี่สูงของสัญญาณเสียง (Pre-emphasis) ก่อนนำสัญญาณไปประมวลผล

3) สัญญาณที่ผ่านการทำพรีเอมฟาซิส จะถูกนำมาแบ่งเป็นเฟรม ๆ โดยแต่ละเฟรมมีความยาว 400 แซมเปิล แต่ละเฟรมจะมีระยะห่างระหว่างเฟรมเป็น 160 แซมเปิล และทุก ๆ เฟรมจะซ้อนทับเป็นความยาว 80 แซมเปิล แสดงการแบ่งเฟรมของสัญญาณเสียง ดังรูปที่ 3.3



รูปที่ 3.3 การแบ่งเฟรมของสัญญาณเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) จากนั้นนำเฟรมไปคูณกับฟังก์ชันวินโดว์ ซึ่งในปริญญานิพนธ์นี้เลือกใช้ฟังก์ชันวินโดว์ที่มีขนาดเป็น 1 ตลอดช่วงความถี่ ซึ่งความยาวของวินโดว์จะมีขนาดเท่ากับ 400 แซมเปิล เมื่อเฟรมผ่านการคูณฟังก์ชันวินโดว์จะเติมศูนย์ต่อท้ายเฟรม เพื่อให้เฟรมมีความยาวเป็น 512 เท่ากับความยาวของ DFT ก่อนนำเฟรมไปแปลง DFT ในขั้นตอนต่อไป

5) นำเฟรมที่ผ่านการคูณวินโดว์มาวิเคราะห์โดยใช้สเปกตรัมเชิงกำลังของสัญญาณเสียงในโดเมนความถี่ซึ่งจะวิเคราะห์ทีละเฟรม ขั้นตอนแรกทำการแปลงดิสครีตฟูเรียร์ทรานฟอร์ม (Discrete Fourier Transform: DFT) และขั้นตอนที่สองทำการหาค่าสเปกตรัมเชิงกำลังของสัญญาณเสียง (Power Spectrum) ซึ่งในขั้นตอนนี้แต่ละเฟรมจะมีความยาวเป็น 257 แซมเปิล

6) ทำการหาชุดตัวกรองความถี่บนสเกลเมล มีขั้นตอนดังนี้

1. ทำการเปลี่ยนความถี่ต่ำสุด (0 Hz) และความถี่สูงสุด (8k Hz) บนสเกลทางความถี่เฮิร์ตซ์ให้เป็นความถี่สเกลเมล ตามสมการที่ 2.4 จะได้ความถี่ต่ำสุดบนสเกลเมลเป็น 0 และความถี่สูงสุดบนสเกลเมลเป็น 6359.4

2. ในปริญญานิพนธ์นี้จะสร้างตัวกรองทั้งหมด 26 ตัว โดยที่ตัวกรองทั้ง 26 ตัวจะอยู่ระหว่างระยะห่างเชิงเส้นบนสเกลเมลของความถี่ต่ำสุดและความถี่สูงสุด ซึ่งจะมีระยะห่างเชิงเส้นบนสเกลเมลทั้งหมด 28 จุด แสดงดังรูปที่ 3.4

```
f =

   1.0e+03 *

  Columns 1 through 8

         0    0.2422    0.4844    0.7266    0.9688    1.2110    1.4532    1.6954

  Columns 9 through 16

    1.9376    2.1798    2.4220    2.6642    2.9064    3.1486    3.3908    3.6330

  Columns 17 through 24

    3.8752    4.1174    4.3596    4.6018    4.8440    5.0862    5.3284    5.5706

  Columns 25 through 28

    5.8128    6.0550    6.2972    6.5394
```

รูปที่ 3.4 ระยะห่างเชิงเส้นบนสเกลเมล

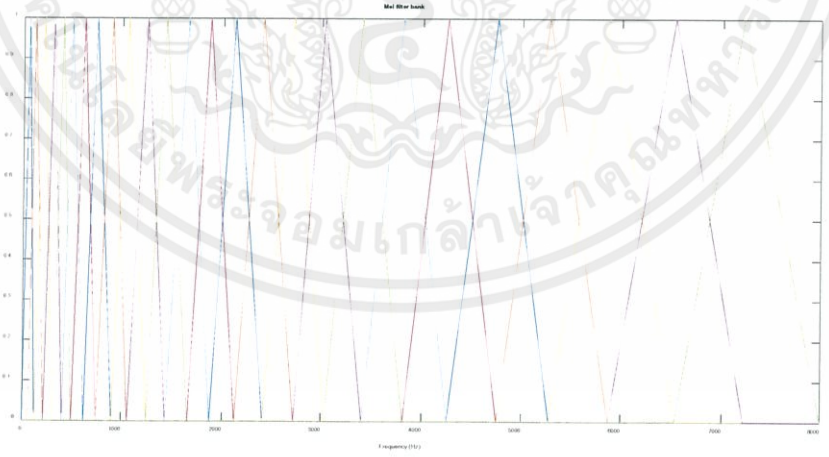
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ทำการแปลงความถี่บนสเกลเมลกลับเป็นความถี่เฮิร์ตซ์ ตามสมการที่ 2.5 แสดงดังรูปที่ 3.5

```
fm =

   1.0e+03 *

  Columns 1 through 8

         0    0.0685    0.1437    0.2262    0.3168    0.4163    0.5255    0.6454

  Columns 9 through 16

    0.7770    0.9215    1.0801    1.2542    1.4454    1.6553    1.8857    2.1386

  Columns 17 through 24

    2.4163    2.7212    3.0559    3.4233    3.8267    4.2695    4.7557    5.2894

  Columns 25 through 28

    5.8753    6.5186    7.2247    8.0000
```

รูปที่ 3.5 ระยะห่างเชิงเส้นบนสเกลเฮิร์ตซ์

4. สร้างชุดตัวกรองความถี่ โดยตัวกรองตัวแรกจะเริ่มที่จุดแรกของกลุ่มความถี่จากขั้นตอนที่ 3 ไปถึงจุดสูงสุดที่จุดที่สอง จากนั้นกลับไปสู่ศูนย์ที่จุดที่สาม และตัวกรองที่สองจะเริ่มที่จุดที่สองไปถึงจุดสูงสุดที่จุดที่สาม จากนั้นกลับไปสู่ศูนย์ที่จุดที่สี่ เป็นต้น โดยสร้างตามสมการที่ 2.6 ซึ่งชุดตัวกรองความถี่ทั้ง 26 ตัว แสดงดังรูปที่ 3.6



รูปที่ 3.6 ชุดตัวกรองความถี่บนสเกลทั้ง 26 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7) นำสเปกตรัมเชิงกำลังของแต่ละเฟรมไปคูณชุดตัวกรองความถี่ตามสมการที่ 2.7 ในที่นี่มีตัวกรองทั้งหมด 26 ตัว เมื่อดำเนินการเสร็จในแต่ละเฟรมจะได้ตัวเลขทั้งหมด 26 ตัว ที่แสดงถึงพลังงานของตัวกรองแต่ละตัว

8) นำผลลัพธ์ของแต่ละเฟรมจากการคูณชุดตัวกรองความ ไปผ่านฟังก์ชันลอการิทึมตามสมการที่ 2.8

9) นำผลลัพธ์จากขั้นตอนที่ 8 มาหาค่าสัมประสิทธิ์เซพสตรัลบนสเกลเมลด้วยการแปลงโคซายน์ไม่ต่อเนื่อง (Discrete Cosine Transform: DCT) ตามสมการที่ 2.9 จะได้ค่าสัมประสิทธิ์เซพสตรัลทั้งหมด 26 ตัว

10) ยกระดับค่าสัมประสิทธิ์เซพสตรัลด้วยวิธีการนำค่าสัมประสิทธิ์เซพสตรัลคูณกับน้ำหนัก ในปริญญานิพนธ์นี้น้ำหนักได้มาจากการยกระดับค่าสัมประสิทธิ์ด้วยสัญญาณไซนูซอยดอล (Sinusoidal lifter) ซึ่งแสดงดังรูปที่ 3.7

รูปที่ 3.7 การยกระดับค่าสัมประสิทธิ์ด้วยสัญญาณไซนูซอยดอล (Sinusoidal lifter)

จากรูปที่ 3.7 การยกระดับค่าสัมประสิทธิ์ด้วยสัญญาณไซนูซอยดอล (Sinusoidal lifter) สามารถเพิ่มประสิทธิภาพค่าสัมประสิทธิ์เซพสตรัล[8] เพื่อให้ระบบจำแนกเสียงของแต่ละบุคคลได้ดียิ่งขึ้น โดยวิธีการยกระดับค่าสัมประสิทธิ์เซพสตรัลด้วยสัญญาณไซนูซอยดอลเป็นดังสมการที่ 3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$weights = 1 + \frac{L}{2}\sin\left(\frac{\pi n}{L}\right) \quad (3.1)$$

โดยที่ n คือ จำนวนค่าสัมประสิทธิ์เซพสตรัม

L คือ จำนวนค่าสัมประสิทธิ์ของการยกระดับ ในที่นี่กำหนดจำนวนเป็น 22 ค่า

11) คำนวณพลังงานทั้งหมดในแต่ละเฟรม จากนั้นนำค่าพลังงานทั้งหมดไปผ่านฟังก์ชันลอการิทึม และนำผลลัพธ์ไปแทนที่ค่าประสัมประสิทธิ์เซพสตรัลตัวแรกของแต่ละเฟรม

12) คำนวณหาค่าสัมประสิทธิ์เดลต้า (Delta Coefficients) เพื่อเพิ่มประสิทธิภาพให้ลักษณะเด่นซึ่งเป็นตัวแทนของสัญญาณเสียง หรือสร้างคุณสมบัติเด่นที่ดีของลักษณะเด่นให้มีความแตกต่างกัน เพื่อส่งผลต่อระบบรู้จำให้ประสิทธิภาพดีขึ้น ซึ่งหาได้จากสมการที่ 2.10

13) สุดท้ายในแต่ละเฟรมจะได้ลักษณะเด่นทั้งหมด 40 ลักษณะเด่น โดยลักษณะเด่น 20 ตัวแรกมาจากการหาค่าสัมประสิทธิ์เซพสตรัลบนสเกลเมล และลักษณะเด่นอีก 20 ตัวมาจากการหาค่าสัมประสิทธิ์เดลต้า

3.1.3 การออกแบบการจำแนกบุคคลด้วยเสียง

ค่าสัมประสิทธิ์เซพสตรัลบนสเกลเมลเหล่านี้จะถูกนำไปเรียนรู้ฝึกฝนเพื่อสร้างรูปแบบสำหรับจำแนกผู้พูดด้วยเสียงด้วยแบบจำลองผสมแบบเกาส์ ซึ่งเป็นการหาความน่าจะเป็นของแต่ละสัญญาณเสียงเพื่อเป็นตัวแทนของการแสดงตัวของประชากรย่อยในกลุ่มประชากรรวม ซึ่งตัวแทนกลุ่มเหล่านี้จะถูกบันทึกให้อยู่ในรูปแบบไฟล์ .gmm สำหรับใช้ในการทดสอบสัญญาณเสียงอื่น ๆ ต่อไป

3.1.4 การหาค่าขีดแบ่งสำหรับตัดสินใจ

การหาค่าขีดแบ่งสำหรับตัดสินใจว่าไฟล์เสียงอินพุตนั้นเป็นใคร และเป็นบุคคลในฐานข้อมูลหรือไม่ โดยหาค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐานของค่า Z_1 และค่า Z_2 จากกราฟฮิสโทแกรมซึ่งค่า Z_1 คือ จำนวนความน่าจะเป็นที่ไม่รวมค่าความน่าจะเป็นที่มากที่สุด และค่า Z_2 คือ ความน่าจะเป็นที่มากที่สุด แล้วนำมาคำนวณค่าขีดแบ่ง ดังสมการที่ 3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\text{THRESHOLD} = \mu_1 + \left(\frac{\sigma_1 \mu_2 - \sigma_2 \mu_1}{\sigma_1 + \sigma_2} \right) \quad (3.2)$$

โดยที่ μ_1 คือ ค่าเฉลี่ยของ Z_1

σ_1 คือ ค่าเบี่ยงเบนมาตรฐานของ Z_1

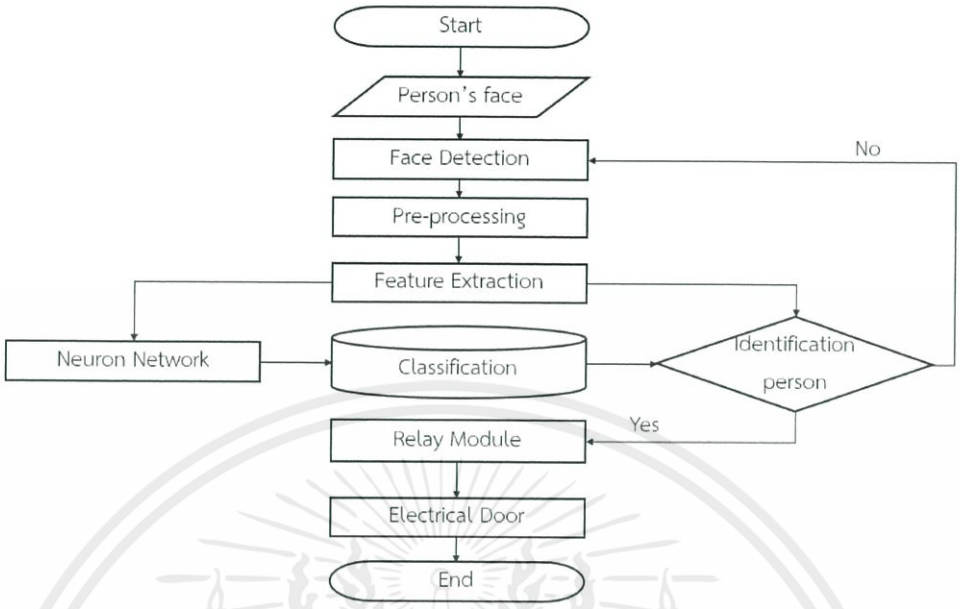
μ_2 คือ ค่าเฉลี่ยของ Z_2

σ_2 คือ ค่าเบี่ยงเบนมาตรฐานของ Z_2

3.2 ระบบระบุตัวตนด้วยปัญญาประดิษฐ์ สำหรับระบุตัวตนด้วยใบหน้า

การออกแบบระบบระบุตัวตนด้วยปัญญาประดิษฐ์ สำหรับระบุตัวตนด้วยใบหน้านั้น เริ่มต้นด้วยการตรวจจับใบหน้าบุคคล จากนั้นทำการเตรียมรูปภาพใบหน้าที่ตรวจจับได้ให้อยู่ในรูปแบบที่เหมาะสมต่อการใช้งาน และทำการดึงคุณลักษณะเด่นของรูปภาพ เพื่อเป็นตัวแทนของรูปภาพนั้น ๆ สำหรับการจำแนกรู้จำในการเรียนรู้ของโครงข่ายประสาทเทียมเพื่อสร้างโมเดลสำหรับจำแนกบุคคล การใช้งานระบบเริ่มต้นจากตรวจจับใบหน้าเมื่อมีคนยืนในตำแหน่งกล้องและบันทึกรูปภาพที่ถูกตรวจจับ จากนั้นเตรียมรูปภาพและดึงคุณลักษณะเด่นของรูปภาพใบหน้าที่ตรวจจับได้ ต่อมาทำนายผลลัพธ์ของรูปภาพอินพุตด้วยโมเดล ถ้าบุคคลที่ตรวจจับใบหน้าได้เป็นบุคคลที่อยู่ในหมวดหมู่ของระบบ ระบบจะสั่งการให้หยุดกระแสไฟที่จ่ายไปยังรีเลย์โมดูล เพื่อให้บุคคลนั้นเข้าไปยังพื้นที่ส่วนนั้นได้ แต่ถ้าไม่สามารถระบุได้ว่าบุคคลนั้นเป็นใคร ระบบจะกลับไปเริ่มต้นตรวจจับใบหน้าอีกครั้ง สามารถแสดงกระบวนการทำงานทั้งหมดในรูปโฟวชาร์ต ดังรูปที่ 3.8

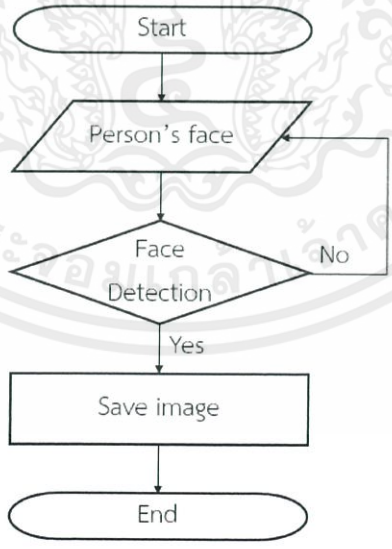
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 โฟวชาร์ตการทำงานของระบบระบุตัวตนด้วยใบหน้า

3.2.1 การตรวจจับและบันทึกใบหน้าบุคคล

การตรวจจับและบันทึกใบหน้าบุคคล เป็นกระบวนการที่คอยตรวจจับใบหน้าเมื่อมีคนยืนบริเวณตำแหน่งกล้อง และบันทึกรูปภาพ สามารถเขียนวิธีการตรวจจับและบันทึกใบหน้าบุคคลในรูปโฟวชาร์ต ดังรูปที่ 3.9



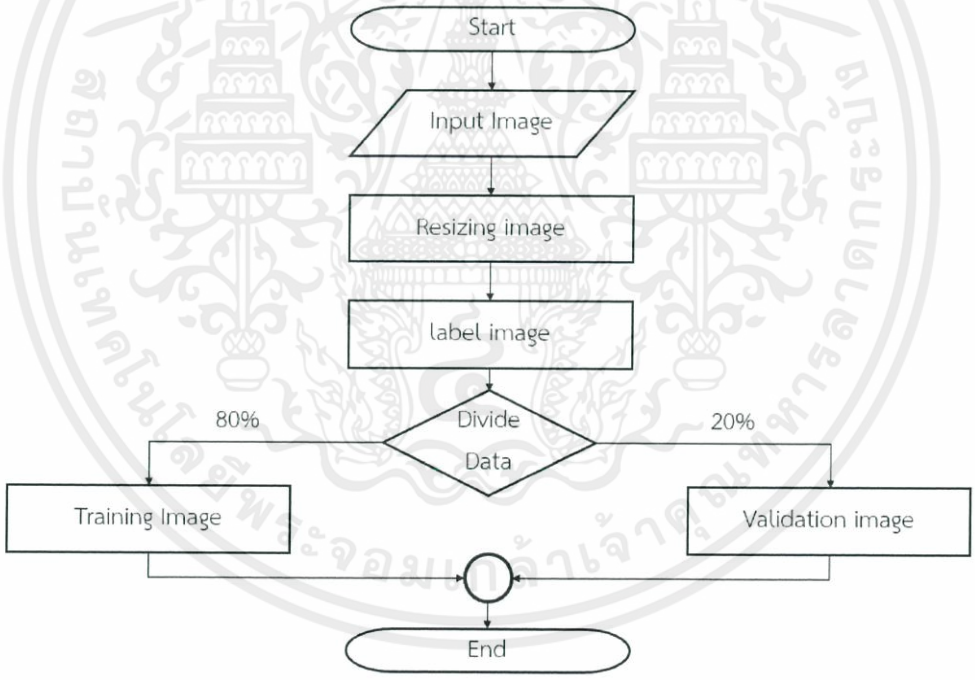
รูปที่ 3.9 โฟวชาร์ตของกระบวนการตรวจจับและบันทึกรูปภาพใบหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กระบวนการทำงานของขั้นตอนนี้คือ เมื่อมีคนยืนบริเวณตำแหน่งกล้อง การตรวจจับใบหน้าจะเริ่มค้นหาใบหน้าบุคคลด้วยโมเดล “haarcascade_frontalface_default.xml” ถ้าโมเดลตรวจพบใบหน้า กระบวนการทำงานจะเริ่มบันทึกรูปภาพและบันทึกเป็นไฟล์นามสกุล .jpg (Jpeg Format) โดยรูปภาพสำหรับฝึกสอนเพื่อสร้างโมเดลจำแนกบุคคล จะใช้รูปภาพใบหน้าจากบุคคลทั้งหมด 10 คน คนละ 55 รูป

3.2.2 การเตรียมรูปภาพใบหน้าบุคคล

ทำการปรับขนาดของรูปภาพให้มีขนาดข้อมูล 100x100 พิกเซล จากนั้นกำหนดป้ายชื่อให้รูปภาพทุก ๆ รูป ตามคลาสของรูปนั้น ๆ และกำหนดค่าอัตราส่วนเท่ากับ 0.8 หมายถึงรูปภาพทั้ง 55 รูปจะถูกแบ่งสำหรับฝึกสอนเพื่อสร้างโมเดลคิดเป็นร้อยละ 80 ของรูปทั้งหมด และอีกร้อยละ 20 จะใช้สำหรับตรวจสอบคุณภาพของโมเดล สามารถแสดงกระบวนการเตรียมรูปภาพใบหน้าบุคคลในรูปโฟวชาร์ต ดังรูปที่ 3.10



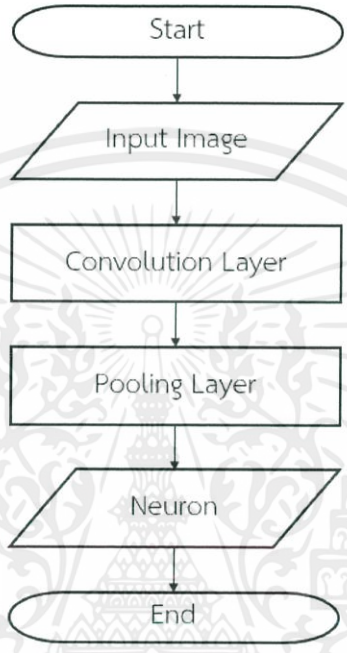
รูปที่ 3.10 โฟวชาร์ตของกระบวนการเตรียมรูปภาพใบหน้าบุคคล

3.2.3 การดึงลักษณะเด่นของรูปภาพใบหน้าบุคคล

การดึงคุณลักษณะเด่นของรูปภาพใบหน้าบุคคล เป็นกระบวนการหาตัวแทนของรูปภาพที่มีจำนวนข้อมูลมาก ให้อยู่ในรูปแบบข้อมูลแบบใหม่ที่มีปริมาณข้อมูลน้อยลง สำหรับการ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

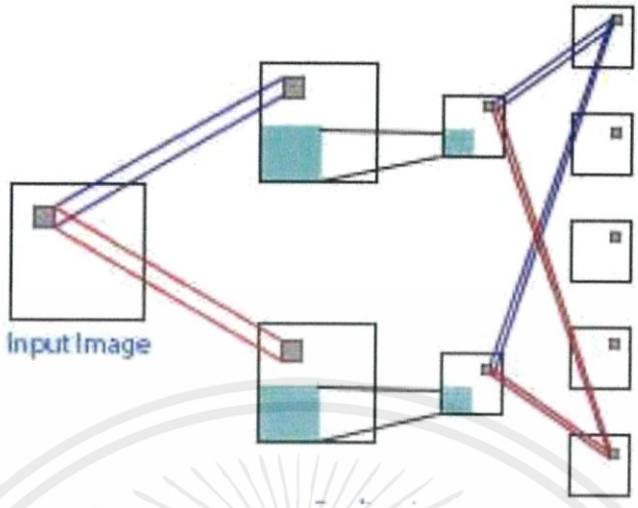
จำแนกรู้จำของระบบ ซึ่งใบหน้าของแต่ละบุคคลคนย่อมมีความแตกต่างกัน การหาตัวแทนของรูปภาพใบหน้าของบุคคลจะช่วยทำให้การจำแนกข้อมูลทำได้ง่ายมากขึ้น อีกทั้งยังช่วยประหยัดเวลาในการประมวลผล สามารถเขียนวิธีการดึงคุณลักษณะเด่นของรูปภาพใบหน้าบุคคลในรูปโฟวชาร์ตดังรูปที่ 3.11



รูปที่ 3.11 โฟวชาร์ตของกระบวนการดึงคุณลักษณะเด่นของรูปภาพใบหน้าบุคคล

การทำงานของชั้นคอนโวลูชั่นคือ จะใช้ตัวกรองที่มีขนาดข้อมูลเท่ากันเลื่อนเป็นจังหวะไปทั่วรูปภาพ เพื่อดึงคุณลักษณะเด่นของรูปภาพ จำนวนข้อมูลที่ได้จะเท่ากับจำนวนตัวกรองที่ใช้ในแต่ละชั้น และชั้นพลูลิ่งจะทำการลดขนาดข้อมูลที่ได้จากชั้นคอนโวลูชั่น เพื่อให้ระบบทำงานได้เร็วขึ้น โดยออกแบบให้ชั้นคอนโวลูชั่นมี 4 ชั้นและชั้นพลูลิ่งมี 4 ชั้นเช่นกัน แสดงการทำงานของการดึงคุณลักษณะของรูปภาพนี้ดังรูปที่ 3.12

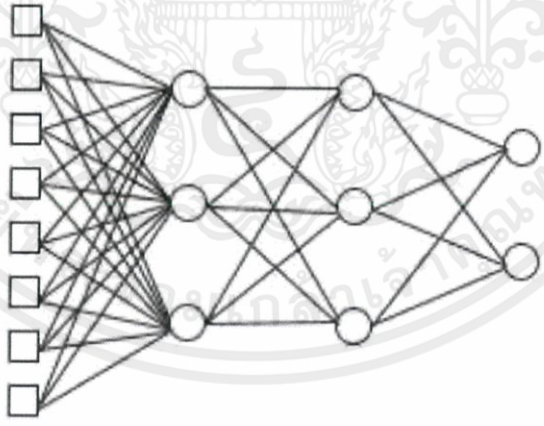
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 การดึงคุณลักษณะของรูปภาพ

3.2.4 การออกแบบระบบรู้จำบุคคลด้วยใบหน้า

เมื่อรูปภาพอินพุตทำการดึงคุณลักษณะเด่นของรูปภาพแล้ว จะได้นิวรอน (Neuron) ซึ่งเป็นข้อมูลที่ใช้ในการเรียนรู้แบบโครงข่ายประสาทเทียม โดยออกแบบให้ชั้นนี้มีทั้งหมด 2 ชั้น ผลลัพธ์จากการเรียนรู้แบบโครงข่ายประสาทเทียมจะได้โมเดลสำหรับจำแนกบุคคล สำหรับใช้ในการทดสอบรูปภาพใบหน้าอื่น ๆ ต่อไป ซึ่งการทำงานของระบบรู้จำแสดง ดังรูปที่ 3.13



รูปที่ 3.13 การเรียนรู้แบบโครงข่ายประสาทเทียม

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 เครื่องมือที่ใช้ในการทดลอง

ปริญญานิพนธ์นี้ มีอุปกรณ์และเครื่องมือที่ใช้ดังต่อไปนี้

3.3.1 Raspberry Pi Camera Module

การรับรูปภาพจากผู้ใช้งานจำเป็นต้องใช้กล้องที่สามารถเก็บรายละเอียดของภาพได้ครบถ้วน โดยที่ภาพนั้นไม่มีความเสียหาย เช่น การเบลอของรูปภาพ เป็นต้น และอีกทั้งยังต้องการกล้องที่มีขนาดเล็ก สามารถพกพาและเคลื่อนย้ายได้อย่างสะดวกอย่าง Raspberry Pi 3 Camera Module 5MP ดังรูปที่ 3.14 มีคุณสมบัติดังนี้

รูปที่ 3.14 Raspberry Pi Camera Module

- ความละเอียด 5 ล้านพิกเซล
- เซนเซอร์ชนิด Omnivision OV5647 Color CMOS QSXGA (5-megapixel)

ข้อเด่น คือ ความคมชัดของกล้องจะมากขึ้น เมื่อถ่ายในระยะห่างประมาณ 1.5 เมตรขึ้นไป และมีความคมชัดไม่จำกัด (infinity Focus) จึงทำให้ภาพคมชัดมาก

3.3.2 Kinobo – USB mini microphone

การเก็บเสียงที่ใช้เป็นข้อมูลที่ใช้สอน จำเป็นต้องใช้ไมโครโฟนที่มีคุณภาพสูง และสามารถลดเสียงรบกวนที่อาจจะทำให้เสียงมีคุณภาพต่ำ นอกจากนั้นยังต้องการไมโครโฟนที่สามารถรับเสียงได้รอบทิศทาง เพื่อเพิ่มความสะดวกให้กับผู้ใช้งาน และมีขนาดเล็ก พกพาง่าย เหมาะสำหรับการติดตั้งและขนย้ายสะดวก ปริญญานิพนธ์นี้จึงเลือกใช้ Kinobo – USB mini microphone ดังรูปที่ 3.15 โดยคุณสมบัติของ Kinobo – USB mini microphone มีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.15 Kinobo – USB mini microphone

- เป็นไมโครโฟนขนาดเล็ก
- ทำงานโดยไม่ต้องใช้ไดรเวอร์
- มีประสิทธิภาพในการลดเสียงรบกวน
- สามารถรับเสียงได้รอบทิศทาง

3.3 การจัดเก็บผลการทดลอง

การจัดเก็บผลการทดลองเพื่อนำผลลัพธ์ที่ได้ไปวัดประสิทธิภาพของโมเดล โดยใช้ตารางคอนฟูชัน เมททริก (Confusion Matrix) สามารถออกแบบตาราง ดังตารางที่ 3.1

ตารางที่ 3.1 การออกแบบตารางคอนฟูชันแมททริก สำหรับวัดประสิทธิภาพโมเดล

Actual/Predicted	Yes	No
Yes	TP	FP
No	FN	TN

โดยที่ TP คือ จำนวนข้อมูลที่ทำนายถูกต้อง และเป็นคลาสที่กำลังสนใจ

TN คือ จำนวนข้อมูลที่ทำนายถูกต้อง และเป็นคลาสที่ไม่ได้สนใจ

FP คือ จำนวนข้อมูลที่ทำนายไม่ถูกต้อง และเป็นคลาสที่กำลังสนใจ

FN คือ จำนวนข้อมูลที่ทำนายไม่ถูกต้อง และเป็นคลาสที่ไม่ได้สนใจ

การวัดประสิทธิภาพโมเดลของ Data Mining จะมีตัววัดที่ใช้ในการวัดอยู่ 3 ค่า คือ

1) Precision คือ การวัดความแม่นยำของข้อมูล โดยพิจารณาแยกทีละคลาส สามารถคำนวณได้ ดังสมการที่ 3.2

$$\text{PRECISION} = \frac{TP}{TP + FP} \tag{3.3}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) Recall คือ การวัดความถูกต้องของโมเดล โดยพิจารณาแยกทีละคลาส สามารถคำนวณได้ ดังสมการที่ 3.3

$$\text{RECALL} = \frac{TP}{TP + FN} \tag{3.4}$$

3) Accuracy คือ การวัดความถูกต้องของโมเดล โดยพิจารณารวมทุกคลาส สามารถคำนวณได้ ดังสมการที่ 3.4

$$\text{ACCURACY} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.5}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

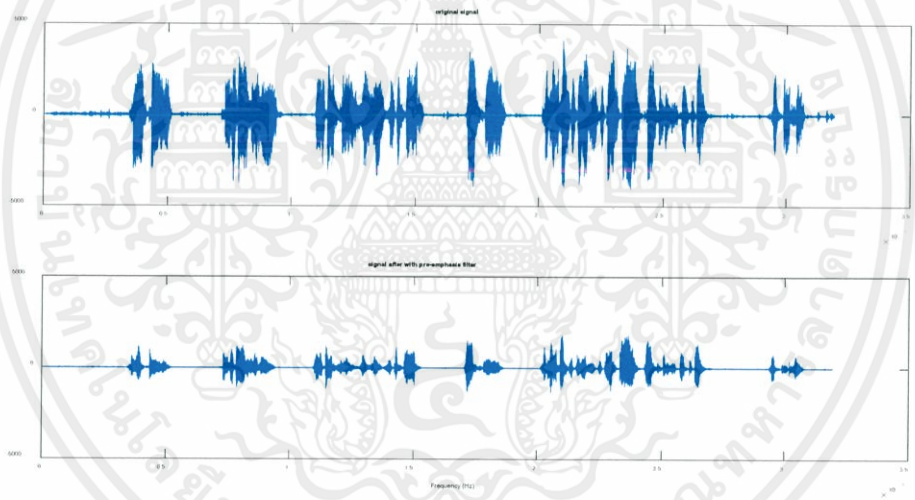
การทดลองจะแบ่งออกเป็น 2 ส่วนคือ ผลการทดลองระบบรู้จำเสียงผู้พูด และระบบรู้จำใบหน้าบุคคล

4.1 ระบบรู้จำเสียงบุคคล

4.1.1 การดึงคุณลักษณะเด่นของสัญญาณ

กระบวนการดึงลักษณะเด่นของสัญญาณเสียงมีผลลัพธ์ในแต่ละขั้นตอนดังต่อไปนี้

1) การทำพรีเอมฟาซิสให้กับสัญญาณเสียง ในปริญญานิพนธ์นี้เลือกใช้ค่าสัมประสิทธิ์ α มีค่าเท่ากับ 0.97 แสดงดังรูปที่ 4.1

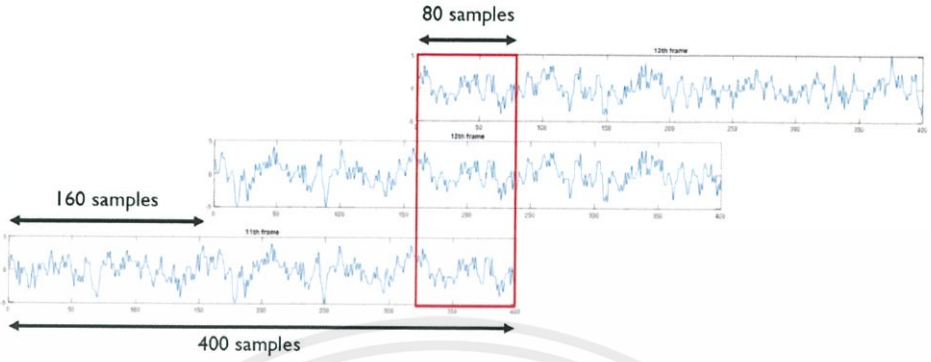


รูปที่ 4.1 การทำพรีเอมฟาซิส

จากรูปที่ 4.1 ภาพด้านบนเป็นสัญญาณเสียงก่อนการทำพรีเอมฟาซิส ซึ่งยังคงมีสัญญาณรบกวนอยู่มาก และภาพด้านล่างเป็นสัญญาณที่ผ่านการทำพรีเอมฟาซิส จะมีสัญญาณรบกวนลดน้อยลงทำให้คุณภาพของสัญญาณเสียงดีขึ้น ก่อนที่จะนำสัญญาณไปใช้ในขั้นตอนต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

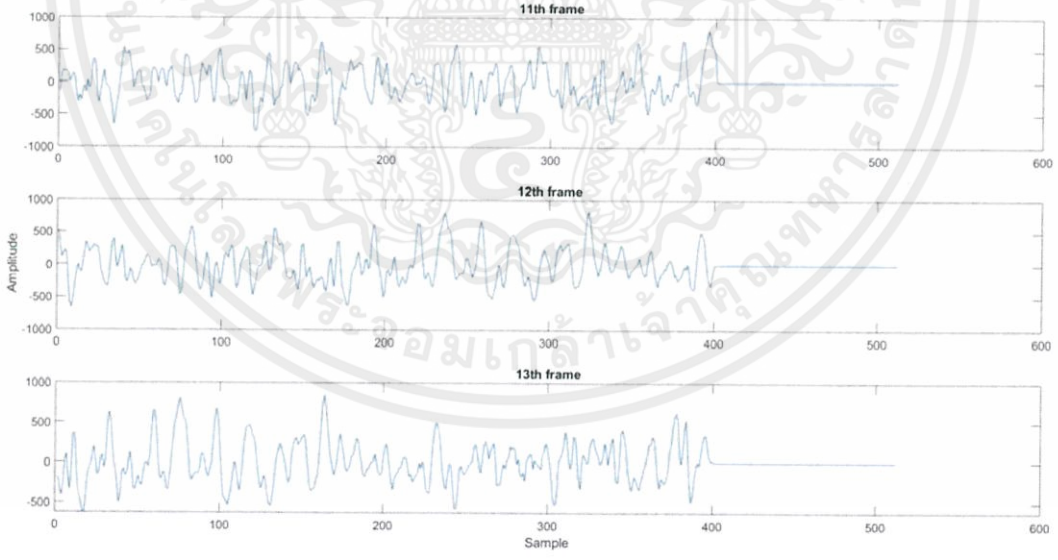
2) การแบ่งเฟรมของสัญญาณเสียง (Framing) แสดงดังรูปที่ 4.2



รูปที่ 4.2 การแบ่งเฟรมสัญญาณเสียง

จากรูปที่ 4.2 เมื่อสัญญาณเสียงถูกแบ่งออกเป็นเฟรม ๆ ในที่นี่ยกตัวอย่างเป็นเฟรมที่ 11, 12 และ 13 ซึ่งแต่ละเฟรมมีความยาว 400 แซมเปิลและระยะห่างของเฟรมมีความยาวเป็น 160 แซมเปิล ทำให้เกิดการซ้อนทับกันของเฟรม โดยการซ้อนทับจะมีความยาวเป็น 80 แซมเปิล

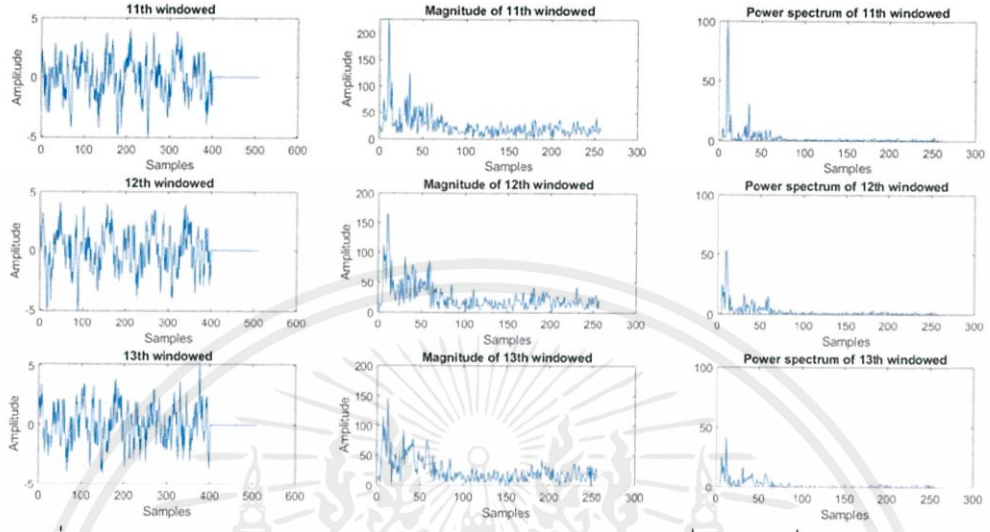
3) การคูณเฟรมของสัญญาณกับฟังก์ชันวินโดว์ และเติมค่าศูนย์ในเฟรมให้มีขนาด 512 แซมเปิล แสดงการทำฟังก์ชันวินโดว์ในเฟรมที่ 11, 12 และ 13 ดังรูปที่ 4.3



รูปที่ 4.3 การทำฟังก์ชันวินโดว์ในเฟรมที่ 11, 12 และ 13

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

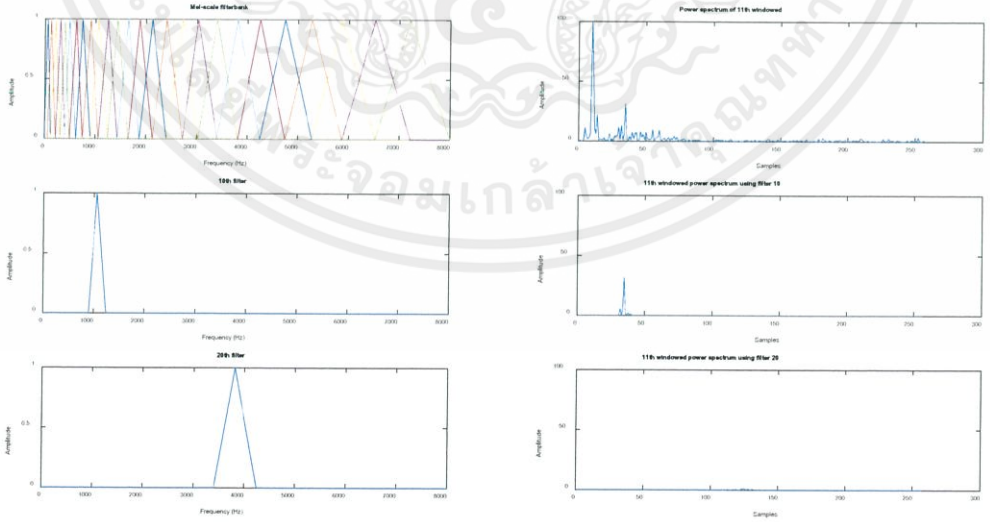
4) การหาค่าสเปกตรัมเชิงกำลังของสัญญาณเสียงในโดเมนความถี่ในเฟรมที่ 11, 12 และ 13 แสดงดังรูปที่ 4.4



รูปที่ 4.4 สเปกตรัมเชิงกำลังของสัญญาณเสียงในโดเมนความถี่ในเฟรมที่ 11, 12 และ 13

จากรูปที่ 4.4 เฟรมที่ 11, 12 และ 13 ของสัญญาณเสียงที่คูณด้วยฟังก์ชันวินโดว์ จะนำไปแปลงเป็นโดเมนความถี่ด้วยดิสครีตฟูเรียร์ทรานฟอร์มหรือ DFT โดยความยาวของเฟรมจะสอดคล้องกับความยาวเฟรมในโดเมนเวลาคือ 512 แซมเปิล จากนั้นหาสเปกตรัมเชิงกำลังของเฟรม ซึ่งจะเก็บค่าสัมประสิทธิ์เฉพาะ 257 แซมเปิลเท่านั้น

5) การทำสเปกตรัมของสเกลเมลคือ การนำสเปกตรัมเชิงกำลังของแต่ละเฟรมไปคูณชุดตัวกรองความถี่ แสดงดังรูปที่ 4.5

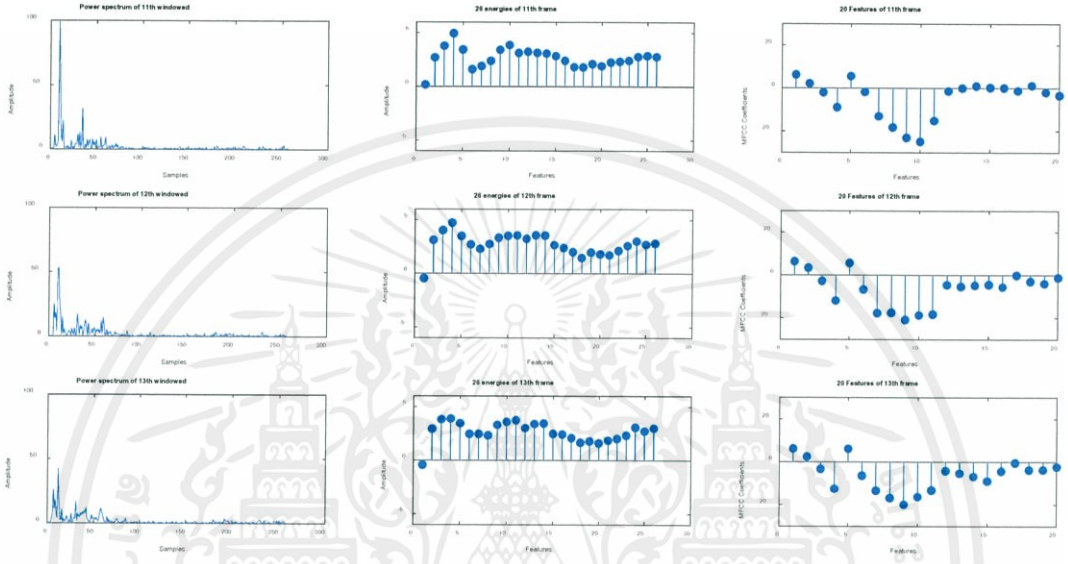


รูปที่ 4.5 สเปกตรัมของเฟรมที่ 11 ที่คูณกับตัวกรองตัวที่ 10 และ 20

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.5 นำสเปกตรัมเชิงกำลังของเฟรมที่ 11 มาคูณกับตัวกรองที่ 10 และ 20 ตามลำดับ ผลลัพธ์ที่ได้คือค่าเฉลี่ยจากการคูณกันระหว่างสเปกตรัมเชิงกำลังกับตัวกรอง

6) ค่าสัมประสิทธิ์เซพสตรัลบนสเกลเมล ทำการหาค่าสัมประสิทธิ์เซพสตรัลบนสเกลเมล ซึ่งผลลัพธ์ได้ดังรูปที่ 4.6



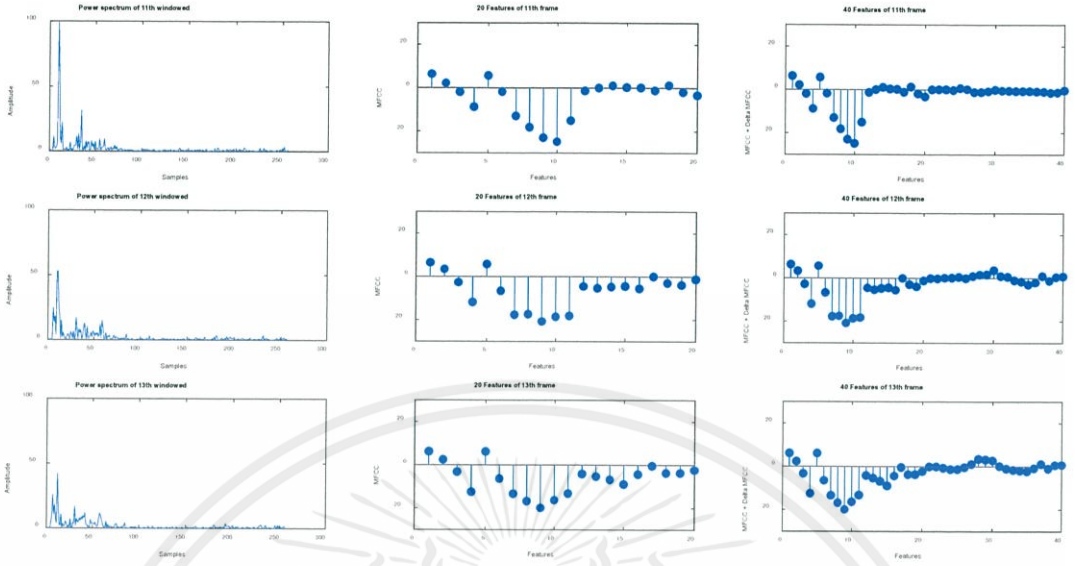
รูปที่ 4.6 การหาค่าสัมประสิทธิ์เซพสตรัลบนสเกลเมล 26 ค่าของเฟรมที่ 11, 12 และ 13

จากรูปที่ 4.6 ค่าเฉลี่ยจากการคูณกันระหว่างสเปกตรัมเชิงกำลังของเฟรมที่ 11, 12 และ 13 กับชุดตัวกรองความถี่บนสเกลเมล 26 ตัวกรอง จะนำไปผ่านฟังก์ชันลอการิทึมและหาค่าสัมประสิทธิ์เซพสตรัลด้วยการแปลงโคซายน์แบบไม่ต่อเนื่องหรือ DCT ซึ่งจะได้ค่าสัมประสิทธิ์เซพสตรัลทั้งหมด 26 ค่าต่อเฟรม จากนั้นนำค่าสัมประสิทธิ์เซพสตรัลมาคูณกับน้ำหนักที่ได้จากวิธีการยกระดับค่าสัมประสิทธิ์ด้วยสัญญาณไซนูซอยดอลและจะเลือกเก็บค่าสัมประสิทธิ์เซพสตรัลเฉพาะ 20 ค่าตัวแรกเท่านั้น เนื่องจากค่าสัมประสิทธิ์เซพสตรัลตัวแรกมีค่าน้อยมาก จึงได้ทำการคำนวณค่าพลังงานรวมของเฟรมที่ 11 และนำไปแทนที่ค่าสัมประสิทธิ์เซพสตรัลตำแหน่งแรก

7) ค่าสัมประสิทธิ์เดลต้าของเซพสตรัลบนสเกลเมล

ทำการหาค่าสัมประสิทธิ์เดลต้าของเซพสตรัลบนสเกลเมล ซึ่งผลลัพธ์ได้ดังรูปที่ 4.7

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 ค่าสัมประสิทธิ์เดลต้าของเฟรมที่ 11, 12 และ 13

จากรูปที่ 4.7 ค่าสัมประสิทธิ์เดลต้าของเฟรมที่ 11, 12 และ 13 จะมี 20 ค่าเท่ากับจำนวนของค่าสัมประสิทธิ์เซพสตรัล จากนั้นนำค่าสัมประสิทธิ์เดลต้ามาเรียงต่อท้ายค่าสัมประสิทธิ์เซพสตรัล ดังนั้นในเฟรมที่ 11, 12 และ 13 จะได้ค่าสัมประสิทธิ์ที่เป็นคุณลักษณะเด่นของสัญญาณเสียงทั้งหมด 40 ค่า ซึ่งคุณลักษณะเด่น 20 ตัวแรกได้มาจากการหาค่าสัมประสิทธิ์เซพสตรัล และคุณลักษณะเด่น 20 ตัวท้ายได้มาจากการหาค่าสัมประสิทธิ์เดลต้าของเซพสตรัล

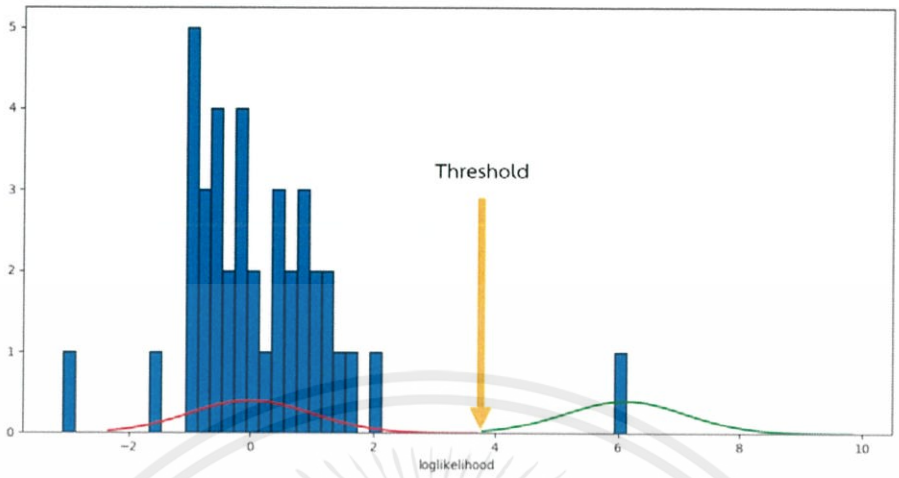
4.1.2 ระบบรู้จำเสียงบุคคล

เมื่อได้ตัวแทนของสัญญาณเสียงจากการคัดเลือกคุณลักษณะเด่น จะทำการจำแนกรู้จำด้วยแบบจำลองผสมแบบเกาส์ จะได้ไฟล์รูปแบบ .gmm ของแต่ละบุคคลในฐานข้อมูลสำหรับใช้การจำแนกบุคคล

4.1.3 การหาค่าขีดแบ่งสำหรับตัดสินใจ

การหาค่าขีดแบ่งสำหรับตัดสินใจว่าไฟล์เสียงอินพุตนั้นเป็นใคร และเป็นบุคคลในฐานข้อมูลหรือไม่ สามารถหาได้จากกราฟฮิสโทแกรมแสดงดังรูปที่ 4.8

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 การหาค่าขีดแบ่งสำหรับตัดสินใจ จากกราฟฮิสโทกราฟ

จากรูปที่ 4.8 ฮิสโทกราฟที่ได้จากการทดสอบประสิทธิภาพของระบบจะคำนวณค่าขีดแบ่งสำหรับตัดสินใจจากสมการที่ 3.1 ซึ่งจะได้ค่าขีดแบ่งสำหรับตัดสินใจของระบบโดยค่าประมาณ 3.75

4.1.4 การทดลองเพื่อวัดประะสิทธิภาพระบบรู้จำเสียงบุคคล

การทดลองเพื่อวัดประสิทธิภาพของระบบรู้จำเสียงบุคคล มีปัจจัยในการทดลองคือความยาวของไฟล์เสียงสำหรับฝึกสอนที่มีความยาวตั้งแต่ 10 วินาที, 20 วินาที และ 30 วินาทีและจำนวนลักษณะเด่นตั้งแต่ 16 ตัว, 26 ตัว และ 40 ตัว แสดงผลการทดลองดังตารางที่ 4.1

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 ผลการทดลองระบบรู้จำเสียงบุคคล

Amount of training speech	The number of people used to train model	Duration of testing speech	
		5 sec	10 sec
10 sec	16	71.11%	60%
	26	88.89%	88.89%
	40	93.33%	88.89%
20 sec	16	84.44%	73.33%
	26	93.33%	93.33%
	40	100%	93.33%
30 sec	16	75.56%	68.89%
	26	93.33%	93.33%
	40	97.78%	95.56%

จากตารางที่ 4.1 จะเห็นว่าระบบรู้จำเสียงบุคคลที่มีประสิทธิภาพดีที่สุด สามารถนำไปใช้งานได้จริงในสภาพแวดล้อมที่มีสิ่งรบกวนคือ ระบบที่ใช้ไฟล์เสียงสำหรับฝึกสอนระบบที่มีความยาว 20 วินาที จำนวนลักษณะเด่นต่อเฟรมอยู่ที่ 40 ตัว และมีความยาวของไฟล์เสียงทดสอบคือ 5 วินาที

4.2 ระบบรู้จำใบหน้าบุคคล

แสดงผลลัพธ์ของการทดลองระบบรู้จำใบหน้าบุคคล ในแต่ละขั้นตอนดังนี้

4.2.1 การตรวจจับและบันทึกรูปภาพใบหน้าบุคคล

การตรวจจับใบหน้าจะทำการค้นหาใบหน้าจากรูปภาพ เมื่อค้นหาใบหน้าเจอจะทำการบันทึกรูปภาพใบหน้าให้อยู่ในรูปแบบ .jpg

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 การเตรียมรูปภาพใบหน้าบุคคล

แสดงผลลัพธ์รูปภาพสำหรับฝึกสอนโมเดลที่ทำการปรับขนาดแล้ว ดังรูปที่ 4.9

รูปที่ 4.9 รูปภาพใบหน้าขนาด 100x100x3 พิกเซล สำหรับฝึกสอนโมเดล

4.2.3 การดึงคุณลักษณะเด่นของรูปภาพใบหน้าบุคคล

แสดงผลลัพธ์การดึงคุณลักษณะเด่นของรูปภาพด้วยตัวกรองขนาด 5x5 จำนวน 32 ตัว ในชั้นคอนโวลูชัน ดังรูปที่ 4.10

Example Image size 100x100x3 pixels

The first layer 5x5x32 filter

รูปที่ 4.10 การดึงคุณลักษณะเด่นของรูปภาพ

4.2.4 ระบบรู้จำ

นำผลลัพธ์จากการดึงคุณลักษณะเด่นของรูปภาพมาเรียนรู้ด้วยระบบโครงข่ายประสาทเทียม สุดท้ายจะได้โมเดลสำหรับจำแนกบุคคลด้วยใบหน้าที่ถูกบันทึกในรูปแบบไฟล์ .ckpt.meta ชื่อว่า “Model_01” ดังรูปที่ 4.11

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

model_01.ckpt.m
eta

รูปที่ 4.11 โมเดลสำหรับจำแนกบุคคลด้วยใบหน้า ชื่อ “Model_01.ckpt.meta”

4.2.5 การทดลองเพื่อวัดประสิทธิภาพระบบรู้จำใบหน้าบุคคล

การทดลองเพื่อวัดประสิทธิภาพของระบบรู้จำใบหน้าบุคคล จะทดสอบโดยใช้รูปภาพที่ไม่เกี่ยวข้องกับการฝึกสอนโมเดลของแต่ละบุคคล โดยทดสอบทั้งหมด 10 คลาสและทดสอบแต่ละคลาสเป็นจำนวน 10 ครั้ง แสดงผลการทดลองดังตารางที่ 4.2-4.4

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 ผลการทดลองระบบรู้จำใบหน้าบุคคล

No.	ฟ้า		เหมียว		ฟลุค		ตุ้งติ้ง		นอธ		แจ็บ		กิ๊ฟ		แนน		ปุ๊กกี้		สมาร์ท	
	A	P	A	P	A	P	A	P	A	P	A	P	A	P	A	P	A	P	A	P
1	✗	✗	✓	✓	✗	✓	✓	✗	✓	✓	✗	✗	✓	✓	✗	✗	✗	✓	✓	✗
2	✓	✓	✗	✗	✓	✓	✓	✓	✗	✗	✓	✓	✗	✓	✓	✓	✗	✗	✗	✓
3	✓	✓	✗	✓	✓	✗	✗	✗	✓	✓	✗	✓	✓	✗	✗	✓	✓	✓	✗	✓
4	✗	✗	✓	✗	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✓	✓	✗	✗
5	✓	✓	✗	✓	✗	✗	✗	✗	✓	✓	✗	✓	✓	✓	✓	✓	✗	✗	✓	✓
6	✗	✓	✓	✓	✗	✓	✓	✓	✗	✗	✗	✗	✓	✓	✗	✗	✓	✓	✓	✗
7	✗	✓	✓	✓	✓	✓	✗	✓	✓	✓	✗	✗	✓	✓	✗	✓	✓	✗	✗	✓
8	✗	✗	✗	✗	✓	✗	✓	✗	✗	✗	✓	✓	✗	✗	✓	✓	✗	✗	✓	✗
9	✓	✓	✗	✓	✓	✓	✗	✓	✗	✗	✓	✗	✗	✗	✓	✓	✓	✓	✗	✗
10	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗	✓	✓	✗	✗	✓	✗	✗	✗	✓	✓

จากตารางที่ 4.2 Actual (A) คือ คลาสที่อยู่ในข้อมูลดาต้าและ Predict (P) คือคลาสที่โมเดลทำนายมาได้ ในที่นี่จะยกตัวอย่างผลการทดสอบระบบรู้จำใบหน้าของคลาสชื่อ “ฟ้า” สามารถอธิบายได้ดังนี้ จำนวนข้อมูลที่ทำนายถูกว่าเป็นคลาส (✓) ให้เป็น TP (True Positive) เป็นจำนวน 4 ตัว (ในแถวที่ 2, 3, 5 และ 9) จำนวนข้อมูลที่ทำนายถูกว่าเป็นคลาส (✗) ให้เป็น TN (True Negative) เป็นจำนวน 1 ตัว (ในแถวที่ 10) จำนวนข้อมูลที่ทำนายผิดมาเป็นคลาส (✓) ให้เป็น FP (False Positive) เป็นจำนวน 2 ตัว (ในแถวที่ 6 และ 7) และจำนวนข้อมูลที่ทำนายผิดมาเป็น

คลาส (✕) ให้เป็น FN (False Negative) เป็นจำนวน 2 ตัว ในแถวที่ (1 และ 8) จากนั้นนำข้อมูลเหล่านี้ไปสร้างตารางคอนฟูชั่นแมททริกในตารางที่ 4.3

ตารางที่ 4.3 ผลการทดลองการวัดประสิทธิภาพข้อมูลของตาราง Confusion Matrix (คอนฟูชั่น แมททริก)

A/P	ฟ้า		เหมียว		ฟลุค		ตุ้งติ้ง		นอธ		แจ็บ		กี๊ฟ		แนน		ปุ๊กกี้		สมาร์ท	
	Yes	No	Yes	No	Yes	No	Yes	No	Yes	No	Yes	No	Yes	No	Yes	No	Yes	No	Yes	No
Yes	4	2	3	3	4	2	4	2	5	0	3	2	4	1	4	2	3	1	3	3
No	3	1	2	2	2	2	3	2	5	0	2	3	4	1	3	1	4	2	2	2

จากตารางที่ 4.3 จะยกตัวอย่างการพิจารณาคลาสชื่อ “ฟ้า” ในระบบรู้จำใบหน้าบุคคล สามารถอธิบายได้ดังนี้ ค่าความแม่นยำ (Precision) ของคลาส “ฟ้า” คำนวณได้จากสมการที่ 3.3 คิดเป็นร้อยละ 0.67 ค่าความถูกต้องของคลาส “ฟ้า” คำนวณได้จากสมการที่ 3.4 คิดเป็นร้อยละ 0.67 และพิจารณาความถูกต้องของโมเดลโดยพิจารณารวมทุกคลาส คำนวณได้จากสมการที่ 3.5 คิดเป็นร้อยละ 53

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

ปริญญานิพนธ์นี้นำเสนอระบบรู้จำเสียงผู้พูดและใบหน้าบุคคลด้วยการเรียนรู้แบบปัญญาประดิษฐ์ โดยระบบรู้จำเสียงบุคคลจะรู้จำเสียงบุคคลทั้งหมด 5 คน ทำการคัดเลือกคุณลักษณะเด่นของสัญญาณเสียงผู้พูดแต่ละคนด้วยวิธีการหาค่าสัมประสิทธิ์เซบสตรัมบนสเกลเมลและจำแนกการรู้จำด้วยวิธีการแบบจำลองผสมแบบเกาส์ ผลลัพธ์การทดสอบระบบรู้จำเสียงผู้พูดเพื่อวัดประสิทธิภาพโมเดล ระบบรู้จำเสียงบุคคลที่มีประสิทธิภาพดีที่สุดคือระบบที่ใช้ไฟล์เสียงสำหรับฝึกสอนระบบที่มีความยาว 20 วินาที จำนวนลักษณะเด่นต่อเฟรมอยู่ที่ 40 ตัว และมีความยาวของไฟล์เสียงทดสอบคือ 5 วินาที โดยพิจารณารวมทุกคลาสคิดเป็นร้อยละ 100

ระบบรู้จำใบหน้าบุคคลจะรู้จำใบหน้าบุคคลทั้งหมด 5 คน ทำการคัดเลือกคุณลักษณะเด่นและจำแนกการรู้จำของใบหน้าบุคคลด้วยวิธีการโครงข่ายประสาทเทียมแบบคอนโวลูชัน ผลลัพธ์การทดสอบระบบรู้จำใบหน้าบุคคลเพื่อวัดประสิทธิภาพโมเดลคิดเป็นร้อยละ 53

5.2 ข้อเสนอแนะ

เนื่องจากอัตราความถูกต้องขึ้นอยู่กับจำนวนและความคลายคลึงกันของข้อมูลชนิดเดียวกัน จำเป็นต้องเตรียมข้อมูลสำหรับการเรียนรู้แบบปัญญาประดิษฐ์ให้ดีและมีมาตรฐานเพื่อให้ได้ระบบรู้จำที่มีความแม่นยำและน่าเชื่อถือมากขึ้น และระบบระบุตัวตนด้วยปัญญาประดิษฐ์นี้เป็นการประยุกต์ใช้ระบบในการตรวจสอบบุคคลก่อนเข้าถึงพื้นที่นั้น ๆ ซึ่งเป็นการนำระบบมาใช้แบบนำร่องของการรักษาความปลอดภัยเบื้องต้น โดยยังสามารถนำระบบไปประยุกต์ในงานด้านอื่น ๆ ที่ต้องการตรวจสอบบุคคลได้อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] X. Huang, A. Acero, and H. Hon. *"Spoken Language Processing: A guide to theory, algorithm, and system development"*. Prentice Hall, 2001

[2] Abhijeet Kumar. *"Spoken Speaker Identification based on Gaussian Mixture Models"*. https://appliedmachinelearning.blog/2017/11/14/

[3] paoonline. *"การวัดประสิทธิภาพโมเดลของ (Data Mining)"*. https://www.mindphp.com/forums/viewtopic.php?f=29&t=42840

[4] karpathy. *"Convolutional Neural Networks (CNNs / ConvNets)"*. http://cs231n.github.io/convolutional-networks/

[5] ศิรเสกข์ ตราชื่นต้อง และอธิษฐ์ แย้มสุข. "การพัฒนาระบบรู้จำเสียงพูด/ภาพใบหน้าและแนวทางการประยุกต์ใช้งาน". ปริญญานิพนธ์หลักสูตรวิศวกรรมศาสตรบัณฑิต, สาขาวิชาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์, สจล, 2559.

[6] Davis, S. Mermelstein, P. (1980) *Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences*. In IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 28 No. 4, pp. 357-366

[7] Jameslyons. *"Mel Frequency Cepstral Coefficient (MFCC) tutorial"*. http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/

[8] K.K. Paliwal. *"DECORRELATED AND LIFTERED FILTER-BANK ENERGIES FOR ROBUST SPEECH RECOGNITION"*. School of Microelectronic Engineering Grith University Brisbane, QLD 4111, Australia

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

Source Code Speaker Recognition

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) Speaker Feature Extraction

```
import numpy as np
from sklearn import preprocessing
import python_speech_features as mfcc
import python_speech_features as logfbank
import matplotlib.pyplot as plt

def calculate_delta(array):
    """Calculate and returns the delta of given feature vector matrix"""

    rows,cols = array.shape
    deltas = np.zeros((rows,20))
    N = 2
    for i in range(rows):
        index = []
        j = 1
        while j <= N:
            if i-j < 0:
                first = 0
            else:
                first = i-j
            if i+j > rows -1:
                second = rows -1
            else:
                second = i+j
            index.append((second,first))
            j+=1
        deltas[i] = ( array[index[0][0]]-array[index[0][1]] + (2 * (array[index[1][0]]-
array[index[1][1]])) ) / 10
    return deltas
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
def extract_features(audio,rate):
    """extract 20 dim mfcc features from an audio, performs CMS and combines
    delta to make it 40 dim feature vector"""

    mfcc_feat = mfcc.mfcc(audio,rate, 0.025, 0.01,20,appendEnergy = True)

    mfcc_feat = preprocessing.scale(mfcc_feat)
    delta = calculate_delta(mfcc_feat)
    combined = np.hstack((mfcc_feat,delta))

    return combined

if __name__ == "__main__":
     print("In main, Call extract_features(audio,signal_rate) as parameters")
```

2) Train Models

```
import numpy as np
from sklearn import preprocessing
import python_speech_features as mfcc
import python_speech_features as logfbank
import matplotlib.pyplot as plt

def calculate_delta(array):
    """Calculate and returns the delta of given feature vector matrix"""

    rows,cols = array.shape
    deltas = np.zeros((rows,20))
    N = 2
    for i in range(rows):
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        index = []
        j = 1
        while j <= N:
            if i-j < 0:
                first = 0
            else:
                first = i-j
            if i+j > rows -1:
                second = rows -1
            else:
                second = i+j
            index.append((second,first))
            j+=1
        deltas[i] = ( array[index[0][0]]-array[index[0][1]] + (2 * (array[index[1][0]]-
array[index[1][1]])) ) / 10
    return deltas

def extract_features(audio,rate):
    """extract 20 dim mfcc features from an audio, performs CMS and combines
    delta to make it 40 dim feature vector"""

    mfcc_feat = mfcc.mfcc(audio,rate, 0.025, 0.01,20,appendEnergy = True)

    mfcc_feat = preprocessing.scale(mfcc_feat)
    delta = calculate_delta(mfcc_feat)
    combined = np.hstack((mfcc_feat,delta))

    return combined

if __name__ == "__main__":
    print("In main, Call extract_features(audio,signal_rate) as parameters")
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) Test Speaker

```
import os
import pickle as cPickle
import numpy as np
import scipy.io.wavfile
from speakerfeatures import extract_features
import warnings
warnings.filterwarnings("ignore")
import time
import matplotlib.pyplot as plt

# Define a function
def testsound():

    speaker_features = __import__("speakerfeatures")

    extract_features = speaker_features.extract_features

    #path to training data
    source   = "test_speaker\\"

    modelpath = "speaker_models_2\\"

    test_file = "test_speaker.txt"

    file_paths = open(test_file, 'rb')


    gmm_files = [os.path.join(modelpath,fname) for fname in
            os.listdir(modelpath) if fname.endswith('.gmm')]
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#Load the Gaussian gender Models
models    = [cPickle.load(open(fname, 'rb')) for fname in gmm_files]
speakers   = [fname.split("\\")[-1].split(".gmm")[0] for fname
         in gmm_files]

s2=[]
# Read the test directory and get the list of test audio files
for path in file_paths:

    path = path.strip()
    print(path)
    sr,audio = scipy.io.wavfile.read(source + path.decode('utf-8'))
    vector   = extract_features(audio,sr)
    #print(vector)

    log_likelihood = np.zeros(len(models))


    for i in range(len(models)):
        gmm    = models[i]     #checking with each model one by one
        scores = np.array(gmm.score(vector))
        log_likelihood[i] = scores.sum()

    a = np.delete(log_likelihood, np.argmax(log_likelihood))
    z = (max(log_likelihood) - np.mean(a)) / np.std(a)
    z1 = [((e - np.mean(a)) / np.std(a)) for e in log_likelihood]
    #print(z1)
    print(z)

    if z >= 3.75:
        winner = np.argmax(log_likelihood)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            print("\tdetected as - ", speakers[winner])
        else:
            print("\tUnknown")
            time.sleep(1.0)
```

4) Audio Recording

```
import audioop
import pyaudio
import wave
from test_speaker_module import testsound
import matplotlib.pyplot as plt
import time
import numpy as np


FORMAT = pyaudio.paInt16
CHANNELS = 1
RATE = 16000
CHUNK = 1024
RECORD_SECONDS = 5
WAVE_OUTPUT_FILENAME = "temp.wav"

# this is the threshold that determines whether or not sound is detected
THRESHOLD = 100

#open your audio stream
while True:
    audio  = pyaudio.PyAudio()
    stream = audio.open(format=FORMAT, channels=CHANNELS,
                rate=RATE, input=True,
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            frames_per_buffer=CHUNK)

    # wait until the sound data breaks some level threshold
    while True:
        data = stream.read(CHUNK)
        # check level against threshold,
        rms = audioop.rms(data, 2)  #width=2 for format=paInt16
        #print(rms)
        if rms > THRESHOLD:
            break

    # record for however long you want
    #audio = pyaudio.PyAudio()

    # start Recording
    print("recording...")
    frames = []

    for i in range(0, int(RATE / CHUNK * RECORD_SECONDS)):
        data = stream.read(CHUNK)
        frames.append(data)
    print("finished recording")

    # stop Recording
    stream.stop_stream()
    # close the stream
    stream.close()
    audio.terminate()

    waveFile = wave.open("D:\\Speaker-identification-using-GMMs-
master\\test_speaker\\test_file\\wav\\" + WAVE_OUTPUT_FILENAME, 'wb')
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    waveFile.setnchannels(CHANNELS)
    waveFile.setsampwidth(audio.get_sample_size(FORMAT))
    waveFile.setframerate(RATE)
    waveFile.writeframes(b''.join(frames))
    waveFile.close()

    raw = ' '
    wav = "D:\\Speaker-identification-using-GMMs-
master\\test_speaker\\test_file\\wav\\temp.wav"
    elements = wav.split(' ')
    raw = ' '.join(['\x00'+e if len(e)==1 else e for e in elements])
    amp = np.fromstring(raw, np.uint8)
    print(amp)

    testsound()

    time.sleep(1.0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

Source Code Facial Recognition

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) Train Models

```
from skimage import io,transform
import matplotlib.pyplot as plt
import glob
import os
import tensorflow as tf
import numpy as np
import time

path='bigset/'
model_path='Model/model_8.ckpt'

w=100
h=100
c=3


def read_img(path):
    cate=[path+x for x in os.listdir(path) if os.path.isdir(path+x)]
    imgs=[]
    labels=[]
    for idx,folder in enumerate(cate):
        for im in glob.glob(folder+'/*.jpg'):
            #print('reading the images:%s'%(im))
            img=io.imread(im)
            img=transform.resize(img,(w,h))
            imgs.append(img)
            labels.append(idx)
    return np.asarray(imgs,np.float32),np.asarray(labels,np.int32)

data,label=read_img(path)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#print(data)
#print(label)

num_example=data.shape[0]
arr=np.arange(num_example)
np.random.shuffle(arr)
data=data[arr]
label=label[arr]

ratio=0.8
s=np.int(num_example*ratio)
x_train=data[:s]
y_train=label[:s]
x_val=data[s:]
y_val=label[s:]

x=tf.placeholder(tf.float32,shape=[None,w,h,c],name='x')
y_=tf.placeholder(tf.int32,shape=[None,],name='y_')

def inference(input_tensor, train, regularizer):
    with tf.variable_scope('layer1-conv1'):
        conv1_weights = tf.get_variable("weight",[5,5,3,32],initializer=tf.truncated_normal_initializer(stddev=0.1))
        conv1_biases = tf.get_variable("bias", [32], initializer=tf.constant_initializer(0.0))
        conv1 = tf.nn.conv2d(input_tensor, conv1_weights, strides=[1, 1, 1, 1], padding='SAME')
        relu1 = tf.nn.relu(tf.nn.bias_add(conv1, conv1_biases))

    with tf.name_scope("layer2-pool1"):
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                pool1 = tf.nn.max_pool(relu1, ksize =
[1,2,2,1],strides=[1,2,2,1],padding="VALID")

            with tf.variable_scope("layer3-conv2"):
                conv2_weights =
tf.get_variable("weight",[5,5,32,64],initializer=tf.truncated_normal_initializer(stddev=0.1)
)
                conv2_biases = tf.get_variable("bias", [64],
initializer=tf.constant_initializer(0.0))
                conv2 = tf.nn.conv2d(pool1, conv2_weights, strides=[1, 1, 1, 1],
padding='SAME')
                relu2 = tf.nn.relu(tf.nn.bias_add(conv2, conv2_biases))

            with tf.name_scope("layer4-pool2"):
                pool2 = tf.nn.max_pool(relu2, ksize=[1, 2, 2, 1], strides=[1, 2, 2, 1],
padding='VALID')

            with tf.variable_scope("layer5-conv3"):
                conv3_weights =
tf.get_variable("weight",[3,3,64,128],initializer=tf.truncated_normal_initializer(stddev=0.
1))
                conv3_biases = tf.get_variable("bias", [128],
initializer=tf.constant_initializer(0.0))
                conv3 = tf.nn.conv2d(pool2, conv3_weights, strides=[1, 1, 1, 1],
padding='SAME')
                relu3 = tf.nn.relu(tf.nn.bias_add(conv3, conv3_biases))

            with tf.name_scope("layer6-pool3"):
                pool3 = tf.nn.max_pool(relu3, ksize=[1, 2, 2, 1], strides=[1, 2, 2, 1],
padding='VALID')
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            with tf.variable_scope("layer7-conv4"):
                conv4_weights = tf.get_variable("weight",[3,3,128,128],initializer=tf.truncated_normal_initializer(stddev=0.1))
                conv4_biases = tf.get_variable("bias", [128], initializer=tf.constant_initializer(0.0))
                conv4 = tf.nn.conv2d(pool3, conv4_weights, strides=[1, 1, 1, 1], padding='SAME')
                relu4 = tf.nn.relu(tf.nn.bias_add(conv4, conv4_biases))

            with tf.name_scope("layer8-pool4"):
                pool4 = tf.nn.max_pool(relu4, ksize=[1, 2, 2, 1], strides=[1, 2, 2, 1], padding='VALID')
                nodes = 6*6*128
                reshaped = tf.reshape(pool4,[-1,nodes])

            with tf.variable_scope('layer9-fc1'):
                fc1_weights = tf.get_variable("weight", [nodes, 1024],

initializer=tf.truncated_normal_initializer(stddev=0.1))
                if regularizer != None: tf.add_to_collection('losses', regularizer(fc1_weights))
                fc1_biases = tf.get_variable("bias", [1024], initializer=tf.constant_initializer(0.1))

                fc1 = tf.nn.relu(tf.matmul(reshaped, fc1_weights) + fc1_biases)
                if train: fc1 = tf.nn.dropout(fc1, 0.5)

            with tf.variable_scope('layer10-fc2'):
                fc2_weights = tf.get_variable("weight", [1024, 512],
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
initializer=tf.truncated_normal_initializer(stddev=0.1))
                    if regularizer != None: tf.add_to_collection('losses',
regularizer(fc2_weights))
                    fc2_biases = tf.get_variable("bias", [512],
initializer=tf.constant_initializer(0.1))

                    fc2 = tf.nn.relu(tf.matmul(fc1, fc2_weights) + fc2_biases)
                    if train: fc2 = tf.nn.dropout(fc2, 0.5)

                with tf.variable_scope('layer11-fc3'):
                    fc3_weights = tf.get_variable("weight", [512, 5],

initializer=tf.truncated_normal_initializer(stddev=0.1))
                    if regularizer != None: tf.add_to_collection('losses',
regularizer(fc3_weights))
                    fc3_biases = tf.get_variable("bias", [5],
initializer=tf.constant_initializer(0.1))
                    logit = tf.matmul(fc2, fc3_weights) + fc3_biases

                return logit

            regularizer = tf.contrib.layers.l2_regularizer(0.0001)
            logits = inference(x,False,regularizer)
            print(logits)

            b = tf.constant(value=1,dtype=tf.float32)
            logits_eval = tf.multiply(logits,b,name='logits_eval')
            print(logits_eval)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
loss=tf.nn.sparse_softmax_cross_entropy_with_logits(logits=logits,
labels=y_)
train_op=tf.train.AdamOptimizer(learning_rate=0.001).minimize(loss)
correct_prediction = tf.equal(tf.cast(tf.argmax(logits,1),tf.int32), y_)
acc= tf.reduce_mean(tf.cast(correct_prediction, tf.float32))

def
minibatches(inputs=None,targets=None,batch_size=None,shuffle=False):
    assert  len(inputs) == len(targets)
    if shuffle:
        indices = np.arange(len(inputs))
    for start_idx in range(0,len(inputs) - batch_size + 1,batch_size):
        if shuffle:
            excerpt = indices[start_idx:start_idx + batch_size]
        else:
            excerpt = slice(start_idx,start_idx+batch_size)
        yield inputs[excerpt],targets[excerpt]


n_epoch=10
batch_size=8
saver=tf.train.Saver()
sess=tf.Session()
sess.run(tf.global_variables_initializer())

for epoch in range(n_epoch):
    print("epoch:", epoch+1)
    start_time = time.time()

    #training
    train_loss, train_acc, n_batch = 0, 0, 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            for x_train_a, y_train_a in minibatches(x_train, y_train, batch_size,
shuffle=True):
                _,err,ac=sess.run([train_op,loss,acc], feed_dict={x: x_train_a, y_:
y_train_a})
                train_loss += err; train_acc += ac; n_batch += 1
            print("   train loss: %f" % (np.sum(train_loss) / n_batch))
            print("   train acc: %f" % (np.sum(train_acc) / n_batch))

            #validation
            val_loss, val_acc, n_batch = 0, 0, 1
            for x_val_a, y_val_a in minibatches(x_val, y_val, batch_size,
shuffle=False):
                err, ac = sess.run([loss,acc], feed_dict={x: x_val_a, y_: y_val_a})
                val_loss += err; val_acc += ac; n_batch += 1
            print("   validation loss: %f" % (np.sum(val_loss) / n_batch))
            print("   validation acc: %f" % (np.sum(val_acc) / n_batch))
            print(" epoch time %f:"%(time.time()-start_time))
            print('----------------------------------------------------')
        saver.save(sess,model_path)
        sess.close()
```

2) Test Models

```
from skimage import io,transform
import tensorflow as tf
import numpy as np


path1 = "pookie.jpg"
path2 = "smart.jpg"

flower_dict = {0:'Pookie',1:'Smart'}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
w=100
h=100
c=3

def read_one_image(path):
    img = io.imread(path)
    img = transform.resize(img,(w,h))
    return np.asarray(img)

with tf.Session() as sess:
    data = []
    data1 = read_one_image(path1)
    data2 = read_one_image(path2)
    data.append(data1)
    data.append(data2)

    saver = tf.train.import_meta_graph('Model/voice_01.ckpt.meta')
    saver.restore(sess,tf.train.latest_checkpoint('Model/'))

    # output model parameter
    graph = tf.get_default_graph()
    x = graph.get_tensor_by_name("x:0")
    feed_dict = {x:data}
    #print(feed_dict)

    logits = graph.get_tensor_by_name("logits_eval:0")
    print(logits)

    classification_result = sess.run(logits,feed_dict)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
# print prediction matrix, and maximum value of each row of the prediction matrix
print(classification_result)
print(tf.argmax(classification_result,1).eval())

output = []
output = tf.argmax(classification_result,1).eval()
for i in range(len(output)):
    print("Person",i+1,"prediction:"+flower_dict[output[i]])
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้