# ระบบเช็คชื่อเข้าเรียนโดยใช้การรู้จำใบหน้าบนสมาร์ทโฟน

# Class Attendance System on Smartphone using Face Recognition


**นายพลสิต สุขมา**

Ponlasit Sukma

**นายธนพล สังข์อ้น**

Thanaphon Sangon


**โครงงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตร์บัณฑิต**

**ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์**

**มหาวิทยาลัยศรีนครินทรวิโรฒ ปีการศึกษา พ.ศ. 2563**

# ระบบเช็คชื่อเข้าเรียนโดยใช้การรู้จำใบหน้าบนสมาร์ทโฟน

# Class Attendance System on Smartphone using Face Recognition


**นายพลสิต สุขมา**

Ponlasit Sukma

**นายธนพล สังข์อ้น**

Thanaphon Sangon


**โครงงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตร์บัณฑิต**

**ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์**

**มหาวิทยาลัยศรีนครินทรวิโรฒ ปีการศึกษา พ.ศ. 2563**

**คณะวิทยาศาสตร์มหาวิทยาลัยศรีนครินทรวิโรฒ**

**ชื่อหัวข้อโครงงาน** **ระบบเช็คชื่อเข้าเรียนโดยใช้การรู้จำใบหน้าบนสมาร์ทโฟน**

Class Attendance System on Smartphone

using Face Recognition

**นิสิต** **นายพลสิต สุขมา 60102010340**

**นายธนพล สังข์อ้น 60102010565**

**ปริญญา** **วิทยาศาสตร์บัณฑิต (วท.บ.)**

**ภาควิชา** **วิทยาการคอมพิวเตอร์**

**อาจารย์ที่ปรึกษาโครงงาน** **ผศ.ศศิวิมล สุขพัฒน์**

**ลงชื่อ.....**

**(ผศ. ศศิวิมล สุขพัฒน์)**

**อาจารย์ที่ปรึกษาโครงงาน**

# บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์ในการพัฒนาระบบเช็คชื่อเข้าเรียนด้วยแอปพลิเคชันบนสมาร์ท-โฟนด้วยการรู้จำใบหน้า การรู้จำใบหน้านั้นได้นำ FaceNet มาใช้ในกระบวนการคัดแยกคุณลักษณะเด่น (feature extraction) โดย FaceNet ทำการฝังคุณลักษณะเด่น (Embedding Features) จากบริเวณที่สำคัญของใบหน้า ได้แก่ ตา จมูก และ ปาก ให้เป็นข้อมูลเว็กเตอร์ 128 มิติ จากนั้นจะนำไปทำการเทรนโมเดลกับตัวจำแนก (classifier) 3 แบบ ได้แก่ K-Nearest Neighbors, Multi-layer Perceptron และ Support Vector Machine จากการทดลองพบว่า KNN สามารถจำแนกได้ดีที่สุด โดยจะได้ค่าความถูกต้องอยู่ที่ 98.7% และค่าเฉลี่ยของความน่าจะเป็นต่อใบหน้าอยู่ที่ 99.2% ในส่วนของการพัฒนาบนแอปพลิเคชันนั้น ได้ใช้ Flutter Framework ในการพัฒนาแอปพลิเคชัน โดยสามารถทำงานได้ทั้งบนระบบแอนดรอยด์และ iOS ผู้วิจัยได้นำโมเดลที่ได้ไปประยุกต์ใช้กับระบบเช็คชื่อเข้าเรียน โดยแบ่งการทำงานเป็น 2 ส่วน คือส่วนของแอปพลิเคชันเช็คชื่อเข้าเรียนบนสมาร์ทโฟน และส่วนของโมเดลรู้จำใบหน้าและฐานข้อมูลการเช็คชื่อเข้าเรียนบนเว็บเซิร์ฟเวอร์ ในส่วนของแอปพลิเคชันบนสมาร์ทโฟนจะส่งภาพใบหน้าผู้เรียนจากกล้องของสมาร์ทโฟนไปวิเคราะห์กับโมเดลบนเว็บเซิร์ฟเวอร์เพื่อทำการทำนายใบหน้าที่รับมาและจะส่งข้อความยืนยันผลการทำนายใบหน้า จากนั้นแอปพลิเคชันจะทำการบันทึกข้อมูลการเช็คชื่อเข้าเรียนของผู้ใช้ลงในฐานข้อมูล

# Abstract

This research aims to develop a system to check class attendance by using a mobile application with face recognition. Face recognition has applied FaceNet in the feature extraction process. FaceNet creates features from keys areas of the face, including eyes, nose, and mouth, into 128-dimensional vector data. To select a classifier in our model, we consider among three classifiers: K-Nearest Neighbors, Multi-layer Perceptron, and Support Vector Machine. We found that K-Nearest Neighbors was the best classifier with an accuracy of 98.7% and the mean of probability at 99.2%. Therefore, we applied K-Nearest Neighbors to our proposed model to the attendance check-in system. We also use the flutter framework for developing our mobile applications, which support running on both Android and iOS platforms. The attendance checking system is divided into two parts. The first part is the class attendance checking process which consists of the enrollment course of the student. The part runs on a smartphone. The second part is the face recognition process which uses our proposed face recognition model and the class attendance database, which runs on the webserver. After selecting a class to attend, the student has to take his/her picture and upload it to the face classifying model on the webserver. After that, our model will predict the received image and send a message confirming the face prediction result. Finally, the application will record the user's class attendance information into the database.

# กิตติกรรมประกาศ

โครงงานระบบเช็คชื่อเข้าเรียนโดยใช้การรู้จำใบหน้าบนสมาร์ทโฟน(Class Attendance System on Smartphone using Face Recognition) สามารถสำเร็จลุล่วงได้ด้วยความกรุณาจาก อาจารย์ ผศ.ศศิวิมล สุขพัฒน์ ซึ่งเป็นอาจารย์ที่ปรึกษาโครงงานนี้ท่านได้ให้คำแนะนำและข้อคิดเห็นต่างๆ อันเป็นประโยชน์อย่างยิ่งในการทำวิจัย ตลอดจนปรับปรุงแก้ไขข้อบกพร่องต่างๆที่เกิดขึ้นระหว่างการดำเนินการทำโครงงาน รวมทั้งให้กำลังใจแก่คณะผู้จัดทำเสมอมา คณะผู้จัดทำกราบขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณอาจารย์ทุกๆท่านในภาควิชาวิทยาการคอมพิวเตอร์ที่กรุณาให้ความรู้และสามารถนำความรู้ที่ได้เรียนมาประยุกต์ใช้ในการดำเนินโครงงานรวมทั้งให้คำปรึกษาและคำแนะนำที่เป็นประโยชน์ ทำให้คณะผู้จัดทำสามารถแก้ไขโครงงานได้สมบูรณ์มากยิ่งขึ้น ขอขอบพระคุณมหาวิทยาลัยศรีนครินทรวิโรฒ, คณะวิทยาศาสตร์และภาควิชาวิทยาการคอมพิวเตอร์ ที่เป็นแหล่งศึกษาหาความรู้ให้การสนับสนุนสถานที่และอุปกรณ์ต่างๆสำหรับการดำเนินโครงงานนอกจากนี้ยังรวมถึงประสบการณ์ในการทำกิจกรรมต่างๆ เพื่อนำไปประยุกต์ใช้ในการทำงานและการดำเนินชีวิตต่อไป

ขอขอบพระคุณบิดา มารดา รวมถึงครอบครัวที่ได้ให้การช่วยเหลือและสนับสนุนตลอดจนถึงความห่วงใยและยังเป็นกำลังใจสำคัญในการทำโครงงานเสมอมา

ขอขอบคุณเพื่อนๆ นิสิตหลักสูตรวิทยาศาสตรบัณฑิต ภาควิชาวิทยาการคอมพิวเตอร์และภาควิชาสถิติ มหาวิทยาลัยศรีนครินทรวิโรฒ ที่ยินดีให้ข้อมูลรูปภาพนำมาใช้ในการทำงานวิจัย และช่วยให้คำแนะนำอันเป็นประโยชน์และช่วยเหลือเกื้อกูลเพื่อให้โครงงานนี้ ลุล่วงไปได้ด้วยดี

สุดท้ายนี้ขอขอบคุณผู้มีส่วนเกี่ยวข้องทุกๆท่านที่ทำให้โครงงานนี้สำเร็จลุล่วงไปได้ด้วยดีและหวังว่าโครงงานระบบเช็คชื่อเข้าเรียนโดยใช้การรู้จำใบหน้าบนสมาร์ทโฟน(Class Attendance System on Smartphone using Face Recognition) จะเป็นประโยชน์ต่อผู้ที่สนใจศึกษาและสามารถเป็นส่วนหนึ่ง สำหรับผู้ที่สนใจจะนำไปต่อยอดในบริบทต่างๆต่อไป

**คณะผู้จัดทำ**

# สารบัญ

บทคัดย่อ.....ก

Abstract.....ข

กิตติกรรมประกาศ.....ค

สารบัญตาราง.....

สารบัญรูปภาพ.....

บทที่1 บทนำ.....1

1.1 ที่มาและความสำคัญ.....1
1.2 วัตถุประสงค์ของงานวิจัย.....1
1.3 ขอบเขตของการวิจัย.....2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....2

บทที่2 องค์ความรู้ที่เกี่ยวข้อง.....3

2.1 การรู้จำใบหน้า (Face recognition).....3
2.2 การตรวจจับใบหน้า (Face detection).....3
2.3 Image Pre-processing.....3
2.3.1 Face cropping.....3
2.3.2 Image resizing.....4
2.4 Deep learning .....5
2.4.1 Convolutional Neural Network.....5
2.4.2 FaceNet.....6
2.5 Classification .....7
2.5.1 Support Vector Machine(SVM).....7
2.5.2 Multilayer Perceptron (MLP).....9
2.5.3 K-Nearest Neighbors (KNN).....10

2.6.งานวิจัยที่เกี่ยวข้อง.....10
2.6.1 Attendance System based on Deep Learning Face Recognition without Queue.....10
2.6.2 Face and Lip-reading Authentication System Based on Android Smart Phones.....11
2.6.3 Real-Time Smart Attendance System using FaceRecognition Techniques.....12
บทที่3 วิธีการดำเนินงาน.....14
3.1 วิธีการดำเนินงาน.....14
3.2 แผนการดำเนินงานตลอดโครงงาน.....15
3.3 อุปกรณ์และเครื่องมือที่ใช้ในงานวิจัย.....16
3.3.1 ฮาร์ดแวร์ (Hardware).....16
3.3.2 ซอฟต์แวร์ (Software).....16
3.3.3 ภาษาที่ใช้ (Programming language).....16
3.3.4 Dataset.....16
3.4 แนวคิดการทำงานของระบบหลัก.....17
3.4.1 แผนภาพการทำงานของระบบ.....18
3.5 ขั้นตอนการ train classifier.....19
3.5.1 รวบรวมรูปภาพใบหน้า.....19
3.5.2 ครอปตัดภาพเพื่อเอาเฉพาะส่วนใบหน้า.....19
3.5.3 การทำ Feature Extraction.....19
3.5.4 การ train classifier.....20
บทที่4 สรุปผลการดำเนินงาน.....21
4.1 การนำโมเดลมาใช้ทำนายภาพ.....21
4.2 การพัฒนาส่วนแอปพลิเคชัน.....22

**บทที่5 สรุปผลและอภิปรายผล ข้อเสนอแนะ.....32**

**5.1 สรุปผลการดำเนินงาน.....32**

**5.2 ปัญหาและอุปสรรค.....33**

**5.3 ข้อเสนอแนะและแนวทางในการพัฒนา.....33**

**บรรณานุกรม.....34**

# สารบัญรูปภาพ

รูปภาพประกอบที่ 1 ภาพที่ได้ทำการ detect ส่วนที่เป็นใบหน้าแล้ว.....4

รูปภาพประกอบที่ 2 ภาพที่ได้รับการครอบตัดส่วนที่เป็นใบหน้า.....4

รูปาพประกอบที่ 3 เปรียบเทียบความแตกต่างของชั้น hidden layer ระหว่าง Artificial Neural Network และ Deep learning โดยทางซ้ายคือ Artificial Neural Network และทางขวาคือ Deep learning.....5

รูปภาพประกอบที่ 4 ประกอบการอธิบายคร่าวๆเกี่ยวกับ Convolutional Neural Network.....6

รูปภาพประกอบที่ 5 Convolutional Neural Network Architecture.....6

รูปภาพประกอบที่ 6 ตัวอย่างการใช้ embeddings ใบหน้าร่วมกับ triplet loss function.....7

รูปภาพประกอบที่ 7 ตัวอย่างรูป SVM.....7

รูปภาพประกอบที่ 8 ตัวอย่างการแบ่งข้อมูล.....8

รูปภาพประกอบที่ 9 ตัวอย่าง parameter C.....8

รูปภาพประกอบที่ 10 ตัวอย่างการทำ Kernels.....9

รูปภาพประกอบที่ 12 ตัวอย่าง MLP แบบ one hidden layer MLP.....9

รูปภาพประกอบที่ 13 สมการเพื่อวัดระยะทางของ KNN.....10

รูปภาพประกอบที่ 14 ตารางผลลัพธ์ของ algorithms 3 แบบ.....10

รูปภาพประกอบที่ 15 ตัวอย่างของ 10 eigenvectors แรกโดยวิธี LPP.....11

รูปภาพประกอบที่ 16 Activity diagram สำหรับ smart attendance system.....13

รูปภาพประกอบที่ 17 แผนภาพการทำงานของระบบ.....18

รูปภาพประกอบที่ 18 ตัวอย่างรูปภาพใบหน้าที่ทำการปรับขนาดและตรวจจับใบหน้า.....19

รูปภาพประกอบที่ 19 แสดงจำนวน parameter ของ FaceNet โดยมีจำนวน parameter ทั้งหมดที่ 22,808,144.....19

รูปภาพประกอบที่ 20 ข้อมูลภาพใบหน้าที่ถูก Embedding Features ให้เป็น 128-dimensional vector.....20

รูปภาพประกอบที่ 21 แผนภาพขั้นตอนการ train classifier.....20

รูปภาพประกอบที่ 22 หน้า user interface เข้าสู่ระบบ.....23

รูปภาพประกอบที่ 23 หน้า user interface ของนักเรียน(ซ้าย)และครูผู้สอน(ขวา) โดยมีปุ่มกดออกจากระบบ.....24

รูปภาพประกอบที่ 24 หน้า user interface แสดงรายวิชาของนักเรียน(ซ้าย)และแสดงรายวิชาของที่คุณครูสอน(ขวา).....25

รูปภาพประกอบที่ 25 หน้า user interface แสดงรายละเอียดการเข้าเช็คชื่อของนักเรียนในรายวิชา.....26

รูปภาพประกอบที่ 26 หน้า user interface แสดงรายวิชาของที่คุณครูสอน(ซ้าย)และหน้า user interface แสดงรายละเอียดการเข้าเช็คชื่อของนักเรียน.....27

รูปภาพประกอบที่ 27 หน้า user interface การตรวจจับใบหน้าของนักเรียนก่อนส่งภาพไปที่ web server.....28

รูปภาพประกอบที่ 28 หน้า user interface แสดงผลลัพธ์การทำนายใบหน้า.....29

# สารบัญตาราง

ตารางที่ 1 แผนการดำเนินโครงการ.....15

ตารางที่ 2 ตารางแสดงค่า Accuracy ของ train data และ test data ของแต่ละ classifier.....21

ตารางที่ 3 ตารางแสดงค่า Accuracy, Precision, Recall และ F1-score ของ test data ของแต่ละ classifier.....21

ตารางที่ 4 ตารางเปรียบเทียบค่าเฉลี่ย probability ของผลการทำนายแต่ละโมเดล.....22

ตารางที่ 5 ตารางแสดงผลการทดลองเช็คชื่อบนแอปพลิเคชัน.....30

ตารางที่ 6 ตารางแสดงผลการทดลองวัดประสิทธิภาพเวลาที่ใช้ในการทำนายใบหน้าบนเซิร์ฟเวอร์.....31

# บทที่ 1
# บทนำ

## 1.1 ที่มาและความสัญของโครงงาน

ในปัจจุบันการเช็คชื่อเข้าเรียนยังคงใช้การเรียกชื่อนักเรียนเพื่อเช็คชื่อ ซึ่งอาจทำให้เสียเวลาได้ ถ้าหากมีนักเรียนเป็นจำนวนมาก หรือการเช็คชื่อเข้าเรียนด้วยการเซ็นชื่อลงในกระดาษ โดยวิธีนี้ก็อาจทำให้มีการใช้ทรัพยากรกระดาษเพิ่มมากขึ้น และยังอาจขาดความแม่นยำในการระบุตัวตนของนักเรียนที่เซ็นชื่อ เนื่องจากอาจมีการเซ็นชื่อแทนเพื่อน

งานวิจัยที่เกี่ยวกับเทคโนโลยีในการนำภาพมาวิเคราะห์และประมวลผลนั้นมีมากมายหลายอย่าง และหนึ่งในนั้นก็คือ “Face recognition” โดย Face recognition หรือระบบรู้จำใบหน้านั้นเป็นกระบวนการนำภาพใบหน้าที่ทำการตรวจจับ มาวิเคราะห์และประมวลผลด้วยเทคโนโลยี AI ( Artificial Intelligence ) พร้อมเปรียบเทียบลักษณะใบหน้าจากฐานข้อมูล เพื่อตรวจสอบว่าภาพใบหน้าที่ตรวจจับนั้นตรงกับบุคคลใด โดยได้มีการนำเอาเทคโนโลยี Face recognition มาประยุกต์ใช้ในปัจจุบันมากมาย เช่น นำมาใช้ในด้านการรักษาความปลอดภัย นำมาใช้ในการบันทึกเวลาเข้างานของพนักงานในหน่วยงานหรือองค์กรต่างๆ เป็นต้น

ดังนั้นจึงได้เกิดแนวความคิดที่จะศึกษาเทคโนโลยี Face recognition เพื่อจะนำมาพัฒนาระบบการเช็คชื่อเข้าเรียนด้วยระบบรู้จำใบหน้าในรูปแบบแอปพลิเคชันบนมือถือระบบปฏิบัติการแอนดรอยด์ เพื่อแก้ปัญหาเรื่องความล่าช้าจากการเรียกเช็คชื่อ แก้ปัญหาการใช้ปริมาณกระดาษที่เพิ่มขึ้น และแก้ปัญหาเรื่องความผิดพลาดจากการยืนยันตัวตนของผู้เซ็นชื่อเข้าเรียนดังที่ได้กล่าวมา

## 1.2 วัตถุประสงค์ของโครงงาน

1.2.1 เพื่อศึกษาอัลกอริทึมในการทำ Face detection

1.2.2 เพื่อศึกษาหลักการและเทคนิคในด้าน Image Pre-processing

1.2.3 เพื่อศึกษาอัลกอริทึมในการรู้จำใบหน้าด้วย Deep learning และ Classification

1.2.4 เพื่อประยุกต์ใช้ระบบการรู้จำใบหน้าในการพัฒนาแอปพลิเคชันบนสมาร์ทโฟน

1.2.5 เพื่อพัฒนาระบบบันทึกการเข้าเรียนด้วยระบบการรู้จำใบหน้า

**1.3 ขอบเขตของโครงงาน**

1.3.1 พัฒนาระบบตรวจจับใบหน้าเพื่อทำการเช็คชื่อผ่านแอปพลิเคชันบนสมาร์ทโฟน โดยใช้ MTCNN (Multi-task Cascaded Convolutional Networks) ในการตรวจจับใบหน้า ใช้ Deep Learning และ Classification เป็นอัลกอริทึมในการรู้จำใบหน้า

1.3.2 พัฒนาแอปพลิเคชันเช็คชื่อเข้าเรียนบนสมาร์ทโฟน โดยแอปพลิเคชันสามารถเลือกวิชาที่ต้องการเช็คชื่อ จากนั้นทำการสแกนใบหน้าด้วยมือถือเพื่อเช็คชื่อ จากนั้นให้ทำการแสดงผลว่าใบหน้าได้รับการยืนยัน แล้วบันทึกลงฐานข้อมูลโดยมีรหัสวิชาที่เรียน Section ชื่อ รหัสนิสิต วันที่ จากนั้นให้แสดงผลผ่านทางหน้าแอปพลิเคชันเพื่อบอกว่าเช็คชื่อเข้าเรียนเรียบร้อย

**1.4 ประโยชน์ที่คาดว่าจะได้รับ**

1.4.1 สามารถลดเวลาจากการเรียกเช็คชื่อได้

1.4.2 สามารถแก้ปัญหาการเซ็นชื่อแทนกันในใบเซ็นชื่อได้และช่วยลดทรัพยากรกระดาษจากใบเซ็นชื่อได้

1.4.3 สามารถนำองค์ความรู้ต่างๆที่ได้รับจากงานวิจัย นำไปต่อยอดงานในด้านอื่นๆได้

# บทที่ 2
# องค์ความรู้ที่เกี่ยวข้อง

## 2.1 การรู้จำใบหน้า (Face recognition)

เป็นวิธีการระบุหรือยืนยันตัวตนของบุคคลโดยใช้ใบหน้า ระบบการรู้จำใบหน้าสามารถใช้ระบุตัวบุคคลในภาพถ่ายหรือวิดีโอได้แบบเรียลไทม์ โดยใช้อัลกอริทึมเพื่อเลือกรายละเอียดที่เฉพาะเจาะจงบริเวณใบหน้าของบุคคล เช่น ระยะห่างระหว่างดวงตาหรือรูปร่างของคาง จากนั้นจะถูกนำไปแปลงข้อมูลให้อยู่ในรูปแบบทางคณิตศาสตร์เพื่อประมวลผลและวิเคราะห์ และนำไปเปรียบเทียบกับใบหน้าอื่นๆที่รวบรวมไว้ในฐานข้อมูล โดยแต่ละใบหน้าก็จะมีข้อมูลต่างๆบนใบหน้าที่ต่างกัน เพื่อที่จะสามารถแยกแยะใบหน้าของแต่ละใบหน้าได้ [1]

## 2.2 การตรวจจับใบหน้า (Face detection)

Face detection เป็นส่วนสำคัญของการรู้จำใบหน้า โดยก่อนที่ระบบจะทำการจดจำใบหน้าได้นั้น ต้องทำการตรวจจับพื้นที่ส่วนใบหน้าก่อน เพื่อที่ระบบจะได้รู้ว่าจะทำการคัดแยกลักษณะบนใบหน้าจากส่วนไหนของภาพ

### 2.2.1 Face Detection using MTCNN (Multi-task Cascaded Convolutional Networks)

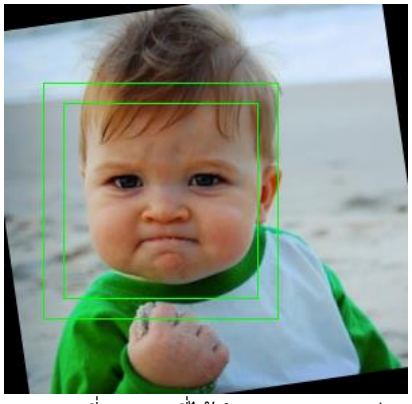
เป็นเฟรมเวิร์กที่พัฒนาขึ้นเพื่อเป็นโซลูชันสำหรับทั้งการตรวจจับใบหน้าและการจัดตำแหน่งใบหน้า กระบวนการนี้ประกอบด้วยสามขั้นตอนของ Convolutional Network ที่สามารถจดจำใบหน้าและจุดสำคัญบนใบหน้า เช่น ตาจมูกและปาก [2]

## 2.3 Image Pre-processing

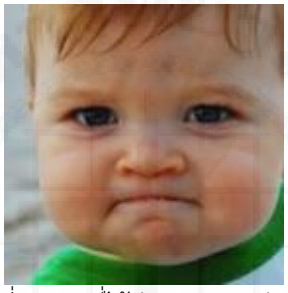
เป็นขั้นตอนที่ใช้เพื่อลดความซับซ้อนและเพิ่มความแม่นยำของอัลกอริทึมที่ใช้ เนื่องจากภาพที่ได้มานั้นบางทีอาจจะมีความคมชัดที่ต่างกัน รวมถึงขนาดและสีที่ต่างกัน เราไม่สามารถเขียนอัลกอริทึมเฉพาะสำหรับแต่ละเงื่อนไขที่ภาพมีต่างกันได้ ดังนั้นเราจึงต้องจัดการกับข้อมูลที่เป็นรูปภาพ ให้อยู่ในเงื่อนไขที่เหมือนกัน เพื่อให้อัลกอริทึมที่ใช้ในการทำนายนั้นมีความแม่นยำมากขึ้น โดย Image Pre-processing ก็มีหลากหลายวิธีการให้ใช้เพื่อให้เหมาะสมกับรูปภาพและโจทย์ปัญหาที่ต้องการแก้ [3]

### 2.3.1 Face cropping

การครอบตัดใบหน้าจะช่วยให้ neural network สามารถทำงานในการจำแนกใบหน้าได้ดียิ่งขึ้น โดยวิธีนี้จะช่วยกำจัดข้อมูลที่ไม่สำคัญในการจำแนกใบหน้า เช่น พื้นหลัง เสื้อผ้า หรืออุปกรณ์เสริมต่างๆ โดยวิธีการก็คือ ทำการ detect ส่วนที่เป็นใบหน้า จากนั้นทำการครอบตัดส่วนที่ detect ได้ว่าเป็นใบหน้า [4]



รูปภาพประกอบที่ 1 ภาพที่ได้ทำการ detect ส่วนที่เป็นใบหน้าแล้ว
(https://medium.com/yottabytes/a-quick-guide-on-preprocessing-facial-images-for-neural-networks-using-opencv-in-python-47ee3438abd4)



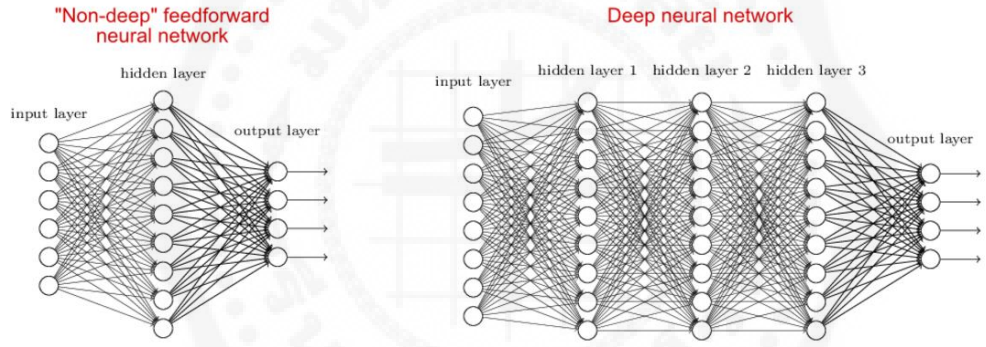
รูปภาพประกอบที่ 2 ภาพที่ได้รับการครอบตัดส่วนที่เป็นใบหน้า
(https://medium.com/yottabytes/a-quick-guide-on-preprocessing-facial-images-for-neural-networks-using-opencv-in-python-47ee3438abd4)

2.3.2 Image resizing

เนื่องจาก Neural networks ที่เราจะใช้ในการจำแนกใบหน้านั้น ต้องการภาพทั้งหมดที่มีรูปร่างและขนาดเท่ากัน เพื่อให้การประมวลผลสามารถทำได้เร็วขึ้น ดังนั้นเราจะทำการปรับขนาดของภาพทั้งหมดที่จะใช้ให้เท่ากันก่อนที่จะนำภาพไปทำการ train โมเดล [4]

2.4 Deep learning

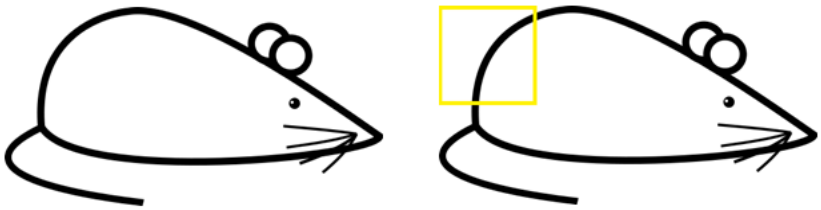
เป็นสาขาของ machine learning พื้นฐานของการเรียนรู้เชิงลึกคือ อัลกอริทึมที่พยายามจะสร้างแบบจำลองเพื่อแทนความหมายของข้อมูลในระดับสูงโดยการสร้างสถาปัตยกรรมข้อมูลขึ้นมาที่ประกอบไปด้วยโครงสร้างย่อยๆหลายอัน ซึ่งมีที่มาจาก Artificial Neural Network หมายถึง โครงข่ายใยประสาทเทียม โดยอาศัยแนวคิดและเทคนิคจากการทำงานของระบบโครงข่ายใยประสาทในระบบประสาทของมนุษย์ โดยจำลองการทำงานเหมือนกับกลุ่มเซลล์ประสาทที่เชื่อมโยงกันเป็นระบบประสาทที่สามารถรับรู้หลายๆ สิ่งในเวลาเดียวกัน ด้วยการประมวลผลแบบขนาน (Paralle Network) ทำให้ระบบสามารถตัดสินใจได้ใกล้เคียงกับมนุษย์ ในการที่เครื่องจะสามารถเข้าใจสิ่งต่างๆ ได้ก็จำเป็นที่จะต้องมี “องค์ความรู้ (Knowledge)” เสียก่อน [input layer] จากนั้นก็จะประเมินชุดข้อมูล [hidden layer] และนำเสนอหรือแทนองค์ความรู้นั้น [output layer] แต่ความแตกต่างระหว่าง Deep learning กับ Artificial Neural Network ก็คือระดับ hidden layer ที่ใน Deep learning มี hidden layer มากกว่าใน Artificial Neural Network [5]




รูปภาพประกอบที่ 3 เปรียบเทียบความแตกต่างของชั้น hidden layer ระหว่าง Artificial Neural Network และ Deep learning โดยทางซ้ายคือ Artificial Neural Network และทางขวาคือ Deep learning (https://medium.com/machines-school/deep-leaning-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3-785e16d01773)

2.4.1 Convolutional Neural Network

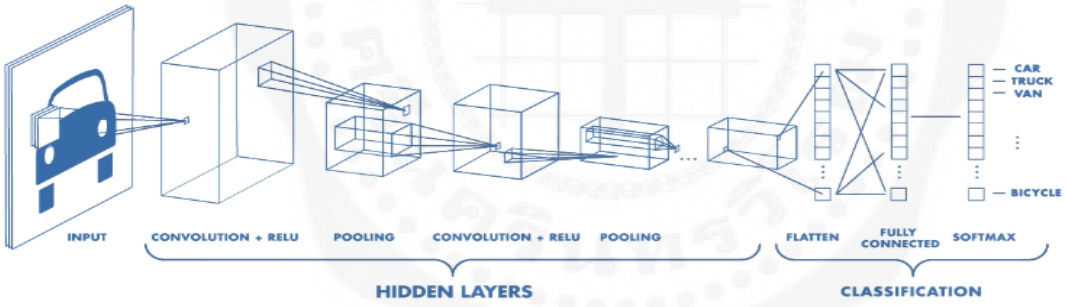
Convolutional Neural Network (CNN) หรือ โครงข่ายประสาทแบบคอนโวลูชัน เป็นโครงข่ายประสาทเทียมหนึ่งในกลุ่ม bio-inspired โดยที่ CNN จะจำลองการมองเห็นของมนุษย์ที่มองพื้นที่เป็นที่ย่อยๆ และนำกลุ่มของพื้นที่ย่อยๆมาผสานกัน เพื่อดูว่าสิ่งที่เห็นอยู่เป็นอะไรกันแน่



รูปภาพประกอบที่ 4 ประกอบการอธิบายคร่าวๆเกี่ยวกับ Convolutional Neural Network (https://medium.com/@natthawatphongchit/%E0%B8%A1%E0%B8%B2%E0%B8%A5%E0%B8%AD%E0%B8%87%)

ถ้าเราเปรียบว่ากรอบสี่เหลี่ยมสีเหลืองนั้น คือพื้นที่ที่มนุษย์กำลังให้ความสนใจอยู่ แต่เราสามารถรับรู้ได้ว่าสิ่งนี้คือหนู เพราะเรากวาดสายตามองรอบๆ

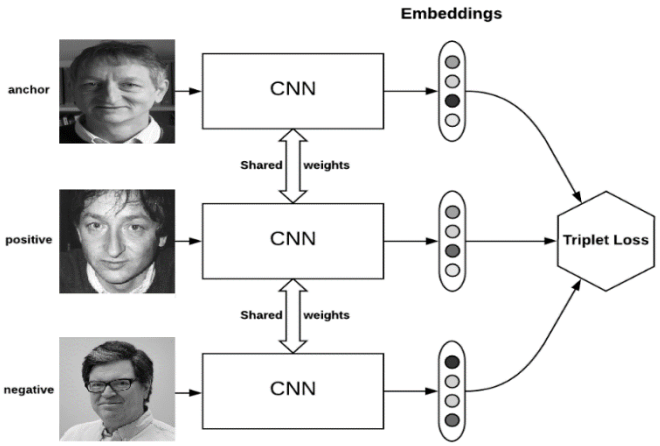
Convolutional Neural Network จัดว่าเป็น Deep learning ประเภทหนึ่ง โดยประกอบไปด้วยหลายๆ layer เช่น input layer, convolution layer, pooling layer และ fully connected เป็นต้น [6]




รูปภาพประกอบที่ 5 Convolutional Neural Network Architecture (https://medium.com/@natthawatphongchit/%E0%B8%A1%E0%B8%B2%E0%B8%A5%E0%B8%AD%E0%B8%87%)

2.4.2 FaceNet

FaceNet นั้นใช้ Deep convolutional neural network (CNN) ในการเรียนรู้ โดยเป็นเครือข่ายที่ได้เรียนรู้ถึง squared L2 distances ระหว่าง embedding ของใบหน้า ภาพที่ใช้สำหรับการเรียนรู้นั้นจะถูกปรับขนาด แปลง และครอบบริเวณใบหน้า และอีกอย่างหนึ่งที่สำคัญคือ FaceNet ก็คือ loss Function โดยที่จะใช้ triplet loss function โดยในการคำนวณ triplet loss นั้น เราต้องใช้ภาพ 3 ภาพ โดยแต่ละภาพจะแทนโดยการเรียกว่า anchor, positive และ negative [7]



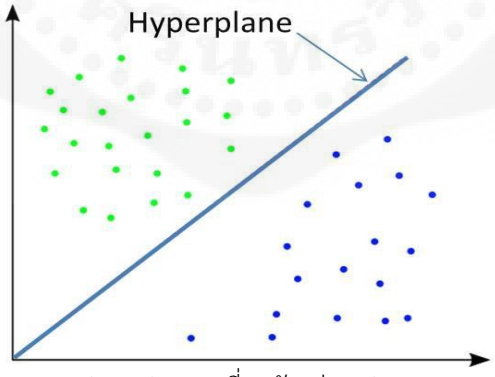

รูปภาพประกอบที่ 6 ตัวอย่างการใช้ embeddings ใบหน้าร่วมกับ triplet loss function (https://medium.com/analytics-vidhya/introduction-to-facenet-a-unified-embedding-for-face-recognition-and-clustering-dbdac8e6f02)

## 2.5 Classification

เป็นกระบวนการจัดกลุ่มข้อมูลที่มีความคล้ายกันให้อยู่ด้วยกัน โดยการสร้างโมเดลขึ้นมาเพื่อที่จะจำแนกข้อมูล(Classifier) โดยการ predict ข้อมูล วิธีนี้สามารถนำมาใช้ในการเพิ่มความเร็วและประสิทธิภาพในการ predict ใบหน้าได้

### 2.5.1 Support Vector Machines (SVM)

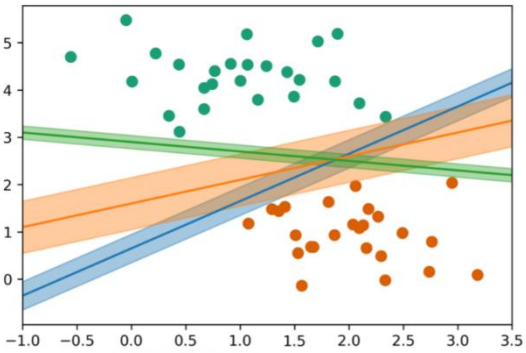
เป็นหนึ่งในโมเดล Machine Learning ที่ใช้ในการจำแนกข้อมูล หรือแบ่งกลุ่มข้อมูลโดยจะสร้างเส้นตรงที่ใช้แบ่งกลุ่มข้อมูล (Hyperplane) และหาเส้นที่ดีที่สุด




รูปภาพประกอบที่ 7 ตัวอย่างรูป SVM (https://medium.com/@pradyasin/support-vector-machines-svm-943f9a732a69)

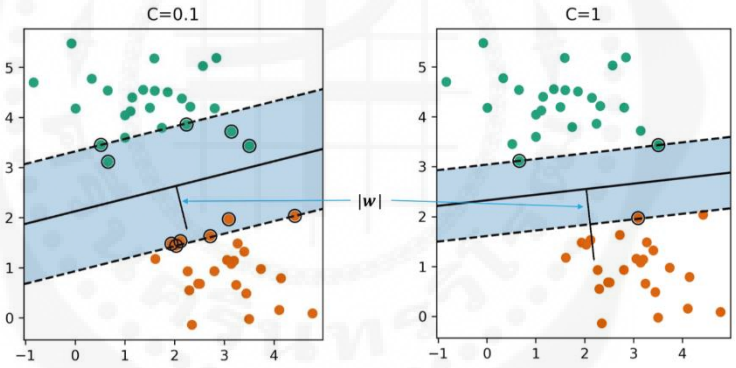
Max-Margin and Support Vectors เป็นวิธีในการแบ่งข้อมูลโดยเราจะเลือกเส้นที่มี Margin มากที่สุด คือ เส้นที่มีระยะแบ่งกว้างที่สุด เช่น เส้นสีส้มมีระยะมากที่สุด หาก Margin แคบไป

ขยับข้อมูลเดียวอาจจะทำให้ข้ามไปอีกฝั่งหนึ่งได้เลยทำให้มีโอกาส Overfit สูง ดังนั้นเราจะเลือก Margin เยอะ ทำให้ Overfit น้อย หรือเรียกว่า Soft Margin



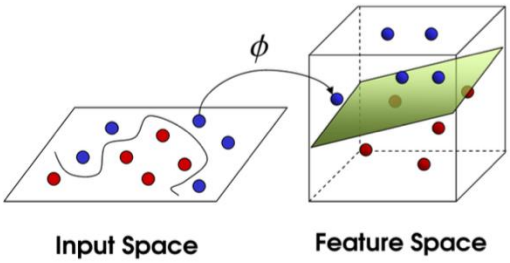
รูปภาพประกอบที่ 8 ตัวอย่างการแบ่งข้อมูล
(https://medium.com/@pradyasin/support-vector-machines-svm-943f9a732a69)

การปรับ parameter C จะทำให้ขนาดของเส้นแบ่งเปลี่ยนแปลงได้โดยที่ C มากจะทำให้พื้นที่แคบลง C น้อยจะทำให้พื้นที่กว้างขึ้น




รูปภาพประกอบที่ 9 ตัวอย่าง parameter C
(https://medium.com/@pradyasin/support-vector-machines-svm-943f9a732a69)

หากข้อมูลไม่สามารถแบ่งกลุ่มได้ด้วยเส้นตรง (linear) จึงได้มีวิธีการ Kernels ที่เป็น non-linear เข้ามาแก้ไขปัญหาที่เกิดขึ้น โดยวิธีการคือ สร้างมิติขึ้นมาจากเดิม 2D เป็น 3D แล้วลากเส้นตัดผ่านตรงกลางจะทำให้สามารถแบ่งข้อมูลออกไปกลุ่มได้ [8]

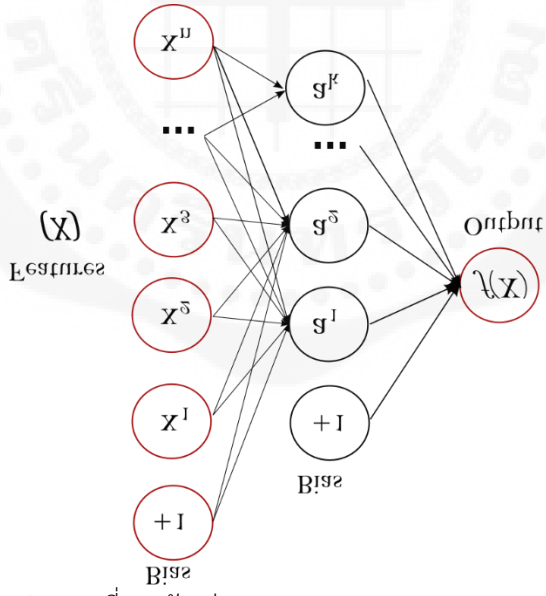



รูปภาพประกอบที่ 10 ตัวอย่างการทำ Kernels

(https://medium.com/@pradyasin/support-vector-machines-svm-943f9a732a69)

2.5.2 MLPClassifier (MLP)

Multilayer Perceptron (MLP) เป็นชั้นหนึ่งของโครงข่ายประสาทเทียมแบบ feedforward MLP ประกอบด้วยโหนดอย่างน้อยสามชั้น ได้แก่ input layer, hidden layer และ output layer แต่ละโหนดเป็นเซลล์ประสาทที่ใช้ฟังก์ชันการกระตุ้นแบบไม่เชิงเส้นยกเว้นโหนดอินพุต MLP ใช้เทคนิคการเรียนรู้ supervised learning ที่เรียกว่า backpropagation สำหรับการฝึกอบรม หลายเลเยอร์และการกระตุ้นแบบไม่เป็นเชิงเส้นทำให้ MLP แตกต่างจากตัวรับรู้เชิงเส้น สามารถแยกแยะข้อมูลที่ไม่สามารถแยกออกจากกันได้ [9]

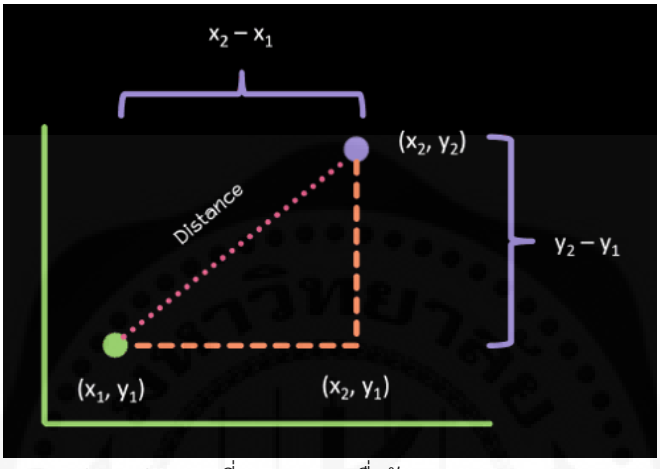


รูปภาพประกอบที่ 12 ตัวอย่าง MLP แบบ one hidden layer MLP

https://scikit-learn.org/stable/modules/neural_networks_supervised.html

2.5.3 K-Nearest Neighbors (KNN)

ขั้นตอนวิธีการเพื่อนบ้านใกล้ที่สุด (K-Nearest Neighbour Algorithm) เป็นวิธีที่ใช้ในการจัดแบ่งคลาส โดยเทคนิคนี้จะตัดสินใจว่า คลาสใดที่จะแทนเงื่อนไขหรือกรณีใหม่ๆ ได้บ้าง โดยการตรวจสอบจำนวนบางจำนวน ในขั้นตอนวิธีการเพื่อนบ้านใกล้ที่สุด ของกรณีหรือเงื่อนไขที่เหมือนกันหรือใกล้เคียงกันมากที่สุด โดยจะหาผลรวม (Count Up) ของจำนวนเงื่อนไข หรือกรณีต่างๆ สำหรับแต่ละคลาส และกำหนดเงื่อนไขใหม่ๆ ให้คลาสที่เหมือนกันกับคลาสที่ใกล้เคียงกันมากที่สุด [10]




รูปภาพประกอบที่ 13 สมการเพื่อวัดระยะทางของ KNN
https://www.glurgeek.com/education/knn/

2.6 Random Search

เป็นเทคนิคที่ใช้ชุดค่าผสมแบบสุ่มของไฮเปอร์พารามิเตอร์เพื่อค้นหาโซลูชันที่ดีที่สุดสำหรับโมเดลที่สร้างขึ้น คล้ายกับการค้นหาแบบกริด แต่ก็พิสูจน์แล้วว่าให้ผลลัพธ์ที่ดีกว่าเมื่อเปรียบเทียบกัน ข้อเสียของการค้นหาแบบสุ่มคือให้ผลต่างระหว่างการคำนวณสูง เนื่องจากการเลือกพารามิเตอร์เป็นแบบสุ่มอย่างสมบูรณ์ [11]

**2.7.งานวิจัยที่เกี่ยวข้อง**

2.7.1 Attendance System based on Deep Learning Face Recognition without Queue

ผู้แต่ง : Bana Handaga, Budi Murtiyasa, Jan Wantoro

งานวิจัยนี้เกี่ยวกับการนำเทคโนโลยีจดจำใบหน้ามาใช้เพื่อเช็คในชั้นเรียนโดยจะทำการเปรียบเทียบระหว่าง Machine learning algorithms ทั้ง 3 แบบ คือ Haar cascade, Deep learning, Hog(Histogram of Oriented Gradient) ซึ่งผลลัพธ์ที่ได้คือ deep learning เหมาะที่จะนำมาใช้ในห้องเรียนมากที่สุดดังตารางนี้ [12]

TABLE I. THE RESULTS OF DETECTION OF THREE FACE DETECTORS

| Method | Number of face | True Positive | False Positive |
|---|---|---|---|
| Haarcascade | 15 | 9 | 2 |
| Deep Learning | 15 | 11 | 0 |
| Hog | 15 | 3 | 0 |

รูปภาพประกอบที่ 14 ตารางผลลัพธ์ของ algorithms 3 แบบ

(https://ieeexplore-ieee-org.clvpn.swu.ac.th/document/8985697)

**ข้อดี :** เนื้อหาจากงานวิจัยมีการนำอัลกอริทึมการตรวจจับใบหน้าทั้ง 3 แบบ ได้แก่ Haarcascade Deep learning และ HOG มาเปรียบเทียบประสิทธิภาพของการทำนายใบหน้า ซึ่งทำให้สามารถนำวิธีนี้ไปใช้และทำตามงานวิจัย เพื่อวัดประสิทธิภาพของอัลกอริทึมได้

**ข้อเสีย :** การตรวจจับใบหน้าของงานวิจัยนี้เป็นการตรวจจับใบหน้าแบบหลายบุคคลพร้อมกัน ซึ่งถ้าหากมีจำนวนเยอะก็อาจทำให้อัลกอริทึมที่ใช้ในการตรวจจับใบหน้าและทำนายใบหน้าใช้เวลานานในการตรวจจับและเกิดความผิดพลาดในการตรวจจับใบหน้าได้ ซึ่งจากผลการทดลองใบหน้าด้วยอัลกอริทึมที่ดีที่สุด Deep learning ก็ยังมีผลการทดลองที่เป็น True Positive แค่ 11 ใบหน้าจากทั้งหมด 15 ใบหน้า ซึ่งความแม่นยำก็ยังคงไม่มากพอ

**แนวทางการพัฒนา :** นำระบบการรู้จำใบหน้ามาพัฒนาบนโทรศัทพ์มือถือในรูปแบบแอปพลิเคชันแอนดรอยด์ โดยผู้ใช้แต่ละคนก็จะมีแอปพลิเคชันนี้บนมือถือของตัวเองเพื่อแก้ปัญหาการต่อคิวเช็คชื่อตามแนวทางของงานวิจัยนี้และยังแก้ปัญหาการตรวจจับใบหน้าที่ผิดพลาดจากการต้องตรวจจับใบหน้าพร้อมกันหลายๆคน

2.7.2 Face and Lip-reading Authentication System Based on Android Smart Phones

ผู้แต่ง : Dongpeng Shang, Xinman Zhang, Xuebin Xu

งานวิจัยนี้เกี่ยวกับการใช้ใบหน้าและการอ่านปากเพื่อระบุตัวบุคคล บนมือถือแอนดรอยด์ ซึ่งประกอบไปด้วยการตรวจจับใบหน้าโดยใช้ AdaBoost algorithm,การ face feature extraction ด้วย Linearity Preserving Projection (LPP),การ lip feature extraction ด้วย Projection Local Spatiotemporal Descriptor (PLSD) [13]

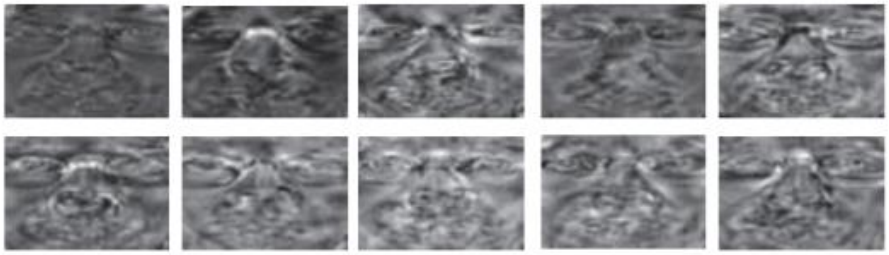


Fig.2 Example of the fist 10 eigenvectors solved by LPP.

รูปภาพประกอบที่ 15 ตัวอย่างของ 10 eigenvectors แรกโดยวิธี LPP
( https://ieeexplore-ieee-org.clvpn.swu.ac.th/document/8623298)

**ข้อดี :** เนื้อหาในงานวิจัยนี้นำเสนอการยืนยันตัวตนด้วยใบหน้า และการยืนยันตัวตนด้วย Lip-reading มารวมกัน เพื่อเพิ่มประสิทธิภาพในการยืนยันตัวตน โดยทำการเปรียบเทียบการยืนยันตัวตนด้วยใบหน้าด้วยอัลกอริทึม Linearity Preserving Projection (LPP) ในการทำ feature extraction และ Lip-reading ด้วยอัลกอริทึม Projection Local Spatiotemporal Descriptor (PLSD) ในการทำ feature extraction ก่อน จากนั้นจึงนำทั้งสอง แบบมารวมกัน โดยจากผลการทดลอง ทำให้เห็นได้ว่าการนำทั้งสองอัลกอริทึมมารวมกันนั้น สามารถเพิ่มประสิทธิภาพของการยืนยันตัวตนได้ ซึ่งเป็นแนวทางได้ในการทำงานวิจัยที่เกี่ยวกับการยืนยันตัวตนด้วยใบหน้า โดยนำจุดเด่นของแต่ละอัลกอริทึมมารวมกัน เพื่อเพิ่มประสิทธิภาพในการทำนายใบหน้า
**ข้อเสีย :** ในงานวิจัยนี้นั้น เนื้อหาของงานวิจัยจะมุ่งเน้นที่การรวมอัลกอริทึมทั้งสองเข้าด้วยกัน และมุ่งเน้นที่ส่วนของ Lip-reading ซึ่งทำให้ในส่วนของเนื้อหาและวิธีการของ Face Recognition ของงานวิจัยนี้ไม่ได้อธิบายละเอียดมากนัก จึงต้องไปศึกษาจากงานวิจัยอื่นๆเพิ่มเติม
**แนวทางการพัฒนา :** นำแนวทางการรวมอัลกอริทึมในการยืนยันตัวตนจากงานวิจัยนี้ไปปรับใช้ในงานวิจัยที่จะทำเพื่อเพิ่มประสิทธิภาพในการทำนายใบหน้าในงานวิจัย โดยอาจจะนำจุดเด่นของแต่ละอัลกอริทึมในการทำ feature extraction มาใช้ร่วมกัน จากนั้นดูผลการทดลองที่เกิดขึ้น

2.7.3 Real-Time Smart Attendance System using FaceRecognition Techniques
ผู้แต่ง : Shreyak Sawhney, Karan Kacker, Samyak Jain, Shailendra Narayan Singh, RakeshGarg

งานวิจัยนี้แสดงให้เห็นว่าการจดจำใบหน้านั้นเหมาะกับการนำมาใช้เช็คชื่อเข้าเรียนเป็นอย่างมาก เนื่องจากมีความแม่นยำที่สูง และยากที่จะแทรกแซง นอกจากนี้ยังไม่จำเป็นต้อง hardware ที่มีราคาสูง โดยมีแค่ กล้อง,คอมพิวเตอร์ และฐานข้อมูลก็พอ [14]

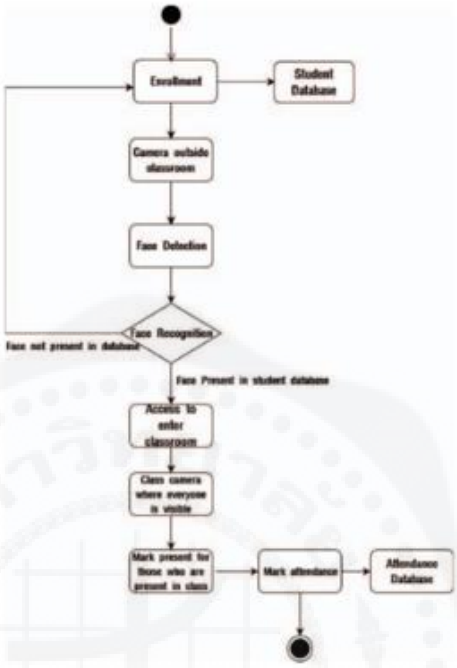



Fig. 2. Activity diagram for the smart attendance system

รูปภาพประกอบที่ 16 Activity diagram สำหรับ smart attendance system (https://ieeexplore-ieee-org.clvpn.swu.ac.th/document/8776934)

**ข้อดี :** เนื้อหาในงานวิจัยมีการนำเสนอเทคโนโลยีระบบรู้จำใบหน้ามาปรับใช้ในระบบการเช็คชื่อได้เป็นอย่างดี โดยนำเสนอระบบและขั้นตอนได้ชัดเจนและครบถ้วน

**ข้อเสีย :** ในขั้นตอน Methodology ในส่วนของ Face Recognition นั้นนำเสนอเพียงแต่การนำวิธีการ PCA มาใช้วิเคราะห์ eigenface เพื่อแยกแยะใบหน้าเท่านั้น ซึ่งแค่เพียงวิธีนี้อาจไม่เพียงพอในการเพิ่มความแม่นยำและถูกต้องสำหรับระบบการรู้จำใบหน้า

**แนวทางการพัฒนา :** นำ Deep learning มาใช้เพื่อเพิ่มประสิทธิภาพในการรู้จำใบหน้าและใช้การ Classification เพื่อเพิ่มประสิทธิภาพในการทำนายใบหน้า

# บทที่ 3
# วิธีการดำเนินงาน

## 3.1 วิธีการดำเนินงาน

3.1.1 ศึกษาข้อมูลเกี่ยวกับ Face recognition

3.1.2 ศึกษาอัลกอริทึมในการ Face Detection และ Image Preprocessing

3.1.3 ศึกษาอัลกอริทึมและงานวิจัยที่เกี่ยวกับ Deep learning ในการทำ Face recogition

3.1.4 เก็บรวบรวมรูปภาพใบหน้าของนิสิตคณะวิทยาศาสตร์จำนวน 63 คน จำนวน 3,891 รูป โดยแบ่งข้อมูลไว้สำหรับ train จำนวน 3,173 รูป และ test จำนวน 718 รูป

3.1.5 นำรูปภาพมาทำการตรวจจับใบหน้าด้วย MTCNN (Multi-task Cascaded Convolutional Networks) และทำการปรับขนาดรูปภาพให้เป็น 160 x 160 pixels

3.1.6 นำรูปภาพที่ผ่านการทำตรวจจับใบหน้าและปรับขนาดรูปภาพมาทำการ feature extraction โดยใช้ FaceNet โดย FaceNet จะทำการสร้าง Embedding Features จากบริเวณที่สำคัญของใบหน้า ได้แก่ ตา จมูก และ ปาก ให้เป็น 128-dimensional vector

3.1.7 ทำการ classification รูปภาพที่ผ่านการ feature extraction โดยใช้ KNN classifier, MLP classifier และ SVM classifier และเปรียบเทียบประสิทธิภาพการทำงาน เพื่อหาโมเดลที่ดีที่สุด

3.1.8 ทดสอบการทำนายรูปภาพใบหน้า จากนั้นทำการวิเคราะห์ประสิทธิภาพของโมเดลที่ได้

3.1.9 ปรับปรุงแก้ไขหาค่าที่เหมาะสมของโมเดล

3.1.10 พัฒนาและประยุกต์ใช้โมเดลที่ได้บนแอปพลิเคชัน

3.1.11 พัฒนาเว็บไซต์เพื่อแสดงข้อมูลที่ทดลอง

3.1.12 สรุปผลการดำเนินงานและจัดทำรูปเล่ม

### 3.2 แผนการดำเนินงานตลอดโครงงาน

| ขั้นตอนการดำเนินงาน | 2563 | | | | | 2564 | |
|---|---|---|---|---|---|---|---|
| | ส.ค. | ก.ย. | ต.ค. | พ.ย. | ธ.ค. | ม.ค. | ก.พ. |
| **เริ่มต้นและวางแผนโครงการ** | | | | | | | |
| 1. เลือกหัวข้อโครงการ | | | | | | | |
| 2. หาข้อมูลและศึกษาทฤษฎี | | | | | | | |
| 3. เลือกเครื่องมือ | | | | | | | |
| 4.กำหนดรายละเอียดของโครงการ | | | | | | | |
| **การวิเคราะห์ระบบ** | | | | | | | |
| 5. ศึกษาการวิเคราะห์และประมวลผลข้อมูลที่ใช้ | | | | | | | |
| 6. ศึกษาโปรแกรมที่ใช้ในการทำโครงการ | | | | | | | |
| 7. ศึกษาอัลกอริธึมที่ใช้ในการทำวิจัย | | | | | | | |
| **การออกแบบและพัฒนาระบบ** | | | | | | | |
| 8. รวบรวมข้อมูลที่ใช้ | | | | | | | |
| 9. ทำการแปลงข้อมูลก่อนการประมวลผล | | | | | | | |
| 10. นำข้อมูลไปทำการเทรนโมเดล | | | | | | | |
| 11. ทดสอบโมเดล | | | | | | | |
| 12. ปรับปรุงแก้ไขโมเดล | | | | | | | |
| **การปรับใช้ระบบจริง** | | | | | | | |
| 13. พัฒนาและประยุกต์ใช้โมเดลบนแอปพลิเคชันแอนดรอยด์ | | | | | | | |
| 14. พัฒนาเว็บไซต์สำหรับแสดงผลข้อมูลที่ทดลอง | | | | | | | |
| 15. สรุปงานวิจัย | | | | | | | |
| 16. จัดทำรูปเล่มรายงานและเผยแพร่งานวิจัย | | | | | | | |

ตารางที่ 1 แผนการดำเนินโครงการ

**3.3 อุปกรณ์และเครื่องมือที่ใช้ในงานวิจัย**

3.3.1 ฮาร์ดแวร์ (Hardware):

3.3.1.1 Notebook: AMD Ryzen 7 3750H with Radeon Vega Mobile Gfx 2.30 GHz

3.3.1.2 โทรศัพท์มือถือ: Smartphone Android and iOS

3.3.1.3 เครื่องเซิร์ฟเวอร์: ระบบปฏิบัติการ CentOS 7 (64bit)

3.3.2 ซอฟต์แวร์ (Software):

3.3.2.1 Google Collaboratory

3.3.2.2 Android Studio

3.3.2.3 Visual Studio Code

3.3.2.4 Flutter

3.3.2.5 Flask

3.3.3 ภาษาที่ใช้ (Programming language):

3.3.3.1 Python

3.3.3.2 Dart

3.3.3.3 PHP

3.3.3.4 SQL

3.3.4 Dataset

3.3.4.1 รูปภาพใบหน้าของนิสิตคณะวิทยาศาสตร์ 63 คน ภาพใบหน้าจากในอินเทอร์เน็ตจำนวน 12 คน รวมทั้งสิ้น 75 คน จำนวน 4,506 รูป โดยแบ่งเป็น

-train 3,728 ใบหน้า

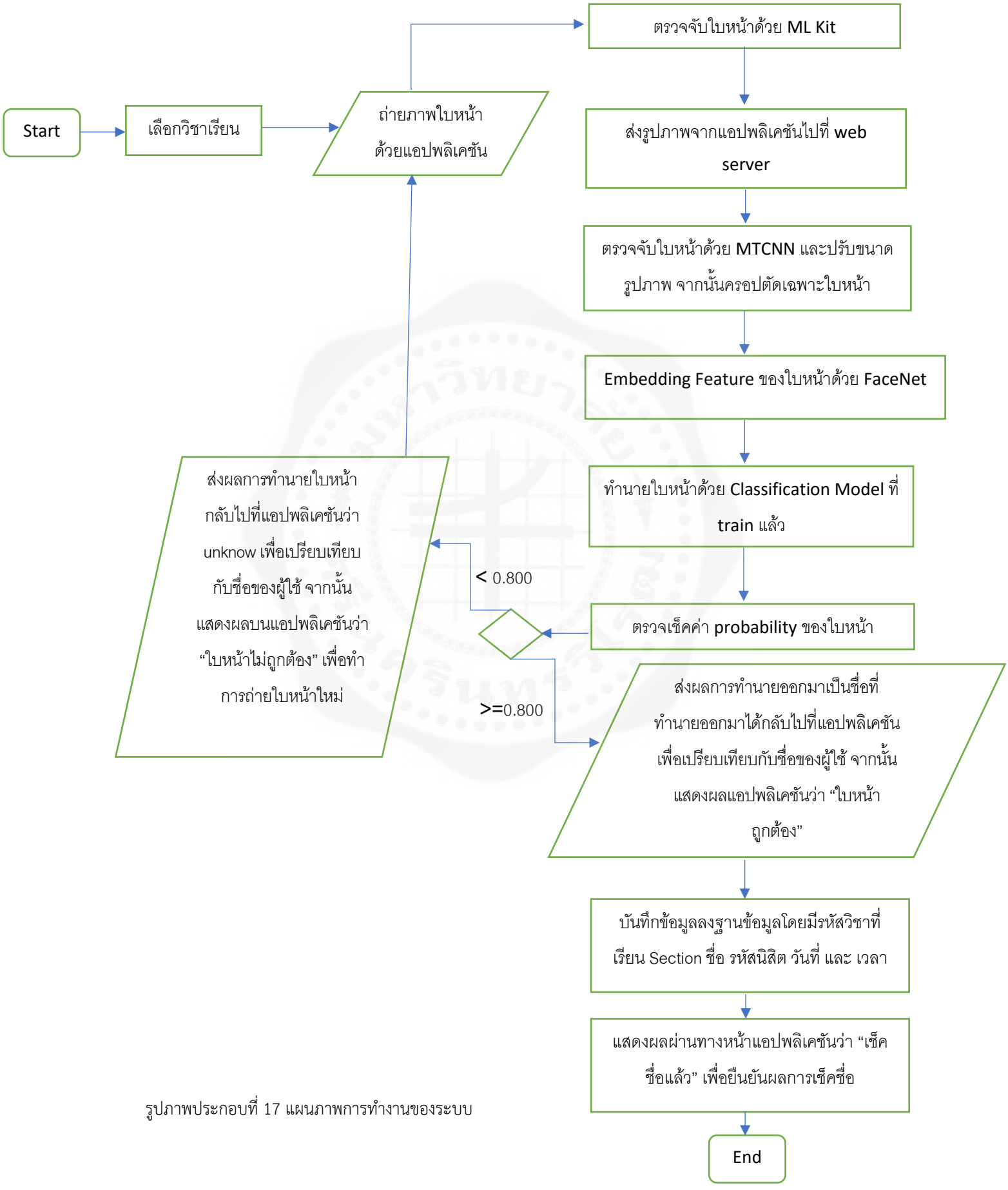
-test 778 ใบหน้า

**3.4 แนวคิดการทำงานของระบบหลัก**

แนวคิดการทำงานของระบบมีดังนี้

1) เมื่อผู้ใช้เข้าสู่ระบบแล้ว จะต้องทำการเลือกวิชาที่ตัวเองได้ลงทะเบียนไว้ โดยจะมีให้เลือกวิชาเรียน
2) จากนั้นผู้ใช้ทำการใช้กล้องของสมาร์ทโฟนถ่ายภาพใบหน้าตัวเอง
3) แอปพลิเคชันจะทำการตรวจจับใบหน้าจากภาพที่รับเข้ามาโดยใช้ Face Detection ของ ML Kit
4) เมื่อตรวจจับพบใบหน้าแล้วจะทำการส่งรูปภาพไปที่ web server เพื่อประมวลผลรูปภาพ โดยทำการตรวจจับใบหน้าโดยใช้ MTCNN และปรับขนาดรูปภาพ จากนั้นครอบตัดเฉพาะส่วนใบหน้า
5) เมื่อประมวลผลเสร็จแล้ว จะนำภาพใบหน้าไปทำการ Feature Extraction โดยใช้ FaceNet ทำการ Embedding Features จากบริเวณที่สำคัญของใบหน้า ได้แก่ ตา จมูก และ ปาก ให้เป็น 128-dimensional vector
6) จากนั้นจะนำภาพจากขั้นตอนที่ 5 ไปทำนายใบหน้าโดยใช้ Classification model ที่ผ่านการ train มาแล้ว
7) โดยจะส่งค่าที่ทำนายได้เป็นชื่อที่ทำนายออกมา โดยก่อนที่จะส่งค่ากลับไปที่แอปพลิเคชันนั้น จะทำการตรวจค่า probability ที่ทำนายออกมาก่อน โดยถ้าค่า probability ของภาพมีค่ามากกว่าหรือเท่ากับ 0.800 จึงจะทำการส่งข้อความกลับไปที่แอปพลิเคชันเป็นชื่อที่ทำนายออกมาได้ เพื่อเปรียบเทียบกับชื่อผู้ใช้ ถ้าตรงกันก็จะแสดงผลบนแอปพลิเคชันว่า “ใบหน้าถูกต้อง” แต่ถ้าไม่ตรงกันก็จะแสดงผลบนแอปพลิเคชันว่า “ใบหน้าไม่ถูกต้อง” และหากค่า probability ต่ำกว่า 0.800 จะทำการส่งข้อความกลับไปที่แอปพลิเคชันว่า “Unknow” เพื่อเปรียบเทียบกับชื่อผู้ใช้ จากนั้นแสดงผลบนแอปพลิเคชันว่า “ใบหน้าไม่ถูกต้อง” เพื่อทำการถ่ายใบหน้าใหม่
8) เมื่อได้รับการยืนยันใบหน้าแล้ว จึงจะทำการบันทึกข้อมูลลงฐานข้อมูลโดยมีรหัสวิชาที่เรียน Section ชื่อ รหัสนิสิต วันที่ จากนั้นให้แสดงผลผ่านทางหน้าแอปพลิเคชันว่า “เช็คชื่อแล้ว” เพื่อยืนยันผลการเช็คชื่อ

3.4.1 แผนภาพการทำงานของระบบ




รูปภาพประกอบที่ 17 แผนภาพการทำงานของระบบ

## 3.5 ขั้นตอนการ train classifier

3.5.1 รวบรวมรูปภาพใบหน้า

ทำการเก็บรวบรวมรูปภาพใบหน้าของนิสิตคณะวิทยาศาสตร์จำนวน 63 คน ด้วยกล้องจากโทรศัพท์มือถือและรวบรวมภาพใบหน้าจากในอินเทอร์เน็ตจำนวน 12 คน รวมทั้งสิ้น 75 คน โดยเก็บรวบรวมรูปภาพได้ทั้งหมด 4,506 รูป โดยแบ่งข้อมูลไว้สำหรับ train จำนวน 3,728 รูป และ test จำนวน 778 รูป

3.5.2 ครอปตัดภาพเพื่อเอาเฉพาะส่วนใบหน้า

การ crop ภาพเพื่อเอาเฉพาะใบหน้า ได้ใช้library ของ python ชื่อ MTCNN (Multi-task Cascaded Convolutional Networks) ซึ่งเป็นระบบการ detect ใบหน้าซึ่งใช้ neural network ซึ่งมีการเทรนมาแล้วมาใช้ในการหาภาพของใบหน้า แล้วเก็บเฉพาะใบหน้าของบุคคลและทำการเปลี่ยนขนาด ของภาพให้เป็นขนาด 160x160 pixels เพื่อให้ตรงกับขนาด input ของตัว pre-trained FaceNet model

รูปภาพประกอบที่ 18 ตัวอย่างรูปภาพใบหน้าที่ทำการปรับขนาดและตรวจจับใบหน้า

3.5.3 การทำ Feature Extraction

ในขั้นตอนนี้จะใช้ pre-trained FaceNet model มาใช้ในการทำ Feature Extraction โดย FaceNet จะทำการสร้าง Embedding Features จากบริเวณที่สำคัญของใบหน้า ได้แก่ ตา จมูก และ ปาก ให้เป็น 128-dimensional vector

| | | | |
|---|---|---|---|
| Block8_6_Branch_0_Conv2d_1x1_Ac | (None, 3, 3, 192) | 0 | Block8_6_Branch_0_Conv2d_1x1_Batc |
| Block8_6_Branch_1_Conv2d_0c_3x1 | (None, 3, 3, 192) | 0 | Block8_6_Branch_1_Conv2d_0c_3x1_B |
| Block8_6_Concatenate (Concatena | (None, 3, 3, 384) | 0 | Block8_6_Branch_0_Conv2d_1x1_Acti<br>Block8_6_Branch_1_Conv2d_0c_3x1_A |
| Block8_6_Conv2d_1x1 (Conv2D) | (None, 3, 3, 1792) | 689920 | Block8_6_Concatenate[0][0] |
| Block8_6_ScaleSum (Lambda) | (None, 3, 3, 1792) | 0 | Block8_5_Activation[0][0]<br>Block8_6_Conv2d_1x1[0][0] |
| AvgPool (GlobalAveragePooling2D | (None, 1792) | 0 | Block8_6_ScaleSum[0][0] |
| Dropout (Dropout) | (None, 1792) | 0 | AvgPool[0][0] |
| Bottleneck (Dense) | (None, 128) | 229376 | Dropout[0][0] |
| Bottleneck_BatchNorm (BatchNorm | (None, 128) | 384 | Bottleneck[0][0] |

Total params: 22,808,144
Trainable params: 22,779,312
Non-trainable params: 28,832

รูปภาพประกอบที่ 19 แสดงจำนวน parameter ของ FaceNet โดยมีจำนวน parameter ทั้งหมดที่ 22,808,144

```
[[ 0.90788305 -0.76819015  0.04160831 ... -1.5078577   0.320834
  -0.29714647]
 [ 0.90788305 -0.76819015  0.04160831 ... -1.5078577   0.320834
  -0.29714647]
 [ 0.97867775 -0.8037925   0.15453851 ... -1.6509684   0.44819847
  -0.16910978]
 ...
 [ 0.82214856 -1.1532798  -0.35560974 ... -1.9135838   0.11437713
  -0.6188901 ]
 [ 0.38325924 -1.4644454   0.19646975 ... -1.3076812   0.8852212
   0.06499656]
 [ 0.8712493  -1.0904417  -0.2606301  ... -1.178979    0.67921776
  -0.263938  ]]
```

รูปภาพประกอบที่ 20 ข้อมูลภาพใบหน้าที่ถูก Embedding Features ให้เป็น 128-dimensional vector

3.5.4 การ train classifier

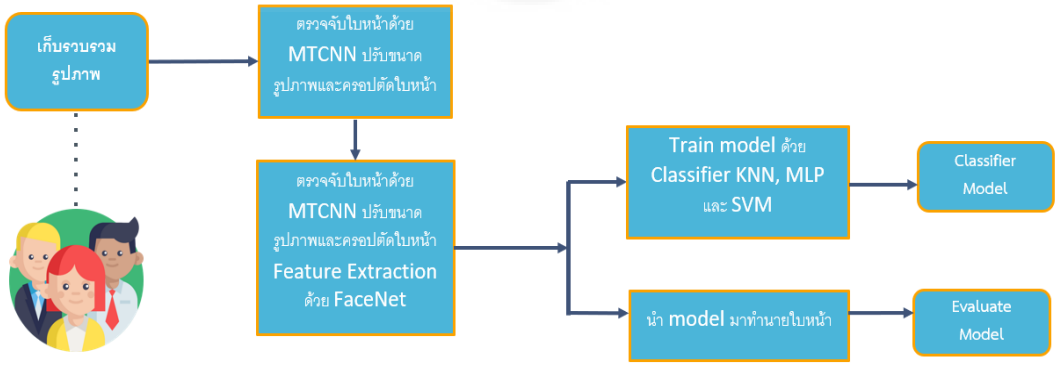
ในส่วนนี้จะทำการ train ข้อมูลด้วย classifier 3 แบบ ได้แก่ KNN(K-Nearest Neighbors), MLP(Multilayer Perceptron) และ SVM(Support Vector Machines) เพื่อเปรียบเทียบประสิทธิภาพการทำงานของตัว classifier แบบต่างๆ โดยจะทำการเลือกโมเดลที่ใช้ classifier ที่ให้ผลที่ดีที่สุด โดยได้ใช้ Random Search ในการหาค่า parameter ของแต่ละ classifier ที่เหมาะสมสำหรับข้อมูลและงานวิจัยนี้ โดย parameter ที่ดีที่สุดในการใช้ Random Search มีดังนี้

**KNN : n_neighbors** = 10, **weights** = uniform, **metric** = manhattan, **leaf size** = 52, **algorithm** = auto และ p = 2

**MLP : hidden_layer_sizes** = 20, **activation** = relu, **solver** = adam, **learning rate** = constant, **Alpha** = 0.0001, และ **max_iter**=100

**SVM : C** = 70, **gamma** = 0.1 และ **kernel** = linear

3.5.4.1 แผนภาพขั้นตอนการ train classifier




รูปภาพประกอบที่ 21 แผนภาพขั้นตอนการ train classifier

# บทที่ 4

# สรุปผลการดำเนินงาน

## 4.1 การพัฒนาส่วนโมเดล

เมื่อได้โมเดลที่ทำการ train เรียบร้อยแล้ว ก็จะนำมาโมเดลที่ได้มาทำการหาค่าต่างๆ เพื่อเลือกโมเดลที่เหมาะสมที่สุดที่จะนำไปใช้ทำนายใบหน้าบนเซิร์ฟเวอร์ โดยผลลัพธ์ที่ได้มีดังนี้

เมื่อนำแต่ละโมเดลมาหาค่า accuracy ของ train data และ test data ก็จะได้ผลลัพธ์ดังตาราง

| Classifier | Train | Test |
|---|---|---|
| KNN | 0.999 | 0.987 |
| MLP | 0.999 | 0.967 |
| SVM | 1.000 | 0.987 |

ตารางที่ 2 ตารางแสดงค่า Accuracy ของ train data และ test data ของแต่ละ classifier

เมื่อนำแต่ละโมเดลมาหาค่า Accuracy, Precision, Recall และ F1-score ของ test data ก็จะได้ผลลัพธ์ดังตาราง

| Classifier | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|
| KNN | 0.987 | 0.98 | 0.98 | 0.98 |
| MLP | 0.967 | 0.95 | 0.96 | 0.95 |
| SVM | 0.987 | 0.98 | 0.98 | 0.98 |

ตารางที่ 3 ตารางแสดงค่า Accuracy, Precision, Recall และ F1-score ของ test data ของแต่ละ classifier

เมื่อนำแต่ละโมเดลมาหาค่าเฉลี่ยของ probability ในแต่ละใบหน้า test data โดยทำการหาค่าเฉลี่ยในแต่ละโมเดลจำนวน 5 ครั้งและนำ 5 ครั้งนั้นมาหาค่าเฉลี่ยอีกรอบ ก็จะได้ผลลัพธ์ดังตาราง

| ครั้งที่ | KNN | MLP | SVM |
|---|---|---|---|
| 1 | 0.995 | 0.855 | 0.451 |
| 2 | 0.991 | 0.852 | 0.439 |
| 3 | 0.990 | 0.855 | 0.449 |
| 4 | 0.989 | 0.862 | 0.442 |
| 5 | 0.995 | 0.849 | 0.454 |
| ค่าเฉลี่ย | 0.992 | 0.854 | 0.447 |

ตารางที่ 4 ตารางแสดงค่าเฉลี่ยของ probability ของใบหน้าใน test data ในแต่ละ classifier

จากผลลัพธ์การทดลองที่ได้นั้น จะทำการเลือกโมเดลที่นำไปใช้บนเซิร์ฟเวอร์ โดยกำหนดจากโมเดลที่ได้ค่า probability ของใบหน้าใน test data ที่ดีที่สุด โดยโมเดลที่เหมาะสมที่สุดจะนำไปใช้บนเซิร์ฟเวอร์คือ KNN (K-Nearest Neighbors) โดยมีค่าความถูกต้องในการทำนายใบหน้าอยู่ที่ 98.7% และได้ค่าเฉลี่ยของ probability ของใบหน้าใน test data อยู่ที่ 99.2%

### 4.2 การพัฒนาส่วนแอปพลิเคชัน

ในส่วนนี้ได้ทำการพัฒนาหน้า user interface ส่วนหน้าเข้าสู่ระบบ และมีแบ่งส่วนของนักเรียนและครูผู้สอน

**นักเรียน :** หน้า user interface ของนักเรียนโดยมีปุ่มกดออกจากระบบ หน้าแสดงรายชื่อวิชา หน้าแสดงรายละเอียดการเช็คชื่อของรายวิชาของนักเรียน ส่วนตรวจจับใบหน้าจากกล้องก่อนส่งภาพไปที่ web server เพื่อทำนายใบหน้า

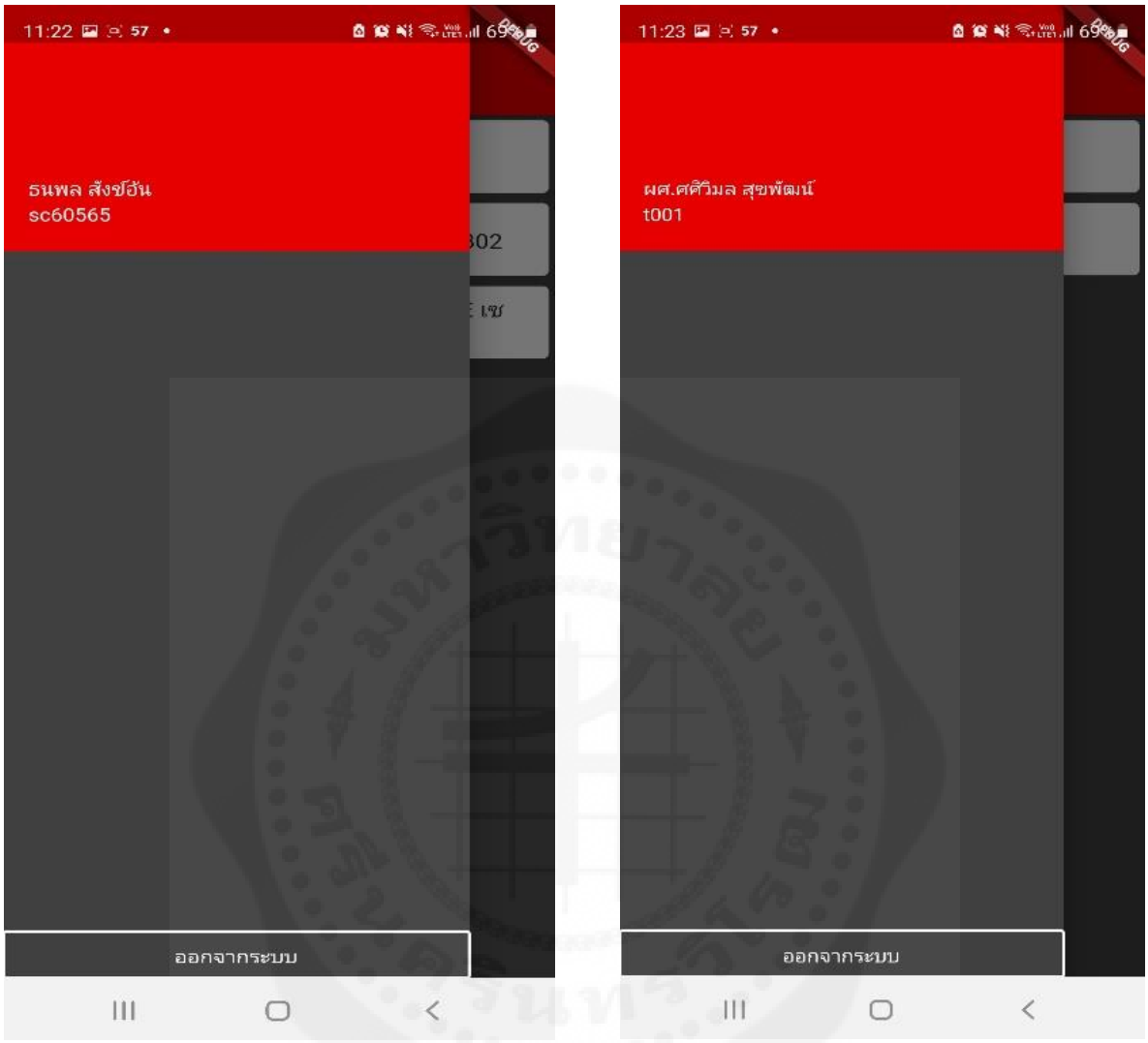
**ครูผู้สอน :** หน้า user interface ของครูผู้สอนโดยมีปุ่มกดออกจากระบบ แสดงรายละเอียดวิชาของครูผู้สอน โดยมีปุ่มกดเพิ่มครั้งที่ทำการสอน เพื่อให้นักเรียนสามารถเข้าเช็คชื่อได้ หน้าแสดงรายละเอียดการเข้าเช็คชื่อของนักเรียนในรายวิชา

โดยได้ทำการเชื่อมต่อกับฐานข้อมูล phpMyAdmin บนเซิร์ฟเวอร์ เพื่อแสดงรายละเอียดต่างๆในแต่ละหน้า user interface

4.2.1 หน้า user interface เข้าสู่ระบบ

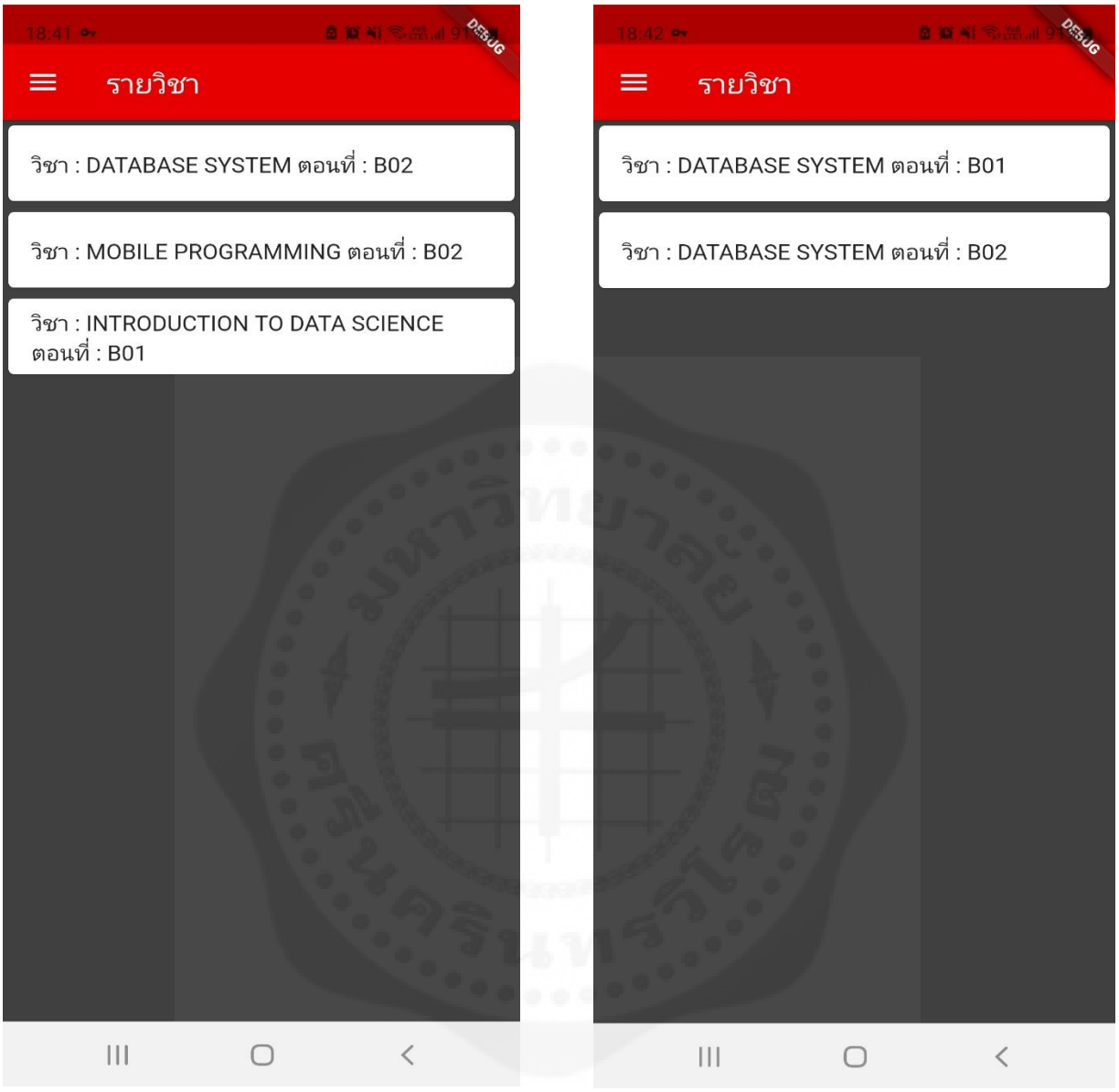
รูปภาพประกอบที่ 22 หน้า user interface เข้าสู่ระบบ

4.2.2 หน้า user interface ของนักเรียนและสำหรับครูผู้สอน โดยมีปุ่มกดออกจากระบบ




รูปภาพประกอบที่ 23 หน้า user interface ของนักเรียน(ซ้าย)และครูผู้สอน(ขวา) โดยมีปุ่มกดออกจากระบบ

4.2.3 หน้า user interface แสดงรายวิชาของนักเรียนและแสดงรายวิชาของที่สำหรับครูผู้สอน



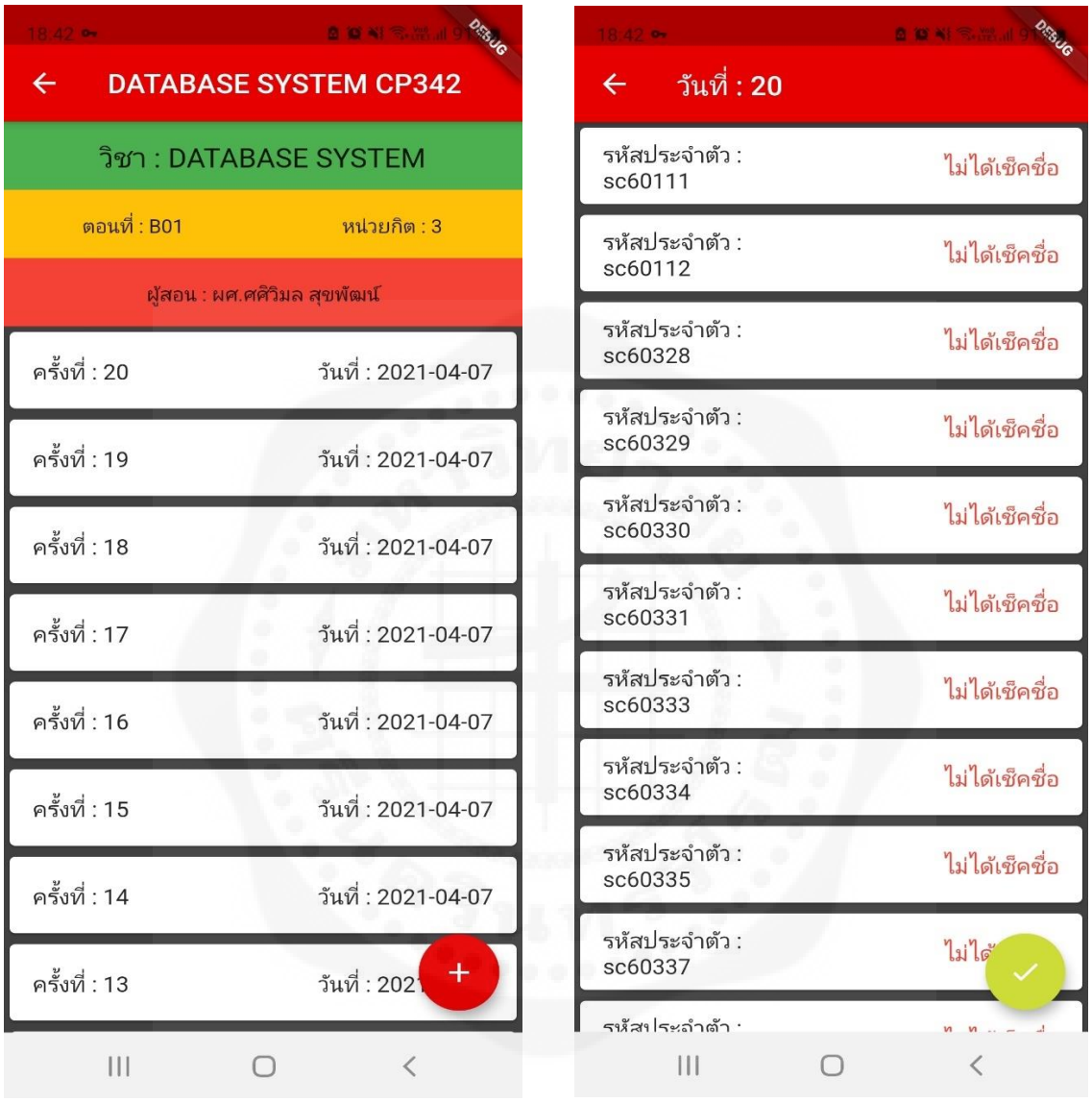

รูปภาพประกอบที่ 24 หน้า user interface แสดงรายวิชาของนักเรียน(ซ้าย)และแสดงรายวิชาของที่คุณครูสอน(ขวา)

4.2.4 หน้า user interface แสดงรายละเอียดการเข้าเช็คชื่อและสถานะการเช็คชื่อของนักเรียนในรายวิชา

รูปภาพประกอบที่ 25 หน้า user interface แสดงรายละเอียดการเข้าเช็คชื่อของนักเรียนในรายวิชา

4.2.5 หน้า user interface สำหรับแสดงรายละเอียดวิชาของครูผู้สอน โดยมีปุ่มกดเพิ่มครั้งที่ทำการสอน

เพื่อให้นักเรียนสามารถเข้าเช็คชื่อได้ และ หน้า user interface แสดงรายละเอียดการเข้าเช็คชื่อของนักเรียน

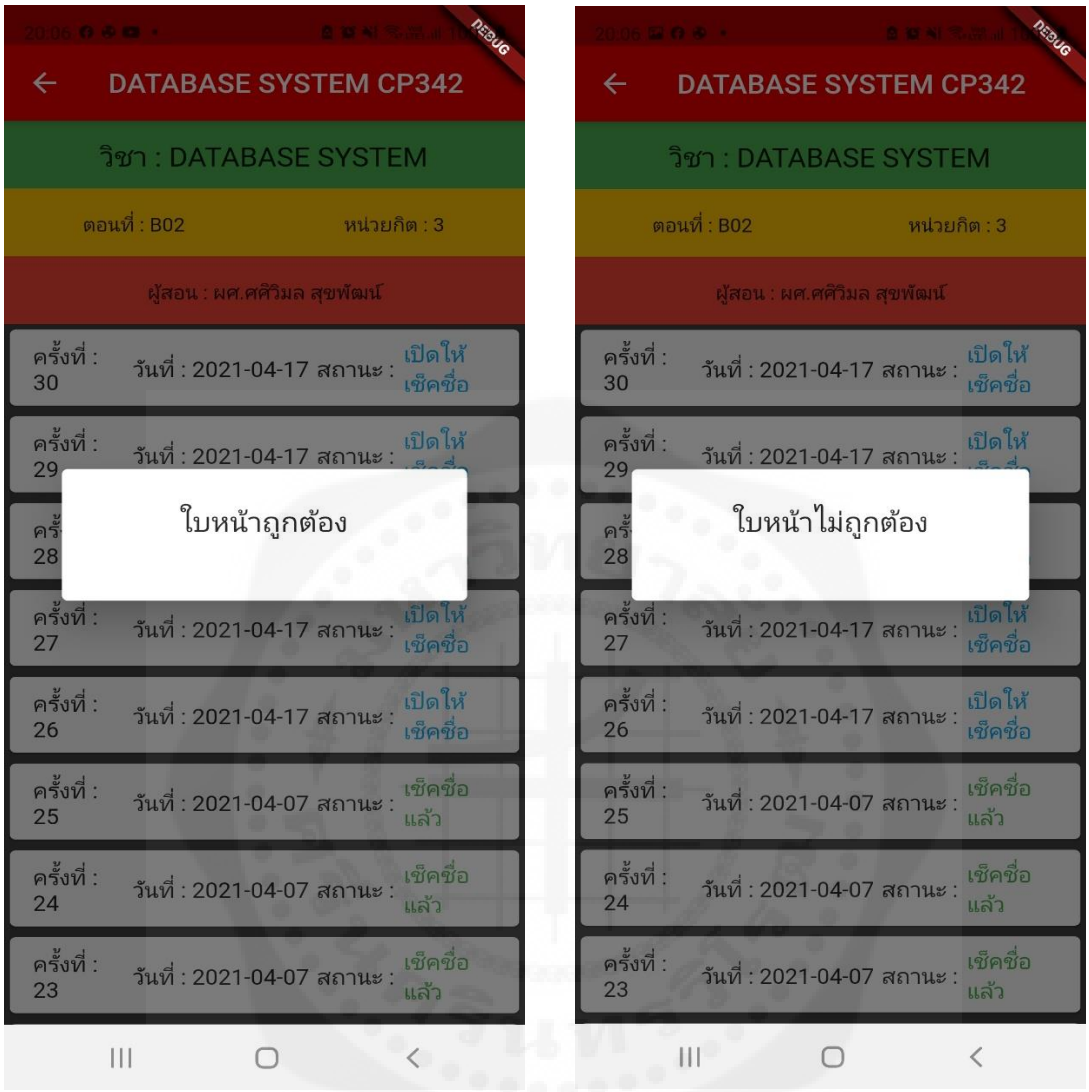



รูปภาพประกอบที่ 26 หน้า user interface แสดงรายวิชาของที่คุณครูสอน(ซ้าย)และหน้า user interface แสดงรายละเอียดการเข้าเช็คชื่อของนักเรียน

4.2.6 หน้า user interface การตรวจจับใบหน้าของนักเรียนก่อนส่งภาพไปที่ web server เพื่อทำนายใบหน้า

รูปภาพประกอบที่ 27 หน้า user interface การตรวจจับใบหน้าของนักเรียนก่อนส่งภาพไปที่ web server

4.2.6 หน้า user interface แสดงผลลัพธ์การทำนายใบหน้า




รูปภาพประกอบที่ 28 หน้า user interface แสดงผลลัพธ์การทำนายใบหน้า

## 4.3 ผลการทดลองทำนายใบหน้าและเช็คชื่อบนแอปพลิเคชันจริง

เมื่อได้โมเดลและแอปพลิเคชันที่พร้อมทำงานแล้ว กลุ่มนิสิตก็ได้ทำการทดลองทำนายใบหน้า โดยใช้แอปพลิเคชันที่พัฒนาขึ้นมา โดยให้นิสิต 5 คน ทำการเข้าสู่ระบบและใช้ใบหน้าในการเช็คชื่อ โดยผลลัพธ์ที่ได้มีดังนี้

ผลการทดลองเช็คชื่อบนแอปพลิเคชัน

| ใบหน้า | ชื่อของใบหน้า | ชื่อผู้ใช้ | ผลการทำนาย | ผลการเช็คชื่อ |
|---|---|---|---|---|
| | Thanaphon | Thanaphon | Thanaphon<br>0.903 | ผ่าน |
| | Ponlasit | Ponlasit | Ponlasit<br>0.884 | ผ่าน |
| | Nantich | Nantich | Nantich<br>0.910 | ผ่าน |
| | Pulipach | Pulipach | Pulipach<br>0.887 | ผ่าน |
| | Tanyatorn | Panotporn | Tanyatorn<br>0.898 | ไม่ผ่าน เนื่องจากใบหน้าที่ใช้ทำนายและชื่อที่ทำนายได้ไม่ตรงกับชื่อผู้ใช้ |

ตารางที่ 5 ตารางแสดงผลการทดลองเช็คชื่อบนแอปพลิเคชัน

เพื่อวัดประสิทธิภาพเวลาที่ใช้ในการทำนายใบหน้าบนเซิร์ฟเวอร์ว่าจะใช้เวลาในการประมวลผลเท่าไหร่และสามารถประมวลผลพร้อมกันได้กี่ใบหน้า จึงได้ทำการทดลอง โดยส่งใบหน้าขึ้นไปบนเซิร์ฟเวอร์ในช่วงเวลาที่พร้อมกันและใกล้เคียงกันมากๆ และได้ผลลัพธ์การทดลองดังนี้

| จำนวนใบหน้า | ค่าเฉลี่ยเวลาที่ใช้ในการประมวลผลของแต่ละใบหน้า (หน่วยวินาที) |
|---|---|
| 5 | 7.896526 |
| 10 | 16.528294 |
| 15 | 35.965245 |

ตารางที่ 6 ตารางแสดงผลการทดลองวัดประสิทธิภาพเวลาที่ใช้ในการทำนายใบหน้าบนเซิร์ฟเวอร์

โดยจากผลการทดลองสามารถสรุปได้ว่า เซิร์ฟเวอร์สามารถประมวลผลทำนายใบหน้า 15 ใบหน้าได้พร้อมกัน แต่ระยะเวลาที่ใช้ในการประมวลผลเพื่อทำนายใบหน้าก็จะเพิ่มขึ้นตามด้วย ซึ่งถ้าหากต้องการให้การประมวลผลสามารถดำเนินการได้เร็วเพิ่มขึ้น อาจจะต้องใช้เซิร์ฟเวอร์ที่มีทรัพยากรและสเปคของเครื่องเซิร์ฟเวอร์ที่สูงขึ้นกว่านี้

# บทที่5

# สรุปผลและอภิปรายผล ข้อเสนอแนะ

## 5.1 สรุปผลการดำเนินงาน

จากผลการดำเนินงาน ในส่วนของ Face Recognition นั้น ได้ผลลัพธ์ประสิทธิภาพของโมเดลออกมาดังนี้ โมเดลที่ได้จากการเทรน Classifier ด้วย KNN มีค่าความถูกต้องของการทำนายใบหน้าอยู่ที่ 98.7% โมเดลที่ได้จากการเทรน Classifier ด้วย MLP มีค่าความถูกต้องของการทำนายใบหน้าอยู่ที่ 96.7% และ โมเดลที่ได้จากการเทรน Classifier ด้วย SVM มีค่าความถูกต้องของการทำนายใบหน้าอยู่ที่ 98.7% โดยในการเลือกโมเดลที่เหมาะสมที่สุดที่จะนำไปใช้ทำนายใบหน้าบนเซิร์ฟเวอร์จะทำการเลือกจากการหาค่าเฉลี่ยของ probability ของใบหน้าใน test data ในแต่ละ classifier ที่ดีที่สุด โดยโมเดลที่ให้ผลลัพธ์ที่ดีที่สุดคือ KNN (K-Nearest Neighbors) โดยมีค่าความถูกต้องในการทำนายใบหน้าใน test data อยู่ที่ 98.7% และได้ค่าเฉลี่ยของ probability ของใบหน้าใน test data อยู่ที่ 99.2%

ในส่วนของแอปพลิเคชันได้ทำการพัฒนาหน้า user interface โดยเชื่อมต่อกับฐานข้อมูล phpMyAdmin เพื่อแสดงรายละเอียดส่วนหน้า user interface ต่างๆได้ โดยสามารถดึงข้อมูลมาแสดงได้ในหน้าแสดงรายชื่อวิชาของนักเรียน หน้าแสดงรายละเอียดการเช็คชื่อของรายวิชาของนักเรียน หน้าแสดงรายละเอียดวิชาของครูผู้สอน โดยมีปุ่มกดเพิ่มครั้งที่ทำการสอน เพื่อให้นักเรียนสามารถเข้าเช็คชื่อได้ หน้า user interface เข้าสู่ระบบ และส่วนตรวจจับใบหน้าจากกล้องก่อนส่งภาพไปที่ web server เพื่อทำนายใบหน้าและสามารถทำการบันทึกข้อมูลลงบนฐานข้อมูลบนเซิร์ฟเวอร์ได้

ในส่วนของเซิร์ฟเวอร์สามารถใช้งาน phpMyAdmin เป็นฐานข้อมูลได้ สามารถใช้งาน model บนเซิร์ฟเวอร์เพื่อทำนายใบหน้าและส่งผลการทำนายกลับไปที่แอปพลิเคชันได้

จากผลลัพธ์การดำเนินงานทั้งหมดนั้น แอปพลิเคชันที่พัฒนานั้น สามารถทำการถ่ายใบหน้าและส่งไปบนเซิร์ฟเวอร์เพื่อใช้โมเดลทำนายใบหน้าและส่งผลการทำนายใบหน้ากลับมาที่แอปพลิเคชันเพื่อยืนยันผลการทำนายใบหน้าและสามารถทำการเช็คชื่อเข้าเรียนได้ และจากผลการดำเนินงานทั้งหมด สามารถสรุปได้ว่าการพัฒนาระบบเช็คชื่อเข้าเรียนโดยใช้การรู้จำใบหน้าบนสมาร์ทโฟนนั้น สามารถแก้ปัญหาการใช้ทรัพยากรกระดาษในการเช็คชื่อและช่วยในการประหยัดเวลาในการเรียกชื่อเพื่อเช็คชื่อผู้เรียนได้ โดยที่ระบบสามารถทำการทำนายใบหน้าได้พร้อมกันหลายใบหน้า แต่ก็ขึ้นอยู่กับประสิทธิภาพของเครื่องเซิร์ฟเวอร์ด้วย ซึ่งถ้าเครื่องเซิร์ฟเวอร์มีทรัพยากรของเครื่องที่สูง ก็จะยิ่งสามารถประมวลผลการทำนายใบหน้าได้รวดเร็วยิ่งขึ้น

## 5.2 ปัญหาและอุปสรรค

เนื่องจากในช่วงที่ดำเนินการทำโครงงาน เกิดวิกฤตไวรัสโคโรน่า หรือ ไวรัสโควิด-19 ซึ่งทำให้เกิดความล่าช้าในการเก็บรวบรวมรูปภาพที่จะนำมาใช้ในการ train โมเดล จึงทำให้ใช้เวลานานจนกว่าจะได้ model ที่ให้ผลลัพธ์ที่ดีที่สุดในงายวิจัยนี้ และจากที่วางเป้าหมายไว้ วิกฤตไวรัสโคโรน่าทำให้มีปัญหาในการเก็บรวบรวมรูปภาพของนิสิตคณะวิทยาศาสตร์เพิ่มเติม เพื่อให้ได้รูปภาพที่มากกว่าเดิมและหลากหลายขึ้น เพื่อให้โมเดลสามารถเรียนรู้ภาพใบหน้าของนิสิตคณะวิทยาศาสตร์ได้หลากหลายและมากกว่าเดิม จึงไม่สามารถดำเนินการในส่วนนี้เพิ่มเติมได้

## 5.3 ข้อเสนอแนะและแนวทางในการพัฒนา

การพัฒนาระบบเช็คชื่อเข้าเรียนโดยใช้การรู้จำใบหน้าบนสมาร์ทโฟนนั้นสามารถแก้ปัญหาการใช้ทรัพยากรกระดาษในการเช็คชื่อและช่วยในการประหยัดเวลาในการเรียกชื่อเพื่อเช็คชื่อผู้เรียนได้ แต่ยังไม่สามารถแก้ปัญหาการเช็คชื่อแทนกันได้ โดยยังสามารถนำรูปภาพหรือวิดีโอที่มีใบหน้าของผู้ที่ต้องการเช็คชื่อ นำมาใช้แทนใบหน้าจริงๆของผู้ที่ต้องการเช็คชื่อได้อยู่ ดังนั้นจึงต้องศึกษาทฤษฎีและความรู้เกี่ยวกับการตรวจจับคลื่นโดเมนความถี่ของรูปภาพหรือวิดีโอเพิ่มเติมในการแก้ปัญหานี้

# บรรณานุกรม

[1] Electronic Frontier Foundation (EFF). (2017). **Face Recognition.** สืบค้นเมื่อ 11 มิถุนายน 2563. จาก https://www.eff.org/pages/face-recognition

[2] Rosa Gradilla. (2020). **Multi-task Cascaded Convolutional Networks (MTCNN) for Face Detection and Facial Landmark Alignment** สืบค้นเมื่อ 11 มิถุนายน 2563. จาก https://medium.com/@iselagradilla94/multi-task-cascaded-convolutional-networks-mtcnn-for-face-detection-and-facial-landmark-alignment-7c21e8007923

[3] Manning Publications. (2019). **The Computer Vision Pipeline, Part 3: image. preprocessing** สืบค้นเมื่อ 11 มิถุนายน 2563. จาก https://medium.com/@ManningBooks/the-computer-vision-pipeline-part-3 pipeline-part-3-image-preprocessing-c662094835ec

[4] Jaime Durán. (2019). **A Quick Guide on Preprocessing Facial Images for Neural Networks using OpenCV in Python.** สืบค้นเมื่อ 11 มิถุนายน 2563 จาก https:/ medium.com/yottabytes/a-quick-guide-on-preprocessing-facial-images-for-neural-networks-using-opencv-in-python-47ee3438abd4

[5] Athiwat. (2017). **Deep Learning คืออะไร.** สืบค้นเมื่อ 11 มิถุนายน 2563. จาก https://medium.com/machines-school/deep-leaning-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3-785e16d01773

[6] Natthawat Phongchit. (2018). **Convolutional Neural Network (CNN) คืออะไร.** สืบค้นเมื่อ 11 มิถุนายน 2563. จาก https://medium.com/@natthawatphongchit/%E0%B8%A1%E0%B8%B2%E0%B8%A5%E0%B8%AD%E0%B8%87%E0%B8%94%E0%B8%B9%E0%B8%A7%E0%B8%B4%E0%B8%98%E0%B8%B5%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B8%84%E0%B8%B4%E0%B8%94%E0%B8%82%E0%B8%AD%E0%B8%87-cnn-%E0%B8%81%E0%B8%B1%E0%B8%99-e3f5d73eebaa

[7] Dhairya Kumar. (2019). **Introduction to FaceNet: A Unified Embedding for Face Recognition and Clustering.** สืบค้นเมื่อ 12 มิถุนายน 2563. จาก https://medium.com/analytics-vidhya/introduction-to-facenet-a-unified-embedding-for-face-recognition-and-clustering-dbdac8e6f02

[8] PradyaSin. (2019). **Support Vector Machines (SVM).** สืบค้นเมื่อ 12 มิถุนายน 2563. จาก https://medium.com/@pradyasin/support-vector-machines-svm-943f9a732a69

[9] Unitech. (2018). สืบค้นเมื่อ 11 มิถุนายน 2563. **Multilayer Perceptron (MLP) vs Convolutional Neural Network in Deep Learning.** จาก https://medium.com/data-science-bootcamp/multilayer-perceptron-mlp-vs-convolutional-neural-network-in-deep-learning-c890f487a8f1

[10] Nati Thaiyathum. (2019). สืบค้นเมื่อ 11 มิถุนายน 2563. **KNN หรือ K-Nearest Neighbors คืออะไร.** จาก https://www.glurgeek.com/education/knn/

[11] Kishan Maladkar. (2018). สืบค้นเมื่อ 11 มิถุนายน 2563. **Why Is Random Search Better Than Grid Search For Machine Learning.** จาก https://analyticsindiamag.com/why-is-random-search-better-than-grid-search-for-machine-learning/

[12] Bana Handaga, Budi Murtiyasa, Jan Wantoro. (2019). **Attendance System based on Deep Learning Face Recognition without Queue.** สืบค้นเมื่อ 11 มิถุนายน 2563. จาก https://ieeexplore-ieee-org.clvpn.swu.ac.th/document/8985697

[13] Dongpeng Shang, Xinman Zhang, Xuebin Xu. (2018). สืบค้นเมื่อ 11 มิถุนายน 2563. **Face and Lip-Reading Authentication System Based on Android Smart Phones.** จาก https://ieeexplore-ieee-org.clvpn.swu.ac.th/document/8623298

[14] Shreyak Sawhney, Karan Kacker, Samyak Jain, Shailendra Narayan Singh, Rakesh Garg. (2019). สืบค้นเมื่อ 11 มิถุนายน 2563. **Real-Time Smart Attendance System using Face Recognition Techniques.** จาก https://ieeexplore-ieee-org.clvpn.swu.ac.th/document/8776934