รายงานผลงานเรื่องเต็มการทดลองที่สิ้นสุด

แผนบูรณาการ วิจัยและพัฒนาเพื่อความยั่งยืนของกล้วยไม้และไม้ดอกไม้ประดับ

แผนงานวิจัย วิจัยและพัฒนาเครื่องจักรกลเกษตรสำหรับการผลิตกล้วยไม้คุณภาพเพื่อการส่งออก

โครงการวิจัย ออกแบบ และพัฒนาชุดระบบตรวจสอบแมลงศัตรูสำคัญในกล้วยไม้เพื่อควบคุมการให้สารเคมีตามระบบ IPM

โครงการนี้ประกอบด้วย 2 การทดลอง ดังนี้

การทดลองที่ 1 ออกแบบ และพัฒนาระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ในกล้วยไม้สกุลหวาย (ปีเริ่มต้น ต.ค. 62 – สิ้นสุด ก.ย. 63)

การทดลองที่ 2 พัฒนาระบบควบคุมการพ่นสารเคมีร่วมกับระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ในกล้วยไม้สกุลหวายแบบอัตโนมัติ (ปีเริ่มต้น ต.ค. 63 – สิ้นสุด ก.ย. 64)

กรมวิชาการเกษตร

ออกแบบ และพัฒนาระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ในกล้วยไม้สกุลหวาย

Design and Development of a Detecting System for Thrips and Orchid Midge with Dendrobium Orchid

คณะผู้ดำเนินงาน

ตฤณสิษฐ์ ไกรสินบุรศักดิ์, อนุชา เชาว์โชติ, มงคล ตุ่นเฮ้า[1],พุทธธินันทร์ จารุวัฒน์[2],
ปรีชา อานันท์รัตนกุล[3], อนุชิต ฉ่ำสิงห์[3],ศรีจำนรรจ์ ศรีจันทรา[4],
พฤทธิชาติ ปุณวัฒโฑ[5], จิระวีณ์ ไกรสินบุรศักดิ์[6]

Tinnasit Kaisinburasak, Anucha Chaochot, Mongkol Tunhaw[1], Puttinun Jaruwat[2],
Preecha Ananratanakul[3], Anuchit Chamsing[3], Srijumnun Srijuntra[4],
Pruetthichat Punyawattoe[5], Jiravee Kaisinburasak[6]

หัวหน้าการทดลอง ตฤณสิษฐ์ ไกรสินบุรศักดิ์

ผู้ร่วมงาน อนุชา เชาว์โชติ

สถาบันวิจัยเกษตรวิศวกรรม กรมวิชาการเกษตร แขวงลาดยาว เขตจตุจักร กรุงเทพฯ 10900

Agricultural Engineering Research Institute, Department of Agriculture, Lat Yao, Chatuchak, Bangkok 10900, Thailand

กรมวิชาการเกษตร

บทคัดย่อ

เพลี้ยไฟ และบั่วกล้วยไม้ เป็นแมลงศัตรูพืชกล้วยไม้ที่สำคัญและมีการระบาดตลอดทั้งปี เนื่องจากเพลี้ยไฟมีขนาดเล็กเกษตรกรส่วนใหญ่จะตรวจพบเพลี้ยไฟในสถานการณ์ที่เริ่มมีการระบาดแล้ว ส่วนบั่วกล้วยไม้จะตรวจการระบาดได้จากอาการฉ่ำน้ำและบิดเบี้ยวที่พบบนดอกตูมซึ่งยากต่อการสังเกต งานวิจัยนี้มีวัตถุประสงค์เพื่อออกแบบระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ในกล้วยไม้สกุลหวาย โดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันในการวิเคราะห์ และจำแนกภาพ เพื่อลดความผิดพลาดจากการตรวจสอบโดยแรงงานคน ทำการทดสอบกับกล้วยไม้สกุลหวายในระยะให้ผลผลิตจำนวน 30 ก้อน เปรียบเทียบกับการใช้แรงงานคนจำนวน 1 คนที่มีความชำนาญในการตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ พบว่า เครื่องมีความสามารถในการตรวจสอบเพลี้ยไฟ 81.1% ตำแหน่งที่ถูกทำลายจากบั่วกล้วยไม้ 88.1% ใช้เวลาในการตรวจสอบเฉลี่ย 25.10 วินาทีต่อก้อน ส่วนแรงงานคนมีความสามารถในการตรวจสอบเพลี้ยไฟ 75.8% ตำแหน่งที่ถูกทำลายจากบั่วกล้วยไม้ 83.3% ใช้เวลาในการตรวจสอบเฉลี่ย 53.37 วินาทีต่อก้อน โดยช่วงแรกของการตรวจแรงงานคนมีความผิดพลาดน้อยกว่า แต่ความอ่อนล้าส่งผลให้ความผิดพลาดเพิ่มขึ้น

คำสำคัญ: เพลี้ยไฟ บั่วกล้วยไม้ กล้วยไม้สกุลหวาย โครงข่ายประสาทเทียมแบบคอนโวลูชัน

---

[1]ศูนย์วิจัยเกษตรวิศวกรรมขอนแก่น ตำบลบ้านทุ่ม อำเภอเมือง จังหวัดขอนแก่น 40000

[2]ศูนย์วิจัยเกษตรวิศวกรรมจันทบุรี 27 หมู่ 1 ตำบลพลับพลา อำเภอเมือง จังหวัดจันทบุรี 22000

[3]กลุ่มวิจัยวิศวกรรมหลังการเก็บเกี่ยว หมู่ 13 ตำบลคลองหนึ่ง อำเภอคลองหลวง จังหวัดปทุมธานี 12120

[4]กลุ่มบริหารศัตรูพืช สำนักวิจัยพัฒนาการอารักขาพืช กรมวิชาการเกษตร แขวงลาดยาว เขตจตุจักร กรุงเทพฯ 10900

[5]กลุ่มงานวิจัยการใช้สารป้องกันกำจัดศัตรูพืช สำนักวิจัยพัฒนาการอารักขาพืช กรมวิชาการเกษตร แขวงลาดยาว เขตจตุจักร กรุงเทพฯ 10900

[6]กลุ่มพัฒนาระบบตรวจสอบเมล็ดพันธุ์ กองวิจัยพัฒนาเมล็ดพันธุ์พืช กรมวิชาการเกษตร แขวงลาดยาว เขตจตุจักร กรุงเทพฯ 10900

กรมวิชาการเกษตร

ABSTRACT

Thrips and Orchid midge were major insect pests of Dendrobium orchid crops which could disperse every year. Farmers found the thrips after an outbreak because they had a small size. Orchid midge could detect an outbreak by considering from the vitrification and distortion on the orchid flower buds that they difficult observed. The objective of this research was to design the mechanism of detection for thrips and orchid midge which could reduce an error of the human labors. A Convolutional Neural Network (CNN) had been used for images classifications. It was tested with Dendrobium orchid as compared to a human labor. The number of growing media was 30 clumps. The experimental results showed that the efficacy of detection for thrips was 81.1%, the efficacy of detection for orchid midge was 88.1% and the average time of detection was 25.10 second/clump which were detected by the prototype, the efficacy of detection for thrips was 75.8%, the efficacy of detection for orchid midge was 83.3% and the average time of detection was 53.37 second/clump which were detected by a human labor. When a human labor first detected orchid he worked it better than prototype but later on, he worked it worse than prototype because he was fatigued.

Keywords: Thrips, Orchid midge, Dendrobium orchid, Convolutional Neural Network

---

[1]Khon Kaen Agicultural Engineering Research Center Bantum subdistrict Meuang district Khon Kaen province 40000

[2]Chanthaburi Agicultural Engineering Research Center Phlapphla subdistrict Mueang district Chanthaburi province 22000

[3]Post-harvest Engineering Research Group, Agricultural Engineering Research Institute, Department of Agriculture, Khlong Nueng, Khlong Luang, Pathum Thani 12120, Thailand

[4]Pest Management Research Group, Plant Protection Research and Development Office, Department of Agriculture, Chatuchak, Bangkok 10900, Thailand

[5]Pesticide Application Research Technique Section, Entomology and Zoology Group, Plant Protection Research and Development Office, Department of Agriculture, Chatuchak, Bangkok 10900, Thailand

[6]Seed Research and Development Division, Department of Agriculture, Lat Yao, Chatuchak, Bangkok 10900, Thailand

กรมวิชาการเกษตร

คำนำ

ปัญหาสำคัญอย่างหนึ่งสำหรับเกษตรกรผู้ปลูกกล้วยไม้คือปัญหาด้านแมลงศัตรูพืช สำหรับแมลงศัตรูพืชกล้วยไม้ที่สำคัญได้แก่ เพลี้ยไฟ และบั่วกล้วยไม้ ซึ่งมีการระบาดตลอดทั้งปี (สุภราดาและคณะ, 2554) เพื่อให้มีการใช้สารป้องกันกำจัดศัตรูพืชอย่างมีเหตุผล และไม่ให้มีการใช้เกินความจำเป็นจึงมีการนำระบบบริหารศัตรูพืช (Integrated Pest Management : IPM) มาใช้ประเมินสถานการณ์ศัตรูพืชเพื่อตัดสินใจในการพ่นสารเคมีโดยพ่นทั่วทั้งโรงเรือนหลังจากการประเมินสถานการณ์ศัตรูพืชซึ่งมีค่าเกินกว่าระดับการตัดสินใจที่กำหนด ระบบนี้จะทำการสุ่มช่อดอกกล้วยไม้จำนวน 40 ช่อดอก/ไร่ ในกรณีของเพลี้ยไฟจะประเมินสถานการณ์การระบาดจากการตรวจนับตัวอ่อนและตัวเต็มวัย หรือประเมินแบบตรวจพบ (มี-ไม่มี) ที่บริเวณดอกบานของกล้วยไม้ ส่วนบั่วกล้วยไม้จะใช้การประเมินจากการดูการทำลายของดอกตูม โดย 40 ช่อดอกที่ได้จากการสุ่มถ้าพบตั้งแต่ 4 ดอกขึ้นไปจะประเมินให้ทำการพ่นสารเคมีทั่วทั้งโรงเรือน (ศรีจำนรรจ์ และคณะ, 2559) การตรวจหาศัตรูกล้วยไม้โดยการประเมินศัตรูพืชแบบตรวจพบและตรวจนับ ในกรณีของเพลี้ยไฟซึ่งมีขนาดเล็กระยะแรกของการทำลายจะดูค่อนข้างยาก ทำให้การประเมินสถานการณ์ศัตรูพืชในแปลงกล้วยไม้เป็นเรื่องยากต่อการปฏิบัติของเกษตรกร โดยเกษตรกรส่วนใหญ่จะตรวจพบเพลี้ยไฟในระยะตัวเต็มวัยซึ่งเริ่มมีการระบาดแล้ว ส่วนบั่วกล้วยไม้จะตรวจพบการระบาดได้จากอาการที่พบบนดอกตูม โดยเกษตรกรสามารถประเมินการระบาดได้เมื่อพบอาการฉ่ำน้ำและบิดเบี้ยวในดอกตูมซึ่งแสดงว่ามีการเข้าทำลายในแปลงมากแล้ว แม้แต่ในกรณีของนักวิชาการเกษตรด้านกีฏวิทยาซึ่งมีความชำนาญในการตรวจนับและประเมินสถานการณ์ยังต้องใช้วิธีและเครื่องมือในการตรวจที่ซับซ้อน การตรวจอาจมีความผิดพลาดเนื่องจากอาการอ่อนล้า รวมถึงความสม่ำเสมอในการตรวจสอบขึ้นอยู่กับลักษณะของแต่ละบุคคล

ปัจจุบันเทคโนโลยีความแม่นยำทางการเกษตรโดยอาศัยการจำแนกภาพซึ่งเป็นงานที่ต้องอาศัยการวิเคราะห์ โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional neural network: CNN) ได้ถูกนำมาใช้กันอย่างแพร่หลายในการรับรู้ภาพ การจำแนกภาพ และการตรวจจับวัตถุที่มีประสิทธิภาพใกล้เคียงกับมนุษย์ (พิมพา, 2562) โดยการสร้างแบบจำลอง Machine Learning เพื่อให้เครื่องคอมพิวเตอร์เรียนรู้คุณสมบัติเชิงภาพ ซึ่งสามารถนำไปประยุกต์ใช้กับงานมากมาย ตัวอย่างเช่น Zafrulla (2011) ศึกษาความแตกต่างของภาษามือที่ใช้ในประเทศสหรัฐอเมริกาและประเทศอังกฤษ Makoto Koike (2018) สร้างแบบจำลอง Machine Learning เพื่อให้เครื่องคอมพิวเตอร์เรียนรู้คุณสมบัติเชิงภาพของแตงกวา และคัดแยกแตงกวาออกเป็น 9 เกรด โดยใช้ภาพถ่ายแตงกวาเกรดต่างๆจำนวน 7,000 ภาพ ขนาด 80x80 พิกเซลโดยใช้การประมวลผลแบบ Deep Learning ผลการทดสอบในการแยกแยะมีความถูกต้อง 95% ซึ่งถ้าหากใช้คนเรียนรู้ความแตกต่างของแตงกวาแต่ละเกรดต้องใช้เวลาหลายเดือนในการเรียนรู้

กรมวิชาการเกษตร

งานวิจัยนี้จึงมีแนวคิดในการประยุกต์ใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันสำหรับจำแนกภาพถ่ายเพลี้ยไฟ และอาการทำลายจากบั่วกล้วยไม้มาพัฒนาเป็นระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ในกล้วยไม้สกุลหวาย โดยมีวัตถุประสงค์เพื่อลดความผิดพลาดจากการตรวจสอบโดยใช้แรงงานคน ได้แก่ ความสม่ำเสมอในการตรวจสอบ และเวลาในการตรวจสอบ ซึ่งช่วยให้ผู้เพาะปลูกกล้วยไม้สกุลหวายสามารถป้องกันกำจัดเพลี้ยไฟและบั่วกล้วยไม้ได้อย่างรวดเร็ว รวมถึงลดต้นทุนการใช้สารเคมี ส่งผลให้ดอกกล้วยไม้สกุลหวายมีคุณภาพดีและเพิ่มราคาผลผลิต

วิธีดำเนินการ

อุปกรณ์

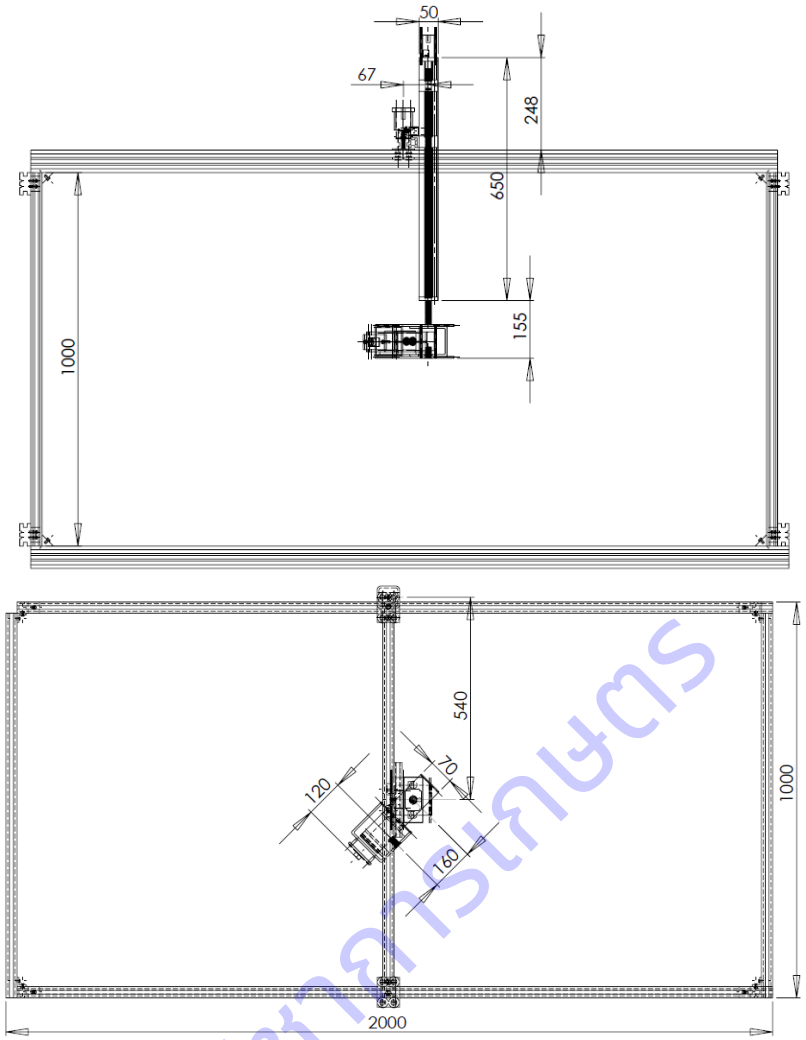
อุปกรณ์ที่ใช้ในระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ประกอบด้วย อุปกรณ์ที่ประกอบเป็นโครงสร้างและลักษณะจำเพาะของเครื่องต้นแบบ และอุปกรณ์ที่ใช้ในการทดสอบดังนี้

1. โครงสร้าง และลักษณะจำเพาะของระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้

ระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ที่ออกแบบและสร้างแสดงในภาพที่ 1 และ 2 ประกอบด้วยโครงสร้าง 3 ส่วน คือโครงสำหรับการเคลื่อนที่ของแขนกล แขนกล และกล้องดิจิทัลสำหรับใช้ในการตรวจสอบ โดยแขนกลสามารถเคลื่อนที่บนโครงผ่านล้อขับซึ่งทำจากพลาสติกแข็ง ส่วนโครงทำจากอลูมิเนียมพื้นที่หน้าตัดสี่เหลี่ยมผืนผ้า ขนาด 50 x 100 มม. ความยาว 2,000 มม. จำนวน 4 ท่อน ความยาว 1,000 มม. จำนวน 4 ท่อน และอลูมิเนียมพื้นที่หน้าตัดสี่เหลี่ยมจตุรัสขนาด 60 x 60 มม. สูง 1,000 มม. จำนวน 4 ท่อน ประกอบเป็นรูปสี่เหลี่ยมผืนผ้า แต่ละส่วนถูกยึดด้วยบานพับสามารถถอดประกอบได้

แขนกลประกอบด้วยโลหะทำจากแผ่นอลูมิเนียมและเชื่อมต่อเข้ากับมอเตอร์ทั้ง 4 ตัว มอเตอร์ที่ใช้เป็นเซอร์โวมอเตอร์ ส่วนปลายแขนกลติดตั้งโมดุลของกล้องดิจิทัล Sony A6000 ที่มีเลนส์ระยะ 18 -105 มม. แสดงในภาพที่ 3

กรมวิชาการเกษตร



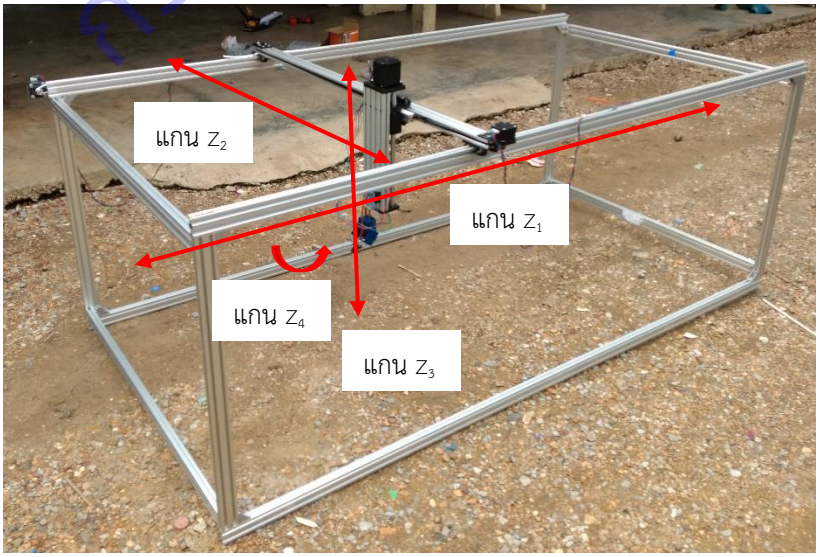
ภาพที่ 1 ภาพฉาย 2 มิติ Orthographic ของระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้

กรมวิชาการเกษตร

ภาพที่ 2 ระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ที่ติดตั้งในโรงเรือน

ภาพที่ 3 การติดตั้งเซอร์โวมอเตอร์ทั้ง 4 ตัวของแขนกล

การเคลื่อนที่ของแขนกลจะอยู่ภายใต้พื้นที่ 2 ตร.ม. ครอบคลุมพื้นที่ปลูกกล้วยไม้ 60 ก้อน มอเตอร์และกระปุกเกียร์ที่ใช้ในแขนกลมี 2 ขนาด โดยมอเตอร์ตัวที่ 1 2 และ 3 มีขนาดแรงบิด 5.8 นิวตัน-เมตร อัตราทดเกียร์ 1:20 ส่วนมอเตอร์ตัวที่ 4 มีขนาดแรงบิด 2.2 นิวตัน-เมตร อัตราทดเกียร์ 1:25 มอเตอร์ตัวที่ 1 และ 2 ใช้หมุนขับล้อขับผ่านเฟืองทดขนาด 1:1.5 ให้เคลื่อนที่ตามแกน Z_1 และ Z_2 ส่วนมอเตอร์ตัวที่ 3 ใช้หมุนขับสกรูโดยตรงให้เคลื่อนที่ตามแกน Z_3 และมอเตอร์ตัวที่ 4 ใช้หมุนขับข้อต่อให้หมุนตามแกน Z_4 โดยมอเตอร์ตัวที่ 1 2 และ 3 มีขนาดใหญ่กว่าและมีกำลังมากกว่าเพราะเป็นมอเตอร์ที่ใช้ขับเคลื่อนแขนกลตามแนวแกนซึ่งมีน้ำหนักมากกว่า แสดงในภาพที่ 4



ภาพที่ 4 การเคลื่อนที่ของแขนกล 4 แกน

กรมวิชาการเกษตร

2. อุปกรณ์ไฟฟ้า

2.1 เราเตอร์ยี่ห้อ D-Link/DIR-612

2.2 บอร์ดควบคุมขนาดเล็ก (Microcontroller) ยี่ห้อ Arduino Mega 2560

2.3 คอมพิวเตอร์แบบพกพา (Notebook)

2.4 อุปกรณ์เชื่อมต่อคอมพิวเตอร์ด้วยสัญญาณ Wi-Fi TP-LINK/TL-WN321G

2.5 บอร์ดขับเคลื่อนมอเตอร์ยี่ห้อ Accelus รุ่น ASP-180-09

2.6 โมดุลรีเลย์สำหรับเชื่อมต่อ 8 ช่อง

2.7 บอร์ด Ethernet W5100 R3 Wireless Wi-Fi

2.8 โมดุล Mega 2560 R3 Ch340G Wireless Wi-Fi

2.9 Servo motor ขนาดแรงบิด 5.8 และ 2.2 นิวตัน-เมตร

3. อุปกรณ์วัดที่ใช้ทดสอบระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้

3.1 ตลับเมตร

3.2 นาฬิกาจับเวลา

3.3 แว่นขยายเลนส์ขนาด 25x แบบสวมหัว

4. เครื่องมือ และวัสดุที่ใช้ในการสร้างระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้

4.1 เครื่องเชื่อมไฟฟ้า

4.2 เครื่องเชื่อมแก๊ส

4.3 เครื่องกลึง

4.4 เครื่องเจาะ

4.5 เครื่องตัด

4.6 เครื่องดัด

4.7 เหล็กสำหรับทำโครงสร้าง

4.8 อะลูมิเนียมสำหรับทำแขนกล

4.9 เฟืองทดขนาด 1:1.5 จำนวน 2 คู่

กรมวิชาการเกษตร

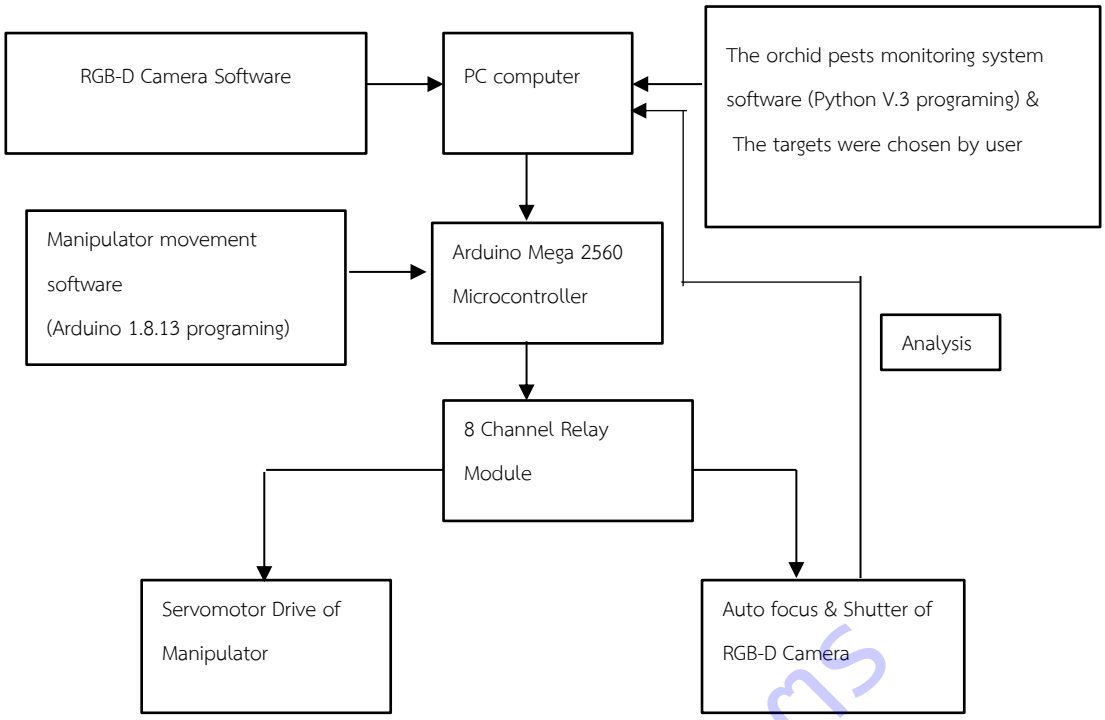
วิธีการ

วิธีการดำเนินการประกอบด้วย หลักการทำงานของเครื่องต้นแบบ การควบคุมแขนกล 4 แกน การตรวจจับเพลี้ยไฟ และบั่วกล้วยไม้ การนับและแสดงผล การส่งข้อมูลผ่านระบบเน็ตเวิร์ค และการทดสอบความสามารถในการตรวจสอบดังนี้

5. หลักการทำงานของระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ในกล้วยไม้สกุลหวาย

หลักการทำงานของระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ในกล้วยไม้สกุลหวาย แสดงในภาพที่ 5 โดยเริ่มจากการโหลดซอฟต์แวร์ของ Camera และซอฟต์แวร์ระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ที่พัฒนาขึ้นจากโปรแกรม Python V.3 เข้าสู่คอมพิวเตอร์ ส่วนโปรแกรมการเคลื่อนที่ของแขนกลที่พัฒนาขึ้นจากโปรแกรม Arduino 1.8.13 ถูกโหลดเข้าสู่บอร์ดควบคุม (Arduino Mega 2560) จากนั้นผู้ใช้งาน (User) จะเลือกกล้วยไม้ช่อที่ต้องการตรวจสอบจากหน้าจอคอมพิวเตอร์ และ คอมพิวเตอร์จะส่งข้อมูลไปที่บอร์ดควบคุม (Arduino Mega 2560) ซึ่งจะทำการคำนวณพิกัด (x,y,z) ตำแหน่งมุม และตำแหน่งเคลื่อนที่ของแขนกล รวมถึงส่งสัญญาณควบคุมไปที่ชุดขับเคลื่อนแขนกล (Motor Drive) ให้แขนกลเคลื่อนที่ไปยังตำแหน่งดอกกล้วยไม้ที่ต้องการตรวสอบ คอมพิวเตอร์จะเริ่มทำการตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ผ่านโปรแกรมตรวจสอบ โดยส่งสัญญาณสั่งงานผ่านบอร์ดควบคุมไปที่มอเตอร์สำหรับหมุนเพิ่มอัตรากำลังขยายของกล้องดิจิทัล เมื่อพบเป้าหมายคอมพิวเตอร์จะทำการวิเคราะห์ภาพที่พบโดยใช้เทคนิคโครงข่ายประสาทเทียมแบบ CNN และส่งสัญญาณสั่งงานผ่านบอร์ดควบคุมไปที่ชัตเตอร์ของกล้องเพื่อทำการถ่ายภาพ ภาพที่ได้จะถูกนำไปวิเคราะห์เพื่อนับจำนวนเพลี้ยไฟ และจำนวนตำแหน่งที่ถูกทำลายจากบั่วกล้วยไม้เพื่อดูแนวโน้มการระบาดที่เกิดขึ้น

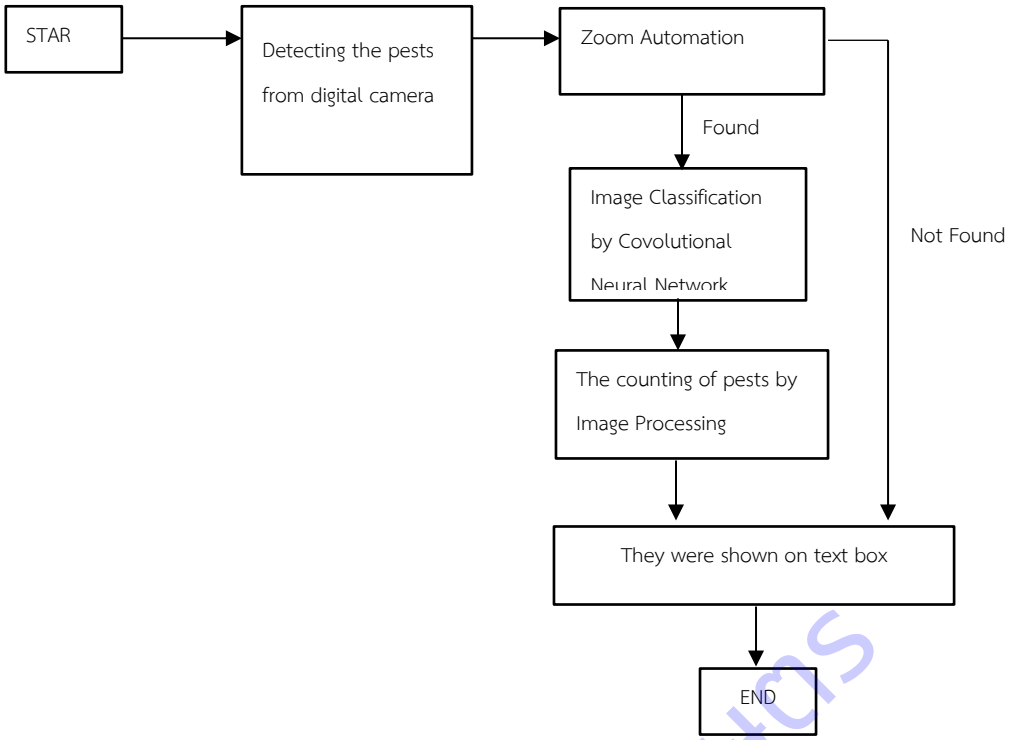
กรมวิชาการเกษตร



ภาพที่ 5 หลักการทำงานของระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้

ส่วนการออกแบบ และพัฒนาระบบชุดคำสั่งเฉพาะสำหรับระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ กระทำโดยการค้นคว้า และรวบรวมข้อมูลเกี่ยวกับอัลกอริทึมของระบบชุดคำสั่งเฉพาะได้แก่ ลักษณะของภาษาที่ใช้ในการเขียนชุดคำสั่งเฉพาะ ลักษณะของอัลกอริทึมที่เหมาะสมกับงาน โดยมีหลักการในการประมวลผลตามภาพที่ 6 กล้วยไม้ซึ่งถูกตรวจจากกล้องดิจิทัลโดยระบบตรวจสอบเริ่มใช้เทคนิคการเพิ่มอัตรากำลังขยายของกล้องแบบอัตโนมัติเพื่อขยายภาพเพลี้ยไฟซึ่งมีขนาดเล็ก ส่วนบั่วกล้วยไม้เข้าทำลายดอกกล้วยไม้จากภายในจึงเลือกภาพลักษณะการทำลายของบั่วกล้วยไม้ซึ่งแสดงว่ามีบั่วกล้วยไม้อยู่ เมื่อพบเป้าหมายคอมพิวเตอร์จะทำการแยก Classification ของภาพโดยใช้เทคนิคระบบโครงข่ายประสาทเทียมแบบ CNN และถ่ายภาพส่งมาวิเคราะห์โดยใช้กระบวนการของ Image processing เพื่อนับจำนวนเพลี้ยไฟที่พบ และจำนวนตำแหน่งที่ถูกบั่วกล้วยไม้เข้าทำลายไปแสดงผลในหน้าจอ text box บนคอมพิวเตอร์ กรณีที่ตรวจไม่พบจะส่งข้อมูลไปแสดงผลในหน้าจอ text box เช่นเดียวกันพร้อมทั้งบันทึกข้อมูลลง SD card ในคอมพิวเตอร์

กรมวิชาการเกษตร

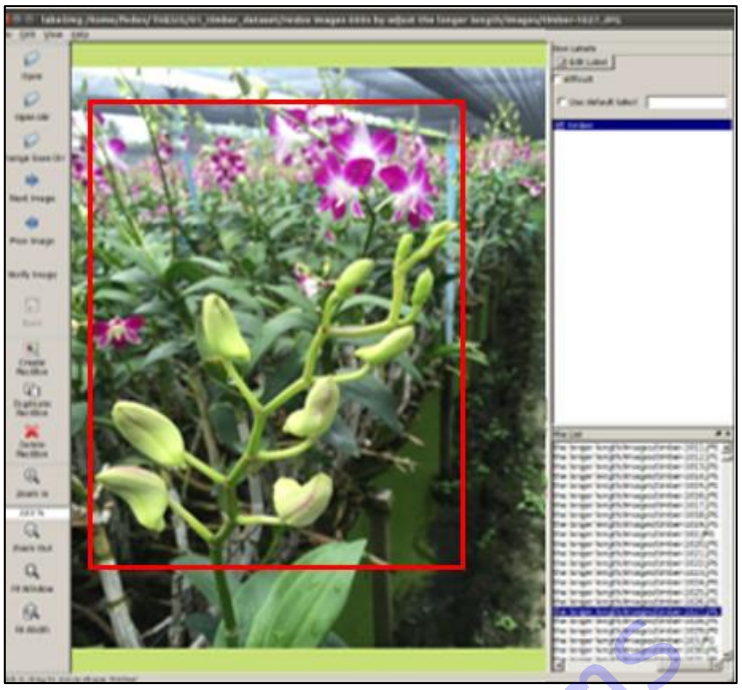


ภาพที่ 6 หลักการวิเคราะห์ และตรวจสอบของระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้

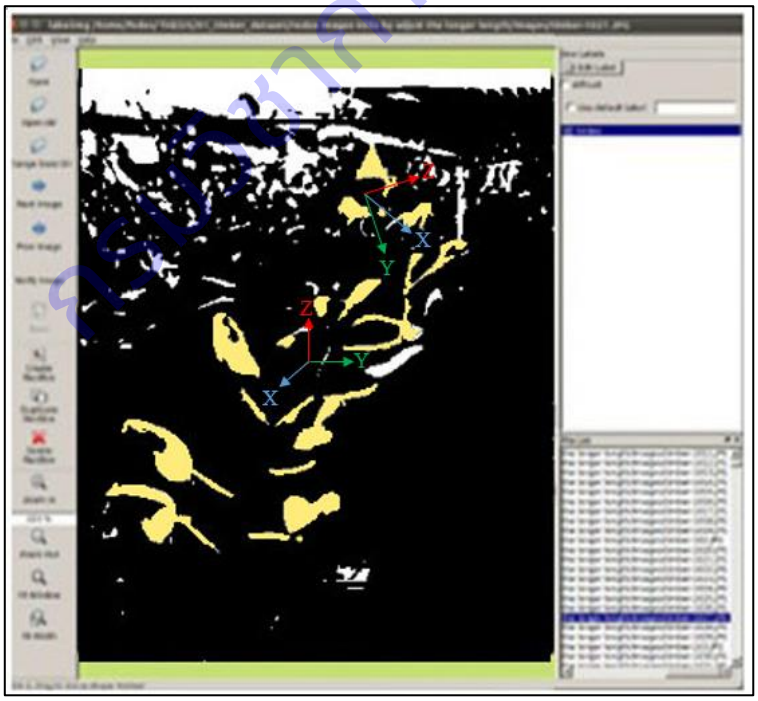
6. การควบคุมแขนกล 4 แกน (4 DOFs) ของระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้

ระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้จะเริ่มทำการตรวจสอบโดยส่งภาพผ่านกล้องดิจิทัลไปแสดงที่หน้าจอคอมพิวเตอร์ผ่านโปรแกรม Imaging Edge Desktop ซึ่งเป็น software ของกล้อง Sony A6000 แสดงในภาพที่ 7 แล้วใช้วิธีเพิ่มคุณลักษณะพิเศษของข้อมูล (Data augmentation) เพื่อระบุตำแหน่งในที่นี้คือช่อกล้วยไม้ โดยใช้ซอฟแวร์ Arduino 1.8.13 เพื่อหาพิกัดเป้าหมายปลายแขนกลโดยใช้วิธี forward kinematic และ inverse kinematic แสดงในภาพที่ 8 พิกัดเป้าหมายปลายแขนกลสามารถแสดงผลในรูปแบบ Textbox ทั้งหมด 4 ช่อง ได้แก่ X-Axis, Y-Axis, Z-Axis และ Rotation ตามลำดับ ซึ่งช่อง Rotation คือค่าองศาการหมุนสำหรับแกนหมุนของแขนกล ส่วน X-Axis, Y-Axis, Z-Axis จะเป็นตำแหน่งของปลายแขนกลในระบบพิกัดคาร์เทเชียลที่ต้องการให้ปลายแขนกลเคลื่อนที่ไปแสดงในภาพที่ 9

กรมวิชาการเกษตร

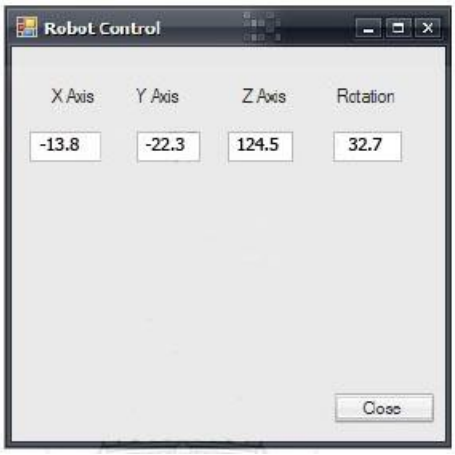


ภาพที่ 7 แสดงภาพหน้าจอด้วยโปรแกรม Imaging Edge Desktop และกรอบการตรวจจับวัตถุโดยใช้โปรแกรม Arduino 1.8.13



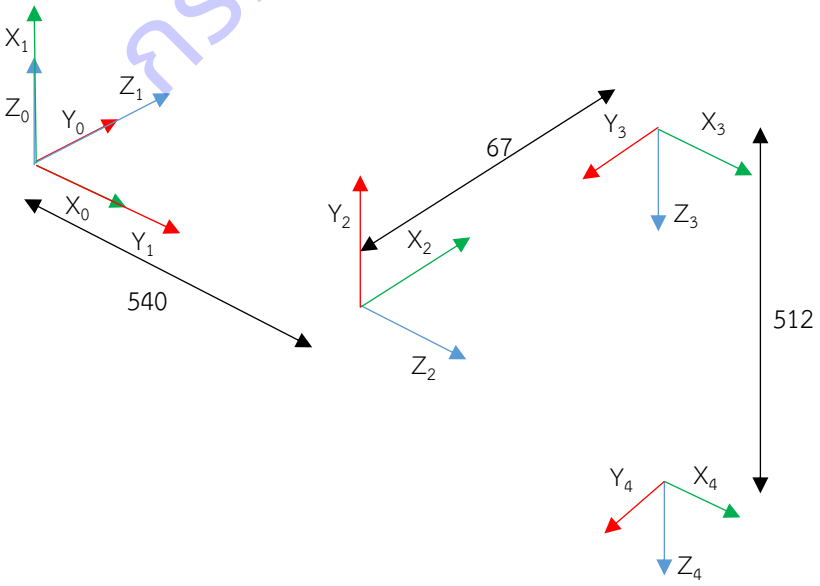
ภาพที่ 8 เพิ่มคุณลักษณะพิเศษของข้อมูล (Data augmentation) พร้อมกับแสดงแกนพิกัดเป้าหมายปลายแขนกลโดยใช้ซอฟแวร์ Arduino 1.8.13

สำนักวิทยบริการฯ



ภาพที่ 9 ค่าพิกัดเป้าหมายปลายแขนกลแสดงผลในรูปแบบ Textbox

พิกัดเป้าหมายปลายแขนกลที่ได้จะถูกส่งไปเป็นพิกัดที่เทียบกับพิกัดฐานของแขนกล ซึ่งการที่จะควบคุมแขนกลให้ปลายแขนสามารถเคลื่อนที่ไปสู่ตำแหน่งต่างๆที่ต้องการได้นั้นจะต้องอาศัยหลักการทางจลศาสตร์ และพลศาสตร์ เมื่อการเคลื่อนที่นั้นใช้ความเร็วต่ำ และสูงตามลำดับ เพื่อคำนวณหาตำแหน่งต่างๆของแต่ละแกนของหุ่นยนต์ที่เคลื่อนที่ไปในระบบพิกัด (X, Y, Z) โดยใช้ เวกเตอร์เพื่อจะช่วยระบุได้ทั้งขนาดและและทิศทางในการเคลื่อนที่ของแขนกล ในภาพที่ 10 แสดงการตั้งแกนของแขนกลแบบ 4 แกนที่ใช้ในการทดลองโดยใช้หลักการของ Denavit Hartenberg เพื่อสร้างแบบจำลองทางจลศาสตร์ของแขนกล 4 แกน โดยกำหนดให้แกน Z ของแต่ละแกนเป็นแกนหมุน และนำข้อมูลที่ได้จากการตั้งแกนไปวิเคราะห์เพื่อทำการสร้างตาราง Denavit Hartenberg แสดงในตารางที่ 1



ภาพที่ 10 การกำหนดแกนของแขนกล 4 แกน

กรมวิชาการเกษตร

ตารางที่ 1 Denavit-Hartenberg สำหรับแขนกล 4 แกน

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	-90^o	0	d_1	$\theta_1(-90^o)$
2	-90^o	0	$d_2 + 540$	$\theta_2(-90^o)$
3	90^o	a_2 (67)	d_3	$\theta_3(90^o)$
4	0	0	d_4 (512)	θ_4

โดยที่ a_{i-1} = ความยาวของแขนหุ่นยนต์ (มม.)

α_{i-1} = มุมที่แขนบิดซึ่งเทียบกับข้อต่อก่อนหน้า (องศา)

d_i = ระยะห่างที่เกิดจากรูปร่างแขนที่ไม่เป็นเส้นตรง (มม.)

θ_i = มุมที่หมุนได้ในที่นี้คือมอเตอร์ (องศา)

จากตาราง Denavit - Hartenberg มาแทนในสมการ Transformation Matrix จะได้สมการจลศาสตร์ไปข้างหน้า (Forward Kinematic) ของแขนกล 4 แกน ซึ่งตำแหน่งของปลายแขนกลจะเป็นฟังชันก์ที่แสดงโดยเมทริกซ์ แสดงในสมการที่ 1

$$T_4^0 = T_1^0 \times T_2^1 \times T_3^2 \times T_4^3 \quad (1)$$

$$T_4^0 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_2 + 540 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 0 & -1 & 0 & 67 \\ 0 & 0 & -1 & -d_3 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times$$

$$\begin{bmatrix} \cos(\theta_4) & -\sin(\theta_4) & 0 & 0 \\ \sin(\theta_4) & \cos(\theta_4) & 0 & 0 \\ 0 & 0 & 1 & 512 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_4^0 = \begin{bmatrix} \cos(\theta_4) & -\sin(\theta_4) & 0 & d_2 + 540 \\ -\sin(\theta_4) & -\cos(\theta_4) & 0 & d_1 + 67 \\ 0 & 0 & -1 & -d_3 - 512 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

และจากสมการจลศาสตร์ไปข้างหน้า เราสามารถคำนวณหาสมการจลศาสตร์แบบผกผัน (Inverse Kinematic) ของแขนกล 4 แกนได้โดยอาศัยหลักการคำนวณหาจาโคเบียน (Jacobian) ที่แสดงความสัมพันธ์ระหว่างความเร็วเชิงเส้นของปลายแขน และความเร็วเชิงมุมของแต่ละข้อต่อดังแสดงในสมการที่ 2

กรมวิชาการเกษตร

$$\dot{X_4} = J(q)\dot{q} \quad (2)$$

โดยที่

$$j(q) = \frac{\partial f}{\partial q} = [\frac{\partial f}{\partial q_1} \frac{\partial f}{\partial q_2} \dots \frac{\partial f}{\partial q_4}] \quad (3)$$

เมื่อ

$f = T_4^0[\, x_0\ y_0\ z_0]^T$ คือ เวกเตอร์ตำแหน่งปลายแขนกลหรือก้านโยงที่ 4

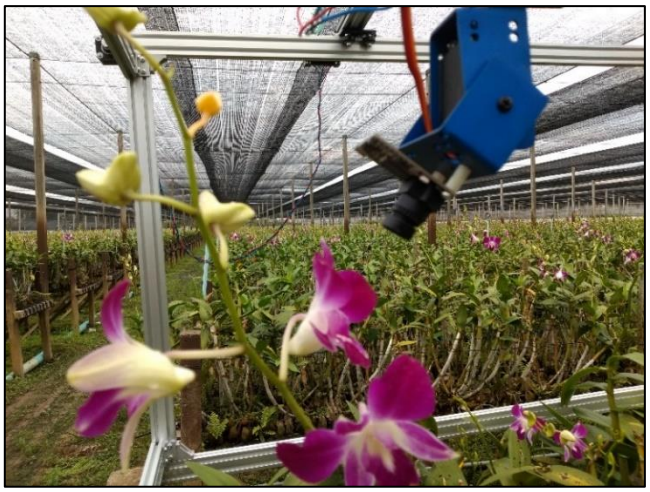
$\dot{q_0} = [\dot{q_1}\ \dot{q_2} \dots \dot{q_4}]^T$ คือ เวกเตอร์ความเร็วเชิงมุมของแต่ละข้อต่อ

$\dot{x_4} = [\dot{x_4}\ \dot{y_4}\ \dot{z_4}]^T$ คือ เวกเตอร์ความเร็วเชิงเส้นของปลายแขนกลท่อนที่ 4

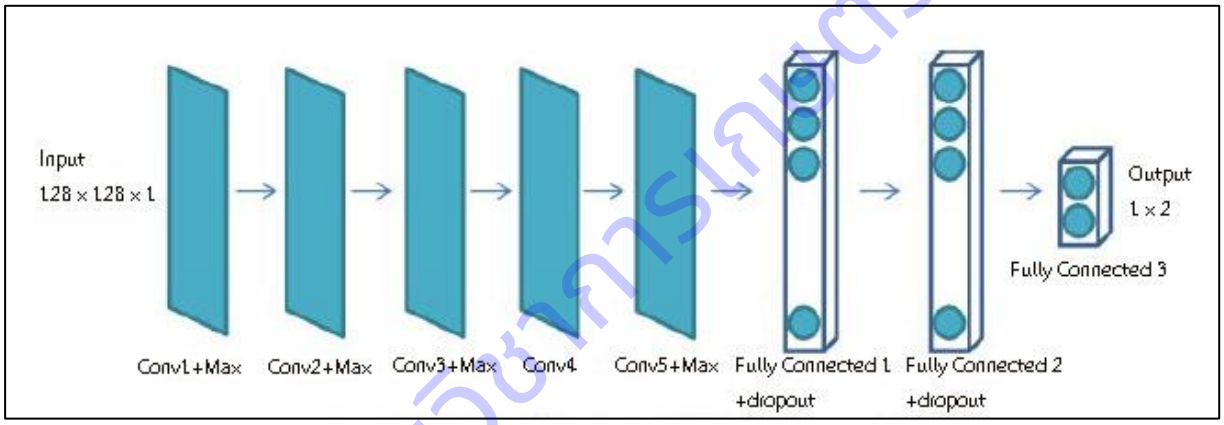
7. การตรวจจับเพลี้ยไฟ และบั่วกล้วยไม้ของระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้

เมื่อแขนกลเคลื่อนที่มาถึงตำแหน่งหน้าดอกกล้วยไม้แสดงในภาพที่ 11 กล้องที่ปลายแขนกลจะทำหน้าที่ในการถ่ายภาพเป็นเฟรมเพื่อนำมาใช้ในการคำนวณสำหรับการแยกชนิดของภาพ (Classification) งานวิจัยนี้ได้นำโครงข่ายประสาทเทียมแบบคอนโวลูชันมาประยุกต์ใช้แทนสมการทางคณิตศาสตร์โดยใช้โปรแกรม Python V.3 ประกอบด้วย 5 ชั้นคอนโวลูชัน (Conv Layer) 3 ชั้นเชื่อมโยงเต็มรูปแบบ (Fully-connected Layer) และ 2 ชั้นดรอปเอาซ์ (Dropout Layer) แต่ละ Conv layer จะประกอบด้วย Relu layers เพื่อลดขนาดมิติที่ใช้คำนวณลง (Dimensionality Reduction) Conv layer จะทำหน้าที่ดึงคุณลักษณะที่โดดเด่นออกมา (Distinct Feature) ชั้น Conv1 มี 96 ตัวกรอง ขนาดตัวกรองที่ใช้คือ 11x11 ชั้น Conv2 มี 256 ตัวกรอง ขนาดตัวกรองที่ใช้คือ 5x5 ชั้น Conv3 มี 512 ตัวกรอง ชั้น Conv4 และ Conv5 มีตัวกรองเท่ากันคือ 1024 ตัวกรอง ขนาดตัวกรองของชั้น Conv3, Conv4 และ Conv5 ที่ใช้คือ 3x3 ชั้น Con1, Conv2, Conv3 และ Conv5 จะใช้ชั้นการรวมที่มีค่ามากที่สุด (Max Pooling Layer) มีขนาด 2x2 และในชั้น Conv1 จะใช้ค่าสไตรด (Stride) เป็น 4 พิกเซล (Pixel) ส่วนชั้นที่เหลือจะใช้ค่าสไตรดเป็น 1 พิกเซล ฟังก์ชันกระตุ้น (Activation Function) ที่ใช้ในแต่ละชั้นยกเว้นชั้น FC3 คือ Relu function อัตราการเรียนรู้ (Learning Rate) ที่ใช้คือ 0.001 และตัวทำให้เหมาะสมที่สุด (Optimizer) ที่ใช้คือ Adam ในชั้น Fully-connected 3 (FC3) จะใช้ฟังก์ชันกระตุ้นเป็น softmax function เพื่อที่จะให้ชั้น FC3 นี้เป็นชั้นที่ไว้ทำนายภาพว่าภาพชนิดใดคือเพลี้ยไฟ อาการทำลายจากบั่วกล้วยไม้ หรือกล้วยไม้ช่อสมบูรณ์ โดยโครงสร้างคอนโวลูชันที่ออกแบบแสดงในภาพที่ 12 ใช้จำนวนภาพในการฝึกสอนทั้งหมด 672 ภาพ ตัวอย่างภาพที่ใช้ในการฝึกสอนแสดงในภาพที่ 13

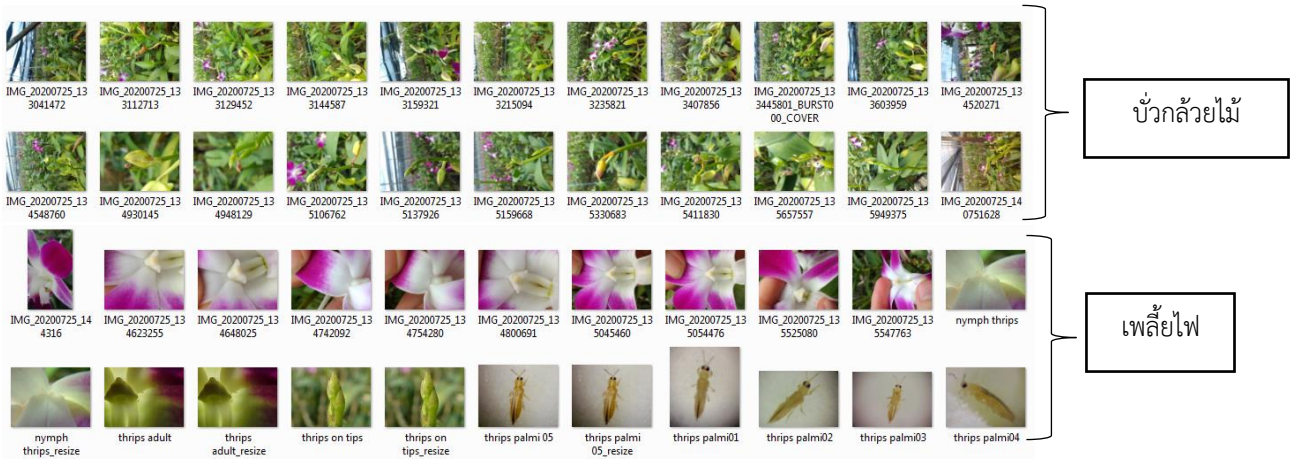
ลิขสิทธิ์ กรมวิชาการเกษตร



ภาพที่ 11 ตำแหน่งของกล้องและด้านหน้าของดอกกล้วยไม้



ภาพที่ 12 การออกแบบ Convolutional Neural Network

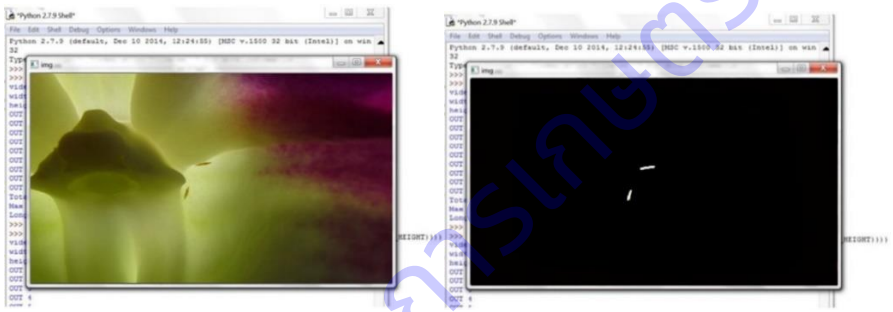


ภาพที่ 13 ตัวอย่างของภาพที่ใช้ในการฝึกสอน

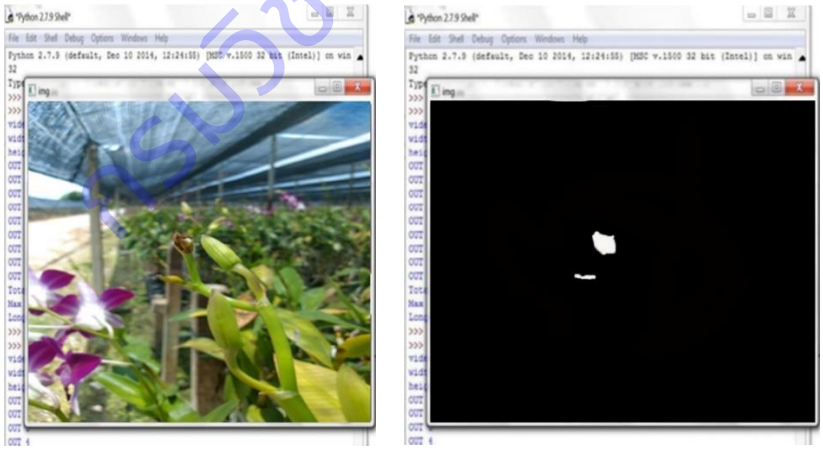
กรมวิชาการเกษตร

8. การนับและแสดงผลจำนวนเพลี้ยไฟ และบั่วกล้วยไม้ของระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้

ภาพที่ผ่านการแยกชนิดของภาพจากโครงข่ายประสาทเทียมแบบคอนโวลูชันจะถูกนำมานับจำนวนเพลี้ยไฟ และบั่วกล้วยไม้ที่พบโดยใช้โปรแกรม Python V.3 ไลบรารีที่ใช้คือ NumPy, OpenCV, HSV และ MessageBox โดยอาศัยหลักการตรวจจับตำแหน่งวัตถุด้วยการเปรียบเทียบค่าของสีโดยแปลงระบบสีของภาพจาก RGB เป็น HSV เพื่อลดผลกระทบจากค่าความเข้มแสง จากนั้นทำการแยกแต่ละแชนแนลสีเพื่อทำ Binary Threshold และใช้กระบวนการนับ Count โดยสีขาวเป็น foreground ที่เราสนใจ ส่วนสีดำคือ background แสดงในภาพที่ 14 ระบบการนับจำนวนวัตถุที่ได้พัฒนาขึ้นแสดงในภาพที่ 15 แสดงผลการนับในรูปแบบกล่องข้อความแสดงในภาพที่ 16 โดยกล่องข้อความ thrips แสดงจำนวนเพลี้ยไฟที่พบ orchid midge แสดงจำนวนตำแหน่งที่ถูกบั่วกล้วยไม้เข้าทำลาย และ orchid แสดงกล้วยไม้ก้อนที่ตรวจ



(ก)

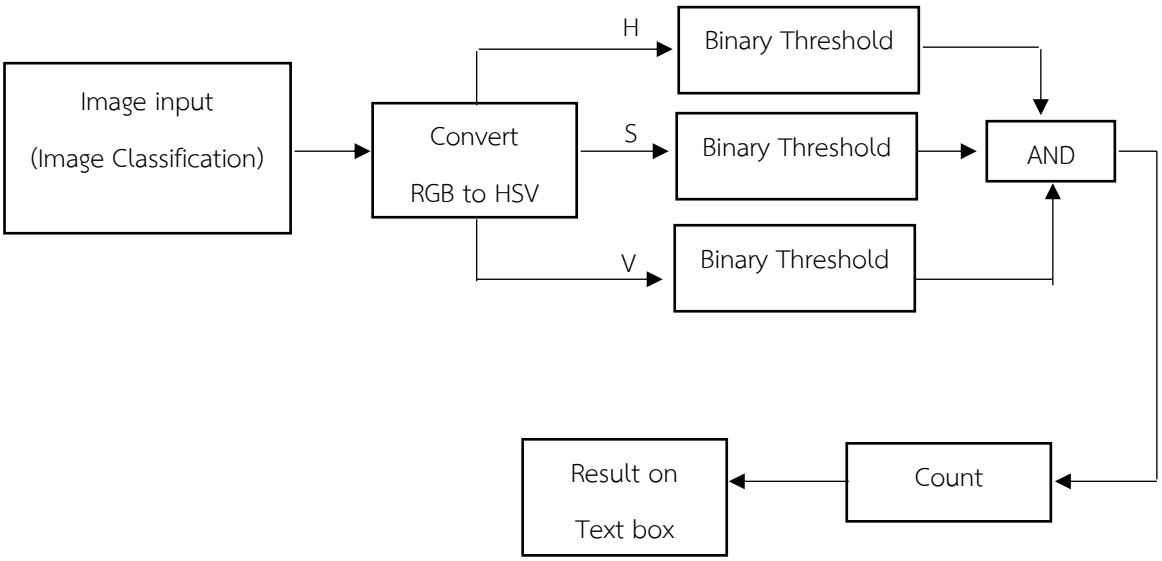


(ข)

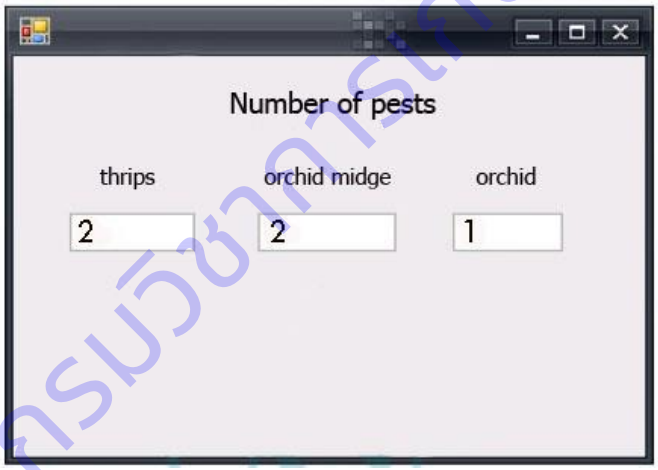
ภาพที่ 14 (ก) แสดง foreground และ background ของภาพเพลี้ยไฟในโปรแกรม Python V.3

(ข) แสดง foreground และ background ของภาพบั่วกล้วยไม้ในโปรแกรม Python V.3

ลิขสิทธิ์การเกษตร



ภาพที่ 15 Block diagram ของระบบการนับจำนวนวัตถุ



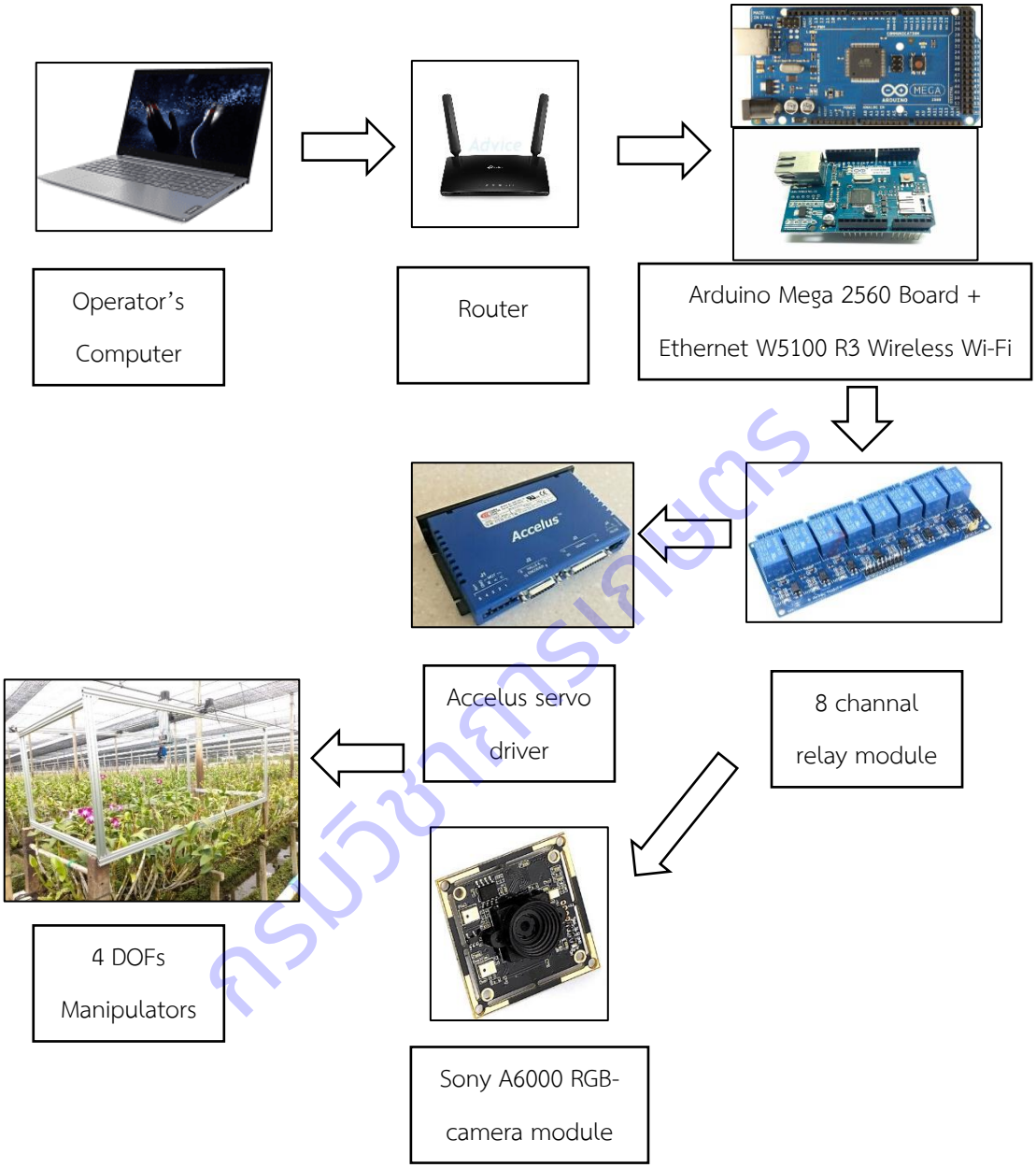
ภาพที่ 16 แสดงผลการนับจำนวนเพลี้ยไฟ และบั่วกล้วยไม้ในรูปแบบ Textbox

9. การส่งข้อมูลผ่านระบบเน็ตเวิร์คของระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้

การควบคุมตำแหน่งแขนกล 4 แกน และส่งข้อมูลการตรวจสอบผ่านระบบเน็ตเวิร์คเพื่อสะดวกในการใช้งานในโรงเรือนปลูกกล้วยไม้มี Router เป็นตัวกลางในการเชื่อมต่อข้อมูลสัญญาณการควบคุม ประกอบด้วย คอมพิวเตอร์แบบพกพา (Notebook) และโมดุล Mega 2560 R3 Ch340G Wireless Wi-Fi ของบอร์ดควบคุม Arduino Mega 2560 ในการรับสัญญาณมาประมวลผลการสั่งงานการตรวจสอบและควบคุมการทำงานของแขน

กรมวิชาการเกษตร

กลที่บอร์ดควบคุม พร้อมทั้งส่งสัญญาณการตรวจสอบ และสัญญาณการควบคุมไปที่อุปกรณ์ฮาร์ดแวร์แสดงในภาพที่ 17



ภาพที่ 17 การส่งถ่ายข้อมูลผ่านระบบเน็ตเวิร์คของระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้

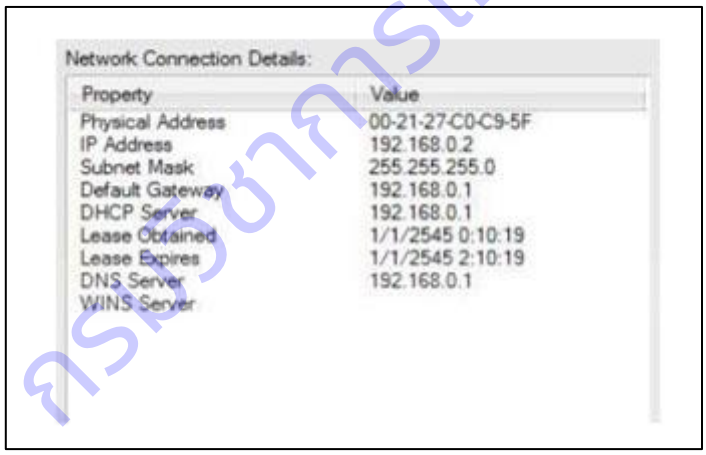
จากการออกแบบการเชื่อมต่อและส่งถ่ายข้อมูลผ่านระบบคอมพิวเตอร์เน็ตเวิร์คนี้ ผู้ทดลองทำการออกแบบให้บอร์ดควบคุม Arduino Mega 2560 ทำหน้าที่ในการประมวลผลทางคณิตศาสตร์พร้อมส่งสัญญาณการควบคุมไปที่โมดุลกล้อง Sony A6000 RGB-camera และสัญญาณควบคุมการทำงานของแขนกลไปที่ Accelus servo driver ของข้อต่อแขนกลแต่ละแกน ผ่าน Relay module โดยทำหน้าที่เสมือนเป็นเซิฟเวอร์

กรมวิชาการเกษตร

(Server) ของระบบเน็ตเวิร์คนี้ ซึ่งจะรับค่าตำแหน่งปลายแขนกล และคำสั่งการตรวจสอบมาจากคอมพิวเตอร์ของผู้ควบคุม โดยที่คอมพิวเตอร์ของผู้ควบคุมจะทำหน้าที่เป็นไคลเอนต์ (Client) ของระบบเน็ตเวิร์คนี้

เราเตอร์ (Router) เป็นอุปกรณ์คอมพิวเตอร์ที่ทำหน้าที่จ่าย IP Address ให้กับ บอร์ดควบคุม Arduino Mega 2560 ซึ่งมีโมดุล R3 Ch340G Wireless Wi-Fi ในการรับสัญญาณ Wi-Fi ให้สามารถติดต่อข้อมูลกับคอมพิวเตอร์ของผู้ควบคุม และค้นหาเส้นทางการส่งข้อมูลระหว่างเครื่องคอมพิวเตอร์ต้นทางไปยังเครื่องคอมพิวเตอร์ปลายทางที่ต้องการ โดยเราเตอร์ที่ใช้เป็นยี่ห้อ D-Link/DIR-612 มีความถี่ที่ใช้ 2.4 GHz ความเร็วในการส่งข้อมูล 54 Mbps ส่วนการเชื่อมต่อคอมพิวเตอร์ของผู้ควบคุมเข้ากับระบบเน็ตเวิร์คด้วยสัญญาณ Wi-Fi ใช้อุปกรณ์ TP-LINK/TL-WN321G เป็นอุปกรณ์ในการเชื่อมต่อคอมพิวเตอร์ของผู้ควบคุมเข้ากับเราเตอร์ D-Link/DIR-612

ระบบเน็ตเวิร์คนี้จะมี Defalut Gateway เป็น 192.168.0.1 มี IP Address สำหรับคอมพิวเตอร์ของผู้ควบคุมเป็น 192.168.0.2 และมี Physical Address เป็น 00-21-27-C0-C9-5F แสดงในภาพที่ 18



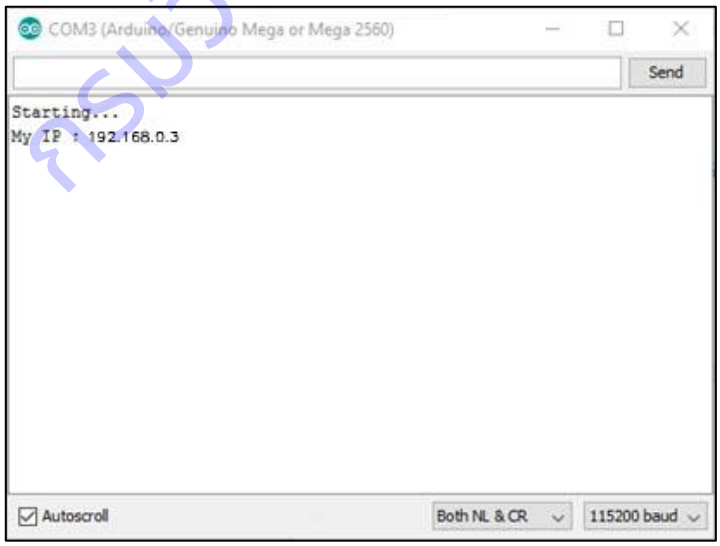
ภาพที่ 18 รายละเอียดการเชื่อมต่อเน็ตเวิร์คของคอมพิวเตอร์ของผู้ควบคุม

ทำการเชื่อมต่อบอร์ดควบคุม Arduino Mega 2560 เข้ากับบอร์ด Ethernet W5100 R3 Wireless Wi-Fi เข้ากับระบบเน็ตเวิร์คผ่านทางสาย LAN โดยจ่ายไฟเลี้ยง หรือใช้ Adapter 7-12 V หลังจากนั้นเข้าโปรแกรม Arduino IDE เขียนโค้ดตามภาพที่ 19 ลงในโปรแกรมเพื่อต่อ Arduino Mega 2560 เข้ากับบอร์ด Ethernet Shield ผ่าน Router แล้วคอมไพล์โค้ด ทำการคลิกที่ปุ่ม Serial monitor จะมี IP Address เป็น 192.168.0.3 แสดงผลบนจอในภาพที่ 20 เมื่อสามารถเชื่อมต่อผ่านระบบเน็ตเวิร์คได้จะสามารถส่งผ่านข้อมูลระหว่างกันได้โดยไม่ต้องใช้สาย LAN จากคอมพิวเตอร์ของผู้ควบคุม

กรมวิชาการเกษตร

```
#include<Ethernet.h>
byteMy_MAC_address[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
voidsetup()
{
Serial.begin(115200);
Serial.println("Starting...");
// start the Ethernet connection:
while(Ethernet.begin(My_MAC_address) != 1)
{
Serial.print(".");
}
Serial.print("My IP :");
Serial.println(Ethernet.localIP());
}
voidloop()
{
}
```

ภาพที่ 19 โค้ดสำหรับเชื่อมต่อบอร์ด Arduino Mega 2560 กับบอร์ด Ethernet W5100 R3



ภาพที่ 20 รายละเอียดการเชื่อมต่อเน็ตเวิร์คของบอร์ด Arduino Mega 2560

ศรีวิชาการเกษตร

10. ทดสอบความสามารถในการตรวจสอบของระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้

การทดสอบความสามารถในการตรวจสอบของระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ จะแบ่งการทดสอบออกเป็น 2 ส่วน ดังนี้

10.1 ทดสอบความสามารถในการแยกชนิดของภาพโดยโครงข่ายประสาทเทียมแบบคอนโวลูชันที่ออกแบบ ข้อมูลที่ใช้เป็นภาพถ่ายในโรงเรือนปลูกกล้วยไม้สกุลหวายจำนวน 840 ภาพ แต่ละภาพมีขนาด 128x128 พิกเซล แบ่งเป็น 2 หมวด คือ เพลี้ยไฟ และอาการทำลายจากบั่วกล้วยไม้ และใช้ 672 ภาพในการฝึกสอนให้คอมพิวเตอร์เรียนรู้เป็นจำนวน 100 รอบ ส่วน 168 ภาพใช้ทดสอบความถูกต้องของแบบจำลอง ตัววัดที่ใช้ในการประเมินแบบจำลอง คือ Precision Recall และ F-Measure (ตะวันส่องแสง, 2561)

10.2 ทดสอบความสามารถในการตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ในโรงเรือนปลูกกล้วยไม้เปรียบเทียบกับแรงงานคน กระทำโดยคัดเลือกกล้วยไม้สกุลหวายในระยะให้ผลผลิตจำนวน 30 ก้อน (วัสดุปลูกกล้วยไม้เรียกเป็นก้อน) โดยก่อนการทดสอบผู้ทดสอบจะเลือกสุ่มกล้วยไม้บางก้อนแล้วใส่เพลี้ยไฟลงไปในช่อดอกกล้วยไม้ และเลือกกล้วยไม้ที่มีอาการทำลายจากบั่วกล้วยไม้ รวมถึงกล้วยไม้ที่สมบูรณ์ไม่มีแมลงศัตรูพืชมาใช้ทดสอบ ซึ่งผู้ทดสอบจะทราบข้อมูลก่อนการทดสอบว่ากล้วยไม้ก้อนใดมีเพลี้ยไฟจำนวนกี่ตัว ตำแหน่งการทำลายจากบั่วกล้วยไม้กี่ตำแหน่ง และกล้วยไม้ก้อนใดมีความสมบูณ์ เพื่อใช้เปรียบเทียบผลการตรวจสอบและค่าความผิดพลาดที่เกิดขึ้น กล้วยไม้แต่ละก้อนวางเรียงต่อกันในโรงเรือนปลูกกล้วยไม้สกุลหวาย โดยจัดแบ่งเป็น 2 ชุด สำหรับการตรวจสอบโดยใช้เครื่อง และแรงงานคน

10.2.1) การตรวจสอบโดยใช้เครื่อง ระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้จะทำการตรวจสอบกล้วยไม้ที่ได้เตรียมไว้ในข้อ 6.2 เมื่อกล้องตรวจสอบอยู่ในตำแหน่งด้านหน้าของดอกกล้วยไม้ หลังจากนั้นจดบันทึกข้อมูลจากการตรวจสอบของเครื่อง ได้แก่ จำนวนเพลี้ยไฟ จำนวนตำแหน่งที่ถูกทำลายจากบั่วกล้วยไม้ ค่าความผิดพลาดจากการตรวจสอบ และเวลาที่ใช้ในการตรวจสอบ

10.2.2) การตรวจสอบโดยใช้แรงงานคน จะใช้แรงงานคนจำนวน 1 คน ที่มีความชำนาญในการตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ทำการตรวจสอบกล้วยไม้ที่ได้เตรียมไว้ในข้อ 6.2 หลังจากนั้นจดบันทึกข้อมูลจากการตรวจสอบของแรงงานคน ได้แก่ จำนวนเพลี้ยไฟ จำนวนตำแหน่งที่ถูกทำลายจากบั่วกล้วยไม้ ค่าความผิดพลาดจากการตรวจสอบ และเวลาที่ใช้ในการตรวจสอบ

กรมวิชาการเกษตร

โรงเรือนปลูกกล้วยไม้สกุลหวายที่ใช้ในการทดสอบนี้เป็นโรงเรือนปลูกกล้วยไม้ของสวนกล้วยไม้ชาณธนิต อำเภอเมือง จังหวัดนครปฐม โดยระหว่างการทดสอบจะให้เกษตรกรผู้ปลูกกล้วยไม้สกุลหวายจำนวน 10 ราย และนักวิชาการเกษตรของสำนักวิจัยพัฒนาการอารักขาพืช กรมวิชาการเกษตร ร่วมกันพิจารณา

เวลาและสถานที่

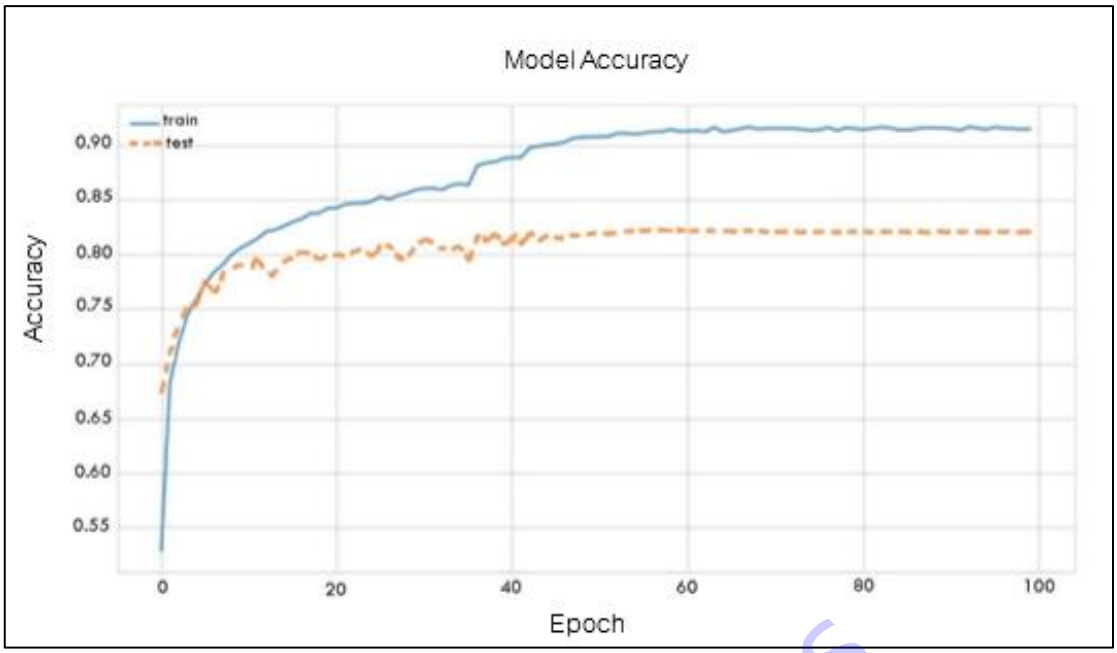
สถานที่ในการทดสอบเป็นโรงเรือนปลูกกล้วยไม้สกุลหวายของสวนกล้วยไม้ชาณธนิต อ.เมือง จ.นครปฐม กล้วยไม้ในโรงเรือนมีดอกบานประมาณ 3 ใน 4 ของจำนวนดอกทั้งช่อ เวลาในการทำวิจัย เริ่มดำเนินงานเดือน ต.ค. พ.ศ. 2562 ถึงเดือน ก.ย. พ.ศ. 2563

ผลการทดลองและวิจารณ์

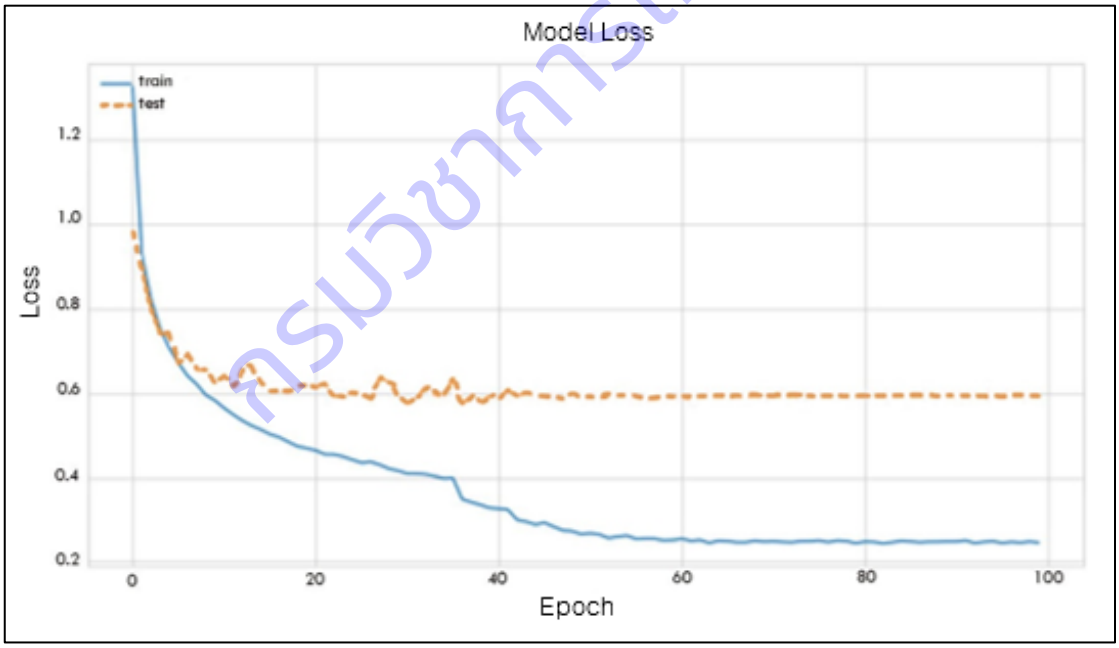
1. ทดสอบความสามารถในการแยกชนิดของภาพโดยโครงข่ายประสาทเทียมแบบคอนโวลูชันที่ออกแบบ

ในการจำแนกภาพด้วยโครงข่ายประสาทเทียมแบบคอนโวลูชันโดยกำหนดให้เครื่องคอมพิวเตอร์เรียนรู้จำนวน 100 รอบ พบว่า เมื่อเพิ่มจำนวนรอบในการฝึกสอนความแม่นยำในการทำนายภาพจะสูงขึ้นแสดงในภาพที่ 21 จากกราฟจะเห็นว่ารอบที่ 60-100 ค่าความแม่นยำในการทำนายภาพเริ่มคงที่ มีการเปลี่ยนแปลงเพียงเล็กน้อย ส่วนค่า Loss เริ่มคงที่ ไม่ลดลงที่รอบ 60 ทั้งชุดฝึกสอนและชุดทดสอบแสดงในภาพที่ 22 โดยกราฟเส้นทึบแสดงค่าความแม่นยำจากชุดฝึกสอนมีค่า 91.26% ซึ่งมากกว่าค่าความแม่นยำที่ได้จากชุดทดสอบที่มีค่า 82.06% เมื่อนำแบบจำลองมาคำนวณค่า Precision Recall และ F-Measure จะได้ผลลัพธ์แสดงในตารางที่ 2 โดยแบบจำลองสามารถทำนายลักษณะการทำลายจากบั่วกล้วยไม้มีค่าความแม่นยำ 78% และเพลี้ยไฟมีค่าความแม่นยำ 86%

กรมวิชาการเกษตร



ภาพที่ 21 กราฟแสดงความสัมพันธ์ระหว่าง accuracy และรอบของการฝึกสอน



ภาพที่ 22 กราฟแสดงความสัมพันธ์ระหว่าง Loss และรอบของการฝึกสอน

ประสิทธิภาพของแบบจำลองนอกจากจะดูที่ค่าความแม่นยำในการทำนายภาพแล้ว จะดูที่ค่า Overfitting ซึ่งเป็นเหตุการณ์ที่แบบจำลองทำนายได้ดีมากในขั้นตอนของการสอน (Training data) แต่เมื่อนำแบบจำลองนั้น

กรมวิชาการเกษตร

ไปทำงานกับข้อมูลที่ต้องการทดสอบ (Testing data) ประสิทธิภาพในการทำนายต่ำมาก ซึ่งเป็นข้อมูลที่แบบจำลองไม่เคยเห็นมาก่อน และปัญหาจากการที่มีข้อมูลไม่เพียงพอในการฝึกสอนให้คอมพิวเตอร์เรียนรู้ โดยค่า loss ทั้งชุดฝึกสอนและชุดทดสอบไม่ลดลงหรือเพิ่มขึ้นหลังจากรอบที่ 60 แสดงว่าไม่เกิด Overfitting จนเกินไป สำหรับแบบจำลองนี้ฝึกสอน 60 รอบเพียงพอ

ตารางที่ 2 ผลการทดสอบค่าตัววัด Precision Recall และ F-Measure สำหรับการจำแนกภาพ

Type	Precision	Recall	F-Measure
orchid midge	0.78	0.79	0.78
thrips	0.86	0.85	0.85
Mean	0.82	0.82	0.82

2. ทดสอบความสามารถในการตรวจสอบของระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ในโรงเรือนปลูกกล้วยไม้เปรียบเทียบกับแรงงานคน

ผลการทดสอบระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้โดยทำการตรวจสอบกล้วยไม้ทั้งหมด 30 ก้อนในโรงเรือนปลูกกล้วยไม้ระบบเปิดสภาวะแวดล้อมปลูกจริงเปรียบเทียบกับการใช้แรงงานคนในการตรวจสอบแสดงในตารางที่ 3 พบว่า เพลี้ยไฟจำนวน 132 ตัว เครื่องตรวจพบ 107 ตัว ตรวจไม่พบ 25 ตัว คิดเป็นความผิดพลาด 18.9% แรงงานคนตรวจพบ 100 ตัว ตรวจไม่พบ 32 ตัว คิดเป็นความผิดพลาด 24.2% ส่วนตำแหน่งที่ถูกทำลายจากบั่วกล้วยไม้จำนวน 42 ตำแหน่ง เครื่องตรวจพบ 37 ตำแหน่ง ตรวจไม่พบ 5 ตำแหน่ง คิดเป็นความผิดพลาด 11.9% แรงงานคนตรวจพบ 35 ตำแหน่ง ตรวจไม่พบ 7 ตำแหน่ง คิดเป็นความผิดพลาด 16.7% โดยเครื่องใช้เวลาในการตรวจสอบทั้งหมด 753 วินาที หรือ 12.6 นาที เฉลี่ย 25.10 วินาทีต่อก้อน และแรงงานคนใช้เวลาในการตรวจสอบทั้งหมด 1,601 วินาที หรือ 26.7 นาที เฉลี่ย 53.37 วินาทีต่อก้อน เครื่องใช้เวลาในการตรวจสอบกล้วยไม้ก้อนที่ 9 นานที่สุด 29 วินาที มีค่าความผิดพลาดในการตรวจเพลี้ยไฟสูงสุด 2 ตัว และตำแหน่งที่ถูกทำลายจากบั่วกล้วยไม้สูงสุด 1 ตำแหน่ง ส่วนกล้วยไม้ก้อนที่ 4 และ 15 ใช้เวลาในการตรวจสอบเร็วที่สุด 20 วินาที และเป็นกล้วยไม้ที่สมบูรณ์ไม่มีเพลี้ยไฟ และบั่วกล้วยไม้ เช่นเดียวกับแรงงานคนที่ใช้เวลาในการตรวจสอบกล้วยไม้ก้อนที่ 9 นานที่สุด 76 วินาที มีค่าความผิดพลาดในการตรวจเพลี้ยไฟสูงสุด 1 ตัว ไม่มีความผิดพลาดจากการตรวจตำแหน่งที่ถูกทำลายจากบั่วกล้วยไม้ ส่วนกล้วยไม้ก้อนที่ 4 แรงงานคนใช้เวลาในการตรวจสอบเร็วที่สุด 41 วินาที

กรมวิชาการเกษตร

และเป็นกล้วยไม้ที่สมบูรณ์ไม่มีเพลี้ยไฟ และบั่วกล้วยไม้ โดยกล้วยไม้ก้อนที่ 9 เป็นกล้วยไม้ที่มีลักษณะพันกัน กลีบดอกซ้อนกัน ส่วนกล้วยไม้ก้อนที่ 4 และ 15 เป็นกล้วยไม้ที่มีช่อดอกตั้งตรง กลีบดอกไม่ซ้อนกัน

ตารางที่ 3 ผลการทดสอบความสามารถในการตรวจสอบของระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้เปรียบเทียบกับแรงงานคน

No. Orchid (clump)	Thrips			The quantity of position damaged by orchid midge			Detection of prototype (second/ clump)	Detection of human labors (second/ clump)
	Real value	prototype	human labors	Real value	prototype	human labors		
1	5	5	5	0	0	0	25	46
2	2	2	2	1	1	1	24	43
3	7	5	7	1	1	1	28	52
4	0	0	0	0	0	0	20	41
5	8	7	7	2	2	2	28	62
6	3	3	3	3	2	2	24	45
7	0	0	0	2	2	2	22	42
8	6	4	5	1	1	1	26	58
9	9	7	8	2	1	2	29	76
10	0	0	0	0	0	0	21	42
11	7	5	6	2	2	2	26	52
12	4	4	4	1	1	1	24	48
13	9	7	8	2	2	2	28	65
14	0	0	0	2	2	2	22	48
15	0	0	0	0	0	0	20	44
16	6	5	5	2	2	2	26	55
17	9	8	8	2	2	2	28	69
18	8	6	7	2	1	2	27	68
19	0	0	0	3	2	2	25	47
20	7	5	6	1	1	1	27	54

กรมวิชาการเกษตร

21	0	0	0	0	0	0	22	46
22	4	3	2	1	1	1	25	49
23	3	3	1	2	2	2	24	50
24	6	5	3	1	1	1	28	55
25	0	0	0	2	2	1	24	48
26	8	6	4	0	0	0	26	62
27	7	6	4	2	1	1	27	61
28	0	0	0	2	2	1	24	48
29	5	4	2	1	1	0	26	56
30	9	7	3	2	2	1	27	69
Sum	132	107	100	42	37	35	753	1,601
Error		25	32		5	7		
Mean							25.10	53.37
T-test		0.32^{NS}			0.33^{NS}		15.91^{*}	

Note: * = significant at 5% level, NS = not significant

เมื่อพิจารณาจากค่าความผิดพลาด เวลาในการตรวจสอบ และความสม่ำเสมอในการตรวจสอบ พบว่า ในช่วงแรกของการตรวจแรงงานคนมีค่าความผิดพลาดน้อยกว่า สาเหตุเกิดจากแรงงานคนสามารถตรวจเพลี้ยไฟที่หลบอยู่ตามซอกเหลือบของกลีบดอกหรือบริเวณที่ซ้อนทับกันของกลีบดอกได้โดยใช้มือ แต่เครื่องไม่มีระบบกลไกในการแหวกกลีบดอก ส่วนตำแหน่งที่ถูกทำลายจากบั่วกล้วยไม้เครื่องสามารถตรวจสอบโดยมีค่าความผิดพลาดเท่ากับแรงงานคน แต่ไม่พบบางตำแหน่งในกล้วยไม้ก้อนที่ 9 และ 18 เนื่องจากรูปร่างของดอกตูมที่เกิดจากการทำลายของบั่วกล้วยไม้มีการบิดเบี้ยวน้อยมาก แต่เครื่องสามารถเรียนรู้เพิ่มเติมได้จากการฝึกสอนให้กับแบบจำลอง ซึ่งจะทำให้การตรวจมีความแม่นยำมากขึ้น การตรวจเพลี้ยไฟโดยแรงงานคนเริ่มมีค่าความผิดพลาดมากกว่าการตรวจโดยเครื่องเมื่อตรวจกล้วยไม้ตั้งแต่ก้อนที่ 22 ขึ้นไป โดยมีค่าความผิดพลาดจากการตรวจเพลี้ยไฟสูงสุด 6 ตัว ที่กล้วยไม้ก้อนที่ 30 เช่นเดียวกับการตรวจตำแหน่งที่ถูกทำลายจากบั่วกล้วยไม้ที่มีค่าความผิดพลาดมากกว่าการตรวจโดยเครื่องตั้งแต่กล้วยไม้ก้อนที่ 25 ขึ้นไป ซึ่งแสดงให้เห็นว่าค่าความผิดพลาดจากการตรวจโดยแรงงานคนเพิ่มขึ้นเนื่องจากอาการอ่อนล้า แตกต่างจากเครื่องซึ่งความสม่ำเสมอในการตรวจสอบ และใช้เวลาในการตรวจสอบรวดเร็วกว่า เมื่อวิเคราะห์ทางสถิติ พบว่า การตรวจกล้วยไม้จำนวน 30 ก้อนความสามารถในการตรวจสอบเพลี้ยไฟ และตำแหน่งที่ถูกทำลายจากบั่วกล้วยไม้โดยเครื่องและแรงงานคนไม่มีความแตกต่างกัน ที่ระดับความเชื่อมั่น 95% แต่จะแตกต่างกันเมื่อตรวจกล้วยไม้จำนวนมากขึ้น

กรมวิชาการเกษตร

ระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ในกล้วยไม้สกุลหวายสามารถตรวจสอบครอบคลุมพื้นที่ปลูกกล้วยไม้จำนวน 60 ก้อนในโรงเรือน มีค่าใช้จ่ายในการสร้างแสดงในตารางที่ 4

ตารางที่ 4 ค่าใช้จ่ายทั้งหมดในการสร้างเครื่องต้นแบบระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ในกล้วยไม้สกุลหวาย

รายการ	จำนวนเงิน (บาท)
วัสดุทำเครื่องต้นแบบ	
1. อะลูมิเนียมสำหรับทำแขนกล	20,000
2. สายพานวี ร่อง M หน้ากว้าง 9.5 มม. พร้อมเฟืองทด ขนาด 1:1.5 จำนวน 2 คู่	3,500
3. อะลูมิเนียมสำหรับทำโครงสร้าง และระบบราง	20,000
4. เซอร์โวมอเตอร์ขนาดแรงบิด 5.8 นิวตัน-เมตร พร้อมกระปุกเกียร์อัตราทด 1:20 จำนวน 3 ตัว	12,000
5. เซอร์โวมอเตอร์ขนาดแรงบิด 2.2 นิวตัน-เมตร พร้อมกระปุกเกียร์อัตราทด 1:25 จำนวน 1 ตัว	3,000
6. โมดุลกล้อง Sony A6000	9,000
7. อุปกรณ์ไฟฟ้า และอุปกรณ์วัดคุม	40,000
8. ชุดล้อขับบนรางอะลูมิเนียมจำนวน 3 ชุด	3,000
8. อื่นๆ	10,000
รวมค่าวัสดุในการสร้างเครื่อง	120,500
ค่าแรงประกอบสร้างและอื่นๆ	5,000
รวมค่าใช้จ่ายทั้งหมดในการสร้างเครื่องต้นแบบ	125,500

กรมวิชาการเกษตร

สรุปผลการทดลองและข้อเสนอแนะ

ระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ในกล้วยไม้สกุลหวาย โดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันที่ออกแบบในการวิเคราะห์ และจำแนกภาพ พบว่า สามารถทำนายอาการทำลายจากบั่วกล้วยไม้มีค่าความแม่นยำ 78% และเพลี้ยไฟมีค่าความแม่นยำ 86% โดยมีค่า Precision Recall และ F-Measure 82% หลังจากการเรียนรู้รอบที่ 60 ไม่เกิด Overfitting

เมื่อนำระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้มาตรวจสอบกล้วยไม้จำนวน 30 ก้อนในโรงเรือนปลูกกล้วยไม้สกุลหวายเปรียบเทียบกับการใช้แรงงานคนจำนวน 1 คน พบว่า เครื่องมีความสามารถในการตรวจสอบเพลี้ยไฟ 81.1% ตำแหน่งที่ถูกทำลายจากบั่วกล้วยไม้ 88.1% เวลาในการตรวจสอบเฉลี่ย 25.10 วินาทีต่อก้อน ส่วนแรงงานคนมีความสามารถในการตรวจสอบเพลี้ยไฟ 75.8% ตำแหน่งที่ถูกทำลายจากบั่วกล้วยไม้ 83.3% เวลาในการตรวจสอบเฉลี่ย 53.37 วินาทีต่อก้อน โดยช่วงแรกของการตรวจแรงงานคนมีค่าความผิดพลาดน้อยกว่า แต่ความอ่อนล้าส่งผลให้ค่าความผิดพลาดเพิ่มขึ้น โดยผลการตรวจสอบของเครื่องเกษตรกรผู้ปลูกกล้วยไม้ และนักกีฏวิทยาของกรมวิชาการเกษตรยอมรับได้

ข้อจำกัดของเครื่องต้นแบบระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ที่พัฒนาขึ้น ผู้ใช้จำเป็นต้องป้อนตำแหน่ง (x,y,z) เพื่อให้แขนกลเคลื่อนที่ไปตรวจสอบ ไม่สามารถหาตำแหน่งที่ต้องการตรวจสอบเองได้ จึงต้องพัฒนาเครื่องต้นแบบต่อไป ส่วนผลการทดสอบเป็นการเปรียบเทียบกับผู้ชำนาญการ 1 คน จึงควรทดสอบเพิ่มเติมกับผู้ชำนาญการหลายคน เพื่อให้ผลการทดสอบมีความเที่ยงตรง และแม่นยำมากขึ้น

การนำผลงานวิจัยไปใช้ประโยชน์

ศูนย์วิจัยต่างๆของกรมวิชาการเกษตร หน่วยงานรัฐวิสาหกิจ บริษัทเอกชน เกษตรกร และกลุ่มเกษตรกรผู้ผลิตปลูกกล้วยไม้สกุลหวาย ทั้งตัดดอกเพื่อส่งขายภายในประเทศ หรือส่งออกต่างประเทศ สามารถนำระบบตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้ในกล้วยไม้สกุลหวายไปใช้งานในโรงเรือนปลูกกล้วยไม้ของตนเองได้ เนื่องจากสามารถถอดประกอบในแต่ละชิ้นส่วนได้ง่าย สายไฟ และชุดควบคุมต่างๆทำกล่องเก็บเป็นชิ้นส่วน สามารถต่อขั้วสายไฟใช้งานได้ทันที โปรแกรมที่ใช้ในระบบตรวจสอบเป็นโปรแกรมที่พัฒนาขึ้นจากภาษา Python V.3 ซึ่งสามารถหาซื้อได้ทั่วไป ผู้ใช้สามารถโหลดโปรแกรมระบบตรวจสอบ และโปรแกรมการนับ และแสดงผล แล้วทำการ Compile ผ่านโปรแกรม Python เพื่อใช้งาน การใช้งานง่าย ไม่ยุ่งยากซับซ้อน ผู้ใช้สามารถเลือกพิกัดของดอกกล้วยไม้ที่ต้องการตรวจสอบผ่านหน้าจอคอมพิวเตอร์โดยใช้เม้าส์ หรือแป้นพิมพ์ จากนั้นระบบจะทำการตรวจสอบและแสดงผลแบบ Real time นอกจากนี้สามารถเก็บข้อมูลผ่าน SD card เพื่อนำไปใช้เป็นข้อมูลสำหรับดูแนวโน้มการระบาดของเพลี้ยไฟ และบั่วกล้วยไม้ในแปลง ซึ่งช่วยให้ผู้เพาะปลูกกล้วยไม้สกุลหวายสามารถป้องกันกำจัดเพลี้ยไฟ และบั่วกล้วยไม้ได้อย่างรวดเร็ว รวมถึงลดต้นทุนการใช้สารเคมี ส่งผลให้ดอกกล้วยไม้สกุลหวายมีคุณภาพดีและเพิ่มราคาผลผลิต

กรมวิชาการเกษตร

เอกสารอ้างอิง

ตะวันส่องแสง การย์กวินพงศ์. 2561. **การจำแนกรอยโรควัณโรคปอดด้วยโครงข่ายแคปซูล.** วิทยานิพนธ์ปริญญาโท, จุฬาลงกรณ์มหาวิทยาลัย.

พิมพา ชีวาประกอบกิจ. 2562. การปรับปรุงประสิทธิภาพในการจำแนกภาพด้วยโครงข่ายประสาทแบบคอนโวลูชันโดยใช้เทคนิคการเพิ่มภาพ. **TNI Journal of Engineering and Technology.** 7(1): 59-64.

ศรีจำนรรจ์ ศรีจันทรา, ทัศนาพร ทัศคร, สุภราดา สุคนธาภิรมย์ ณ พัทลุง, ณิชกานต์ นเรวุฒิกุล, วรางคนา โชติเศรษฐี, ยุรวรรณ อนันตนมณี, พิเชฐ เชาวน์วัฒนวงศ์, ปราสาททอง พรหมเกิด, วัชรี วิทยวรรณกุล และดาราพร รินทะรักษ์. 2559. การบริหารศัตรูกล้วยไม้แบบผสมผสาน. **วารสารกีฏและสัตววิทยา.** 34(1): 2-16.

สุภราดา สุคนธาภิรมย์ ณ พัทลุง, สมศักดิ์ ศิริพลตั้งมั่น, พวงผกา อ่างมณี และวนาพร วงษ์นิคง. 2554. กลไกความต้านทานต่อสารฆ่าแมลงในเพลี้ยไฟฝ้าย, น.911-916. *ใน* **รายงานผลการวิจัยประจำปี 2554 สำนักวิจัยพัฒนาการอารักขาพืช.** กรมวิชาการเกษตร กระทรวงเกษตรและสหกรณ์, กรุงเทพฯ.

Makoto Koike. 2018. **Automatic cucumber sorting system from pictures @ Cucumber Farm in Japan (Part 2/2).** Available Source: https://mgronline.com/daily/detail/9590000091327, December 20, 2018.

Z. Zafrulla, H. Brashear, n T. Starner, H. Hamilton, and P. Presti. 2011. American Sign Language Recognition with the Kinect ICMI'11, pp. 279-286. *In* **Proceedings of the 13th international conference on multimodal interfaces.** 14 -18 November 2011, Alicante, Spain.

กรมวิชาการเกษตร

ภาคผนวก ก

การออกแบบและพัฒนาโปรแกรมสำหรับตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้

การออกแบบและพัฒนาโปรแกรมสำหรับตรวจสอบเพลี้ยไฟ และบั่วกล้วยไม้มีขั้นตอนดังนี้

1. ทำการเก็บข้อมูลเพลี้ยไฟ และบั่วกล้วยไม้ รวมถึงลักษณะกล้วยไม้ที่ถูกเพลี้ยไฟ และบั่วกล้วยไม้เข้าทำลายในรูปแบบ image เพื่อนำมาใช้เป็นข้อมูลในการทำ Machine Learning ใน Neural Network ที่สวนกล้วยไม้ชาณธนิต อำเภอเมือง จังหวัดนครปฐม พบว่า

1.1 เพลี้ยไฟ เป็นแมลงจำพวกปากดูด ลักษณะทางกายภาพมีขนาดเล็กลำตัวยาวประมาณ 1-2 มิลลิเมตร เมื่อโตเต็มวัยจะมีสีน้ำตาลและมีปีก แสดงในภาพผนวก ก1 ส่วนตัวอ่อนมีสีเหลืองอ่อน ไม่มีปีก แสดงในภาพผนวก ก2 เพลี้ยไฟจะวางไข่บริเวณใบของกล้วยไม้ ระยะเวลาตั้งแต่ตัวอ่อนถึงตัวเต็มวัยนานประมาณ 15 วัน

ภาพผนวก ก1 เพลี้ยไฟตัวเต็มวัย

ภาพผนวก ก2 เพลี้ยไฟระยะตัวอ่อน

ลักษณะการทำลายและการระบาด เพลี้ยไฟทั้งตัวอ่อนและตัวเต็มวัยจะทำลายกล้วยไม้ โดยใช้ปากเขี่ยดูดน้ำเลี้ยงที่ดอก และก้านช่อดอกทำให้กลีบดอกมีตำหนิ สีดอกจาง เกิดรอยด่างขาวทั่วไปบนกลีบดอก แสดงในภาพผนวก ก 3 พบการระบาดในทุกฤดู โดยเฉพาะในอากาศร้อนแห้งแล้งหรือฝนทิ้งช่วงนานติดต่อกัน

ภาพผนวก ก3 ลักษณะกล้วยไม้ที่ถูกเพลี้ยไฟเข้าทำลาย

1.2 บั่วกล้วยไม้ เป็นแมลงวันชนิดหนึ่ง ลักษณะทางกายภาพตัวเต็มวัยมีลำตัวสีดำ ปีก 1 คู่สีขาวใส แสดงในภาพผนวก ก4 ส่วนระยะตัวหนอนมีสีเหลือง รูปร่างค่อนข้างแบน ขนาดประมาณ 2 มิลลิเมตร แสดงในภาพผนวก ก5 ตัวเต็มวัยจะวางไข่ในเนื้อเยื่อของก้านช่อดอก ระยะเวลาตั้งแต่ตัวหนอนถึงตัวเต็มวัยนานประมาณ 34 วัน

ภาพผนวก ก4 บั่วกล้วยไม้ตัวเต็มวัย

กรมวิชาการเกษตร

ภาพผนวก ก5 บั่วกล้วยไม้ระยะตัวหนอน

ลักษณะการทำลายและการระบาด บั่วกล้วยไม้ในระยะตัวหนอนจะกัดกินกลีบดอกด้านในบริเวณเกสร ทำให้กลีบดอกเกิดอาการผิดปกติ ดอกตูมหยุดการเจริญเติบโต และบิดเบี้ยว ต่อมาดอกจะมีอาการเน่าเหลือง ฉ่ำน้ำ และหลุดร่วงจากช่อดอก แสดงในภาพผนวก ก6 ถ้าการระบาดรุนแรงดอกตูมจะหลุดร่วงอย่างรวดเร็วจนเหลือแต่ก้านดอก พบการระบาดในทุกฤดูโดยเฉพาะในฤดูฝน

ภาพผนวก ก6 ลักษณะกล้วยไม้ที่ถูกบั่วกล้วยไม้เข้าทำลาย

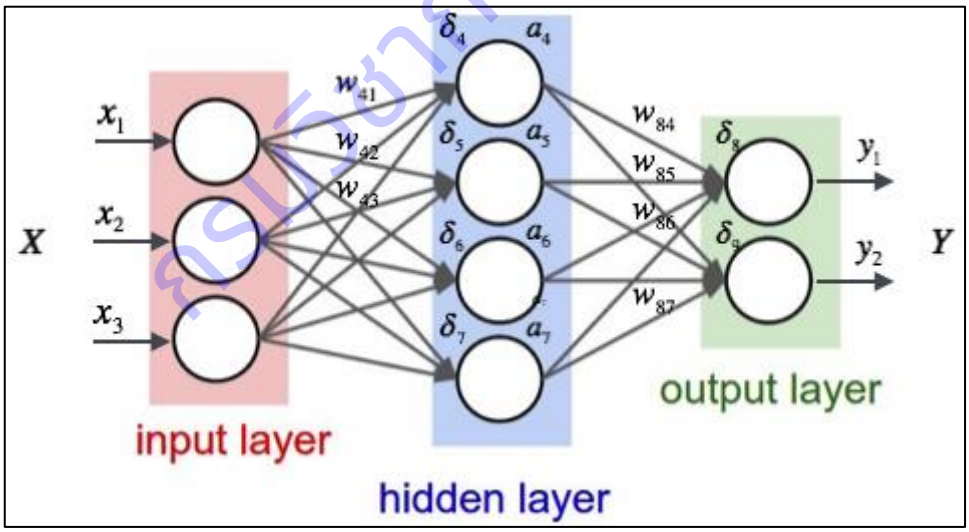
กรมวิชาการเกษตร

นำภาพของเพลี้ยไฟระยะตัวอ่อน และตัวเต็มวัย บั่วกล้วยไม้ระยะตัวหนอน และตัวเต็มวัยรวมถึงลักษณะกล้วยไม้ที่ถูกเพลี้ยไฟ และบั่วกล้วยไม้เข้าทำลายที่เก็บข้อมูลทั้งหมด (ตรงตาม KPIsไตรมาศที่ 1) มาทำการจำแนกรูปภาพ (Image Classification) ซึ่งเป็นงานหนึ่งที่ต้องใช้ความรู้ในการพัฒนาขั้นตอนวิธีเพื่อให้คอมพิวเตอร์สามารถ “**จำแนก แยกแยะ จัดประเภทรูปภาพต่างๆ**” ได้ตรงตามความต้องการ

Image Classification จัดเป็น Machine Learning แบบ Supervised Learning กล่าวคือ การที่ให้คอมพิวเตอร์เรียนรู้รูปภาพต่างๆ โดยมีข้อมูลสำหรับการฝึกเรียนรู้และจำให้กับคอมพิวเตอร์

2. การทำ Machine Learning โดยใช้เทคนิค Deep Learning Model

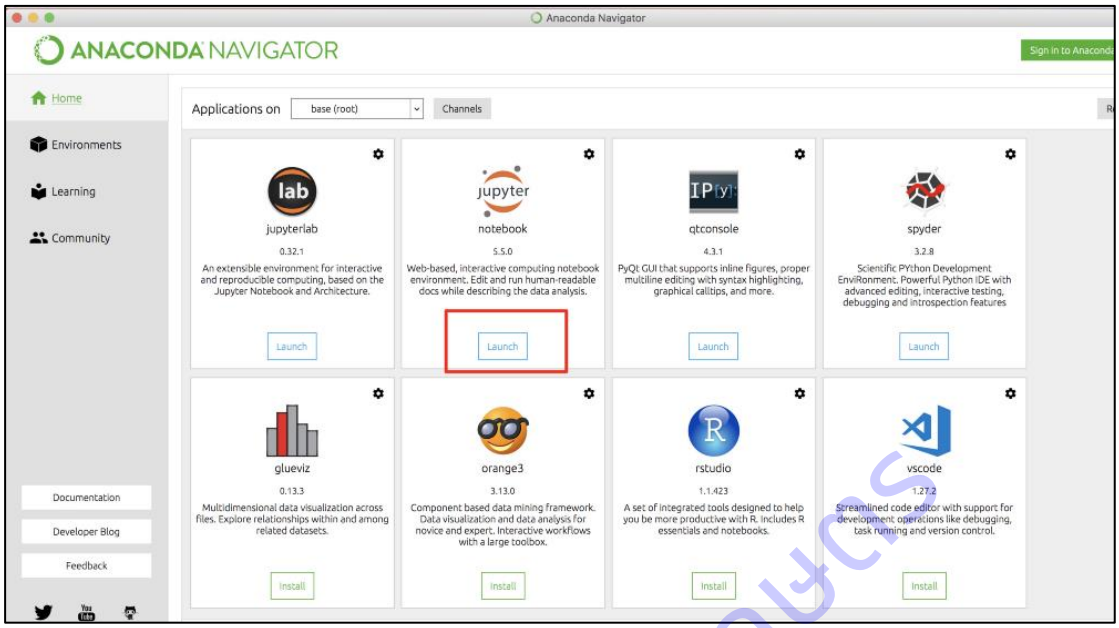
ในส่วนของขั้นตอนวิธี (Algorithm) ที่จะทำให้คอมพิวเตอร์เรียนรู้และจดจำรูปภาพ แล้วจำแนกประเภทรูปภาพได้นั้น โครงการวิจัยนี้ใช้ **Neural Network** ซึ่งเป็น Model ที่เลียนแบบโครงข่ายประสาทของมนุษย์ โดยจุดเด่นของ Model นี้ คือสามารถปรับค่าตัวมันเองให้เข้ากับข้อมูล (ข้อมูลชุดฝึกสอน) ที่ได้เรียนรู้ แสดงในภาพผนวก ก7 เมื่อผู้ออกแบบฝึกฝนจนเป็นที่พอใจแล้วสุดท้ายก็จะได้ Model ที่สามารถบอกประเภทของรูปภาพต่างๆ ได้



ภาพผนวก ก7 Neural Network Model

โปรแกรมที่ใช้พัฒนาคือ Python V.3 , Anaconda ไลบรารี่ที่ใช้ NumPy, OpenCV, Scikit-Learn และ Keras หลังจากที่ติดตั้ง Python กับ Anaconda แล้วให้เปิดตัว Anaconda Navigator ขึ้นมาแล้วเลือก Launch ตัว jupyter notebook ซึ่ง jupyter notebook จะคล้าย IDE ที่ให้ผู้ใช้สามารถพิมพ์คำสั่งแล้วให้ประมวลผลคำสั่ง

นั้นได้ และยังแสดงผลลัพธ์ต่อท้ายชุดคำสั่งของช่องนั้นๆ ตัว notebook สามารถเก็บ code และตัว markdown ที่ช่วยให้แสดงข้อมูลในแบบต่างๆได้ด้วย แสดงในภาพผนวก ก8



ภาพผนวก ก8 Anaconda Navigator

หลังจากเลือก Launch แล้ว Anaconda จะไปเปิดตัว jupyter notebook ขึ้นมาและ notebook server จะรันที่ http://localhost:8888/ แสดงในภาพผนวก ก9

ภาพผนวก ก9 jupyter notebook

หลังจากเปิด jupyter notebook ขึ้นมาแล้ว ให้สร้าง folder เพื่อที่จะเก็บไฟล์และข้อมูลต่างๆสำหรับการทำ project ในครั้งนี้ หลังจากสร้าง folder แล้วให้คลิกที่ New แล้วเลือก Python 3 เพื่อสร้างไฟล์ใหม่แสดงในภาพผนวก ก10

ภาพผนวก ก10 การสร้างไฟล์ใน Python 3

ภาพที่ใช้ในการสอนแบบจำลองเป็นภาพถ่ายในโรงเรือนปลูกกล้วยไม้สกุลหวายจำนวน 840 ภาพ แต่ละภาพมีขนาด 128x128 พิกเซล แบ่งเป็น 2 หมวด คือ ภาพเพลี้ยไฟจำนวน 290 ภาพ และภาพแสดงอาการทำลายจากบั่วกล้วยไม้จำนวน 382 ภาพ รวมทั้งหมด 672 ภาพในการฝึกสอนให้แบบจำลองเรียนรู้จำนวน 100 รอบ ส่วน 168 ภาพใช้ทดสอบความถูกต้องของแบบจำลอง นำภาพทั้งหมดมาไว้ในโฟลเดอร์เดียวกับไฟล์ python เพื่อให้ง่ายต่อการเรียกใช้ โดยตั้งชื่อโฟลเดอร์ใหม่ว่า orchids&insects จากนั้นทำการอ่านภาพจากโฟลเดอร์ตามโค้ดด้านล่าง

```
import cv2
import matplotlib.pyplot as plt
from imutils import paths
image_paths = list(paths.list_images('datasets/orchids&insects/'))
image = cv2.imread(image_paths[10])
new_image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
plt.figure()
plt.imshow(new_image)
```

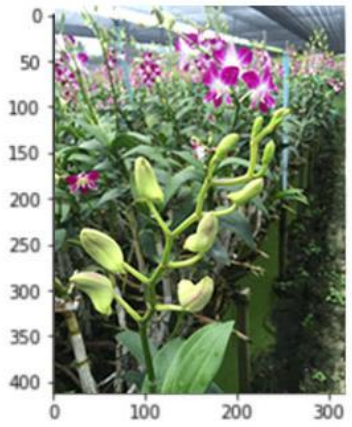
ผลลัพธ์ที่ได้จากการรันแสดงในภาพผนวก ก11

กรมวิชาการเกษตร

```
In [2]: import cv2
        import matplotlib.pyplot as plt
        from imutils import paths

        image_paths = list(paths.list_images('datasets/orchids&insects/'))
        image = cv2.imread(image_paths[10])
        new_image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
        plt.figure()
        plt.imshow(new_image)

Out[2]: <matplotlib.image.AxesImage at 0x1134f1a20>
```



ภาพผนวก ก11 อ่านภาพจากโฟลเดอร์ orchids&insects ใน Python 3

จากนั้นทำการปรับภาพ ความละเอียด และสีของทุกภาพจาก 128x128 พิกเซล เป็น 32x32 พิกเซล เพื่อสะดวกในการเทรนข้อมูลตามโค้ดด้านล่าง (ภาพผนวก ก12)

```
import random
import numpy as np
random.shuffle(image_paths)
data = []
labels = []
for image_path in image_paths:
        image = cv2.imread(image_path)
        image = cv2.resize(image, (32, 32))
        labels.append(image_path.split('/')[-2])
        data.append(image)
data = np.array(data)
```

กรมวิชาการเกษตร

```
labels = np.array(labels)
data = data.reshape((3000, 3072))
data = data/255.0
```

```
In [3]: import random
        import numpy as np

        random.shuffle(image_paths)
        data = []
        labels = []

        for image_path in image_paths:
            image = cv2.imread(image_path)
            image = cv2.resize(image, (32, 32))
            labels.append(image_path.split('/')[-2])
            data.append(image)

        data = np.array(data)
        labels = np.array(labels)
        data = data.reshape((3000, 3072))
        data = data/255.0

In [ ]:
```

ภาพผนวก ก12 การปรับลักษณะของภาพให้เหมาะสมสำหรับการเทรนข้อมูล

หลังการปรับภาพนำภาพมาแปลงเป็นข้อมูลตัวเลขฐานสองในเมทริกซ์ขนาด 1x3 ตามโค้ดด้านล่าง (ภาพผนวก ก13)

```
from sklearn.preprocessing import LabelBinarizer
lb = LabelBinarizer()
labels = lb.fit_transform(labels)
```

กรมวิชาการเกษตร

```
labels = lb.fit_transform(labels)
print(labels)
```

```
[[0 0 1]
 [0 1 0]
 [0 1 0]
 ...
 [0 1 0]
 [0 0 1]
 [0 1 0]]
```

ภาพผนวก ก13 แปลงภาพให้เป็นข้อมูลตัวเลขฐานสองในเมทริกซ์ขนาด 1x3

เมื่อแปลงภาพเป็นตัวเลขคอมพิวเตอร์จะเข้าใจค่า labels โดยรู้ว่า orchid midge เป็น [1 0 0] และ thrips เป็น [0 0 1] แสดงในภาพผนวก ก14

```
In [6]: from sklearn.preprocessing import LabelBinarizer
        lb = LabelBinarizer()

        labels = lb.fit_transform(labels)
        lb.classes_

Out[6]: array(['orchid midge', 'thrip'], dtype='<U5')
```

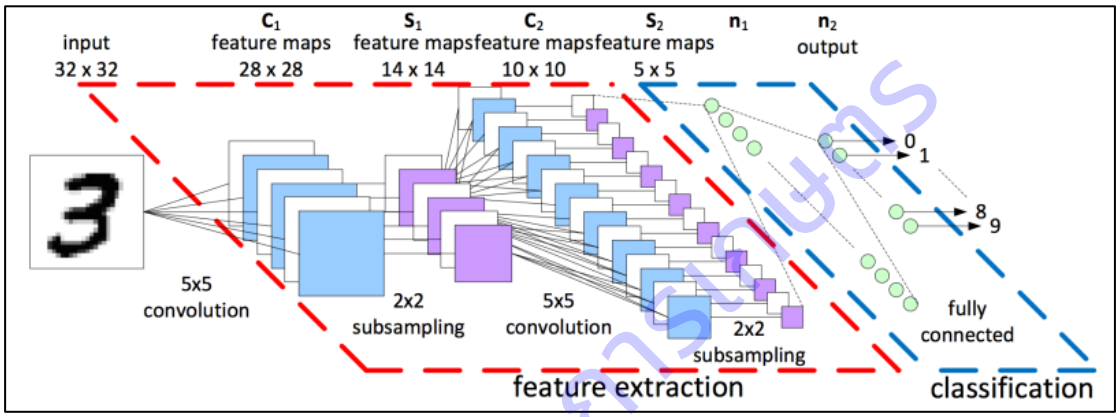
ภาพผนวก ก14 เลขฐานสองในเมทริกซ์ขนาด 1x3 ของ orchid midge และ thrips

เมื่อพิมพ์คำสั่ง lb.classes_ เพื่อดูค่า labels ที่สามารถจำแนกประเภท โดยค่าที่จำแนกประเภทออกมามี 2 ประเภทก็คือ orchid midge และ thrips สามารถพิสูจน์ได้ว่า labels สามารถจำแนกคำตอบออกมาได้จริงหรือไม่ ให้ใช้คำสั่ง lb.inverse_transform ซึ่งเป็นคำสั่งการแปลงจากตัวเลขกลับไปเป็นตัวหนังสือที่เป็นค่าก่อนที่จะเปลี่ยนตัวเลขนั่นเอง

```
lb.inverse_transform(np.array([[0.6, 0.4, 0.2]]))
#ถ้าเราพิมพ์คำสั่งนี้ไป เราจะได้คำตอบเป็น orchid midge เพราะค่า [0.6, 0.4, 0.2] มีค่าใกล้ [1, 0, 0]
output: array(['orchid midge'], dtype='<U5')
lb.inverse_transform(np.array([[0.6, 0.5, 0.9]]))
#ส่วนอันนี้ก็จะได้คำตอบเป็น thrips เพราะค่า [0.6, 0.5, 0.9] มีค่าใกล้ [0, 0, 1]
output: array(['thrips'], dtype='<U5')
```

กรมวิชาการเกษตร

หลังจากมี data สำหรับสอนแบบจำลองกับ labels ที่เป็นแม็พคำตอบจะนำข้อมูลเข้าฟังก์ชั่นเทรน โดยโครงการวิจัยนี้โจทย์ Image Classification ไม่ได้ใช้ Neural Network ทั่วๆไป แต่ใช้ CNN (Convolutional Neural Network) ซึ่งเป็นรูปแบบหนึ่งของ feed-forward และจัดเป็น Deep Learning ประเภทหนึ่ง โดยส่วนของ hidden layer ของ CNN จะมี Convolutional Layers เพิ่มขึ้นมาเป็นส่วนที่ใช้ในการฟิลเตอร์รูปโดยมี Kernel function เพื่อใช้แปลงคุณลักษณะ และแยกองค์ประกอบออกมา เช่น ขอบรูป สี ลักษณะรูปทรง เป็นต้น และมี Activation Function ที่ช่วยแปลงค่าให้อยู่ในรูปที่ง่ายต่อการคำนวนและได้ผลลัพธ์ดียิ่งขึ้น รวมถึงส่วนที่เรียกว่า Pooling โดยส่วนนี้จะทำหน้าที่ปรับไซส์ของข้อมูลให้มีขนาดเล็กลงโดยที่รายละเอียดของข้อมูลนั้นยังคงเดิม (ยังคงอยู่ไว้ได้มากที่สุด) และในส่วนสุดท้ายคือ Fully connected layer ที่เชื่อมต่อแต่ละ perceptron ในแต่ละชั้นเข้าด้วยกันแสดงในภาพผนวก ก15



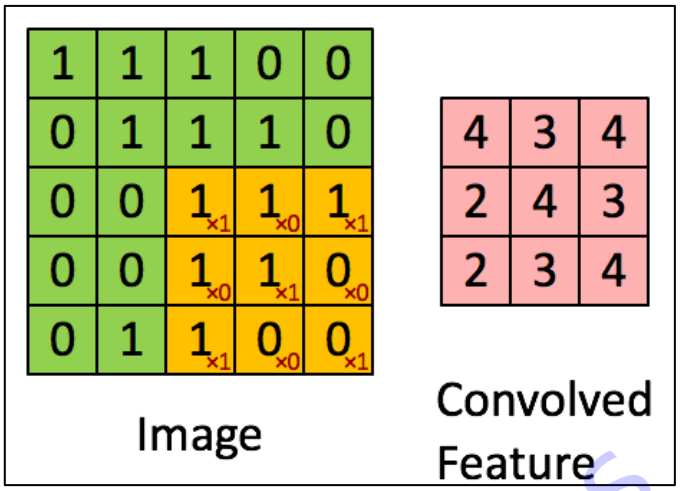
ภาพผนวก ก15 Convolutional Neural Network: CNN. or ConvNet.

ขั้นตอนต่อมาเป็นการสร้างแบบจำลองในโครงการนี้ตั้งชื่อว่า orchids&insects โดยใช้ keras เขียนเป็นโค้ดดังนี้

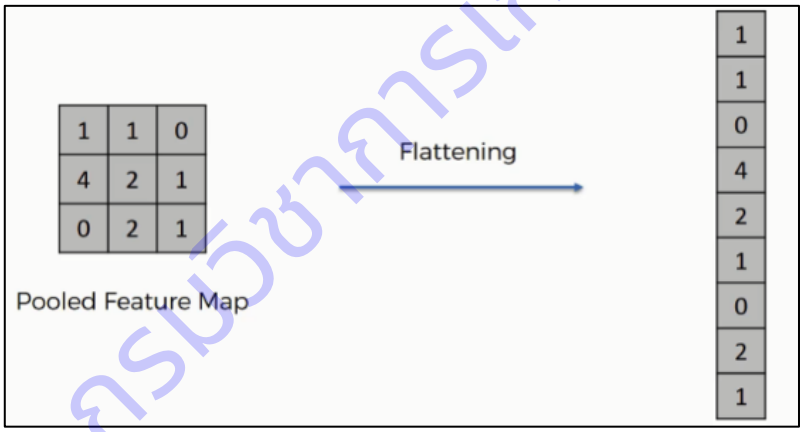
```
from keras import backend
from keras.models import orchids&insects
from keras.layers.convolutional import Conv2D
from keras.layers.core import Dense, Flatten
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(data, labels, test_size=0.2, random_state=5)
model = orchids&insects
model.add(Conv2D(32, (3, 3), input_shape=(32, 32 ,3), activation='relu', padding='same'))
model.add(Flatten())
model.add(Dense(3, activation='softmax'))
```

กรมวิชาการเกษตร

การทำงานของฟิลเตอร์ Kernel แสดงในภาพผนวก ก16 และการทำงานของ Flatten แสดงในภาพผนวก ก17



ภาพผนวก ก16 การทำงานของฟิลเตอร์ Kernel



ภาพผนวก ก17 การทำงานของ Flatten

โค้ดโปรแกรมแสดงในภาพผนวก ก18 และรายละเอียดภายในแบบจำลองแสดงในภาพผนวก ก19

กรมวิชาการเกษตร

```
In [5]: from keras import backend
        from keras.models import orchids&insects
        from keras.layers.convolutional import Conv2D
        from keras.layers.core import Dense, Flatten
        from sklearn.model_selection import train_test_split

        X_train, X_test, y_train, y_test = train_test_split(data, labels, test_size=0.2, random_state=5)

        model = orchids&insects
        model.add(Conv2D(32, (3, 3), input_shape=(32, 32 ,3), activation='relu', padding='same'))
        model.add(Flatten())
        model.add(Dense(3, activation='softmax'))
```

ภาพผนวก ก18 โค้ดในโปรแกรม python สำหรับการใช้ไลบรารี keras

```
In [6]: model.summary()

_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 32, 32, 32)        896
_________________________________________________________________
flatten_1 (Flatten)          (None, 32768)             0
_________________________________________________________________
dense_1 (Dense)              (None, 3)                 98307
=================================================================
Total params: 99,203
Trainable params: 99,203
Non-trainable params: 0
_________________________________________________________________
```

ภาพผนวก ก19 รายละเอียดภายในแบบจำลองที่สร้าง

ทำการเซ็ทค่ากระบวนการเรียนรู้ให้กับแบบจำลองก่อนจะนำไปใช้เทรนข้อมูลด้วยฟังก์ชั่น compile มีโค้ดดังนี้

```
model.compile(loss='categorical_crossentropy', optimizer='adam')
model.fit(X_train, y_train, epochs=10, validation_split=0.2)
```

หลังจากการรันคำสั่งจะเห็นผลการรันข้อมูลแต่ละ epoch ขึ้นมา และจะเห็นว่ามีข้อมูล เทรนทั้งหมด 840 ตัวอย่าง โดยมีข้อมูลที่ใช้สำหรับการทำ validation 168 ตัวอย่างแสดงในภาพผนวก ก20

กรมวิชาการเกษตร

```
In [7]: model.compile(loss='categorical_crossentropy', optimizer='adam')
        model.fit(X_train, y_train, epochs=10, validation_split=0.2)

        Train on 840 samples, validate on 168 samples
        Epoch 1/10
        1920/1920 [==============================] - 2s 952us/step - loss: 0.9770 - val_loss: 0.7679
        Epoch 2/10
        1920/1920 [==============================] - 1s 402us/step - loss: 0.6900 - val_loss: 0.6954
        Epoch 3/10
        1920/1920 [==============================] - 1s 329us/step - loss: 0.5945 - val_loss: 0.6858
        Epoch 4/10
        1920/1920 [==============================] - 1s 485us/step - loss: 0.4927 - val_loss: 0.7360
        Epoch 5/10
        1920/1920 [==============================] - 1s 384us/step - loss: 0.4325 - val_loss: 0.7650
        Epoch 6/10
        1920/1920 [==============================] - 1s 362us/step - loss: 0.3537 - val_loss: 0.6620
        Epoch 7/10
        1920/1920 [==============================] - 1s 514us/step - loss: 0.3049 - val_loss: 0.7201
        Epoch 8/10
        1920/1920 [==============================] - 1s 514us/step - loss: 0.2580 - val_loss: 0.7172
        Epoch 9/10
        1920/1920 [==============================] - 1s 457us/step - loss: 0.2135 - val_loss: 0.6680
        Epoch 10/10
        1920/1920 [==============================] - 1s 351us/step - loss: 0.1859 - val_loss: 0.7063

Out[7]: <keras.callbacks.History at 0x1a1dfaae10>
```

ภาพผนวก ก20 แสดงผลการรันข้อมูลแต่ละ epoch

ภายหลังจากการเทรนแบบจำลองจะใช้ข้อมูลเทสมาทดสอบให้แบบจำลองทำการ classified โดยใช้ฟังก์ชั่น predict

```
y_pred = model.predict(X_test)
from sklearn.metrics import accuracy_score, classification_report
accuracy_score(y_test.argmax(axis=1), y_pred.argmax(axis=1))
```

ที่ต้องใช้ y_test.argmax(axis=1), y_pred.argmax(axis=1) ในการใช้ฟังก์ชั่น accuracy_score เนื่องจากค่า y_pred ที่ได้จาก model.predict เป็นค่าที่ไม่เรียงต่อกัน และไม่สามารถเอามาเป็นตัวบ่งชี้การ classification ได้ ถ้าลองใส่ค่าฟังก์ชั่น accuracy_score เป็น y_test และ y_pred แบบธรรมดาจะพบกับ error แบบข้างล่างนี้

```
ValueError: Classification metrics can't handle a mix of multilabel-indicator and multiclass
targets
```

ความหมายคือไม่สามารถใช้ label ที่เป็นค่าผสมแบบนี้ได้ ต้องใช้ฟังก์ชั่น argmax(axis=1) ของ numpy เพื่อหา index ที่มีค่ามากที่สุดของอาร์เรย์แต่ละช่อง และค่าถูกจัดกลุ่ม โดย y_pred เป็นค่าอาร์เรย์ 2 มิติ axis=1 คืออาเรย์มิติที่ 2 ของตัว y_pred นั่นเอง (ถ้าเป็นมิติแรก axis=0) แสดงในภาพผนวก ก21

กรมวิทยาศาสตร์

```
y_pred.argmax(axis=1)
```

```
Out[8]: array([1, 1, 0, 0, 0, 0, 1, 0, 0, 2, 2, 1, 2, 0, 2, 0, 1, 1, 0, 2, 1, 2,
               0, 0, 1, 1, 1, 2, 1, 2, 2, 2, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1,
               1, 2, 1, 2, 2, 1, 0, 1, 1, 1, 0, 2, 2, 2, 0, 0, 2, 2, 1, 1, 1, 2,
               2, 0, 2, 1, 2, 1, 0, 1, 1, 1, 2, 0, 0, 1, 1, 2, 1, 2, 1, 0, 1, 1,
               0, 1, 1, 1, 1, 0, 1, 1, 2, 1, 1, 2, 1, 1, 1, 0, 0, 1, 1, 2, 1, 1,
               1, 0, 2, 0, 2, 2, 0, 1, 1, 2, 2, 2, 2, 0, 1, 1, 2, 2, 0, 2, 2, 2,
               0, 0, 2, 1, 1, 1, 1, 0, 2, 1, 0, 2, 2, 2, 0, 0, 0, 0, 2, 2, 1, 0,
               0, 2, 0, 2, 2, 1, 2, 0, 0, 1, 2, 0, 2, 0, 2, 1, 0, 0, 2, 2, 2, 2,
               0, 1, 2, 1, 1, 1, 1, 2, 1, 2, 0, 1, 0, 2, 2, 0, 2, 1, 1, 0, 1, 2,
               1, 0, 1, 1, 1, 1, 0, 0, 2, 0, 2, 2, 1, 0, 1, 2, 1, 2, 1, 2, 0, 2,
               2, 1, 1, 1, 2, 0, 2, 1, 2, 2, 2, 0, 0, 2, 2, 0, 2, 2, 0, 0, 0, 1,
               2, 2, 2, 0, 2, 1, 0, 2, 0, 0, 1, 0, 0, 2, 0, 1, 0, 1, 0, 1, 0, 2,
               0, 0, 0, 0, 2, 0, 2, 2, 1, 1, 2, 0, 1, 2, 1, 1, 2, 0, 0, 1, 2, 2,
               1, 1, 2, 2, 2, 0, 0, 2, 1, 1, 2, 0, 2, 0, 1, 2, 2, 2, 1, 0, 2, 1,
               0, 2, 2, 0, 2, 1, 0, 2, 1, 0, 1, 1, 0, 1, 2, 0, 1, 1, 2, 1, 1, 1,
               2, 1, 1, 0, 2, 2, 2, 2, 0, 2, 1, 1, 0, 1, 1, 2, 0, 0, 1, 0, 2, 0,
               0, 0, 2, 1, 1, 1, 2, 1, 2, 1, 2, 0, 0, 2, 2, 2, 0, 0, 0, 1, 1, 0,
               1, 1, 1, 1, 2, 0, 2, 0, 2, 0, 1, 2, 0, 2, 1, 0, 0, 2, 1, 2, 0, 0,
               1, 2, 2, 2, 1, 2, 2, 0, 0, 1, 2, 0, 1, 2, 1, 0, 0, 1, 1, 2, 2, 2,
               0, 0, 2, 1, 2, 2, 2, 2, 0, 1, 0, 1, 1, 1, 0, 0, 2, 1, 1, 2, 2, 1,
               2, 1, 2, 0, 2, 0, 2, 2, 1, 2, 0, 0, 1, 2, 0, 2, 2, 1, 1, 2, 2, 1,
               0, 2, 1, 1, 0, 2, 1, 1, 1, 1, 2, 1, 0, 0, 0, 1, 0, 0, 2, 2, 2, 2,
               2, 0, 0, 2, 0, 0, 0, 2, 1, 2, 0, 1, 0, 1, 0, 0, 2, 0, 1, 1, 0, 0,
               1, 2, 1, 2, 2, 2, 2, 2, 2, 0, 0, 1, 0, 1, 0, 2, 1, 1, 1, 1, 1, 0,
               0, 2, 2, 2, 1, 2, 2, 1, 2, 0, 2, 0, 0, 1, 0, 1, 0, 0, 2, 0, 1, 0,
               1, 2, 1, 2, 2, 2, 2, 2, 1, 1, 2, 2, 1, 0, 2, 0, 0, 2, 2, 0, 0, 0,
               2, 2, 0, 0, 2, 2, 2, 2, 1, 2, 2, 0, 1, 0, 0, 1, 2, 2, 2, 0, 1, 0,
               0, 2, 1, 0, 1, 1])
```

ภาพผนวก ก21 ค่าที่ได้จากการใช้ฟังก์ชั่น argmax(axis=1)

หลังจากรัน accuracy_score จะเห็นความถูกต้องที่ได้ 0.820 แสดงในแสดงในภาพผนวก ก22

```
In [10]: from sklearn.metrics import accuracy_score, classification_report

         accuracy_score(y_test.argmax(axis=1), y_pred.argmax(axis=1))
```

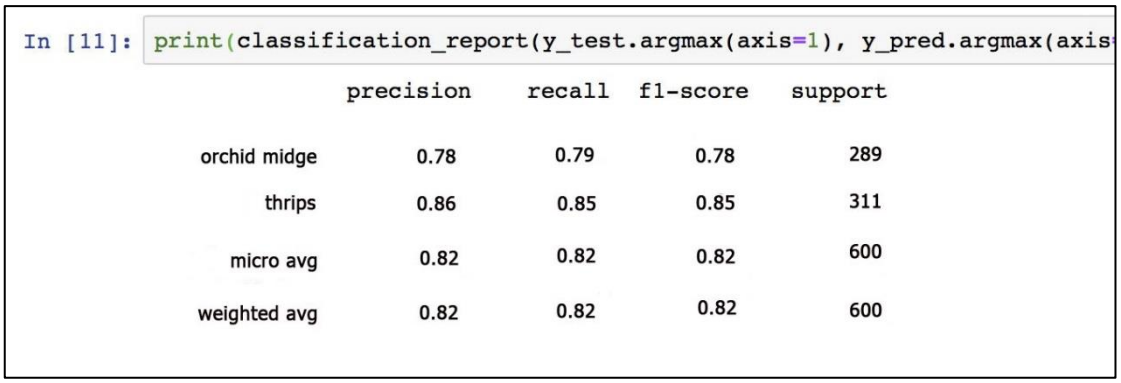
```
Out[10]: 0.820
```

ภาพผนวก ก22 ค่า accuracy_score = 0.820

แสดง classification อย่างละเอียดโดยใช้โค้ดด้านล่าง และผลการรันข้อมูลแสดงในภาพผนวก ก23

print(classification_report(y_test.argmax(axis=1), y_pred.argmax(axis=1),
target_names=lb.classes_))

กรมวิชาการเกษตร



```
In [11]: print(classification_report(y_test.argmax(axis=1), y_pred.argmax(axis
```

	precision	recall	f1-score	support
orchid midge	0.78	0.79	0.78	289
thrips	0.86	0.85	0.85	311
micro avg	0.82	0.82	0.82	600
weighted avg	0.82	0.82	0.82	600

ภาพผนวก ก23 แสดงรายละเอียด classification

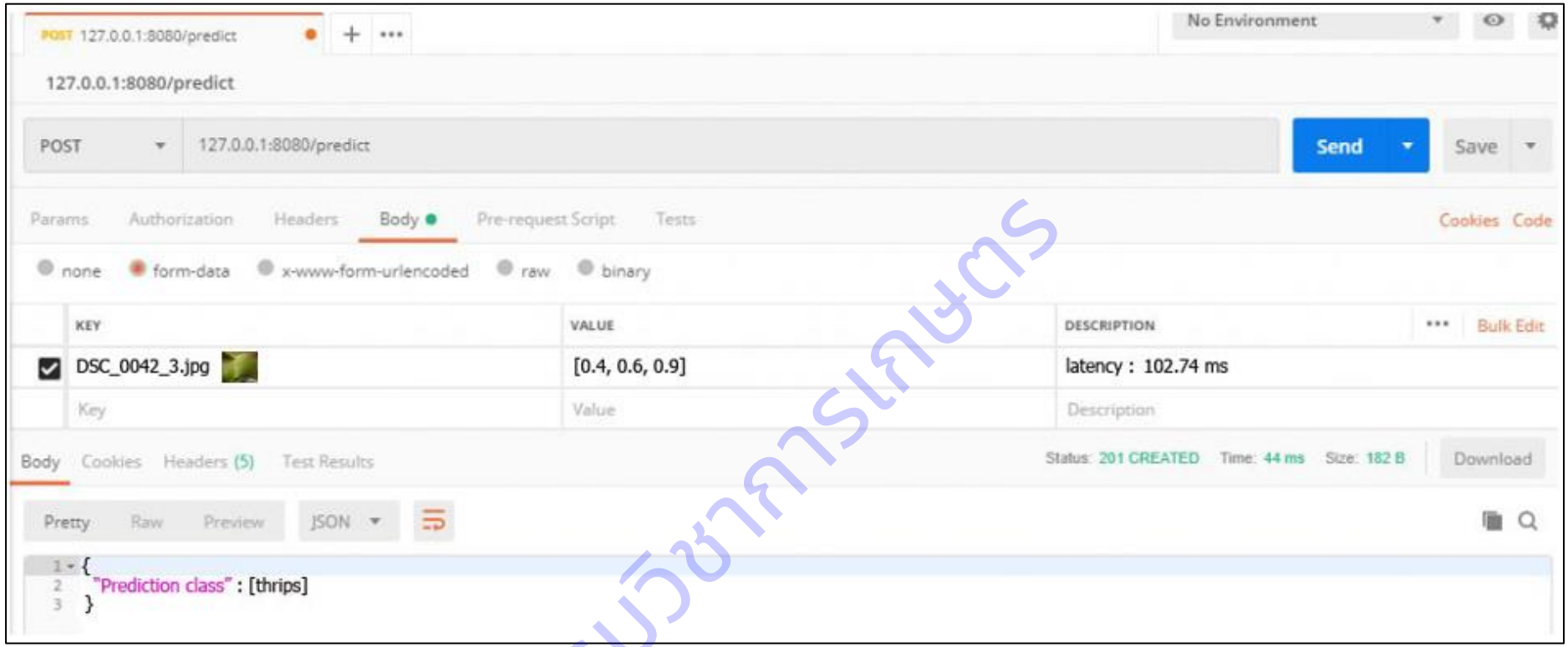
เมื่อแสดงค่าออกมาจะเห็นว่าจากผลการทำนายค่า precision มีความแม่นยำในการจำแนกเฉลี่ย 82% โดยแบ่งเป็น 2 ประเภท คือ orchid midge มีความแม่นยำในการจำแนกอยู่ 78% และ thrips มีความแม่นยำในการจำแนกอยู่ 86%

แบบจำลองที่ใช้ในการเรียนรู้มีประสิทธิภาพค่า “Accuracy” เท่ากับ **82.06%** ทำการ save แบบจำลองจะได้แบบจำลองชื่อ orchids&insects_model จาก Keras แล้วนำไปใช้ใน Production Server โดยเรียกใช้งานด้วยคำสั่ง

```
from keras.models import load_model
model = load_model('./orchids&insects_model')
```

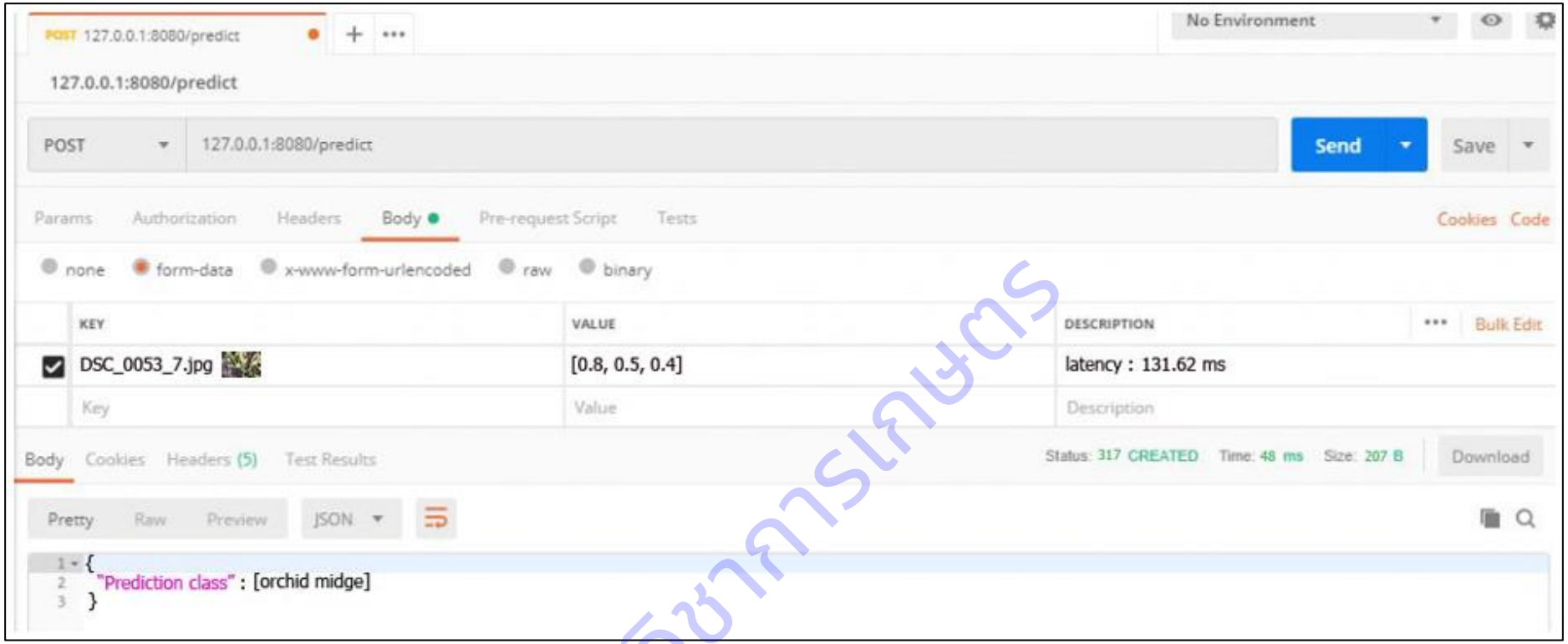
แบบจำลอง orchids&insects_model นำไปใช้ใน Production Server จะมีหน้าตาของโปรแกรมตามภาพผนวก ก24 และ ก25

กรมวิชาการเกษตร



ภาพผนวก ก24 โปรแกรมระบบตรวจสอบทำการระบุภาพเพลี้ยไฟที่ทำการตรวจพบ

กรมวิชาการเกษตร



ภาพผนวก ก25 โปรแกรมระบบตรวจสอบทำการระบุภาพบั่วกล้วยไม้ที่ทำการตรวจพบ

กรมวิชาการเกษตร