ตัวอ่านบาร์โค้ดผลิตภัณฑ์บนมือถือสำหรับคนตาบอดไทย

นางสาวนิตต์ชญาณ์ เจตจารุรัชต์

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2556
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)
เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ ที่ส่งผ่านทางบัณฑิตวิทยาลัย
The abstract and full text of theses from the academic year 2011 in Chulalongkorn University Intellectual Repository (CUIR)
are the thesis authors' files submitted through the University Graduate School.

MOBILE PRODUCT BARCODE READER FOR THAI BLINDS

Miss Nitchaya Jethjarurach

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Software Engineering
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2013

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	ตัวอ่านบาร์โค้ดผลิตภัณฑ์บนมือถือสำหรับคนตาบอดไทย
โดย	นางสาวนิตต์ชญาณ์ เจตจารุรัชต์
สาขาวิชา	วิศวกรรมซอฟต์แวร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	รองศาสตราจารย์ ดร. ญาใจ ลิ่มปิยะกรณ์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

.....คณบดีคณะวิศวกรรมศาสตร์

(ศาสตราจารย์ ดร. บัณฑิต เอื้ออาภรณ์)

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ

(ศาสตราจารย์ ดร. บุญเสริม กิจศิริกุล)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

(รองศาสตราจารย์ ดร. ญาใจ ลิ่มปิยะกรณ์)

.....กรรมการภายนอกมหาวิทยาลัย

(อาจารย์ ดร. ภาสกร อภิรักษ์วรพินิต)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

นิตต์ชญาณ์ เจตจารุรัชต์ : ตัวอ่านบาร์โค้ดผลิตภัณฑ์บนมือถือสำหรับคนตาบอดไทย. (MOBILE PRODUCT BARCODE READER FOR THAI BLINDS) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: รศ. ดร. ญาใจ ลิ่มปิยะกรณ์, 50 หน้า.

ข้อกำหนดผลิตภัณฑ์มีอิทธิพลต่อการตัดสินใจซื้อสินค้าอุปโภคบริโภคของผู้ซื้อ อย่างไรก็ตาม สำหรับผู้พิการทางสายตาเป็นเรื่องยากหรือเป็นไปไม่ได้ที่จะซื้อสินค้าจากห้างร้านด้วยตนเอง เพื่อพัฒนาคุณภาพชีวิตและสนับสนุนการดำรงชีวิตได้ด้วยตนเองของผู้พิการทางสายตา งานวิจัยนี้ได้นำเสนอแนวทางการพัฒนาแอปพลิเคชันบนมือถือโปรแกรมอ่านบาร์โค้ดผลิตภัณฑ์ โดยใช้ศาสตร์การประมวลผลภาพร่วมกับเทคโนโลยีการสังเคราะห์เสียงพูด ด้วยเทคโนโลยีปัจจุบันสมาร์ตโฟนสามารถใช้เป็นอุปกรณ์อ่านบาร์โค้ดได้ โดยการจับภาพบาร์โค้ดและอ่านออกเสียงบอกรายละเอียดสินค้าอุปโภคบริโภคสำหรับช่วยคนตาบอดในการเลือกซื้อผลิตภัณฑ์ แอปพลิเคชันที่พัฒนาขึ้นนี้เหมาะกับคนตาบอดที่มีทักษะความชำนาญถ่ายรูปด้วยสมาร์ตโฟนได้

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต .....

สาขาวิชา วิศวกรรมซอฟต์แวร์ ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก .....

ปีการศึกษา 2556

5570982621 : MAJOR SOFTWARE ENGINEERING
KEYWORDS: IMAGE PROCESSING / PRODUCT BARCODE READER / TEXT-TO-SPEECH / MOBILE APPLICATION / BLINDS

NITCHAYA JETHJARURACH: MOBILE PRODUCT BARCODE READER FOR THAI BLINDS. ADVISOR: ASSOC. PROF. YACHAI LIMPIYAKORN, 50 pp.

The product specification, more or less, influences the buyer’s decision making, when purchasing consumer products. However, it is impossible or difficult for the visual disabled themselves to purchase products from the stores. In order to improve quality of life and to support independent living for the blind, this article presents an approach to developing the mobile application of product barcode reader using image processing discipline integrated with Text-to-Speech synthesis technology. With today technology, smartphones can be used as a barcode reader device to capture the barcode image and vocalize the product detail to assist the blind for product buying. The implemented application suits for the blind who have skill in photo taking with smartphones.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

Department: Computer Engineering Student's Signature .....
Field of Study: Software Engineering Advisor's Signature .....
Academic Year: 2013

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดีจากความอนุเคราะห์ของรองศาสตราจารย์ ดร.ญาใจ ลิ่มปิยะกรณ์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ซึ่งท่านได้เสียสละเวลาในการให้ความรู้ แนะแนวทางพัฒนา และแก้ไขตรวจสอบความเรียบร้อย ข้าพเจ้าจึงกราบระลึกถึงพระคุณของรองศาสตราจารย์ ดร.ญาใจ ลิ่มปิยะกรณ์ไว้ ณ ที่นี้ และขอขอบพระคุณ ศาสตราจารย์ ดร.บุญเสริม กิจศิริกุล ประธานกรรมการสอบวิทยานิพนธ์ และอาจารย์ ดร.ภาสกร อภิรักษ์วรพินิต กรรมการสอบวิทยานิพนธ์ ที่กรุณาสละเวลาให้คำแนะนำ ตรวจสอบ และชี้จุดบกพร่องที่ควรแก้ไข ท้ายที่สุด ข้าพเจ้าขอกราบขอบพระคุณ คุณพ่อ คุณแม่ และครอบครัว สำหรับกำลังใจที่มีค่ายิ่ง รวมถึงขอขอบพระคุณนิสิตสาขาวิศวกรรมซอฟต์แวร์ ที่ให้กำลังใจ ให้การสนับสนุนและความช่วยเหลือในด้านต่างๆ และท่านอื่นๆ ที่มิได้กล่าวชื่อไว้ ณ ที่นี้ที่มีส่วนช่วยให้วิทยานิพนธ์ของข้าพเจ้าสำเร็จไปได้ด้วยดี

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง

บทคัดย่อภาษาอังกฤษ ..... จ

กิตติกรรมประกาศ ..... ฉ

สารบัญ ..... ช

สารบัญตาราง ..... ฎ

สารบัญภาพ ..... ฏ

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญของปัญหา 1

1.2 วัตถุประสงค์ของการวิจัย 1

1.3 ขอบเขตของการวิจัย 1

1.4 วิธีดำเนินงานวิจัย 2

1.5 ข้อตกลงเบื้องต้น 2

1.6 ประโยชน์ที่คาดว่าจะได้รับ 2

1.7 ผลงานที่ตีพิมพ์จากวิทยานิพนธ์ 2

1.8 ลำดับการจัดเรียงเนื้อหาในวิทยานิพนธ์ 3

บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 4

2.1 ทฤษฎีที่เกี่ยวข้อง 4

2.1.1 บาร์โค้ด (Barcode) [1] 4

2.1.2 ระบบสังเคราะห์เสียงพูด (Text to Speech - TTS) [5] 5

2.2 งานวิจัยที่เกี่ยวข้อง 7

2.2.1 An algorithm enabling blind users to find and read barcodes [6] 7

2.2.2 Automatic real-time barcode localization in complex scenes [7] 10

บทที่ 3 วิธีดำเนินงานวิจัย 11

3.1 แนวคิดในการพัฒนา 11

3.2 การถ่ายภาพ 12

3.3 การระบุตำแหน่งบาร์โค้ด 13

3.4 การถอดรหัสบาร์โค้ด 14

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

หน้า

3.5 การค้นหาข้อกำหนดผลิตภัณฑ์ 15
3.6 การแสดงผลข้อกำหนดผลิตภัณฑ์ และอ่านข้อกำหนดผลิตภัณฑ์ 15
บทที่ 4 การออกแบบและพัฒนาระบบ 17
4.1 สถาปัตยกรรมระบบ 17
4.2 สภาพแวดล้อมและเครื่องมือที่ใช้ในการพัฒนา 18
4.2.1 สภาพแวดล้อม 18
4.2.2 เครื่องมือที่ใช้ในการพัฒนา 18
4.3 การพัฒนาระบบ 18
4.3.1 ตัวจับภาพผลิตภัณฑ์ 18
4.3.2 ตัวตรวจสอบภาพบาร์โค้ด 20
4.3.3 ตัวแปลงรหัสบาร์โค้ด 27
4.3.4 ตัวค้นหาข้อกำหนดผลิตภัณฑ์ 29
4.3.5 ตัวแสดงผลข้อกำหนดผลิตภัณฑ์ 30
บทที่ 5 การประเมินและการวัดผล 32
5.1 แนวทางการประเมินผลงานวิจัย 32
5.2 ผลการเปรียบเทียบตัวอย่างที่ 1 32
5.2.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ .. 32
5.2.2 ผลลัพธ์ที่ได้จากระบบ 32
5.3 ผลการเปรียบเทียบตัวอย่างที่ 2 33
5.3.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ .. 33
5.3.2 ผลลัพธ์ที่ได้จากระบบ 33
5.4 ผลการเปรียบเทียบตัวอย่างที่ 3 33
5.4.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ .. 33
5.4.2 ผลลัพธ์ที่ได้จากระบบ 34
5.5 ผลการเปรียบเทียบตัวอย่างที่ 4 34
5.5.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ .. 34
5.5.2 ผลลัพธ์ที่ได้จากระบบ 34

หน้า

5.6 ผลการเปรียบเทียบตัวอย่างที่ 5 35
5.6.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ .. 35
5.6.2 ผลลัพธ์ที่ได้จากระบบ 35
5.7 ผลการเปรียบเทียบตัวอย่างที่ 6 35
5.7.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ .. 35
5.7.2 ผลลัพธ์ที่ได้จากระบบ 36
5.8 ผลการเปรียบเทียบตัวอย่างที่ 7 36
5.8.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ .. 36
5.8.2 ผลลัพธ์ที่ได้จากระบบ 37
5.9 ผลการเปรียบเทียบตัวอย่างที่ 8 37
5.9.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ .. 38
5.9.2 ผลลัพธ์ที่ได้จากระบบ 38
5.10 ผลการเปรียบเทียบตัวอย่างที่ 9 39
5.10.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ 39
5.10.2 ผลลัพธ์ที่ได้จากระบบ 39
บทที่ 6 สรุปผลการวิจัย และข้อเสนอแนะ 42
6.1 สรุปผลการวิจัย 42
6.2 ข้อจำกัด 42
6.3 แนวทางการวิจัยต่อ 42
รายการอ้างอิง 43
ภาคผนวก 44
ภาคผนวก ก การปรับตั้งค่าไดเรกทอรี เพื่อรองรับการเขียนโปรแกรมแอนดรอยด์ 45
ภาคผนวก ข การติดตั้ง SDK เพื่อรองรับการเขียนโปรแกรมแอนดรอยด์ 48
ประวัติผู้เขียนวิทยานิพนธ์ 50

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญตาราง

หน้า

ตารางที่ 1 รายละเอียดกรณีทดสอบของระบบ..... 40

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

สารบัญภาพ

หน้า

ภาพที่ 1 บาร์โค้ด UPC-A [2] 4
ภาพที่ 2 บาร์โค้ด EAN-13 [3] 5
ภาพที่ 3 ตัวอย่างบาร์โค้ดในประเทศไทย [4] 5
ภาพที่ 4 สถาปัตยกรรมของ Text to Speech Engine [5] 6
ภาพที่ 5 กระบวนการค้นหาบาร์โค้ด [6] 7
ภาพที่ 6 อัลกอริทึม Line scan [6] 8
ภาพที่ 7 อัลกอริทึม Line Tally [6] 9
ภาพที่ 8 อัลกอริทึมถอดรหัสบาร์โค้ด [6] 9
ภาพที่ 9 กรอบงานการจำกัดตำแหน่งบาร์โค้ด [7] 10
ภาพที่ 10 Bar Axis Projection [7] 10
ภาพที่ 11 Effective Part [7] 10
ภาพที่ 12 ภาพรวมการดำเนินการตัวอ่านบาร์โค้ดผลิตภัณฑ์บนมือถือ 12
ภาพที่ 13 ภาพผลิตภัณฑ์ที่มีบาร์โค้ด 12
ภาพที่ 14 ภาพผลิตภัณฑ์ที่ไม่มีบาร์โค้ด 13
ภาพที่ 15 ตัวอย่างภาพผลิตภัณฑ์หลังผ่านกระบวนการเตรียมภาพ 13
ภาพที่ 16 ตัวอย่างภาพผลิตภัณฑ์ที่ได้จากฟังก์ชัน Hough Line Transform 14
ภาพที่ 17 ตัวอย่างภาพบาร์โค้ดที่เส้นบาร์โค้ดขาดหายทางด้านซ้าย 14
ภาพที่ 18 ตัวอย่างภาพบาร์โค้ดที่เส้นบาร์โค้ดขาดหายทางด้านขวา 14
ภาพที่ 19 ตัวอย่างฟังก์ชันการถอดรหัสบาร์โค้ดในไลบรารี Zxing 15
ภาพที่ 20 ตัวอย่างฐานข้อมูลข้อกำหนดผลิตภัณฑ์ใน SQLite 15
ภาพที่ 21 ผลลัพธ์ที่ได้จากกระบวนการแสดงผลข้อกำหนดผลิตภัณฑ์ 16
ภาพที่ 22 การเรียกใช้ซอฟต์แวร์ Vaja 16
ภาพที่ 23 สถาปัตยกรรมระบบที่พัฒนา 17
ภาพที่ 24 ฟังก์ชัน onAutofocus 18
ภาพที่ 25 การเรียกใช้งานฟังก์ชัน Autofocus อัตโนมัติ เมื่อมีการเคลื่อนย้ายกล้อง 19
ภาพที่ 26 ฟังก์ชัน onPictureTaken 20

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาพที่ 27 การแปลงรูปแบบของภาพเป็น Mat 21
ภาพที่ 28 ฟังก์ชัน Adaptive thresholding ใน OpenCV [11] 21
ภาพที่ 29 การเรียกใช้งานฟังก์ชัน Adaptive thresholding ใน OpenCV 22
ภาพที่ 30 ฟังก์ชัน Eroding ใน OpenCV [12] 22
ภาพที่ 31 ฟังก์ชัน Dilating ใน OpenCV [13] 22
ภาพที่ 32 การเรียกใช้ฟังก์ชัน Eroding ใน OpenCV 22
ภาพที่ 33 การเรียกใช้ฟังก์ชัน Dilating ใน OpenCV 22
ภาพที่ 34 ฟังก์ชัน Finding contours ใน OpenCV [14] 23
ภาพที่ 35 ฟังก์ชัน Hough Line Transform ใน OpenCV [15] 23
ภาพที่ 36 การเรียกใช้งานฟังก์ชัน Finding contours ใน OpenCV 24
ภาพที่ 37 อัลกอริทึมที่ใช้สำหรับคัดเลือกรูปทรงผลลัพธ์ 24
ภาพที่ 38 กระบวนการตัดภาพผลิตภัณฑ์ในส่วนที่เป็นรูปทรงที่ถูกเลือก 25
ภาพที่ 39 การเรียกใช้งานฟังก์ชัน Hough Line Transform ใน OpenCV 25
ภาพที่ 40 กรณีตำแหน่งของเส้นติดขอบซ้าย 26
ภาพที่ 41 กรณีตำแหน่งของเส้นติดขอบขวา 26
ภาพที่ 42 กรณีตำแหน่งของเส้นติดขอบบน 26
ภาพที่ 43 กรณีตำแหน่งของเส้นติดขอบล่าง 26
ภาพที่ 44 กรณีไม่มีตำแหน่งของเส้น 27
ภาพที่ 45 การหมุนภาพและการถอดรหัสบาร์โค้ด 28
ภาพที่ 46 การตรวจสอบผลลัพธ์การถอดรหัสบาร์โค้ด 28
ภาพที่ 47 การสืบค้นข้อกำหนดผลิตภัณฑ์จากฐานข้อมูล 29
ภาพที่ 48 คลาสของฐานข้อมูล 30
ภาพที่ 49 การแสดงผลลัพธ์ และการอ่านผลลัพธ์ให้ผู้ใช้งานทราบด้วยเสียง 31
ภาพที่ 50 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 1 32
ภาพที่ 51 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 2 33
ภาพที่ 52 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 3 34
ภาพที่ 53 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 4 34
ภาพที่ 54 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 5 35

ภาพที่ 55 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 6 36
ภาพที่ 56 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 7 37
ภาพที่ 57 ภาพผลลัพธ์การแสดงข้อกำหนดผลิตภัณฑ์ตัวอย่างที่ 7 37
ภาพที่ 58 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 8 38
ภาพที่ 59 ภาพผลลัพธ์การแสดงข้อกำหนดผลิตภัณฑ์ตัวอย่างที่ 8 38
ภาพที่ 60 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 9 39
ภาพที่ 61 ภาพผลลัพธ์การแสดงข้อกำหนดผลิตภัณฑ์ตัวอย่างที่ 9 39
ภาพที่ 62 การเข้าสู่เมนูเมนูตั้งค่าระบบ 45
ภาพที่ 63 การเข้าสู่เมนูตั้งค่าไดเรกทอรีของตัวแปรที่ต้องการ 45
ภาพที่ 64 การเพิ่มไดเรกทอรีของตัวแปรที่ต้องการ 46
ภาพที่ 65 การระบุตัวแปร JAVA_HOME และไดเรกทอรี 46
ภาพที่ 66 การระบุตัวแปร PATH และค่า Variable value 46
ภาพที่ 67 การปรับตั้งค่าไดเรกทอรีเสร็จสมบูรณ์ 47
ภาพที่ 68 การเลือกซอฟต์แวร์ที่ต้องการเพื่อติดตั้ง 48
ภาพที่ 69 เลือกติดตั้งแอนดรอยด์ SDK เวอร์ชันที่ต้องการติดตั้ง 48
ภาพที่ 70 ยืนยันการติดตั้งแอนดรอยด์ SDK เวอร์ชันที่ต้องการติดตั้ง 49
ภาพที่ 71 สถานะการติดตั้ง 49

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบัน ประเทศต่างๆได้ให้ความสำคัญกับการพัฒนาคุณภาพชีวิตของคนทุกชนชั้น ไม่เว้นแม้แต่ผู้พิการทางสายตา ดังจะเห็นได้จากเทคโนโลยีที่พัฒนาให้กับกลุ่มผู้พิการทางสายตามีมากขึ้น อาทิเช่น โน้ตบุ๊คคนตาบอด เครื่องรู้จำอักขระด้วยแสงหรือโอซีอาร์ (Optical Character Recognition - ORC) โปรแกรมอ่านหน้าจอ เป็นต้น อย่างไรก็ตาม ยังไม่มีเทคโนโลยีที่ช่วยในการบอกข้อกำหนดผลิตภัณฑ์ให้กับผู้พิการทางสายตาทราบ ซึ่งการเลือกซื้อผลิตภัณฑ์สำหรับอุปโภคบริโภคเป็นสิ่งจำเป็นในการดำรงชีวิตประจำวัน

บาร์โค้ด (Barcode) เป็นระบบที่ใช้แพร่หลายทั่วโลก ใช้สำหรับออกหมายเลขประจำตัวผลิตภัณฑ์เพื่อเป็นเครื่องชี้บอกถึงความแตกต่างของผลิตภัณฑ์แต่ละชนิด โดยผลิตภัณฑ์แต่ละชนิดจะมีหมายเลขบาร์โค้ดไม่ซ้ำกันเลย จึงทำให้บาร์โค้ดสามารถนำมาใช้เพื่อบ่งบอกข้อกำหนดผลิตภัณฑ์แต่ละชนิดได้

สมาคมคนตาบอด และอาสาสมัครเพื่อช่วยคนตาบอด มีโครงการสอนคนตาบอดให้มีทักษะในการใช้สมาร์ตโฟน เช่น สามารถถ่ายภาพได้ โดยอาศัยการสอนวางตำแหน่งของกล้อง และการกะระยะห่างด้วยเสียง ซึ่งทำให้คนตาบอดสามารถถ่ายภาพได้ด้วยตนเอง จึงไม่ใช่เรื่องยากที่จะนำทักษะการถ่ายภาพมาประยุกต์ใช้กับการถ่ายภาพบาร์โค้ดบนผลิตภัณฑ์

งานวิจัยนี้ได้ทำการศึกษาปัญหา และความต้องการของผู้พิการทางสายตา เพื่อนำมาพัฒนาวิธีการและแอปพลิเคชันบนมือถือสำหรับตรวจจับหาตำแหน่งบาร์โค้ดบนผลิตภัณฑ์สินค้าอุปโภคบริโภค และช่วยบอกข้อกำหนดผลิตภัณฑ์ให้ผู้พิการทางสายตาทราบด้วยเสียง ทั้งนี้ เป็นการอำนวยความสะดวกให้แก่ผู้พิการทางสายตาในด้านการเลือกซื้อสินค้าอุปโภคบริโภคในชีวิตประจำวัน ทำให้ผู้พิการทางสายตาสามารถดำรงชีวิตอยู่ได้ โดยสามารถช่วยเหลือตัวเองในด้านการเลือกซื้อผลิตภัณฑ์ต่างๆ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

1.2 วัตถุประสงค์ของการวิจัย

เพื่อนำเสนอวิธีการและพัฒนาแอปพลิเคชันบนมือถือสำหรับตรวจจับหาตำแหน่งบาร์โค้ดบนผลิตภัณฑ์สินค้าอุปโภคบริโภค เพื่ออำนวยความสะดวกแก่ผู้พิการทางสายตาให้ทราบถึงข้อมูลของผลิตภัณฑ์ผ่านการอ่านออกเสียง

1.3 ขอบเขตของการวิจัย

1. งานวิจัยรองรับการตรวจจับหาตำแหน่งบาร์โค้ดบนผลิตภัณฑ์จากบาร์โค้ดชนิด EAN-13 ซึ่งเป็นบาร์โค้ดบนผลิตภัณฑ์ขายปลีก
2. แอปพลิเคชันบนมือถือที่พัฒนาขึ้นทำงานบนระบบปฏิบัติการแอนดรอยด์

1.4 วิธีดำเนินงานวิจัย

1. ศึกษาปัญหาและความต้องการของผู้พิการทางสายตา
2. ศึกษาข้อมูลบาร์โค้ดบนผลิตภัณฑ์ที่มีการใช้อยู่ตั้งแต่เริ่มต้นจนถึงปัจจุบัน
3. ศึกษาและทำความเข้าใจทฤษฎีที่เกี่ยวข้อง
4. ศึกษาและทำความเข้าใจงานวิจัยที่เกี่ยวข้อง
5. ศึกษาขั้นตอนการพัฒนาแอปพลิเคชันในการตรวจจับระบุตำแหน่งบาร์โค้ดบนผลิตภัณฑ์
6. วิเคราะห์และกำหนดระเบียบวิธีวิจัย
7. ออกแบบ ตั้งสมมติฐาน ที่เกี่ยวข้องกับงานวิจัย
8. พัฒนาแอปพลิเคชันบนมือถือในการตรวจจับระบุตำแหน่งบาร์โค้ดบนผลิตภัณฑ์
9. ทดสอบและประเมินผลงานวิจัย
10. สรุปผลงานวิจัย
11. ตีพิมพ์ผลงานวิจัย
12. **จัดทำรูปเล่มวิทยานิพนธ์**

1.5 ข้อตกลงเบื้องต้น

1. ตัวอ่านบาร์โค้ดผลิตภัณฑ์บนมือถือรองรับการตรวจจับหาตำแหน่งบาร์โค้ดบนผลิตภัณฑ์จากบาร์โค้ดชนิด EAN-13 ซึ่งเป็นบาร์โค้ดบนผลิตภัณฑ์ขายปลีก และสามารถบอกข้อกำหนดผลิตภัณฑ์ให้กับผู้พิการทางสายตาทราบด้วยเสียง
2. แอปพลิเคชันบนมือถือที่พัฒนาขึ้นทำงานบนระบบปฏิบัติการแอนดรอยด์
3. ผู้พิการทางสายตาสามารถใช้สมาร์ตโฟนถ่ายรูปได้

1.6 ประโยชน์ที่คาดว่าจะได้รับ

ได้แอปพลิเคชันบนมือถือในการตรวจจับระบุตำแหน่งบาร์โค้ดบนผลิตภัณฑ์และอ่านออกเสียงบอกข้อกำหนดของผลิตภัณฑ์ เพื่ออำนวยความสะดวกแก่ผู้พิการทางสายตาสำหรับการเลือกซื้อสินค้าอุปโภคบริโภคในการดำรงชีวิตประจำวัน

1.7 ผลงานที่ตีพิมพ์จากวิทยานิพนธ์

ส่วนหนึ่งของวิทยานิพนธ์นี้ได้รับการตีพิมพ์เป็นบทความวิชาการเรื่อง “Mobile Product Barcode Reader for Thai Blinds”, International Conference on Information Science and Applications (ICISA), Vol. 5 ,May 2014 , pp.455-458

1.8 ลำดับการจัดเรียงเนื้อหาในวิทยานิพนธ์

วิทยานิพนธ์นี้แบ่งเนื้อหาออกเป็น 6 บทดังต่อไปนี้ บทที่ 1 เป็นบทนำกล่าวถึงความเป็นมาและความสำคัญของปัญหา วัตถุประสงค์ของการวิจัย ขอบเขตของการวิจัย วิธีดำเนินงานวิจัย ข้อตกลงเบื้องต้น ประโยชน์ที่คาดว่าจะได้รับ และผลงานตีพิมพ์จากวิทยานิพนธ์ บทที่ 2 กล่าวถึงทฤษฎีและงานวิจัยที่เกี่ยวข้อง บทที่ 3 กล่าวถึงวิธีดำเนินการวิจัย บทที่ 4 กล่าวถึงการออกแบบและพัฒนาระบบตามแนวทางการวิจัยที่นำเสนอ บทที่ 5 กล่าวถึงวิธีการประเมินและวัดผลการทดลอง และบทที่ 6 สรุปผลการวิจัย ข้อเสนอแนะ และแนวทางสำหรับการวิจัยต่อในอนาคต

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 บาร์โค้ด (Barcode) [1]

บาร์โค้ดเป็นเครื่องหมายแทนข้อมูลที่ได้รับการยอมรับอย่างแพร่หลายในการนำมาประยุกต์ใช้บนผลิตภัณฑ์ เพราะการนำเข้าข้อมูลจากรหัสบาร์โค้ดเป็นวิธีที่รวดเร็ว และความน่าเชื่อถือของข้อมูลมีสูง รวมทั้งให้ความสะดวกแก่ผู้ใช้งาน

บาร์โค้ดถูกพัฒนาขึ้นเมื่อ ค.ศ. 1950 โดยประเทศสหรัฐอเมริกาได้จัดตั้งคณะกรรมการเฉพาะกิจทางด้านพาณิชย์ขึ้นสำหรับค้นคว้ารหัสมาตรฐาน และสัญลักษณ์ที่สามารถช่วยกิจการด้านอุตสาหกรรม และสามารถจัดพิมพ์ระบบบาร์โค้ด UPC (Universal Product Code) ขึ้นได้ในปี 1973 ต่อมาในปี 1975 กลุ่มประเทศยุโรปจัดตั้งคณะกรรมการด้านวิชาการเพื่อสร้างระบบบาร์โค้ดเรียกว่า EAN (European Article Number) และระบบบาร์โค้ด EAN เริ่มเข้ามาในประเทศไทยเมื่อปี 1987

ปัจจุบันบาร์โค้ดมี 3 แบบ ได้แก่ 1 มิติ, 2 มิติ และ 3 มิติ แต่ที่ใช้กันทั่วไปบนผลิตภัณฑ์เป็นแบบ 1 มิติ ประเภทของบาร์โค้ดแบ่งได้เป็น 2 ประเภท

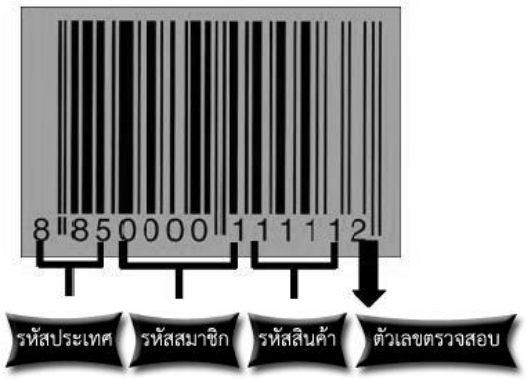
1. โค้ดภายใน (Internal Code) เป็นบาร์โค้ดที่จัดทำขึ้นเพื่อใช้ในองค์กร ไม่สามารถนำออกไปใช้ภายนอกได้
2. โค้ดมาตรฐานสากล (Standard Code) เป็นบาร์โค้ดที่นิยมใช้กันอย่างแพร่หลายทั่วโลก มี 2 ระบบ ประกอบด้วย
    - ระบบบาร์โค้ด UPC ใช้แพร่หลายในประเทศสหรัฐอเมริกา และแคนาดา ดังแสดงในภาพที่ 1

ภาพที่ 1 บาร์โค้ด UPC-A [2]

    - ระบบบาร์โค้ด EAN ใช้แพร่หลายในภาคพื้นยุโรป เอเชีย และแปซิฟิก รวมทั้งประเทศไทย ดังแสดงในภาพที่ 2

ภาพที่ 2 บาร์โค้ด EAN-13 [3]

สำหรับบาร์โค้ดในประเทศไทยเริ่มนำมาใช้อย่างจริงจัง โดยมีสถาบันสัญลักษณ์รหัสแท่งไทย (Thai Article Numbering Council - TANC) เป็นองค์กรตัวแทน EAN ภายใต้การดูแลของสภาอุตสาหกรรมกรรมแห่งประเทศไทย ทั้งนี้ ระบบ EAN ที่ประเทศไทยใช้นั้นจะมีลักษณะเป็นเลขชุด 13 หลัก ดังแสดงในภาพที่ 3

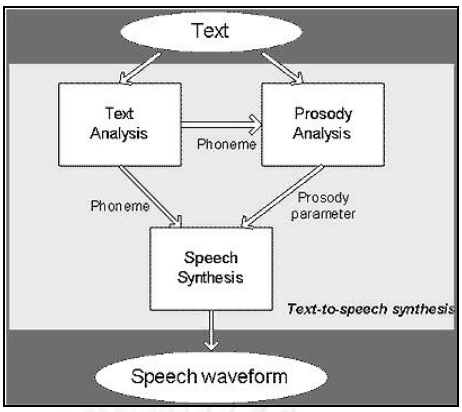


ภาพที่ 3 ตัวอย่างบาร์โค้ดในประเทศไทย [4]

2.1.2 ระบบสังเคราะห์เสียงพูด (Text to Speech - TTS) [5]

ระบบสังเคราะห์เสียงพูด คือ เทคโนโลยีสำหรับการอ่านข้อความประกอบด้วยตัวหนังสือ ตัวเลข และสัญลักษณ์ แล้วแปลงเป็นเสียงพูด โดยอาศัยวิธีการสังเคราะห์เสียงจากการผสมเสียงจากตัวอักษรที่ประกอบเป็นคำ หรือประโยค เพื่อให้เครื่องคอมพิวเตอร์สามารถสร้างเสียงเลียนแบบเสียงมนุษย์ และมีความหมายที่สามารถเข้าใจได้ ประโยชน์ของการประยุกต์ใช้เทคโนโลยีสังเคราะห์เสียงพูดในงานวิจัยนี้เพื่อเป็นการเพิ่มโอกาสให้ผู้พิการทางสายตาได้รับทราบอ่านข้อมูลผ่านทางการรับรู้ด้วยเสียง

ระบบสังเคราะห์เสียงพูดจะถูกนำมาใช้ในโปรแกรมอ่านหน้าจอเพื่อให้ผู้ใช้รับรู้รายละเอียดที่อยู่บนหน้าจอ โครงสร้างของระบบสังเคราะห์เสียง โดยทั่วไป สามารถแบ่งการทำงานเป็นส่วนย่อยๆ ได้เป็น 3 ส่วน ดังแสดงในภาพที่ 4



ภาพที่ 4 สถาปัตยกรรมของ Text to Speech Engine [5]

1. ส่วนการวิเคราะห์ข้อความ (Text Analysis) : ส่วนนี้จะมีหน้าที่วิเคราะห์ข้อความนำเข้า ได้แก่ ข้อความที่โปรแกรมอ่านหน้าจอต้องการให้อ่าน เพื่อแปลงเป็นข้อมูลเสียงอ่าน (Phoneme) ของคำนั้นๆ และส่งต่อให้ส่วนของการสังเคราะห์เสียง (Speech synthesis) ต่อไป นอกจากนี้ ส่วนนี้ยังทำหน้าที่อย่างอื่น ได้แก่

    - การแบ่งประโยคจากข้อความที่ยาว (Sentence breaking) การตัดคำ และพยางค์
    - การทำข้อความให้อยู่ในรูปปกติ (Text Normalization) ได้แก่ การแปลงตัวเลข คำย่อ และเครื่องหมายอื่นๆที่ไม่ใช่ข้อความให้กลายเป็นข้อความ
    - การหาขอบเขตของวลีของการอ่านในประโยค

    โดยทั่วไป หน้าที่การตัดคำจะไม่ถูกผนวกเป็นส่วนหนึ่งของระบบสังเคราะห์เสียง แต่เนื่องจากภาษาไทยจำเป็นต้องใช้การตัดคำมาช่วยในการสังเคราะห์เสียง ดังนั้น ส่วนของการวิเคราะห์ข้อความจึงรวมเอาส่วนของการตัดคำไว้ด้วย

2. ส่วนการวิเคราะห์สัทสัมพันธ์ (Prosody Analysis) : ส่วนนี้มีหน้าที่ในการวิเคราะห์และสังเคราะห์ข้อมูลสัทสัมพันธ์ของประโยคใดๆจากข้อมูลเสียงอ่าน และข้อความข้อมูลสัทสัมพันธ์ที่วิเคราะห์ออกมาได้ในระบบทั่วไป ได้แก่

    - Segment Duration คือ ความยาวของเสียงย่อยที่ต้องการสังเคราะห์ ค่านี้จะมีผลต่อจังหวะของเสียงที่ทำการสังเคราะห์ เช่น ถ้ากำหนดให้ค่าความยาวของเสียงย่อยที่ต้องการสังเคราะห์มีขนาดสั้นเสียงที่สังเคราะห์ขึ้นก็จะเหมือนกับการพูดเร็วๆ
    - Pitch Contour คือ ค่าความสัมพันธ์ของความถี่มูลฐานกับเวลา ค่านี้จะมีผลต่อเสียงสูงต่ำของประโยคนั้นๆ

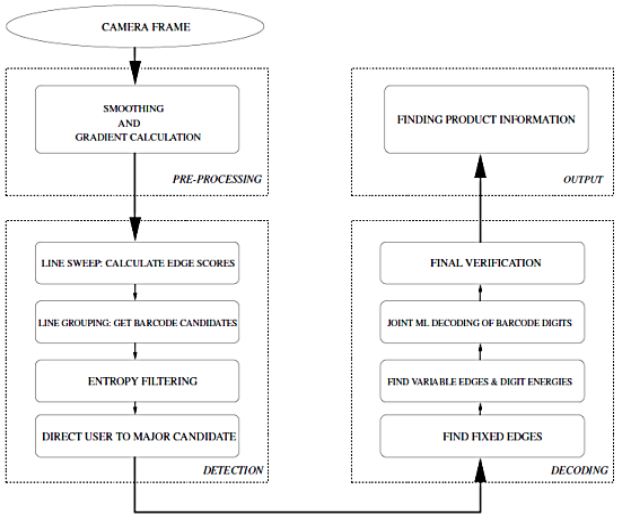
3. ส่วนของการสังเคราะห์เสียง : ส่วนนี้มีหน้าที่ในการสร้างสัญญาณคลื่นเสียงจากข้อมูลเสียงอ่าน (Phonetic transcription) และข้อมูลสัทสัมพันธ์ (Prosody transcription) แล้วส่งออกสู่ลำโพง เพื่อให้เปล่งเสียงพูดข้อความที่รับเข้ามาจากโปรแกรมอ่านข้อความ โดยทั่วไปวิธีการที่ใช้ในส่วนนี้สามารถแบ่งตามเทคนิควิธีการสังเคราะห์เสียงได้ 3 ประเภท คือ

- Formant Synthesis เทคนิคการสังเคราะห์วิธีแบบนี้ ข้อมูลเสียงอ่านจะถูกกำหนดไว้อยู่ในรูปของความถี่ฟอร์แมนท์ เมื่อต้องการสังเคราะห์เสียง จะนำข้อมูลเหล่านี้มาทำการสังเคราะห์ให้เป็นสัญญาณเสียง วิธีการนี้มีข้อดี คือ สามารถควบคุมค่าความเปลี่ยนแปลงของความถี่ฟอร์แมนท์ (Formant transition) ที่บริเวณรอยต่อระหว่างเสียงได้ง่าย แต่ข้อเสีย คือ การจะแทนเสียงด้วยค่าฟอร์แมนท์ทำได้ยาก จะต้องมีกฎในการสังเคราะห์เสียงจำนวนมาก และเสียงที่สังเคราะห์ออกมาได้จะไม่เป็นธรรมชาติ
- Articulation Synthesis สำหรับวิธีนี้ข้อมูลเสียงที่ต้องการสังเคราะห์จะอยู่ในรูปของค่าพารามิเตอร์ของโครงสร้างทางกายภาพของการเคลื่อนไหวของอวัยวะที่ทำให้เกิดเสียงต่างๆ วิธีการนี้ค่อนข้างยากในแง่ของการโมเดลเสียงต่างๆ เพราะจะต้องศึกษาจากอวัยวะที่ใช้ในการออกเสียงจริง
- Concatenation synthesis เทคนิควิธีการนี้เป็นวิธีที่ได้รับความนิยมมากที่สุดในปัจจุบัน โดยเสียงที่ทำการสังเคราะห์ขึ้น เกิดจากการนำหน่วยเสียงย่อยที่ทำการเก็บไว้ก่อน ดูจากการนำหน่วยเสียงนำมาเรียงต่อกันจนเป็นเสียงพูดที่ต้องการ โดยทั่วไปหน่วยเสียงย่อยที่ทำเก็บไว้จะอยู่ระดับต่ำกว่าคำ เช่น หน่วยของเสียงพยางค์ หน่วยของเสียงครึ่งพยางค์ (demisyllable) และหน่วยของเสียงคู่เสียง (diphone) เป็นต้น

2.2 งานวิจัยที่เกี่ยวข้อง

2.2.1 An algorithm enabling blind users to find and read barcodes [6]

งานวิจัยนี้นำเสนออัลกอริทึมสำหรับค้นหาบาร์โค้ด และอ่านบาร์โค้ดชนิด UPC-A โดยมีขั้นตอนกระบวนการที่สำคัญ ดังแสดงในภาพที่ 5



ภาพที่ 5 กระบวนการค้นหาบาร์โค้ด [6]

1. Pre-processing : เป็นขั้นตอนในการคำนวณหาส่วนของภาพที่มีลักษณะเป็นแท่งบาร์โค้ด
2. Detection : เป็นขั้นตอนในการตรวจจับภาพที่มีลักษณะเป็นแท่งบาร์โค้ด โดยอัลกอริทึมที่ใช้ ได้แก่ อัลกอริทึม Line scan และอัลกอริทึม Line Tally แสดงในภาพที่ 6 และภาพที่ 7 ตามลำดับ และขั้นตอน Detection ประกอบด้วยกระบวนการย่อย ดังนี้
    - คำนวณการไล่ระดับสีทุกที่ในภาพ โดยจะทำการคำนวณ 4 ทิศทาง ได้แก่ แนวนอน, แนวตั้ง, เส้นทแยงมุม +45 องศา และเส้นทแยงมุม -45 องศา และลำดับการสแกนของภาพจะไล่จากซ้ายไปขวา และจากบนลงล่าง ตามลำดับ
    - ค้นหาตำแหน่งของภาพที่มีลักษณะเหมือนแท่งบาร์โค้ด
    - ค้นหาตำแหน่งภาพในทิศทางตั้งฉากกับครั้งแรก เพื่อนำค่ามาเปรียบเทียบความเป็นไปได้ที่จะเป็นบาร์โค้ด
    - บันทึกค่าที่ได้ เพื่อนำไปใช้ในกระบวนการถัดไป
3. Decoding : เป็นขั้นตอนในการแปลงภาพให้เป็นรหัสบาร์โค้ด โดยใช้อัลกอริทึมถอดรหัสบาร์โค้ดจะแสดงในภาพที่ 8
4. Output : เป็นขั้นตอนในการนำรหัสบาร์โค้ดที่ได้ไปค้นหาข้อกำหนดผลิตภัณฑ์ในฐานข้อมูล และอ่านข้อกำหนดผลิตภัณฑ์ให้กับผู้ใช้งาน

INITIALIZATION:
τ_G = minimum gradient threshold
n_E = minimum # of edges required
d_E = maximum distance between consecutive edges
SWEEP:
for orientation $t = 0, 45, 90, 135$ **do**
  for line $l = 1, \ldots, lastLineInThisOrientation$ **do**
    $count \leftarrow 0$
    for pixel $i = 1, \ldots, lastPixelOnThisLine$ **do**
      Let j be the last pixel on this line that was counted.
      if $|\nabla I_i| > \tau_G$ **then**
        //Gradient above threshold and angle approx. perpendicular to sweep line
        if $\angle \nabla I_i \approx \perp$ orientation **then**
          if $|\nabla I_i| \geq \max\{|\nabla I_{i-1}|, |\nabla I_{i+1}|\}$ **then** //non-maximum suppression
            if $\angle \nabla I_i$ is ≈ 180 degrees out of phase with $\angle \nabla I_j$, and $d_{ij} < d_E$ **then**
              $count \leftarrow count + 1$ //count this pixel
        else
          $count \leftarrow count - 1$ //pixel with strong gradient at wrong orientation
      else if $d_{ij} > d_E$ **then** //no edge pixel seen in a while
        $count \leftarrow count - 1$;
        if $d_{ij} > 2 * d_E$ **then** //no edges in a long while
          $count \leftarrow 0$ //end of candidate segment
      if $count = 0$ **then** //see if end of segment has been reached
        $score \leftarrow \max_{i \in lastSegment} count(i)$ //score is the max count for this segment
        if $score > n_E$ **then** //if the minimum # edges has been seen
          Record this segment as a barcode candidate segment for this line
        else
          Discard this segment

ภาพที่ 6 อัลกอริทึม Line scan [6]

INITIALIZATION:
n_L = minimum # of lines required
d_L = maximum distance between matching lines
τ_S = minimum score required to declare barcode candidate
$\mathcal{B} = \{\}$ //list of candidate areas
SWEEP:
for orientation $t = 0, 45, 90, 135$ **do**
  for line $l = 0, \ldots, lastLineInThisOrientation$ **do**
    for barcode segment candidate $c = 1, \ldots, lastCandidateOnThisLine$ **do**
      if $\exists b \in \mathcal{B}$: $begin_b \approx begin_c, end_b \approx end_c$ and $d_{lb} < d_L$ **then**
        //There was a previous barcode area candidate with similar beginning and end a little earlier
        $count_b \leftarrow count_b + 1$
      else
        $\mathcal{B} \leftarrow \mathcal{B} + c$ //Add this segment as the beginning of a new candidate area
        $count_c \leftarrow 1$
      for $b \in \mathcal{B}$ **do** //check that the candidates are still valid
        if $d_{lb} > d_L$ **then** //have not seen a match in a while
          $count_b \leftarrow count_b - 1$
          if $d_{lb} > 2 * d_L$ **then** //have not seen a match in a long while
            $count_b \leftarrow 0$
        if $count_b = 0$ **then** //end of candidate
          $score_b \leftarrow max_{l \in b} count(l)$ //score is the max count over all lines in this area
          if $score_b > \tau_S$ **then** //if the minimum # lines has been seen
            Record b as a barcode area segment
          else
            $\mathcal{B} \leftarrow \mathcal{B} - b$ //Discard this candidate

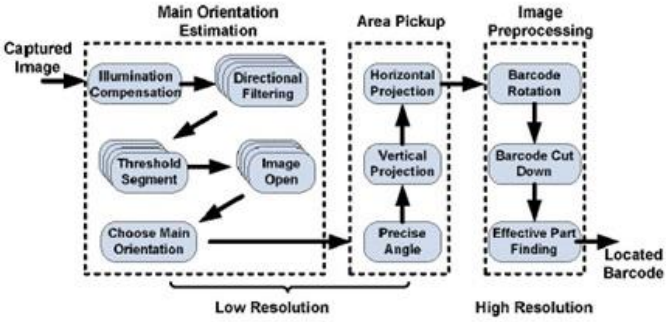
ภาพที่ 7 อัลกอริทึม Line Tally [6]

INITIALIZATION:
$X_{initial} = \frac{lastEdge - firstEdge}{95}$
FIND EDGES AND DIGIT PROBABILITIES:
Find the N_{slices} lines in the barcode with the highest edge count
for Slice $i = 1, \ldots, N_{slices}$ **do**
  Estimate $N_{fixedEdgeEstimates}$ fixed edges
  for Fixed Edge Estimate $j = 1, \ldots, N_{fixedEdgeEstimates}$ **do**
    for Barcode digit $d = 1, \ldots, 12$ **do**
      Get digit probabilities for each numeric digit $0, \ldots, 9$
  Marginalize digit probabilities over all fixed edge estimates
Marginalize digit probabilities over all slices
BARCODE ESTIMATION:
for Barcode digit $d = 1, \ldots, 12$ **do**
  Calculate auxiliary running parity check digit probabilities.
ML Estimation of the 2 most likely sequence of auxiliary random variables
Convert auxiliary random variables back to barcode digits
BARCODE VERIFICATION:
if Probability of most likely sequence $> K \times$ Probability of the second most likely sequence **then**
  if Most likely sequence differs from individually most likely digits by at most 1 digit **then**
    OUTPUT BARCODE
Get new frame

ภาพที่ 8 อัลกอริทึมถอดรหัสบาร์โค้ด [6]

2.2.2 Automatic real-time barcode localization in complex scenes [7]

งานวิจัยนี้นำเสนอวิธีการจำกัดพื้นที่ และถอดรหัสบาร์โค้ดอัตโนมัติที่มีประสิทธิภาพจากภาพบาร์โค้ดที่มีสิ่งรบกวนในภาพ หรือมีความละเอียดต่ำ โดยมีขั้นตอนหลักของกระบวนการจำกัดพื้นที่บรรจุบาร์โค้ด ดังแสดงในภาพที่ 9



ภาพที่ 9 กรอบงานการจำกัดตำแหน่งบาร์โค้ด [7]

1. Main Orientation Estimation : เป็นขั้นตอนในการเพิ่มคุณภาพของภาพ โดยการปรับแสง ลดสิ่งรบกวนในภาพ และเลือกทิศทางหลักของภาพบาร์โค้ด
2. Area Pickup : เป็นขั้นตอนในการจัดเก็บภาพที่เป็นบาร์โค้ด โดยจะคำนวณมุมของภาพที่เกิดจากแท่งบาร์โค้ด เพื่อนำมาใช้หาความกว้างของบาร์โค้ด ดังแสดงในภาพที่ 10

ภาพที่ 10 Bar Axis Projection [7]

3. Image Preprocessing เป็นขั้นตอนในการเตรียมภาพบาร์โค้ดก่อนเข้ากระบวนการถอดรหัสบาร์โค้ด โดยจะปรับมุมของภาพบาร์โค้ด และตัดส่วนของภาพที่ไม่ใช่บาร์โค้ดออก ดังแสดงในภาพที่ 11

ภาพที่ 11 Effective Part [7]

บทที่ 3
วิธีดำเนินงานวิจัย

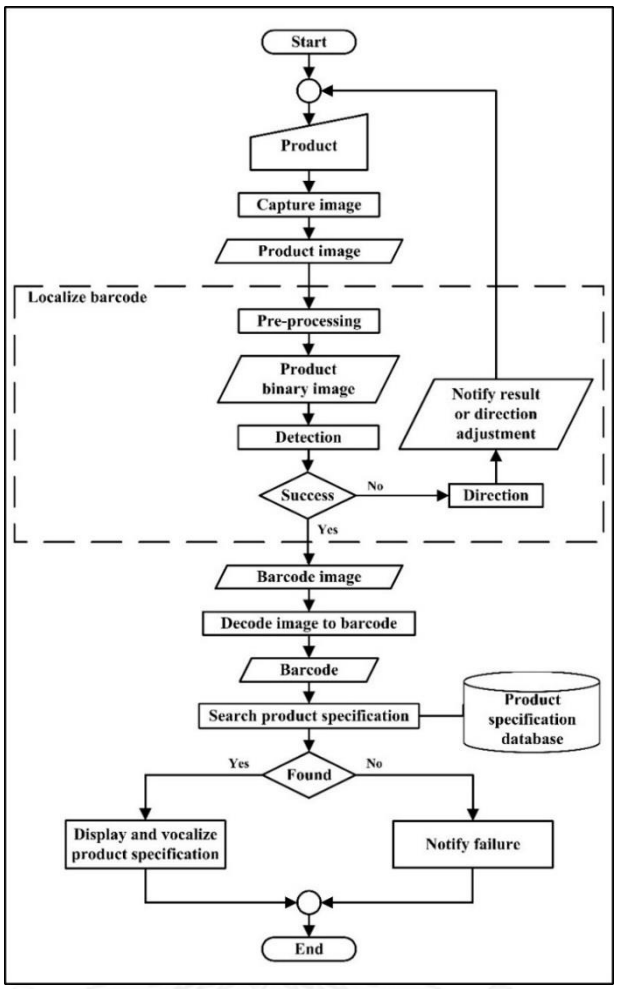
3.1 แนวคิดในการพัฒนา

งานวิจัยนี้ได้นำเสนอแนวทางและพัฒนาแอปพลิเคชันบนมือถือสำหรับตรวจจับหาตำแหน่งบาร์โค้ดบนผลิตภัณฑ์สินค้าอุปโภคบริโภค และช่วยบอกข้อกำหนดผลิตภัณฑ์ให้ผู้พิการทางสายตาทราบด้วยเสียง เพื่อช่วยอำนวยความสะดวกให้กับผู้พิการทางสายตาได้ทราบถึงข้อกำหนดของผลิตภัณฑ์นั้นๆ โดยการพัฒนาระบบจะใช้ซอฟต์แวร์อีคลิปส์ในการพัฒนา เนื่องจากเป็นซอฟต์แวร์โอเพ่นซอร์ส และมีความเหมาะสมในการประยุกต์ใช้งาน รวมทั้งสามารถนำไลบรารีหรือซอฟต์แวร์อื่นมาประยุกต์ใช้ร่วมกันได้ เช่น OpenCV, Zxing [8], Vaja [9] และ SQLite [10] วิธีการที่นำเสนอเริ่มต้นจากการถ่ายภาพผลิตภัณฑ์อุปโภคบริโภคจากผู้ใช้งาน โดยระบบจะมีการโฟกัสภาพให้อัตโนมัติเพื่อรองรับการใช้งานของผู้พิการทางสายตา หลังจากได้ภาพถ่ายผลิตภัณฑ์แล้ว จะทำการตรวจสอบหาตำแหน่งบาร์โค้ดบนภาพผลิตภัณฑ์ นอกจากจะตรวจสอบหาตำแหน่งบาร์โค้ดแล้ว ในกรณีที่บาร์โค้ดไม่สมบูรณ์ ขั้นตอนนี้ยังแนะนำทิศทางในการถ่ายภาพผลิตภัณฑ์ครั้งต่อไปให้ผู้ใช้งานทราบด้วยเสียง เพื่อให้ผู้ใช้งานสามารถถ่ายภาพผลิตภัณฑ์ที่มีบาร์โค้ดอยู่ได้ ส่วนกรณีที่บาร์โค้ดครบสมบูรณ์จะทำการถอดรหัสบาร์โค้ดจากภาพผลิตภัณฑ์ และนำรหัสบาร์โค้ดที่ได้ไปสืบค้นข้อกำหนดผลิตภัณฑ์จากฐานข้อมูลข้อกำหนดผลิตภัณฑ์ และแสดงผลข้อกำหนดผลิตภัณฑ์ที่ได้จากการสืบค้นผ่านหน้าจอสมาร์ตโฟน พร้อมทั้งอ่านข้อกำหนดผลิตภัณฑ์ให้ผู้ใช้งานทราบด้วยเสียง โดยใช้เทคโนโลยีของซอฟต์แวร์ Vaja ซึ่งสามารถรองรับได้ทั้งภาษาไทยและภาษาอังกฤษ รายละเอียดภาพรวมของระบบจะแสดงในภาพที่ 12

จากภาพรวมของระบบสามารถอธิบายการดำเนินการตัวอ่านบาร์โค้ดผลิตภัณฑ์บนมือถือได้โดยจะมีข้อมูลนำเข้าเป็นภาพบาร์โค้ดบนผลิตภัณฑ์ ซึ่งการทำงานสามารถแบ่งออกเป็น 5 ส่วน ดังนี้

1. การถ่ายภาพ (Capture image)
2. การระบุตำแหน่งบาร์โค้ด (Localize barcode)
3. การถอดรหัสบาร์โค้ด (Decode barcode)
4. การค้นหาข้อกำหนดผลิตภัณฑ์ (Search product specification)
5. การแสดงผลข้อกำหนดผลิตภัณฑ์ และอ่านข้อกำหนดผลิตภัณฑ์ (Display and vocalize product specification)

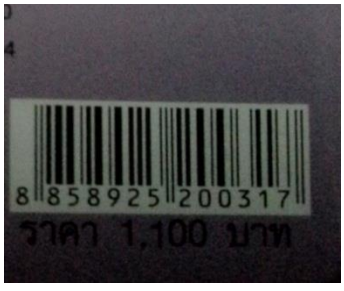
จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University



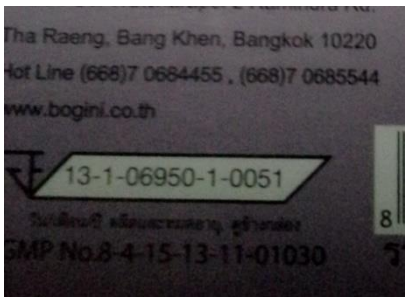
ภาพที่ 12 ภาพรวมการดำเนินการตัวอ่านบาร์โค้ดผลิตภัณฑ์บนมือถือ

3.2 การถ่ายภาพ

การถ่ายภาพ เป็นขั้นตอนการถ่ายภาพผลิตภัณฑ์อุปโภคบริโภค ระบบจะทำการโฟกัสภาพให้อัตโนมัติ เพื่อให้ภาพผลิตภัณฑ์ที่ได้มีความคมชัด ซึ่งภาพผลิตภัณฑ์ที่ได้อาจจะมีบาร์โค้ดหรือไม่มีบาร์โค้ดก็ได้ขึ้นอยู่กับการถ่ายภาพของผู้ใช้งาน โดยภาพผลิตภัณฑ์ที่ได้จะถูกนำเข้ากระบวนการ ระบุตำแหน่งบาร์โค้ดต่อไป ภาพที่ได้จากกระบวนการถ่ายภาพ แสดงในภาพที่ 13 และ ภาพที่ 14



ภาพที่ 13 ภาพผลิตภัณฑ์ที่มีบาร์โค้ด



ภาพที่ 14 ภาพผลิตภัณฑ์ที่ไม่มีบาร์โค้ด

3.3 การระบุตำแหน่งบาร์โค้ด

การระบุตำแหน่งบาร์โค้ด เป็นขั้นตอนการตรวจสอบหาตำแหน่งบาร์โค้ดบนภาพผลิตภัณฑ์ โดยประกอบด้วยกระบวนการย่อย ดังนี้

1. การเตรียมภาพ (Pre-processing) : เป็นขั้นตอนการแปลงภาพผลิตภัณฑ์ โดยภาพผลิตภัณฑ์จะถูกแปลงเป็นภาพไบนารีดังแสดงในภาพที่ 15 โดยใช้ฟังก์ชัน Adaptive thresholding [11] ในไลบรารี OpenCV จากนั้นจะทำการกร่อนภาพและขยายภาพโดยใช้ฟังก์ชัน Eroding [12] และ Dilating [13] ในไลบรารี OpenCV เพื่อลดสิ่งรบกวนและเพิ่มขนาดส่วนของเส้นบาร์โค้ด ซึ่งจะช่วยในการตรวจสอบหาตำแหน่งบาร์โค้ดในกระบวนการถัดไป

ภาพที่ 15 ตัวอย่างภาพผลิตภัณฑ์หลังผ่านกระบวนการเตรียมภาพ

2. การตรวจสอบ (Detection) : เป็นขั้นตอนในการตรวจสอบหาตำแหน่งของบาร์โค้ดบนภาพผลิตภัณฑ์โดยใช้ฟังก์ชัน Finding contours [14] ในไลบรารี OpenCV สำหรับการหาส่วนของภาพบาร์โค้ด และใช้ฟังก์ชัน Hough Line Transform [15] ในไลบรารี OpenCV สำหรับสร้างเส้นไว้ตรวจสอบ Barcode ดังแสดงในภาพที่ 16 โดยมีเงื่อนไขการตรวจสอบภาพบาร์โค้ดบนผลิตภัณฑ์ ดังนี้

    - กรณีที่สามารถตรวจสอบภาพบาร์โค้ดได้ครบสมบูรณ์ จะนำภาพบาร์โค้ดที่ได้เข้าสู่กระบวนการถอดรหัสบาร์โค้ด
    - กรณีที่สามารถตรวจสอบภาพบาร์โค้ดได้บางส่วน จะนำไปสู่กระบวนการชี้นำทาง (Direction)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

- กรณีที่ไม่สามารถตรวจสอบภาพบาร์โค้ดได้ ระบบจะแจ้งเตือนสถานะการตรวจสอบบาร์โค้ดด้วยเสียงให้กับผู้ใช้งาน

ภาพที่ 16 ตัวอย่างภาพผลิตภัณฑ์ที่ได้จากฟังก์ชัน Hough Line Transform

3. การชี้นำทาง (Direction) **:** เป็นขั้นตอนที่เกิดขึ้นในกรณีที่ตรวจสอบภาพบาร์โค้ดเจอบางส่วน โดยขั้นตอนนี้จะแนะนำการวางตำแหน่งกล้องในการถ่ายภาพผลิตภัณฑ์ในครั้งต่อไปให้กับผู้ใช้งานด้วยเสียง ตัวอย่างภาพบาร์โค้ดบนผลิตภัณฑ์ที่ไม่สมบูรณ์จะแสดงในภาพที่ 17 และภาพที่ 18 กรณีในภาพที่ 17 ผู้ใช้งานจะได้รับการแนะนำให้เลื่อนตำแหน่งกล้องไปทางด้านซ้าย และกรณีในภาพที่ 18 ผู้ใช้งานจะได้รับการแนะนำให้เลื่อนตำแหน่งกล้องไปทางด้านขวา

ภาพที่ 17 ตัวอย่างภาพบาร์โค้ดที่เส้นบาร์โค้ดขาดหายทางด้านซ้าย

ภาพที่ 18 ตัวอย่างภาพบาร์โค้ดที่เส้นบาร์โค้ดขาดหายทางด้านขวา

3.4 การถอดรหัสบาร์โค้ด

การถอดรหัสบาร์โค้ด เป็นขั้นตอนการแปลงภาพบาร์โค้ดเป็นรหัสบาร์โค้ด โดยใช้ไลบรารี (library) Zxing [8] ดังแสดงในภาพที่ 19

```
private void decode(byte[] data, int width, int height) {
	long start = System.currentTimeMillis();
	Result rawResult = null;
	PlanarYUVLuminanceSource source =
	activity.getCameraManager().buildLuminanceSource(data, width,
height);
	if (source != null) {
	BinaryBitmap bitmap = new BinaryBitmap(new
HybridBinarizer(source));
	try {
		rawResult = multiFormatReader.decodeWithState(bitmap);
	} catch (ReaderException re) {
	// continue
	} finally {
		multiFormatReader.reset();
	}
}
```

ภาพที่ 19 ตัวอย่างฟังก์ชันการถอดรหัสบาร์โค้ดในไลบรารี Zxing

3.5 การค้นหาข้อกำหนดผลิตภัณฑ์

การค้นหาข้อกำหนดผลิตภัณฑ์ เป็นขั้นตอนการค้นหาข้อกำหนดผลิตภัณฑ์จากฐานข้อมูล โดยจะใช้รหัสบาร์โค้ดในการค้นหาข้อมูล ฐานข้อมูลที่ใช้เก็บข้อกำหนดผลิตภัณฑ์นั้นเป็น SQLite [10] ตัวอย่างฐานข้อมูลข้อกำหนดผลิตภัณฑ์จะแสดงในภาพที่ 20 โดยมีเงื่อนไขการค้นหาข้อกำหนดผลิตภัณฑ์ ดังนี้

- กรณีที่ค้นหาข้อกำหนดผลิตภัณฑ์พบ ข้อมูลผลิตภัณฑ์จะถูกส่งไปยังกระบวนการแสดงผลข้อมูลผลิตภัณฑ์ และอ่านข้อมูลผลิตภัณฑ์
- กรณีที่ค้นหาข้อกำหนดผลิตภัณฑ์ไม่พบ ระบบจะแจ้งเตือนสถานะการค้นหาข้อกำหนดผลิตภัณฑ์ด้วยเสียงให้กับผู้ใช้งาน

	barcode	productname	brand	size	flavour	weight	price	category
1	8858925200317	อาหารเสริม	ออไรท์	30 แคปซูล		17.88 กรัม	1,100 ฿	เสริมอาหาร
2	8851123764186	Hair Gel	เอ็กซิท			100 กรัม	550 ฿	ผลิตภัณฑ์ดูแลเส้นผม
3	8851019010137	ป๊อกกี้	กูลิโกะ		ช็อกโกแลต	47 กรัม	15 ฿	ขนม

ภาพที่ 20 ตัวอย่างฐานข้อมูลข้อกำหนดผลิตภัณฑ์ใน SQLite

3.6 การแสดงผลข้อกำหนดผลิตภัณฑ์ และอ่านข้อกำหนดผลิตภัณฑ์

การแสดงผลข้อกำหนดผลิตภัณฑ์ และอ่านข้อกำหนดผลิตภัณฑ์ เป็นขั้นตอนการแสดงผลข้อกำหนดผลิตภัณฑ์จากฐานข้อมูลที่ได้จากกระบวนการค้นหาข้อกำหนดผลิตภัณฑ์ ดังแสดงในภาพที่ 21

ภาพที่ 21 ผลลัพธ์ที่ได้จากกระบวนการแสดงผลข้อกำหนดผลิตภัณฑ์

ส่วนการอ่านข้อกำหนดบนผลิตภัณฑ์ ระบบจะอ่านข้อกำหนดผลิตภัณฑ์ที่แสดงบนจอสมาร์ตโฟนไล่จากบนลงล่าง และซ้ายไปขวา โดยใช้ซอฟต์แวร์ Vaja [9] โดยการเรียกใช้ Vaja จะแสดงในภาพที่ 22

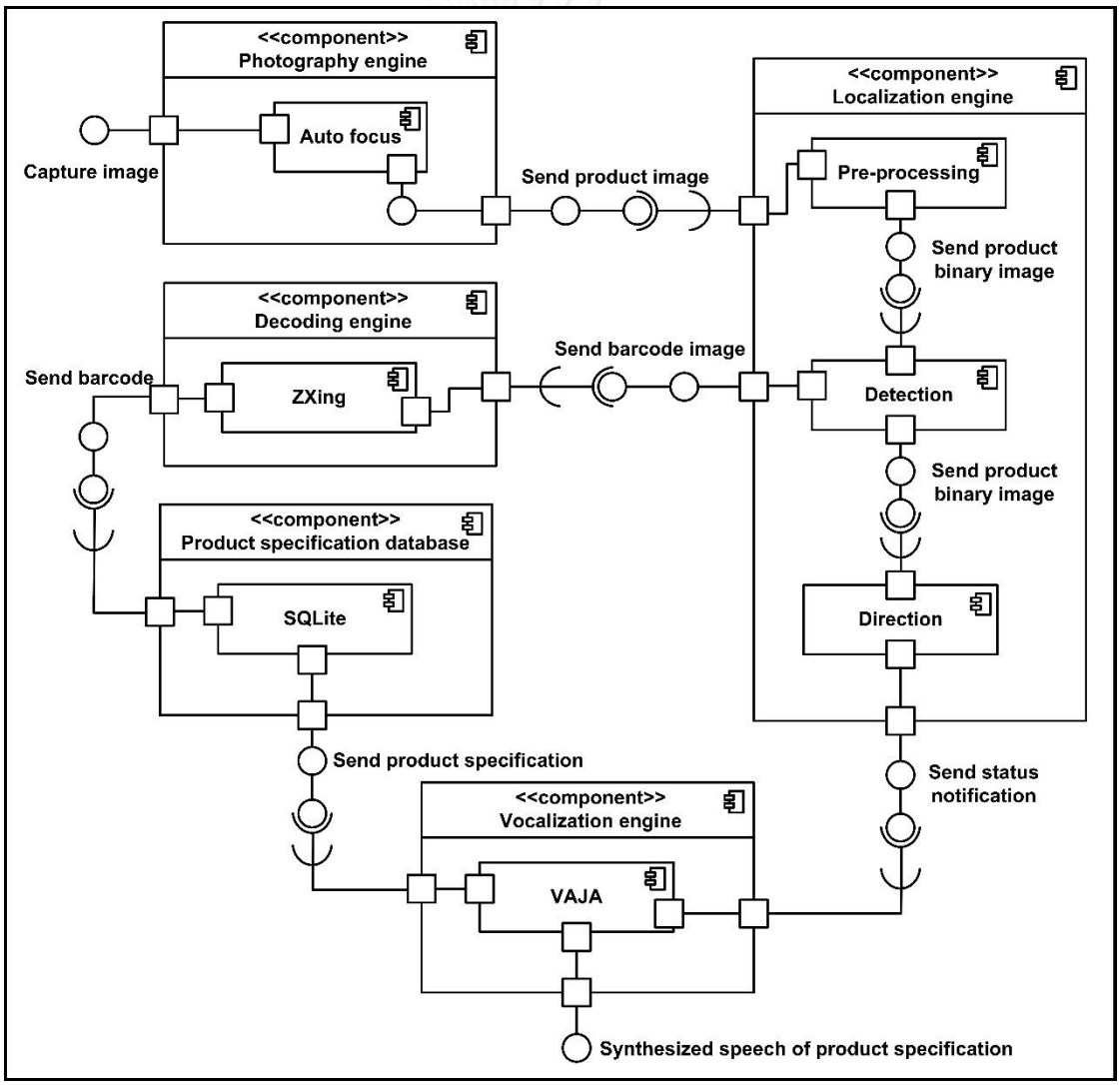
```
public void checkTTSEngineInstalled(String packageName) {
       boolean isVajaInstalled = isAppInstalled(VAJA_TTS_ENGINE);
       if(isVajaInstalled) {
              if(tts == null)
              tts = new TextToSpeech(Main.this, null, VAJA_TTS_ENGINE);
              } else if(!isVajaInstalled){
              Toast.makeText(getApplicationContext()
              , "กรุณาติดตั้ง VAJA Text-to-Speech Engine"
              , Toast.LENGTH_LONG).show();
              Intent vajaIntent = new Intent(Intent.ACTION_VIEW);
              vajaIntent.setData(Uri.parse("market://details?id=" +
              VAJA_TTS_ENGINE));
               startActivity(vajaIntent);
       }
}
```

ภาพที่ 22 การเรียกใช้ซอฟต์แวร์ Vaja

บทที่ 4
การออกแบบและพัฒนาระบบ

4.1 สถาปัตยกรรมระบบ

สถาปัตยกรรมระบบที่พัฒนาขึ้นแบ่งออกเป็น 5 ส่วนหลัก คือ ตัวจับภาพผลิตภัณฑ์ ตัวตรวจสอบภาพบาร์โค้ด ตัวแปลงรหัสบาร์โค้ด ตัวค้นหาข้อกำหนดผลิตภัณฑ์ และตัวแสดงผลข้อกำหนดผลิตภัณฑ์ โดยมีแผนภาพสถาปัตยกรรมระบบดังภาพที่ 23



ภาพที่ 23 สถาปัตยกรรมระบบที่พัฒนา

ซึ่งตัวจับภาพผลิตภัณฑ์มีหน้าที่ในการบันทึกภาพผลิตภัณฑ์แล้วส่งไปยังตัวตรวจสอบภาพบาร์โค้ด ตัวตรวจสอบภาพบาร์โค้ดมีหน้าที่ในการตรวจจับภาพบาร์โค้ดบนภาพผลิตภัณฑ์ หรือชี้นำทางในการถ่ายภาพครั้งถัดไปให้กับผู้ใช้งาน ตัวแปลงรหัสบาร์โค้ดมีหน้าที่ในการแปลงภาพบาร์โค้ดเป็นรหัสบาร์โค้ด ตัวค้นหาข้อกำหนดผลิตภัณฑ์มีหน้าที่ในการนำรหัสบาร์โค้ดที่ได้จากการถอดรหัสมา

ค้นหาในฐานข้อมูลข้อกำหนดผลิตภัณฑ์ และส่งข้อกำหนดผลิตภัณฑ์ที่ค้นหาได้ไปยังตัวแสดงผลข้อกำหนดผลิตภัณฑ์ ตัวแสดงผลข้อกำหนดผลิตภัณฑ์มีหน้าที่ในการแสดงผลข้อกำหนดผลิตภัณฑ์ผ่านทางหน้าจอสมาร์ตโฟน พร้อมทั้งบอกรายละเอียดข้อกำหนดผลิตภัณฑ์ให้ผู้ใช้งานทราบด้วยเสียง

4.2 สภาพแวดล้อมและเครื่องมือที่ใช้ในการพัฒนา

สภาพแวดล้อมและเครื่องมือที่ใช้ในการพัฒาระบบประกอบด้วยรายการฮาร์ดแวร์และซอฟต์แวร์ดังต่อไปนี้

4.2.1 สภาพแวดล้อม

1. หน่วยประมวลผลอินเทล คอร์ ไอ3-2.13 กิกะเฮิร์ต (CPU Intel Core i3 2.13GHz)
2. หน่วยความจำ 3 กิกะไบต์ (3 GB RAM)
3. ฮาร์ดดิสก์ความจุ 500 กิกะไบต์ (500 GB HDD)
4. ระบบปฏิบัติการไมโครซอฟต์วินโดว์7 (Microsoft Windows 7) แบบ 32 บิต

4.2.2 เครื่องมือที่ใช้ในการพัฒนา

1. อีคลิปส์ (Eclipse)
2. เอสคิวไลท์ ดาต้าเบส บราวเซอร์ 2.0 (SQLite Database Browser 2.0)

4.3 การพัฒนาระบบ

การพัฒนาระบบในส่วนต่างๆ จะถูกพัฒนาโดยซอฟต์แวร์อีคลิปส์ เป็นเครื่องมือหลัก และติดตั้งซอฟต์แวร์เอสคิวไลท์ ดาต้าเบส บราวเซอร์ 2.0 เพื่อใช้เป็นฐานข้อมูลระบบร่วมในงานที่พัฒนาระบบที่ได้จะอยู่ในลักษณะของแอปพลิเคชันบนมือถือระบบปฏิบัติการแอนดรอยด์

4.3.1 ตัวจับภาพผลิตภัณฑ์

การพัฒนาตัวจับภาพผลิตภัณฑ์สามารถแบ่งออกเป็น 2 ส่วนย่อย ดังนี้

1. ส่วนโฟกัสภาพ : มีการเรียกใช้ฟังก์ชัน Autofocus ซึ่งมีอยู่ในระบบกล้องบนสมาร์ตโฟนสำหรับใช้โฟกัสภาพให้มีความคมชัด โดยจะมีฟังก์ชัน โดยฟังก์ชัน onAutofocus สำหรับตรวจสอบสถานะการโฟกัสของกล้อง จะแสดงในภาพที่ 24 และการเรียกใช้งานฟังก์ชัน Autofocus จะแสดงในภาพที่ 25

```
public void onAutoFocus(boolean success, Camera camera) {
        Log.d("CameraSystem","onAutoFocus");
}
```

ภาพที่ 24 ฟังก์ชัน onAutofocus

สำหรับฟังก์ชัน onAutofocus จะมีพารามิเตอร์ 2 ตัว ได้แก่ boolean success สำหรับบอกสถานะการโฟกัสของกล้อง และ Camera camera เป็นคลาสของกล้องที่เรียกใช้งาน

```
public void onSensorChanged(SensorEvent event) {
        if(Math.abs(event.values[0] - motionX) > 1
           || Math.abs(event.values[1] - motionY) > 1
           || Math.abs(event.values[2] - motionZ) > 1 ) {
            Log.d("Camera System", "Refocus");
            try {
                    mCamera.autoFocus(this);
            } catch (RuntimeException e) { }
            motionX = event.values[0];
            motionY = event.values[1];
            motionZ = event.values[2];
        }
}
```

ภาพที่ 25 การเรียกใช้งานฟังก์ชัน Autofocus อัตโนมัติ เมื่อมีการเคลื่อนย้ายกล้อง

สำหรับการเรียกใช้งานฟังก์ชัน Autofocus จะเป็นการใช้ sensor ตรวจจับตำแหน่งของตัวเครื่อง เมื่อมีค่าแกน X,Y,Z เปลี่ยนไปมากกว่าค่าที่กำหนด จะมีการเรียกฟังก์ชัน Autofocus ใหม่อัตโนมัติ จากภาพที่ 25 จะตั้งค่าแต่ละแกนไว้ที่ 1 เมื่อเรียกใช้ฟังก์ชัน Autofocus แล้วจะทำการเก็บค่าแต่ละแกนไว้ใหม่

2. ส่วนถ่ายภาพผลิตภัณฑ์ : มีการพัฒนาฟังก์ชัน onPictureTaken สำหรับถ่ายภาพผลิตภัณฑ์ ซึ่งผู้พัฒนาจะทำการบันทึกภาพผลิตภัณฑ์ในรูปแบบของ .jpg ฟังก์ชัน onPictureTaken จะแสดงในภาพที่ 26

```
public void onPictureTaken(byte[] arg0, Camera arg1) {
    int imageNum = 0;
    Intent imageIntent = new
    Intent(MediaStore.ACTION_IMAGE_CAPTURE);
    File imagesFolder =
        new File(Environment.getExternalStorageDirectory()
                    , "DCIM/CameraSnap");
    imagesFolder.mkdirs();
    String fileName = "IMG_" + String.valueOf(imageNum) +
    ".jpg";
    File output = new File(imagesFolder, fileName);
    while (output.exists()){
        imageNum++;
        fileName = "IMG_" + String.valueOf(imageNum) +
        ".jpg";
        output = new File(imagesFolder, fileName);
    }
    Uri uri = Uri.fromFile(output);
    imageIntent.putExtra(MediaStore.EXTRA_OUTPUT, uri);
    ContentValues image = new ContentValues();
```

```
    String dateTaken = DateFormat.getDateTimeInstance()
                 .format(Calendar.getInstance().getTime());
           image.put(Images.Media.TITLE, output.toString());
           image.put(Images.Media.DISPLAY_NAME,
           output.toString());
           image.put(Images.Media.DATE_ADDED, dateTaken);
           image.put(Images.Media.DATE_TAKEN, dateTaken);
           image.put(Images.Media.DATE_MODIFIED, dateTaken);
           image.put(Images.Media.MIME_TYPE, "image/jpg");
           image.put(Images.Media.ORIENTATION, 0);
    String path =
    output.getParentFile().toString().toLowerCase();
    String name =
    output.getParentFile().getName().toLowerCase();
    image.put(Images.ImageColumns.BUCKET_ID,
    path.hashCode());
    image.put(Images.ImageColumns.BUCKET_DISPLAY_NAME,
name);
    image.put(Images.Media.SIZE, output.length());
    image.put(Images.Media.DATA, output.getAbsolutePath());
    OutputStream os;
    try {
    os = getContentResolver().openOutputStream(uri);
    os.write(arg0);
    os.flush();
    os.close();
    Toast.makeText(test.this, fileName,
        Toast.LENGTH_SHORT).show();
    }
    catch (FileNotFoundException e) { }
    catch (IOException e) { }

Log.d("Camera","Restart Preview");
         mCamera.stopPreview();
         mCamera.startPreview();
         saveState = false;
     }
```

ภาพที่ 26 ฟังก์ชัน onPictureTaken

สำหรับฟังก์ชัน onPictureTaken จะมีรูปแบบในการบันทึกชื่อไฟล์ภาพเป็น IMG_+ลำดับของภาพ+.jpg ยกตัวอย่างเช่น IMG_1.jpg ซึ่งก่อนการบันทึกภาพทุกครั้งจะมีการตรวจสอบลำดับของภาพที่มีล่าสุดก่อน จากนั้นจึงบันทึกภาพโดยเพิ่มลำดับภาพขึ้น 1 ลำดับ

4.3.2 ตัวตรวจสอบภาพบาร์โค้ด

การพัฒนาตัวตรวจสอบภาพบาร์โค้ด แบ่งส่วนการพัฒนาเป็น 3 ส่วนย่อย ได้แก่ ส่วนการเตรียมภาพ ส่วนการตรวจสอบภาพ และส่วนการชี้นำทาง

1. ส่วนการเตรียมภาพ : การพัฒนาส่วนของการเตรียมภาพ จะมีการแปลงภาพเป็นไบนารี โดยใช้ฟังก์ชัน Adaptive thresholding ใน OpenCV เพื่อปรับภาพให้ง่ายต่อการตรวจสอบหาตำแหน่งบาร์โค้ดบนภาพ ซึ่งก่อนจะแปลงภาพเป็นไบนารี จะต้องนำภาพที่ได้จากกระบวนการถ่ายภาพผลิตภัณฑ์มาแปลงให้อยู่ในรูปแบบของ Mat ก่อน จากนั้นจะทำการลดสิ่งรบกวนและเพิ่มขนาดส่วนของเส้นบาร์โค้ดโดยใช้ฟังก์ชัน Eroding และ Dilating ในไลบรารี OpenCV ซึ่งจะช่วยในการตรวจสอบหาตำแหน่งบาร์โค้ดในกระบวนการถัดไป โดยการแปลงรูปแบบของภาพเป็น Mat จะแสดงในภาพที่ 27 และฟังก์ชัน Adaptive thresholding จะแสดงในภาพที่ 28

```
imgsourcegray =
Highgui.imread(output.getAbsolutePath(),Highgui.CV_LOAD_IMAGE_GR
AYSCALE);
imgoutgray = new Mat();
```

ภาพที่ 27 การแปลงรูปแบบของภาพเป็น Mat

จากภาพที่ 27 เป็นการโหลดภาพที่ได้จากกระบวนการถ่ายภาพผลิตภัณฑ์เก็บในตัวแปร โดยภาพที่ได้จะเป็นภาพ grayscale และมีรูปแบบเป็น Mat เพื่อจะนำไปใช้เป็นภาพนำเข้าในการเรียกใช้ฟังก์ชัน Adaptive thresholding ดังแสดงในภาพที่ 29

```
Public static void adaptiveThreshold(Mat src,Mat dst,
                    double maxValue,int adaptiveMethod,
                    int thresholdType,
                    int blockSize,double C)
```

ภาพที่ 28 ฟังก์ชัน Adaptive thresholding ใน OpenCV [11]

จากภาพที่ 28 จะเห็นได้ว่า ฟังก์ชัน Adaptive thresholding มีพารามิเตอร์ทั้งหมด 7 ตัว ซึ่งแต่ละค่าสามารถอธิบายได้ ดังนี้

- src - ภาพก่อนการแปลง
- dst - ภาพหลังการแปลง
- maxValue - ค่าที่ไม่เท่ากับ 0 เป็นค่าพิกเซล
- adaptiveMethod - อัลกอริทึมที่ใช้ ได้แก่ ADAPTIVE_THRESH_MEAN_C หรือ ADAPTIVE_THRESH_GAUSSIAN_C
- thresholdType - ชนิดของ threshold ได้แก่ THRESH_BINARY หรือ THRESH_BINARY_INC
- blockSize - ขนาดของพิกเซลใกล้เคียงที่ใช้ในการคำนวณค่า threshold
- c - ค่าคงที่

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ส่วนการเรียกใช้งานฟังก์ชัน Adaptive thresholding จะแสดงในภาพที่ 29 โดยจะมีการเรียกใช้งานฟังก์ชัน Adaptive thresholding เมื่อมีการโหลดภาพผลิตภัณฑ์สินค้าจากตัวจับภาพผลิตภัณฑ์

```
Imgproc.adaptiveThreshold(imgsourcegray, imgoutgray, 255,
Imgproc.ADAPTIVE_THRESH_MEAN_C, Imgproc.THRESH_BINARY, 15, 4);
```

ภาพที่ 29 การเรียกใช้งานฟังก์ชัน Adaptive thresholding ใน OpenCV

ฟังก์ชัน Eroding และ Dilating เป็นฟังก์ชันกร่อนภาพและขยายภาพ ซึ่งช่วยเพิ่มประสิทธิภาพในการตรวจสอบหาภาพบาร์โค้ดบนภาพผลิตภัณฑ์ในกระบวนการถัดไป โดยฟังก์ชัน Eroding และ Dilating จะแสดงในภาพที่ 30 และภาพที่ 31 ตามลำดับ

```
public static void erode(Mat src,Mat dst,Mat kernel)
```

ภาพที่ 30 ฟังก์ชัน Eroding ใน OpenCV [12]

```
public static void dilate(Mat src,Mat dst,Mat kernel)
```

ภาพที่ 31 ฟังก์ชัน Dilating ใน OpenCV [13]

ส่วนการเรียกใช้งานฟังก์ชัน Eroding และ Dilating จะแสดงในภาพที่ 32 และภาพที่ 33 โดยจะมีการเรียกใช้งานฟังก์ชัน Eroding และ Dilating หลังจากแปลงภาพผลิตภัณฑ์เป็นภาพไบนารี

```
Imgproc.erode(imgoutgray,imgoutgray,
             Imgproc.getStructuringElement(Imgproc.MORPH_RECT,
             new Size(5, 5)));
```

ภาพที่ 32 การเรียกใช้ฟังก์ชัน Eroding ใน OpenCV

```
Imgproc.dilate(imgoutgray, imgoutgray,
             Imgproc.getStructuringElement(Imgproc.MORPH_RECT,
             new Size(9, 9)));
```

ภาพที่ 33 การเรียกใช้ฟังก์ชัน Dilating ใน OpenCV

2. ส่วนการตรวจสอบภาพ : การพัฒนาส่วนของการตรวจสอบภาพนั้น หลักสำคัญอยู่ที่การตรวจสอบหาแท่งบาร์บนภาพผลิตภัณฑ์ เพื่อหาตำแหน่งของภาพบาร์โค้ดบนภาพผลิตภัณฑ์ โดยขั้นตอนแรกจะมีการเรียกใช้ฟังก์ชัน Finding contours [14] ในไลบรารี OpenCV สำหรับสร้างกรอบตรวจจับหาเส้นบาร์โค้ดบนภาพผลิตภัณฑ์ จากนั้นจะใช้ฟังก์ชัน Hough Line Transform ในไลบรารี OpenCV สำหรับการค้นหาเส้นของภาพที่ได้หลังจากการทำ Finding contours แล้ว โดยฟังก์ชัน Finding contours จะแสดงในภาพที่ 34 และฟังก์ชัน Hough Line Transform จะแสดงในภาพที่ 35

```
public static void findContours(Mat image,
                java.util.List<MatOfPoint> contours,
                Mat hierarchy,int mode,int method)
```

ภาพที่ 34 ฟังก์ชัน Finding contours ใน OpenCV [14]

จากภาพที่ 34 จะเห็นได้ว่า ฟังก์ชัน Finding contours มีพารามิเตอร์ทั้งหมด 5 ตัว ซึ่งแต่ละค่าสามารถอธิบายได้ ดังนี้

- Image - ภาพนำเข้า
- Contours - รูปทรงที่ถูกตรวจพบ โดยแต่ละรูปทรงจะถูกเก็บไว้ในรูปแบบของเวกเตอร์
- Hierarchy - ตัวเลือกของเวกเตอร์เอาท์พุท
- Mode - โหมดของการดึงเส้น ได้แก่ CV_RETR_EXTERNA, CV_RETR_LIST, CV_RETR_CCOM และ CV_RETR_TREE
- Method – วิธีการประมาณรูปทรง ได้แก่ CV_CHAIN_APPROX_NONE, CV_CHAIN_APPROX_SIMPLE และ CV_CHAIN_APPROX_TC89_L1, CV_CHAIN_APPROX_TC89_KCOS

```
public static void HoughLinesP(Mat image,Mat lines,
                        double rho,
                        double theta,int threshold,
                        double minLineLength,
                        double maxLineGap)
```

ภาพที่ 35 ฟังก์ชัน Hough Line Transform ใน OpenCV [15]

จากภาพที่ 35 จะเห็นได้ว่า ฟังก์ชัน Hough Line Transform มีพารามิเตอร์ทั้งหมด 7 ตัว ซึ่งแต่ละค่าสามารถอธิบายได้ ดังนี้

- Image - ภาพนำเข้า
- Lines - เวกเตอร์เอาท์พุทของเส้น
- rho - ความละเอียดของพิกเซล
- theta - ความละเอียดมุมเรเดียน
- threshold - ค่า threshold สะสม ถ้ามีการส่งผลลัพธ์เส้นกลับมา ค่านี้จะมีค่ามากกว่าค่า threshold เดิม

- minLineLength - ค่าความยาวของเส้นที่สามารถส่งผลลัพธ์กลับมาได้ ถ้าความยาวของเส้นน้อยกว่าค่านี้ จะไม่ถูกส่งผลลัพธ์กลับมา
- maxLineGap – ค่ามากสุดของช่องว่างที่จะยอมให้เส้นยังคงสามารถเชื่อมต่อกัน

การเรียกใช้งานฟังก์ชัน Finding contours จะแสดงในภาพที่ 36 โดยภาพนำเข้าจะได้มาจากส่วนของการเตรียมภาพ

```
Imgproc.findContours(imgoutgray, contours, lines2,
                     Imgproc.RETR_TREE,
                     Imgproc.CHAIN_APPROX_SIMPLE);
```

ภาพที่ 36 การเรียกใช้งานฟังก์ชัน Finding contours ใน OpenCV

หลังจากได้รูปทรงผลลัพธ์จากฟังก์ชัน Finding contours ซึ่งจะเก็บอยู่ในตัวแปร contours จากนั้นจะนำผลลัพธ์ที่ได้ไปคัดเลือกค่าที่ดีที่สุด โดยวัดจากขนาดของรูปทรง โดยอัลกอริทึมที่ใช้สำหรับคัดเลือกรูปทรงผลลัพธ์จะแสดงให้เห็นในภาพที่ 37

```
MatOfPoint2f approxCurve = new MatOfPoint2f();
double temp=0;
int choose=0;
for (int i=0; i<contours.size(); i++)
{
  //แปลง  contours(i) จากรูปแบบ  MatOfPoint ให้อยู่ในรูปแบบ  MatOfPoint2f
   MatOfPoint2f contour2f = new MatOfPoint2f(
contours.get(i).toArray() );

  //กระบวนการ  mMOP2f1 โดยจะต้องมีรูปแบบเป็น  MatOfPoint2f
   double approxDistance = Imgproc.arcLength(contour2f,
true)*0.02;
   Imgproc.approxPolyDP(contour2f, approxCurve,
approxDistance, true);

  //แปลงให้อยู่ในรูปแบบ MatOfPoint
   MatOfPoint points = new MatOfPoint( approxCurve.toArray()
);

  // สร้าง  bounding rect ของ  contour
   Rect rect = Imgproc.boundingRect(points);
   if(rect.area()>temp)
   {
       startx=rect.x;
       starty=rect.y;
       endx=rect.x+rect.width;
       endy=rect.y+rect.height;
       temp=rect.area();
       choose=i;
   }
}
```

ภาพที่ 37 อัลกอริทึมที่ใช้สำหรับคัดเลือกรูปทรงผลลัพธ์

การเรียกใช้งานฟังก์ชัน Hough Line Transform จะเรียกใช้เมื่อได้รูปทรงที่ดีที่สุด และตัดภาพในส่วนของรูปทรงที่เลือกแล้ว เพื่อนำมาเป็นภาพนำเข้าของ ฟังก์ชัน Hough Line Transform โดยกระบวนการตัดภาพผลิตภัณฑ์ในส่วนที่เป็นรูปทรงที่ถูกเลือกจะแสดงในภาพที่ 38 และการเรียกใช้งานฟังก์ชัน Hough Line Transform จะแสดงในภาพที่ 39

```
MatOfPoint2f contour2f = new MatOfPoint2f(
contours.get(choose).toArray() );
double approxDistance = Imgproc.arcLength(contour2f,
true)*0.02;
Imgproc.approxPolyDP(contour2f, approxCurve, approxDistance,
true);
MatOfPoint points = new MatOfPoint( approxCurve.toArray() );
Rect rect = Imgproc.boundingRect(points);

// เลือกส่วนของภาพผลิตภัณฑ์ที่เป็นส่วนของรูปทรงที่ถูกเลือก
Mat roi = new Mat(imgsourcegray,new
Rect(rect.x,rect.y,rect.width,rect.height));
```

ภาพที่ 38 กระบวนการตัดภาพผลิตภัณฑ์ในส่วนที่เป็นรูปทรงที่ถูกเลือก

```
Imgproc.HoughLinesP(roi, lines, 1, Math.PI/180, threshold,
                    minLineSize, lineGap);
```

ภาพที่ 39 การเรียกใช้งานฟังก์ชัน Hough Line Transform ใน OpenCV

หลังจากเรียกใช้งานฟังก์ชัน Hough Line Transform จะทำการตรวจสอบจำนวนเส้นผลลัพธ์ที่ได้ สามารถแบ่งกรณีได้ ดังนี้

- กรณีมีจำนวนเส้นในระนาบเดียวกันมากกว่าหรือเท่ากับที่กำหนด จะส่งภาพผลิตภัณฑ์ไปยังตัวแปลงบาร์โค้ด
- กรณีมีเส้นน้อยกว่าที่กำหนด แต่ไม่น้อยกว่าค่าต่ำสุดของจำนวนเส้นที่สามารถพบบาร์โค้ด จะทำการตรวจสอบเส้นผลลัพธ์ที่ใกล้ขอบของภาพมากสุด เพื่อส่งผลลัพธ์ไปยังส่วนการชี้นำ
- กรณีมีเส้นน้อยกว่าค่าต่ำสุดของจำนวนเส้นที่สามารถพบบาร์โค้ด หรือไม่มีเส้น จะส่งผลลัพธ์ไม่มีตำแหน่งของเส้นไปยังส่วนการชี้นำ

3. ส่วนการชี้นำ : การพัฒนาส่วนของการชี้นำนั้น หลักสำคัญอยู่ที่ผลลัพธ์ตำแหน่งของเส้นที่ใกล้กับขอบของภาพที่ได้จากส่วนการตรวจสอบภาพ โดยผลลัพธ์ที่ได้สามารถแบ่งเป็น 5 กรณี ดังนี้ โดยค่าที่ใช้ในการตรวจสอบในแต่ละกรณีเป็นค่าที่ได้มาจากส่วนการตรวจสอบภาพ

    - กรณีตำแหน่งของเส้นติดขอบซ้าย จะแจ้งให้ผู้ใช้งานเลื่อนกล้องไปทางซ้ายในการถ่ายภาพครั้งถัดไปด้วยเสียง จะแสดงในภาพที่ 40

```
if(startx<2)
{
	check=1;
	tts.speak("บาร์โค้ดตกขอบซ้าย กรุณาเลื่อนกล้องไปทางซ้ายและถ่ายภาพใหม่อีกครั้ง",
TextToSpeech.QUEUE_ADD, null);
}
```

ภาพที่ 40 กรณีตำแหน่งของเส้นติดขอบซ้าย

- กรณีตำแหน่งของเส้นติดขอบขวา จะแจ้งให้ผู้ใช้งานเลื่อนกล้องไปทางขวาในการถ่ายภาพครั้งถัดไปด้วยเสียง จะแสดงในภาพที่ 41

```
if(endx>imgsource.width()-2)
{
	check=1;
	tts.speak("บาร์โค้ดตกขอบขวา กรุณาเลื่อนกล้องไปทางขวาและถ่ายภาพใหม่อีกครั้ง",
TextToSpeech.QUEUE_ADD, null);
}
```

ภาพที่ 41 กรณีตำแหน่งของเส้นติดขอบขวา

- กรณีตำแหน่งของเส้นติดขอบบน จะแจ้งให้ผู้ใช้งานเลื่อนกล้องขึ้นบนในการถ่ายภาพครั้งถัดไปด้วยเสียง จะแสดงในภาพที่ 42

```
if(starty<2)
{
	check=1;
	tts.speak("บาร์โค้ดตกขอบบน กรุณาเลื่อนกล้องไปด้านบนและถ่ายภาพใหม่อีกครั้ง",
TextToSpeech.QUEUE_ADD, null);
}
```

ภาพที่ 42 กรณีตำแหน่งของเส้นติดขอบบน

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

- กรณีตำแหน่งของเส้นติดขอบล่าง จะแจ้งให้ผู้ใช้งานเลื่อนกล้องลงล่างในการถ่ายภาพครั้งถัดไปด้วยเสียง จะแสดงในภาพที่ 43

```
if(endy>imgsource.height()-2)
{
	check=1;
	tts.speak("บาร์โค้ดตกขอบล่าง กรุณาเลื่อนกล้องไปด้านล่างและถ่ายภาพใหม่อีกครั้ง",
TextToSpeech.QUEUE_ADD, null);
}
```

ภาพที่ 43 กรณีตำแหน่งของเส้นติดขอบล่าง

- กรณีไม่มีตำแหน่งของเส้น จะแจ้งสถานะไม่พบบาร์โค้ดให้ผู้ใช้งานทราบด้วยเสียง จะแสดงในภาพที่ 44

```
if(lines3.cols()<=30)
{
	check=0;
	tts.speak("ไม่พบบาร์โค้ด กรุณาถ่ายภาพใหม่อีกครั้ง",
TextToSpeech.QUEUE_ADD, null);
}
```

ภาพที่ 44 กรณีไม่มีตำแหน่งของเส้น

4.3.3 ตัวแปลงรหัสบาร์โค้ด

การพัฒนาตัวแปลงรหัสบาร์โค้ดจะมีการเรียกใช้ฟังก์ชัน decode ในไลบรารี Zxing สำหรับแปลงภาพบาร์โค้ดเป็นรหัสบาร์โค้ด ซึ่งได้แสดงตัวอย่างฟังก์ชัน decode ไว้ในบทที่ 3 โดยกระบวนการถอดรหัสบาร์โค้ดจะมีการหมุนภาพในแกนต่างๆ เพื่อเพิ่มประสิทธิภาพในการถอดรหัสบาร์โค้ด ดังแสดงในภาพที่ 45 และการตรวจสอบผลลัพธ์การถอดรหัสบาร์โค้ดจะแสดงในภาพที่ 46

```
Bitmap bmpRotate = null ;
int width,height ;
int newWidth = 500;
int newHeight = 500;
float scaleWidth,scaleHeight;
LuminanceSource sourceRotate;
BinaryBitmap bitmapRotate ;
Reader readerRotate;
readerRotate = new MultiFormatReader();
for (int degree = 0; degree <=360; degree = degree + 5)
{
	width = bmp.getWidth();
	height = bmp.getHeight();
	Matrix matrix = new Matrix();
	scaleWidth = ((float) newWidth)/height  ;
	scaleHeight = ((float) newHeight)/width ;
	matrix.postScale(scaleWidth, scaleHeight);
	matrix.postRotate(degree);
	bmpRotate = Bitmap.createBitmap(bmp, 0, 0,
                                    width, height, matrix, true);
	int[] intArray1 = new
	int[bmpRotate.getHeight()*bmpRotate.getWidth()];
	bmpRotate.getPixels(intArray1, 0, bmpRotate.getWidth(), 0, 0,
bmpRotate.getWidth(), bmpRotate.getHeight());
	sourceRotate = new RGBLuminanceSource(bmpRotate.getWidth(),
bmpRotate.getHeight(),intArray1);
	bitmapRotate = new BinaryBitmap(new
HybridBinarizer(sourceRotate));
```

```
	try {
		//ถอดรหัสบาร์โค้ด
			result = readerRotate.decode(bitmapRotate);
	}
	catch (NotFoundException e) {
		e.printStackTrace();
	}
	catch (ChecksumException e) {
		e.printStackTrace();
	}
	catch (FormatException e) {
		e.printStackTrace();
	}
	if(result==null)
	{
		Log.i("Rotate", "Rotate null");
	}else{
		break;
	}
}
```

ภาพที่ 45 การหมุนภาพและการถอดรหัสบาร์โค้ด

```
public void onActivityResult(int requestCode, int resultCode
		, Intent intent) {
	if (requestCode == REQUEST_QR_SCAN && resultCode == RESULT_OK) {
		String content = intent.getStringExtra("CONTENT");
		String format = intent.getStringExtra("FORMAT");
		String type = intent.getStringExtra("TYPE");
	  } else if(requestCode == REQUEST_QR_SCAN
			&& resultCode == RESULT_CANCELED) {
		textView1.setText("ไม่สามารถสแกนบาร์โค้ดได้");
		checkTTSEngineInstalled(VAJA_TTS_ENGINE);
		if(tts != null)
		   tts.speak(textView1.getText().toString(),
				TextToSpeech.QUEUE_FLUSH, null);
	  }
    }
```

ภาพที่ 46 การตรวจสอบผลลัพธ์การถอดรหัสบาร์โค้ด

จากภาพที่ 46 จะเป็นฟังก์ชันตรวจสอบผลลัพธ์จากถอดรหัสบาร์โค้ด โดยถ้าผลลัพธ์มีค่าเป็น RESULT_OK แสดงว่าสแกนบาร์โค้ดสำเร็จ จะทำการเก็บค่าที่สแกนได้ลงตัวแปร content, format และ type แต่ถ้าผลลัพธ์มีค่าเท่ากับ RESULT_CANCELED แสดงว่าสแกนบาร์โค้ดไม่สำเร็จ ระบบจะแสดงข้อความ *“ไม่สามารถสแกนบาร์โค้ดได้”* บนหน้าจอสมาร์ตโฟน พร้อมทั้งอ่านออกเสียงให้ผู้ใช้งานฟังโดยเรียกใช้ฟังก์ชัน Text to speech ของ VAJA ให้อ่านข้อความที่แสดงบนหน้าจอ

4.3.4 ตัวค้นหาข้อกำหนดผลิตภัณฑ์

การพัฒนาตัวค้นหาข้อกำหนดผลิตภัณฑ์ จะมีส่วนการพัฒนาสำหรับติดต่อฐานข้อมูล ซึ่งใช้ SQLite สำหรับเป็นฐานข้อมูลข้อกำหนดผลิตภัณฑ์ และส่วนการสืบค้นข้อกำหนดผลิตภัณฑ์จากฐานข้อมูล ซึ่งใช้ผลลัพธ์ที่ได้จากกระบวนการถอดรหัสบาร์โค้ดเป็นคีย์ในการสืบค้นข้อมูล ดังแสดงในภาพที่ 47 และคลาสของฐานข้อมูลที่สร้างไว้จะแสดงในภาพที่ 48

```
SQLiteDatabase myDb;
MyDbHelper myHelper;
Cursor Tmpproduct;

myHelper = new MyDbHelper(this);
myDb = myHelper.getWritableDatabase();
Tmpproduct = myDb.rawQuery("SELECT *"
             + " FROM " + MyDbHelper.TABLE_NAME
             +"Where barcode =”+ content , null);
if(Tmpproduct!= null){
       Tmpproduct.moveToFirst();
       String shwbarcode =
       Tmpproduct.getString(Tmpproduct.getColumnIndex(MyDbHelper.COL_barc
ode));
       String shwproductname =
       Tmpproduct.getString(Tmpproduct.getColumnIndex(MyDbHelper.COL_prod
uctname));
       String shwbrand =
       Tmpproduct.getString(Tmpproduct.getColumnIndex(MyDbHelper.COL_bran
d));
       String shwsize =
       Tmpproduct.getString(Tmpproduct.getColumnIndex(MyDbHelper.COL_size
));
       String shwflavour =
       Tmpproduct.getString(Tmpproduct.getColumnIndex(MyDbHelper.COL_flav
our));
       String shwweight =
       Tmpproduct.getString(Tmpproduct.getColumnIndex(MyDbHelper.COL_weig
ht));
       String shwprice =
       Tmpproduct.getString(Tmpproduct.getColumnIndex(MyDbHelper.COL_pric
e));
       String shwcategory =
       Tmpproduct.getString(Tmpproduct.getColumnIndex(MyDbHelper.COL_cate
gory));
}else
{ String shwbarcode = null ;}
```

ภาพที่ 47 การสืบค้นข้อกำหนดผลิตภัณฑ์จากฐานข้อมูล

จากภาพที่ 47 เป็นการแสดงการพัฒนาในส่วนการติดต่อฐานข้อมูล และสืบค้นข้อกำหนดผลิตภัณฑ์จากรหัสบาร์โค้ด และเก็บข้อมูลที่สืบค้นลงตารางชั่วคราว และนำค่าจากตารางชั่วคราวมาเก็บลงตัวแปร เพื่อเตรียมการแสดงผลลัพธ์ในขั้นตอนแสดงผลข้อกำหนดผลิตภัณฑ์

```
class MyDbHelper extends SQLiteOpenHelper {
       private static final String DB_NAME = "product specification
database";
       private static final int DB_VERSION = 1;
       public static final String TABLE_NAME = "Product";
       public static final String COL_barcode = "barcode";
       public static final String COL_productname = "productname";
       public static final String COL_brand = "brand";
       public static final String COL_size = "size";
       public static final String COL_flavour = " flavour";
       public static final String COL_weight = "weight";
       public static final String COL_price = "price";
       public static final String COL_category = "category";

    public MyDbHelper(Context context) {
        super(context, DB_NAME, null, DB_VERSION);
    }
}
```

ภาพที่ 48 คลาสของฐานข้อมูล

4.3.5 ตัวแสดงผลข้อกำหนดผลิตภัณฑ์

การพัฒนาตัวแสดงผลข้อกำหนดผลิตภัณฑ์ หลักสำคัญ คือ การนำผลลัพธ์ที่ได้จากการสืบค้นข้อกำหนดผลิตภัณฑ์มาแสดงบนหน้าจอสมาร์ตโฟน พร้อมทั้งเรียกใช้ฟังก์ชัน Text to speech ของ VAJA เพื่ออ่านข้อกำหนดผลิตภัณฑ์ให้ผู้ใช้งานทราบ ดังแสดงในภาพที่ 49

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

```
if(shwbarcode!= null){}

	TextView txt1 = (TextView)findViewById(R.id.textView1);
	TextView txt2 = (TextView)findViewById(R.id.textView2);
	TextView txt3 = (TextView)findViewById(R.id.textView3);
	TextView txt4 = (TextView)findViewById(R.id.textView4);
	TextView txt5 = (TextView)findViewById(R.id.textView5);
	TextView txt6 = (TextView)findViewById(R.id.textView6);
	TextView txt7 = (TextView)findViewById(R.id.textView7);
	TextView txt8 = (TextView)findViewById(R.id.textView8);

	txt1.setText(shwbarcode);
	txt2.setText(shwproductname);
	txt3.setText(shwbrand);
	txt4.setText(shwsize);
	txt5.setText(shwflavour);
	txt6.setText(shwweight);
	txt7.setText(shwprice);
	txt8.setText(shwcategory);
	txt7.setText(shwcategory);

checkTTSEngineInstalled(VAJA_TTS_ENGINE);
	if(tts != null)
		tts.speak("ชื่อผลิตภัณฑ์ "+txt2.getText().toString()+
				"ยี่ห้อ "+txt3.getText().toString()+
				"ขนาด "+txt4.getText().toString()+
				"รส "+txt5.getText().toString()+
				"น้ำหนัก "+txt6.getText().toString()+
				"ราคา "+txt7.getText().toString()+
				"ประเภท "+txt8.getText().toString()
,TextToSpeech.QUEUE_FLUSH, null);

else{
	TextView txt1 = (TextView)findViewById(R.id.textView1);
	txt1.setText("ไม่พบข้อมูลข้อกำหนดผลิตภัณฑ์");
	checkTTSEngineInstalled(VAJA_TTS_ENGINE);
	if(tts != null)
		tts.speak(txt1.getText().toString(),
		TextToSpeech.QUEUE_FLUSH, null);
}
```

ภาพที่ 49 การแสดงผลลัพธ์ และการอ่านผลลัพธ์ให้ผู้ใช้งานทราบด้วยเสียง

บทที่ 5
การประเมินและการวัดผล

5.1 แนวทางการประเมินผลงานวิจัย

แนวทางการประเมินจะใช้การเปรียบเทียบระหว่างภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ และผลลัพธ์ที่ได้จากระบบ โดยภาพนำเข้าของผลิตภัณฑ์จะแบ่งออกเป็น 3 ประเภท คือ ภาพนำเข้าของผลิตภัณฑ์ที่มีบาร์โค้ดครบถ้วน ภาพนำเข้าของผลิตภัณฑ์ที่มีบาร์โค้ดบางส่วน และภาพนำเข้าของผลิตภัณฑ์ที่ไม่มีบาร์โค้ด โดยแนวทางการประเมินประกอบด้วยรายละเอียด ดังต่อไปนี้

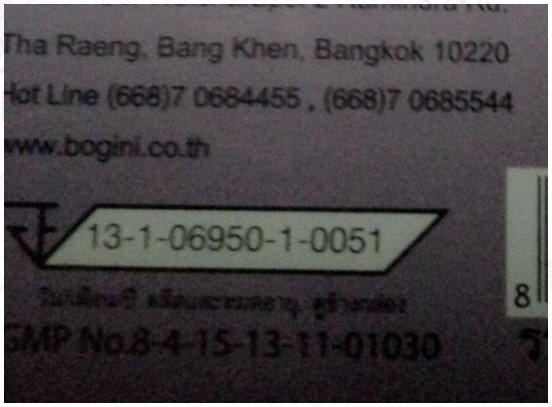
1. ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์
2. ผลลัพธ์ที่ได้จากระบบ

5.2 ผลการเปรียบเทียบตัวอย่างที่ 1

ตัวอย่างที่ 1 เป็นตัวอย่างของภาพนำเข้าของผลิตภัณฑ์ที่ไม่มีภาพบาร์โค้ด เกิดจากผู้ใช้งานไม่ทราบตำแหน่งบาร์โค้ดบนผลิตภัณฑ์ จึงทำให้มีการเลื่อนตำแหน่งกล้องไปถ่ายส่วนของผลิตภัณฑ์ที่ไม่มีบาร์โค้ด

5.2.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์

ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ตัวอย่างที่ 1 ดังภาพที่ 50 ซึ่งไม่มีบาร์โค้ดปรากฏบนภาพผลิตภัณฑ์



ภาพที่ 50 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 1

5.2.2 ผลลัพธ์ที่ได้จากระบบ

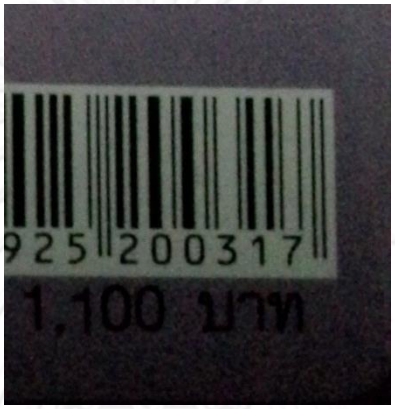
ผลลัพธ์ที่ได้จากระบบจะเป็นการแจ้งเตือนที่อยู่ในส่วนของตัวตรวจสอบภาพบาร์โค้ด ซึ่งจะเป็นการแจ้งเตือนด้วยเสียง โดยมีข้อความว่า *“ไม่พบบาร์โค้ด กรุณาถ่ายภาพใหม่อีกครั้ง”* และระบบจะนำกลับไปสู่กระบวนการถ่ายภาพใหม่

5.3 ผลการเปรียบเทียบตัวอย่างที่ 2

ตัวอย่างที่ 2 เป็นตัวอย่างของภาพนำเข้าของผลิตภัณฑ์ที่มีภาพบาร์โค้ดบางส่วน โดยบาร์โค้ดขาดหายไปทางด้านซ้าย เกิดจากผู้ใช้งานไม่ทราบตำแหน่งบาร์โค้ดบนผลิตภัณฑ์ จึงทำให้มีการเลื่อนตำแหน่งกล้องไปถ่ายส่วนของผลิตภัณฑ์ที่มีบาร์โค้ดไม่ครบ

5.3.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์

ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ตัวอย่างที่ 2 ดังภาพที่ 51 ซึ่งมีบาร์โค้ดปรากฏบนภาพผลิตภัณฑ์ แต่มีบาร์โค้ดบางส่วนขาดหายไปทางด้านซ้าย



ภาพที่ 51 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 2

5.3.2 ผลลัพธ์ที่ได้จากระบบ

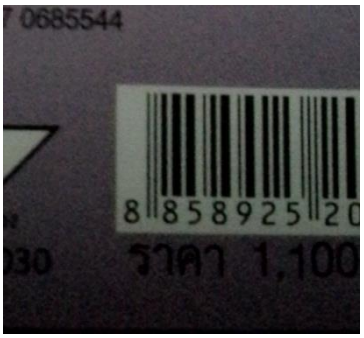
ผลลัพธ์ที่ได้จากระบบจะเป็นการแจ้งเตือนที่อยู่ในส่วนของตัวตรวจสอบภาพบาร์โค้ด ซึ่งจะเป็นการแจ้งเตือนด้วยเสียง โดยมีข้อความว่า *“บาร์โค้ดตกขอบซ้าย กรุณาเลื่อนกล้องไปทางซ้ายและถ่ายภาพใหม่อีกครั้ง”* และระบบจะนำกลับไปสู่กระบวนการถ่ายภาพใหม่

5.4 ผลการเปรียบเทียบตัวอย่างที่ 3

ตัวอย่างที่ 3 เป็นตัวอย่างของภาพนำเข้าของผลิตภัณฑ์ที่มีภาพบาร์โค้ดบางส่วน โดยบาร์โค้ดขาดหายไปทางด้านขวา เกิดจากผู้ใช้งานไม่ทราบตำแหน่งบาร์โค้ดบนผลิตภัณฑ์ จึงทำให้มีการเลื่อนตำแหน่งกล้องไปถ่ายส่วนของผลิตภัณฑ์ที่มีบาร์โค้ดไม่ครบ

5.4.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์

ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ตัวอย่างที่ 3 ดังภาพที่ 52 ซึ่งมีบาร์โค้ดปรากฏบนภาพผลิตภัณฑ์ แต่มีบาร์โค้ดบางส่วนขาดหายไปทางด้านขวา



ภาพที่ 52 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 3

5.4.2 ผลลัพธ์ที่ได้จากระบบ

ผลลัพธ์ที่ได้จากระบบจะเป็นการแจ้งเตือนที่อยู่ในส่วนของตัวตรวจสอบภาพบาร์โค้ด ซึ่งจะเป็นการแจ้งเตือนด้วยเสียง โดยมีข้อความว่า *"บาร์โค้ดตกขอบขวา กรุณาเลื่อนกล้องไปทางขวาและถ่ายภาพใหม่อีกครั้ง"* และระบบจะนำกลับไปสู่กระบวนการถ่ายภาพใหม่

5.5 ผลการเปรียบเทียบตัวอย่างที่ 4

ตัวอย่างที่ 4 เป็นตัวอย่างของภาพนำเข้าของผลิตภัณฑ์ที่มีภาพบาร์โค้ดบางส่วน โดยบาร์โค้ดขาดหายไปทางด้านบน เกิดจากผู้ใช้งานไม่ทราบตำแหน่งบาร์โค้ดบนผลิตภัณฑ์ จึงทำให้มีการเลื่อนตำแหน่งกล้องไปถ่ายส่วนของผลิตภัณฑ์ที่มีบาร์โค้ดไม่พอที่จะสแกนได้

5.5.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์

ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ตัวอย่างที่ 4 ดังภาพที่ 53 ซึ่งมีบาร์โค้ดปรากฏบนภาพผลิตภัณฑ์ แต่มีบาร์โค้ดบาร์โค้ดไม่พอที่จะสแกนได้ เนื่องจากความยาวของเส้นบาร์โค้ดมีความยาวสั้นเกินไป โดยเส้นของบาร์โค้ดขาดหายไปบางส่วนทางด้านบน

ภาพที่ 53 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 4

5.5.2 ผลลัพธ์ที่ได้จากระบบ

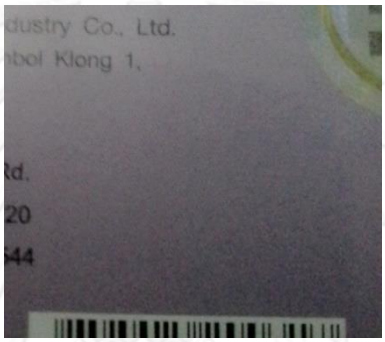
ผลลัพธ์ที่ได้จากระบบจะเป็นการแจ้งเตือนที่อยู่ในส่วนของตัวตรวจสอบภาพบาร์โค้ด ซึ่งจะเป็นการแจ้งเตือนด้วยเสียง โดยมีข้อความว่า *"บาร์โค้ดตกขอบบน กรุณาเลื่อนกล้องไปด้านบนและถ่ายภาพใหม่อีกครั้ง"* และระบบจะนำกลับไปสู่กระบวนการถ่ายภาพใหม่

5.6 ผลการเปรียบเทียบตัวอย่างที่ 5

ตัวอย่างที่ 5 เป็นตัวอย่างของภาพนำเข้าของผลิตภัณฑ์ที่มีภาพบาร์โค้ดบางส่วน โดยบาร์โค้ดขาดหายไปทางด้านล่าง เกิดจากผู้ใช้งานไม่ทราบตำแหน่งบาร์โค้ดบนผลิตภัณฑ์ จึงทำให้มีการเลื่อนตำแหน่งกล้องไปถ่ายส่วนของผลิตภัณฑ์ที่มีบาร์โค้ดไม่พอที่จะสแกนได้

5.6.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์

ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ตัวอย่างที่ 5 ดังภาพที่ 54 ซึ่งมีบาร์โค้ดปรากฏบนภาพผลิตภัณฑ์ แต่มีบาร์โค้ดบาร์โค้ดไม่พอที่จะสแกนได้ เนื่องจากความยาวของเส้นบาร์โค้ดมีความยาวสั้นเกินไป โดยเส้นของบาร์โค้ดขาดหายไปบางส่วนทางด้านล่าง



ภาพที่ 54 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 5

5.6.2 ผลลัพธ์ที่ได้จากระบบ

ผลลัพธ์ที่ได้จากระบบจะเป็นการแจ้งเตือนที่อยู่ในส่วนของตัวตรวจสอบภาพบาร์โค้ด ซึ่งจะเป็นการแจ้งเตือนด้วยเสียง โดยมีข้อความว่า *“บาร์โค้ดตกขอบล่าง กรุณาเลื่อนกล้องไปด้านล่างและถ่ายภาพใหม่อีกครั้ง”* และระบบจะนำกลับไปสู่กระบวนการถ่ายภาพใหม่

5.7 ผลการเปรียบเทียบตัวอย่างที่ 6

ตัวอย่างที่ 6 เป็นตัวอย่างของภาพนำเข้าของผลิตภัณฑ์ที่มีภาพบาร์โค้ดครบสมบูรณ์ เกิดจากผู้ใช้งานสามารถเลื่อนตำแหน่งกล้องไปถ่ายส่วนของผลิตภัณฑ์ที่มีบาร์โค้ดได้ครบสมบูรณ์ แต่ไม่มีข้อมูลอยู่ในฐานข้อมูลข้อกำหนดผลิตภัณฑ์

5.7.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์

ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ตัวอย่างที่ 6 ดังภาพที่ 55 ซึ่งมีบาร์โค้ดปรากฏบนภาพผลิตภัณฑ์ครบสมบูรณ์

ภาพที่ 55 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 6

5.7.2 ผลลัพธ์ที่ได้จากระบบ

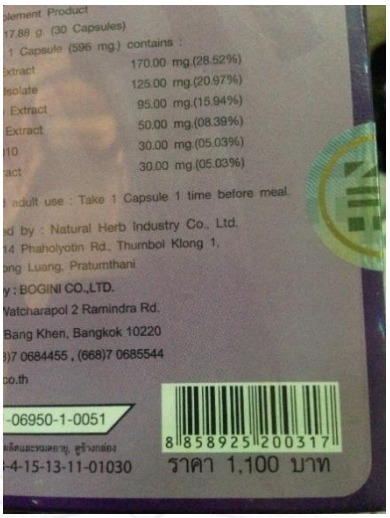
ผลลัพธ์ที่ได้จากระบบจะเป็นการแจ้งเตือนที่อยู่ในส่วนของตัวแสดงผลข้อกำหนดผลิตภัณฑ์ กรณีที่ค้นหาข้อมูลข้อกำหนดผลิตภัณฑ์ในฐานข้อมูลไม่พบ ซึ่งจะเป็นการแจ้งเตือนด้วยเสียง โดยมีข้อความว่า *“ไม่พบข้อมูลข้อกำหนดผลิตภัณฑ์ กรุณาถ่ายภาพใหม่อีกครั้ง”* และระบบจะนำกลับไปสู่กระบวนการถ่ายภาพใหม่

5.8 ผลการเปรียบเทียบตัวอย่างที่ 7

ตัวอย่างที่ 7 เป็นตัวอย่างของภาพนำเข้าของผลิตภัณฑ์ที่มีภาพบาร์โค้ดครบสมบูรณ์ในแนวตั้ง เกิดจากผู้ใช้งานสามารถเลื่อนตำแหน่งกล้องไปถ่ายส่วนของผลิตภัณฑ์ที่มีบาร์โค้ดได้ครบสมบูรณ์ และมีข้อมูลอยู่ในฐานข้อมูลข้อกำหนดผลิตภัณฑ์

5.8.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์

ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ตัวอย่างที่ 7 ดังภาพที่ 56 ซึ่งมีบาร์โค้ดปรากฏบนภาพผลิตภัณฑ์ครบสมบูรณ์ และภาพบาร์โค้ดอยู่ในแนวตั้ง



ภาพที่ 56 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 7

5.8.2 ผลลัพธ์ที่ได้จากระบบ

ผลลัพธ์ที่ได้จากระบบจะเป็นการแสดงผลที่อยู่ในส่วนของตัวแสดงผลข้อกำหนดผลิตภัณฑ์ กรณีที่ค้นหาข้อมูลข้อกำหนดผลิตภัณฑ์ในฐานข้อมูลพบ ซึ่งจะแสดงดังภาพที่ 57

ภาพที่ 57 ภาพผลลัพธ์การแสดงข้อกำหนดผลิตภัณฑ์ตัวอย่างที่ 7

โดยระบบจะมีการอ่านข้อกำหนดให้ผู้ใช้งานทราบมีข้อความว่า *“ชื่อผลิตภัณฑ์ อาหารเสริม ยี่ห้อ ออไรท์ ขนาดสามสิบแคปซูล รส น้ำหนัก สิบเจ็ดจุดแปดแปดกรัม ราคา หนึ่งพันหนึ่งร้อยบาท ประเภท เสริมอาหาร”* หลังจากอ่านข้อกำหนดผลิตภัณฑ์ให้ผู้ใช้งานทราบด้วยเสียงแล้ว ระบบจะนำกลับไปสู่กระบวนการถ่ายภาพใหม่

5.9 ผลการเปรียบเทียบตัวอย่างที่ 8

ตัวอย่างที่ 8 เป็นตัวอย่างของภาพนำเข้าของผลิตภัณฑ์ที่มีภาพบาร์โค้ดครบสมบูรณ์ในแนวนอน เกิดจากผู้ใช้งานสามารถเลื่อนตำแหน่งกล้องไปถ่ายส่วนของผลิตภัณฑ์ที่มีบาร์โค้ดได้ครบสมบูรณ์ และมีข้อมูลอยู่ในฐานข้อมูลข้อกำหนดผลิตภัณฑ์

5.9.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์

ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ตัวอย่างที่ 8 ดังภาพที่ 58 ซึ่งมีบาร์โค้ดปรากฏบนภาพผลิตภัณฑ์ครบสมบูรณ์ และภาพบาร์โค้ดอยู่ในแนวนอน

ภาพที่ 58 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 8

5.9.2 ผลลัพธ์ที่ได้จากระบบ

ผลลัพธ์ที่ได้จากระบบจะเป็นการแสดงผลที่อยู่ในส่วนของตัวแสดงผลข้อกำหนดผลิตภัณฑ์กรณีที่ค้นหาข้อมูลข้อกำหนดผลิตภัณฑ์ในฐานข้อมูลพบ ซึ่งจะแสดงดังภาพที่ 59

ภาพที่ 59 ภาพผลลัพธ์การแสดงข้อกำหนดผลิตภัณฑ์ตัวอย่างที่ 8

โดยระบบจะมีการอ่านข้อกำหนดให้ผู้ใช้งานทราบมีข้อความว่า *“ชื่อผลิตภัณฑ์ แฮร์เจล ยี่ห้อ เอ็กซิท ขนาด รส น้ำหนัก หนึ่งร้อยกรัม ราคา ห้าร้อยห้าสิบบาท ประเภท ผลิตภัณฑ์ดูแลเส้นผม”* หลังจากอ่านข้อกำหนดผลิตภัณฑ์ให้ผู้ใช้งานทราบด้วยเสียงแล้ว ระบบจะนำกลับไปสู่กระบวนการถ่ายภาพใหม่

5.10 ผลการเปรียบเทียบตัวอย่างที่ 9

ตัวอย่างที่ 9 เป็นตัวอย่างของภาพนำเข้าของผลิตภัณฑ์ที่มีภาพบาร์โค้ดครบสมบูรณ์ในแนวนอน เกิดจากผู้ใช้งานสามารถเลื่อนตำแหน่งกล้องไปถ่ายส่วนของผลิตภัณฑ์ที่มีบาร์โค้ดได้ครบสมบูรณ์ และมีข้อมูลอยู่ในฐานข้อมูลข้อกำหนดผลิตภัณฑ์

5.10.1 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์

ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคที่ต้องการสืบค้นข้อมูลข้อกำหนดผลิตภัณฑ์ตัวอย่างที่ 9 ดังภาพที่ 60 ซึ่งมีบาร์โค้ดปรากฏบนภาพผลิตภัณฑ์ครบสมบูรณ์ และภาพบาร์โค้ดอยู่ในแนวนอน

ภาพที่ 60 ภาพนำเข้าของผลิตภัณฑ์อุปโภคบริโภคตัวอย่างที่ 9

5.10.2 ผลลัพธ์ที่ได้จากระบบ

ผลลัพธ์ที่ได้จากระบบจะเป็นการแสดงผลที่อยู่ในส่วนของตัวแสดงผลข้อกำหนดผลิตภัณฑ์ กรณีที่ค้นหาข้อมูลข้อกำหนดผลิตภัณฑ์ในฐานข้อมูลพบ ซึ่งจะแสดงดังภาพที่ 61

ภาพที่ 61 ภาพผลลัพธ์การแสดงข้อกำหนดผลิตภัณฑ์ตัวอย่างที่ 9

โดยระบบจะมีการอ่านข้อกำหนดให้ผู้ใช้งานทราบมีข้อความว่า *“ชื่อผลิตภัณฑ์ ป๊อกกี้ ยี่ห้อ กูลิโกะ ขนาด รส ช็อกโกแลต น้ำหนัก สี่สิบเจ็ดกรัม ราคา สิบห้าบาท ประเภท ขนม”* หลังจากอ่านข้อกำหนดผลิตภัณฑ์ให้ผู้ใช้งานทราบด้วยเสียงแล้ว ระบบจะนำกลับไปสู่กระบวนการถ่ายภาพใหม่

ความต้องการระบบสามารถสรุปเป็นกรณีทดสอบต่างๆได้ทั้งหมด 7 กรณี และรายละเอียดการแจ้งเตือนของระบบในแต่ละกรณี ดังแสดงในตารางที่ 1

ตารางที่ 1 รายละเอียดกรณีทดสอบของระบบ

ลำดับ	กรณีทดสอบ	การแจ้งเตือนของระบบ	อุปกรณ์ที่ใช้ทดสอบ
1.	ไม่มีบาร์โค้ดบนภาพผลิตภัณฑ์	ไม่พบบาร์โค้ด กรุณาถ่ายภาพใหม่อีกครั้ง	Samsung Galaxy Note 8.0
2.	บาร์โค้ดขาดหายไปทางด้านซ้าย	บาร์โค้ดตกขอบซ้าย กรุณาเลื่อนกล้องไปทางซ้ายและถ่ายภาพใหม่อีกครั้ง	Samsung Galaxy Note 8.0
3.	บาร์โค้ดขาดหายไปทางด้านขวา	บาร์โค้ดตกขอบขวา กรุณาเลื่อนกล้องไปทางขวาและถ่ายภาพใหม่อีกครั้ง	Samsung Galaxy Note 8.0
4.	บาร์โค้ดขาดหายไปทางด้านบน	บาร์โค้ดตกขอบบน กรุณาเลื่อนกล้องไปด้านบนและถ่ายภาพใหม่อีกครั้ง	Samsung Galaxy Note 8.0
5.	บาร์โค้ดขาดหายไปทางด้านล่าง	บาร์โค้ดตกขอบล่าง กรุณาเลื่อนกล้องไปด้านล่างและถ่ายภาพใหม่อีกครั้ง	Samsung Galaxy Note 8.0
6.	บาร์โค้ดครบถ้วน แต่ไม่มีข้อมูลในฐานข้อมูลข้อกำหนดผลิตภัณฑ์	ไม่พบข้อมูลข้อกำหนดผลิตภัณฑ์ กรุณาถ่ายภาพใหม่อีกครั้ง	Samsung Galaxy Note 8.0
7.	บาร์โค้ดครบถ้วน และมีข้อมูลในฐานข้อมูลข้อกำหนดผลิตภัณฑ์	อ่านข้อมูลข้อกำหนดผลิตภัณฑ์ที่สืบค้นได้จากฐานข้อมูลให้ผู้ใช้งานทราบ	Samsung Galaxy Note 8.0

อุปกรณ์ที่ใช้ทดสอบระบบคือ Samsung Galaxy Note 8.0 ซึ่งประกอบไปด้วย

1. หน่วยประมวลผล ควอดคอร์ 1.6 กิกะเฮิร์ต (CPU Quad-Core 1.6GHz)
2. หน่วยความจำ 2 กิกะไบต์ (2 GB RAM)
3. ความจุ 16 กิกะไบต์ (16 GB)
4. ระบบปฏิบัติการแอนดรอยด์ 4.2.2

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 6
สรุปผลการวิจัย และข้อเสนอแนะ

6.1 สรุปผลการวิจัย

งานวิจัยนี้ได้นำเสนอวิธีการ และพัฒนาแอปพลิเคชันบนมือถือสำหรับตรวจจับหาตำแหน่งบาร์โค้ดบนผลิตภัณฑ์สินค้าอุปโภคบริโภค และอ่านข้อกำหนดผลิตภัณฑ์ให้ผู้ใช้งานทราบด้วยเสียง เพื่ออำนวยความสะดวกแก่ผู้พิการทางสายตาให้ทราบถึงข้อมูลของผลิตภัณฑ์

จากการประเมินการทดลองใช้งานแอปพลิเคชันบนมือถือสำหรับตรวจจับหาตำแหน่งบาร์โค้ดบนผลิตภัณฑ์สินค้าอุปโภคบริโภคในบทที่ 5 แสดงให้เห็นว่าแอปพลิเคชันสามารถตรวจจับหาตำแหน่งบาร์โค้ดบนผลิตภัณฑ์สินค้าอุปโภคบริโภคได้ถูกต้อง และอ่านข้อกำหนดผลิตภัณฑ์ให้ผู้ใช้งานทราบด้วยเสียงได้ถูกต้อง

6.2 ข้อจำกัด

1. งานวิจัยนี้ยังไม่สามารถรองรับระบบปฏิบัติการอื่นนอกเหนือจาก ระบบปฏิบัติการแอนดรอยด์ได้
2. งานวิจัยนี้ยังไม่รองรับการค้นหาข้อกำหนดผลิตภัณฑ์จากฐานข้อมูลข้อกำหนดผลิตภัณฑ์สากล เป็นเพียงฐานข้อมูลที่สร้างขึ้น และเก็บรวบรวมข้อมูลข้อกำหนดผลิตภัณฑ์เอง
3. งานวิจัยนี้ การตรวจสอบหาตำแหน่งบาร์โค้ดที่ขาดหายไปทางด้านบนและด้านล่าง ยังมีประสิทธิภาพต่ำอยู่ เนื่องจากขั้นตอนในการตรวจสอบภาพบาร์โค้ด อัลกอริทึมที่ใช้ในการเลือกส่วนของภาพมีข้อบกพร่อง

6.3 แนวทางการวิจัยต่อ

1. เพิ่มความสามารถของแอปพลิเคชันให้สามารถรองรับได้หลายระบบปฏิบัติการ เช่น ระบบปฏิบัติการ IOS และระบบปฏิบัติการ Windows Mobile
2. เพิ่มความสามารถของแอปพลิเคชันในการสืบค้นข้อมูลจากฐานข้อมูลข้อกำหนดผลิตภัณฑ์ที่สร้างขึ้นเอง ให้สามารถสืบค้นจากฐานข้อมูลข้อกำหนดผลิตภัณฑ์สากล เพื่อรองรับข้อมูลข้อกำหนดผลิตภัณฑ์ทั้งหมดที่มีอยู่ในปัจจุบัน
3. เพิ่มประสิทธิภาพในการตรวจสอบหาตำแหน่งของบาร์โค้ดที่ขาดหายไปทางด้านบนและด้านล่าง โดยเลือกใช้อัลกอริทึมที่มีประสิทธิภาพในการเลือกส่วนของภาพที่มีตำแหน่งบาร์โค้ดไม่ครบถ้วน

รายการอ้างอิง

1. *Barcode*. 2014; Available from: https://en.wikipedia.org/wiki/Barcode.
2. *Universal Product Code*. 2014; Available from: http://en.wikipedia.org/wiki/Universal_Product_Code.
3. *International Article Number (EAN)*. 2014; Available from: http://en.wikipedia.org/wiki/EAN13#Binary_encoding_of_data_digits_into_EAN-13_barcode.
4. *GS1 Thailand*. 2014; Available from: http://gs1thailand.org/.
5. *Centre For Indian Language Technologies Solution, IIT-Bombay*. 2014; Available from: http://www.cfilt.iitb.ac.in/westernmeetjune12/.
6. Tekin, E. and J.M. Coughlan. *An algorithm enabling blind users to find and read barcodes*. in *Applications of Computer Vision (WACV), 2009 Workshop on*. 2009.
7. Chunhui, Z., et al. *Automatic Real-Time Barcode Localization in Complex Scenes*. in *Image Processing, 2006 IEEE International Conference on*. 2006.
8. *Official ZXing ("Zebra Crossing") project home*. 2014; Available from: https://github.com/zxing/zxing.
9. *Welcome to "Vaja"*. 2014; Available from: http://vaja.nectec.or.th/.
10. *SQLite Home Page*. 2014; Available from: http://www.sqlite.org/.
11. *Adaptive Threshold*. 2014; Available from: http://docs.opencv.org/java/org/opencv/imgproc/Imgproc.html#adaptiveThreshold(org.opencv.core.Mat, org.opencv.core.Mat, double, int, int, int, double).
12. *Eroding*. 2014; Available from: http://docs.opencv.org/java/org/opencv/imgproc/Imgproc.html#erode(org.opencv.core.Mat, org.opencv.core.Mat, org.opencv.core.Mat).
13. *Dilating*. 2014; Available from: http://docs.opencv.org/java/org/opencv/imgproc/Imgproc.html#dilate(org.opencv.core.Mat, org.opencv.core.Mat, org.opencv.core.Mat).
14. *Finding contours*. 2014; Available from: http://docs.opencv.org/java/org/opencv/imgproc/Imgproc.html#findContours(org.opencv.core.Mat, java.util.List, org.opencv.core.Mat, int, int).
15. *Hough Line Transform*. 2014; Available from: http://docs.opencv.org/java/org/opencv/imgproc/Imgproc.html#HoughLinesP(org.opencv.core.Mat, org.opencv.core.Mat, double, double, int, double, double).

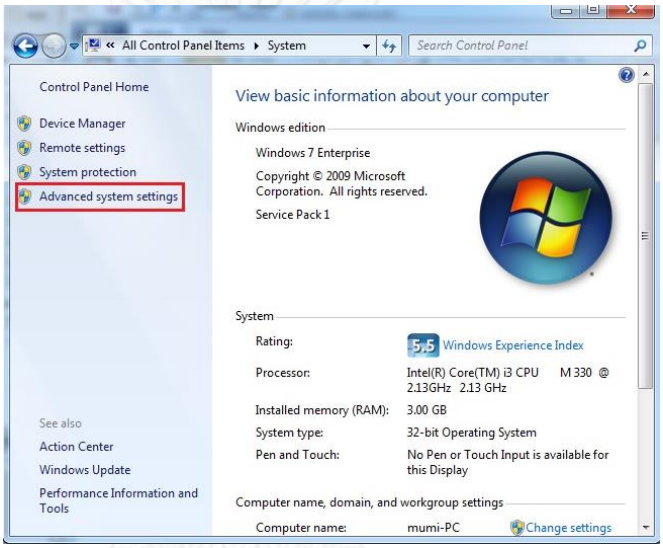
ภาคผนวก

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาคผนวก ก
การปรับตั้งค่าไดเรกทอรี เพื่อรองรับการเขียนโปรแกรมแอนดรอยด์

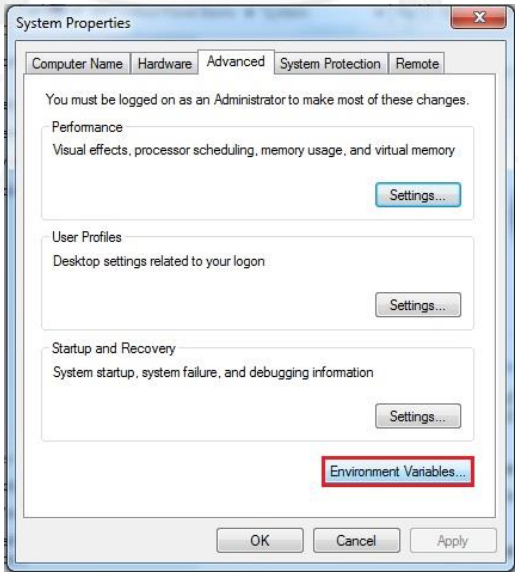
การปรับตั้งค่าไดเรกทอรี ซึ่งมีความจำเป็นในการเขียนโปรแกรมแอนดรอยด์ เพื่อให้สามารถเรียกใช้คอมไพเลอร์ และจาวา รันไทม์ได้ โดยมีขั้นตอนการปรับตั้งค่าดังต่อไปนี้

1. ไป Control Panel แล้วเลือก System จากนั้นให้เลือก Advanced system settings ดังภาพ



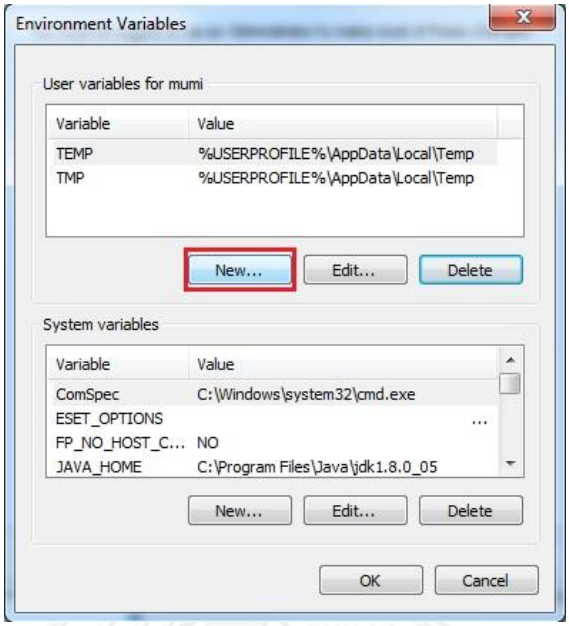
ภาพที่ 62 การเข้าสู่เมนูเมนูตั้งค่าระบบ

2. เลือกแท็บ Advanced และคลิกปุ่ม Environment Variables ดังภาพ



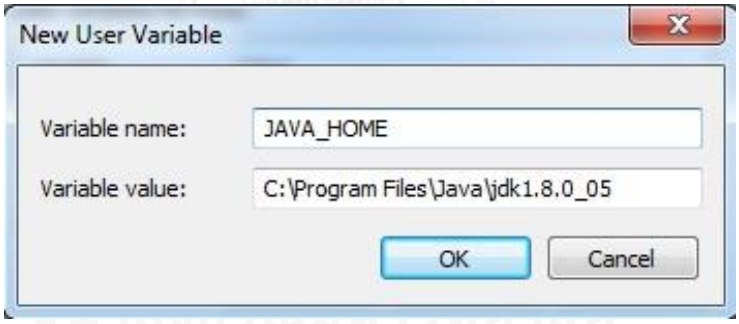
ภาพที่ 63 การเข้าสู่เมนูตั้งค่าไดเรกทอรีของตัวแปรที่ต้องการ

3. คลิกที่ปุ่ม New... ดังภาพ เพื่อตั้งค่าไดเรกทอรีของตัวแปรที่ต้องการ



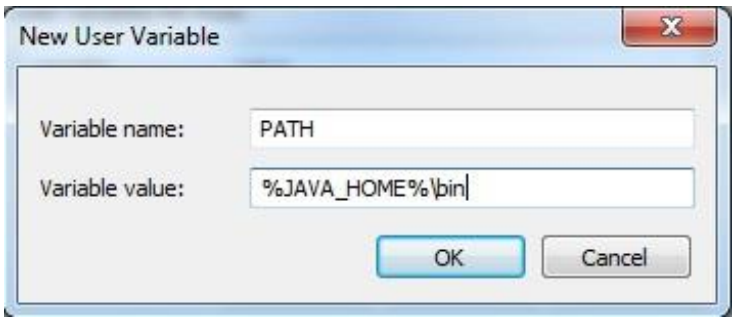
ภาพที่ 64 การเพิ่มไดเรกทอรีของตัวแปรที่ต้องการ

4. ระบุตัวแปร JAVA_HOME และไดเรกทอรีของ JAVA ในเครื่อง ดังภาพ และคลิก OK เพื่อบันทึก



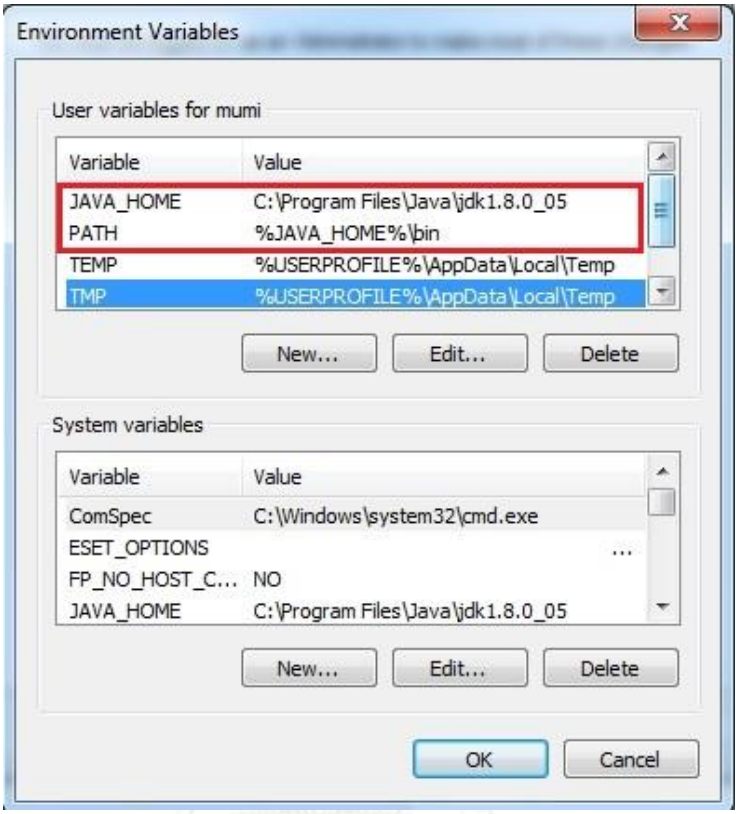
ภาพที่ 65 การระบุตัวแปร JAVA_HOME และไดเรกทอรี

5. ระบุตัวแปร PATH และกำหนดค่า Variable value ดังภาพ และคลิก OK เพื่อบันทึก



ภาพที่ 66 การระบุตัวแปร PATH และค่า Variable value

6. เมื่อปรับตั้งค่าไดเรกทอรีเสร็จ จะแสดงดังภาพที่ 67



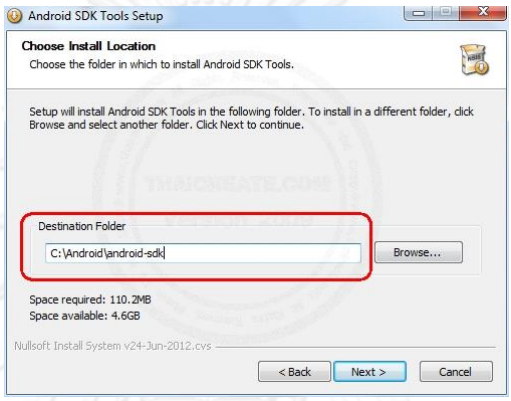
ภาพที่ 67 การปรับตั้งค่าไดเรกทอรีเสร็จสมบูรณ์

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ภาคผนวก ข
การติดตั้ง SDK เพื่อรองรับการเขียนโปรแกรมแอนดรอยด์

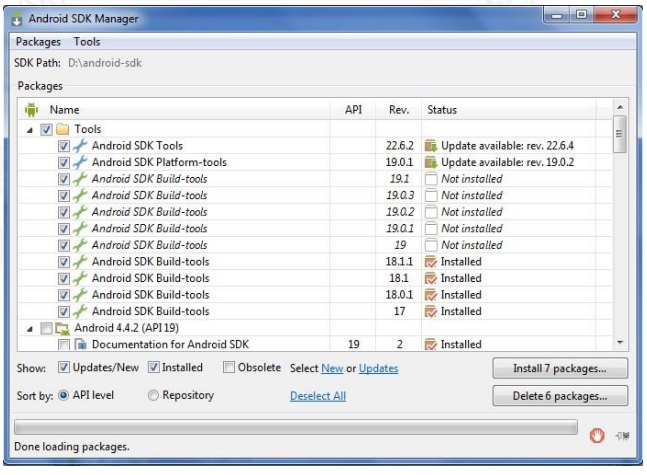
การติดตั้ง SDK (Software Development Kit) ซึ่งเป็นไลบรารีที่จำเป็นในการเขียนโปรแกรมบนแอนดรอยด์ โดยการติดตั้ง SDK มีขั้นตอนดังต่อไปนี้

1. ดาวน์โหลด SDK จาก http://developer.android.com/sdk/index.html
2. ดับเบิ้ลคลิกไฟล์ที่ดาวน์โหลดมา และคลิก Next จนถึงหน้าต่าง ดังภาพที่ 68 ให้ระบุ Destination Folder ดังภาพ และคลิก Next จนติดตั้งเสร็จสมบูรณ์



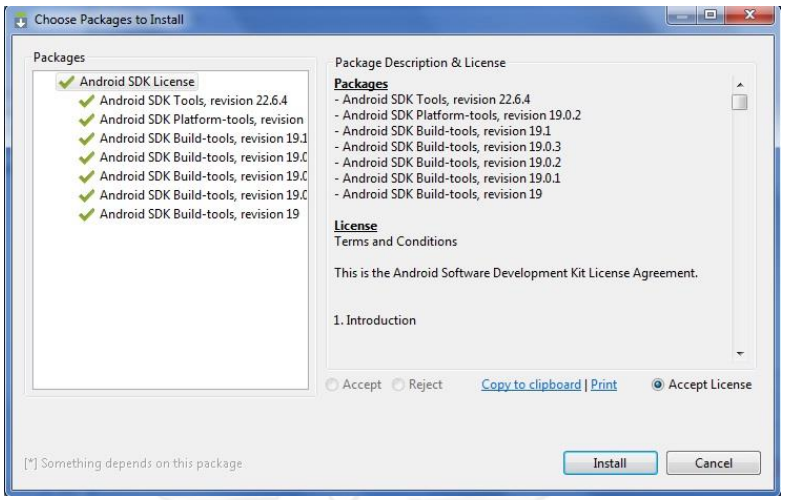
ภาพที่ 68 การเลือกซอฟต์แวร์ที่ต้องการเพื่อติดตั้ง

3. เมื่อโปรแกรมติดตั้งเสร็จสมบูรณ์ แล้วให้ติดตั้งแพ็คเกจ SDK ต่อไป
4. เลือกแอนดรอยด์ SDK เวอร์ชันที่ต้องการติดตั้ง และคลิกปุ่ม Install ดังภาพ



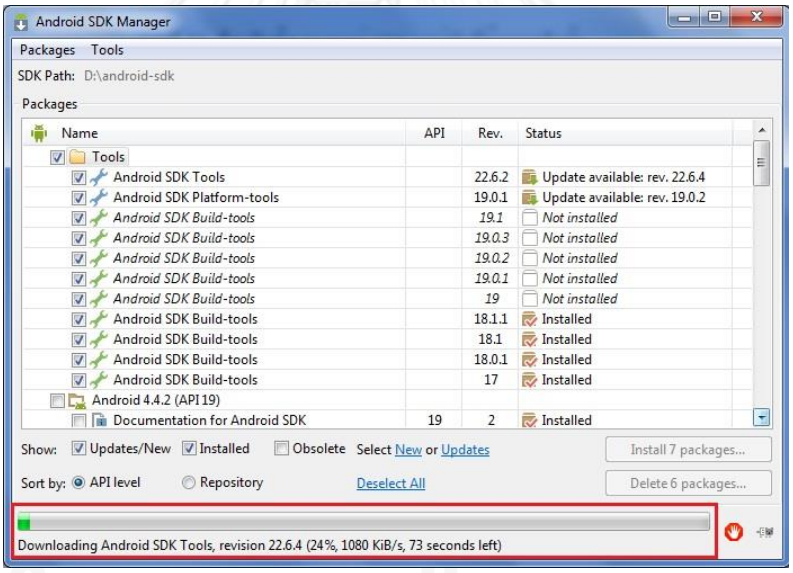
ภาพที่ 69 เลือกติดตั้งแอนดรอยด์ SDK เวอร์ชันที่ต้องการติดตั้ง

5. เลือกยอมรับเงื่อนไข และคลิกปุ่ม Install เพื่อดำเนินการติดตั้ง ดังภาพ



ภาพที่ 70 ยืนยันการติดตั้งแอนดรอยด์ SDK เวอร์ชันที่ต้องการติดตั้ง

6. แสดงสถานะการติดตั้งในภาพที่ 71 และรอจนติดตั้งเสร็จสมบูรณ์



ภาพที่ 71 สถานะการติดตั้ง

ประวัติผู้เขียนวิทยานิพนธ์

นางสาวนิตต์ชญาณ์ เจตจารุรัชต์ เกิดเมื่อวันที่ 8 กรกฎาคม พ.ศ. 2531 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาปริญญาตรีหลักสูตรวิศวกรรมศาสตรบัณฑิต (วศ.บ.) สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยมหิดล ในปีการศึกษา 2554 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิศวกรรมซอฟต์แวร์ ที่ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2555

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University