สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบส่งเสริมการเรียนรู้ศัพท์ภาษามือไทยด้วยไอโฟน

THAI SIGN LANGUAGE LEARNING SYSTEM WITH IPHONE

T117361

พีรวัส พลวิเศษ
พีระ ทั้งสุข
ภัทร นรเศรษฐ์โสภน

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
สารบัญ
เลขทะเบียน 117361
วัน,เดือน,ปี - 1 ส.ค. 2554
b 12343481
i

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2553

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบส่งเสริมการเรียนรู้ศัพท์ภาษามือไทยด้วยไอโฟน

THAI SIGN LANGUAGE LEARNING SYSTEM WITH IPHONE

ผู้จัดทำ

1. นายพีรวัส	พลวิเศษ	รหัสนักศึกษา	50011120
2. นายพีระ	ทั้งสุข	รหัสนักศึกษา	50011123
3. นายภัทร	นรเศรษฐ์โสภน	รหัสนักศึกษา	50011153

ชุติเมษฎ์ ศรีนิลทา อาจารย์ที่ปรึกษา

(ผศ.ดร.ชุติเมษฎ์ ศรีนิลทา)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบส่งเสริมการเรียนรู้ศัพท์ภาษามือไทยด้วยไอโฟน

นายพีรวัส	พลวิเศษ	50011120
นายพีระ	ทั้งสุข	50011123
นายภัทร	นรเศรษฐ์โศภน	50011153
ผศ.ดร.ชุติเมษฏ์	ศรีนิลทา	อาจารย์ที่ปรึกษา

ปีการศึกษา 2553

บทคัดย่อ

ภาษามือเป็นวิธีหลักที่ผู้พิการทางการได้ยินหรือคนหูหนวกใช้ในการสื่อสารทั้งระหว่างผู้พิการด้วยกันเองและกับคนปกติ ปัจจุบันมีการสอนภาษามือในโรงเรียนสำหรับผู้พิการทางการได้ยิน ซึ่งการเรียนรู้ภาษามือนั้นต้องเริ่มจากการเรียนรู้คำศัพท์ โครงงานนี้จึงได้นำเสนอระบบส่งเสริมการเรียนรู้ศัพท์ภาษามือไทยจากวัตถุที่พบในชีวิตประจำวันผ่านโทรศัพท์มือถือ ซึ่งผู้ใช้สามารถเรียนรู้ศัพท์ใหม่ได้ตลอดเวลาจากสิ่งที่พบเห็น โดยเริ่มจากผู้ใช้ถ่ายรูปวัตถุที่ต้องการรู้ภาษามือด้วยโทรศัพท์มือถือ ระบบจะใช้วิธีการประมวลผลภาพหาขอบของวัตถุเพื่อระบุรูปร่างของวัตถุ จากนั้นนำรูปร่างของวัตถุไปประมวลผลใน โครงข่ายประสาทเทียม ซึ่งได้ทำการเรียนรู้รูปร่างของวัตถุต่างๆไว้ล่วงหน้าแล้ว หลังจากนั้นจะให้ผู้ใช้เลือกภาพที่ใกล้เคียงกับวัตถุที่ต้องการมากที่สุด (ในกรณีที่วัตถุรูปร่างคล้ายกัน) จากนั้นระบบจะแสดงภาพตัวละครสามมิติแสดงภาษามือของวัตถุนั้นๆ นอกจากนี้ยังได้พัฒนาระบบให้เกิดความน่าเรียนรู้โดยพัฒนาเป็นเกมซึ่งผู้ใช้สามารถทบทวนคำศัพท์ที่เคยเรียนไปแล้วได้โดยเทรนรูปจำนวน 26 ชนิดโดยผลลัพธ์จากการทดลองการประมวลผลภาพจะเป็นภาพรูปทรงของวัตถุโดยตัววัตถุจะมีสีขาวและพื้นหลังมีสีดำโดยไม่มีรายละเอียดใดๆในพื้นหลังของภาพเลยและนำภาพที่ได้นี้ไปเป็นตัวนำเข้าของโครงข่ายประสาทเทียมเพื่อทำกระบวนวิธีการเรียนรู้โดยใช้กระบวนวิธีเอสซีจีและนำผลลัพธ์ของโครงสร้างที่ได้รับการเรียนรู้มาใช้ในการทำนายภาพต่อไป

โครงงานนี้เลือกใช้ ภาษาC# ในการพัฒนาโปรแกรมในส่วน การประมวลผลภาพ และโครงข่ายประสาทเทียม ร่วมกับการใช้ Framework EmguCV และใช้ Encog Libraryในการทำส่วนของ โครงข่ายประสาทเทียมและในส่วนโปรแกรมประยุกต์บน iPhone ได้ใช้ภาษาObjective-C ในการพัฒนา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thai Sign Language Learning System with iPhone

Mr. Peeravat	Pholvises	50011120
Mr. Peera	Tungsuk	50011123
Mr. Patara	Norasethasopon	50011153
Asst.Prof.Dr. Chutimet	Srinilta	Advisor

Academic Year 2010

ABSTRACT

Deaf or hard-of-hearing people communicate mainly via sign language. The first step to learn a new language is to understand its vocabulary. We proposed a system to facilitate learning Thai sign language vocabulary from things (objects) that are around us. User can learn new words from anywhere at any time. From iPhone, application user takes photo of an object he/she wants to know the sign that is associated with it. The system determines the vocabulary for that object and responds with a 3D animated image showing how to express it with Thai sign language. Image processing techniques are employed to separate object from background to determine object contour. Artificial neural network is trained with SCG algorithm to recognize object in the target image. Games on iPhone are proposed in order to promote learning efficiency. We use 26 kinds of object picture in our project.

C# Programming language is chosen for our image processing and neural network components. The EmguCV framework and Encog library are also used in the neural network component. iPhone application is developed with Objective C programming language.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานฉบับนี้สำเร็จได้อย่างดี ด้วยคำแนะนำ และคำปรึกษาจาก ผศ.ดร.ชุติเมษฎ์ ศรีนิลทา อาจารย์ที่ปรึกษาซึ่งเป็นอาจารย์ที่ปรึกษาโครงงาน ข้าพเจ้ารู้สึกซาบซึ้งในความอนุเคราะห์จากท่านอาจารย์ท่านเป็นอย่างสูงและพี่ภวินท์ แสงสุวรรณ ที่ให้ความช่วยเหลือและแนะนำการเขียนโปรแกรมบนไอโฟน

ขอขอบคุณสาขาวิชาวิศวกรรมคอมพิวเตอร์ที่ได้จัดเตรียมสิ่งอำนวยความสะดวก เพื่อให้การค้นคว้าและพัฒนาโปรแกรมเป็นไปได้ด้วยความสะดวกรวดเร็ว รวมถึงห้องให้บริการ สำหรับค้นคว้าและทำโครงงานนี้

ขอขอบคุณโรงเรียนโสตศึกษาทุ่งมหาเมฆ ที่ได้ให้ความร่วมมือและความอนุเคราะห์ให้เข้าไปทำแบบสอบถามและได้มอบหนังสือแบบเรียนสำหรับผู้พิการทางหู เพื่อนำผลไปปรับปรุงประสิทธิภาพการทำงานของระบบและข้อมูลที่เป็นประโยชน์ต่อการทำโครงงานในครั้งนี้

สุดท้ายนี้ต้องขอขอบพระคุณบุคคลที่สำคัญที่สุดในชีวิตของข้าพเจ้า คือ บิดา มารดา และบุคคลในครอบครัว ซึ่งได้เลี้ยงดู คอยสั่งสอนข้าพเจ้าเป็นอย่างดี พร้อมทั้งให้การศึกษาอย่างเต็มที่และให้กำลังใจ ความรักเสมอมา ข้าพเจ้าขอบขอบพระคุณไว้ ณ ที่นี้ด้วย

พีรวัส พลวิเศษ
พีระ ทั้งสุข
ภัทร นรเศรษฐ์โสภน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญตาราง VI

สารบัญภาพ VII

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญของโครงงาน 1

1.2 วัตถุประสงค์ของโครงงาน 1

1.3 ขอบเขตของโครงงาน 1

1.4 ขั้นตอนของการศึกษา 1

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง 3

2.1 ความรู้ทั่วไปเกี่ยวกับความบกพร่องทางการได้ยิน 3

2.2 ความรู้ทั่วไปเกี่ยวกับภาษามือไทย 7

2.3 การพัฒนาโปรแกรมบน iPhone 15

2.4 HTTP Protocol 23

2.5 Image Processing 27

2.6 โครงข่ายประสาทเทียม (Artificial Neural Network) 39

บทที่ 3 การออกแบบและพัฒนา 53

3.1 บทนำ (Introduction) 53

3.2 องค์ประกอบของระบบ 53

3.3 ยูสเคสไดอะแกรม (Use Case Diagram) 54

3.4 ซีเควนไดอะแกรม (Sequence Diagram) 55

3.5 คลาสไดอะแกรม (Class Diagram) 65

3.6 XML Schema 70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

3.7 การนำไปใช้ 71
3.8 User Interface 75

บทที่ 4 การทดลองและสรุปผลการทดลอง 85
4.1 การทดลอง Image Processing 85
4.2 ผลการทดลอง Neural Network 92

บทที่ 5 บทสรุป 93
5.1 สรุปผลการดำเนินงาน 93
5.2 ปัญหาและอุปสรรค 93
5.3 แนวทางการพัฒนาต่อ 93

บรรณานุกรม 94

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตาราง **หน้า**

2.1 ระดับความผิดปกติของการได้ยิน และลักษณะอาการ 4
2.2 ความแตกต่างของการสะกดนิ้วมือ กับภาษามือโดยสังเขป 11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูป	หน้า
2.1 แบบสะกดตัวอักษรภาษาไทย	12
2.2 แบบสะกดสระ วรรณยุกต์ และสัญลักษณ์อื่นๆ ในภาษาไทย	13
2.3 แบบสะกดตัวเลข	13
2.4 ตัวอย่างวิธีการสะกดนิ้วมือ	14
2.5 ภาพรวมลำดับชั้นของการเขียนโปรแกรม	15
2.6 หน้าตา Xcode เมื่อเปิดโปรแกรม	18
2.7 หน้าตา Xcode เมื่อสร้าง Project ขึ้นมา	19
2.8 หน้าตาส่วนของ Interface Builder	20
2.9 ส่วนที่แตกต่างระหว่าง Application บน Mac กับ iPhone ของ Xcode	20
2.10 Document Windows ใน Xcode	21
2.11 Document Windows ใน Xcode ในส่วนของ File's Owner	22
2.12 Document Windows ใน Xcode ในส่วนของ File's Owner(TestViesController)	22
2.13 iPhone Simulator	23
2.14 พิกัดของภาพดิจิตอล	27
2.15 ตาราง Template ที่ใช้กรอง	29
2.16 ปัญหาที่ของภาพ	30
2.17 การกำหนดดัชนีแบบสะท้อน	31
2.18 การกำหนดดัชนีแบบวนรอบ	31
2.19 ลักษณะการเกิด Noise บนภาพและการ Blur ภาพ	33
2.20 ภาพการกระจายแบบ Gaussian	34
2.21 กราฟการหาค่า Threshold	35
2.22 ตัวอย่างการทำ translation	36
2.23 ตัวอย่างการทำ rotation	36
2.24 ตัวอย่างการทำ Reflect	36
2.25 ตัวอย่างการทำ scaling	38
2.26 การทำ Sub-sampling	38
2.27 การทำการ Zoom	39
2.28 Model ของ Neuron ในสมองมนุษย์	40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูป หน้า

2.29 Model ของ Neuron ในคอมพิวเตอร์ 41

2.30 การแยกแยะระหว่างสี่เหลี่ยมและสามเหลี่ยม 42

2.31 โครงสร้างวงจร Neural Network 42

2.32 แสดงรูปแบบ Back-propagation Neural Network 43

2.33 Neural Network Taxonomy 44

2.34 การเรียนรูปแบบมีการสอน (Supervised Learning) 45

2.35 การเรียนรู้แบบไม่มีการสอน Unsupervised Learning 45

2.36 สถาปัตยกรรมของ Feedforward Network 46

2.37 สถาปัตยกรรมของ Feedback Network 46

2.38 Single-layer Perceptron 48

2.39 โครงสร้างของ Perceptrons 49

3.1 องค์ประกอบและการทำงานของระบบ 53

3.2 แผนภาพ Use case diagram ของระบบ 54

3.3 แผนภาพการแสดงภาษามือไทยจากรูป (Convert Picture To Sign)ของระบบ ส่วนที่1 55

3.4 แผนภาพการแสดงภาษามือไทยจากรูป (Convert Picture To Sign)ของระบบ ส่วนที่2 56

3.5 แผนภาพการแสดงภาษามือไทยจากรูป (Convert Picture To Sign)ของระบบ ส่วนที่3 57

3.6 แผนภาพเกมทายคำศัพท์ของระบบ 58

3.7 แผนภาพเกมทายคำศัพท์ของระบบ(ต่อ) 59

3.8 แผนภาพการจัดเก็บประวัติการถ่าย (Picture History) 60

3.9 แผนภาพการจัดเก็บประวัติการถ่าย (Picture History)(ต่อ) 61

3.10 แผนภาพการฝึกข่ายใยประสาทเทียมของระบบ (Neural Network Training) 62

3.11 แผนภาพการฝึกข่ายใยประสาทเทียมของระบบ (Neural Network Training)(ต่อ) 63

3.12 แผนภาพการแสดงคำศัพท์(Sign Language Performing) 64

3.13 Class Diagram ส่วน Server เกี่ยวกับการรับส่งรูปของระบบ 65

3.14 Class Diagram ส่วน Server เกี่ยวกับการทำนายของ Neural Networkของระบบ 66

3.15 Class Diagram ส่วน Server ในการจัดการ History และ Accountของระบบ 67

3.16 Class Diagram ส่วน iPhone ในการค้นหาคำศัพท์ 68

3.17 Class Diagram ส่วน iPhone ในการควบคุมการถ่ายรูป 69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูป	หน้า
3.18 Class Diagram ส่วน iPhone ในการควบคุมการแสดงคำที่เคยถ่ายไว้แล้ว	70
3.19 XML Schema ของ History แต่ละ User	71
3.20 XML Schema ของ Account แต่ละ User	71
3.21 กรอบรูปบน iPhone	71
3.22 ตัวอย่างภาพผลลัพธ์หลังการเปรียบเทียบความเข้มแสง	72
3.23 แสดงภาพ Smoothing ด้วย Median Filter	73
3.24 การตัดรูปที่อยู่ในกรอบสีแดง	73
3.25 ตัวอย่างภาพขอบภาพ	73
3.26 ภาพกรอบของวัตถุที่แท้จริง	74
3.27 thresholdingที่ใช้แยกแยะระหว่างวัตถุกับ background	74
3.28 รูปที่ผ่านการทำ Thresholding แล้ว	75
3.29 หน้าจอ iPhone เพื่อแสดง Icon ของโปรแกรม	76
3.30 หน้าจอ History View	76
3.31 หน้าจอ Tutorial View	77
3.32 หน้าจอการ Search โดย Search ด้วยกลุ่มของคำ	78
3.33 หน้าจอรูปของคำศัพท์	78
3.34 ภาษามือด้วย Animation 3D	79
3.35 Camera View หน้าจอแสดงกฎและความสามารถในการถ่ายรูป	79
3.36 Camera View ก่อนการถ่ายรูป	80
3.37 Camera View ที่มีปุ่มปรับแต่งเพิ่ม	81
3.38 หน้าจอการปรับแต่งภาพ	81
3.39 Camera View ปุ่มแสดงภาษามือและรูปที่ได้ทำการปรับแต่งไว้	82
3.40 รูปสิ่งของที่ถูกต้องหรือใกล้เคียง	83
3.41 หน้าจอแรกของเกม	84
3.42 หน้าจอหลักในการเล่นเกม	84
4.1 ตัวอย่างผลลัพธ์จากการปรับค่า threshold แบบที่ 1	86
4.2 ตัวอย่างผลลัพธ์จากการปรับค่า threshold แบบที่ 2	86
4.3 ผลลัพธ์ของภาพที่มีความเข้มแสงที่ขอบภาพ และที่ภาพวัตถุ เท่ากัน	86

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูป หน้า

4.4 ตัวอย่างภาพที่สามารถ detect วัตถุในภาพได้ 87

4.5 ตัวอย่างภาพที่ไม่สามารถ detect วัตถุในภาพได้ 87

4.6 ภาพทางซ้ายเป็นต้นฉบับ ทางขวาส่งไปให้ neural network เทรนเป็นภาพที่ไม่มี noise 88

4.7 ภาพที่นำมา predict ทางซ้ายเป็นภาพ input ทางขวาเป็นภาพที่ผ่านการทำ edge detection 88

4.8 การสร้าง file .dll ใน MATLAB 89

4.9 ผลการ add reference ใน visual studio เสร็จ สามารถเรียกใช้ function ใน MATLAB 89

4.10 การทำ smoothing ด้วย MATLAB โดยรูปมีขนาด 300x300 ใช้ตัวกรองขนาด 3x3 90

4.11 การทำ smoothing ด้วย OpenCV โดยรูปมีขนาด 300x300 ตัวกรองขนาด 31x31 90

4.12 กราฟ performance ของ tool ที่ใช้ในการทำ image processing 91

4.13 การเปรียบเทียบประสิทธิภาพของ Training Algorithm ที่ใช้สำหรับ Neural Network 92

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของโครงงาน

ในปัจจุบันสื่อทางโทรทัศน์การสื่อสารในด้านต่างๆได้และการเรียนรู้ได้ให้ความสำคัญต่อการเรียนรู้และความรู้ต่างๆสำหรับผู้พิการมากขึ้นโดยเฉพาะอย่างยิ่งสำหรับผู้พิการทางการได้ยินเช่นคนหูหนวกโดยอุปสรรคที่เกิดในการเรียนรู้ของผู้พิการทางหูคือการจดจำคำศัพท์ต่างๆโครงงานนี้จึงเป็นเหมือนสื่อที่ช่วยให้ผู้พิการทางหูได้ใช้เพื่อจดจำศัพท์ที่พบอยู่ในชีวิตประจำวันได้

1.2 วัตถุประสงค์ของโครงงาน

1) เพื่อเรียนรู้และศึกษาเทคโนโลยีต่างๆให้เข้าใจเพื่อนำมาพัฒนาระบบแล้วสามารถทำงานได้อย่างมีประสิทธิภาพ
2) เพื่อแสดงภาษามือไทยด้วยตัวละคร Graphic Animation3D (3D Graphic Animation)
3) เพื่อเป็นสื่อช่วยในการเรียนรู้ภาษามือของผู้พิการทางหู
4) เพื่อเป็นการส่งเสริมรูปแบบการใช้ภาษามือไทยให้เป็นที่ยอมรับและเข้าใจตรงกันมากยิ่งขึ้น
5) เพื่อเป็นการนำเทคโนโลยีไปใช้ให้เกิดประโยชน์แก่สังคมโดยเฉพาะกลุ่มคนพิการทางการได้ยิน

1.3 ขอบเขตของโครงงาน

1) แสดงภาษามือไทยด้วยตัวละคร Animation 3D
2) ระบบสามารถรับอินพุตในรูปของภาพของวัตถุที่เป็นนามธรรม แล้วแปลงเป็นภาษามือไทยได้
3) ระบบมีการจัดเก็บคำศัพท์ภาษามือไทยเป็นพจนานุกรมเพื่อใช้ในการแปลง

1.4 ขั้นตอนของการศึกษา

1) ทำการศึกษาค้นคว้าข้อมูลที่เกี่ยวข้องกับโครงงาน
2) ศึกษาทฤษฎีและการใช้ภาษามือไทย
3) ศึกษาทฤษฎีเกี่ยวกับการรู้จำของคอมพิวเตอร์และทฤษฎีการประมวลผลภาพของคอมพิวเตอร์
4) วิเคราะห์และออกแบบระบบโดยใช้ UML
5) ศึกษา และตัดสินใจในการเลือกภาษาเพื่อนำใช้ในการพัฒนาระบบ โดยโครงงานนี้เลือกใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาษา C#และ Library OpenCV

6) ศึกษาวิธีการพัฒนาโปรแกรมประยุกต์บนอุปกรณ์โทรศัพท์เคลื่อนที่ โดยเน้นไปที่ iPhone โดยศึกษาภาษา Objective-C

7) ศึกษาและออกแบบส่วนติดต่อกับผู้ใช้ (User Interface)

8) พัฒนาส่วนเก็บและจัดเตรียมภาพ

9) พัฒนาส่วนประมวลภาพอและทดสอบกับภาพตัวอย่าง

10) พัฒนาส่วนรู้จำภาพ และทดสอบกับภาพตัวอย่าง

11) พัฒนาส่วนแสดงผลตัวละครกับภาษามือ

12) รวมส่วนต่างของระบบ และทดสอบการใช้งานจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 ความรู้ทั่วไปเกี่ยวกับความบกพร่องทางการได้ยิน

2.1.1 ความรู้ทั่วไปสำหรับบุคคลที่มีความบกพร่องทางการได้ยิน

การศึกษาสำหรับบุคคลที่มีความบกพร่องทางการได้ยินในประเทศไทยเริ่มเมื่อปี พ.ศ. 2494 โดย ม.ร.ว.เสริมศรี เกษมศรีซึ่งสำเร็จการศึกษาจากวิทยาลัยกอลอเดท(Gallaudet College) ซึ่งเป็นวิทยาลัยทางศิลปศาสตร์แห่งแรกและแห่งเดียวสำหรับคนหูหนวก ม.ร.ว.เสริมศรี เกษมศรีได้เปิดหน่วยทดลองสอนคนหูหนวกขึ้นเป็นครั้งแรกที่โรงเรียนวัดโสมนัส การสอนคนหูหนวกในตอนนั้นเป็นการสอนโดยใช้ท่าภาษามือประกอบ ต่อมาคุณหญิงกมลาไกรฤกษ์ได้สำเร็จการศึกษาจากวิทยาลัยเดียวกันมาเป็นครูใหญ่ที่โรงเรียนสอนคนหูหนวกดุสิต(ปัจจุบันคือโรงเรียนเศรษฐเสถียร)และได้รวบรวมภาษามือขึ้นเป็นหนังสือภาษามือไทย เพื่อใช้สอนคนหูหนวกในประเทศไทย โดยให้มีการสอนพูดรวมกับการใช้ภาษามือและการสะกดนิ้วมือร่วมกับการอ่านและการเขียนตามปกติ ปัจจุบันการจัดการศึกษาสำหรับบุคคลที่มีความบกพร่องทางการได้ยินในประเทศไทย มี 2 แบบคือ

1) การจัดการศึกษาพิเศษ ตามหลักสูตรของกระทรวงศึกษาธิการตั้งแต่ระดับชั้นเด็กเล็กประถมศึกษา และมัธยมศึกษาใช้หลักสูตรของกระทรวงศึกษาธิการโดยมีการปรับปรุงให้เหมาะสมเพียงเล็กน้อย

2) จัดการเรียนร่วมกับคนปกติ จัดให้เข้าเรียนในโรงเรียนเดียวกับเด็กปกติโดยมีครูพิเศษช่วยเหลือในด้านภาษาและการสื่อสาร

2.1.2 ความหมายของบุคคลที่มีความบกพร่องทางการได้ยิน

บุคคลที่มีความบกพร่องทางการได้ยินหมายถึงบุคคลที่มีความสามารถในการรับฟังเสียงต่างๆบกพร่องหรือสูญเสียไปเป็นเหตุให้ได้ยินเสียงต่างๆไม่ชัดเจนในกรณีที่สูญเสียการได้ยินเพียงเล็กน้อย เรียกว่า หูตึงและไม่ได้ยินเสียงพูดในกรณีที่สูญเสียการได้ยินมาก เรียกว่า หูหนวก

2.1.3 ประเภทของบุคคลที่มีความบกพร่องทางการได้ยิน

เด็กหูหนวก หมายถึงเด็กที่ไม่สามารถได้ยินเสียงแม้เสียงจะมีระดับความดังของเสียงมากกว่า 90 เดซิเบล วัดด้วยเสียงบริสุทธิ์ ณ ความถี่ 500,1000และ2,000 Hz ในหูข้างที่ดีกว่าเด็กไม่สามารถใช้การได้ยินให้เป็นประโยชน์ได้อย่างเต็มประสิทธิภาพอาจเป็นผู้ที่สูญเสียการได้ยินมาแต่กำเนิดหรือสูญเสียการได้ยินภายหลังก็ตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เด็กหูตึง หมายถึงเด็กที่สูญเสียการได้ยินที่ระดับเสียงระหว่าง 26-90 เดซิเบล วัดด้วยเสียงบริสุทธิ์ ณ ความถี่ 500,1000 และ2,000 เป็นเด็กที่สูญเสียการได้ยินเล็กน้อยไปจนถึงการสูญเสียการได้ยินขั้นรุนแรง

ตาราง 2.1 ระดับความผิดปกติของการได้ยิน และลักษณะอาการ

ระดับการได้ยิน	ค่าเฉลี่ยของระดับเสียงที่ไม่สามารถได้ยิน	ลักษณะอาการ
ระดับที่ 1 หูปกติ	0 - 25 เดซิเบล	ได้ยินเสียงพูดกระซิบเบาๆ
ระดับที่ 2 หูตึงเล็กน้อย	26 - 40 เดซิเบล	ไม่ได้ยินเสียงพูดเบาๆ แต่ได้ยินเสียงพูดปกติอาจใช้เครื่องช่วยฟังบางโอกาส เช่น เรียนหนังสือ
ระดับที่ 3 หูตึงปานกลาง	41 - 55 เดซิเบล	ไม่ได้ยินเสียงปกติ ต้องพูดดังกว่าปกติจึงจะได้ยินจำเป็นต้องใช้เครื่องช่วยฟังขณะพูดคุย
ระดับที่ 4 หูตึงมาก	56 - 70 เดซิเบล	พูดเสียงดังแล้วยังไม่ได้ยินจำเป็นต้องใช้เครื่องช่วยฟังตลอดเวลา
ระดับที่ 5 หูตึงรุนแรง	71 - 90 เดซิเบล	ต้องตะโกนหรือใช้เครื่องขยายเสียงจึงจะได้ยินแต่ได้ยินไม่ชัด
ระดับที่ 6 หูหนวก	มากกว่า 90 เดซิเบล	ตะโกนหรือใช้เครื่องขยายเสียงแล้วยังไม่ได้ยินและไม่เข้าใจความหมาย

ลำดับขั้นของกระบวนการฟังของมนุษย์อาจแบ่งได้เป็น 5 ระดับดังนี้

1) ขั้นได้ยิน (Hearing) เป็นการฟังในระดับต้นซึ่งมนุษย์สามารถกระทำได้ง่ายโดยรับฟังหรือได้ยินเสียงแม้เสียงนั้นจะเป็นเสียงที่ไม่เคยได้ยินมาก่อนก็รับฟังได้
2) ขั้นรับรู้และแยกเสียง (Matching) เป็นขั้นของการรับรู้และแยกเสียงที่ได้ยินว่ามีความเหมือน ความแตกต่างกัน โดยใช้ความสามารถของผู้ฟังจับเทียบเข้าคู่หรือเทียบเคียงเสียงที่ได้ยินว่าเหมือนกัน หรือแตกต่างกัน
3) ขั้นตีความ (Interpreting) เป็นขั้นที่ผู้ฟังแปลความหมายหรือตีความหมายของข้อความประโยคหรือสิ่งที่ได้ยินได้ฟังเป็นขั้นของการพยายามเจตนาอันแท้จริงของผู้พูด
4) ขั้นเข้าใจ (Understanding) เป็นขั้นการฟังซึ่งผู้ฟังสามารถเข้าใจความหมายของใจความสำคัญของผู้พูดได้อย่างถ่องแท้
5) ขั้นเชื่อ (Believing)เป็นขั้นที่ขึ้นอยู่กับความสามารถของผู้ฟังที่จะตัดสินว่าประโยค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือสิ่งที่ได้ยินมานั้นมีความจริงเพียงใดเชื่อถือได้อย่างไรและยอมรับได้หรือไม่

6) ขั้นตอบสนอง (Reaction) เป็นขั้นของการตอบสนองกลับด้วยวัจนภาษาหรืออวัจนภาษา ได้แก่ การพยักหน้า ยิ้มหัวเราะ หน้าบึ้ง เป็นต้น

จากลำดับขั้นของกระบวนการฟังดังกล่าวสามารถสรุปเป็นระดับของการฟังที่มักใช้ในชีวิตประจำวัน ได้เป็น 3 ระดับ ดังนี้

1) ระดับการได้ยิน การได้ยินเป็นกระบวนการขั้นแรกของการฟังเป็นการรับรู้โดยใช้กลไกของสมรรถภาพทางกาย ได้แก่ อวัยวะในการรับรู้หรือการได้ยินคือ หู และอวัยวะภายในหูเมื่อหูรับคลื่นเสียงแล้วก็จะส่งไปยังสมองสมองจะรับรู้ว่า เรื่องที่ได้ยินนั้นคืออะไรโดยไม่มีการแสดงปฏิกิริยาตอบสนอง

2) ระดับการฟังตามปกติ เป็นระดับการได้ยินที่มีประสิทธิภาพสูงขึ้นต่อเนื่องจากระดับการได้ยิน ผู้ฟังต้องอาศัยสมรรถภาพทางร่างกายทางสมองและจิตใจเชื่อมโยงเสียงที่ได้ยินกับประสบการณ์ และความรู้เกี่ยวกับความหมายของเสียงและตีความจนเข้าใจสารที่ฟังและแสดงปฏิกิริยาตอบสนองสารนั้นอย่างถูกต้องและเหมาะสม

3) ระดับการฟังอย่างมีวิจารณญาณเป็นระดับการฟังที่สูงขึ้นอีกต้องอาศัยสมรรถภาพทางด้านการคิดวิเคราะห์การประเมินค่า การวินิจฉัยและการนำไปใช้ในชีวิตจริงได้ การฟังระดับนี้ต้องอาศัยการฝึกฝนและพัฒนาอย่างต่อเนื่องหากสามารถพัฒนาจนเกิดทักษะแล้ว ผู้รับสารจะได้ประโยชน์สูงสุดจากการฟังแต่ละครั้งดังแผนภูมิต่อไปนี้

2.1.4 สาเหตุของความพิการทางการได้ยิน

1) กรรมพันธุ์

2) ความผิดปกติขณะตั้งครรภ์ เช่น

2.1)พิษจากยา หรือสารบางประเภทที่ได้รับ

2.2) ขาดสารอาหาร

2.3) เลือดแม่กับไม่เข้ากัน

2.4) อุบัติเหตุที่กระทบกระเทือนการตั้งครรภ์

3) ความผิดปกติขณะคลอด เช่น คลอดยาก คลอดก่อนกำหนด ท่าคลอดของเด็กผิดปกติ ใช้เครื่องช่วยคลอด หรือเด็กขาดออกซิเจน เป็นต้นความผิดปกติที่เกิดขึ้นทั่วไป เช่น

3.1) การแพ้พิษยา หรือ สารบางชนิด เช่น ยาปฏิชีวนะเป็นต้น

3.2) การติดเชื้อ เช่น เยื่อหุ้มสมองอักเสบเป็นต้น

3.3) ความผิดปกติที่เกิดขึ้นภายในหู เช่นมีเนื้องอกในหู หูชั้นในอักเสบเป็นต้น

3.4) ได้ยินเสียงอึกทึกดังติดต่อกันเป็นเวลานาน เช่นเสียงปะทัด และเสียงเครื่องจักร เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5) อุบัติเหตุที่กระทบกระเทือนบริเวณหู หรือประสาทหู

2.1.5 การรักษา

ปัจจุบันยังไม่มีวิธีการใดๆ รักษาประสาทหูเสียได้ยกเว้นคนที่ประสาทหูเสียภายหลังอย่างเฉียบพลันและได้รับการรักษา จากแพทย์ทันท่วงที

2.1.6 การป้องกัน

1) หญิงมีครรภ์ควรปฏิบัติดังนี้ฝากครรภ์กับแพทย์ และรับการตรวจตามกำหนด

1.1) รักษาสุขภาพ รับประทานอาหารที่มีประโยชน์

1.2) หลีกเลี่ยงการฉายแสงเอกซเรย์

2) หญิงทุกคนควรฉีดวัคซีนป้องกันหัดเยอรมันเพื่อจะได้ไม่เป็นโรคนี้ขณะตั้งครรภ์

3) หญิง-ชาย ควรตรวจเลือดก่อนแต่งงาน

4) ห้ามซื้อยากินใช้เอง

5) หากพบความผิดปกติเกี่ยวกับการได้ยินควรปรึกษาแพทย์ทันที

6) เด็กแรกเกิดควรได้รับการฉีดวัคซีนภูมิคุ้มกันตามที่แพทย์กำหนด

7) หลีกเลี่ยงการฟังเสียงอึกทึกดังมากเกินไป

2.1.7 หลักการระวังรักษาหูอย่างง่ายๆ

1) อย่าแคะหูหรือนำสิ่งใดเข้าไปในหู

2) ถ้าต้องการทำความสะอาดหูควรใช้น้ำเช็ดเฉพาะรูหูตอนนอกเท่านั้น

3) ป้องกันการกระทบกระเทือนอย่างแรงๆบริเวณหู คอและศีรษะ

4) ถ้าเด็กเอาสิ่งของใส่ในหูอย่าเขี่ยออกเอง ควรพบแพทย์

5) เมื่อน้ำเข้าหูเพียงตะแคงหูไว้น้ำจะไหลออกมาเอง

6) เมื่อมีมดหรือแมลงเข้าหูให้ใช้น้ำมันเหลว หยอดในรูหูตะแคงหูนั้นขึ้นทิ้งไว้สักครู่ มดหรือแมลงจะตายและหลุดออกมาเอง

2.1.8 ข้อแนะนำในการช่วยเหลือและปฏิบัติต่อคนหูหนวก

1) ปฏิบัติต่อคนหูหนวกในฐานะเป็นมนุษยชนเท่าเทียมกับคนทั่วไป

2) ให้โอกาสคนหูหนวกแสดงความคิดเห็นและแสดงฝีมือในสิ่งที่เขาถนัด

3) ถามคนหูหนวกว่าจะติดต่อสื่อสารด้วยวิธีใดจึงจะดีที่สุด

4) ควรเตรียมกระดาษและปากกาให้พร้อมที่จะใช้ในการอ่าน-เขียนสื่อสารกับคนหูหนวก

5) เรียนและใช้ภาษามือสื่อสารกับคนหูหนวก

6) ควรให้เวลาและทำตัวตามสบายในการคุยกับคนหูหนวก

7) ควรมองตรงไปที่คนหูหนวกตลอดเวลาขณะที่พูด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8) สนับสนุนให้คนหูหนวกได้รับการศึกษาให้มากที่สุด

9) ควรสนับสนุนการใช้ความสามารถของคนหูหนวก

10) ไม่ควรบังคับให้คนหูหนวกพูดเปล่งเสียง

11) ไม่ควรล้อเลียนหรือขบขันเมื่อเห็นคนหูหนวกใช้ภาษามือ

12) ถ้าไม่เข้าใจภาษามือควรบอกคนหูหนวก

13) ไม่ควรอมอะไรในปากหรือปิดปากขณะพูดกับคนหูหนวก

14) ควรใช้สัญญาณอันตราย หรือ สัญญาณต่างๆที่เป็นแสงสัญญาณแทนสัญญาณเสียง เพื่อให้คนหูหนวกรับรู้

2.2 ความรู้ทั่วไปเกี่ยวกับภาษามือไทย

ภาษามือ เป็นอวัจนภาษาอย่างหนึ่งที่ใช้ในการสื่อสารของผู้พิการทางการได้ยิน ประกอบด้วย การสื่อสารด้วยมือ การสื่อสารด้วยร่างกาย และการใช้สีหน้า อารมณ์ในการถ่ายทอดแทนการใช้เสียงพูดเหมือนคนปกติทั่วไป เพราะผู้บกพร่องทางการได้ยินจะไม่สามารถได้ยินเสียงพูดเหมือนคนปกติ จึงไม่สามารถพูดได้ แต่สายตาของผู้บกพร่องทางการได้ยินนั้นสามารถมองเห็นได้เป็นปกติ จึงสังเกตกิริยาอาการ ท่าทางต่าง ๆ เพื่อเรียนรู้ความหมายของกิริยาอาการท่าทางต่าง ๆ ซึ่งภาษามือของแต่ละชาติมีความแตกต่างกันขึ้นกับขนบธรรมเนียมประเพณี วัฒนธรรม และลักษณะภูมิศาสตร์ เช่น ภาษามือจีน ภาษามืออเมริกัน และภาษามือไทย เป็นต้น ภาษามือเป็นภาษาที่นักการศึกษาทางด้านการศึกษาของผู้บกพร่องทางการได้ยินได้ตกลงและยอมรับกันแล้วว่า เป็นภาษาหนึ่งสำหรับติดต่อสื่อความหมายระหว่างผู้บกพร่องทางการได้ยินด้วยกัน และระหว่างคนปกติกับผู้บกพร่องทางการได้ยินในภาษา อังกฤษเรียกว่า “Sign Language” หรือ “Manual Communication” ภาษามือไทยเป็นภาษาสัญลักษณ์สำหรับบุคคลที่มีความบกพร่องทางการได้ยิน หรือที่เรียกกันว่า "คนหูหนวก" โดยลักษณะการใช้ภาษามือในประเทศไทย ได้รับอิทธิพลมาจาก "ภาษาธรรมชาติ" และ "ภาษามือสไตล์อเมริกัน"

2.2.1 รูปแบบของการใช้ภาษามือ

ในปัจจุบันภาษามือมีการแบ่งรูปแบบตามลักษณะการใช้ออกเป็น 3 ประเภทดังนี้

1) ภาษามือสากล หมายถึง ภาษามือที่คนทั่วไปเข้าใจ

2) ภาษามือผู้พิการ หมายถึง ภาษามือที่ใช้กันระหว่างผู้พิการทางหู

3) ภาษามือทางทหาร หมายถึง ภาษามือที่ใช้ในกลุ่มทหารที่ปฏิบัติหน้าที่อยู่ในสถานที่ที่ไม่สามารถ สื่อสารกันด้วยเสียงพูดหรือวิธีอื่นได้

2.2.2 ประเภทของภาษามือ

ในปัจจุบันภาษามือมีการแบ่งรูปแบบตามที่มาของภาษามือเป็น 2 ประเภทดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2.1 ภาษามือธรรมชาติ (Sign Language)

เป็นภาษาที่ผู้บกพร่องทางการได้ยินพยายามเลียนแบบธรรมชาติจึงมีความแตกต่างกันของแต่ละชุมชน หรือประเทศชาติ ซึ่งเป็นภาษาที่เข้าใจได้ไม่ยาก เพราะเป็นท่าทางพื้นฐานของคนปกติอาจจะมีการใช้สีหน้า ท่าทางประกอบ เช่น American Sign Language, British Sign Language, Swedish Sign Language

2.2.2.2 ภาษามือประดิษฐ์ (Signed)

ภาษามือประดิษฐ์ (Signed) คือ ภาษามือที่ครู ผู้ปกครอง หรือญาติมิตร ของผู้บกพร่องทางการได้ยินได้พยายามคิดค้น เนื่องจากในการสื่อสารกันโดยปกตินั้นอาจจะมีคำเฉพาะที่ไม่สามารถสื่อสารออกมาเป็นท่าทางปกติตามธรรมชาติได้ เช่น ชื่อของประเทศต่าง ๆ เป็นต้น จึงมีการคิดค้นโดยการกำหนดท่าทางภาษามือที่ใช้แทนพยัญชนะและสระที่มีในภาษาต่าง ๆ เมื่อนำท่าทางของพยัญชนะและสระนั้นมาสะกดให้เกิดเป็นคำที่มีความหมายที่เข้าใจได้ โดยมีการนำแบบสะกดนิ้วมือ (Finger Spelling) มาประสมด้วยเช่น ดีใจ ย่า ยาย ประชาชน (คน+ป) พลเมือง (คน+พ)

2.2.3 ลักษณะของภาษามือ

ภาษามือ เป็นภาษาท่าทางซึ่งมีการเคลื่อนไหวของมือเป็นหลักและใช้กิริยาอาการของหน้าตาและร่างกายส่วนหนึ่ง เป็นส่วนประกอบช่วยให้เกิดความเข้าใจ ท่าภาษามือที่คนหูหนวกยอมรับจะต้องเป็นท่าที่ทำง่าย สะดวก รวดเร็ว มีความหมายใกล้เคียงธรรมชาติและเหมาะกับสรีระศาสตร์ ท่าภาษามือควรทำอย่างมีจังหวะ คือ มีการเว้นระยะ ไม่ทำท่าทางจนเร็วเกินไปและให้อยู่ในรัศมีที่สายตาสามารถมองเห็นได้ชัดเจน

ในท่าภาษามือการแสดงสีหน้าและการเคลื่อนไหวของใบหน้า เช่น คิ้ว ปาก เป็นสิ่งสำคัญช่วยให้เข้าใจความหมายในภาษามือชัดเจนยิ่งขึ้น เช่น

1) การส่ายศีรษะ หมายถึง การปฏิเสธ

2) การขมวดคิ้ว หมายถึง การแสดงความสงสัย

3) การเลิกคิ้ว หมายถึง การแสดงคำถามที่ต้องการคำตอบ

<u>หมายเหตุ</u> การแสดงสีหน้าและการเคลื่อนไหวของใบหน้าประกอบท่ามือควรทำแต่พองาม

2.2.4 ที่มาของภาษามือ

ที่มาของภาษามือไทยแบ่งออกเป็น 3 แหล่งที่มาดังนี้

2.2.4.1 จากชุมนุมคนหูหนวก

แต่ละอาชีพจะมีภาษาที่จำเป็น คำศัพท์ภาษามือที่ได้มาจึงเป็นคำที่ใช้อยู่และคนปกติที่เกี่ยวข้องเข้าใจเมื่อนำคำศัพท์มารวมกันจะได้คำที่จำเป็นต่อการดำรงชีวิตเพิ่มขึ้น เช่นช่างตัดเสื้อมีคำว่า กรรไกร จักร เข็ม ช่างทาสีมีคำว่า ทาสี สีแดง สีขาว ฯลฯ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4.2 จากนักวิจัยภาษามือ

ซึ่งเป็นกลุ่มคนหูหนวกที่รวบรวมภาษามือมาพิจารณาว่าใช้ต่างกันหรือเหมือนกัน (ภาคเหนือ ใต้ กลาง อีสาน) ท่ามือเหล่านั้นต้องไม่ขัดต่อประเพณีวัฒนธรรมของชาติการวิจัยจำเป็นต้องมีคนปกติที่มีความรู้เรื่องภาษามือและภาษาไทยอย่างดีเป็นผู้ช่วยเหลือ

2.2.4.3 จากครูโรงเรียนสอนคนหูหนวก

เนื่องจากอยู่ใกล้ชิดกับนักเรียน การได้สนทนาให้คำปรึกษาระหว่างครู-นักเรียน ผู้ปกครอง ทำให้ครูเข้าใจภาษามือของนักเรียน

2.2.5 โครงสร้างของภาษามือ

โครงสร้างภาษามือไทย คือ ตำแหน่ง ท่าทาง ที่มีความสำคัญต่อการให้ความหมายของคำแต่ละคำโดยแบ่งเป็น 4 ส่วนดังนี้

2.2.5.1 ท่ามือ (The Hand Shape)

เป็นการทำมือในท่าต่าง ๆ เช่น กำมือ แบมือ กางนิ้ว รวมนิ้ว จีบนิ้ว ฯลฯ

2.2.5.2 ตำแหน่งของมือ (The Positions of The Hands)

ตำแหน่งของมือ (The Positions of The Hands) ตำแหน่งที่ทำท่ามือควรจะอยู่ในระดับที่มองเห็นได้ง่าย และชัดเจน คือ บริเวณศีรษะ ใกล้ใบหน้า และไม่ควรต่ำกว่าระดับเอว ท่ามือท่าเดียวกันแต่ตำแหน่งของมือที่แตกต่างกัน จะให้ความหมายที่ต่างกัน เช่น ใช้นิ้วชี้ชี้ที่หน้าอกหมายถึง “ฉัน” ถ้าชี้ที่ขมับ หมายถึง “รู้” ภาษามือสามารถแสดงถึงความรู้สึกต่าง ๆ โดยการแสดงท่ามือในตำแหน่งใกล้เคียงกับความหมายของคำนั้น ๆ เช่น

1) ท่ามือบริเวณศีรษะจะเกี่ยวกับความคิด เช่น รู้ ฝัน ฉลาด

2) ท่ามือบริเวณอกจะเกี่ยวกับความรู้สึก เช่น รัก เสียใจ ขอบคุณ

3) ท่ามือบริเวณลำตัวจะเป็นคำทั่ว ๆ ไป เช่น ลูก ซักผ้า รองเท้า

2.2.5.3 การเคลื่อนไหวของมือ (The Movement of The Hands)

การเคลื่อนไหวของมือ (The Movement of The Hands) ในการแสดงท่าทางนั้นจะเลือกใช้มือตามความถนัดของแต่ละคน ถ้าต้องใช้มือทั้งสองข้างจะใช้มือข้างที่ถนัดในการแสดงท่าทางที่ยากกว่า และใช้มืออีกข้างหนึ่งที่ถนัดน้อยกว่าในการแสดงท่าทางที่ง่ายกว่า ท่ามืออย่างเดียวกัน แต่เคลื่อนไหวไปในทิศทางต่างกัน ความหมายจะแตกต่างกัน เช่น มือทั้งสองตั้งขึ้น หัวแม่มือชิดกันแล้วเลื่อนออกห่าง คือ “เปิด” แต่ถ้ามือห่างกันพอควร แล้วเลื่อนให้หัวแม่มือชิดกันหมายถึง “ปิด” การแสดงท่าทางภาษามือนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.5.4 ทิศทางของฝ่ามือ (The Orientation of the Palms in Relationship to the Body or to each other)

ส่งผลทำให้ท่ามือมีความหมายแตกต่างกัน เช่น ท่ามือท่าเดียวกัน ตำแหน่งเดียวกัน แต่ทิศทางของฝ่ามือต่างกัน ความหมายจะต่างกัน เช่น ตั้งมือขึ้น นิ้วชิดกัน หันฝ่ามือออก ยื่นไปข้างหน้า หมายถึง "ของเขา" แต่ถ้าหันฝ่ามือเข้าหาตัว หมายถึง "ของฉัน"

การทำท่ามือ ถ้าอยู่ในระดับสายตามองเห็นได้ง่าย เช่น บริเวณใบหน้ามักจะทำมือเดียว เช่น สวย อิ่ม แต่ถ้าระดับต่ำกว่าอกจะทำสองมือ เพื่อให้เห็นชัดเจนยิ่งขึ้น เช่น กระโปรง ถ้าทำท่าสองมือจะต้องเคลื่อนมือใดมือหนึ่ง มือที่ถนัดจะเป็นมือที่เคลื่อนไหว มือที่ไม่ถนัดจะทำง่ายกว่า หรืออยู่นิ่งๆเป็นฐาน เพราะมนุษย์ไม่สามารถทำงานได้พร้อมกันทั้งมือซ้ายและมือขวาด้วยท่าทางที่แตกต่างในเวลาเดียวกันได้โดยง่าย

จากที่กล่าวมาเป็นวิธีการที่ทำให้เราสามารถใช้ภาษามือในการสื่อสาร ดังนั้นภาษามือเป็นภาษาสำหรับคนหูหนวกที่ใช้ในการติดต่อสื่อสารเพื่อให้เกิดความเข้าใจตรงกัน แต่ต้องคำนึงถึงสถานการณ์ที่กำลังสนทนากัน ถ้าภาษามือเหมือนกันแต่ละสถานการณ์ความหมายก็อาจจะต่างกันได้ ดังนั้นควรที่จะรู้ความหมายของท่าภาษามือนั้นว่าความหมายคืออะไร เพื่อจะได้เข้าใจตรงกัน ทำให้การสื่อสารมีประสิทธิภาพมากขึ้น

2.2.6 การสะกดตัวอักษรด้วยนิ้วมือ (Fingerspelling)

การสะกดตัวอักษรด้วยนิ้วมือ คือ การแสดงท่าทางด้วยนิ้วมือแทนตัวอักษรแต่ละตัวตามหลักการสะกดคำภาษาไทย

2.2.6.1 ความหมายของการตัวอักษรด้วยนิ้วมือ

การสะกดนิ้วมือ คือการที่บุคคลทำท่าทางด้วยนิ้วมือเป็นรูปต่างๆ แทนตัวพยัญชนะ สระ วรรณยุกต์ ตลอดจนสัญลักษณ์อื่นๆ ของภาษาประจำชาติเพื่อการสื่อภาษา หรือการเขียนในอากาศนั้นเอง โดยทั่วไปแล้วตัวอักษรที่สะกดนิ้วมือ (Manual Alphabet) ของภาษาใดจะมีจำนวนเท่ากับตัวอักษรของภาษานั้นๆ

2.2.6.2 ความแตกต่างของการสะกดนิ้วมือ และภาษามือ

ความแตกต่างของการสะกดนิ้วมือ และภาษามือ มีบุคคลเป็นจำนวนมากที่ไม่คุ้นเคยกับการศึกษาของคนหูหนวก โดยเฉพาะอย่างยิ่งการสอนคนหูหนวก โดยปรัชญาการสอนระบบรวม (Total Communication) ทั้งนี้อาจเป็นเพราะวิธีการสื่อภาษาทั้งสองแบบนี้ใช้มือในการทำท่าทางเหมือนกันนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 2.2 ความแตกต่างของการสะกดนิ้วมือ กับภาษามือโดยสังเขป

การสะกดนิ้วมือ (Finger Spelling)	ภาษามือ (Sign Language)
1.ใช้แทนตัวพยัญชนะ สระ วรรณยุกต์ ตลอดจนสัญลักษณ์อื่น ๆ ของภาษาประจำชาติ	1.เป็นท่ามือที่บอกความหมาย มีลักษณะเป็นคำวลี หรือประโยค
2.ต้องสะกดนิ้วมือเป็นพยัญชนะ สระวรรณยุกต์ตามหลักการเขียน	2.ทำตามหลักภาษามือไทย
3.ใช้เวลามากกว่าภาษามือ เช่น คำว่า "โรงแรม" ผู้ทำจะต้องสะกดตัวอักษรทีละตัวเช่นเดียวกับการพิมพ์ดีด คือ โ-ร-ง-แ-ร-ม	3.ใช้เวลาน้อยกว่า การสะกดนิ้วมือในคำเดียวกัน เช่น โรงแรม
4.หลักในการเขียน เมื่อเขียนตัวอักษรแต่ละตัวให้ขีดระหว่างตัวอักษรนั้น เช่น แ-ม-ว ภาษาอังกฤษจะเขียนเป็น C-A-T	4.ใช้การขีดเส้นใต้ คำที่ต้องการที่จะให้ผู้อ่านทราบว่าคำนั้นจะต้องทำท่าภาษามือ เช่น แมว (ในภาษาอังกฤษ จะใช้ตัวอักษรพิมพ์ใหญ่แทนการขีดเส้นใต้ เช่น CAT)
5.ใช้ร่วมกับภาษามือประจำชาติ ได้แก่ชื่อเฉพาะ เช่น ชื่อคน ชื่อสถานที่ และศัพท์เทคนิคทางวิชาการต่างๆ เป็นต้นภาษามือประดิษฐ์ที่เป็นคำศัพท์ใหม่ ๆ ที่มีท่าสะกดนิ้วมือประสมค่อนข้างมาก เช่น กฎ ระเบียบ วินัย	5.ภาษามือธรรมชาติ ที่ใช้ในระหว่างผู้บกพร่องทางการได้ยิน กับผู้บกพร่องทางการได้ยินด้วยกัน จะมีท่าสะกดนิ้วมือน้อยส่วน
6.วิธีสอนผู้บกพร่องทางการได้ยิน โดยใช้การสะกดนิ้วมือพร้อมกับการพูดเรียกว่า วิธีสอนแบบโรเชสเตอร์ (Rochester Method)	6.วิธีสอนผู้บกพร่องทางการได้ยิน โดยใช้ภาษามือพร้อมกับการพูดเรียกว่า วิธีสอนแบบผสม (มือ+พูด) (Simultaneous Method)

2.2.6.3 การสะกดนิ้วมือไทย

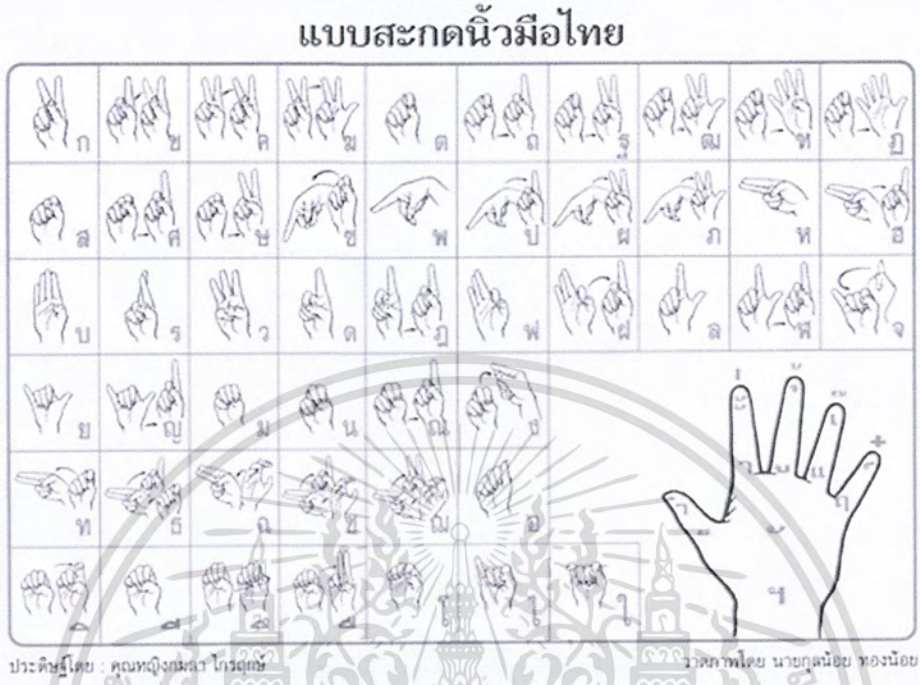
การสะกดนิ้วมือไทยแบ่งออกเป็น 3 ส่วนดังนี้

2.2.6.3.1 แบบสะกดตัวอักษร

การ เรียงลำดับของอักษรไทย สระ วรรณยุกต์ในภาษามือจะแตกต่างกับการเรียงอักษรปรกติ โดยอักษรภาษามือหรือแบบสะกดนิ้วมือไทยได้เรียงลำดับตัวอักษรตามลักษณะท่ามือ ที่มีลักษณะคล้ายคลึงกันเช่น ก ข ค และฆ ส่วนอักษรตัวต่อไปเป็นตัว ต ถ ฐ ฒ ฑ ฏ (ทั้ง 6 ตัวนี้มีลักษณะท่ามือคล้ายกัน) ตัว ฮ ซึ่งเป็นตัวอักษรสุดท้ายของตนปรกติแต่สำหรับคนหูหนวกตัว ฮ จะอยู่ในกลุ่มของ ห เป็นต้นดังนั้นอักษรตัวสุดท้ายของแบบสะกดนิ้วมือไทยคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัว อ ตัวสะกดนิ้วมือของคนหูหนวกไทยไม่มีตัว ฃ ฅ เนื่องจากคนหูหนวก พบเห็นตัวอักษร 2 ตัวนี้น้อยมากโดยลักษณะการสะกดตัวอักษรภาษาไทยด้วยนิ้วมือทั้งหมด มีดังรูป 2.1



รูป 2.1 แบบสะกดตัวอักษรภาษาไทย

2.2.6.3.2 แบบสะกดนิ้วมือ

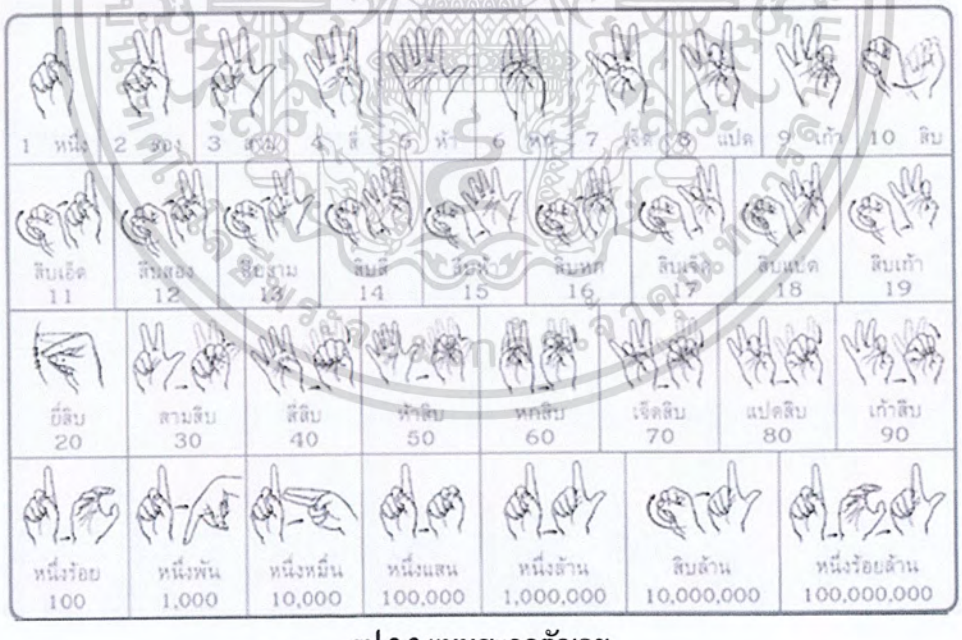
สระ วรรณยุกต์ และสัญลักษณ์อื่นๆซึ่งได้จัดเรียงตามลักษณะที่คนหูหนวกเห็นและใช้อยู่ตามปรกติซึ่งไม่เหมือนกับการเรียงลำดับของสระในภาษาไทยดูได้จากรูป 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 2.2 แบบสะกดสระ วรรณยุกต์ และสัญลักษณ์อื่นๆ ในภาษาไทย

2.2.6.3.3 แบบสะกดตัวเลข

ซึ่งมีตั้งแต่ 1-10 และตัวเลขหลักสิบ ถึง หลักล้านตามลำดับดังรูป 2.3



รูป 2.3 แบบสะกดตัวเลข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะสำคัญของการสะกดนิ้วมือไทยคือ การเคลื่อนไหวต่างๆ สัญลักษณ์ที่แสดงการเคลื่อนไหวเพื่อแสดงลำดับของของการทำท่าอักษรสะกดนิ้วมือโดยใช้ลูกศร(⟶) โดยกำหนดให้โคนอักษรเป็นท่าที่ 1 และปลายศรลูกศรเป็นท่าถัดไป ส่วนภาพที่เป็นเส้นประ (.....) หมายถึงการทำท่าเป็นลำดับแรกดังรูป 2.4

รูป 2.4 ตัวอย่างวิธีการสะกดนิ้วมือ

การ สะกดนิ้วมือ ผู้ใช้ควรแสดงอาการให้งดงามโดยไม่ยกแขนเกะกะ ให้มีจังหวะคล้ายการพูดที่มีการเว้นวรรคตอน มีการเว้นระยะหายใจไม่ทำเร็วจนเกินไป หากในประโยคที่แสดงอาการนั้นจะต้องใช้คำที่มีการสะกดนิ้วมือร่วมด้วย ผู้ใช้ภาษามือควรสะกดคำให้ช้าและถูกต้อง เพื่อความชัดเจนและให้คู่สนทนาได้มีเวลาอ่านทันด้วย

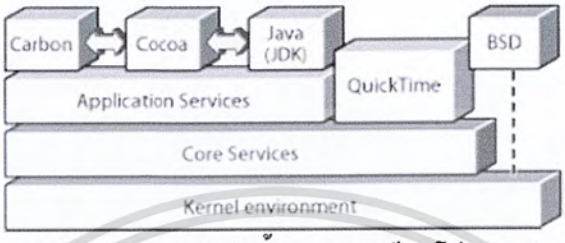
คน หูหนวกส่วนใหญ่ใช้การสะกดนิ้วมือ เมื่อกล่าวถึง บุคคล ชื่อเฉพาะ ชื่อย่อและสถานที่ ที่ไม่คุ้นเคย ทั้งนี้เพื่อความถูกต้องและชัดเจน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การพัฒนาโปรแกรมบน iPhone

2.3.1 Framework

การเขียนโปรแกรมบนแพลตฟอร์ม Macintosh นั้นมี Framework เพื่อให้นักพัฒนาได้ใช้อยู่ด้วยกัน 2 Framework ใหญ่ๆคือ Carbon และ Cocoa เหตุผลที่ต้องมีถึง 2 Framework นั้นก็เพราะว่า Carbon เป็น Framework สำหรับภาษา C/C++ และ Cocoa เป็น Framework สำหรับภาษา Objective-C



รูป 2.5 ภาพรวมลำดับชั้นของการเขียนโปรแกรม

โดยปกติแล้วไม่ว่าจะเขียนด้วย Cocoa หรือ Carbon สอง Framework จะมี Class ที่มีความคล้ายคลึงกันมาก เช่น NSView ใน Cocoa และ HIView ใน Carbon ซึ่งทั้ง 2 class นี้เป็น class สำหรับจัดการการแสดงผลต่างๆบนหน้าจอ

2.3.1.1 Cocoa Framework

Cocoa นั้นประกอบได้ด้วย Framework ย่อยลงไปอีกสามารถแบ่งเป็น 3 Framework หลักคือ

1) Foundation Framework

2) Application Framework

3) Other Framework (CoreData , Sync Services , etc)

ในส่วนของ Foundation Framework นั้นแบ่งเป็น Class ต่างๆโดย Class ที่เกี่ยวข้องกับ Foundation Framework นั้นส่วนมากจะเป็น Class พื้นฐาน ไม่ว่าจะเป็น Class ที่เกี่ยวกับ String, Number, File เป็นต้น

สำหรับ Application Framework ประกอบด้วย Class ที่เกี่ยวข้องกับ Interface ทั้งหลาย เช่น NSView, NSMenu, NSToolbar Class ต่างๆใน Cocoa Framework ส่วนมากสืบทอดมาจาก NSObject

2.3.2 Language

การเขียนโปรแกรมบน iPhone นั้นสามารถเขียนได้ 3 ภาษาหลักๆ คือ

1) Objective C

2) C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) C#

2.3.2.1 ข้อดีและข้อเสียของแต่ละภาษา

2.3.2.1.1 ภาษา C

ข้อดีของภาษา C คือ การหาหนังสือรวมถึงแหล่งอ้างอิงเกี่ยวกับการเขียนโปรแกรมภาษา C ง่าย และภาษา C เป็นภาษาที่แพร่หลายและเก่าแก่ ในปัจจุบันนิยมใช้อยู่

ข้อเสียที่ของภาษา C คือ การเขียนโปรแกรมด้วยภาษา C ให้ได้ดีนั้นค่อนข้างยาก เป็นเพราะว่าตัวภาษาค่อนข้างจะมีรายละเอียดปลีกย่อยค่อนข้างมาก ซึ่งต้องใช้เวลาในการศึกษาเรียนรู้นาน แต่ด้วยความสามารถของตัวภาษา C นั้นสามารถเขียนโปรแกรมในระดับของการจัดการหน่วยความจำหรือแม้กระทั่งควบคุมและติดต่อ Hardware ได้โดยตรง ซึ่งอาจเป็นข้อเสียหนึ่ง ถ้าเขียนโปรแกรมจัดการหน่วยความจำได้ไม่ดี จะส่งผลโดยตรงต่อประสิทธิภาพของโปรแกรม ภาษา C ไม่มี Garbage Collector ช่วยในการจัดการหน่วยความจำที่ม่ได้ใช้งานอัตโนมัติ

การเขียนโปรแกรมด้วย ภาษา C นั้นจำเป็นต้องทำงานร่วมกับภาษา Objective-C เนื่องจากต้องอาศัย Interface Builder ในการทำ User Interface แต่อย่างข้างต้นว่า Library ของภาษา C นั้นมีเยอะมาก สามารถนำ Library ของภาษาที่มีอยู่แล้วนำมาใช้ร่วมกันกับภาษา Objective-C ได้

2.3.2.1.2 ภาษา Objective-C

ข้อดีของภาษา Objective-C คือ เรียนรู้ง่าย ไม่ซับซ้อน มี Framework, Library ที่คอยสนับสนุนการทำงานต่างๆ ภาษามีความยืนหยุ่นสูง และเนื่องจากว่าเป็น Small Super Set of C จะหมายความว่า สามารถใช้ Library หรือ Source Code ของภาษาC ได้ (ไม่ใช่ C++) และภาษา Objective-C ยังอนุญาตให้เข้าไปจัดการหน่วยความจำได้เหมือนกัน แต่ก็ไม่ยืดหยุ่นมากเท่ากับภาษา C และภาษา Objective-C มี Garbage Collector ในกรณีที่ไม่ต้องการจัดการหน่วยความจำเอง (แต่ในกรณีของ iPhone Garbage Collector จะไม่สามารถใช้ได้) และข้อดีที่สุดคือ การเขียน Application บน iPhone นั้น ด้วยภาษา Objective-C เพราะมี Tool ที่ชื่อ Interface Builder ช่วยให้สามารถออกแบบ User Interface ได้ง่ายขึ้น

ข้อเสียของภาษา Objective-C คือ หนังสือที่ใช้ในการศึกษานั้นมีน้อย และจำนวนแหล่งความรู้เพิ่มเติมของ Objective-C นี้ก็มีน้อยถ้าเทียบกับ .net หรือ Java ก็จะมีน้อยกว่ามาก จากสิ่งนี้ทำให้ การหาตัวอย่าง Source Code หรือ Tutorial นั้นยากตามไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2.3.2.1.3 ภาษา C#

ภาษา C# ควรเป็นภาษาที่พิจารณาเป็นลำดับสุดท้าย เพราะเนื่องจากว่า Apple ไม่ได้สนับสนุนการเขียน iPhone ด้วยภาษา C# นั้นต้องการ IDE ของ Mono-Project ซึ่งต้องซื้อเพิ่มเติม ที่สำคัญยังต้องเขียนบนเครื่อง MacBook ร่วมกับ iPhone SDK ซึ่งเหมาะสำหรับโปรแกรมที่มีอยู่แล้วใน Platform อื่นที่เขียนเป็นภาษา C# อยู่แล้วมากกว่า

2.3.3 Tools

การเขียนโปรแกรมใน MacBook นั้นมี Tools ที่จำเป็นในการเขียน Cocoa มีอยู่ด้วยกัน 2 ตัวคือ Xcode IDEกับ Interface Builder

2.3.3.1 Xcode IDE

Xcode IDE เป็น IDE ที่ใช้เขียนโปรแกรมบน MacBook สามารถติดตั้ง Xcode IDE ได้จาก แผ่น DVD/CD ที่มาพร้อมกับเครื่อง หรือ Download ได้ที่เว็บไซต์ของ Apple

2.3.3.2 Interface Builder

เป็น Tool สำหรับออกแบบ User Interface ของโปรแกรมซึ่งช่วยเขียน Cocoa ได้ง่ายขึ้น และยังช่วยลดระยะเวลาในการ Implement code

2.3.4 Cocoa และ Cocoa Touch

หากต้องเขียนโปรแกรมด้วย Objective-C แล้วนั่นก็หมายถึงว่าต้องเขียนด้วย Cocoa Framework และถ้าเป็น iPhone ก็จะเป็น Cocoa Touch ซึ่งเป็น Framework ขนาดเล็กกว่าและเพิ่มเติมในบางส่วน สิ่งที่ควรจะรู้เกี่ยวกับ iPhone และ Cocoa Touch นั้นก็คือ iPhone นั้นสามารถ run ได้เพียง 1 โปรแกรม ณ ขณะใดขณะหนึ่งยกเว้นแต่จะเขียนให้ Application สามารถทำงานแบบ Multitasking ซึ่งมีอยู่ใน iPhone iOS 4.0 ขึ้นไปและiPhone มีขนาดหน้าจอที่จำกัดซึ่งไม่สามารถเข้าไปใช้ File ของ Application ตัวอื่นๆได้นอกจากตัว Application ที่กำลัง run เองเท่านั้น

Cocoa Touch ไม่มี Garbage Collection

Cocoa Touch ไม่มี Binding

iPhone มี Resource ที่จำกัด Ram, Diskspace, Cpu

2.3.5 ส่วนต่างๆของ Xcode IDE

2.3.5.1 หน้าตาของ New Project

เมื่อเปิด Xcodeขึ้นมาและเริ่มสร้าง Project จะพบว่า มีหลายๆ Template ให้เลือก ดังรูป 2.6

117361

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 2.6 หน้าตา Xcode เมื่อเปิดโปรแกรม

แต่ละ Project นั้นก็จะมีรายละเอียดต่างๆ ดังนี้

1) Navigation-Based Applicationเป็น Projectที่แสดงในรูปแบบ Navigation (จะเป็นลักษณะ Hierarchy คือมีลำดับ)

2) OpenGL ES Applicationใช้เขียน Application จำพวก Game

3) Tab Bar หน้าตา User Interface ที่มี Tab เป็นตัวเปลี่ยนหน้าต่างไปมา โดยไม่ได้เรียงลำดับเหมือน Navigation

4) Utility Application

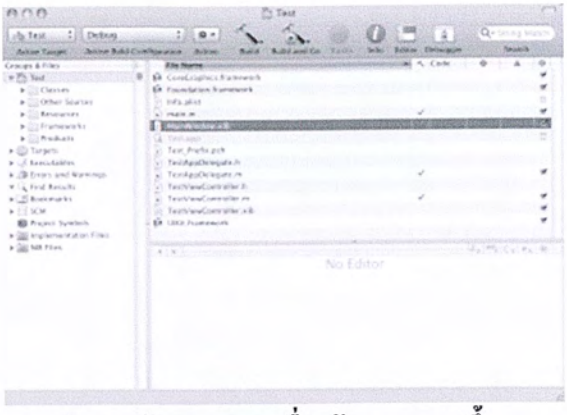
5) View-Based Applicationมี View เพียงอันเดียว

6) Window-Based Applicationเป็นพื้นๆ คือมี Application Delegate มาให้เท่านั้น

2.3.5.2 Files

เมื่อสร้าง Project ขึ้นมาจะพบว่ามี File ต่างๆดังรูป 2.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.7 หน้าตา Xcode เมื่อสร้าง Project ขึ้นมา

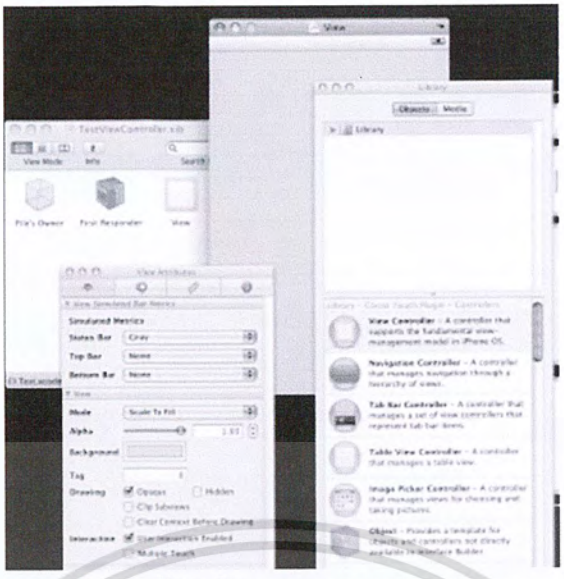
จากรูปข้างบน จะเห็นว่ามี File ต่างๆมากมาย อันได้แก่

1) .m และ .h 2 File นี้คือ Header กับ Source
2) .pch เรียกว่า Pre Compile Header ไฟล์นี้จะรวมรายชื่อ Header ของ Framework อื่นๆที่ต้องใช้ใน Project และทำการ Pre Compile ก่อน เพื่อลดเวลาในการ Compile
3) .framework เป็น Library
4) .plist ไฟล์นี้เป็น Property List เก็บค่า property ของ Application ของเราเช่น บอกว่า Icon ของ Application ใช้ Resource File ไหน
5) .xib เป็น Interface สำหรับโปรแกรม
6) .app ก็คือตัว Appication

2.3.5.3 Interface Builder

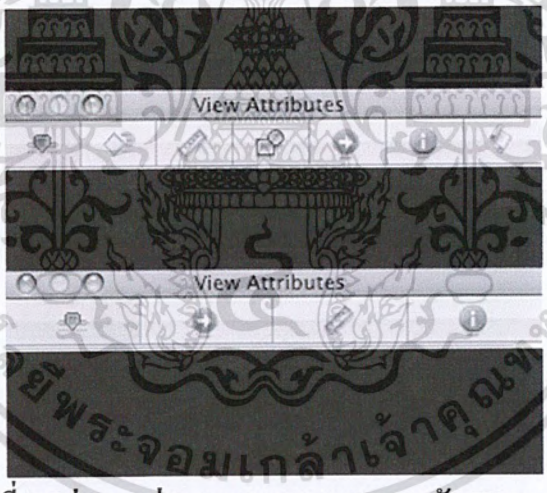
ทุกครั้งที่ออกแบบ User Interface Tool ที่ใช้คือ Interface Builder มี file ที่เกี่ยวข้องก็คือ file นามสกุล .xib ยกเว้นเขียนโปรแกรมที่ต้องใช้ OpenGL ที่ไม่ต้องใช้ Interface Builder และเมื่อเข้าไปใน file นามสกุล .xib ของเราจะเห็นหน้าตาของโปรแกรม Interface Builder ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.8 หน้าตาส่วนของ Interface Builder

ซึ่ง Interface Builder ของ iPhone นั้นไม่ต่างจากการเขียน Application บน MacBook ปกติ แตกต่างกันที่ Tab Bar ของ iPhone นั้นจะมี เพียง 4 Tab Bar



รูป 2.9 ส่วนที่แตกต่างระหว่าง Application บน Mac กับ iPhone ของ Xcode

นั่นก็คือ Attributes , Size , Connection , Identity ซึ่งจะพบว่าสิ่งที่ขาดไปก็คือ Binding , Effect , ApplcScript

2.3.5.4 Xib File

ดูที่ Document Windows จะเห็นว่า มี Object อยู่สองตัว คือ File's Owner กับ First Responder

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 2.10 Document Windows ใน Xcode

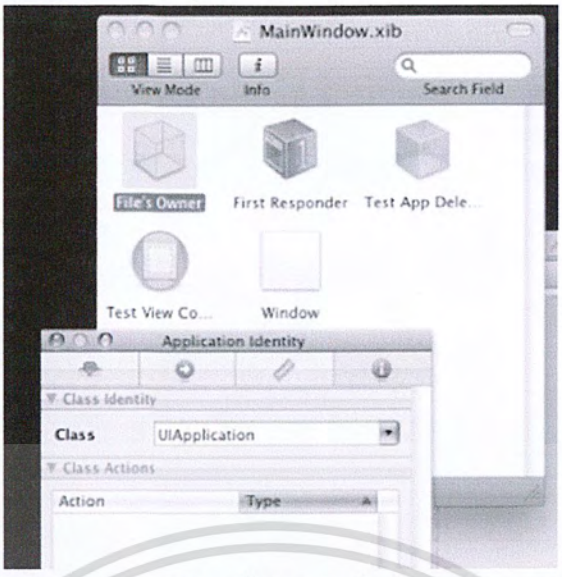
ความสำคัญของ Object 2 ตัวนี้คือ

1) File's Owner เป็นตัวแสดงถึง Object ที่ได้โหลดไฟล์ .xib

2) First Responder เป็น Object ที่จัดการ Interaction ของ Object ที่กำลังโดนกระทำอยู่ ณ ขณะนั้น ยกตัวอย่างเช่นถ้าหากกำลังพิมพ์ข้อความใน Textbox ก็แปลว่า Textbox คือ Responder และหากเปลี่ยนไปกดปุ่ม ปุ่มก็จะกลายเป็น Responder แทนและ First Responderก็จะช่วยให้โปรแกรมรู้ได้ว่า Object ไหนควรจะถูก Responder

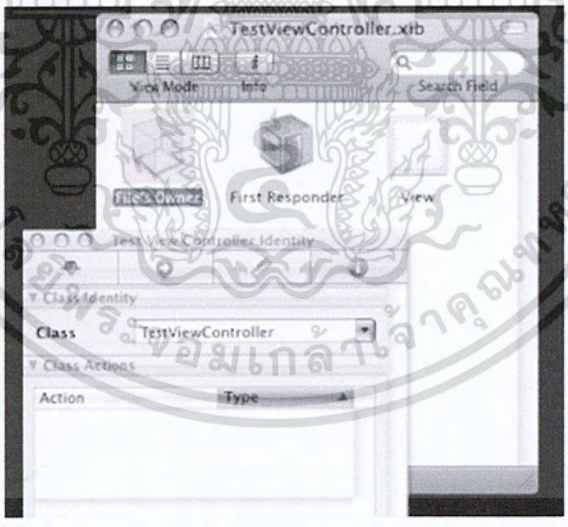
เมื่อได้สร้าง Project โดยให้ชื่อว่า Test และเลือกแบบ View-Based Application จะเห็นว่ามี .xib ทั้งหมดสอง files ด้วยกันคือ MainWindows.xib กับ TestViewController.xib และถ้าเปิด MainWindows.xib ขึ้นมาเลือกไปที่ File's Owner จะพบว่า Object ที่โหลด .xib นี้ก็คือ UIApplication ดังรูปที่ 2.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.11 Document Windows ใน Xcode ในส่วนของ File's Owner

ถ้าหากมาลองเดาจาก Interface จะมีข้อความตรงกลางที่เขียนว่า Loaded From "TestViewController.xib" ตรงนี้ก็จะสามารถตีความได้ว่า View ที่ Test View Controller ตัวนี้เป็นตัวควบคุม จะต้องไปโหลดจาก TextViewController.xib แล้วเปิด TestViewController.xib ขึ้นมาดู แล้วกดไปที่ File's Owner



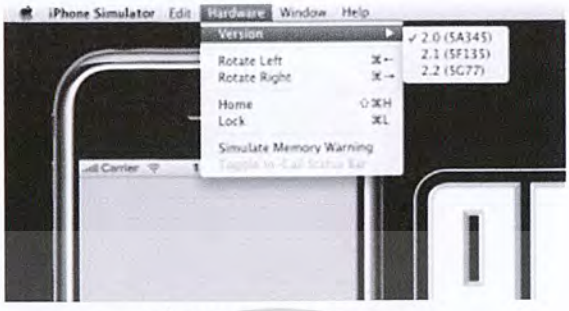
รูป 2.12 Document Windows ใน Xcode ในส่วนของ File's Owner(TestViesController)

จะเห็นว่า File's Owner คือ TestViewController จะสามารถเข้าใจ File's Owner สรุปก็คือเป็น Object ที่แสดงว่า ใครเป็นคนโหลด .xib นี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.5.5 iPhone Simulator

เมื่อได้เขียนโปรแกรมแล้ว และทำการ Compile ไปแล้วนั้นจะพบว่า Xcode จะเปิด Application ตัวใหม่ขึ้นมาคือ iPhone Simulator ตัว Simulator



รูป 2.13 iPhone Simulator

2.4 HTTP Protocol

เป็นวิธีขนส่งข้อความหลายมิติหรือเอชทีทีพี (HyperText Transfer Protocol HTTP) คือโพรโทคอลในระดับชั้นโปรแกรมประยุกต์เพื่อการแจกจ่ายและการทำงานร่วมกันกับสารสนเทศของสื่อผสม ใช้สำหรับการรับทรัพยากรที่เชื่อมโยงกับภายนอก ซึ่งนำไปสู่การจัดตั้งเวิลด์ไวด์เว็บ

การพัฒนาเอชทีทีพีเป็นการทำงานร่วมกันของเวิลด์ไวด์เว็บคอนซอร์เทียม (W3C) และคณะทำงานเฉพาะกิจด้านวิศวกรรมอินเทอร์เน็ต (IETF) ซึ่งมีผลงานเด่นในการเผยแพร่เอกสารขอความเห็น (RFC) หลายชุด เอกสารที่สำคัญที่สุดคือ RFC 2616 ได้ กำหนด HTTP/1.1 ซึ่งเป็นรุ่นที่เอชทีทีพีเป็นมาตรฐานในการร้องขอและการตอบรับระหว่างเครื่องลูกข่ายกับเครื่องแม่ข่าย ซึ่ง

เครื่องลูกข่ายคือผู้ใช้ปลายทาง (End-user) และเครื่องแม่ข่ายคือเว็บไซต์ เครื่องลูกข่ายจะสร้างการร้องขอเอชทีทีพีผ่านทางเว็บเบราว์เซอร์ เว็บครอว์เลอร์ หรือเครื่องมืออื่น ๆ ที่จัดว่าเป็น ตัวแทนผู้ใช้ (User Agent) ส่วนเครื่องแม่ข่ายที่ตอบรับ ซึ่งเก็บบันทึกหรือสร้าง ทรัพยากร (Resource) อย่างเช่นไฟล์เอชทีเอ็มแอลหรือรูปภาพ จะเรียกว่า เครื่องให้บริการต้นทาง (Origin Server) ในระหว่างตัวแทนผู้ใช้กับเครื่องให้บริการต้นทางอาจมีสื่อกลางหลายชนิด อาทิพร็อกซี เกตเวย์ และทุนเนล เอชทีทีพีไม่ได้จำกัดว่าจะต้องใช้ชุดเกณฑ์วิธีอินเทอร์เน็ต (TCP/IP) เท่านั้น แม้ว่าจะเป็นการใช้งานที่นิยมมากที่สุดบนอินเทอร์เน็ตก็ตาม โดยแท้จริงแล้วเอชทีทีพีสามารถ "นำไปใช้ได้บนโพรโทคอลอินเทอร์เน็ตอื่น ๆ หรือบนเครือข่ายอื่นก็ได้" เอชทีทีพีคาดหวังเพียงแค่การสื่อสารที่เชื่อถือได้ นั่นคือโพรโทคอลที่มีการรับรองเช่นนั้นก็สามารถใช้งานได้ใช้กันอย่างกว้างขวางในปัจจุบัน

ปกติเครื่องลูกข่ายเอชทีทีพีจะเป็นผู้เริ่มสร้างการร้องขอก่อน โดยเปิดการเชื่อมต่อด้วยเกณฑ์วิธีควบคุมการขนส่งข้อมูล (TCP) ไปยังพอร์ตเฉพาะของเครื่องแม่ข่าย (พอร์ต 80 เป็นค่าปริยาย)เครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แม่ข่ายเอชทีทีพีที่เปิดรอรับอยู่ที่พอร์ตนั้น จะเปิดรอให้เครื่องลูกข่ายส่งข้อความร้องขอเข้ามา เมื่อได้รับการร้องขอแล้ว เครื่องแม่ข่ายจะตอบรับด้วยข้อความสถานะอันหนึ่ง ตัวอย่างเช่น "HTTP/1.1 200 OK" ตามด้วยเนื้อหาของมันเองส่งไปด้วย เนื้อหานั้นอาจเป็นแฟ้มข้อมูลที่ร้องขอ ข้อความแสดงข้อผิดพลาด หรือข้อมูลอย่างอื่นเป็นต้น

ทรัพยากรที่ถูกเข้าถึงด้วยเอชทีทีพีจะถูกระบุโดยใช้ตัวระบุแหล่งทรัพยากรสากล (URI) (หรือเจาะจงลงไปก็คือ ตัวชี้แหล่งในอินเทอร์เน็ต (URL)) โดยใช้ http: หรือ https: เป็นแผนของตัวระบุ (URI scheme)

2.4.1 ข้อความร้องขอประกอบด้วยสิ่งต่อไปนี้

1) บรรทัดแรก ขึ้นต้นเป็นคำสั่งร้องขอ และเส้นทางไดเรกทอรีของแฟ้มที่ร้องขอ ตามด้วยรุ่นของ HTTP ตัวอย่างเช่น GET /images/logo.gif HTTP/1.1

2) บรรทัดต่อๆ ไปที่ไม่ใช่บรรทัดว่าง เรียกว่าเป็น ส่วนหัว (header) เป็นเมทาดาตาต่าง ๆ ประกอบการร้องขอ ตัวอย่างเช่น Accept-Language: en

3) บรรทัดว่าง เพื่อแบ่งแยกระหว่างส่วนหัวกับเนื้อหา

4) บรรทัดต่อๆ ไป เป็นเนื้อหาข้อมูล ซึ่งบางคำสั่งอาจไม่จำเป็นต้องใช้ส่วนนี้

แต่ละบรรทัดจะต้องลงท้ายด้วย CRLF (อักขระปัดแคร่ตามด้วยอักขระป้อนบรรทัด เหมือนการกดปุ่ม Enter ในวินโดวส์) บรรทัดที่ว่างจะมีเพียงแค่ CRLF เท่านั้นโดยไม่มีอักขระช่องว่างอยู่เลย สำหรับรุ่น HTTP/1.1 ส่วนหัว Host: จำเป็นต้องมีเสมอ แต่ส่วนหัวอื่น ๆ ไม่จำเป็นต้องมี บรรทัดคำสั่งที่มีเพียงเส้นทางไดเรกทอรี (ไม่มีชื่อแฟ้ม) ก็เป็นที่ยอมรับโดยเครื่องแม่ข่ายเพื่อรักษาความเข้ากันได้กับโปรแกรมตัวแทนรุ่นเก่าก่อนที่จะมีข้อกำหนดของ HTTP/1.0 ใน RFC 1945 ส่วน HTTP/1.1 ได้กำหนดไว้ใน RFC 2068

2.4.2 คำสั่งร้องขอ

เอชทีทีพีได้กำหนดคำสั่งร้องขอไว้แปดคำสั่ง (หรือเรียกว่าวิธีการร้องขอ บางครั้งอาจเรียกว่าเป็น "กริยา") แสดงการกระทำที่ต้องการ เพื่อที่จะดำเนินการกับทรัพยากรที่ถูกระบุ สิ่งที่ทรัพยากรนั้นนำเสนอ ไม่ว่าเป็นข้อมูลที่มีอยู่ก่อนหรือสร้างขึ้นมาแบบพลวัตก็ตาม จะขึ้นอยู่กับการนำไปใช้ของเครื่องแม่ข่าย ซึ่งบ่อยครั้งทรัพยากรมักจะสอดคล้องกับไฟล์ หรือผลลัพธ์ส่งออกจากโปรแกรมข้างเคียงในเครื่องแม่ข่ายนั้น เครื่องให้บริการเอชทีทีพีจะต้องสามารถใช้คำสั่ง GET และ HEAD ได้เป็นอย่างน้อย

1) HEAD ร้องขอการตอบรับจากทรัพยากรที่ระบุ คล้ายกับ GET แต่จะไม่มีส่วนเนื้อหาที่ร้องขอกลับมา คำสั่งนี้ใช้ประโยชน์ในการตรวจสอบข้อมูลส่วนหัวของการตอบรับโดยไม่จำเป็นต้องส่งเนื้อหาเต็มมาทั้งหมด

2) GET ร้องขอการนำเสนอจากทรัพยากรที่ระบุ คำสั่งนี้ไม่ควรใช้กับการดำเนินการที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อาจทำให้เกิดผลข้างเคียง เช่นการจัดการในเว็บแอปพลิเคชัน เหตุผลหนึ่งคือคำสั่ง GET มักจะถูกใช้อย่างไม่มีกฎเกณฑ์โดยอินเทอร์เน็ตบอตและเว็บครอว์เลอร์ ซึ่งไม่ควรพิจารณาให้การร้องขอของบอตและครอว์เลอร์ส่งผลกระทบต่อทรัพยากรในเว็บ

3) POST ส่งข้อมูลไปยังทรัพยากรที่ระบุเพื่อให้นำไปประมวลผล โดยเฉพาะข้อมูลที่ส่งมาจากฟอร์มเอชทีเอ็มแอล ข้อมูลที่ส่งจะถูกบรรจุอยู่ในเนื้อหาของการร้องขอสิ่งนี้อาจทำให้เกิดการสร้างทรัพยากรใหม่ หรือการปรับปรุงทรัพยากรที่มีอยู่ หรือทั้งสองกรณี

4) PUT อัปโหลดการนำเสนอของทรัพยากรที่ระบุ

5) DELETE ลบทรัพยากรที่ระบุ

6) TRACE ส่งข้อมูลร้องขอกลับมา เครื่องลูกข่ายจะเห็นว่ามีข้อมูลอะไรบ้างที่สื่อกลางเพิ่มหรือเปลี่ยนแปลงข้อความร้องขอก่อนไปถึงทรัพยากรปลายทาง

7) OPTIONS คืนค่าเป็นรายชื่อคำสั่งเอชทีทีพีที่เครื่องแม่ข่ายนั้นรองรับสำหรับทรัพยากรที่ระบุ สิ่งนี้สามารถใช้ตรวจสอบฟังก์ชันการทำงานของเว็บเซิร์ฟเวอร์ได้โดยใส่ "*" แทนที่การระบุทรัพยากร

8) CONNECT แปลงการเชื่อมต่อของการร้องขอไปเป็นทุนเนล TCP/IP แบบโปร่งใส มักใช้สำหรับแปลงการเชื่อมต่อที่เข้ารหัสแบบ SSL ให้เดินทางผ่านพร็อกซีที่ไม่มีการเข้ารหัสได้ง่ายขึ้น

2.4.3 คำสั่งที่ปลอดภัย

คำสั่งของเอชทีทีพีบางคำสั่งมีการกำหนดว่าเป็นคำสั่งที่ปลอดภัย เช่น HEAD, GET, OPTIONS, TRACE ซึ่งหมายความว่าคำสั่งเหล่านี้มีขึ้นเพื่อการรับข้อมูลเพียงอย่างเดียวและไม่ควรเปลี่ยนสถานะของเครื่องแม่ข่าย หรืออีกนัยหนึ่งคือคำสั่งเหล่านี้ไม่ควรทำให้เกิดผลกระทบข้างเคียงเว้นแต่ผลกระทบนั้นไม่สร้างความเสียหายอาทิ การบันทึกไฟล์ล็อค การเก็บแคช การบริการเว็บแบนเนอร์ หรือการเพิ่มตัวนับผู้เข้าชม การร้องขอแบบ GET แบบไม่มีกฎเกณฑ์โดยอินเทอร์เน็ตบอตและเว็บครอว์เลอร์ จึงยังคงถือว่าปลอดภัยอยู่

ในทางตรงข้าม คำสั่ง POST, PUT, DELETE เป็นการกระทำเพื่อให้เกิดผลกระทบต่อเครื่องแม่ข่าย หรือเกิดผลกระทบภายนอกเช่นทำให้เกิดธุรกรรมทางการเงิน หรือการส่งอีเมลเป็นต้น จึงเป็นคำสั่งที่ไม่ปลอดภัย คำสั่งเช่นนี้ปกติจะไม่ถูกใช้โดยอินเทอร์เน็ตบอตและเว็บครอว์เลอร์ซึ่งทำงานโดยไม่พิจารณาสถานะของเครื่องแม่ข่าย เพราะอาจทำให้ทรัพยากรเสียหายได้

2.4.4 รหัสสถานภาพ

ตั้งแต่ HTTP/1.0 เป็นต้นไป บรรทัดแรกของการตอบรับเอชทีทีพีเรียกว่า บรรทัดสถานภาพ (Status line) ซึ่งประกอบด้วยตัวเลข รหัสสถานภาพ (Status code เช่น 404) และข้อความวลีเหตุผล (Reason phrase เช่น "Not Found") โปรแกรมตัวแทนผู้ใช้จะพิจารณาการตอบรับในส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวโดยขึ้นอยู่กับรหัสสถานภาพเป็นหลัก และตามด้วยวลีเหตุผลเป็นรอง รหัสสถานภาพที่กำหนดขึ้นมาเองก็สามารถใช้ได้ ซึ่งหากตัวแทนผู้ใช้พบกับรหัสสถานภาพที่ไม่รู้จัก มันจะพิจารณาตัวเลขตัวแรกในรหัสเพื่อแยกประเภททั่วไปของการตอบรับ

รหัสสถานภาพของเอชทีทีพีแบ่งออกเป็นกลุ่มใหญ่ได้ 5 กลุ่มได้แก่

1) 1xx ข้อมูลทั่วไป

2) 2xx การร้องขอสำเร็จ

3) 3xx การเปลี่ยนทาง

4) 4xx ความผิดพลาดจากเครื่องลูกข่าย

5) 5xx ความผิดพลาดจากเครื่องแม่ข่าย

2.4.5 การเชื่อมต่อแบบคงอยู่

ใน HTTP/0.9 และ 1.0 การเชื่อมต่อจะถูกปิดทุกครั้งหลังจากการการร้องขอและการตอบรับจบไป ดังนั้นในรุ่น HTTP/1.1 จึงมีการแนะนำกลไกเพื่อให้การเชื่อมต่อยังคงอยู่ ซึ่งจะทำให้การเชื่อมต่อเพียงครั้งเดียวสามารถใช้ซ้ำได้อีกเรื่อย ๆ มากกว่าหนึ่งครั้ง การเชื่อมต่อแบบคงอยู่เช่นนั้นช่วยลดโอกาสของการเกิดความล่าช้า (Lag) เพราะว่าเครื่องลูกข่ายไม่จำเป็นต้องต่อรองการเชื่อมต่อทีซีพีใหม่อีกครั้ง หลังจากข้อความร้องขอแรกได้ถูกส่งไปแล้ว

HTTP/1.1 ได้มีการจัดแบนด์วิดท์ให้ดียิ่งขึ้นไปกว่ารุ่น 1.0 ตัวอย่างเช่น HTTP/1.1 มีการแนะนำการเข้ารหัสเป็นชิ้นส่วน (Chunked Transfer Encoding) เพื่อทำให้เนื้อหาบนการเชื่อมต่อแบบคงอยู่ส่งถ่ายเป็นกระแสข้อมูลได้ (Streaming) แทนที่จะเก็บลงในที่พักข้อมูล (buffer) การทำงานแบบสายท่อของเอชทีทีพี (HTTP Pipelining) ก็เป็นอีกเทคนิคหนึ่งที่ช่วยลดความช้าลงได้อย่างมาก ทำให้เครื่องลูกข่ายสามารถส่งข้อความร้องขอได้หลายข้อความ ก่อนที่จะได้รับข้อความตอบรับของอันแรก อีกพัฒนาการหนึ่งคือการบริการเป็นไบต์ (Byte Serving) ซึ่งจะทำให้เครื่องแม่ข่ายส่งถ่ายข้อมูลมาเพียงแค่ส่วนหนึ่งจากทรัพยากรทั้งอัน ในช่วงตำแหน่งที่เครื่องลูกข่ายต้องการ

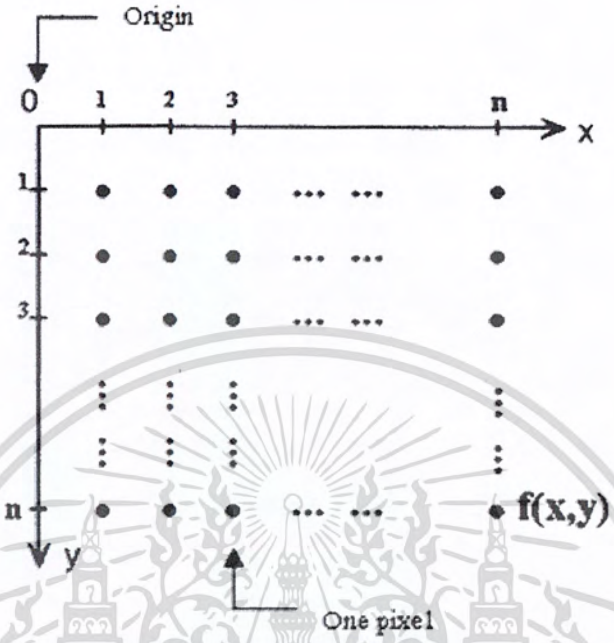
2.4.6 สถานะของเอชทีทีพี

เอชทีทีพีเป็นโพรโทคอลที่ไม่มีการระบุสถานะ ข้อดีของโพรโทคอลแบบนี้คือเครื่องแม่ข่ายไม่จำเป็นต้องดึงข้อมูลอื่นมาจากผู้ใช้ในระหว่างการร้องขอ แต่สิ่งนี้ทำให้ผู้พัฒนาเว็บต้องใช้วิธีการอื่นเพื่อรักษาสถานะของผู้ใช้ ตัวอย่างเช่น ผู้ใช้ที่ต้องการปรับแต่งเนื้อหาเว็บไซต์ เว็บแอปพลิเคชั่นจะต้องบันทึกติดตามกระบวนการต่าง ๆ ของผู้ใช้เป็นหน้าต่อหน้า หรือการล็อกอินเข้าสู่ระบบ ซึ่งจำเป็นต้องทราบว่าผู้ใช้นั้นอยู่ในสถานะล็อกอินหรือไม่ จึงจะส่งข้อความตอบรับได้อย่างเหมาะสม วิธีการที่เป็นปกติสำหรับปัญหานี้คือการรับและส่งคุกกี้ และวิธีการอื่นคือการสร้างตัวแปรวาระ (Session) ทางฝั่งเครื่องแม่ข่าย หรือใช้ตัวแปรซ่อนผ่านทางหน้าแบบฟอร์ม หรือใช้พารามิเตอร์ที่เข้ารหัสแบบตัวชี้แหล่งในอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 Image Processing

2.5.1 Digital Image



รูป 2.14 พิกัดของภาพดิจิตอล

ภาพดิจิตอลสามารถนิยามเป็นฟังก์ชัน 2 มิติ f(x, y) โดยที่ x และ y คือแถวและหลักตามลำดับ Amplitude ที่พิกัด (x, y) ใดๆภายในภาพ คือค่าความเข้มแสงของภาพ(Intensity) ที่ตำแหน่ง (x, y) (Amplitude f เป็นค่าจำกัด(Finite Value)) กำหนดให้ภาพดิจิตอลที่มีขนาด M แถว N หลัก และมีพิกัดของจุดกำเนิดเป็น (0, 0) ดังนั้นจะสามารถเขียนให้อยู่ในรูปสมการเมตริกซ์ (Matrix) ได้ดังนี้

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,N-1) \\ f(1,0) & f(1,1) & & f(1,N-1) \\ \vdots & & \ddots & \vdots \\ f(M-1,0) & f(M-1,1) & \cdots & f(M-1,N-1) \end{bmatrix} \quad (2.1)$$

จากสมการที่ 2.1 ค่าแต่ละค่าที่อยู่ในเมตริกซ์จะเรียกว่า “องค์ประกอบภาพ (Picture Element)” หรือPixel โดยตำแหน่ง(0, 0) จะอยู่ด้านซ้ายมือด้านบนของภาพ การจัดลำดับตำแหน่งของจุดภาพจะเรียงจากซ้ายไปขวาในแต่ละเส้นภาพและจัดลำดับของเส้นภาพจะเรียงจากบนลงล่าง การใช้หน่วยความจำเพื่อจัดเก็บภาพดิจิตอลโดยเนื้อที่ในการเก็บภาพสามารถคำนวณได้จาก จำนวน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แถว M แถว จำนวนหลัก N หลัก และจำนวนบิตของข้อมูลในแต่ละจุดภาพ(pixel)

2.5.1.1 ประเภทของภาพดิจิตอลมีด้วยกัน 4 ประเภทหลักๆ คือ

1) ภาพขาวดำ(Binary Image or Black and White Image)มีค่าความเข้มแสงที่แตกต่างกันอยู่ 2 ระดับคือขาว(1) และดำ(0)

2) ภาพระดับสีเทา(Intensity Image or Monochrome Image or Gray Image)ภาพจะมีลักษณะเป็นโทนสีเทา(Gray Scale)โดยค่าความเข้มแสงที่สามารถแสดงได้ (Gray Levels) จะขึ้นอยู่กับจำนวนบิตที่ใช้แทนภาพในแต่ละภาพได้จาก $L=2^b$ เมื่อ L คือระดับสีเทา k คือจำนวนบิต เช่น ภาพระดับสีเทาขนาด 8 บิต สามารถแสดงค่าสีเทาได้ 256 ระดับและค่าที่เป็นไปได้คือ 0-255 โดยจะไล่จากสีดำ สีเทาไปเรื่อยๆจนขาว

3) ภาพดัชนี(Indexed Image)ในแต่ละ pixel ในภาพ แทนด้วยค่าคงที่ค่าหนึ่งที่ชี้ไปยังตารางสี

4) ภาพสี(Color Image or RGB Image) Pixel ในภาพประกอบด้วยค่าสี 3 สี คือ แดง เขียว และน้ำเงิน

2.5.2 การประมวลผลภาพ

ใน OpenCV ใช้การประมวลผลภาพด้วยเทคนิค convolution ซึ่งเป็นเทคนิคพื้นฐานในการประมวลผลภาพที่ใช้ในการกรองสัญญาณรบกวน เทคนิคนี้จะใช้หน้าต่างหรือเทมเพลตบางครั้งอาจเรียกว่า “ภาพย่อย(Sub-image)” หรือ “หน้ากาก(Mask)” หรือ “เคอเนล(Kernel)” เคลื่อนที่ไปตาม pixel ต่างๆในภาพโดยค่าเทมเพลตนี้เรียกว่า “สัมประสิทธิ์การกรองหรือค่าถ่วงน้ำหนักการกรอง(น้ำหนักถ่วงของเทมเพลตตัวกรอง)” ผลลัพธ์ของตัวเทมเพลตที่จุด pixel ใดๆคือ ผลคูณของค่าถ่วงน้ำหนักและค่าความเข้มแสงในตำแหน่งที่สัมพันธ์กับค่าตัวถ่วงน้ำหนักภายในพื้นที่ๆเทมเพลตคลอบคลุมเป็นไปดังสมการที่ 2.1

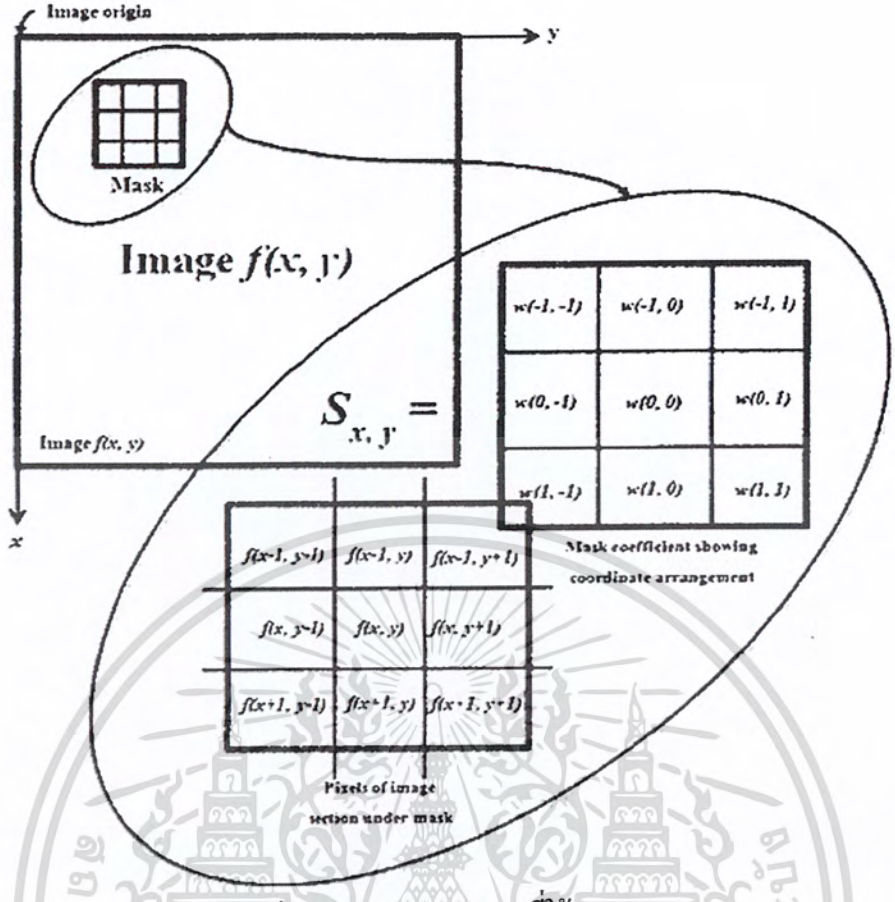
$$G(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x-s, y-t) \qquad (2.2)$$

หรือ $G = h * f$

โดย G คือ ภาพผลลัพธ์จากการทำ Convolution h คือเทมเพลตที่ใช้กรองและ f คือinput imageเทมเพลตตัวกรองมีขนาด m x n และค่า a = (m-1)/2 และ b = (n-1)/2 ตัวอย่าง ถ้าเรามีเทมเพลต ซึ่งมีขนาด 3x3 pixels

$$W = \begin{bmatrix} w(-1,-1) & w(-1,0) & w(-1,1) \\ w(0,-1) & w(0,0) & w(0,1) \\ w(1,1) & w(1,0) & w(1,1) \end{bmatrix} \qquad (2.3)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.15 ตาราง Template ที่ใช้กรอง

ผลตอบสนอง g(x , y) ที่ได้สำหรับตำแหน่ง (x, y) ในภาพมีค่าดังนี้

$$g(x,y) = \begin{pmatrix} w(-1,-1)f(x+1,y+1) + w(-1,0)f(x+1,y) + \\ w(-1,1)f(x+1,y-1) \quad + w(0,-1)f(x,y+1) + \\ w(0,0)f(x,y) \qquad\qquad + w(0,-1)f(x,y-1) + \\ w(-1,-1)f(x-1,y+1) + w(1,0)f(x-1,y) \\ w(1,1f(x-1,y-1)) \end{pmatrix} \quad (2.4)$$

น้ำหนักตัวถ่วงในเทมเพลตตัวกรอง w(0, 0) จะเป็นค่าที่ตรงกับตำแหน่ง (x, y) ที่ต้องการทำการกรองซึ่งเป็นตำแหน่งจุดกึ่งกลางของเทมเพลต ปกติเทมเพลตจะมีขนาด m x n โดย m = 2a +1 และ n = 2b +1 ซึ่ง a และ b เป็นจำนวนเต็มบวก

2.5.2.1 ปัญหาที่เกิดขึ้นจากการทำ Convolution

1) ปัญหาค่าความเข้มแสงที่ได้จากการคำนวณเกินช่วงที่กำหนด เช่น ความเข้มแสงติดลบหรือมีค่ามากเกินช่วงที่กำหนด แก้โดย

1.1) กำหนดชนิดข้อมูลภาพ output เป็นแบบมีเครื่องหมาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2) แปลงค่าความเข้มแสงที่ได้หลังจากทำ Convolution ที่ต่ำกว่า 0 โดยเปลี่ยนจำนวนบิตที่เก็บข้อมูล

1.3) ทำการ Normalize ผลลัพธ์ให้อยู่ในช่วงเดียวกับค่าความเข้มแสงจากภาพต้นฉบับ

2) ปัญหาที่ขอบภาพ บางส่วนในตัวกรองไม่มีค่าความเข้มแสง เนื่องวัตถุจากเลยขอบเขตของภาพดังรูป

รูป 2.16 ปัญหาที่ของภาพ

มีวิธีแก้อยู่หลายวิธีดังนี้

1) ภาพต้นฉบับไม่ต้องทำ Convolution ที่ขอบภาพ นั่นคือบริเวณขอบภาพจะถูก set เป็นสีดำ

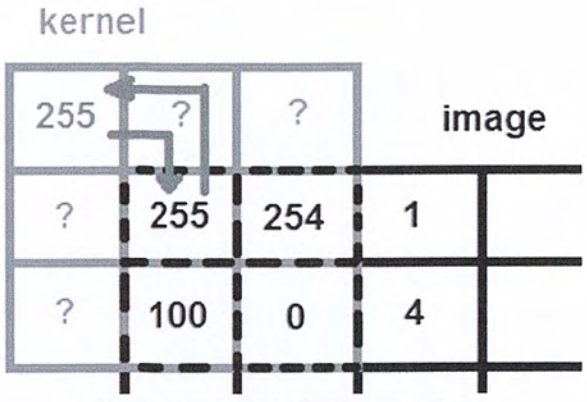
2) กำหนดค่าความเข้มแสงที่ขอบภาพของภาพ output คือค่าความเข้มแสงที่ขอบภาพของภาพต้นฉบับ

3) ภาพต้นฉบับทำการตัดขอบภาพทิ้ง การทำในลักษณะนี้จะทำให้ภาพ output มีขนาดเล็กลง

4) กำหนดตัวกรองให้เหมาะสมกับบริเวณต่างๆของภาพ

5) กำหนดดัชนีแบบสะท้อน(reflected indexing)

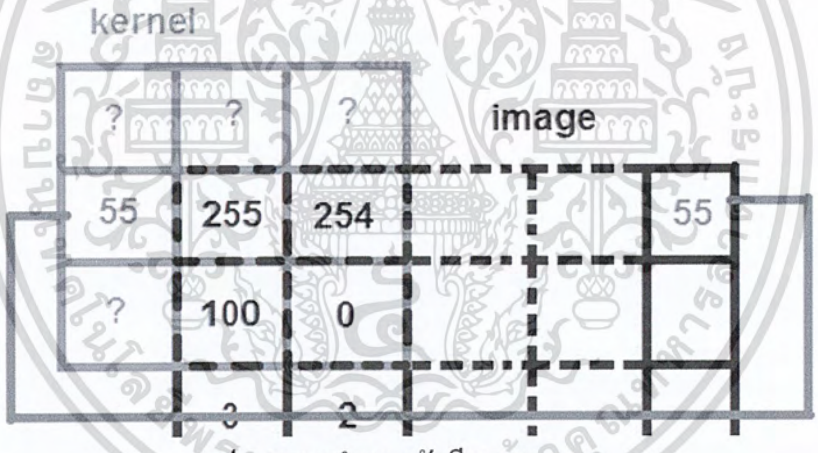
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.17 การกำหนดดัชนีแบบสะท้อน

กรณีเราสนใจ pixel ที่ f(1,1) และกำลังคำนวณที่ w(1,1) w เป็นค่าอะไรเรายังไม่สนใจตอนนี้ จากสมการด้านบน w(1,1)*f(0-1,0-1) จะสังเกตเห็นว่า f(-1,-1) เกินขอบเขตของภาพฉะนั้นจะกำหนดดัชนีแบบสะท้อนคือ f(-1+1,-1+1) = 255

6) ดัชนีภาพแบบวนรอบ(circular indexing)



รูป 2.18 การกำหนดดัชนีแบบวนรอบ

ทั้ง 2 ปัญหา library ของ OpenCV แก้ปัญหาตรงส่วนนี้แล้ว ส่วนปัญหาบริเวณขอบภาพเมื่อมีการทำ Convolution ที่ขอบภาพ function cvCopyMakeBorder() ใน OpenCVจะถูกเรียกโดยอัตโนมัติ โดยใช้ border type IPL_BORDER_REPLICATE หรือแก้ไขแบบ reflected indexing นั่นเอง หรือจะแก้ไขปัญหาด้วยวิธีของเราเองก็ได้ที่ใช้ใน project เราใช้การแก้ไขแบบ reflected indexing เช่นกัน เพราะ เวลาถ่วงน้ำหนัก มันจะได้ค่าที่ใกล้เคียงกับ pixel ที่เราสนใจอยู่ circular indexing ก็ใช้ได้แต่ถ้าเจอค่าความเข้มแสงที่ต่างกันมากเมื่อถ่วงน้ำหนักออกมา ค่าที่ได้มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โอกาสเพี้ยนไปมาก ส่วนวิธีที่เหลือไม่อยู่ในการพิจารณาที่จะนำมาใช้เนื่องจากทำให้ภาพที่ได้ผิดไปจากภาพต้นฉบับอย่างมากไม่เหมาะที่จะนำมาใช้ใน project

2.5.2.2 การปรับปรุงคุณภาพภาพ

ภาพแต่ละภาพมีรายละเอียดในภาพไม่มากก็น้อย การที่จะวาดขอบวัตถุในภาพได้นั้น เราจำเป็นต้องทำให้รายละเอียดต่างๆในภาพลดน้อยลงเหลือแต่ขอบของบางสิ่งบางอย่างในภาพ ด้วยการทำ smoothing หรือเรียกอีกอย่างว่า blurring ซึ่งเป็นการลด noise(pixel ที่โดดเด่นหรือแตกต่างจาก pixel รอบๆ)ไปในตัว smoothing มีด้วยกัน 5 แบบใน OpenCVคือ

1) CV_BLUR คือการทำค่าเฉลี่ย pixel(ตำแหน่ง x, y ใดๆ)กับ pixel รอบข้างโดยใช้ convolution ซึ่งมีค่าถ่วงน้ำหนักของเทมเพลตตัวกรองเป็น 1 ในทุกตำแหน่ง

$$G(x, y) = (1/mn) \sum_{(s,t) \in S_{xy}} f(s, t) \quad (2.5)$$

โดย

m, n คือ จำนวน pixel ด้านกว้างและด้านยาวในเทมเพลตตัวกรอง

G(x, y) คือ ความเข้มแสงผลลัพธ์ที่ได้ ณ ตำแหน่ง x, y

S_{xy}คือ เทมเพลตตัวกรอง หรือวินโดว์

s และ t คือ ความเข้มแสงที่ตำแหน่ง x และ y ใดๆตามลำดับ

2) CV_BLUR_NO_SCALE เหมือนกับ CV_BLUR แต่ไม่ได้หารด้วยขนาดของเทมเพลต ตัวกรองผลลัพธ์ ที่ได้จะเร็วกว่า CV_BLUR (เพราะใช้ผลรวมในขณะที่ CV_BLUR ต้องหารด้วย param1*param2 อีกที) ข้อควรระวังคือ dst จะต้องมี bit depth ที่มากกว่า src ไม่งั้นจะเกิดการ overflow ได้นั่นคือภาพที่ได้หลังจากการทำ smoothing จะใหญ่ขึ้น

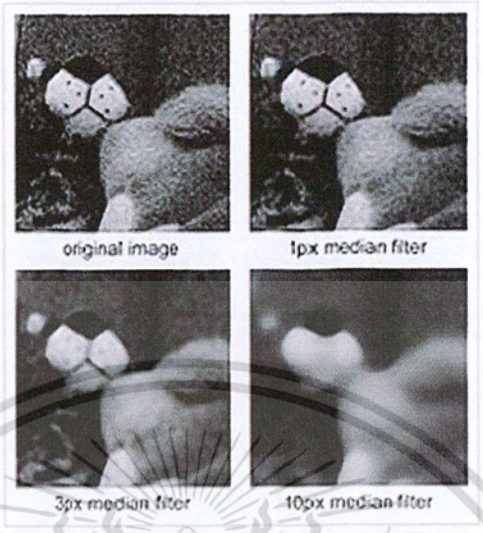
3) CV_MEDIAN เหมือน CV_BLUR แต่ใช้ค่ามัธยฐานแทนค่าเฉลี่ยเลขคณิต เทคนิคนี้จะแทนค่า pixel นั้นๆด้วยค่ามัธยฐานของค่าความเข้มแสงใน pixel ที่อยู่ในพื้นที่ของเทมเพลตตัวกรอง ซึ่งมีสมการดังนี้

$$g(x,y) = \underset{(s,t) \in Sx_{xy}}{median\{f(s,t)\}} \quad (2.6)$$

โดย f คือ ค่าความเข้มของ pixel ที่ถูกจัดเรียงแล้ว จากน้อยไปมากการหาค่ามัธยฐานทำได้ดังนี้ ในตัวกรองเทมเพลตขนาด n x n และ n เป็นเลขคี่ ซึ่งหลังจากการจัดเรียงค่าความเข้มแสงของ pixel(f) ที่อยู่ในเทมเพลตตัวกรองหรือวินโดว์ (S_{xy})จากค่าความเข้มแสงในเทมเพลตตัวกรองแทนที่ลงไปยังค่า pixel นั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(g(x,y))ด้วยค่ากลางที่ตำแหน่ง ($n^2/2$)+1 จากค่าความเข้มแสงที่ถูกจัดเรียงไว้ การทำ smoothing แบบนี้ใช้ลดภาพที่มี noise แบบ salt & pepper ได้ดี



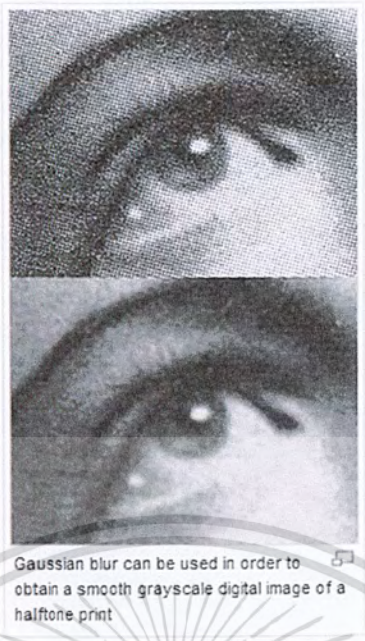
รูป 2.19 ลักษณะการเกิด Noise บนภาพและการ Blur ภาพ

4) CV_GAUSSIANคือการใช้ค่าเฉลี่ยแบบ GAUSSIAN โดยค่า pixel แต่ละ pixel จะถูกถ่วงน้ำหนักด้วยค่าที่แปรผกผันกับระยะห่างของ pixel ยิ่งไกลมากผลของ pixel นั้นก็จะน้อย

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \qquad (2.7)$$

โดย ค่า x และ y คือค่าความคว้างความยาวของรูปจากจุด origin ส่วน σ เป็นส่วนเบี่ยงเบนมาตรฐานของการกระจายของ Gaussian

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.20 ภาพการกระจายแบบ Gaussian

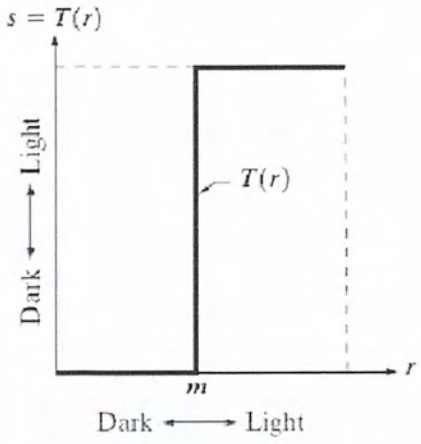
5) CV_BILATERAL โดยหลักการของ bilateral filtering นั้นจะคำนวณ pixel โดยถ่วงน้ำหนักจากทั้ง space (ระยะห่าง ยิ่งห่างมากยิ่งมีน้ำหนักหรือความเข้มน้อยลง) และ range(ความแตกต่างของสียิ่งแตกต่างมากยิ่งมีน้ำหนักน้อย) ใน OpenCVการถ่วงน้ำหนักที่ใช้ จะใช้ Guassian distribution ทั้ง space และ range smooth แบบนี้จะรักษา edge ของภาพไว้ได้ดีกว่าแบบ CV_GAUSSIAN

จากคุณสมบัติของ smoothing ทั้ง 5 แบบเราเลือกใช้ CV_MEDIAN เพราะสามารถลด noise และรายละเอียดของภาพได้เป็นอย่างดี ตรงตามความต้องการที่จะนำมาใช้ใน project ภาพที่ได้จากการทำ smoothing รายละเอียดในภาพลดลง สีบางส่วนกลืนกัน และเห็นขอบของส่วนต่างๆในภาพชัดเจน แต่ก็ยังมีหลายสีอยู่ เพื่อให้ง่ายต่อการ detect ขอบเราต้องแปลงค่าความเข้มแสงให้เหลือเพียง 2 ระดับ(ขาว-ดำ)ด้วยเทคนิค thresholding ซึ่งเป็นฟังก์ชันที่แบ่งระดับภาพสีเทาเป็น 2 ระดับ (Binary Image) ก่อนที่จะทำการ thresholding เราจำต้องแปลง image ใดๆเป็น gray image ด้วยสมการ

$$Y = 0.299 *R + 0.587 *G + 0.114 * B \quad (2.8)$$

โดย Y คือความเข้มแสง และ RGB เป็นแสงสีแดง เขียวและน้ำเงิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.21 กราฟการหาค่า Threshold

จากรูป S หรือแกน y คือ output image , r หรือแกน x คือ input image ส่วน T(r) คือ บริเวณที่เห็นความแตกต่างระหว่างสีขาวสีดำชัดเจน เป็นฟังก์ชันที่ใช้แปลงค่าความเข้มแสงของ input image r, m คือ ค่าที่เราเลือกมาเป็นค่า threshold ส่วนใหญ่จะเป็นค่ากลางของความเข้มแสงของ input image หรือจะกำหนดเองก็ได้ ตรงนี้เราต้องกำหนดค่าๆหนึ่งให้เหมาะสมกับ input image ที่เข้ามาทุกรูปซึ่งต้องทำการทดลองเพื่อหาค่าดังกล่าว ใน OpenCVแบ่ง type ของ threshold ออกเป็น 5type คือ

1) CV_THRESHOLD_BINARY dst = (src>T)?M:0

2) CV_THRESHOLD_BINARY_INV dst = (src>T)?0:M

3) CV_THRESHOLD_TRUNC dst = (src>T)?M:src

4) CV_THRESHOLD_TOZERO_INV dst = (src>T)?0:src

5) CV_THRESHOLD_TOZERO dst = (src>T)?src:0

โดย srcคือ ค่าความเข้มแสง ณ ตำแหน่งใดๆในภาพของ input image

dstคือ ค่าความเข้มแสง ณ ตำแหน่งใดๆในภาพของ output image

T คือ ค่าของ threshold ที่กำหนดไว้

M คือ ค่าความเข้มแสงสูงสุดที่กำหนดไว้

ส่วนนี้จะนำ CV_THRESHOLD_BINARY มาใช้เพราะเราได้ไม่ต้องการ input image แต่ต้องการความแตกต่างระหว่างวัตถุ กับ background อย่างชัดเจน

2.5.2.3 Image Transforms using Affine transform

เป็นการแปลงค่า pixel ในลักษณะเชิงเส้นของภาพ 2D โดย map ค่าของ pixel ตำแหน่ง x, y เก่า(x1,y1) ใน input image ไปเป็นตำแหน่ง x, y ใหม่(x2,y2)ใน output image การแปลงค่าแบบเชิงเส้นนี้ แบ่งออกได้เป็น translation, rotation, scaling และ reflect

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Affine transform โดยทั่วไปสามารถเขียนให้อยู่ในรูปของ coordinate ใหม่ในภาพ output ได้ดังนี้

$$\begin{vmatrix} x_2 \\ y_2 \end{vmatrix} = A \times \begin{vmatrix} x_1 \\ y_1 \end{vmatrix} + B \tag{2.9}$$

โดยที่ matrix A และ B ขึ้นอยู่กับการ transform แต่ละแบบ

1) Translation

$$A = \begin{vmatrix} 1 & 0 \\ 0 & 1 \end{vmatrix}, B = \begin{vmatrix} b_1 \\ b_2 \end{vmatrix} \tag{2.10}$$

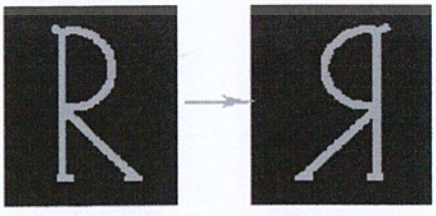
รูป 2.22 ตัวอย่างการทำ translation

2) Rotation

$$A = \begin{vmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{vmatrix}, B = \begin{vmatrix} 0 \\ 0 \end{vmatrix} \tag{2.11}$$

รูป 2.23 ตัวอย่างการทำ rotation

3) Reflect



รูป 2.24 ตัวอย่างการทำ Reflect

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นการ set pixel หรือความเข้มแสงตำแหน่งใหม่(x2, y2)ของภาพ output ให้อยู่ ในตำแหน่งตรงกันข้ามกับตำแหน่งเดิม(x1, y1)ในภาพต้นฉบับ ซึ่งอ้างอิงจากตำแหน่ง pixel ตรงกลางภาพ(x0, y0) coordinate ใหม่นี้หาได้จากสมาการด้านล่าง ดังนี้

3.1) ภาพสะท้อนในแนวตั้ง

$$x_2 = -x_1 + (2 * x_0) \tag{2.12}$$

$$y_2 = y_1 \tag{2.13}$$

3.2) ภาพสะท้อนในแนวนอน

$$x_2 = x_1 \tag{2.14}$$

$$y_2 = -y_1 + (2 * y_0) \tag{2.15}$$

3.3) ภาพสะท้อนในแนวองศา

$$x_2 = x_1 + 2 * \Delta * (-\sin\theta) \tag{2.16}$$

$$y_2 = y_1 + 2 * \Delta * (cos\,\theta) \tag{2.17}$$

$$\text{โดยที่ } \Delta = (x_1 - x_0) * \sin\theta - (y_1 - y_0) * \cos\theta \tag{2.18}$$

กรณีที่อ้างอิงจาก (x0, y0) แต่ (x0, y0) ไม่ใช่จุดกึ่งกลางของภาพ คือ ทำการreflect เฉพาะ area จะใช้สมการดังนี้

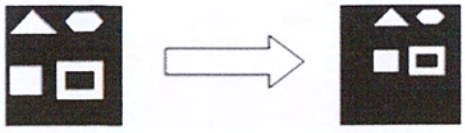
$$x_2 = -x_1 + (2 * x_0) \tag{2.19}$$

$$y_2 = -y_1 + (2 * y_0) \tag{2.20}$$

4) Scaling

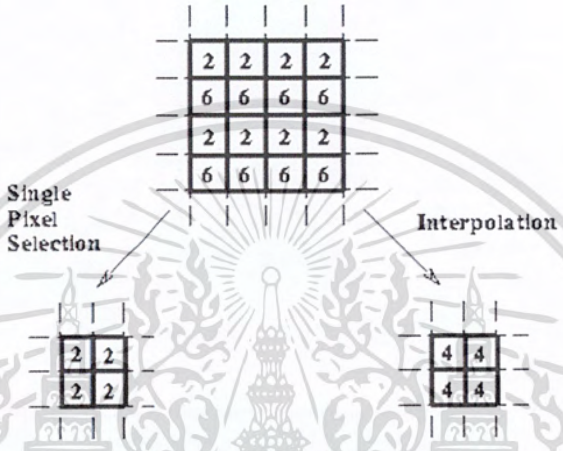
$$A = \begin{vmatrix} a_{11} & 0 \\ 0 & a_{22} \end{vmatrix}, B = \begin{vmatrix} 0 \\ 0 \end{vmatrix} \tag{2.21}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.25 ตัวอย่างการทำ scaling

Scaling หรือการ resize สามารถอธิบายเพิ่มเติมได้ว่า เป็นการบีบ หรือขยายภาพตามแนวพิกัด ซึ่งจะใช้เทคนิค sub-sampling และการ zoom



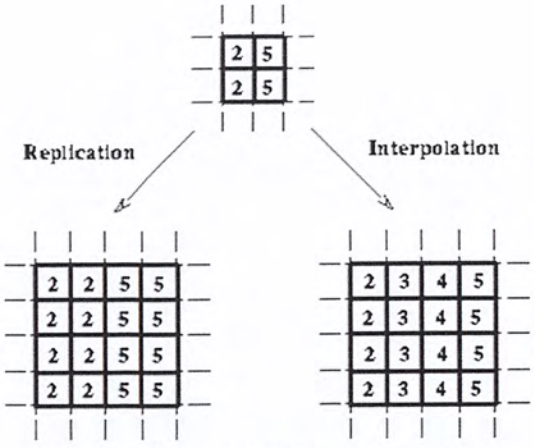
รูป 2.26 การทำ Sub-sampling

4.1) จากรูป 2.26 ใช้วิธีการของ sub-sampling มี 2 แบบ ดังนี้

4.1.1) แบบแรก พิจารณา pixel ที่สนใจและ neighborhood รอบๆ pixel ที่สนใจหรือ pixel ทั้งหมดในตัวกรอง แล้วเลือกค่าความเข้มแสงที่ต่ำสุด(กรณีที่ neighborhoods มีความเข้มแสงที่มากกว่า)

4.1.2) แบบที่สอง ทำการแก้ไขค่า pixel หรือความเข้มแสงในตัวกรองด้วยการประมาณค่า หรือปรับให้เป็นค่ากลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.27 การทำการ Zoom

4.2) จากรูป 2.27 ใช้วิธีการ zoom ซึ่งมีด้วยกัน 2 แบบดังนี้

4.2.1) แบบแรก คือการทำซ้ำ(replication) โดยการนำค่าความเข้มแสงจากภาพเดิมมาใส่ให้พอดีกับขนาดของรูป

4.2.2) แบบที่ 2 ทำการแก้ไขโดยนำค่าความเข้มแสงจากภาพดั้งเดิมมาขยายออก ส่วนที่ถูกขยายจะถูก set เป็นค่าความเข้มแสงที่เพิ่มขึ้นเรื่อยๆทีละ 1 จนถึง pixel(ความเข้มสง)ดั้งเดิม

ในการทำ scaling ใน project จะใช้หลักของ interpolation ทั้งหมด เพราะทำให้ภาพที่ได้คงสภาพเดิมมากที่สุด

2.5.2.4 Edge detection

Canny edge detection ใน OpenCV เป็น method ที่ใช้ในการหา edges ซึ่งพิจารณาจากความเข้มแสงที่ถูกคำนวณด้วยอนุพันธ์ลำดับที่ 1นั่นคือทำให้กราฟความเข้มแสงมีความชันเพิ่มขึ้น ทำให้เห็นเป็น รอยต่อ เส้นหรือขอบ ที่ชัดเจนขึ้น เส้นที่ปรากฏนั้นขึ้นอยู่กับช่วง threshold ที่กำหนดด้วย ค่า threshold แบ่งเป็นค่า threshold สูงและต่ำ เสมือนเป็นการกรองเอาเฉพาะช่วงความเข้มแสงที่ต้องการ เมื่อทำการกรองแล้วเส้นที่มีความเข้มแสงที่อยู่ในช่วงของค่า threshold จะปรากฏขึ้นมา

2.6 โครงข่ายประสาทเทียม (Artificial Neural Network)

โครงข่ายประสาทเทียม (Artificial Neural Network) หรือที่มักจะเรียกสั้นๆว่า ข่ายงานประสาท (Neural Network หรือ Neural Net) คือโมเดลทางคณิตศาสตร์ สำหรับประมวลผลสารสนเทศด้วยการคำนวณแบบคอนเนคชันนิสต์ (Connectionist) เพื่อจำลองการทำงานของเครือข่ายประสาทในสมองมนุษย์ด้วยวัตถุประสงค์ที่จะสร้างเครื่องมือซึ่งมีความสามารถในการเรียนรู้การจดจำแบบรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Pattern Recognition) และการอุปมานความรู้ (Knowledge Deduction) เช่นเดียวกับความสามารถที่มีในสมองมนุษย์แนวคิดเริ่มต้นของเทคนิคนี้ได้มาจากการศึกษาข่ายงานไฟฟ้าชีวภาพ (Bioelectric Network) ในสมอง ซึ่งประกอบด้วย เซลล์ประสาท หรือ “นิวรอน” (Neurons) และ จุดประสานประสาท (Synapses) แต่ละเซลล์ประสาทประกอบด้วยปลายในการรับกระแสประสาท เรียกว่า "เดนไดรท์" (Dendrite) ซึ่งเป็น Input และปลายในการส่งกระแสประสาทเรียกว่า "แอคซอน" (Axon) ซึ่งเป็นเหมือน Output ของเซลล์ เซลล์เหล่านี้ทำงานด้วยปฏิกิริยาไฟฟ้าเคมีเมื่อมีการกระตุ้นด้วยสิ่งเร้าภายนอกหรือกระตุ้นด้วยเซลล์ด้วยกันกระแสประสาทจะวิ่งผ่านเดนไดรท์เข้าสู่นิวเคลียสซึ่งจะเป็นตัวตัดสินว่าต้องกระตุ้นเซลล์อื่น ๆต่อหรือไม่ ถ้ากระแสประสาทแรงพอ นิวเคลียสก็จะกระตุ้นเซลล์อื่น ๆต่อไปผ่านทางแอคซอนของมัน

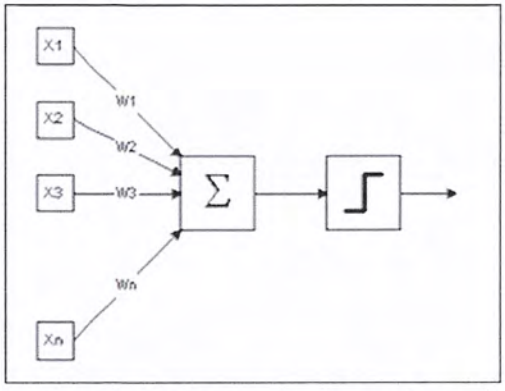
ตามโมเดลนี้ข่ายงานประสาทเกิดจากการเชื่อมต่อระหว่างเซลล์ประสาท จนเป็นเครือข่ายที่ทำงานร่วมกัน

รูป 2.28 Model ของ Neuron ในสมองมนุษย์

2.6.1 โครงสร้าง

นักวิจัยส่วนใหญ่ในปัจจุบันเห็นตรงกันว่าข่ายงานประสาทเทียมมีโครงสร้างแตกต่างจากข่ายงานในสมอง แต่ก็ยังเหมือนสมอง ในแง่ที่ว่าข่ายงานประสาทเทียม คือการรวมกลุ่มแบบขนานของหน่วยประมวลผลย่อย ๆ และการเชื่อมต่อนี้เป็นส่วนสำคัญที่ทำให้เกิดสติปัญญาของข่ายงาน เมื่อพิจารณาขนาดแล้วสมองมีขนาดใหญ่กว่าข่ายงานประสาทเทียมอย่างมาก รวมทั้งเซลล์ประสาทยังมีความซับซ้อนกว่าหน่วยย่อยของข่ายงาน อย่างไรก็ดีหน้าที่สำคัญของสมอง เช่น การเรียนรู้ยังคงสามารถถูกจำลองขึ้นอย่างง่ายด้วยโครงข่ายประสาทนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.29 Model ของ Neuron ในคอมพิวเตอร์

2.6.2 หลักการ

สำหรับในคอมพิวเตอร์ Neurons ประกอบด้วย Input และ Output เหมือนกัน โดยจำลองให้ Input แต่ละอันมี Weight เป็นตัวกำหนดน้ำหนักของ Input โดย Neuron แต่ละหน่วยจะมีค่า Threshold เป็นตัวกำหนดว่าน้ำหนักรวมของ Input ต้องมากขนาดไหนจึงจะสามารถส่ง Output ไปยัง Neurons ตัวอื่นได้ เมื่อนำ Neuron แต่ละหน่วยมาต่อกันให้ทำงานร่วมกันการทำงานนี้ในทางตรรกะแล้วก็จะเหมือนกับปฏิกิริยาเคมีที่เกิดในสมองเพียงแต่ในคอมพิวเตอร์ทุกอย่างเป็นตัวเลขเท่านั้นเอง

2.6.3 การทำงาน

การทำงานของ Neural Networks คือเมื่อมี Input เข้ามายัง Network ก็เอา Input มาคูณกับ Weight ของแต่ละขา ผลที่ได้จาก Input ทุก ๆ ขาของ Neuron จะเอามารวมกันแล้วก็เอามาเทียบกับ Threshold ที่กำหนดไว้ ถ้าผลรวมมีค่ามากกว่า Threshold แล้ว Neuron ก็จะส่ง Output ออกไป Output นี้ก็จะถูกส่งไปยัง Input ของ Neuron อื่น ๆ ที่เชื่อมกันใน Network ถ้าค่าน้อยกว่า Threshold ก็จะไม่เกิด Output เขียนออกมาได้ดังนี้

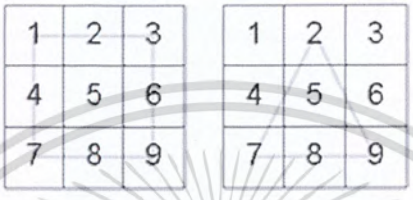
$$\textit{if (sum(input * weight) > threshold) then output} \qquad (2.22)$$

สิ่งสำคัญคือเราต้องทราบค่า Weight และ Threshold สำหรับสิ่งที่เราต้องการเพื่อให้คอมพิวเตอร์รู้จำซึ่งเป็นค่าที่ไม่แน่นอนแต่สามารถกำหนดให้คอมพิวเตอร์ปรับค่าเหล่านั้นได้โดยการสอนให้มันรู้จัก Pattern ของสิ่งที่เราต้องการให้มันรู้จำ เรียกว่า "Back propagation" ซึ่งเป็นกระบวนการย้อนกลับของการรู้จำ ในการฝึก Feed-forward Neural Networks จะมีการใช้อัลกอริทึมแบบ Back-propagation เพื่อใช้ในการปรับปรุงน้ำหนักคะแนนของเครือข่าย (Network Weight) หลังจากใส่รูปแบบข้อมูลสำหรับฝึกให้แก่เครือข่ายในแต่ละครั้งแล้ว ค่าที่ได้รับ (Output) จาก

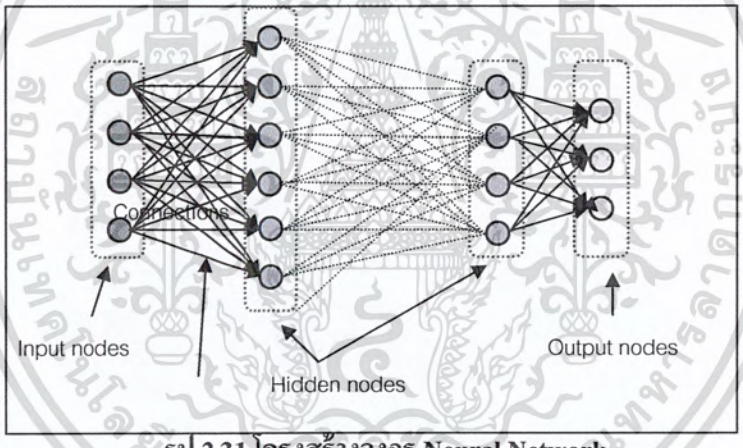
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครือข่ายจะถูกนำไปเปรียบเทียบกับผลที่คาดหวังแล้วทำการคำนวณหาค่าความผิดพลาดซึ่งค่าความผิดพลาดนี้จะถูกส่งกลับเข้าสู่เครือข่ายเพื่อใช้แก้ไขค่าน้ำหนักคะแนนต่อไป

อย่างเช่นจะรู้จำรูปสามเหลี่ยมกับรูปสี่เหลี่ยม เราอาจแบ่ง Input เป็น 9 ตัวคือเป็นตาราง 3x3 ถ้าวาดรูปสี่เหลี่ยมหรือสามเหลี่ยมให้เต็มกรอบ 3x3 พอดีสี่เหลี่ยมจะมีส่วนของขอบอยู่ในช่อง 1,2,3,4,6,7,8,9 ก็สมมติให้น้ำหนักตรงช่องเหล่านี้มีค่ามาก ๆถ้ามีเส้นขีดผ่านก็เอามาคูณกับน้ำหนักแล้วก็เอามารวมกันตั้งค่าให้พอเหมาะก็จะสามารถแยกแยะระหว่างสี่เหลี่ยมกับสามเหลี่ยมได้ ซึ่งนี่คือหลักการของ Neural Network



รูป 2.30 การแยกแยะระหว่างสี่เหลี่ยมและสามเหลี่ยม



รูป 2.31 โครงสร้างวงจร Neural Network

Output ของแต่ละNode

$$y_i = f(w_i^1 x_1 + w_i^2 x_2 + w_i^3 x_3 + \cdots + w_i^m x_m) = f(\sum_j w_i^j x_j) \quad (2.23)$$

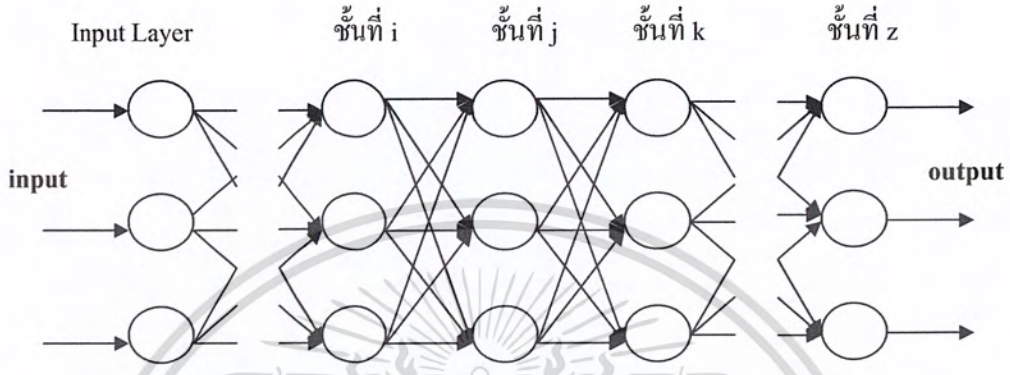
เมื่อ Xi = Input จากโหนดอื่นๆ

Wij = น้ำหนัก(weight)ของแต่ละแขน (connection)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.4 Back propagationAlgorithm

Back-propagation เป็นอัลกอริทึมที่ใช้ในการเรียนรู้ของเครือข่ายใยประสาทวิธีหนึ่งที่นิยมใช้ใน Multilayer Perceptron เพื่อปรับค่าน้ำหนักในเส้นเชื่อมต่อระหว่างโหนดให้เหมาะสม โดยการปรับค่านี้จะขึ้นกับความแตกต่างของค่าเอาต์พุตที่คำนวณได้กับค่าเอาต์พุตที่ต้องการ พิจารณารูปต่อไปนี้ประกอบ



รูป 2.32 แสดงรูปแบบ Back-propagation Neural Network

2.6.4.1 ขั้นตอนของ Back-propagation Algorithm มีดังนี้

1) กำหนดค่าอัตราเร็วในการเรียนรู้ (Rate parameter : r)

2) สำหรับแต่ละตัวอย่างอินพุตให้ทำตามขั้นตอนต่อไปนี้จนกว่าได้ระดับ Performance ที่ต้องการ

2.1) คำนวณหาค่าเอาต์พุต โดยใช้ค่าน้ำหนักเริ่มต้นซึ่งอาจได้จากการสุ่ม

2.2) คำนวณหาค่า β: แทนประโยชน์ที่จะได้รับสำหรับการเปลี่ยนค่าเอาต์พุตของแต่ละโหนด

2.3) ในชั้นเอาต์พุต (Output Layer)

$$\beta_z = d_z - o_z \tag{2.24}$$

เมื่อ d_z = ค่าเอาต์พุตที่ต้องการ

o_z = ค่าเอาต์พุตที่คำนวณได้

2.4) ในชั้นซ่อน (Hidden Layer)

$$\beta_j = \sum w_{j\ k}\, o_k (1 - o_k)\, \beta_k \tag{2.25}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

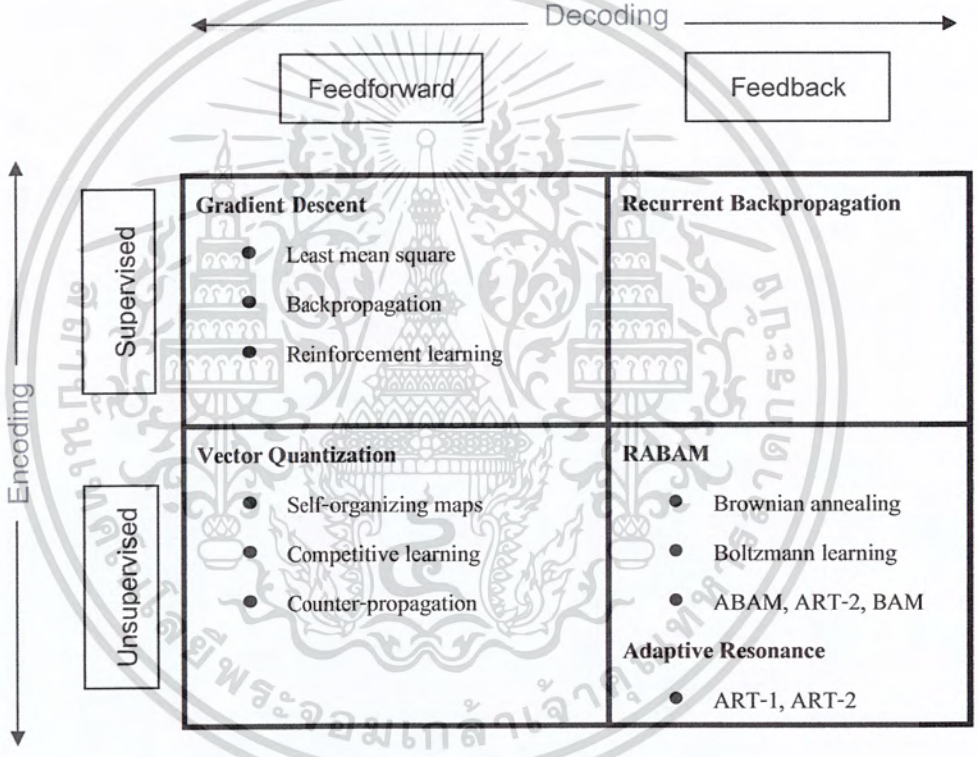
เมื่อ $w_{j\ k}$ = น้ำหนักของเส้นเชื่อมระหว่างชั้นที่ j กับ k

2.5) คำนวณค่าน้ำหนักที่เปลี่ยนแปลงไปสำหรับในทุกน้ำหนัก ด้วยสมการต่อไปนี้

$$\Delta w_{i\ j} = r\, o_i\, o_j\, (1 - o_j)\, \beta_j \qquad (2.26)$$

2.6) เพิ่มค่าน้ำหนักที่เปลี่ยนแปลง สำหรับตัวอย่างอินพุตทั้งหมด และเปลี่ยนค่าน้ำหนัก

2.6.5 Neural Network Taxonomy



รูป 2.33 Neural Network Taxonomy

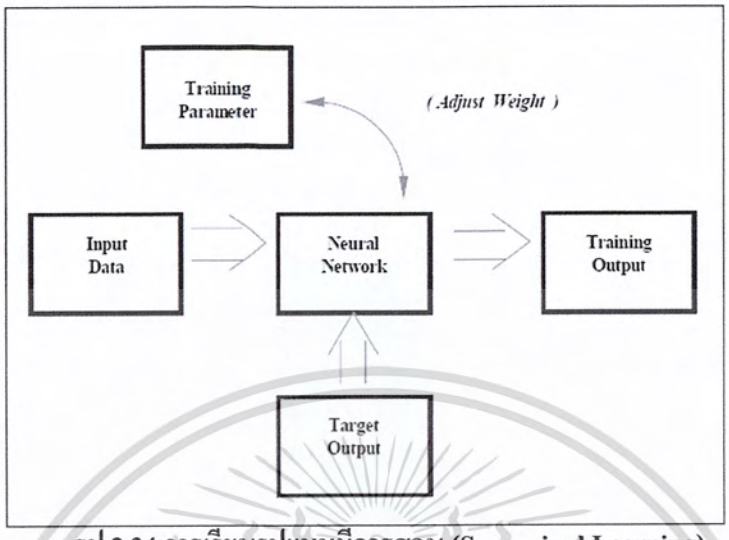
2.6.6 การเรียนรู้สำหรับ Neural Network

2.6.6.1 Supervised Learning การเรียนแบบมีการสอน

เป็นการเรียนแบบที่มีการตรวจคำตอบเพื่อให้วงจรข่ายปรับตัว ชุดข้อมูลที่ใช้สอนวงจรข่ายจะมีคำตอบไว้คอยตรวจดูว่าวงจรข่ายให้คำตอบที่ถูกหรือไม่ถ้าตอบไม่ถูกวงจรข่ายก็จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

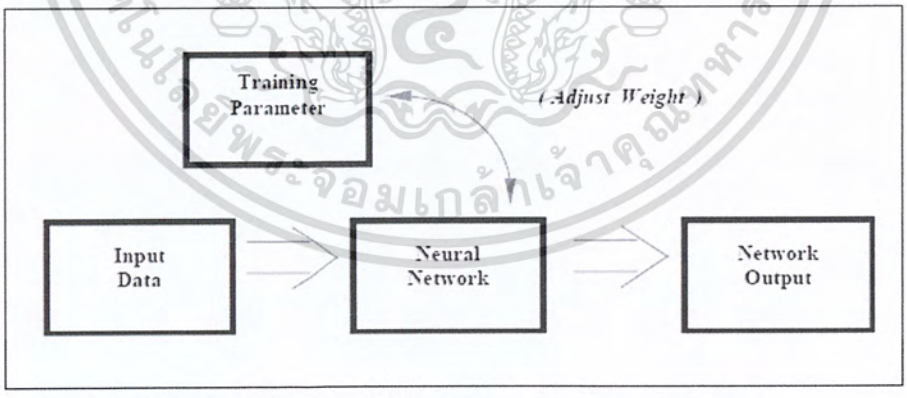
ปรับตัวเองเพื่อให้ได้คำตอบที่ดีขึ้น (เปรียบเทียบกับคน เหมือนกับการสอนนักเรียนโดยมีครูผู้สอนคอยแนะนำ)



รูป 2.34 การเรียนรูปแบบมีการสอน (Supervised Learning)

2.6.6.2 Unsupervised Learning การเรียนแบบไม่มีการสอน

เป็นการเรียนแบบไม่มีผู้แนะนำ ไม่มีการตรวจคำตอบว่าถูกหรือผิด วงจรข่ายจะจัดเรียงโครงสร้างด้วยตัวเองตามลักษณะของข้อมูล ผลลัพธ์ที่ได้ วงจรข่ายจะสามารถจัดหมวดหมู่ของข้อมูลได้ (เปรียบเทียบกับคน เช่น การที่เราสามารถแยกแยะพันธุ์พืช พันธุ์สัตว์ตามลักษณะรูปร่างของมันได้เองโดยไม่มีใครสอน)



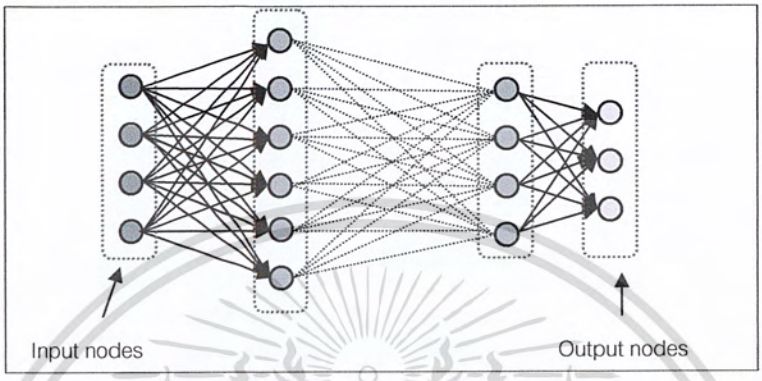
รูป 2.35 การเรียนรู้แบบไม่มีการสอน Unsupervised Learning

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.7 Network Architecture

2.6.7.1 Feedforward network

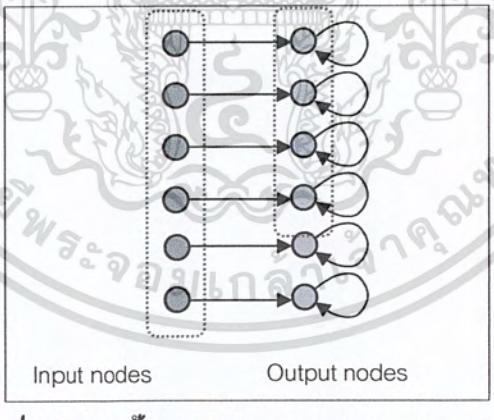
ข้อมูลที่ประมวลผลในวงจรข่ายจะถูกส่งไปในทิศทางเดียวจาก Input nodes ส่งต่อมาเรื่อย ๆ จนถึง Output nodes โดยไม่มีการย้อนกลับของข้อมูล หรือแม้แต่ Nodes ใน layer เดียวกันก็ไม่มีการเชื่อมต่อกัน



รูป 2.36 สถาปัตยกรรมของ Feedforward Network

2.6.7.2 Feedback network

ข้อมูลที่ประมวลผลในวงจรข่ายจะมีการป้อนกลับเข้าไปยังวงจรข่ายหลาย ๆ ครั้ง จนกระทั่งได้คำตอบออกมา (บางทีเรียกว่า Recurrent network)



รูป 2.37 สถาปัตยกรรมของ Feedback Network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.7.3 Network Layer

พื้นฐานสามัญที่สำคัญของ Artificial Neural Network ประกอบไปด้วย 3 ส่วนหรือ 3 layer ได้แก่ ชั้นของ Input units ที่ถูกเชื่อมต่อกับชั้นของ Hidden units ซึ่งเชื่อมต่อกับชั้นของ Output units

การทำงานของ Input unit จะทำหน้าที่แทนส่วนของข้อมูลดิบ ที่จะถูกป้อนเข้าสู่เครือข่าย

การทำงานของแต่ละ Hidden units จะถูกกำหนด โดยการทำงานของ Input units และค่าน้ำหนักบนความสัมพันธ์ระหว่าง Input units และ Hidden units

พฤติกรรมการทำงานของ Output units จะขึ้นอยู่กับการทำงานของ Hidden units และค่าน้ำหนักระหว่าง Hidden units และ Output units

ประเภทของเครือข่ายนี้เป็นที่น่าสนใจ เพราะเราสามารถกำหนดการแทนค่าให้แก่ Input units ได้อย่างอิสระ ค่าน้ำหนักระหว่าง Input units และ Hidden units จะถูกกำหนดเมื่อ Hidden unit กำลังทำงาน ฉะนั้นเวลาที่แก้ไขค่าน้ำหนัก Hidden units จะสามารถเลือกว่าอะไรคือค่าที่เราแทนเข้ามา

Architecture of Layerสามารถจำแนกสถาปัตยกรรมของชั้น (Layer) ออกเป็น 2 ประเภทคือ Single-layer และ Multi-layer

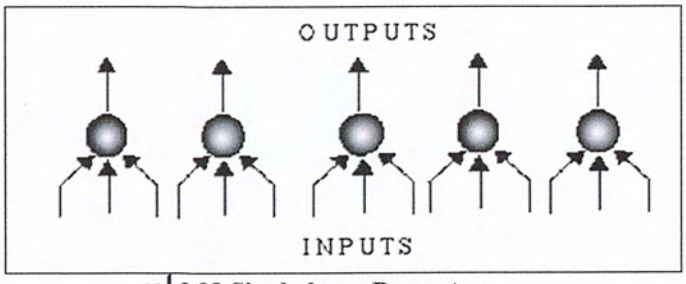
1) Single-layer Perceptronเครือข่ายใยประสาทที่ประกอบด้วยชั้นเพียงชั้นเดียว จำนวน Input Nodes ขึ้นอยู่กับจำนวน Components ของ Input Data และ Activation function ขึ้นอยู่กับลักษณะข้อมูลของ Output เช่นถ้า Output ที่ต้องการเป็น “ใช่” หรือ “ไม่ใช่” เราจะต้องใช้ Threshold function

$$f(x)=\begin{cases}1 & \text{if } x \geq T \\ 0 & \text{if } x < T\end{cases} \quad \textit{T=Threshold level} \tag{2.27}$$

หรือถ้า Output เป็นค่าตัวเลขที่ต่อเนื่อง เราต้องใช้ Continuous function เช่น Sigmoid function

$$f(x)=\frac{1}{1+e^{-\alpha x}} \tag{2.28}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.38 Single-layer Perceptron

2) Multi-layer Perceptronเครือข่ายใยประสาทจะประกอบด้วยหลายชั้นโดยในแต่ละชั้นจะประกอบด้วยโหนด(Nodes)หรือเปรียบได้กับตัวเซลล์ประสาท (Neurons) ค่าน้ำหนักของเส้นที่เชื่อมต่อระหว่างโหนดของแต่ละชั้น(เมทริก W), ค่า Bias Vector (b) และค่า Output Vector (a) โดย m เป็นตัวเลขบอกลำดับชั้นกำกับไว้ด้านบน เมื่อ p เป็น Input Vector การคำนวณค่าเอาต์พุตสำหรับเครือข่ายใยประสาทที่มี M ชั้นจะเป็นดังสมการ

$$a^{m+1} = f^{m+1}(W^{m+1}a^{m} + b^{m+1}) \qquad (2.29)$$

เมื่อ m = 0, 2, ..., M-1

$a^{0} = p$

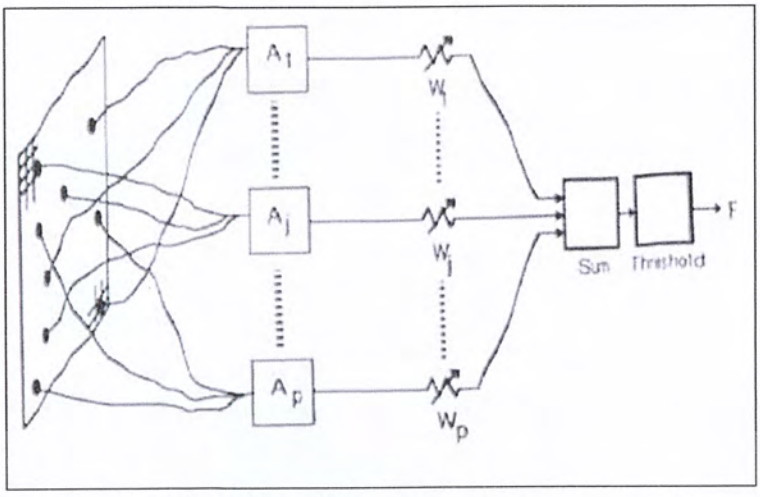
$a = a^{m}$

และ f เป็น Transfer function

2.6.7.4 Perceptrons

ในยุค 60s งานส่วนใหญ่ของข่ายงานได้รับการวิพากษ์วิจารณ์ในหัวข้อเรื่อง Perceptronsซึ่งค้นพบโดย Frank Rosenblatt โดย Perceptronsซึ่งกลายเป็น MCP model (Neuron with Weighted Inputs) พร้อมกับส่วนต่อเติม จากรูปในส่วน A1,A2,Aj,Ap เรียกว่า Association units การทำงานเพื่อคัดเลือกสิ่งที่แตกต่างออกมาจากรูปภาพที่รับเข้าไป โดย Perceptronsสามารถคัดลอกความคิดพื้นฐานภายในของสัตว์เลี้ยงลูกด้วยนม หลัก ๆ แล้วจะใช้ในรูปแบบ Recognition และสามารถขยายให้มีความสามารถสูงกว่านี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.39 โครงสร้างของ Perceptrons

ในปี ค.ศ.1969 MinskeyและPapertได้เขียนหนังสืออธิบายเกี่ยวกับขอบเขตของ Single-layer Perceptronsผลกระทบที่ได้รับจากหนังสือเล่มนั้นร้ายแรง เป็นเหตุให้นักวิจัยสาขา Neural Network สูญเสียผลประโยชน์เนื่องจากหนังสือสามารถถ่ายทอดออกมาได้ดี และแสดงข้อมูลในเชิงคำนวนว่า Single-layer Perceptronsไม่สามารถที่จะสร้างรูปแบบการจดจำพื้นฐาน (Basic Pattern Recognition Operation) ได้ เช่น การกำหนดความคล้ายคลึงของรูปร่าง หรือกำหนดว่ารูปร่างใดสัมพันธ์กันหรือไม่แต่สิ่งที่นักวิจัยไม่รู้จนกระทั่งยุค 80s คือ การได้รับการฝึกฝนที่ถูกต้อง ซึ่ง Multi-layer Perceptronsสามารถดำเนินการแก้ไขสิ่งเหล่านี้ได้

2.6.8 Notation

1) ให้ weight ของ network แสดงในรูปของ vector notation ใน real Euclidean space $\Re^N$ ให้ w เป็น weightvector define โดย

$$w = (\ldots, W_{ij}^{(l)}, W_{i+1j}^{(l)}, \ldots, W_{Nj}^{(l)}, \theta_j^{(l+1)}, W_{ij+1}^{(l)}, W_{i+1j+1}^{(l)}, \ldots) \qquad (2.30)$$

2) E'(w) เป็น global error function (backward pass) คำนวณโดย

$$E'(w) = (\ldots, \sum_{p=1}^{P} \frac{dE_p}{dw_{ij}^{(l)}}, \sum_{p=1}^{P} \frac{dE_p}{dw_{i+1j}^{(l)}}, \ldots, \sum_{p=1}^{P} \frac{dE_p}{dw_{Nj}^{(l)}}, \sum_{p=1}^{P} \frac{dE_p}{d\theta_j^{(l+1)}}, \sum_{p=1}^{P} \frac{dE_p}{dw_{ij+1}^{(l)}}, \ldots) \qquad (2.31)$$

โดยที่ P คือจำนวนของpattern และ Ep คือ error ของที่เกี่ยวกับ pattern นั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) weight length นิยามโดย

$$|w| = [\sum_{j=1}^{N}(w^i)^2]^{\frac{1}{2}} \quad (2.32)$$

4) E(w) เป็น error function ที่จุดที่กำหนดให้ (w+y)

$$E(w+y) = E(w)+ E'(w)^T y + \frac{1}{2}y^T E''(w)y + \ldots \quad (2.33)$$

5) matrix A NxN จะเป็น positive ถ้า

$$y^T Ay > 0 \qquad \forall\, y \in \Re^N \quad (2.34)$$

ให้ p1,..,pkเป็น set ของ non zero weight vectors ใน $\Re^N$ จะเป็น conjugate system ถ้า ตรงตามเงื่อนไขต่อไปนี้

$$p_i^T Ap_j = 0 \qquad (i \neq j.\ i = 1,\ldots k) \quad (2.35)$$

และ set ของ จุด w ใน $\Re^N$ เป็นไปตามสมการ

$$w = w_1 + \alpha_1 p_1 + \ldots + \alpha_k p_k,\ \ \alpha_i \in \Re, \quad (2.36)$$

โดย w1 เป็นจุดใน weight space และ p1....pk เป็น จุดใน conjugate system

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.8.1 Pseudo code for scaled conjugate gradient algorithm

โปรแกรมที่ 2.1 Pseudo code for scaled conjugate gradient algorithm

Choose weight vector w1 and scalars $\sigma > 0, \lambda_1 > 0$ and $\overline{\lambda_1} = 0$

Set $p_1 = r_1 = -E'(w_1), k = 1$ and success = true.

If success = true then calculate second order information:

$$\sigma_k = \frac{\sigma}{|p_k|}$$

$$s_k = \frac{E'(w_k + \sigma_k p_k) - E'(w_k)}{\sigma_k}$$

$$\delta_k = p_k^T s_k$$

Scalesk:

$$s_k = s_k + (\lambda_k - \overline{\lambda_k})p_k$$

$$\delta_k = \delta_k + (\lambda_k - \overline{\lambda_k})|p_k|^2$$

If $\delta_k \leq 0$ then make the Hessian matrix positive definite

$$s_k = s_k + (\lambda_k - 2\frac{\delta_k}{|p_k|^2})p_k$$

$$\overline{\lambda_k} = 2(\lambda_k - \frac{\delta_k}{|p_k|^2})$$

$$\delta_k = -\delta_k + \lambda_k|p_k|^2, \lambda_k = \overline{\lambda_k}$$

Calculate step size:

$$\mu_k = p_k^T r_k, \alpha_k = \frac{\mu_k}{\delta_k}$$

Calculate the comparison parameter:

$$\Delta_k = \frac{2\delta_k[E(w_k) - E(w_k + \alpha_k p_k)]}{\mu_k^2}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

If $\Delta_k \geq 0$ then a successful reduction in error can be made:

$$w_{k+1} = w_k + \alpha_k p_k$$

$$r_{k+1} = -\mathrm{E}'(\mathrm{w}_{k+1})$$

$$\overline{\lambda_k} = 0, \text{success} = \text{true}$$

If k mod N = 0 then restart algorithm: $\mathrm{p}_{k+1} = \mathrm{r}_{k+1}$ else create new conjugate direction:

$$\beta_k = \frac{|\mathrm{r}_{k+1}|^2 - \mathrm{r}_{k+1}\mathrm{r}_k}{\mu_k}$$

$$\mathrm{p}_{k+1} = \mathrm{r}_{k+1} + \alpha_k \mathrm{p}_k$$

If $\Delta_k \geq 0.75$ then reduce the scale parameter: $\lambda_k = \frac{1}{2}\lambda_k$ else a reduction in error is not possible: $\lambda_k = \lambda_k$, success = false

If $\Delta_k < 0.25$ then increase the scale parameter: $\lambda_k = 4\lambda_k$

If the steepest descent direction $\mathrm{r}_k \neq 0$ then set k = k+1 and go to 2 else terminate and return w_{k+1} as the desired minimum

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

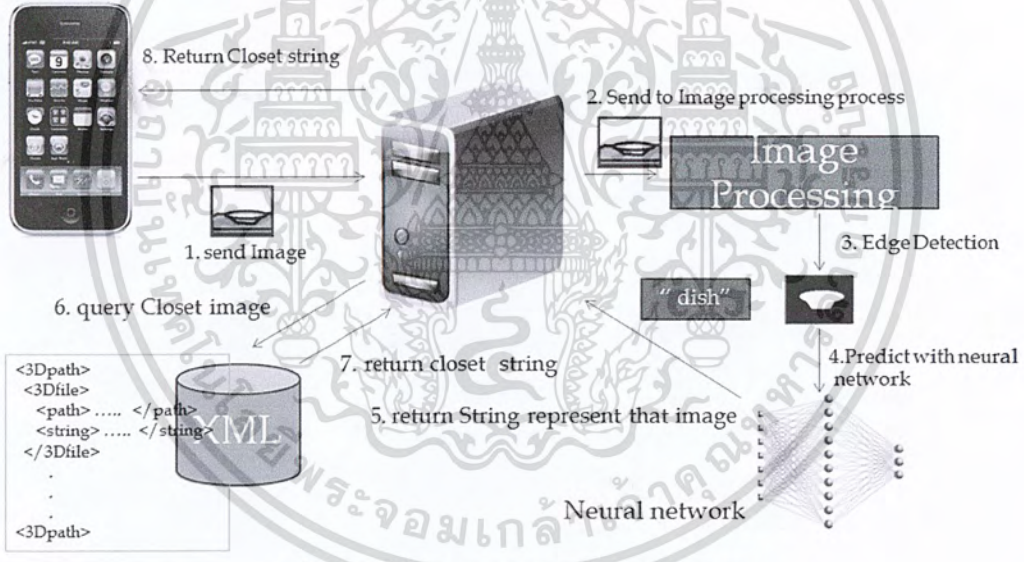
บทที่ 3

การออกแบบและพัฒนา

3.1 บทนำ (Introduction)

โดยขั้นตอนแรกในการออกแบบระบบคือ การศึกษาและจัดเก็บความต้องการของผู้ใช้งาน (User) ก่อนว่าความต้องการของผู้ใช้เป็นอย่างไรบ้าง เพื่อนำมาเป็นส่วนช่วยในการออกแบบตัวโปรแกรม หลังจากขั้นตอนการจัดทำ User Requirement ของระบบแล้ว ก็ทำการการวิเคราะห์และออกแบบระบบซอฟแวร์ด้วยกระบวนการพัฒนาเชิงวัตถุ โดยได้นำภาษาโมเดลมาตรฐาน คือ Unified Modeling Language (UML) มาเป็นภาษาที่ใช้ในการสร้างแบบพิมพ์เขียวให้แก่กระบวนการพัฒนาโปรแกรม ในการพัฒนาโปรแกรมโครงงานนี้ได้เลือกใช้ภาษา C#, objective c ในการพัฒนาระบบซึ่งเป็นภาษาที่มีรูปแบบการเขียนแบบ Object-oriented programming

3.2 องค์ประกอบของระบบ



รูป 3.1 องค์ประกอบและการทำงานของระบบ

3.2.1 องค์ประกอบของระบบ

1) iPhone ทำหน้าที่เป็นตัวถ่ายรูปและส่งรูปไปยัง server

2) เครื่อง Server ให้บริการผ่านทาง HTTP ในการแปลงภาพเป็นภาษามือ

3) Process การทำ Image Processing

4) Process การจัดการ Neural Network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) Process การ Map String เป็น ภาษามือ

3.2.2 การทำงานของระบบ

1) เครื่อง iPhone ทำการถ่ายรูปภาพและส่งรูปภาพไปยัง Server โดยใช้บริการ HTTP Protocol

2) เครื่อง Server ส่งภาพที่ได้ไปทำ Image Processing ภาพ

3) ทำกระบวนการ Edge detection

4) ส่งภาพที่เหลือแต่ขอบไปให้ Neural Network ทำการ Predict

5) Neural Network ส่งผลลัพธ์การ Predict กลับในรูป String

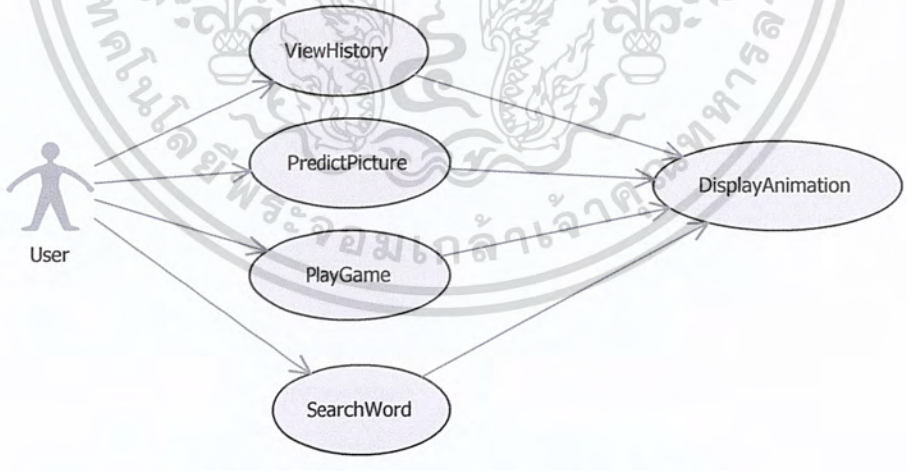
6) ทำการ Map String กับ Animation 3D ใน DB ซึ่งเก็บเป็น XML ในรูปของ Path

7) Return ผลลัพธ์การแปลง String เป็น ภาษามือ

8) ส่งผลลัพธ์กลับไปให้ iPhone

3.3 ยูสเคสไดอะแกรม (Use Case Diagram)

จากฟังก์ชันการทำงานของระบบ เป็นการออกแบบระบบของระบบส่งเสริมการเรียนรู้ศัพท์ภาษามือไทยด้วยไอโฟน(Thai Sign Language Learning System With iPhone) ด้วย ยูสเคสไดอะแกรม (Use Case Diagram) สามารถบอกความสัมพันธ์ระหว่างผู้ใช้งานกับระบบ โดยผู้ใช้งานสามารถที่จะแปลงรูปภาพเป็นภาษามือได้ สามารถค้นหาคำศัพท์ได้ สามารถค้นหาคำศัพท์ภาษามือ และสามารถเล่นทายคำถาม ดังรูป 4.1



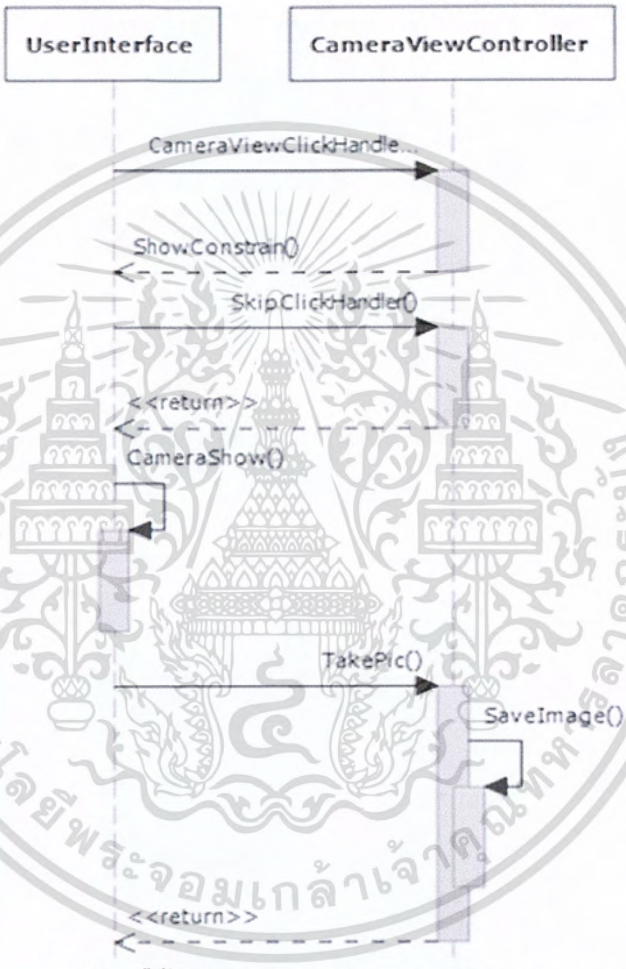
รูป 3.2 แผนภาพ Use case diagram ของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 ซีเควนไดอะแกรม (Sequence Diagram)

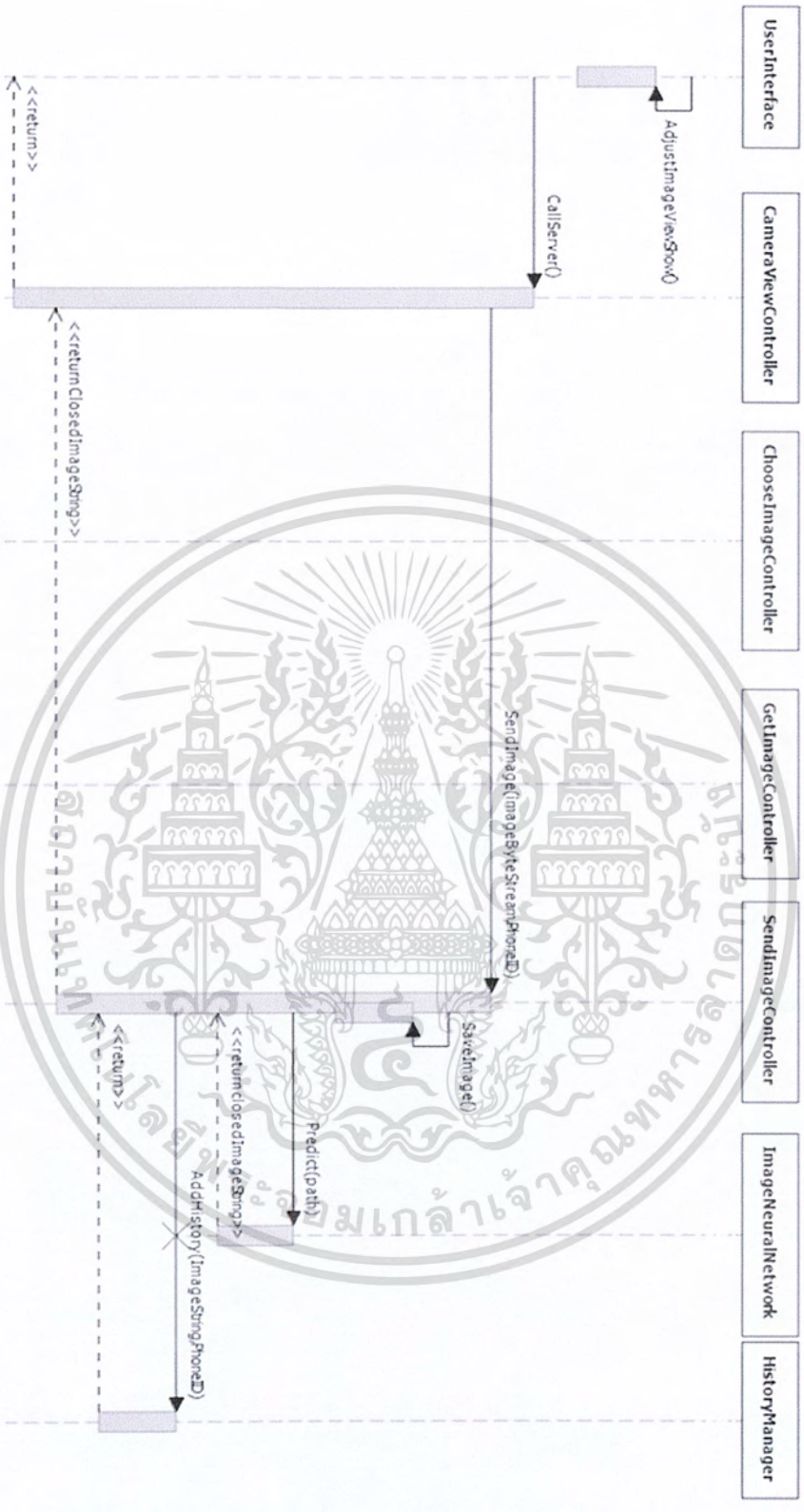
แผนภาพซีเควนไดอะแกรม(Sequence Diagram) เป็นแผนภาพที่ใช้แสดงปฏิสัมพันธ์ (Interaction) ระหว่างออบเจ็กต์ตามลำดับของเหตุการณ์ที่เกิดขึ้น โดยเป็นการแสดงขั้นตอนการทำงานของแต่ละยูสเคส (Use case) ในระบบที่ออบเจ็กต์ต่างๆส่งข้อความถึงกันและกัน โดยมีซีเควนของระบบดังนี้

3.4.1 การแสดงภาษามือไทยจากรูป (Convert Picture To Sign)



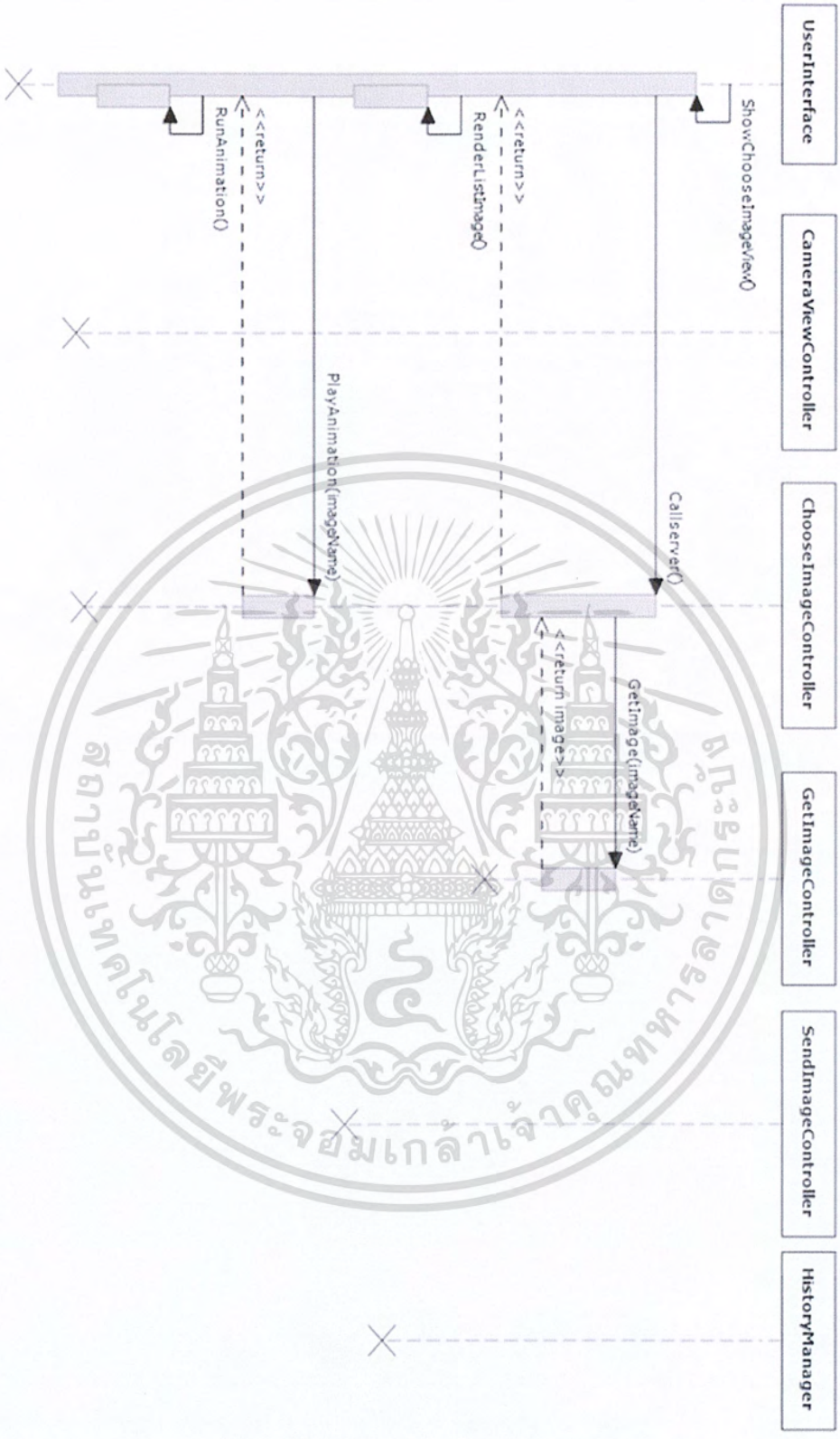
รูป 3.3 แผนภาพการแสดงภาษามือไทยจากรูป (Convert Picture To Sign)ของระบบ ส่วนที่1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.4 แผนภาพการแสดงภาษามือไทยจากรูป (Convert Picture To Sign)ของระบบ ส่วนที่2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

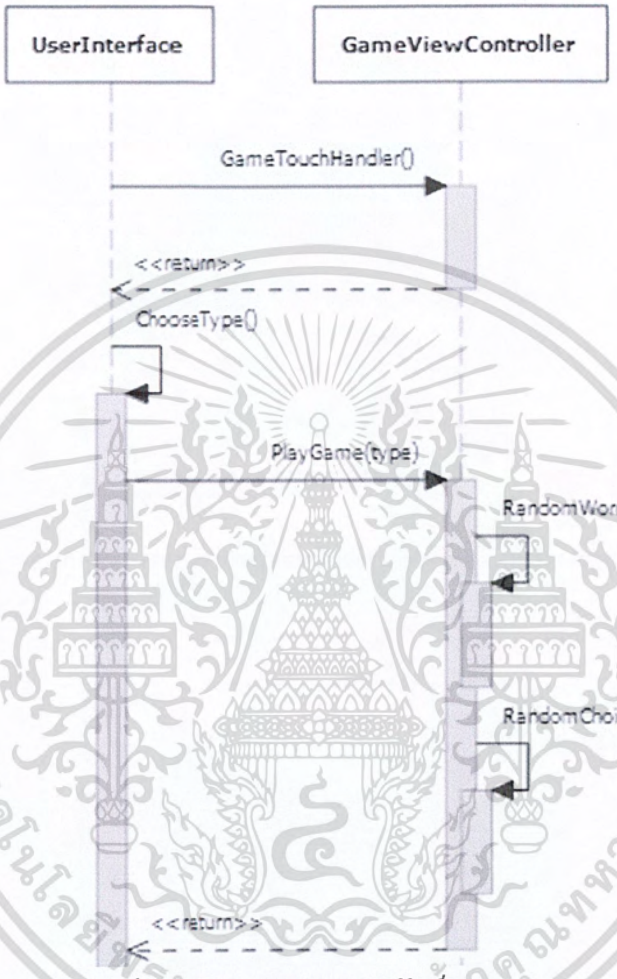


รูป 3.5 แผนภาพการแสดงภาษามือไทยจากรูป (Convert Picture To Sign)ของระบบ ส่วนที่3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

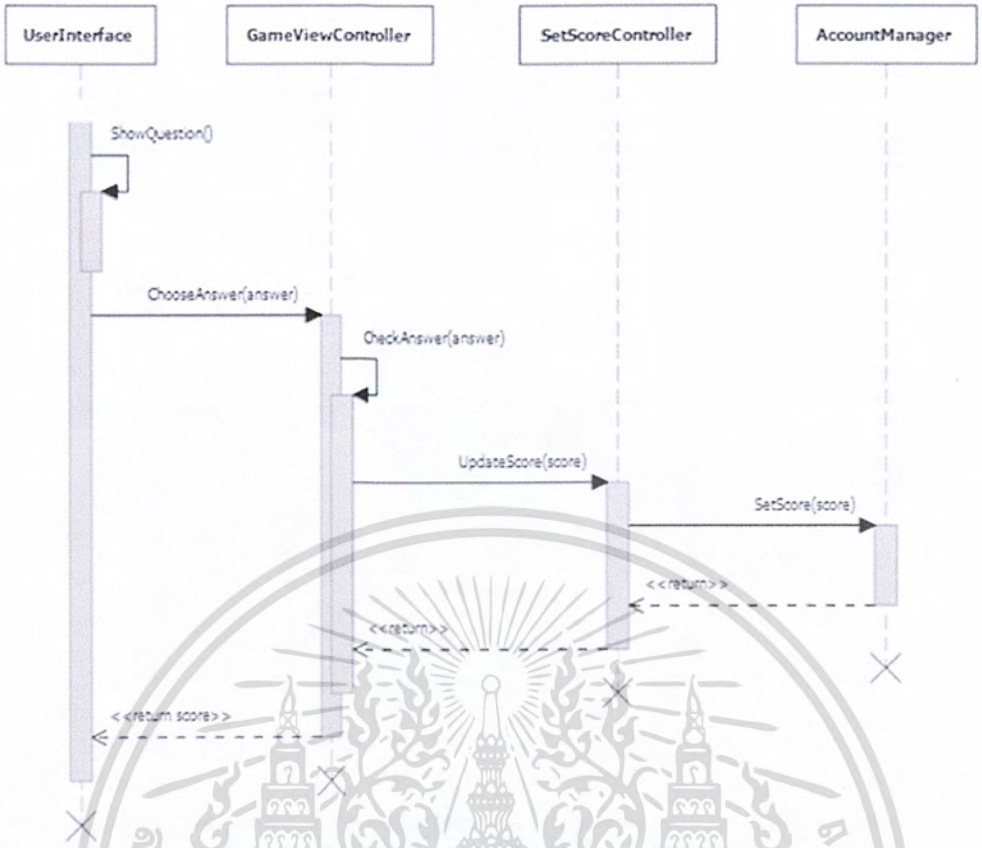
ระบบจะทำการรับรูปภาพมาแล้วผ่านการตัดแต่งภาพบน iPhone จากนั้นส่งมาให้ Server ทำ Image Processing แล้วส่งไป Predict ด้วย Neural Network ส่งผลลัพธ์กลับไปให้ iPhone เลือกภาพที่ใกล้เคียงที่สุดแล้วแสดงภาษามือของภาพนั้น

3.4.2 การเล่นทายคำศัพท์



รูป 3.6 แผนภาพเกมทายคำศัพท์ของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

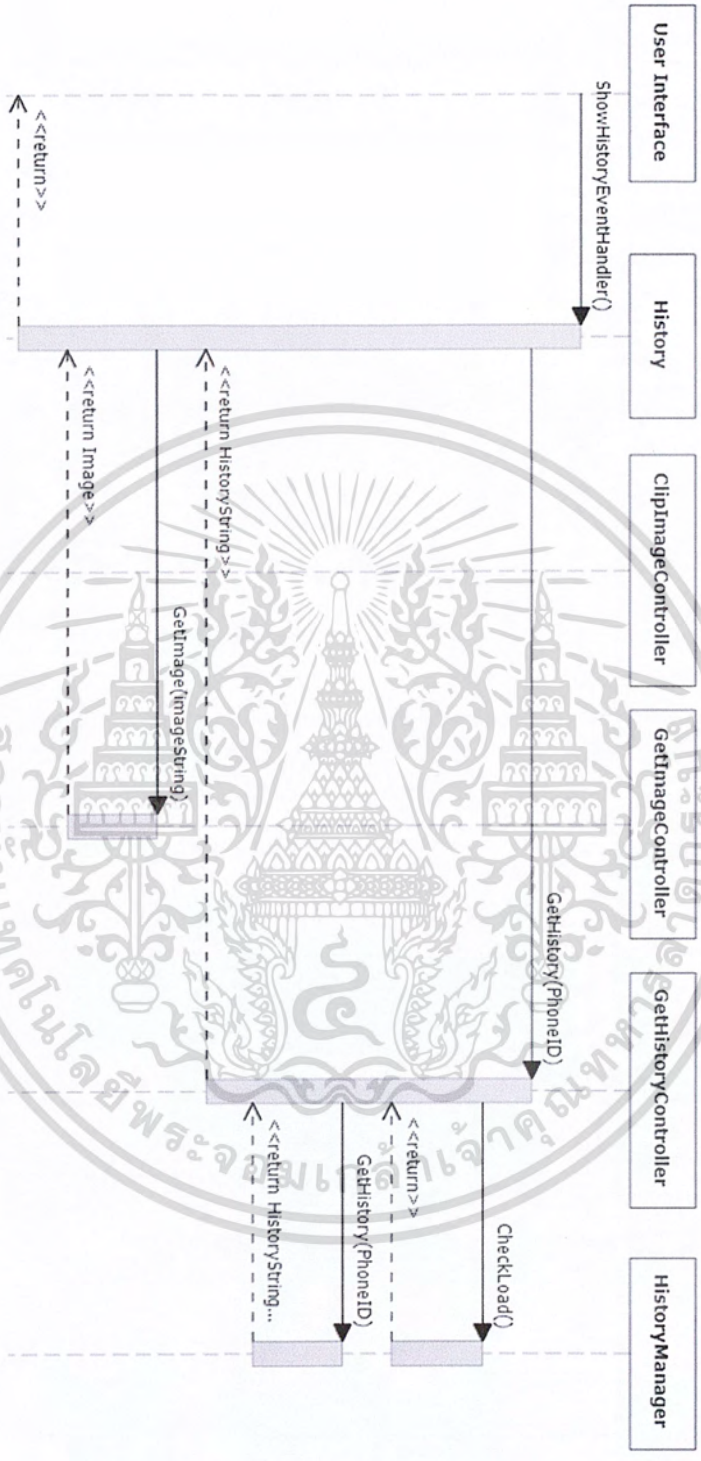


รูป 3.7 แผนภาพเกมทายคำศัพท์ของระบบ(ต่อ)

ระบบจะให้ผู้ใช้เลือกว่าจะเล่นแบบไหนหลังจากนั้นจะทำงานในโหมดนั้นๆโดยจะทำการสุ่มคำถามและคำตอบให้ผู้ใช้เลือกหลังจากผู้ใช้เลือกแล้วจะทำการตรวจสอบคำตอบและ Update ค่าคะแนนรวมของคนๆนั้นใน Database แล้วแสดงคะแนนรวมให้ผู้ใช้

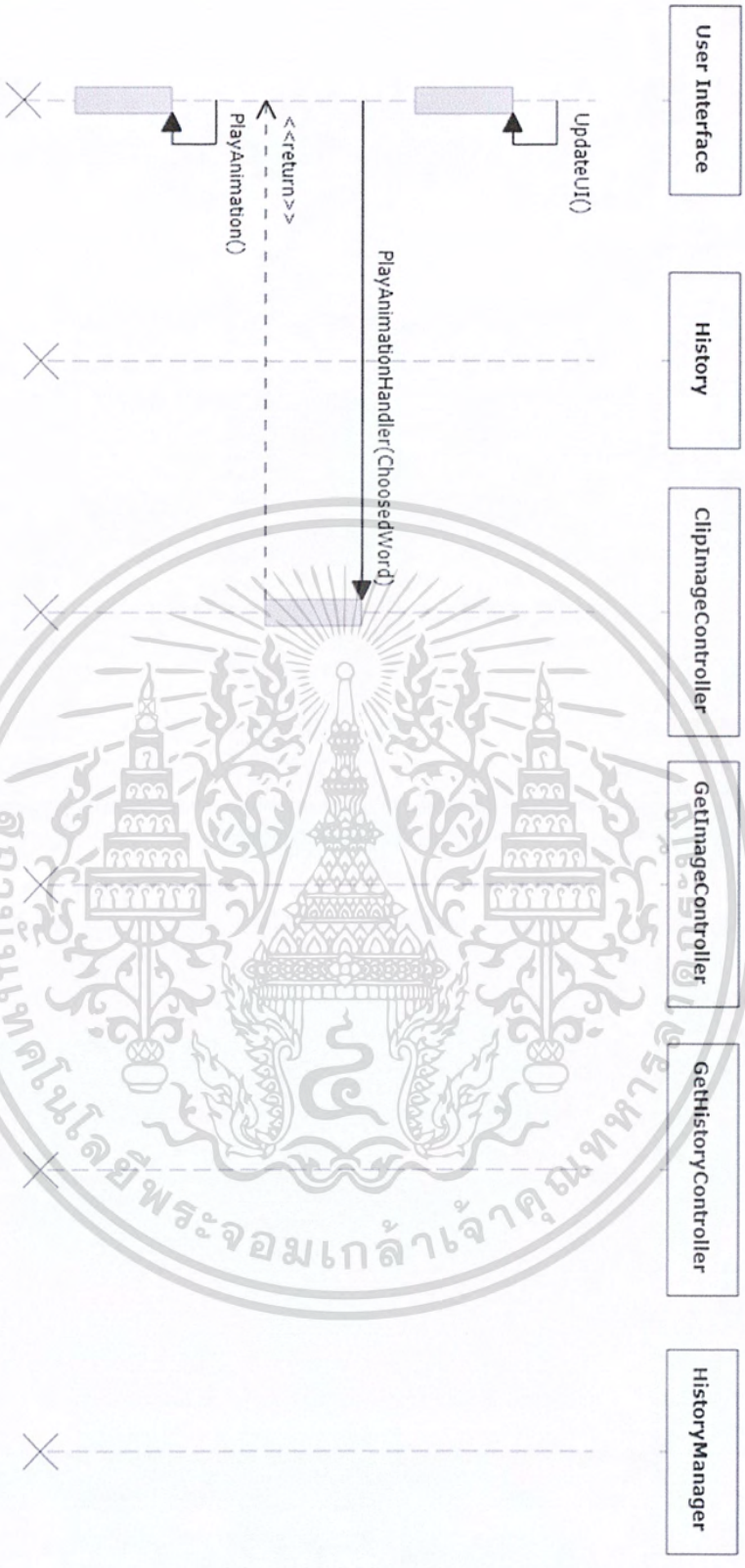
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.3 การจัดเก็บประวัติการถ่าย (Picture History)



รูป 3.8 แผนภาพการจัดเก็บประวัติการถ่าย (Picture History)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

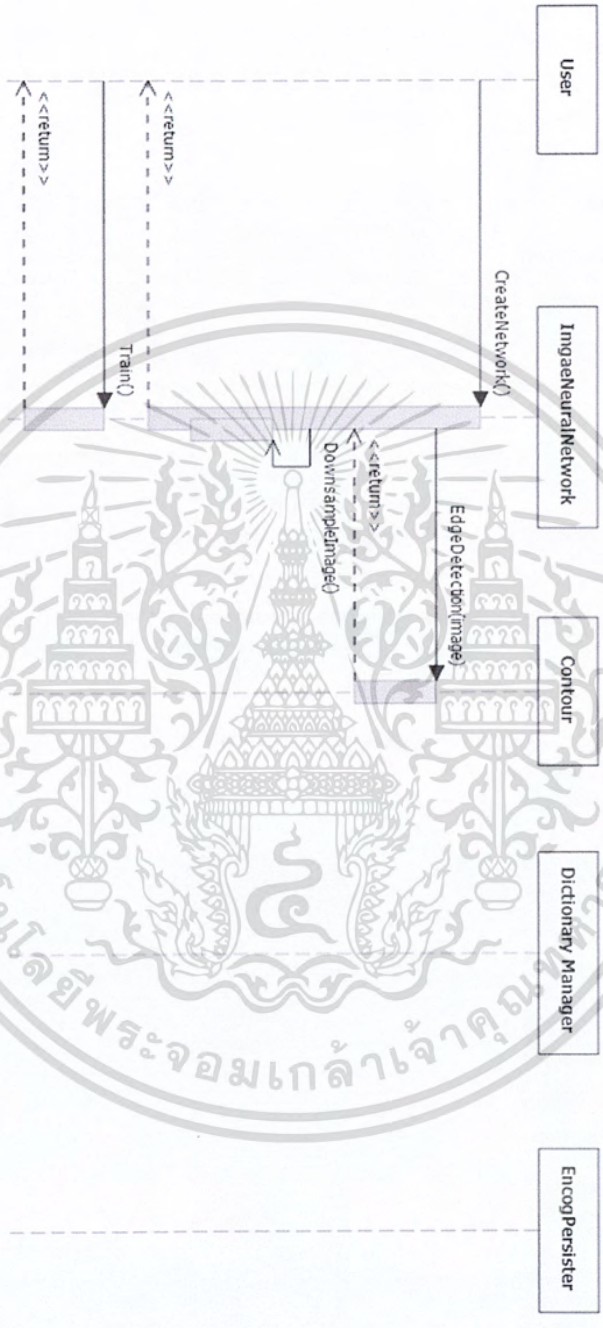


รูป 3.9 แผนภาพการจัดเก็บประวัติการถ่าย (Picture History)(ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

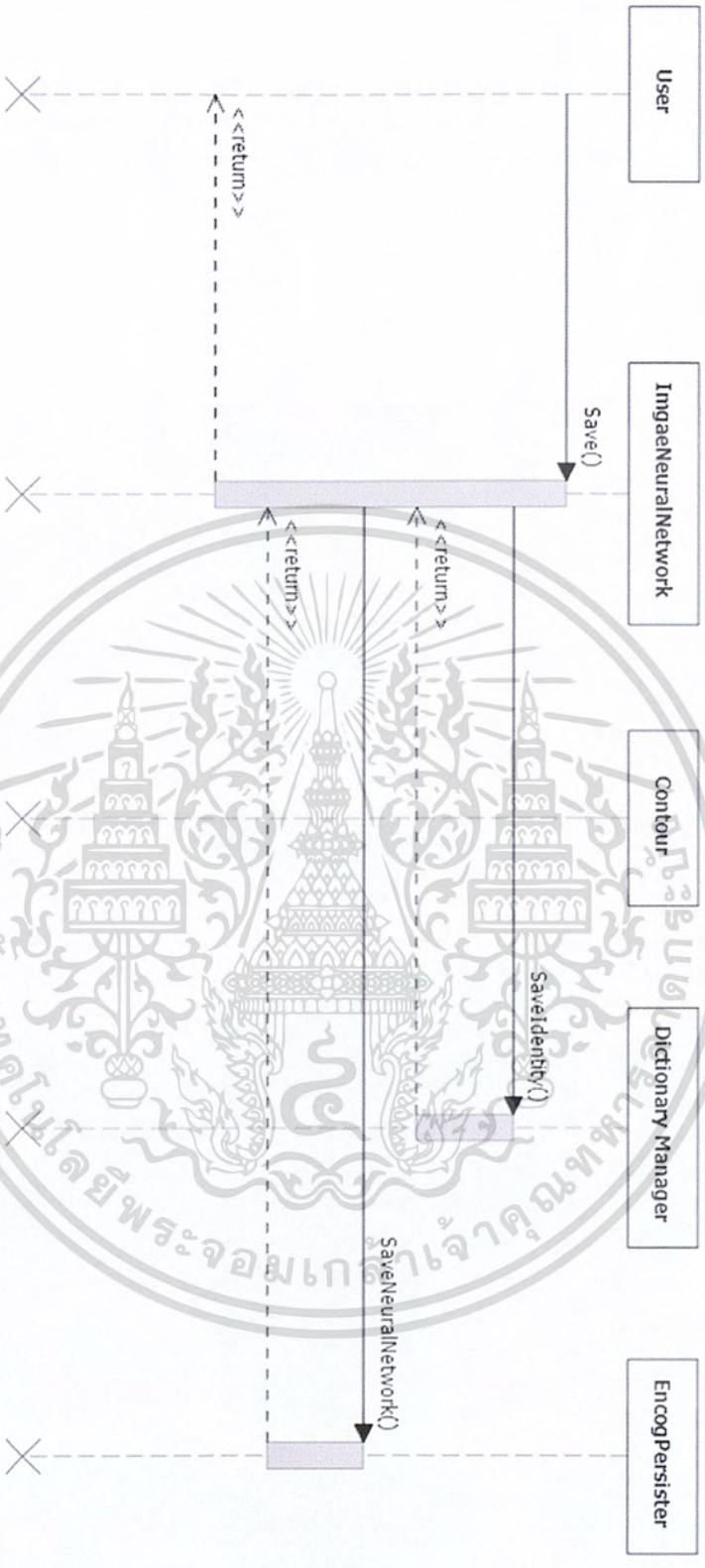
ระบบจะทำการดึงข้อมูลรูปที่เคยถ่ายไว้ในdatabase มาแสดงให้ผู้ใช้ดูเมื่อผู้ใช้ต้องการทบทวนภาษามือของคำๆนั่นก็ทำการแสดงภาษามือให้ผู้ใช้ดู

3.4.4 การฝึกข่ายใยประสาทเทียมของระบบ (Neural Network Training)



รูป 3.10 แผนภาพการฝึกข่ายใยประสาทเทียมของระบบ (Neural Network Training)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

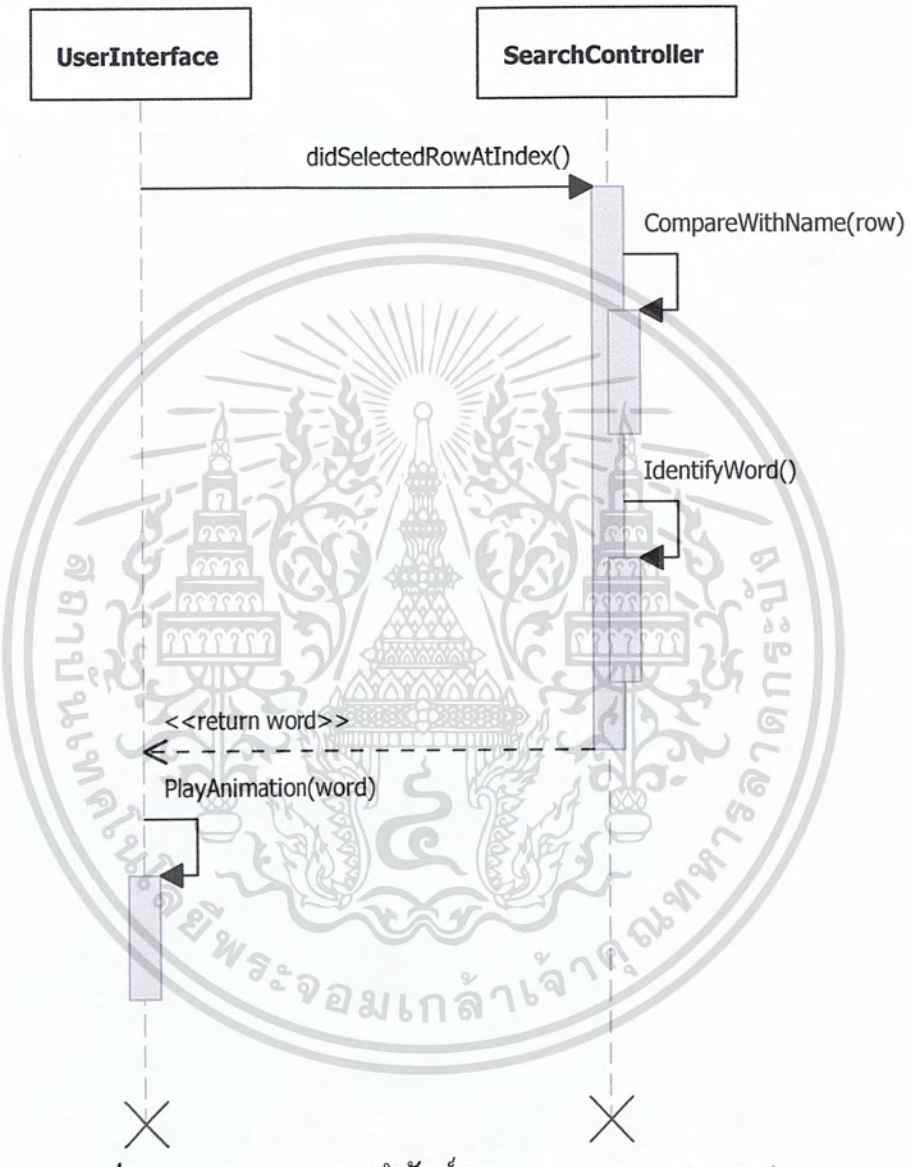


รูป 3.11 แผนภาพการฝึกข่ายใยประสาทเทียมของระบบ (Neural Network Training)(ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบจะทำการสร้างโครงสร้างข่ายใยประสาทหลังจากรูปที่ ได้ทำการเพิ่มไว้แล้วล่วงหน้า แล้วมาทำกระบวนการหาขอบรูปของแต่ละรูปแล้วนำมารู้จำรูปโดยใช้ข่ายใยประสาทเทียมแล้วทำการบันทึกโครงสร้างที่ได้รับการรู้จำแล้วเพื่อมาใช้ในการทำนายต่อไป

3.4.5 การแสดงคำศัพท์ภาษามือ (Sign Language Performing)



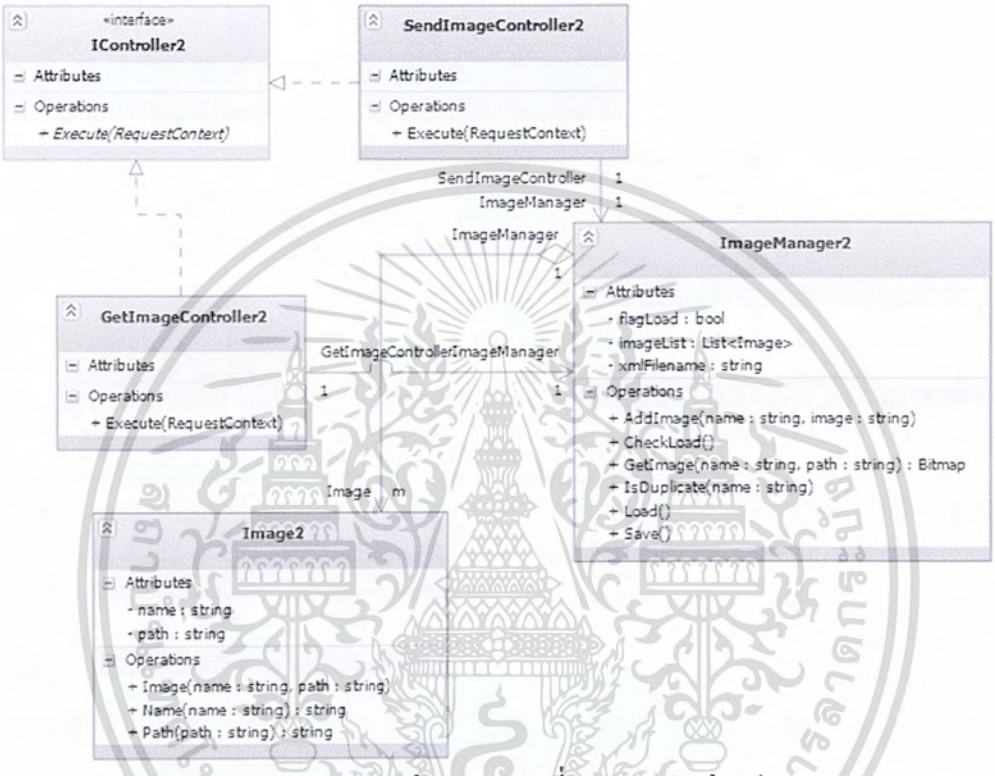
รูป 3.12 แผนภาพการแสดงคำศัพท์(Sign Language Performing)

ผู้ใช้จะทำการเลือกคำศัพท์ที่ต้องการหลังจากนั้นจะมาเทียบกับ Database ภาษามือแล้วนำมาแสดงตามต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

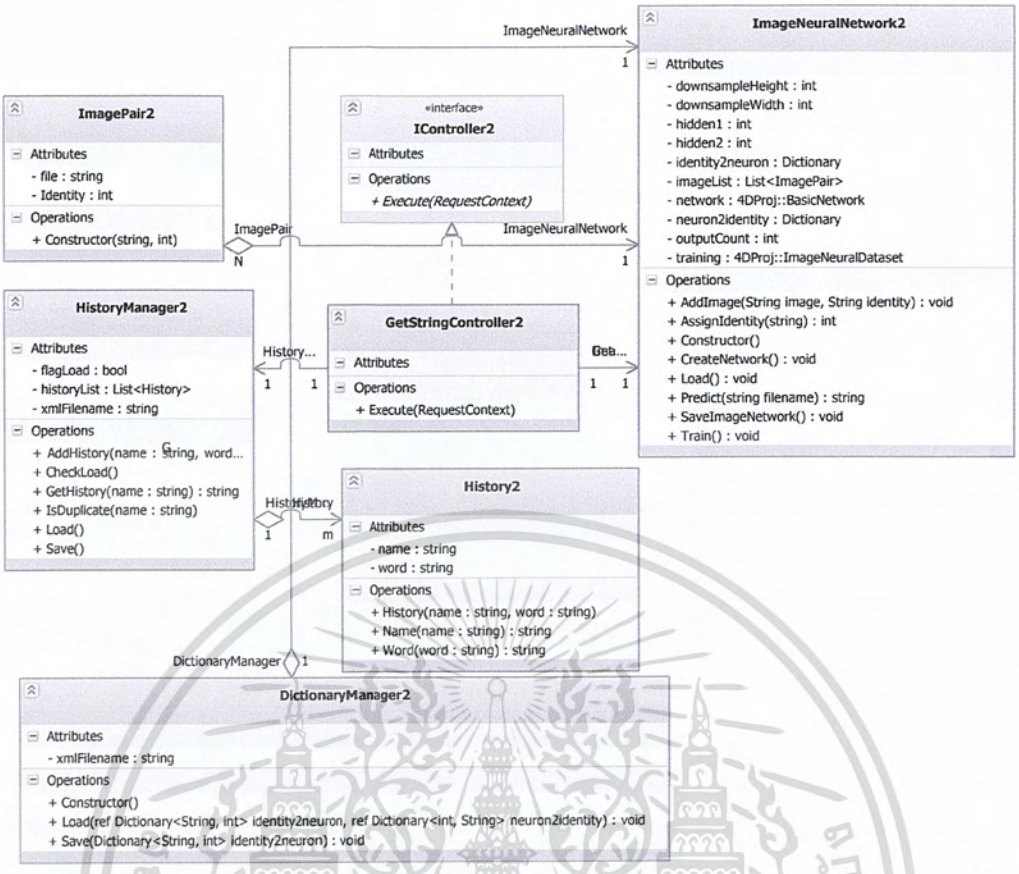
3.5 คลาสไดอะแกรม (Class Diagram)

คลาสไดอะแกรม (Class diagram) ประกอบด้วยคลาส (Class) และความสัมพันธ์ระหว่างคลาส ที่สามารถแสดงรายละเอียดได้ว่ามีเมธอดและแอตทริบิวต์อย่างไร โดยในรูป นี้ได้แสดงแผนภาพคลาสไดอะแกรมของระบบแสดงภาษามือไทยด้วยตัวละครแอนนิเมชัน 3 มิติ โดยประกอบไปด้วยคลาสต่างๆ ดังรูป



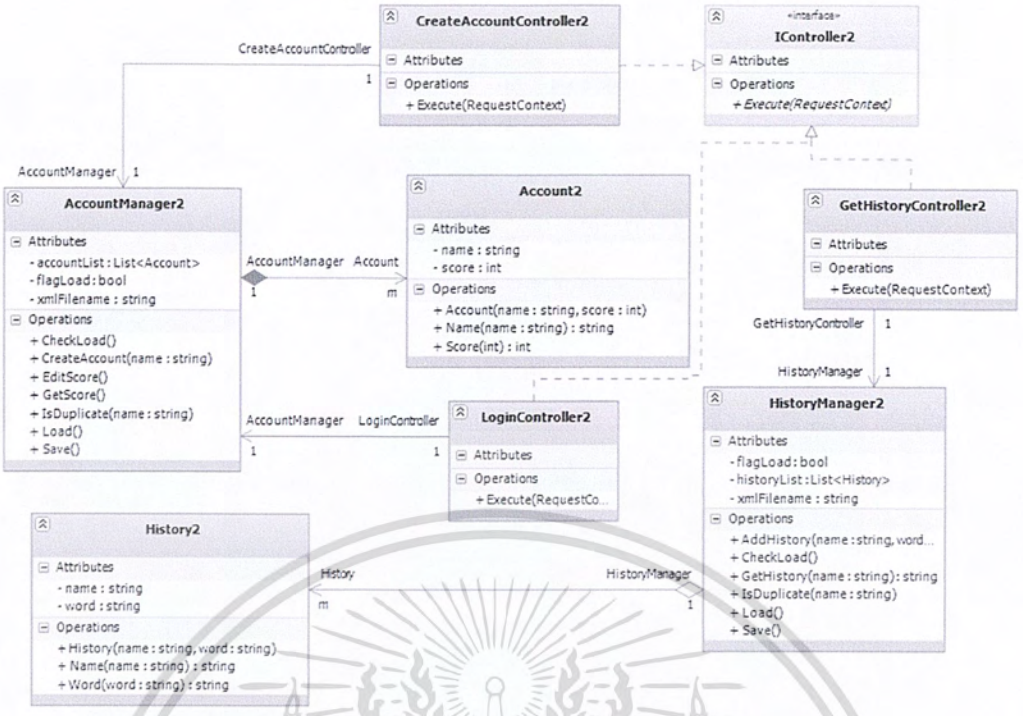
รูป 3.13 Class Diagram ส่วน Server เกี่ยวกับการรับส่งรูปของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



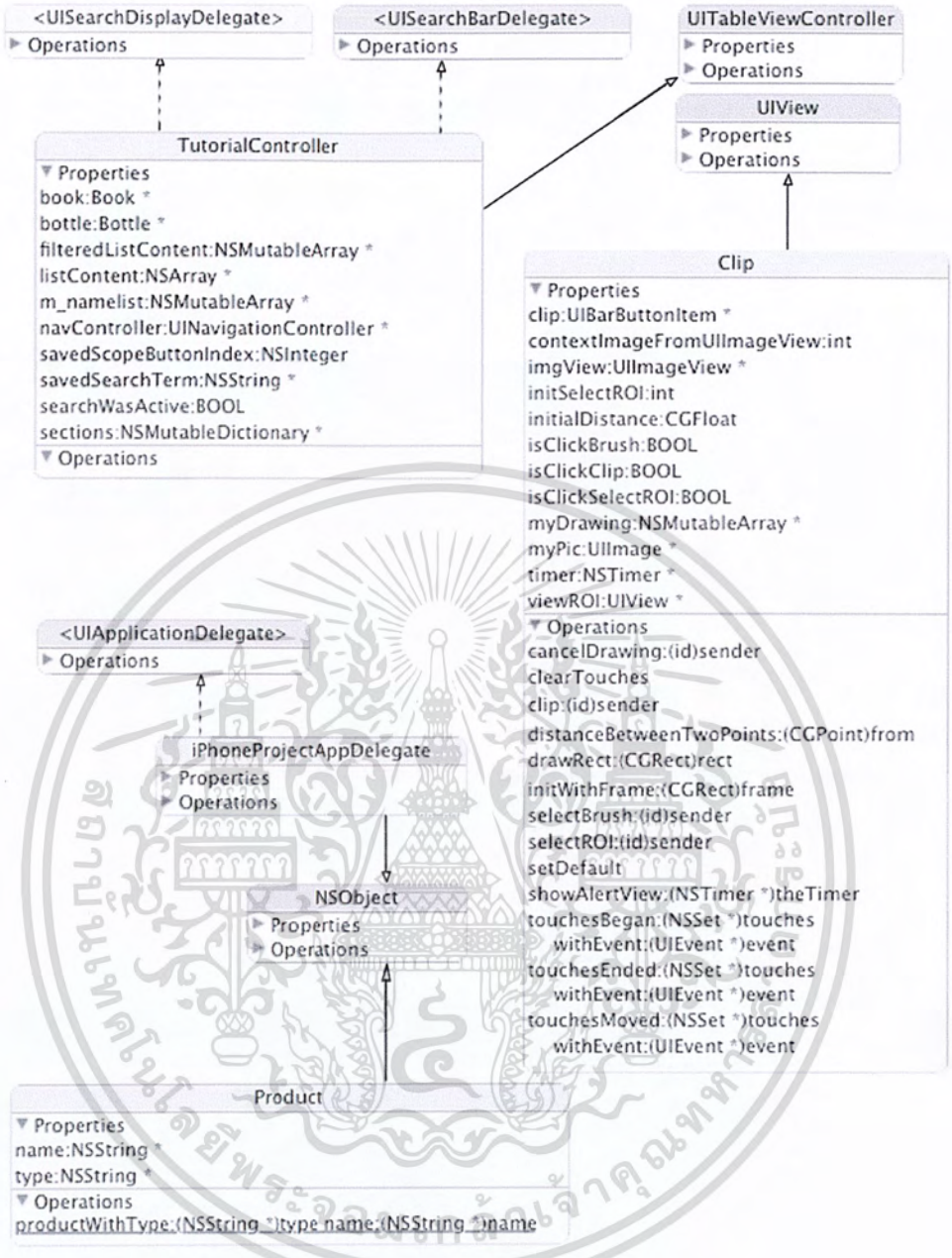
รูป 3.14 Class Diagram ส่วน Server เกี่ยวกับการทำนายของ Neural Networkของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



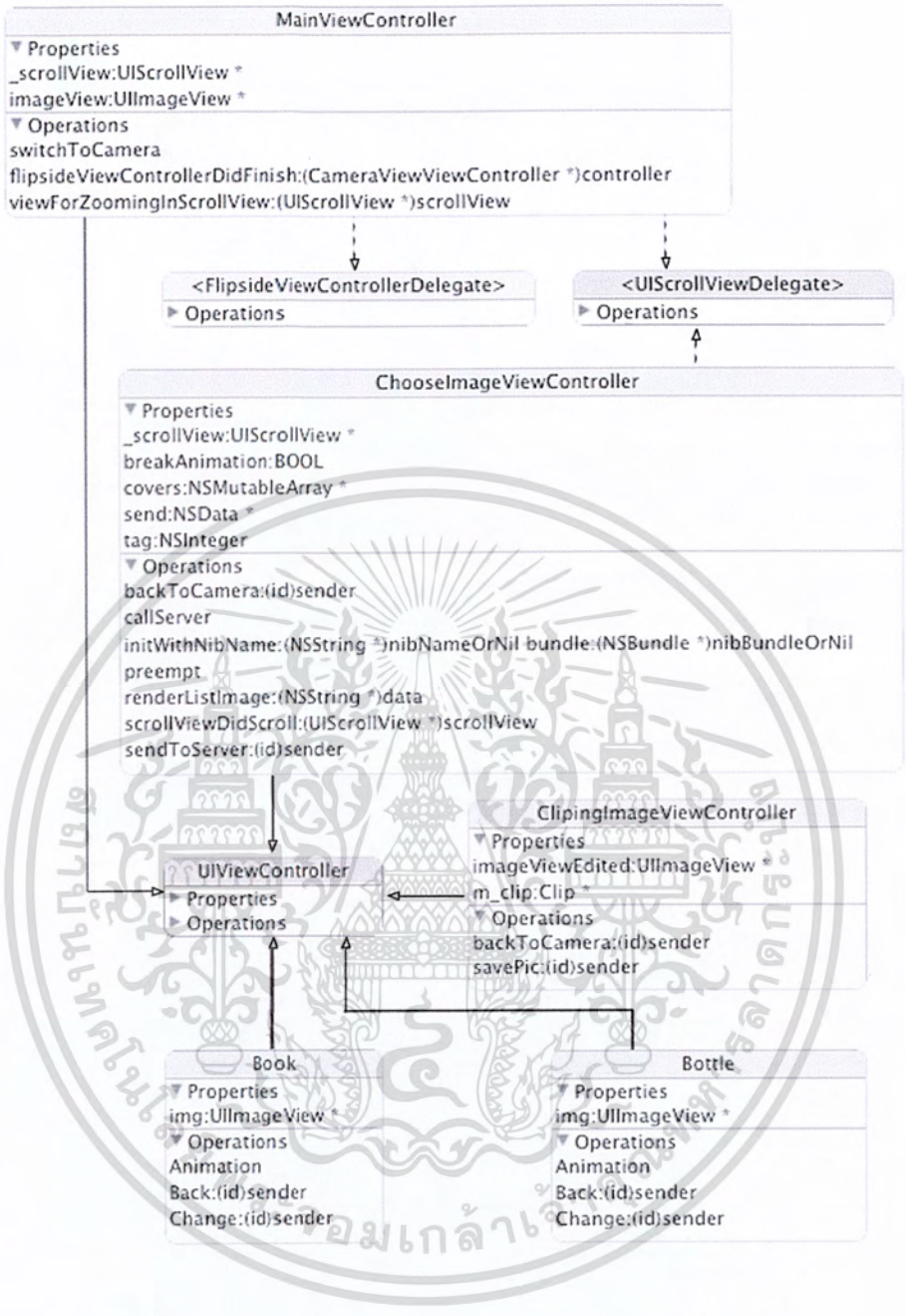
รูป 3.15 Class Diagram ส่วน Server ในการจัดการ History และ Accountของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



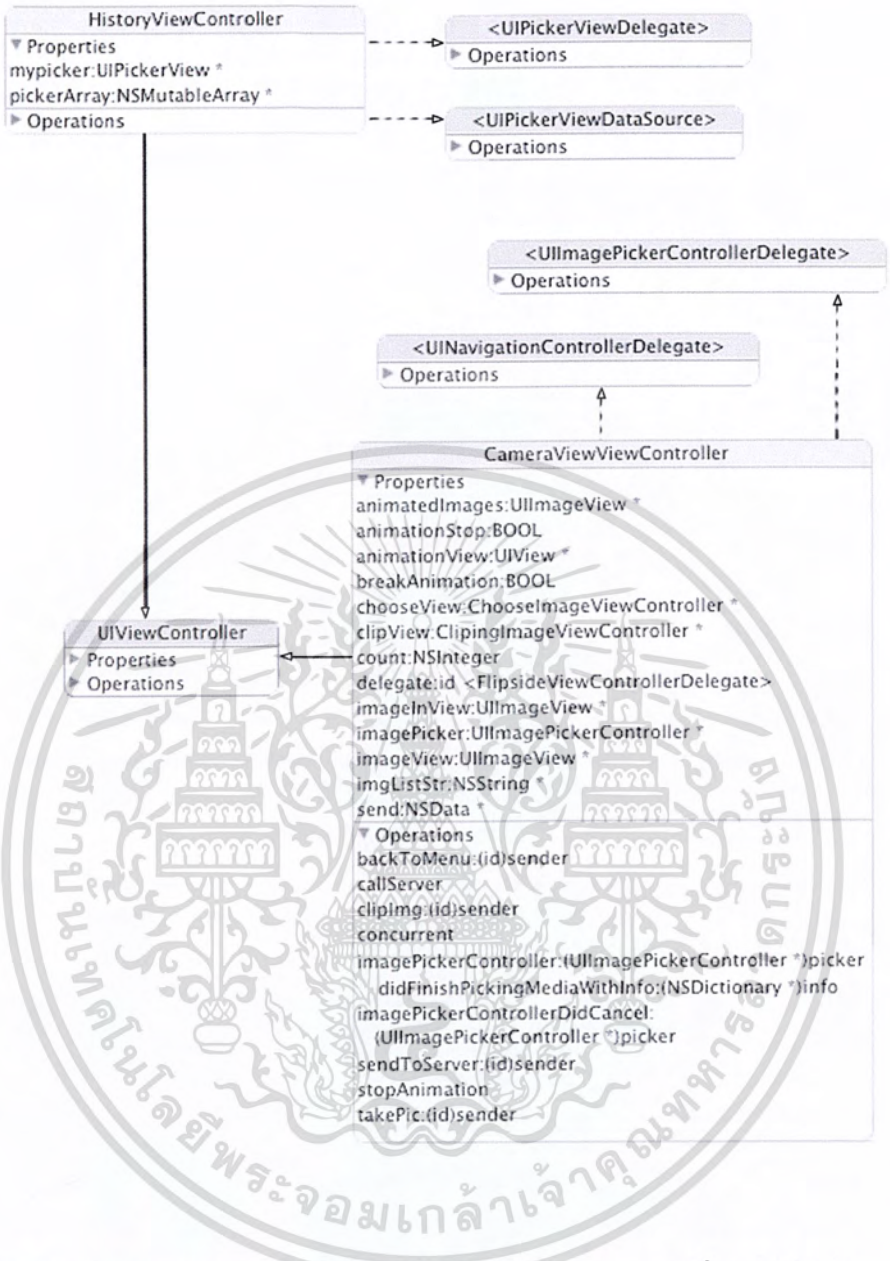
รูป 3.16 Class Diagram ส่วน iPhone ในการค้นหาคำศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.17 Class Diagram ส่วน iPhone ในการควบคุมการถ่ายรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.18 Class Diagram ส่วน iPhone ในการควบคุมการแสดงคำที่เคยถ่ายไว้แล้ว

3.6 XML Schema

ระบบใช้ XML Schema ในการเก็บข้อมูลของระบบโดยเก็บประวัติการถ่ายรูป,phoneID,ดัง แสดงในรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<?xml version="1.0" encoding="utf-8"?>
<UserAccount>
  <Account>
    <name> ..... </name>
    <word> ..... </word>
  </Account>
</UserAccount>
```

รูป 3.19 XML Schema ของ History แต่ละ User

```
<?xml version="1.0" encoding="utf-8"?>
<UserAccount>
  <Account>
    <name>aee1fc7a1c5e140dce3c1530cfb2bd3863db8fca</name>
    <score>0</score>
  </Account>
</UserAccount>
```

รูป 3.20 XML Schema ของ Account แต่ละ User

3.7 การนำไปใช้

ระบบ Auto-detect Object แบ่งเป็น 2 ส่วน iPhone และ Server

รูป 3.21 กรอบรูปบน iPhone

3.7.1 ในส่วนของ iPhone

1) สร้าง application ที่ใช้ในการถ่ายภาพโดยเพิ่ม feature ในส่วนที่เป็นกรอบสีแดง ดังรูป ส่วนนี้เอาไว้สำหรับระบุว่าอะไรคือ object ที่ user สนใจในภาพนั้นๆ นั่นคือ ถ้า user สนใจ object ใดต้องมี object นั้นอยู่ในกรอบสีแดง

2) เพิ่ม function ในการ resize และตัดภาพ เนื่องจากภาพถ่ายที่ได้จาก iPhone มีขนาด 2048x1536 pixel ซึ่งใหญ่มากไม่เหมาะที่จะส่งไปประมวลผล จึงทำการ resize เหลือขนาด 400x300 pixel แล้วตัดภาพให้ได้ขนาด 300x300 pixel เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การตัดภาพในส่วนที่ไม่จำเป็นทิ้งไปจะได้ขนาดประมาณกรอบสีเทาดังรูป ด้านบน แล้วส่งไปประมวลผลที่ server

ข้อจำกัดในการถ่ายรูป

1) พื้นหลังกับวัตถุต้องมีสีคนละโทน(มืด,สว่าง)

2) วัตถุที่สนใจต้องอยู่ในกรอบสีแดง

3) ภาพที่ถ่าย ไม่ควรมีแสงสะท้อนจากหลอดไฟ

3.7.2 ในส่วนของ server(กรณี user ถ่ายรูปโดยทำตามขั้นตอนที่ทางเราได้กำหนดไว้)

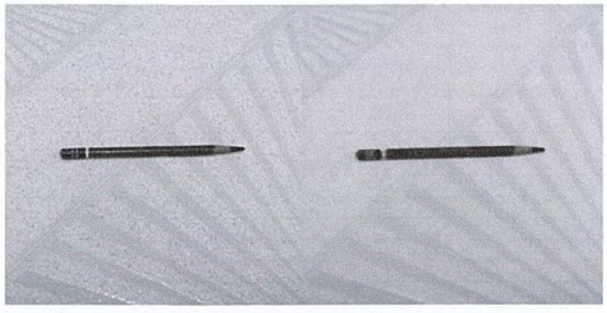
Auto-detect object เป็น function ที่ให้ server ประมวลผลภาพแล้ววิเคราะห์แยกแยะ หาวัตถุเอง ซึ่งจะใช้วิธีเปรียบเทียบความเข้มแสงของภาพระหว่างพื้นที่ในกรอบสีแดงและนอกกรอบสีแดงที่มา จาก iPhone โดยมีขั้นตอนดังนี้

1) ทำการแปลงภาพสีที่รับเข้ามาเป็นภาพขาว-ดำ(gray image) เพื่อให้ง่ายต่อการแยกแยะวัตถุกับ background (เนื่องจากภาพสีมีความเข้มแสงหรือจำนวนสีที่มากกว่าภาพขาว-ดำมากนัก การเปรียบเทียบด้วยภาพสี โอกาสที่จะพบความเข้มแสงที่ตรงกันมีน้อยมาก ฉะนั้นภาพสีจึงไม่เหมาะที่นำมาเปรียบเทียบ)

รูป 3.22 ตัวอย่างภาพผลลัพธ์หลังการเปรียบเทียบความเข้มแสง

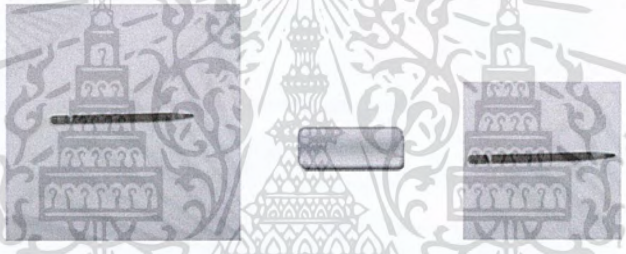
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) ทำ smoothing ด้วย median filter เพื่อลดรายละเอียดในภาพลง



รูป 3.23 แสดงภาพ Smoothing ด้วย Median Filter

3) แยกภาพพื้นที่ที่อยู่ในกรอบสีแดง(ภาพวัตถุ) กับนอกกรอบสีแดง(ขอบภาพ)ออกจากกันด้วยการนำภาพ gray image ที่ยังไม่แยกวัตถุกับ background มาลบ (sub)ด้วยภาพวัตถุ



รูป 3.24 การตัดรูปที่อยู่ในกรอบสีแดง

เราจะได้ขอบภาพออกมา ส่วนที่ถูกลบออกไป จะเป็นสีดำ (ความเข้มแสง = 0)

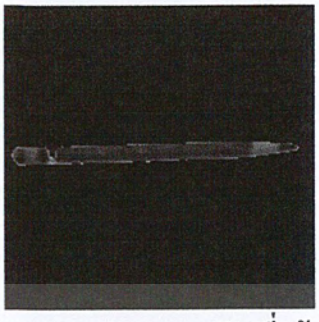
รูป 3.25 ตัวอย่างภาพขอบภาพ

4) เก็บความเข้มแสงระดับต่างๆของขอบภาพ และภาพวัตถุเก็บลง array คนละตัว

5) นำความเข้มแสงของขอบภาพ และภาพวัตถุมาเปรียบเทียบกันพร้อมกับ check เป็น percent ว่าความเข้มแสงตรงกันกี่ percent กรณีความเข้มแสงตรงกันให้เปลี่ยน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความเข้มแสงบริเวณนั้นเป็นสีดำ(ความเข้มแสงเป็น 0) ทำแบบนี้ไปเรื่อยๆ จนครบทุก pixel ในภาพวัตถุ เมื่อครบทุก pixel เราจะได้ภาพวัตถุที่แท้จริงออกมา



รูป 3.26 ภาพกรอบของวัตถุที่แท้จริง

6) จากการทดลอง ค่า percent ที่ได้มีผลกับค่า thresholdingที่ใช้แยกแยะระหว่างวัตถุกับ background ในลักษณะความสัมพันธ์ที่สวนทางกัน

รูป 3.27 thresholdingที่ใช้แยกแยะระหว่างวัตถุกับ background

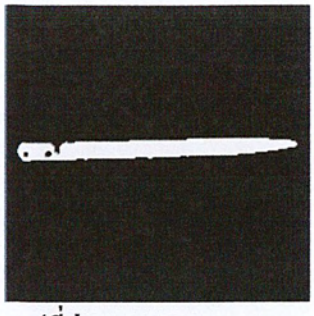
ยิ่ง percent มีค่ามากค่า threshold ก็ยิ่งน้อยลงโดยจะดูจากค่ากลางของ percent ก็คือ 50 ถ้า 50% ค่า threshold จะเท่ากับ 127 ถ้ามากกว่า 50% อยู่กี่ percent ก็ไปลบออกจาก 127 ตามสัดส่วนที่ควรเป็น(Diff) แต่ถ้าน้อยกว่า 50% ก็บวกเพิ่มจาก 127 ตามสัดส่วนที่ควรเป็นตามสมการด้านล่าง

*Diff={(percent-50)*127}/50*

$$threshold = \begin{cases} 127 - Diff & for\ Diff < 0 \text{หรือ} Diff > 0 \text{แต่ไม่เท่ากับ } 50 \\ 127 & for\ Diff = 50 \end{cases} \quad (3.1)$$

7) ทำ thresholdingด้วยค่า threshold ที่หามาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.28 รูปที่ผ่านการทำ Thresholding แล้ว

8) ทำ smoothing แบบ guassian อีกรอบเพื่อป้องกัน noise ที่อาจจะเกิดจากการทำ thresholding

9) กรณีความเข้มแสงต่ำสุด(ไม่นับความเข้มแสงเท่ากับ 0)ในภาพวัตถุมีค่ามากกว่า 127 ไม่สามารถคำนวณตามสมการด้านบนได้ จะทำให้ได้ค่า threshold ที่ผิดพลาด เนื่องจาก ณ ความเข้มแสง 128 หากมีค่าน้อยกว่านี้ถือเป็นความเข้มแสงที่ค่อนข้างต่ำ แต่ถ้ามากกว่า 128 ถือว่าเป็นความเข้มแสงที่มีค่าค่อยข้างสูง ดังนั้น ความเข้มแสงต่ำสุดที่มากกว่า 127 ถือเป็นความเข้มแสงที่สูง จึงนำเอาค่าความเข้มแสงดังกล่าว มาลบออกจาก 128 จะได้ค่า threshold ที่ควรเป็น

3.8 User Interface

เป็นการแสดง View ต่างๆใน Application และการทำงานของ Application

3.8.1 หน้าจอหลักของ iPhone

คลิกที่ Icon ชื่อ TSL เพื่อเข้าใช้งาน Application

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 3.29 หน้าจอ iPhone เพื่อแสดง Icon ของโปรแกรม

3.8.2 History View

รูป 3.30 หน้าจอ History View

แสดง History View หรือ หน้าจอที่เก็บรูปภาพที่ได้เคยถ่ายไว้แล้วในอดีตโดยนำมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โชว์ในรูปแบบ Scroll และสามารถกด Play เพื่อแสดงภาษามือ

3.8.3 Tutorial View

Tutorial เป็นเสมือน Dictionary ที่จะช่วยในการ แปลจากภาษาไทยเป็นภาษามือๆ

หน้าจอหลักจะแสดง List ของคำศัพท์ทั้งหมดที่สามารถแสดงเป็นภาษามือได้ โดยสามารถทำการ Search คำ และสามารถ Search เป็นกลุ่มของคำ

รูป 3.31 หน้าจอ Tutorial View

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

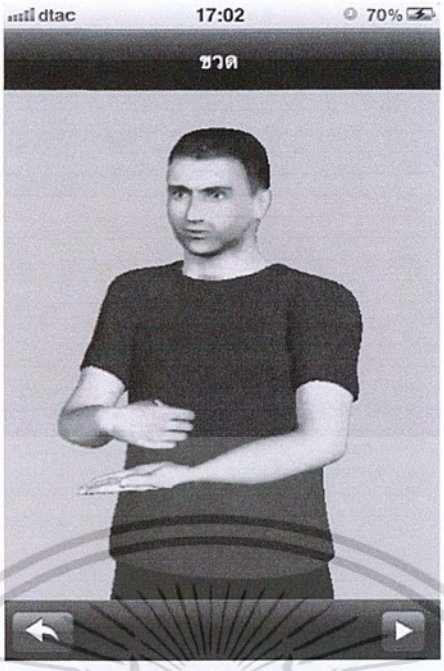
รูป 3.32 หน้าจอการ Search โดย Search ด้วยกลุ่มของคำ

เมื่อทำการ Search คำได้แล้ว ก็จะไปแสดงรูปของสิ่งของที่เราได้ใช้คำศัพท์ Search

รูป 3.33 หน้าจอรูปของคำศัพท์

เมื่อทำการกด เล่น ก็จะมาโชว์ศัพท์ภาษามือด้วย Animation 3D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.34 ภาษามือด้วย Animation 3D

3.8.4 Camera View

โดย View หลักนั้นจะมีการแสดงกฎการถ่ายรูปและความสามารถในการถ่ายภาพด้วย

รูป 3.35 Camera View หน้าจอแสดงกฎและความสามารถในการถ่ายรูป

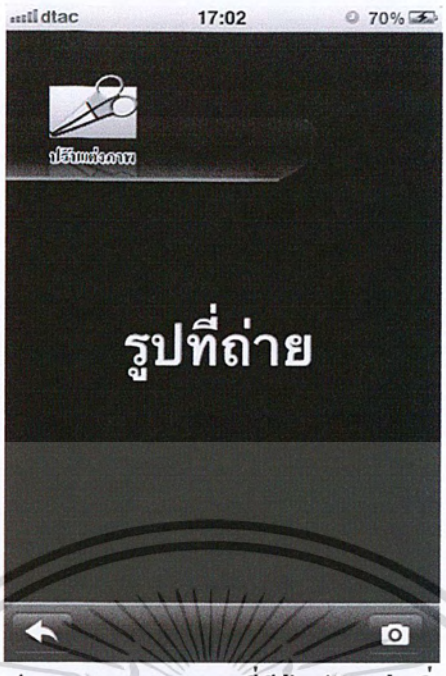
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเข้ามายัง Camera View ถัดไปเราจะต้องกดที่รูปกล้องถ่ายรูปเพื่อทำการถ่ายรูปก่อน

รูป 3.36 Camera View ก่อนการถ่ายรูป

เมื่อเราถ่ายรูปเสร็จก็จะมายัง View เดิมโดยจะมีปุ่มให้กดเพื่อทำการตกแต่งรูปให้ตรงตามกฎ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



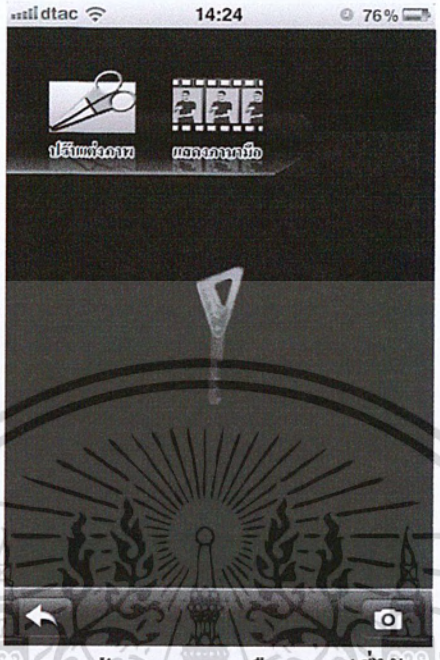
รูป 3.37 Camera View ที่มีปุ่มปรับแต่งเพิ่ม

เมื่อเข้าไปยังหน้าจอการปรับแต่งเพิ่ม จะต้องทำการปรับแต่งรูปให้ตรงตามกฎที่ได้ตั้งไว้ แล้วทำการบันทึกรูป

รูป 3.38 หน้าจอการปรับแต่งภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการบันทึกรูปเสร็จแล้ว ก็จะออกมายัง Camera View เดิมแต่จะมีปุ่มให้กดเพิ่มคือ ปุ่มแสดงภาษามือและรูปที่ได้ทำการปรับแต่งไว้แล้ว กดเพื่อทำรอการตอบรับจาก Server



รูป 3.39 Camera View ปุ่มแสดงภาษามือและรูปที่ได้ทำการปรับแต่งไว้

เมื่อทำการส่งไปการส่งไปยัง Server แล้วก็จะมีการรอเพื่อตอบรับการประมวลผลจาก Server และสักพักจะโชว์รูปสิ่งของที่ถูกหรือใกล้เคียงกลับมายัง Application แล้วเราทำการกดเลือก เพื่อแสดงภาษามือ Animation 3D

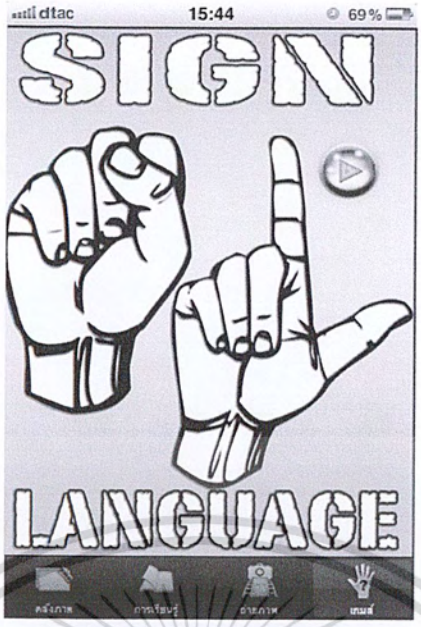
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 3.40 รูปสิ่งของที่ถูกต้องหรือใกล้เคียง

3.8.5 Games View

ขั้นตอนในการเล่นเกมทำการเขย่าครั้งแรก เพื่อให้รูปภาพหมุนไปเรื่อยๆ แล้วเราก็ทำการเขย่าอีกทีเพื่อทำการหยุดรูป ในการเลือกหาคำตอบนั้นให้กดที่ปุ่มเล่น ที่ใกล้ๆปุ่มวงกลมใหญ่ๆ ซึ่งเป็นปุ่มเพื่อกดเลือกตอบ Animation อันนั้นเป็นคำตอบ ถ้าคำตอบถูกจะโชว์คำว่า “ถูก” ถ้าผิดจะโชว์คำว่า “ผิด” และมี Score เพื่อบอกคะแนน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.41 หน้าจอแรกของเกม

รูป 3.42 หน้าจอหลักในการเล่นเกม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและสรุปผลการทดลอง

4.1 การทดลอง Image Processing

4.1.1 Resize

จุดประสงค์การทดลอง : เพื่อหาขนาดของภาพที่เล็กที่สุดที่สามารถนำไป predict แล้วถูกต้อง

ขั้นตอนการทดลอง : ขนาดของภาพที่นำมาทดลองมี 50x50, 100x100, 150x150, 200x200, 250x250, 300x300 ผ่านการเทรน และลองนำภาพต้นฉบับเหล่านี้มา predict (ในที่นี้ยังไม่ได้ใช้การเปรียบเทียบ pixel ใช้มา predict)

ผลการทดลอง : รูปที่มีขนาด 50x50 predict ไม่ถูกเลย มันกลายเป็นรูปที่ใกล้เคียงกับมัน ส่วนขนาด 100x100, 150x150 ยังมี predict ผิดอยู่บ้าง 200x200, 250x250, 300x300 predict ถูกหมด แต่ขั้นตอนใน image processing นั้นใช้การเปรียบเทียบ pixel ระหว่างภาพขอบและภาพวัตถุ การจะเปรียบเทียบกับขอบๆต้องมีความหนาพอสมควร และเนื่องจากขนาดภาพที่เล็กที่สุดที่ predict ถูกหมดคือ 200x200 ฉะนั้นเราจึงเลือกใช้ภาพ input ที่มีขนาด 300x300

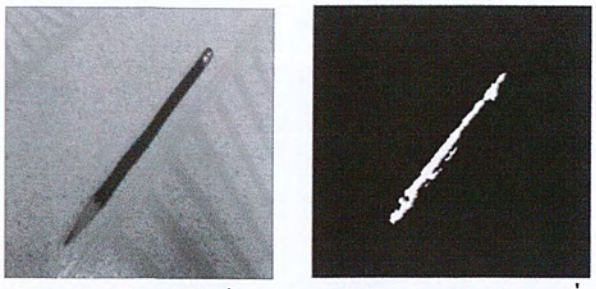
4.1.2 หาค่า threshold

จุดประสงค์การทดลอง : เพื่อหาค่า threshold ที่เหมาะสมกับทุกภาพโดยอัตโนมัติ

ขั้นตอนการทดลอง : ค่า threshold ที่ใช้ มาจากการทดลองเปลี่ยนค่า threshold ของวัตถุแต่ละแบบ ในแต่ละสภาพแวดล้อม ซึ่งมีค่า threshold แตกต่างกันไป ในการทดลองได้นำภาพส่วนหนึ่งประมาณ 10 ภาพที่มีสภาพแวดล้อม รวมไปถึงโทนสีที่แตกต่างกันมาลองปรับหาค่า threshold ที่เหมาะสม

ผลการทดลอง : พบความสัมพันธ์แบบผกผัน หรือตรงกันข้ามระหว่างค่า threshold ของภาพนั้น กับ percent ที่เหมือนกันของ pixel สำหรับภาพขอบและภาพวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.1 ตัวอย่างผลลัพธ์จากการปรับค่า threshold แบบที่ 1

รูป 4.2 เป็นตัวอย่างภาพ input(ภาพทางด้านซ้าย) ที่มีค่า percent การเปรียบเทียบระหว่างภาพขอบและภาพวัตถุที่มีค่าน้อยๆส่งผลให้ต้องปรับค่า threshold ให้มีค่ามากๆ จึงได้ภาพ output ที่แยกแยะระหว่างวัตถุกับพื้นหลังได้(ภาพทางด้านขวา)

รูป 4.2 ตัวอย่างผลลัพธ์จากการปรับค่า threshold แบบที่ 2

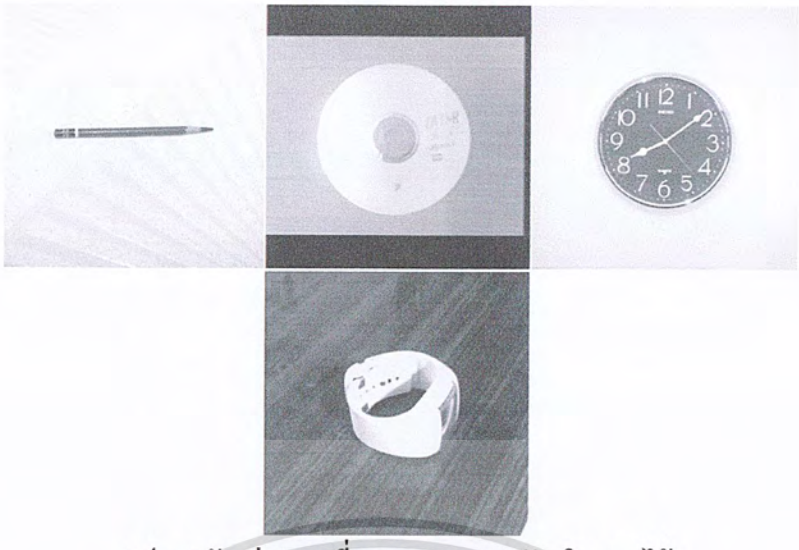
รูป 4.3 ผลลัพธ์ของภาพที่มีความเข้มแสงที่ขอบภาพ และที่ภาพวัตถุ เท่ากัน

4.1.3 ตัวอย่างภาพที่ใช้ในการทดลอง

4.1.3.1 ภาพที่สามารถ detect วัตถุในภาพได้

ทดลองโดยนำภาพที่ตรงตามเงื่อนไขในการถ่ายภาพจำนวน 20 ภาพ ผลลัพธ์ มีภาพที่ detect วัตถุในภาพได้จำนวน 20 ภาพลักษณะของภาพที่ detect วัตถุในภาพได้ แสดงในรูปที่ 4.4

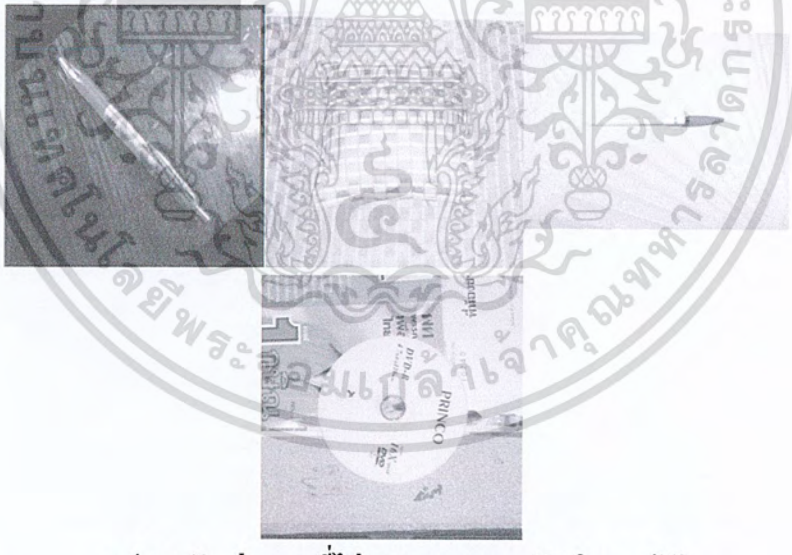
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.4 ตัวอย่างภาพที่สามารถ detect วัตถุในภาพได้

4.1.3.2 ภาพที่ไม่สามารถ detect วัตถุในภาพได้

ทดลองโดยนำภาพที่ตรงตามเงื่อนไขในการถ่ายภาพจำนวน 20 ภาพ ผลลัพธ์ มีภาพที่ detect วัตถุในภาพได้จำนวน 0 ภาพลักษณะของภาพที่ไม่สามารถ detect วัตถุในภาพได้แสดงในรูปที่ 4.5



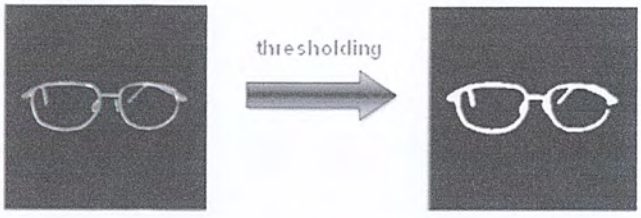
รูป 4.5 ตัวอย่างภาพที่ไม่สามารถ detect วัตถุในภาพได้

4.1.4 ปัญหาที่พบและทางแก้

การถ่ายภาพวัตถุที่พื้นหลังมีสีเดียวกับวัตถุ เช่น แก้ว, หมอนกับโซฟาที่มีลายเดียวกัน เป็นต้น กรณีนี้ความเหมือนกันของ pixel ในภาพขอบและภาพวัตถุเหมือนกันเกือบ 100% (นั่นคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วัตถุสีเดียวกับพื้นหลัง หรือวัตถุเป็นแก้ว)จะทำให้ไม่ได้ภาพวัตถุออกมาเลย ฉะนั้นเราจะนำภาพวัตถุ มาหา edge detection เพื่อให้ได้ภาพมาประมวลผลต่อไป



รูป 4.6 ภาพทางซ้ายเป็นต้นฉบับ ทางขวาส่งไปให้ neural network เทรนเป็นภาพที่ไม่มี noise

รูป 4.7 ภาพที่นำมา predict ทางซ้ายเป็นภาพ input ทางขวาเป็นภาพที่ผ่านการทำ edge detection

ผลที่ได้จากการเดา คือมันเดาได้ว่าเป็นรูปแว่นตานั่นถือเป็นผลลัพธ์ที่น่าพอใจ เพราะแม้จะมี noise ที่เป็นแสงสะท้อนจากดวงไฟ และขาแว่น แต่ก็ยังเดาออกมาถูกต้อง ดังนั้น ปัญหาการถ่ายภาพวัตถุที่พื้นหลังมีสีเดียวกับวัตถุ สามารถแก้ได้โดยการทำ edge detection แต่ไม่สามารถแก้ด้วยวิธีนี้ได้ทุกกรณี ทางแก้อีกทางหนึ่งคือ การกำหนดข้อจำกัดในการถ่ายภาพให้แก่ user ซึ่งได้กล่าวไว้ตามข้างต้น

4.1.5 Tool ที่ใช้สำหรับ image processing

Tool ที่ทดลองใช้ใน project มี visual studio 2008 using EmguCV และ visual studio 2008 ร่วมกับ MATLAB

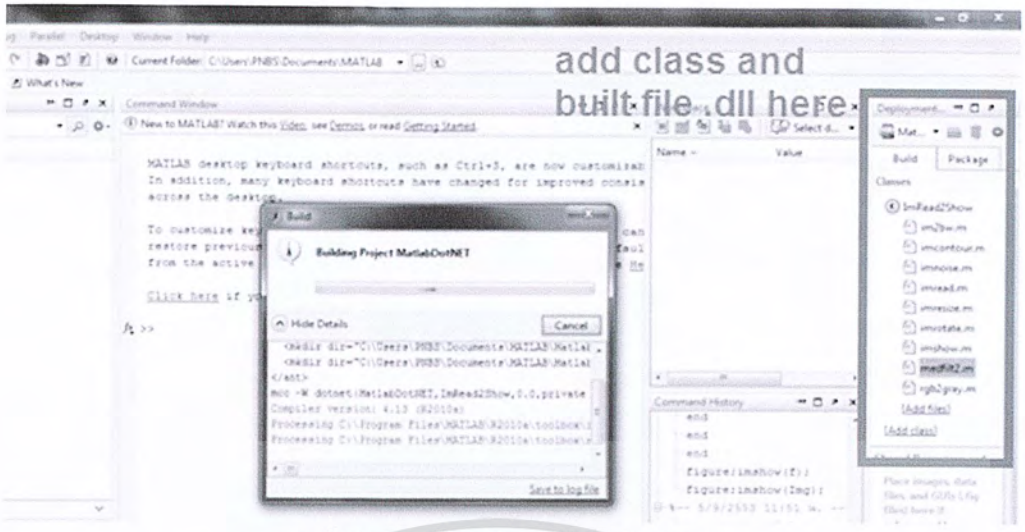
4.1.5.1 Visual Studio 2008 using EmguCV

EmguCV เป็น wrapper ของ library ของ OpenCV ซึ่ง implement code ด้วย C# ซึ่งเป็น เป็น open source ที่มี function ที่ design มาเพื่อการทำ image processing โดยเฉพาะ

4.1.5.2 Visual Studio 2008 ใช้ร่วมกับ MATLAB

เป็นการนำเอา function ที่ใช้ใน MATLAB มาใช้ผ่าน visual studio 2008 โดยการ add reference จาก file นามสกุล .dll ที่ทำการ add function ที่ต้องการใช้ใน MATLAB แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.8 การสร้าง file .dll ใน MATLAB

รูป 4.9 ผลการ add reference ใน visual studio เสร็จ สามารถเรียกใช้ function ใน MATLAB

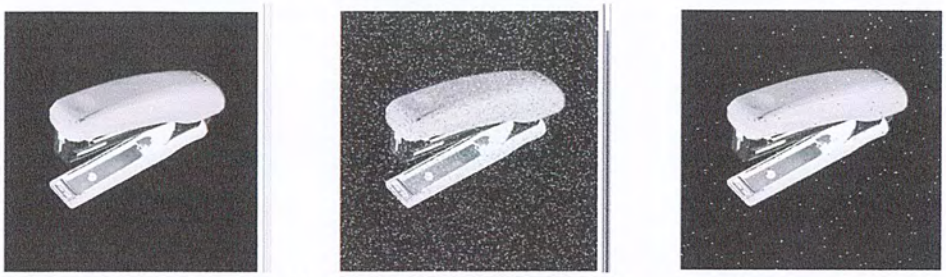
เหตุที่นำเอา MATLAB มาใช้ร่วมกับ visual studio เพื่อให้ง่ายต่อการเชื่อมโยงระหว่าง process เช่น application server ส่วนที่เป็น UI ส่งมา image processing หรือ image processing ส่งเข้า neural network รวมถึงการออกแบบ UI ต่างๆด้วยซึ่งสะดวกกว่าการใช้ MATLAB เพียงอย่างเดียว

การเลือกใช้ tool เราเลือกใช้ tool ในข้อ 1 เพราะ

1) เป็น open source สามารถหา download ได้บนเว็บไซต์

2) Function และผลลัพธ์ที่ได้จากการ process เหมาะที่จะนำไปใช้ในขั้นตอนถัดไป เช่น function การทำ smoothing แบบ median

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.10 การทำ smoothing ด้วย MATLAB โดยรูปมีขนาด 300x300 ใช้ตัวกรองขนาด 3x3

ใน MATLAB ต้องมี noise ในภาพจึงจะทำ smoothing ได้(ทำแล้วจึงเห็นการเปลี่ยนแปลง) ซึ่งเป็นไปตามทฤษฎีโดยภาพต้นฉบับ(รูปซ้ายมือ) ภาพที่เพิ่ม noise(ภาพกลาง) และภาพที่ทำ smoothing ด้วย(ภาพขวามือ) ส่วนใน OpenCV ไม่จำเป็นต้องมี noise ก็สามารถทำ smoothing ได้ แต่ขนาดตัวกรองต้องเป็นเลขคี่(มิเช่นนั้นภาพที่ได้จะเพี้ยน)

รูป 4.11 การทำ smoothing ด้วย OpenCV โดยรูปมีขนาด 300x300 ตัวกรองขนาด 31x31

รูปที่ได้เป็นผลลัพธ์ที่เราต้องการ คือ ลดรายละเอียดในภาพ ให้เหลือแค่รูปทรงให้ได้มากที่สุดเท่าที่เป็นไปได้

1) การ process ที่เร็วกว่า MATLAB เนื่องจาก MATLAB ถูกสร้างมาจากการภาษา Java ซึ่ง Java มาจากภาษา C ทำให้มีผลในการทำ image processing เล็กน้อย เพราะไปเสียเวลา interpret code ของ MATLAB (ในที่นี้ไม่มีการทดลองเป็นตัวเลข เนื่องจากทดลอง run ให้มันทำการ render ภาพ ช้ากว่า OpenCV อย่างเห็นได้ชัดประมาณ 2-3 วินาที แต่การคำนวณทั่วไปไม่ค่อยแตกต่าง)
2) การเรียกใช้งานหรือการแสดงผลบางอย่างใน MATLAB อยู่ในรูปของ toobox เช่น การทำ neural network ซึ่งไม่สะดวกที่จะนำมาใช้ร่วมกับ process ในส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อื่นๆ

4.1.5.3 Tool อื่นๆ

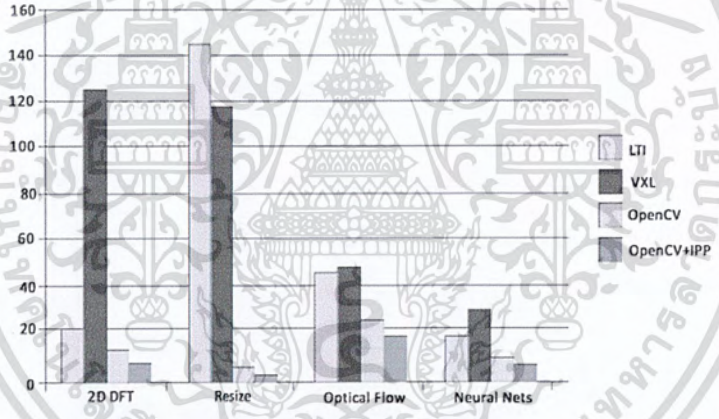
ในการทำ image processing ยังมี tool อื่นๆให้เรียกใช้อีกมาก ซึ่งในส่วนนี้ยังไม่ได้ทำการทดลองแต่จะขออ้างอิงจากคำอวดอ้างของ OpenCV โดย tool ที่ใช้มีดังนี้

1) VXL เป็น vision พื้นฐานในบาง library ซึ่ง implement code ด้วยภาษา C++

2) LTI เป็น library แนว object oriented ถูกสร้างมาจากภาษา C++ ประกอบไปด้วย class ในการทำ encapsulate multithreading, synchronization, serial port access เป็นต้น ตัว library นี้สามารถ run ได้หลาย platform

3) OpenCV เป็น library open source ที่มี function ที่ใช้ในทาง image processing ถึง 500 functions สามารถติดต่อกับกล้องและทำงานทางด้าน machine learning ได้ ในการ implement code ใช้ภาษา C++

4) OpenCV + IPP OpenCV ได้ร่วมพัฒนา library ตัวนี้กับ intel ความสามารถที่ได้คือใช้เวลาในการ execute เร็วขึ้นแต่ไม่เป็น open source ต้องซื้อมาใช้



รูป 4.12 กราฟ performance ของ tool ที่ใช้ในการทำ image processing

จากรูปด้านบนเขาทำการเปรียบเทียบ performance ของ tool แต่ละตัวโดย run บนเครื่อง

Pentium M 1.7GHz แล้วทำการทดลอง 4 อย่างมีรายละเอียดดังนี้

1) ทำ DFT กับภาพ 2D โดยรูปมีขนาด 512x512

2) ทำ Resize ภาพ 8 bits ขนาด 512x512 เป็น 384x384

3) ทำ Optical flow 520 points were tracked with a 41×41 window and 4 pyramid levels

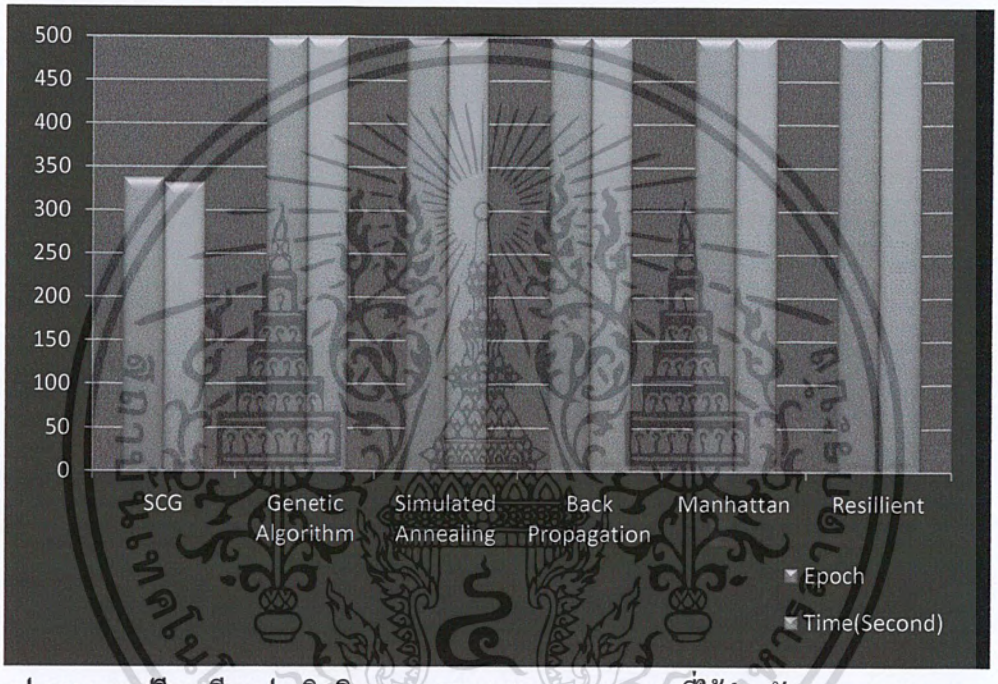
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) ทำ Neural network the mushroom benchmark of FANN

จากกราฟเราจึงทำการเลือกใช้ OpenCV ส่วน OpenCV + IPP ประสิทธิภาพดีกว่า แต่ไม่เป็น open source การนำมาใช้จึงไม่ค่อยสะดวกมากนัก

4.2 ผลการทดลอง Neural Network

ผลการทดลองเปรียบเทียบประสิทธิภาพของ Training Algorithm ที่ใช้สำหรับ Neural Network ทำการทดลองtrain neural network โดยใช้ภาพเป็น training set จำนวน 40 ภาพ ได้ผลการทดลองดังกราฟ โดยที่กราฟที่แตะเส้น 500 หมายถึงใช้มากกว่า 500 ขึ้นไป



รูป 4.13 การเปรียบเทียบประสิทธิภาพของ Training Algorithm ที่ใช้สำหรับ Neural Network

4.1.1 สรุปผลการทดลอง

ได้ว่า SCG(scale conjugate gradient algorithm) เป็น algorithm ที่มีประสิทธิภาพดีกว่า algorithm อื่นในการนำมาใช้เพื่อ train neural network ให้จดจำ pattern รูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุป

5.1 สรุปผลการดำเนินงาน

โครงงานนี้ได้ทำการศึกษาและพัฒนาระบบส่วนเสริมการเรียนรู้ภาษามือไทย โดยระบบแบ่งเป็น 3 ส่วนหลักคือ ส่วนติดต่อผู้ใช้ซึ่งติดตั้งอยู่บน iPhone ส่วนประมวลผลภาพ และส่วนรู้จำภาพซึ่งอยู่บนเครื่องแม่ข่าย ส่วนประมวลผลภาพใช้เทคนิค Image Processing โดยใช้การเปรียบเทียบค่าความเข้มแสงระหว่างบริเวณที่สนใจ(ในกรอบสีแดง) กับส่วนที่ไม่สนใจ(นอกกรอบสีแดง) ส่วนรู้จำภาพใช้เทคนิค Neural Network SCG(Scale Conjugate Gradient Method) ในการศึกษาได้มีการทำการทดลองในแต่ละส่วน เพื่อหาค่าพารามิเตอร์ที่เหมาะสม จากนั้นได้ทำการทดลองเพื่อทดสอบระบบโดยรวม โดยมีภาพวัตถุที่เตรียมไว้ 20 ภาพ(20 ชนิด) ระบบสามารถค้นหาภาพวัตถุ และแสดงภาษามือของวัตถุนั้นๆผ่านตัวละคร 3 มิติได้ นอกจากนี้ยังได้พัฒนาเกมบน iPhone เพื่อช่วยให้การเรียนรู้ภาษามือน่าสนใจและมีประสิทธิภาพมากยิ่งขึ้น

5.2 ปัญหาและอุปสรรค

ในการดำเนินการได้พบปัญหา Overfiting ของ Neural Network ปัญหาการควบคุมพื้นหลังของรูปก่อนนำมาทำการประมวลผลภาพและการสื่อสารระหว่างเครื่องแม่ข่ายและ iPhone ซึ่งมีชนิดข้อมูล(Data Type) ไม่ตรงกัน ปัญหาความเร็วในการรับส่งข้อมูลระหว่าง iPhone กับเครื่องแม่ข่าย

5.3 แนวทางการพัฒนาต่อ

1) พัฒนาระบบให้สามารถทำนายคำศัพท์ที่เป็นรูปธรรมได้ทั้งหมด
2) ทำให้ระบบสามารถใช้ได้หลายๆ Mobile Platform
3) ทำให้ระบบสามารถแปลงเสียงเป็นภาษามือได้
4) ทำให้ระบบสามารถแสดงภาษามือได้หลากหลายชาติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

Gary Bradski and Adrian Kaehler. 2008. **Learning OpenCV**. 1st ed. Gravenstein Highway North, Sebastopol : O'Reilly Media.

โรงเรียนโสตศึกษาทุ่งมหาเมฆ. **หนังสือสอนผู้พิการทางหู**

ผศ.ดร.อรณัตร จิตต์โสภักตร. 2552. **DIGITAL IMAGE PROCESSING.** พิมพ์ครั้งที่ 1. กรุงเทพฯ : บริษัท สงวนกิจ พริ้นท์ แอนด์ มีเดีย

Michael negnevitsky,**Aritficial intelligence A guide to intelligent systems 2nd edition 2003.** pearson education

Martin F. Møller **A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning** 1990. paper

Bill Dudney,Chris Adamson,2009. **iPhone SDK Development Building iPhone Applications.** The Pragmatic Bookshelf

Erica Sadun,2009. **The iPhone Developer's Cookbook Building Application with iPhone**

กรกมล ตั้งจิตเพียรพงศ์, ชัยพงศ์ คฤหเดชรัตนา,ลูกแพร ถาวรายุศม์ 2552. **โครงงาน ระบบแสดงภาษามือไทยด้วยตัวละครแอนนิเมชั่น 3 มิติ.**ปริญญานิพนธ์วิศวกรรมศาสตรบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์,สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

SDK. Addison-Weslwey

Apple iPhone Deveoper. [Online]. Available: http://developer.apple.com/devcenter/ios/index.action

Christopher Allen, Shannon Appelcline,2009. **iPhone in Action Introduction Web and SDK Development**. Manning Publications Co.

Objective-C Tutorial.[Online] Available: http://www.macfeteria.com/

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอชทีทีพี – วิกิพีเดีย. [Online].Available: th.wikipedia.org/wiki/เอชทีทีพี /

R. Fisher, S. Perkins, A. Walker and E. Wolfart. 2003. **Affine Transformation**. [Online]. Available : http://homepages.inf.ed.ac.uk/rbf/HIPR2/affine.htm

Paul DiLascia . **Using Matlab functions in C# (2009b).** [Online]. Available : http:// Xinyustudio.wordpress.com/2009/11/12using-matlab-functions-in-c-2009b/

Encog neural network library.Vaja.[Online].Avaliable:http://www.heatonresearch.com/encog

Utkarsh Sinha. **Why OpenCV**. [Online]. Available : http://www.aishack.in/2010/02/why-opencv/

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้