

# ระบบแบ่งปันสูตรการทำอาหารและค้นหาสูตรการทำอาหารจากภาพวัตถุดิบ ด้วยเทคนิคการเรียนรู้เชิงลึก

นพรัตน์ มาน้อย อัมพล บุญจันดา และ ชูพันธุ์ รัตนโกคา\*

## บทคัดย่อ

ปัจจุบันการรับประทานอาหารเพื่อสุขภาพกำลังได้รับความนิยมเป็นอย่างมาก ผู้คนจึงเริ่มทำอาหารรับประทานกันด้วยตัวเองจากวัตถุดิบในการปรุงอาหารที่มีอยู่ ซึ่งบางครั้งไม่ทราบว่าสามารถนำไปทำอาหารอะไรได้บ้าง จึงทำได้แต่รายการอาหารเดิม ๆ ทำให้เกิดความจำเจในการรับประทานอาหาร และไม่สนุกกับการทำอาหาร บทความวิจัยนี้นำเสนอการออกแบบและพัฒนาระบบแบ่งปันสูตรการทำอาหาร และค้นหาสูตรการทำอาหารจากภาพวัตถุดิบด้วยเทคนิคการเรียนรู้เชิงลึก โดยใช้งานได้นับอุปกรณ์พกพา ซึ่งผู้ใช้งานสามารถแบ่งปันสูตรการทำอาหาร และสามารถใช้อุปกรณ์พกพาถ่ายรูปวัตถุดิบที่มีอยู่แล้วในครัว เช่น กระเทียม เนื้อหมู ผัก ฯลฯ เพื่อส่งภาพเข้ามาในระบบให้ค้นหาสูตรอาหารจากวัตถุดิบที่มีอยู่ ทำให้เพิ่มความสะดวกสบายในการค้นหาสูตรอาหารให้แก่ผู้ใช้ ส่วนประกอบหลักของระบบประกอบด้วย (1) แอปพลิเคชันบนอุปกรณ์พกพาสำหรับผู้ใช้งานทั่วไปที่พัฒนาด้วย React Native ซึ่งผู้ใช้งานสามารถเพิ่มสูตรการทำอาหาร และค้นหาสูตรการทำอาหารได้โดยใช้ชื่อวัตถุดิบ รวมถึงการใช้อุปกรณ์พกพาถ่ายภาพวัตถุดิบเพื่อค้นหาสูตรการทำอาหาร, (2) เว็บแอปพลิเคชันพัฒนาบน MERN stack สำหรับผู้ดูแลระบบ เพื่อใช้ในการเพิ่มค่าหลักในการค้นหาให้กับวัตถุดิบ รวมถึงทดสอบแบบจำลองที่ได้สร้างขึ้น และ (3) โครงข่ายประสาทเทียมการเรียนรู้เชิงลึกที่มีการใช้อัลกอริทึม YOLO ผ่านไลบรารี Darknet สำหรับการสร้างแบบจำลองในการรู้จำภาพแบบจำลองได้ฝึกสอนให้สามารถรู้จำวัตถุดิบได้จำนวน 20 ประเภท โดยสอนด้วยภาพวัตถุดิบประเภทละ 100 ภาพ หลังจากฝึกสอนแบบจำลองจำนวน 36,000 รอบ แบบจำลองมีค่าสูญเสียเฉลี่ยอยู่ที่ 0.0408 และมีค่า Precision, Recall และ F1-score อยู่ที่ 0.96, 0.98 และ 0.97 ตามลำดับ

**คำสำคัญ :** การเรียนรู้เชิงลึก, อัลกอริทึม YOLO, การรู้จำภาพ, การจำแนกหมวดหมู่รูปภาพ, แอปพลิเคชันบนอุปกรณ์พกพา

---

ภาควิชาเทคโนโลยีวิศวกรรมอิเล็กทรอนิกส์, วิทยาลัยเทคโนโลยีอุตสาหกรรม, มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

\* ผู้ติดต่อ, อีเมล: choopanr@kmutnb.ac.th รับเมื่อ 1 มิถุนายน 2562 ตอบรับ 5 สิงหาคม 2562

## A System for Cooking Recipe Sharing and Cooking Recipe Finding by an Image of Ingredients using Deep Learning Technique

Nopparat Manoi Ampol Bunjanda and Choopan Rattanapoka \*

### Abstract

Nowadays, healthy eating is very popular. People start to cook their own food from existing cooking ingredients. However, sometimes they do not know what food can be cooked from existing ingredients. Therefore, they cook the same food, resulting in monotonous eating and not enjoy cooking. This research article presents the design and development of a system for cooking recipe sharing and cooking recipe finding by an image of ingredients using deep learning techniques. Users can use the application on mobile devices to share cooking recipes. Moreover, users can take a picture of ingredients that users already have in the kitchen such as garlic, pork, vegetables, etc. and send that picture into the system to search for cooking recipes from existing ingredients. This process will make users convenient for searching cooking recipes. The main components of the system include (1) A mobile application for general users developed with React Native, which users can add cooking recipes and search for cooking recipes by entering the ingredient names. Also, the user can use the mobile device to take a picture of ingredients to find cooking recipes, (2) Web application developed on the MERN stack for system administrators, which system administrators can add keywords of the search term for ingredients and test the model that has been created, and (3) Deep convolutional neural network using the YOLO algorithm through the Darknet library for creating the image recognition model. The model has taught to be able to recognize 20 types of ingredients using 100 images of each type of ingredient. After training our model for 36,000 rounds, the model has an average loss of 0.0408 with the precision, recall and F1-score at 0.96, 0.98 and 0.97 respectively.

**Keywords :** Deep learning, YOLO algorithm, Image recognition, Image classification, Mobile application

---

Department of Electronic Engineering Technology, College of Industrial Technology, King Mongkut's University of Technology North Bangkok. \* Corresponding author, E-mail: choopanr@kmutnb.ac.th, Received: 1 June 2019, Accepted: 5 August 2019

## 1. บทนำ

Deep Learning หรือการเรียนรู้เชิงลึก ถือว่าเป็นหนึ่งในระบบการเรียนรู้ของเครื่อง (Machine Learning) ซึ่งมีโครงสร้างและการประมวลผลคล้ายกับสมองของมนุษย์ที่เรียกว่า โครงข่ายประสาทเทียม (Artificial Neural Network) โดย อัน ที่ จ ริ ง แล้ ว Deep Learning คือ โครงข่ายประสาทเทียมขนาดใหญ่ ซึ่งหมายถึงเป็นโครงข่ายประสาทเทียมที่มีชั้นในการประมวลผลหลายชั้น

ช่วงไม่กี่ปีที่ผ่านมาการเรียนรู้เชิงลึกได้รับความนิยมและมีงานวิจัยที่เกี่ยวข้องมากขึ้น เช่น ใน [1] ได้พัฒนาระบบการแนะนำ (Recommendation system) ด้วยการใช้เทคนิคการเรียนรู้เชิงลึก หรือแม้กระทั่งใน [2] ได้มีการใช้เทคนิคการเรียนรู้เชิงลึกในการพัฒนาระบบตรวจจับผู้บุกรุก (Intrusion Detection System) แต่ด้านการใช้งานที่ทำให้เทคนิคการเรียนรู้เชิงลึกได้รับความนิยมมากมาจากงานด้านรู้จำภาพ โดยเป็นการใช้โครงข่ายประสาทเทียมประเภท Convolutional Neural Network ดังอธิบายใน [3] และมีบทความวิจัยที่ศึกษาและประยุกต์ใช้การรู้จำภาพอย่างมากมาย เช่น ใน [4] เป็นโครงข่ายประสาทเทียมสำหรับตรวจจับภาพอาหาร และ [5], [6] และ [7] เป็น โครงข่ายประสาทเทียมสำหรับตรวจจับอารมณ์ในภาพ

ในปัจจุบันแนวโน้มการรับประทานอาหารเพื่อสุขภาพกำลังได้รับความนิยมเป็นอย่างมาก เนื่องจากโรคประจำตัวต่างๆ ซึ่งส่วนใหญ่แล้วมักเกิดจากการรับประทานอาหาร ผู้คนส่วนใหญ่จึงเริ่มทำอาหารรับประทานกันด้วยตัวเอง แต่สำหรับผู้เริ่มต้นทำอาหารอาจจะมีความลำบากและยุ่งยากในการค้นหาสูตรการทำอาหารจากหนังสือทำอาหาร หรือจากอินเทอร์เน็ต อีกทั้งบางครั้งมีวัตถุดิบในการปรุงอาหารอยู่ที่บ้าน แต่

คิดไม่ออกว่าสามารถนำไปทำอาหารอะไรได้บ้าง จึงทำได้แต่รายการอาหารเดิม ๆ จึงเกิดความจำเจในการรับประทานอาหาร และไม่สนุกกับการทำอาหาร

ดังนั้นบทความวิจัยนี้จึงต้องการนำเสนอการออกแบบและพัฒนาระบบแบ่งปันสูตรการทำอาหารและค้นหาสูตรการทำอาหารจากภาพวัตถุดิบด้วยเทคนิคการเรียนรู้เชิงลึก โดยใช้งานได้นับอุปกรณ์พกพา ซึ่งผู้ใช้งานสามารถแบ่งปันสูตรการทำอาหาร และสามารถใช้อุปกรณ์พกพาถ่ายรูปวัตถุดิบที่ผู้ใช้มีอยู่แล้วในครัว เช่น กระเทียม เนื้อหมู ผัก ฯลฯ เพื่อส่งภาพเข้ามาในระบบให้ค้นหาสูตรอาหารจากวัตถุดิบที่มีอยู่ เพื่อเพิ่มความสะดวกสบายในการค้นหาสูตรอาหารให้แก่ผู้ใช้

## 2. ทฤษฎีพื้นฐานและงานวิจัยที่เกี่ยวข้อง

### 2.1 MERN Stack

MERN Stack เป็นเฟรมเวิร์คที่ถูกออกแบบมาเพื่อเป็นตัวช่วยในการพัฒนาเว็บไซต์และรวมถึงเว็บแอปพลิเคชัน MERN Stack ประกอบด้วย 4 ส่วนการทำงาน คือ MongoDB, Express JS, React JS และ Node JS โดยมี React JS ทำงานในส่วนของไคลเอนต์ ในขณะที่ Express, Node JS และ MongoDB ทำงานที่ในส่วนของผู้เซิร์ฟเวอร์

MongoDB [8] เป็นระบบฐานข้อมูลเชิงเอกสาร ซึ่งได้ถูกจำแนกให้เป็นระบบฐานข้อมูลประเภท NoSQL มีการเก็บข้อมูลเป็นในรูปแบบเอกสารที่มีโครงสร้างคล้ายกับเอกสาร JSON การบันทึกข้อมูลทุก ๆ Record ใน MongoDB จะเรียกว่า Document ซึ่งจะเก็บค่าเป็น Key และ Value และการเก็บข้อมูล Document ใน MongoDB จะถูกเก็บไว้ใน Collections

Express JS คือ Web Application Framework ที่ได้รับความนิยมมาก สำหรับทำงานบนแพลตฟอร์มของ Node JS ซึ่งทำหน้าที่เซิร์ฟเวอร์ โดยทั้ง Express JS และ Node JS ต่างก็ใช้ภาษา JavaScript ในการพัฒนา

React JS [9] คือ JavaScript Library ทำหน้าที่เป็น View ที่มาจากพื้นฐานแนวความคิดแบบ MVC (Model View Controller) รองรับการทำงานด้วย JSX (JavaScript Syntax Extension) โดยภายในเว็บจะมองส่วนต่าง ๆ เป็น Component และข้อมูลที่อยู่ใน Component แต่ละชั้น จะเรียกว่า State ส่วน ข้อมูลที่ถูกส่งต่อจาก Component ชั้นบนลง ไปชั้นล่าง เรียกว่า Props

Node JS [10] คือสภาพแวดล้อมการทำงานแบบ Cross-Platform สำหรับฝั่งเซิร์ฟเวอร์ ซึ่งถูกพัฒนาขึ้นด้วยภาษา JavaScript โดยปกติแล้ว JavaScript จะทำงานในส่วน ของฝั่งไคลเอนต์ ซึ่งจะทำงานในเชิงโต้ตอบกับเว็บเบราว์เซอร์ จากนั้น ได้มีการพัฒนา Node JS เพื่อมาเป็น Runtime ในส่วนของฝั่งเซิร์ฟเวอร์ โดย Node JS จะใช้รูปแบบ Event-driven, Non-blocking I/O ทำให้ใช้ทรัพยากรระบบน้อย มีประสิทธิภาพ และสามารถประมวลผลได้อย่างรวดเร็ว

## 2.2 React Native

React Native [11] คือ เครื่องมือที่ใช้สร้างแอปพลิเคชัน สำหรับอุปกรณ์พกพา (Mobile Application) ทั้ง iOS และ Android มีรูปแบบการทำงานเป็น Cross Platform Technology โดยใช้ JavaScript เป็นหลักในการพัฒนา ซึ่ง React Native ถูกสร้างขึ้นโดยทีมงาน Facebook เป็นซอฟต์แวร์เปิด และมีชุมชนผู้ใช้งานที่กว้าง มีนักพัฒนาหลายคนช่วยทำและแจกจ่าย

ไลบรารีออกมาให้ใช้ได้กันอย่างไม่มีค่าใช้จ่ายอีกด้วย จึงเป็นเครื่องมือตัวหนึ่งที่เป็นที่นิยมมากในปัจจุบัน

## 2.3 YOLO (You Only Look Once)

YOLO (You Only Look Once) [12] เป็นอัลกอริธึมที่นำแนวความคิดของการทำนายตำแหน่งและขนาดของกล่อง จากความน่าจะเป็นที่กล่องนั้น จะเป็นกรอบล้อมวัตถุ แต่สิ่งที่ YOLO มีความสามารถและความเร็วเหนือกว่าอัลกอริธึมอื่น เช่น Faster R-CNN [13] ที่มีการทำงานในลักษณะที่จะทำนายตำแหน่งของกรอบล้อมวัตถุและค่อยนำวัตถุในกล่องนั้น ไปผ่านแบบจำลองเพื่อทำนายวัตถุในกล่อง แต่ YOLO นั้นจะทำนายทั้งกรอบล้อมวัตถุ และความน่าจะเป็นของวัตถุบางส่วนที่อยู่ในกรอบออกมาพร้อมกันทีเดียว

YOLO จัดว่าเป็นเทคนิคการตรวจจับวัตถุในภาพซึ่งเป็นซอฟต์แวร์เปิดสำหรับงานปัญญาประดิษฐ์แบบโครงข่ายประสาท (Neural Network) ที่พัฒนาด้วยโปรแกรมภาษา C++ และสามารถทำงานบนหน่วยประมวลผล CUDA ของ GPU ได้เป็นอย่างดี เหมาะกับการประมวลผลภาพแบบ Real Time ภาพจากกล้องหรือวิดีโอ

ปัจจุบัน YOLO มีการพัฒนามาแล้ว 3 เวอร์ชัน YOLO, YOLO v2 [14] และปัจจุบันคือ YOLO v3 [15] โครงสร้างโครงข่ายประสาทเทียมของ YOLO แต่ละเวอร์ชัน จะมี Convolution Box ที่แตกต่างกัน แต่หลักการโดยทั่วไปแล้ว YOLO จะแบ่งภาพออกเป็น Grid Cell เล็ก ๆ และแต่ละ Grid Cell จะถูกทำนายผ่านแบบจำลอง เพื่อหาตำแหน่งจุดกึ่งกลางของวัตถุ และความน่าจะเป็นที่จะมีวัตถุใด ใน Grid Cell

### 2.4 งานวิจัยที่เกี่ยวข้อง

การรับประทานอาหารเช้าเพื่อสุขภาพได้รับความนิยมเป็นอย่างมากในปัจจุบัน ทำให้ผู้คนเริ่มทำอาหารเพื่อรับประทานกันเอง ทำให้มีเว็บไซต์และแอปพลิเคชันที่เกี่ยวข้องกับการสร้างและแบ่งปันสูตรการทำอาหารถูกพัฒนาขึ้นและให้บริการกันอย่างแพร่หลาย เช่น เว็บไซต์ allrecipes.com, foodnetwork.com และ delish.com และแอปพลิเคชันบนอุปกรณ์พกพา เช่น Tasty, Cookpad และ Yummly Recipes & Shopping List ที่มีผู้ใช้งานประมาณ 300,000 คน และได้รับเลือกเป็นแอปพลิเคชัน Editor’s Choice บนระบบปฏิบัติการแอนดรอยด์

ในขณะที่การค้นหาสูตรและขั้นตอนการทำอาหารด้วยข้อความนั้นได้รับความนิยม การจำแนกประเภทจากภาพก็เป็นหัวข้อที่ได้รับความนิยมเป็นอย่างมาก โดยมีการประยุกต์ใช้เทคนิคการเรียนรู้เชิงลึก ประเภทโครงข่ายประสาทเทียม Convolutional neural network เช่น DeepFood [16] ได้นำเสนอการจำแนกวัตถุดิบในการทำอาหารจากภาพ โดยใช้เทคนิคการเรียนรู้เชิงลึก ประเภทโครงข่ายประสาทเทียม Convolutional neural network โดยแบบจำลองเรียนรู้วัตถุดิบจำนวน 41 ประเภท จากภาพวัตถุดิบประเภทละ 100 ภาพ ซึ่งความแม่นยำของแบบจำลองประเภทต่าง ๆ เฉลี่ยอยู่ที่ประมาณ 80% และใน [17] ได้นำเทคนิคการเรียนรู้เชิงลึกประเภทโครงข่ายประสาทเทียม Convolutional neural network ที่เรียกว่า GooLeNet มาใช้ในการทำนายชื่ออาหารไทยจากภาพ ซึ่งแบบจำลองเรียนรู้ประเภทของอาหารไทยไว้ทั้งหมด 11 ประเภท โดยมีความแม่นยำของแบบจำลองอยู่ที่ 88.33%

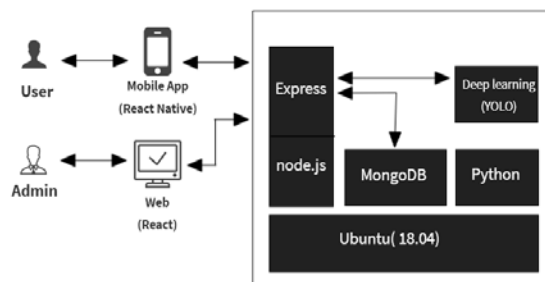
### 3. วิธีการวิจัย

ระบบค้นหาสูตรการทำอาหารจากภาพวัตถุดิบด้วยเทคนิคการเรียนรู้เชิงลึก ได้ถูกออกแบบโดยมีสถาปัตยกรรมดังรูปที่ 1 ซึ่งผู้ใช้งานระบบมี 2 ประเภท

(1) ผู้ใช้งานทั่วไป ใช้งานระบบผ่านทางแอปพลิเคชันบนอุปกรณ์พกพาที่พัฒนาด้วย React Native ซึ่งผู้ใช้งานสามารถเพิ่มสูตรการทำอาหาร และค้นหาสูตรการทำอาหารได้โดยใช้ชื่อวัตถุดิบ รวมถึงการใช้อุปกรณ์พกพาท่าถ่ายภาพวัตถุดิบเพื่อค้นหาสูตรการทำอาหาร

(2) ผู้ดูแลระบบ ใช้งานระบบผ่านเว็บแอปพลิเคชันที่พัฒนาด้วย React JS ซึ่งผู้ดูแลระบบจะสามารถเพิ่มคำหลักของวัตถุดิบแต่ละประเภท และสามารถส่งรูปภาพเข้าไปเพื่อทดสอบแบบจำลองได้

ทั้งแอปพลิเคชันบนอุปกรณ์พกพา และเว็บแอปพลิเคชันจะเชื่อมต่อไปยัง Express JS ผ่านทาง REST API เพื่อเขียน/อ่านข้อมูลในระบบฐานข้อมูล MongoDB แต่ถ้ามีการค้นหาสูตรการทำอาหารโดยใช้ภาพวัตถุดิบ Express JS จะเรียกใช้แบบจำลองการเรียนรู้เชิงลึกเพื่อค้นหาชื่อวัตถุดิบทั้งหมดที่มีอยู่ในภาพ และนำชื่อวัตถุดิบที่ตรวจพบ มาค้นหาสูตรการทำอาหารและคืนค่ากลับไปให้ผู้ใช้



รูปที่ 1 สถาปัตยกรรมโดยรวมของงานวิจัย

ในการพัฒนาระบบ จะประกอบไปด้วยขั้นตอนหลัก 6 ขั้นตอน ดังนี้

**3.1 การเลือกวัตถุดิบสำหรับสร้างแบบจำลอง**

ขอบเขตงานวิจัยต้องการสร้างแบบจำลองเพื่อรู้จำภาพวัตถุดิบในการประกอบอาหารเบื้องต้นทั้งหมด 20 ชนิด โดยได้นำภาพของวัตถุดิบแต่ละประเภทจาก Search Engine ประเภทละไม่ต่ำกว่า 100 ภาพ โดยแต่ละวัตถุดิบได้ถูกกำหนดฉลาก (Label) ดังแสดงในตารางที่ 1

**3.2 การสร้างและฝึกแบบจำลอง**

งานวิจัยนี้ได้เลือกใช้ไลบรารี Darknet [18] ที่เป็นไลบรารีสำหรับการสร้างแบบจำลอง ที่มีการใช้อัลกอริทึม YOLO ในการค้นหาและรู้จำภาพ เนื่องจากได้กำหนดให้ระบบรู้จำวัตถุดิบได้ 20 ประเภท แบบจำลองนี้จึงแบ่งออกเป็น 20 คลาส (Class) และแบ่งภาพของวัตถุดิบทั้งหมดออกเป็นภาพสำหรับการฝึก 80% และภาพสำหรับการทดสอบ 20%

ไลบรารี Darknet มีแอปพลิเคชันสำหรับการฝึกสอนแบบจำลองให้อยู่แล้ว ด้วยการใส่คำสั่ง

```
darknet_detector_train <ไฟล์ข้อมูล> <ไฟล์ปรับแต่ง> <ไฟล์ค่าน้ำหนักเริ่มต้น>
```

โดยภายใน “ไฟล์ข้อมูล” ประกอบไปด้วยข้อมูลจำนวนคลาส ชื่อไฟล์รวบรวมชื่อภาพสำหรับการฝึก ชื่อไฟล์รวบรวมชื่อภาพสำหรับการทดสอบ ชื่อไฟล์ที่เก็บหมายเลขและชื่อฉลาก และชื่อไดเรกทอรีสำหรับเก็บค่าน้ำหนักของโครงข่ายประสาทเทียมระหว่างการฝึกสอน

ภายใน “ไฟล์ปรับแต่ง” จะเก็บข้อมูลโครงสร้างและพารามิเตอร์ต่าง ๆ ของโครงข่ายประสาทเทียม รวมถึงค่าพารามิเตอร์ต่าง ๆ ที่จำเป็นสำหรับ YOLO เช่น ขนาดของ Grid Cell

สำหรับ “ไฟล์ค่าน้ำหนักเริ่มต้น” ผู้พัฒนาสามารถดาวน์โหลดมาใช้ได้เลย เพื่อให้การเรียนรู้ของโครงข่ายประสาทเทียมรวดเร็วขึ้น

**ตารางที่ 1** รายชื่อฉลากและคำอธิบายวัตถุดิบ

ลำดับที่	ชื่อฉลาก	คำอธิบาย
1	Egg	ไข่
2	Cabbage	กะหล่ำปลี
3	Banana	กล้วย
4	Purple cabbage	กะหล่ำปลีม่วง
5	Chicken	ไก่
6	Carrot	แครอท
7	Chili	พริก
8	Pork	หมู
9	Apple	แอปเปิ้ล
10	Fish	ปลา
11	Corn	ข้าวโพด
12	Garlic	กระเทียม
13	Onion	หัวหอม
14	Lemon	มะนาว
15	Potato	มันฝรั่ง
16	Shrimp	กุ้ง
17	Chinese cabbage	ผักกาดขาว
18	Crab	ปู
19	Tomato	มะเขือเทศ
20	Pineapple	สับปะรด

**ตารางที่ 2** โครงสร้างแบบจำลอง Tiny YOLOv3

ชั้นที่	ชื่อชั้น	พารามิเตอร์	ขนาดผลลัพธ์
0	conv	3 x 3 / 1 (16)	416 x 416 x 16
1	max	2 x 2 / 2	208 x 208 x 16
2	conv	3 x 3 / 1 (32)	208 x 208 x 32
3	max	2 x 2 / 2	104 x 104 x 32
4	conv	3 x 3 / 1 (64)	104 x 104 x 64
5	max	2 x 2 / 2	52 x 52 x 64
6	conv	3 x 3 / 1 (128)	52 x 52 x 128
7	max	2 x 2 / 2	26 x 26 x 128
8	conv	3 x 3 / 1 (256)	26 x 26 x 256
9	max	2 x 2 / 2	13 x 13 x 256
10	conv	3 x 3 / 1 (512)	13 x 13 x 512
11	max	2 x 2 / 1	13 x 13 x 512
12	conv	3 x 3 / 1 (1024)	13 x 13 x 1024
13	conv	3 x 3 / 1 (256)	13 x 13 x 256
14	conv	3 x 3 / 1 (512)	13 x 13 x 512
15	conv	3 x 3 / 1 (75)	13 x 13 x 75
16	yolo		
17	route	13	13 x 13 x 256
18	conv	1 x 1 / 1 (128)	13 x 13 x 128
19	upsample	2x	26 x 26 x 128
20	route	19 8	26 x 26 x 384
21	conv	3 x 3 / 1 (256)	26 x 26 x 256
22	conv	1 x 1 / 1 (75)	26 x 26 x 75
23	yolo		

เนื่องจากแบบจำลอง YOLOv3 ให้ความแม่นยำในการทำงานได้ดีกว่า YOLOv2 แต่ก็มีการทำงานที่ช้ากว่า

ดังนั้นงานวิจัยนี้ได้เลือกใช้แบบจำลองชื่อ Tiny YOLOv3 ซึ่งเป็นแบบจำลองที่มีจำนวนชั้นในโครงสร้างน้อยกว่า YOLOv3 ปกติ แต่มีความเร็วในการทำงานมาก ซึ่งเหมาะสมกับรูปแบบการทำงานของระบบที่ออกแบบมากกว่า

โครงสร้างของแบบจำลอง Tiny YOLOv3 ประกอบด้วยชั้นการทำงานทั้งหมด 23 ชั้นดังตารางที่ 2 โดยภาพที่ส่งเข้าแบบจำลองมีขนาด 416 x 416 พิกเซล ส่วนประกอบของแต่ละชั้นประกอบด้วยหมายเลขชั้นชื่อของการทำงานในชั้น พารามิเตอร์ของชั้น และขนาดผลลัพธ์ของชั้นนั้น

พารามิเตอร์ของชั้น conv อยู่ในรูปแบบ  $f \times f / s (n)$  โดย  $f$  คือ ความกว้างและความยาวของฟิลเตอร์ (filter size),  $s$  คือจำนวนพิกเซลในการเลื่อนฟิลเตอร์ (stride) และ  $n$  คือ จำนวนฟิลเตอร์ที่ใช้ในชั้น ตัวอย่างชั้นที่ 0 เป็นชั้น conv ที่มีพารามิเตอร์คือ  $3 \times 3 / 1 (16)$  หมายถึงในชั้นนี้ใช้ฟิลเตอร์ขนาด  $3 \times 3$  พิกเซล จำนวนทั้งหมด 16 ฟิลเตอร์ และเลื่อนตำแหน่งฟิลเตอร์ทีละ 1 พิกเซล

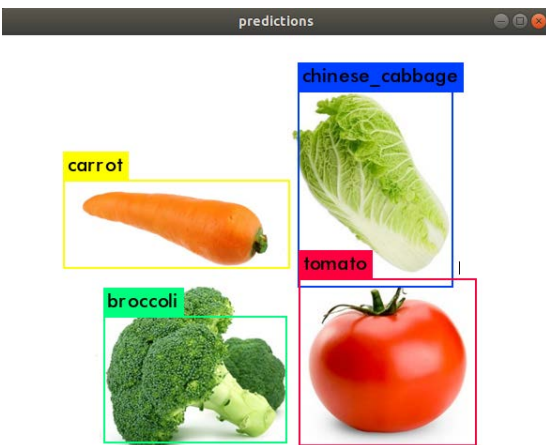
พารามิเตอร์ของชั้น max อยู่ในรูปแบบ  $f \times f / s$  โดย  $f$  คือ ความกว้างและความยาวของฟิลเตอร์ (filter size) และ  $s$  คือจำนวนพิกเซลในการเลื่อนฟิลเตอร์ (stride) ตัวอย่างชั้นที่ 1 เป็นชั้น max ที่มีพารามิเตอร์คือ  $2 \times 2 / 2$  หมายถึง ในชั้นนี้ใช้ฟิลเตอร์ขนาด  $2 \times 2$  พิกเซล และเลื่อนตำแหน่งฟิลเตอร์ทีละ 2 พิกเซล

พารามิเตอร์ของชั้น route อยู่ในรูปแบบตัวเลขที่หมายถึงหมายเลขชั้น เช่น ชั้นที่ 17 มีพารามิเตอร์คือ 13 ซึ่งหมายถึง ให้นำผลลัพธ์ที่เกิดจากชั้นที่ 13 มาเป็นผลลัพธ์ให้กับชั้นที่ 17 และสำหรับชั้นที่ 20 ที่มีพารามิเตอร์คือ 19 8 หมายถึงให้นำผลลัพธ์จากชั้นที่ 19 และ 8 มาต่อกัน เพื่อเป็นผลลัพธ์ให้กับชั้นที่ 20 ซึ่งจะ

เห็นได้ว่าผลลัพธ์ของชั้นที่ 19 มีขนาด 26 x 26 x 128 และผลลัพธ์ของชั้นที่ 8 มีขนาด 26 x 26 x 256 เมื่อนำมารวมกันทำให้ผลลัพธ์ของชั้นที่ 20 มีขนาด 26 x 26 x 384 พารามิเตอร์ของชั้น Upsample คือจำนวนเท่าของการขยายผลลัพธ์จากข้อมูลเข้า ตัวอย่างในชั้นที่ 19 ที่มีพารามิเตอร์คือ 2x หมายถึงให้สร้างผลลัพธ์ที่มีขนาดเป็น 2 เท่าของข้อมูลเข้า ซึ่งจะเห็นได้ว่าข้อมูลเข้าของชั้นที่ 19 (ผลลัพธ์ของชั้นที่ 18) มีขนาด 13 x 13 x 128 เมื่อขยาย 2 เท่าทำให้ผลลัพธ์ที่ออกจากชั้นที่ 19 มีขนาด 26 x 26 x 128

```
Total BFLOPS 21.909
Allocate additional workspace_size = 12.46 MB
Loading weights from backup/yolov3-tiny_obj_100000.weights...
seen 64
Done!
data/m6.png: Predicted in 151.946000 milli-seconds.
carrot: 96%
broccoli: 99%
chinese_cabbage: 100%
tomato: 100%
```

รูปที่ 2 ผลการจากทดสอบแบบจำลองที่แสดงค่าความเชื่อมั่น



รูปที่ 3 ภาพผลลัพธ์จากการทดสอบแบบจำลองโดยจะมีการตีกรอบและติดฉลากให้กับวัตถุ

สำหรับการทดสอบแบบจำลองนั้น Darknet เองก็มีแอปพลิเคชันช่วยเหลือในการทดสอบ โดยการใช้คำสั่ง `darknet detector test <ไฟล์ข้อมูล> <ไฟล์ปรับแต่ง> <ไฟล์ค่าน้ำหนักที่ต้องการให้กับแบบจำลอง>`

ซึ่งเมื่อเรียกใช้คำสั่ง แอปพลิเคชันจะแสดงค่าความเชื่อมั่นของวัตถุที่ตรวจพบ มากไปกว่านั้นยังแสดงภาพที่ส่งไปตรวจสอบ พร้อมกับตีกรอบ และเขียนฉลากกำกับวัตถุที่ค้นพบอีกด้วย ดังรูปที่ 2 และรูปที่ 3

### 3.3 สูตรการทำอาหารและวัตถุดิบที่ใช้

เพื่อทดสอบการทำงานของระบบ จึงได้จัดทำสูตรการทำอาหารและวัตถุดิบที่ต้องใช้ ให้กับระบบเบื้องต้นไว้ก่อนทั้งหมด 50 รายการ ยกตัวอย่าง เช่น

- ไข่เจียวโซเซ ใช้วัตถุดิบคือไข่ไก่
- ลาบหมู ใช้วัตถุดิบ คือ หมู มะนาว ต้นหอม กระเทียม ผักชี และหอมแดง
- กุ้งผัดบล็อกโคลี่ ใช้วัตถุดิบ บล็อกโคลี่ กุ้ง แครอท และกระเทียม เป็นต้น

### 3.4 การกำหนดคำหลักให้กับวัตถุดิบ

เนื่องจากแบบจำลองการเรียนรู้เชิงลึกที่ออกแบบจะสามารถรู้จำภาพวัตถุดิบได้ 20 ประเภท โดยผลลัพธ์ของแบบจำลองคือชื่อฉลาก แต่การทำสูตรการทำอาหารนั้นชื่อวัตถุดิบ เช่น ไข่ และไข่ไก่ มีความหมายเหมือนกันคือ Egg ดังนั้นเพื่อสืบค้นสูตรการทำอาหารจากวัตถุดิบที่ตรวจพบในภาพ จึงต้องมีกำหนดคำหลักให้กับฉลากแต่ละอัน โดยเบื้องต้นได้กำหนดคำหลักให้กับฉลากแต่ละประเภทดังตารางที่ 3 แต่อย่างไรก็ตามผู้ดูแลระบบสามารถเพิ่มคำหลักให้กับฉลากได้ผ่านทางเว็บแอปพลิเคชัน



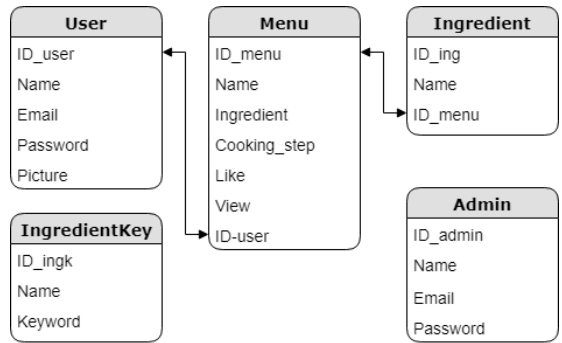
ตารางที่ 3 รายชื่อฉลาดและคำหลักสำหรับค้นหา

ลำดับที่	ชื่อฉลาด	คำหลัก
1	Egg	ไข่, ไข่ไก่, ไข่ต้ม
2	Cabbage	ผักกะหล่ำปลี, กะหล่ำปลี
3	Banana	กล้วย
4	Cabbage purple	ผักกะหล่ำปลีม่วง
5	Chicken	ไก่, เนื้อไก่
6	Carrot	แครอท
7	Chili	พริก, พริกสด
8	Pork	หมู, เนื้อหมู, เนื้อหมูสันใน
9	Apple	แอปเปิ้ล, แอปเปิล
10	Fish	ปลา
11	Corn	ข้าวโพด
12	Garlic	กระเทียม
13	Onion	หัวหอม
14	Lemon	มะนาว
15	Potato	มันฝรั่ง
16	Shrimp	กุ้ง, กุ้งแห้ง
17	Chinese cabbage	ผักกาดขาว
18	Crab	ปู, ปูนา, ปูเค็ม
19	Tomato	มะเขือเทศ
20	Pineapple	สับปะรด

3.5 การออกแบบฐานข้อมูล

ระบบใช้ฐานข้อมูล MongoDB ในเก็บข้อมูล ซึ่งประกอบไปด้วย 5 Collections ดังแสดงในรูปที่ 4

- User ใช้เก็บข้อมูลของผู้ใช้งานระบบ



รูปที่ 4 โครงสร้างฐานข้อมูล

- IngredientKey ใช้สำหรับเก็บคำหลัก เพื่อเป็นคำสืบค้นให้กับวัตถุดิบหลักทั้ง 20 ประเภท
- Menu ใช้เก็บชื่อรายการอาหาร วัตถุดิบและปริมาณที่ใช้ ขั้นตอนการปรุงอาหาร จำนวนครั้งที่มิผู้เข้าชม และจำนวนครั้งที่ผู้ใช้กดชอบ
- Ingredient ใช้เก็บชื่อวัตถุดิบ ที่ต้องใช้ในการทำอาหารของแต่ละสูตรการทำอาหาร
- Admins เก็บข้อมูลของผู้ดูแลระบบ

3.6 การประเมินประสิทธิภาพของแบบจำลอง

การประเมินประสิทธิภาพของแบบจำลองสำหรับการจัดหมวดหมู่นั้น ตามปกติมีค่าที่พิจารณาอยู่ 4 ค่า ได้แก่ (1) True Positive (TP) จำนวนที่ทำนายตรงกับข้อมูลจริงในคลาสที่กำลังพิจารณา (2) True Negative (TN) จำนวนที่ทำนายตรงกับข้อมูลจริงในคลาสที่ไม่ได้กำลังพิจารณา (3) False Positive (FP) จำนวนที่ทำนายผิดเป็นคลาสที่กำลังพิจารณา และ (4) False Negative (FN) จำนวนที่ทำนายผิดเป็นคลาสที่ไม่ได้กำลังพิจารณา

ในงานวิจัยนี้จะประเมินประสิทธิภาพของแบบจำลอง ด้วยกันทั้งหมด 3 ค่า ได้แก่ Precision,

Recall และ F1-score โดยคำนวณได้จากสมการที่ 1 ถึง สมการที่ 3

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$F1\ score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{3}$$

#### 4. ผลการทดลอง

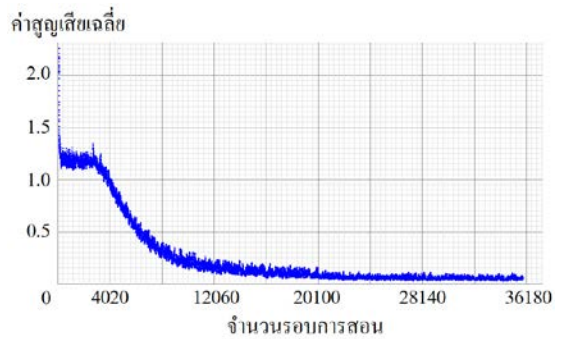
##### 4.1 ผลการสอนโครงข่ายประสาทเทียมการเรียนรู้เชิงลึก

การฝึกแบบจำลองได้ใช้เครื่องคอมพิวเตอร์ที่มีหน่วยประมวลผลกลาง Intel Core i7-5550U และหน่วยความจำขนาด 4 GB โดยมีหน่วยประมวลผลภาพรุ่น NVIDIA Geforce 920M ในการประมวลผล โดยกำหนดรอบในการสอนโครงข่ายประสาทเทียมการเรียนรู้เชิงลึกเอาไว้ที่ 36,000 รอบ ซึ่งใช้เวลาในการประมวลผลประมาณ 90 ชั่วโมง

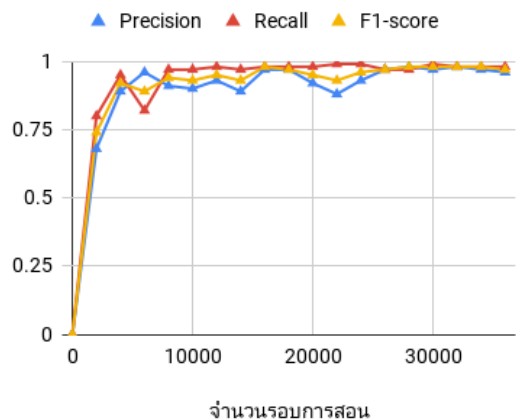
รูปที่ 5 แสดงค่าสูญเสียเฉลี่ย (Average Loss) ของแบบจำลอง โดยจะเห็นได้ว่าแบบจำลองมีค่าสูญเสียเฉลี่ยลดลงเรื่อยๆ และเริ่มคงที่ที่รอบการสอนที่ 20,000 และค่าสูญเสียเฉลี่ยที่ได้เมื่อสอนโครงข่ายครบ 36,000 รอบ อยู่ที่ 0.0408

สำหรับค่า Precision, Recall และ F1-score ของการสอนโครงข่ายแต่ละรอบนั้น แสดงในรูปที่ 6 ซึ่งเมื่อทำการสอนแบบจำลองครบ 36,000 รอบแล้ว ค่า Precision, Recall และ F1-score ของแบบจำลอง มีค่า 0.96, 0.98 และ 0.97 ตามลำดับ

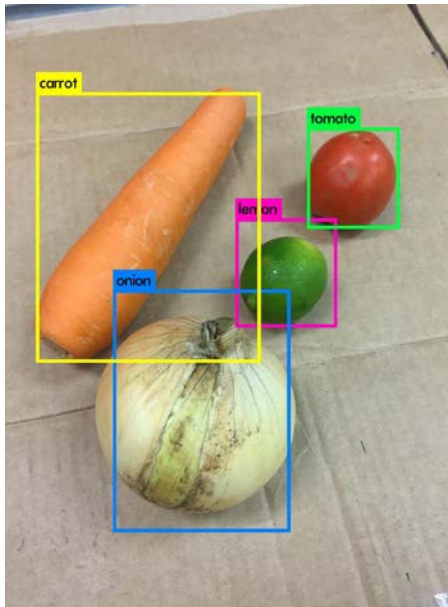
แต่อย่างไรก็ตามเมื่อนำไปใช้จริง แสงสว่างของภาพและฉากหลังของภาพก็มีส่วนในความถูกต้องของการสืบค้นด้วยรูปภาพ ดังรูปที่ 7 พบว่า แบบจำลองสามารถพบวัตถุสิบทั้ง 4 ชนิด คือ แครอท มะเขือเทศ มะนาว และหัวหอม ได้อย่างถูกต้อง แต่เมื่อฉากหลังเป็นสีแดงที่กลมกลืนกับมะเขือเทศ ความสว่างที่กะหล่ำปลีมีมากเกินไป และมะนาวถูกบังค่อนข้างเยอะ ดังรูปที่ 8 ทำให้แบบจำลองเจอวัตถุสิบเพียงแค่ 2 ชนิด และทำนายกะหล่ำปลีผิด เป็นกระเทียม



รูปที่ 5 แผนภูมิค่า Average Loss ของแบบจำลอง



รูปที่ 6 แผนภูมิค่า Precision, Recall และ F1-score

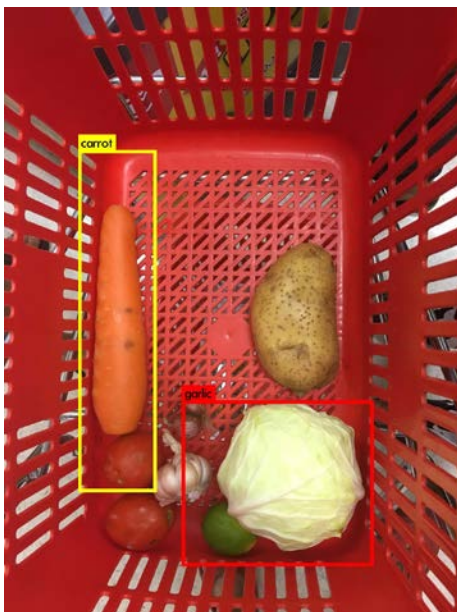


รูปที่ 7 ผลลัพธ์การตรวจจับวัตถุจากภาพจริง (ถูกต้อง)

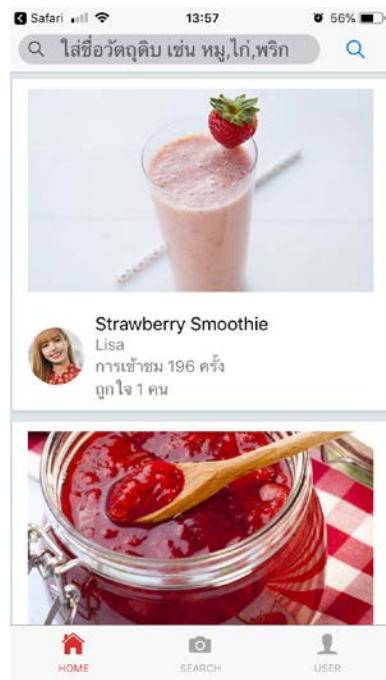
### 4.2 การใช้งานแอปพลิเคชันบนอุปกรณ์พกพา

เมื่อเปิดแอปพลิเคชันเข้ามาหน้าแรก เรียกว่า หน้า Home ดังแสดงในรูปที่ 9 โดยหน้านี้แสดงสูตรการทำอาหารล่าสุดที่ผู้ใช้ระบบเพิ่มรายการเข้ามา รวมถึงผู้ใช้สามารถพิมพ์รายชื่อวัตถุดิบ เพื่อค้นหาสูตรการทำอาหารที่มีวัตถุดิบเหล่านั้น เป็นส่วนประกอบในการปรุงอาหาร

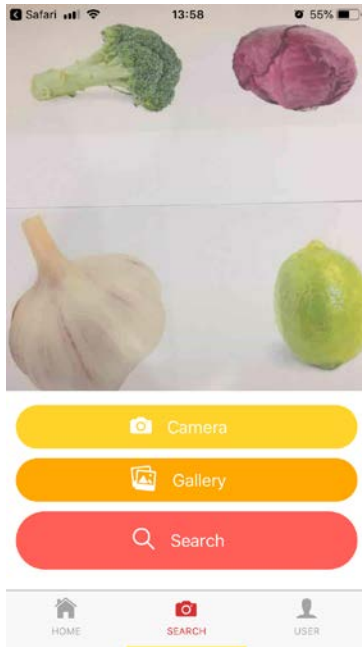
เมื่อผู้ใช้คลิกแท็บ Search หน้าแอปพลิเคชันจะแสดงดังรูปที่ 10 โดยในหน้านี้ผู้ใช้สามารถค้นหาสูตรการทำอาหารโดยใช้ภาพวัตถุดิบ ซึ่งผู้ใช้สามารถเลือกได้ว่า จะถ่ายภาพจากกล้องของอุปกรณ์พกพาเลย (Camera) หรือว่าจะใช้ภาพที่มีอยู่แล้วในอุปกรณ์พกพา (Gallery) เมื่อมีภาพวัตถุดิบที่ต้องการแล้วให้กดปุ่ม Search



รูปที่ 8 ผลลัพธ์การตรวจจับวัตถุจากภาพจริง (ไม่ถูกต้อง)



รูปที่ 9 หน้า Home ของแอปพลิเคชัน



รูปที่ 10 หน้าค้นหาสูตรการทำอาหารด้วยภาพ



รูปที่ 11 หน้าผลลัพธ์การค้นหาสูตรการทำอาหาร

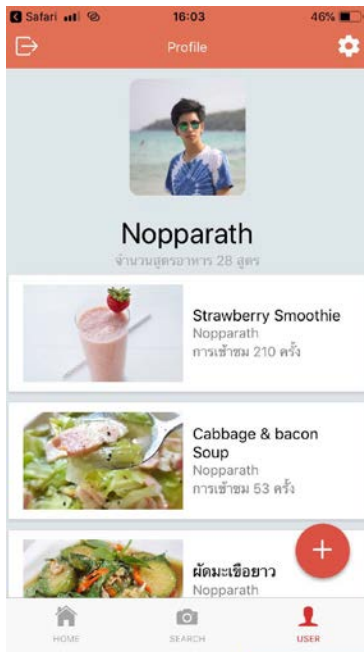
ผลลัพธ์ที่ได้จากการค้นหาแสดงดังรูปที่ 11 ซึ่งจะแสดงสูตรการทำอาหาร โดยจะมีการเรียงข้อมูลที่แสดงตามเปอร์เซ็นต์ ของสูตรการอาหารที่มีการใช้วัตถุดิบนั้นมากที่สุด รวมถึงผู้ใช้สามารถเรียงลำดับการแสดงผลข้อมูลสูตรการทำอาหารตามจำนวนผู้กดขึ้นชอบมากที่สุด หรือตามจำนวนผู้เข้าชมสูงสุด ได้อีกด้วย

เมื่อกดเข้าไปที่สูตรการทำอาหาร จะพบส่วนผสมและวิธีการทำอาหารดังแสดงในรูปที่ 12

สำหรับแท็บเมนูสุดท้ายคือ แท็บ User โดยหน้านี้จะแสดงสูตรการปรุงอาหารที่ผู้ใช้เป็นคนเพิ่มเข้าระบบเอง ดังแสดงในรูปที่ 13 และถ้าผู้ใช้ต้องการเพิ่มสูตรการทำอาหารใหม่ สามารถกดปุ่ม + ได้ โดยแอปพลิเคชันจะแสดงหน้าดังรูปที่ 14 เพื่อให้ผู้ใช้เพิ่มสูตรการทำอาหาร



รูปที่ 12 หน้าแสดงสูตรการทำอาหาร



รูปที่ 13 หน้าส่วนตัวของผู้ใช้งาน



รูปที่ 14 หน้าเพิ่มสูตรการทำอาหาร

### 4.3 การใช้งานเว็บแอปพลิเคชัน

เว็บแอปพลิเคชันในระบบนี้ สร้างขึ้นเพื่อให้ผู้ดูแลระบบสามารถเพิ่ม ลบ และแก้ไขคำหลักของวัตถุดิบแต่ละประเภทที่ต้องการให้สืบค้นด้วยภาพ ได้สะดวกมากขึ้น ดังรูปที่ 15 รวมถึงเพื่อการทดสอบระบบ ผู้ดูแลระบบสามารถอัปโหลดภาพ เพื่อดูข้อมูลวัตถุดิบที่แบบจำลองค้นพบได้ และจะคืนค่าวัตถุดิบที่ค้นพบออกมาในรูปแบบของเอกสาร JSON ด้วย ดังรูปที่ 16

### 5. สรุปผล

บทความวิจัยนี้ได้นำเสนอการออกแบบและพัฒนา ระบบแบ่งปันและค้นหาสูตรอาหารจากการถ่ายภาพ วัตถุดิบด้วยเทคนิคการเรียนรู้เชิงลึก โดยใช้ไลบรารี Darknet ในการสร้างแบบจำลองที่มีการใช้อัลกอริทึม YOLO

### Ingredient List

Label	Keywords	
egg	ไข่ไก่,ไข่,ไข่ต้ม,ไข่แดง,ไข่ขาว,	<a href="#">Edit</a> <a href="#">Keyword</a>
pineapple	สับปะรด,สับรด,สับประรด,	<a href="#">Edit</a> <a href="#">Keyword</a>
banana	กล้วย,กล้วย,	<a href="#">Edit</a> <a href="#">Keyword</a>
cabbage	กะหล่ำปลี,กะหล่ำ,ผักกะหล่ำ,	<a href="#">Edit</a> <a href="#">Keyword</a>
chicken	ไก่,เนื้อไก่,	<a href="#">Edit</a> <a href="#">Keyword</a>

รูปที่ 15 หน้าแก้ไขคำหลัก

ในการค้นหาและรู้จำภาพ โดยแบบจำลองที่พัฒนาขึ้นสามารถรู้จำวัตถุได้ 20 ประเภท จากการสอนด้วยภาพประเภทละ 100 ภาพ จำนวนทั้งหมด 36,000 รอบ ได้ค่า Precision, Recall และ F1-Score อยู่ที่ 0.96, 0.98 และ 0.97 ตามลำดับ

เพื่อใช้งานแบบจำลองที่พัฒนาขึ้น ผู้ใช้สามารถใช้งานผ่าน แอปพลิเคชันบนอุปกรณ์พกพาที่พัฒนาด้วย React native ซึ่งนอกจากผู้ใช้จะสามารถหาสูตรการทำอาหารด้วยการค้นหาจากภาพวัตถุดิบแล้ว ผู้ใช้ยังสามารถแบ่งปันสูตรการทำอาหารของตนให้กับผู้ใช้งานคนอื่นได้ รวมถึงสามารถกดขึ้นชอบสูตรการทำอาหารได้ สำหรับผู้ดูแลระบบจะใช้งานผ่านเว็บแอปพลิเคชันที่พัฒนาด้วย MERN Stack เพื่อเพิ่มคำหลักและทดสอบแบบจำลอง



รูปที่ 16 หน้าทดสอบแบบจำลอง

## 6. เอกสารอ้างอิง

- [1] B. Akay, O. Kaynar and F. Demirkoparan, "Deep Learning Based Recommender Systems", 2017 International Conference on Computer Science and Engineering (UBMK), Antalya, 2017, pp. 645-648.
- [2] G. Karatas, O. Demir and O. Koray Sahingoz, "Deep Learning in Intrusion Detection Systems", 2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT), ANKARA, Turkey, 2018, pp. 113-116.
- [3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, "ImageNet: A large-scale hierarchical image database", CVPR, 2009.
- [4] M. A. Subhi and S. Md. Ali, "A Deep Convolutional Neural Network for Food Detection and Recognition", 2018 IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES), Sarawak, Malaysia, 2018, pp. 284-287.
- [5] M. Chen, L. Zhang and J. P. Allebach, "Learning Deep Features For Image Emotion Classification", 2015 IEEE International Conference on Image Processing (ICIP), Quebec City, QC, 2015, pp. 4491-4495.
- [6] C. Huang, "Combining Convolutional Neural Networks For Emotion Recognition", 2017 IEEE MIT Undergraduate Research Technology Conference (URTC), Cambridge, MA, 2017, pp. 1-4.

- [7] G. A. R. Kumar, R. K. Kumar and G. Sanyal, "Discriminating Real From Fake Smile Using Convolution Neural Network", 2017 International Conference on Computational Intelligence in Data Science (ICCIDS), Chennai, 2017, pp. 1-6.
- [8] MongoDB, [online] Available: <http://www.mongodb.org/>, 21 May 2019.
- [9] React, [online] Available: <https://reactjs.org/>, 21 May 2019.
- [10] Node.js [online] Available: <https://nodejs.org/>, 21 May 2019.
- [11] React Native, [online] Available: <https://facebook.github.io/react-native/>, 21 May 2019.
- [12] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp. 779-788.
- [13] R. Shaoqing, H. Kaiming, G. Ross and S. Jian, "Faster R-CNN: Towards Real-time Object Detection with Region Proposal Networks", Advances in Neural Information Processing System 28, Curran Associates, Inc., 2015, pp. 91-99.
- [14] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger", CoRR, 2016.
- [15] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement", CoRR, 2018.
- [16] L. Pan, S. Pouyanfar, H. Chen, J. Qin and S. Chen, "DeepFood: Automatic Multi-Class Classification of Food Ingredients Using Deep Learning", 2017 IEEE 3rd International Conference on Collaboration and Internet Computing (CIC), San Jose, CA, 2017, pp. 181-189.
- [17] N. Hnoohom and S. Yuenyong, "Thai Fast Food Image Classification Using Deep Learning", 2018 International ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI-NCON), Chiang Rai, 2018, pp. 116-119.
- [18] Darknet, [online] Available: <https://github.com/pjreddie/darknet>, 21 May 2019.