



การศึกษาการตัดหัวปลาทูน่าโดยใช้เทคโนโลยีภาพ
A Study of AI for Tuna Deheading by using Image Processing

โดย

นางสาว ถลัชนันท์ สลัดทุกข์ รหัสนักศึกษา 5910110122

โครงการวิศวกรรมเครื่องกล ปีการศึกษา 2/2563
ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์
มหาวิทยาลัยสงขลานครินทร์

การศึกษาการตัดหัวปลาทูน่าโดยใช้เทคโนโลยีภาพ
A Study of AI for Tuna Deheading by using Image Processing

โดย

นางสาว ถลัชนันท์ สลัดทุกข์ รหัสนักศึกษา 5910110122

โครงการวิศวกรรมเครื่องกล ปีการศึกษา 2/2563
ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์
มหาวิทยาลัยสงขลานครินทร์

ตามหลักสูตรวิศวกรรมศาสตรบัณฑิตสาขาวิศวกรรมเมคคาทรอนิกส์

ได้รับการพิจารณาเห็นชอบโดย

อาจารย์ที่ปรึกษาโครงการ

.....

(ดร.จุฑามณี อ๋อยสกุล)

.....

(ดร.กุลภัสร์ ทองแก้ว)

กรรมการ

.....

(ดร.จุฑามณี อ๋อยสกุล)

.....

(ผศ.ดร.มัทฑาร์ แวหะยี่)

.....

(ดร.จีระภา สุขแก้ว)

.....

(รศ.ดร.สุธรรม นียมवास)

ผู้มีอำนาจลงนามของบริษัท (ถ้ามี)

.....

(.....)



หนังสือยินยอมมอบลิขสิทธิ์ สิทธิบัตร/อนุสิทธิบัตร/ ความลับทางการค้า
และทรัพย์สินทางปัญญาอื่น ๆ ในโครงงานนักศึกษาให้แก่มหาวิทยาลัยสงขลานครินทร์

ที่...../.....

เขียนที่มหาวิทยาลัยสงขลานครินทร์
วันที่.....เดือน.....พ.ศ.....

ข้าพเจ้า นางสาว ถลัชนันท์ สลัดทุกข์ รหัสประจำตัว 5910110122 เป็นนักศึกษาระดับปริญญาตรีคณะ
วิศวกรรมศาสตร์สาขาวิชาวิศวกรรมเมคคาทรอนิกส์ มหาวิทยาลัยสงขลานครินทร์ ตกลงยินยอมมอบลิขสิทธิ์
สิทธิบัตร/อนุสิทธิบัตร สิทธิในงานประดิษฐ์ ความลับทางการค้า และทรัพย์สินทางปัญญาอื่น ๆ ในโครงงาน
นักศึกษาที่ได้รับอนุมัติตามหัวข้อเรื่องการศึกษาการตัดหัวปลาทูนาโดยใช้เทคโนโลยีภาพ (A Study of AI for
Tuna Deheading by using Image Processing)

ที่มีคณะกรรมการที่ปรึกษา ดังนี้

- | | | |
|-----------------|---------|---------------------------------|
| 1. ดร.จุฑามณี | อัยสกุล | สังกัด ภาควิชาวิศวกรรมเครื่องกล |
| 2. ผศ.ดร.มังกร | แหวะยี | สังกัด ภาควิชาวิศวกรรมเครื่องกล |
| 3. ดร.จีระภา | สุขแก้ว | สังกัด ภาควิชาวิศวกรรมเครื่องกล |
| 4. รศ.ดร.สุธรรม | นิยมवास | สังกัด ภาควิชาวิศวกรรมเครื่องกล |

ให้แก่มหาวิทยาลัยสงขลานครินทร์ ตลอดอายุการคุ้มครองตามกฎหมายของทรัพย์สินทางปัญญาที่เกี่ยวข้อง
(ลงนาม).....นักศึกษา

(นางสาว ถลัชนันท์ สลัดทุกข์)

(ลงนาม).....อาจารย์ที่ปรึกษา (พยาน)

(ดร.จุฑามณี อัยสกุล)

(ลงนาม).....อาจารย์ที่ปรึกษา (พยาน)

(ดร.กุลภัสร์ ทองแก้ว)

(ลงนาม).....ประธานคณะกรรมการบริหารหลักสูตร/หัวหน้าภาควิชา (พยาน)

(ผศ.ดร. ชีระยุทธ หลีวีจิตร)

ข้อตกลงว่าด้วยการโอนสิทธิในทรัพย์สินทางปัญญาของนักศึกษา

- ข้อ 1 นักศึกษาสามารถนำผลงานวิทยานิพนธ์/สารนิพนธ์/ปัญหาพิเศษ/โครงการ หรือส่วนหนึ่งส่วนใดไปเผยแพร่ในรูปแบบใดๆ เพื่อเป็นผลงานทางวิชาการที่เป็นส่วนหนึ่งของการศึกษาได้โดยจะต้องระบุว่าเป็นผลงานของมหาวิทยาลัยสงขลานครินทร์ทุกครั้งที่มีการเผยแพร่
- ข้อ 2 กรณีที่นักศึกษามีความประสงค์นำผลงานวิทยานิพนธ์/สารนิพนธ์/ปัญหาพิเศษ/โครงการไปเผยแพร่ หรืออนุญาตให้ผู้อื่นทำซ้ำ หรือดัดแปลง หรือเผยแพร่ต่อสาธารณชนหรือกระทำการอื่นใด ตามพระราชบัญญัติลิขสิทธิ์ พ.ศ. 2537 โดยมีค่าตอบแทนหรือนำไปใช้ประโยชน์ในเชิงพาณิชย์จะต้องได้รับความยินยอมเป็นลายลักษณ์อักษรจากมหาวิทยาลัยสงขลานครินทร์ก่อน
- ข้อ 3 กรณีที่นักศึกษามีความประสงค์นำผลงานวิทยานิพนธ์/สารนิพนธ์/ปัญหาพิเศษ/โครงการไปประดิษฐ์ หรือพัฒนาต่อยอด จะต้องได้รับความยินยอมเป็นลายลักษณ์อักษรจากมหาวิทยาลัยสงขลานครินทร์ก่อน
- ข้อ 4 กรณีมีผลประโยชน์เกิดขึ้นจากผลงานวิทยานิพนธ์/สารนิพนธ์/ปัญหาพิเศษ/โครงการ นักศึกษามีสิทธิได้รับการจัดสรรผลประโยชน์ที่เกิดขึ้น ตามระเบียบทรัพย์สินทางปัญญาของมหาวิทยาลัยสงขลานครินทร์ ที่เกี่ยวข้องกับผลงานนั้นๆ
- ข้อ 5 กรณีที่ผลงานวิทยานิพนธ์/สารนิพนธ์/ปัญหาพิเศษ/โครงการ นั้นได้รับสนับสนุนจากแหล่งทุนภายนอก ให้ปฏิบัติไปตามสัญญาการรับทุนเป็นรายกรณีไป
- ข้อ 6 หากผลงานวิทยานิพนธ์/สารนิพนธ์/ปัญหาพิเศษ/โครงการ มีข้อมูลทั้งหมดหรือบางส่วนที่เป็นความลับ และอาจารย์ที่ปรึกษาได้แจ้งแก่นักศึกษาให้ทราบแล้ว นักศึกษาจะต้องไม่เปิดเผยข้อมูลความลับนั้นแก่บุคคลอื่นก่อนได้รับอนุญาตเป็นลายลักษณ์อักษรจากมหาวิทยาลัยสงขลานครินทร์

(ลงนาม)..... นักศึกษา

(นางสาว ถลัชนันท์ สลัดทุกข์)

วันที่.....

(ลงนาม).....อาจารย์ที่ปรึกษา

(ดร.จุฑามณี อู่ยสกุล)

วันที่.....

(ลงนาม)อาจารย์ที่ปรึกษา

(ดร.กุลภัสร์ ทองแก้ว)

วันที่.....

ผู้เขียน นางสาวถลัษณ์ สลัดทุกข์ ปีการศึกษา 2/2563
เรื่อง การศึกษาการตัดหัวปลาทูน่าโดยใช้เทคโนโลยีภาพ
หลักสูตร ปริญญาตรีสาขาวิศวกรรมเมคาทรอนิกส์ ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์
มหาวิทยาลัยสงขลานครินทร์

บทคัดย่อ

ปลาทูน่ากระป๋องเป็นผลิตภัณฑ์อาหารจากปลาซึ่งเป็นที่นิยมอย่างมากสำหรับผู้คนในปัจจุบัน หนึ่งในกระบวนการแปรรูปปลาทูน่าเป็นปลาทูน่ากระป๋อง นั้นคือกระบวนการตัดแยกหัวปลาออกจากตัวปลา ซึ่งในขั้นตอนนี้ทำให้เกิดการสูญเสียเนื้อปลาที่ต้องการจากตัวปลาบางส่วนติดไปกับหัวปลา ทางผู้จัดทำโครงการจึงได้คิดค้นอัลกอริทึมสำหรับตรวจจับอวัยวะและระบุเส้นตัด เพื่อไปใช้เป็นโปรแกรมคำสั่งสำหรับเครื่องตัดพลังน้ำแรงดันสูง (water jet cutting machine) โดยการสร้างโปรแกรมคำสั่งดังกล่าว จะใช้เทคโนโลยีการประมวลผลภาพ (Image Processing)

การดำเนินโครงการเริ่มจากการศึกษาการทำงานของเทคโนโลยีการประมวลผลภาพ (Image Processing) จากนั้นจึงทำการรวบรวมข้อมูล ภาพถ่าย และภาพเคลื่อนไหวจากโรงงานจริงเพื่อมาทำฐานข้อมูลสำหรับโมเดลการตรวจจับวัตถุในตรวจจับอวัยวะที่ต้องการ อีกทั้งใช้สมการทางคณิตศาสตร์ในคำนวณ เพื่อออกแบบให้โปรแกรมสามารถลากและระบุเส้นตัดตามเงื่อนไขต้องการ

ผลการดำเนินการออกแบบโปรแกรมในการตรวจจับและสร้างเส้นตัดสำหรับการตัดหัวปลาทูน่า เมื่อทำการประเมินประสิทธิภาพของโมเดลการตรวจจับวัตถุจะได้ “ค่าความแม่นยำ” ในการทำนายบริเวณที่สนใจไม่น้อยกว่า 80 เปอร์เซ็นต์ และสามารถแสดงเส้นตัดผ่านทางจอภาพตามเงื่อนไขที่โปรแกรมคำสั่งไว้ นั่นคือระยะของเส้นตัดต้องห่างจากบริเวณตาปลาทูน่า จำนวนเท่ากับ 2 เซนติเมตร

คำสำคัญ: ปลาทูน่า, การตัดแยกหัวปลา, โมเดลการตรวจจับวัตถุ, เทคโนโลยีการประมวลผลภาพ (Image Processing) และค่าความแม่นยำ

เลขที่เอกสารอ้างอิงภาควิชา.....

Author Thalachanan Saladtook Academic Year 2/2563
Title A Study of AI for Tuna Deheading by using Image Processing.
Program Bachelor's degree in mechanical engineering. Department of mechatronics engineering,
Faculty of Engineering, Prince of Songkla University

Abstract

Canned tuna, the product from Tuna fish, is the one of popular healthy food in this century. There are many processes of changing Tuna fish form into canned tuna form. One of the processes is cutting the Tuna head of the Tuna body which is called deheading. The deheading process causes the wasting of some Tuna meat which is attached to the Tuna head, therefore the project organizer invented an algorithm for detecting Tuna organs and identifying cutting lines by using Image processing. This algorithm is used as a program for water jet cutting machines.

Implementation of the project begins with the study of image processing technology, and then collects photo and takes video from the factory. This step is for creating a database for an object detection model which detect Tuna organs. By using mathematical equations for calculating and designing the program to be able to specify and draw cutting lines according to the regulative conditions.

The results of the program, which detect organs and draw the cutting line for Tuna deheading, are known by evaluating the performance of the object detection model that has "Average precision" of more than 80%. Including, be able to draw cutting lines according to the regulative condition which has a distance of 2 centimeters from each mouth to eye.

Keywords: Tuna fish, Deheading, the object detection model, Image Processing and Average precision

Department Reference No.....

กิตติกรรมประกาศ

โครงการเรื่อง การศึกษาการตัดหัวปลาทุ่นโดยใช้เทคโนโลยีภาพ สามารถลุล่วงไปได้ด้วยดีด้วยความกรุณาให้ความช่วยเหลือและการแนะนำจากอาจารย์และบุคลากรหลายท่านผู้จัดทำจึงขอขอบพระคุณอาจารย์และบุคลากรดังนี้

อาจารย์จุฑามณี อุ่มสกุล และอาจารย์กุลภัสร์ ทองแก้ว อาจารย์ที่ปรึกษาโครงการทั้งสองท่าน ผู้ซึ่งคอยให้คำปรึกษา อีกทั้งแก้ไขปัญหาต่างๆที่เกิดขึ้นระหว่างการทำโครงการ

คณะกรรมการที่ปรึกษาโครงการ ที่คอยให้คำแนะนำในการแก้ไข และแนะแนวทางในการจัดบกพร่อง อาทิ การเลือกอวัยวะเป้าหมายในการใช้เป็นจุดอ้างอิงเส้นตัด เป็นต้น

นายเมธาสิทธิ์ ขาวนวลศรี ผู้ให้คำแนะนำในการสร้างอัลกอริทึม แนะนำแนวทางใช้ไลบรารีฟังก์ชันการเขียนโปรแกรม ตลอดจนชี้แนะแนวทางแก้ไขความบกพร่องของอัลกอริทึม

ขอขอบพระคุณอาจารย์ และบุคลากรทุกท่าน ที่มีส่วนร่วมกันนำเสนอแนวคิด คำแนะนำติชมในการดำเนินการของโครงการนี้ตลอดเวลาในการดำเนินโครงการ จนสามารถสำเร็จลุล่วงไปได้ด้วยดีได้

นางสาว ถลัชนันท์ สลัดทุกข์

ผู้จัดทำ

สารบัญ

บทคัดย่อ	5
ABSTRACT	6
กิตติกรรมประกาศ.....	7
สารบัญ.....	8
สารบัญภาพ	10
1 บทนำ.....	12
1.1. ปัญหาที่ทำให้โครงการและความสำคัญของปัญหา	13
1.2. วัตถุประสงค์ของโครงการ	14
1.3. ประโยชน์ที่คาดว่าจะได้รับ.....	14
1.4. ขอบเขตของโครงการ	14
2 ทฤษฎีที่เกี่ยวข้อง.....	15
2.1. ทฤษฎีและหลักการ	15
2.1.1. ข้อมูลเกี่ยวกับกระบวนการตัดหัวปลา	15
2.1.2. การตรวจจับสิ่งของโดยใช้ YOLO	16
2.1.3. การประเมินประสิทธิภาพของโมเดลการตรวจจับวัตถุ (Evaluating performance of an object detection model).....	18
2.1.4. สมการทางคณิตศาสตร์	19
2.2. เอกสารที่เกี่ยวข้อง	24
2.2.1. The effect of cutting and fish-orientation systems on the deheading yield of carp (Andrzej Dowgiallo, 2008)	25
3 เครื่องมือและวิธีการดำเนินโครงการ	26
3.1. เครื่องมือที่ใช้ในการทำโครงการ.....	26
3.2. วิธีการดำเนินโครงการ	26
4 ผลการดำเนินโครงการและวิจารณ์.....	28
4.1. ผลจากการตรวจจับอวัยวะที่สนใจ (บริเวณปากบนและตาปลา)	28
4.2. คำสั่งเพื่อหาพิกัดจุดในภาพ เพื่อในการเขียนโปรแกรมจำลอง	34

4.3.	คำสั่งจำลองการลากเส้นตรงโดยใช้จุดอ้างอิงสองจุด (บริเวณปากบนและตาปลา) และผลลัพธ์	35
4.4.	คำสั่งจำลองการลากเส้นวงรี และผลลัพธ์	38
4.5.	คำสั่งที่ใช้ในการปรับเทียบ(CALIBRATE) ขนาดของพิกเซลของภาพที่ถ่ายกับมาตรวัดจากวัตถุจริง (หน่วยเซนติเมตร) โดยการคลิกเมาส์ลากจากจุดเริ่มต้นไปยังจุดสิ้นสุดที่ต้องการวัด และผลลัพธ์ของความยาวของเส้นที่ลากและพิกัด	40
4.6.	เส้นตัดสำหรับใช้ตัดหัวปลา.....	43
5	สรุปผลการดำเนินงานและข้อเสนอแนะ	50
5.1.	ความแม่นยำในการตรวจจับวัตถุ.....	50
5.2.	ตำแหน่ง และค่าความชันของเส้นตัดเฉียงที่เหมาะสม	51
5.3.	แนวทางในการปรับปรุงโครงงานเพิ่มเติม	51
6	บรรณานุกรม	52
7	ภาคผนวก	54
7.1.	ภาคผนวก ก	54
	ประวัติผู้จัดทำ.....	60

สารบัญภาพ

รูปที่ 1	ผลิตภัณฑ์ปลากระป๋องจากปลาทูน่า (เกษตรอับ เซ็นเตอร์, 2020)	12
รูปที่ 2	ปลาทูน่า (นิชากร ศรีเพชรดี, 2018)	12
รูปที่ 3	FISH DEHEADING MACHINE (NIKKO CO., LTD., 2021)	13
รูปที่ 4	DEHEADING PROCESS (NIKKO CO., LTD., 2021)	13
รูปที่ 5	กระบวนการทำปลากระป๋อง	15
รูปที่ 6	ภาพที่ผ่านการตรวจจับวัตถุแบบ REAL TIME ด้วย YOLO (REDMON, 2021)	16
รูปที่ 7	ลำดับการเรียง LABEL IMAGE ในไฟล์ .TXT	16
รูปที่ 8	ตัวแปรที่ใช้แทนขอบเขตวัตถุ (MICHAL MAJ, 2021)	17
รูปที่ 9	อัตราส่วนการคำนวณค่า PRECISION และ RECALL ในรูปแบบ BOUNDING BOX (EVERINGHAM, ET AL., 2021)	18
รูปที่ 10	อัตราส่วนการคำนวณค่า IoU ในรูปแบบ BOUNDING BOX (ALEXEYAB, 2021)	19
รูปที่ 11	กราฟประกอบสมการการหาระยะทางระหว่างจุดสองจุด (สินีนางและเพ็ญพิชชา, 2008)	20
รูปที่ 12	กราฟที่มีความชัน $M1 = M2$ แล้วเส้นตรงทั้ง 2 จะขนานกัน (TUENONG GROUP, 2021)	21
รูปที่ 13	กราฟที่มีความชัน $M1 \times M2$ แล้วเส้นตรงทั้ง 2 จะตั้งฉากกัน (TUENONG GROUP, 2021)	22
รูปที่ 14	หมายเลขเส้นตัด และพิกัดของจุดต่างๆที่ใช้ในการคำนวณ	23
รูปที่ 15	ส่วนประกอบของวงรีและตัวแปรต่างๆ (TEWFREE, 2020)	23
รูปที่ 16	วงรีที่มีจุดศูนย์กลางอยู่ที่จุดกำเนิดและแกนเอกอยู่บนพิกัด (TEWFREE, 2020)	24
รูปที่ 17	รูปแบบเส้นตัดปลาแบบต่างๆ	25
รูปที่ 18	ตัวอย่างการทำนายอวัยวะปลา (ตาและปากบน) สำหรับตัวอย่างสุมที่ 1	28
รูปที่ 19	ตัวอย่างการทำนายอวัยวะปลา (ตาและปากบน) สำหรับตัวอย่างสุมที่ 2	29
รูปที่ 20	ตัวอย่างการทำนายอวัยวะปลา (ตาและปากบน) สำหรับตัวอย่างสุมที่ 3	29
รูปที่ 21	ตัวอย่างการทำนายอวัยวะปลา (ตาและปากบน) สำหรับตัวอย่างสุมที่ 4	30
รูปที่ 22	ค่า MAP ช่วงการเทรนที่ 0-2000 รอบ	31
รูปที่ 23	ค่า MAP ช่วงการเทรนที่ 2000-4000 รอบ (ใช้ไฟล์ WEIGHT ที่ 2000 รอบเป็น PRETRAIN-WEIGHT)	32
รูปที่ 24	ค่าสำหรับประเมินประสิทธิภาพของโมเดลการตรวจจับวัตถุ	33
รูปที่ 25	คำสั่งใน OPENCV สำหรับหาพิกัดจุดต่างๆในภาพตามที่ต้องการโดยการคลิกเมาส์	35
รูปที่ 26	ผลลัพธ์จากคำสั่งสำหรับหาพิกัดจุดต่างๆในภาพตามที่ต้องการโดยการคลิกเมาส์	35
รูปที่ 27	คำสั่งใน OPENCV สำหรับลากเส้นตรงโดยใช้จุดอ้างอิงสองจุด	37
รูปที่ 28	ผลลัพธ์จากคำสั่งสำหรับลากเส้นตรงโดยใช้จุดอ้างอิงสองจุด	37

รูปที่ 29 คำสั่งใน OPENCV สำหรับลากเส้นลากเส้นวงรี	39
รูปที่ 30 ผลลัพธ์จากคำสั่งสำหรับลากเส้นลากเส้นวงรี	39
รูปที่ 31 คำสั่งใน OPENCV สำหรับปรับเทียบ(CALIBRATE) ขนาดของพิกเซลของภาพที่ถ่ายกับมาตรวัดจากวัตถุจริง	41
รูปที่ 32 ผลลัพธ์ในรูปของเส้นตรงจากจุดเริ่มต้นไปยังจุดสิ้นสุดที่ต้องการวัด	42
รูปที่ 33 ผลลัพธ์จากคำสั่งที่แสดงพิกัดจุดเริ่มต้น พิกัดจุดสิ้นสุด และความยาวของเส้นตรงในหน่วยพิกเซล	42
รูปที่ 34 โค้ด OPENCV สำหรับวาดวงกลมเพื่อใช้แสดงจุดอ้างอิง โค้ด OPENCV สำหรับวาดเส้นตัดตรง	44
รูปที่ 35 ผลลัพธ์ในที่แสดงในรูปจุดวงกลมที่ใช้ประกอบการลากเส้นตัด	45
รูปที่ 36 ผลลัพธ์ในที่แสดงในรูปเส้นตรงเพื่อแสดงเส้นทางสำหรับตัดหัวปลาที่ 1	45
รูปที่ 37 ผลลัพธ์ในที่แสดงในรูปเส้นตรงเพื่อแสดงเส้นทางสำหรับตัดหัวปลาที่ 2	46
รูปที่ 38 โค้ด OPENCV สำหรับลากเส้นตัดตามเงื่อนไขแบบใหม่	48
รูปที่ 39 ผลลัพธ์การลากเส้นตัดตามเงื่อนไขแบบใหม่สำหรับตัดหัวปลาตัวที่ 1	48
รูปที่ 40 ผลลัพธ์การลากเส้นตัดตามเงื่อนไขแบบใหม่สำหรับตัดหัวปลาตัวที่ 2	49
รูปที่ 41 ผลลัพธ์การลากเส้นตัดตามเงื่อนไขแบบใหม่สำหรับตัดหัวปลาตัวที่ 3	49

1 บทนำ

ปัจจุบันความตื่นตัวในการบริโภคอาหารที่มีประโยชน์ และกระแสวิงไวในสุขภาพและรูปร่างของผู้บริโภค ทั้งในและต่างประเทศมีมากขึ้น ซึ่งทำให้ความต้องการบริโภคเนื้อปลาขยายตัวอย่างต่อเนื่อง และผลิตภัณฑ์ปลาทูน่าก็เป็นอีกทางเลือกหนึ่งของผู้บริโภค เนื่องจากมีสารอาหารที่มีประโยชน์ เป็นอาหารทะเลที่ให้โปรตีนสูง และไม่มีคลอเรสเตอรอล และมีราคาถูกหากเปรียบเทียบกับอาหารทะเลประเภทอื่น อาทิ กุ้ง และปลาหมึก ที่ถึงแม้ว่าจะมีประโยชน์ ให้สารอาหารโปรตีน แต่ก็มีคลอเรสเตอรอลสูงด้วยเช่นกัน ปลาทูน่ากระป๋องจึงเหมาะแก่การรับประทานได้ทุกเพศ ทุกวัย และสามารถรับประทานได้ทันที เพราะเป็นอาหารที่ปรุงรสสำเร็จ



รูปที่ 1 ผลิตภัณฑ์ปลากระป๋องจากปลาทูน่า (เกษตรอับ เซ็นเตอร์, 2020)



รูปที่ 2 ปลาทูน่า (นิชากร ศรีเพชรดี, 2018)

ดังนั้นปลาทูน่ากระป๋องจึงเป็นผลิตภัณฑ์อาหารจากปลาที่นิยมมากพอ ๆ กับปลากระป๋อง โดยส่วนใหญ่จะเป็นที่นิยมทั้งในและต่างประเทศ ปลาทูน่ากระป๋องถูกคิดค้นขึ้นในช่วงศตวรรษที่ 20 เพื่อเก็บรักษาปลาโดยไม่ต้องทำให้น้ำแข็งหรือกระดิกน้ำแข็งในการแช่ให้ปลาไม่หมดเวลา โดยกรรมวิธีการทำปลาทูน่ากระป๋องนั้นจะเริ่มจากการทำความสะอาดตัวปลาทูน่า จากนั้นนำปลาทูน่าเข้าในเตาอบไอน้ำเป็นเวลา 2-4 ชั่วโมง เมื่อได้ปลาทูน่าที่ปรุงสุกแล้วจะถูกหั่นและแบ่งออกเป็นส่วน ส่วนละ 170 กรัม โดยประมาณ เพื่อให้สามารถนำปลาทูน่าบรรจุเข้า

ไปในกระป๋องได้ ต่อจากนั้นในกระป๋องจะถูกเติมน้ำหรือน้ำมันพืชในระดับปานกลางและปิดผนึกสุญญากาศ กระป๋องให้เรียบร้อย ขั้นตอนสุดท้ายคือการรักษาความร้อนเพื่อกำจัดแบคทีเรียที่อาจเกิดขึ้น

เนื่องด้วยเทคโนโลยีที่มีความก้าวหน้ามากขึ้น กรรมวิธีการทำปลากระป๋องในปัจจุบันจึงได้นำเทคโนโลยี เข้ามาประยุกต์ใช้ในระบบการผลิต ซึ่งหนึ่งในนั้นคือกระบวนการตัดแยกหัวปลาออกจากตัวปลา ได้มีการนำ เครื่องตัดหัวปลา (Fish deheading machine) เข้ามาช่วยด้วยเช่นกัน แต่เครื่องตัดหัวปลาก็ยังมีข้อจำกัดในด้าน ของความแม่นยำ และไม่มีรูปแบบหรือระยะการตัดสำหรับปลาแต่ละตัวเท่าใดนัก



รูปที่ 3 Fish Deheading machine (Nikko Co., Ltd., 2021)



รูปที่ 4 Deheading process (Nikko Co., Ltd., 2021)

1.1. ปัญหาที่ทำได้โครงการและความสำคัญของปัญหา

ขั้นตอนการตัดหัวปลาในปัจุบันได้มีการนำเครื่องตัดหัวปลาเข้ามาใช้ในอุตสาหกรรม แต่เนื่องด้วยเครื่องตัดดังกล่าวมีราคาแพง ความแม่นยำและรูปแบบการตัดที่มีไม่มากนัก จึงอาจทำให้เกิดการสูญเสียปริมาณเนื้อปลาที่ถูกตัดติดไปพร้อมกับหัวปลา นอกจากนี้การตัดด้วยเครื่องตัดในปัจจุบันไม่สามารถวัดระยะห่างระหว่างตาปลาและบริเวณเส้นที่ต้องการตัดให้เท่ากับระยะ 2 เซนติเมตรสำหรับปลาทุกๆตัว (จากขนาดของปลาแต่ละตัวที่ต่างกัน) ตามความต้องการของโรงงานที่ได้ไปทำการศึกษาค้นคว้าได้

1.2. วัตถุประสงค์ของโครงการ

1. เพื่อพัฒนาอัลกอริทึม AI ในการตรวจจับตำแหน่งตำแหน่งลูกตาปลา, ริมปากด้านบนของปลา, สามารถแสดงเส้นตัดที่มีระยะห่างจากตาปลาไปทางด้านผังก้างตัว 2 เซนติเมตร และเส้นโค้งในส่วนที่เป็น red meat ที่ติดกับหัวปลาได้
2. เพื่อทดสอบประสิทธิภาพในการทำนายตำแหน่งที่สนใจโดยใช้ข้อมูลจากภาพถ่ายจริงและภาพที่ได้รับการปรับแต่ง

1.3. ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถทำนายบริเวณที่สนใจได้อย่างแม่นยำ
2. สามารถนำอัลกอริทึมที่ได้พัฒนามาใช้กับเครื่องตัดหัวปลาจริงได้

1.4. ขอบเขตของโครงการ

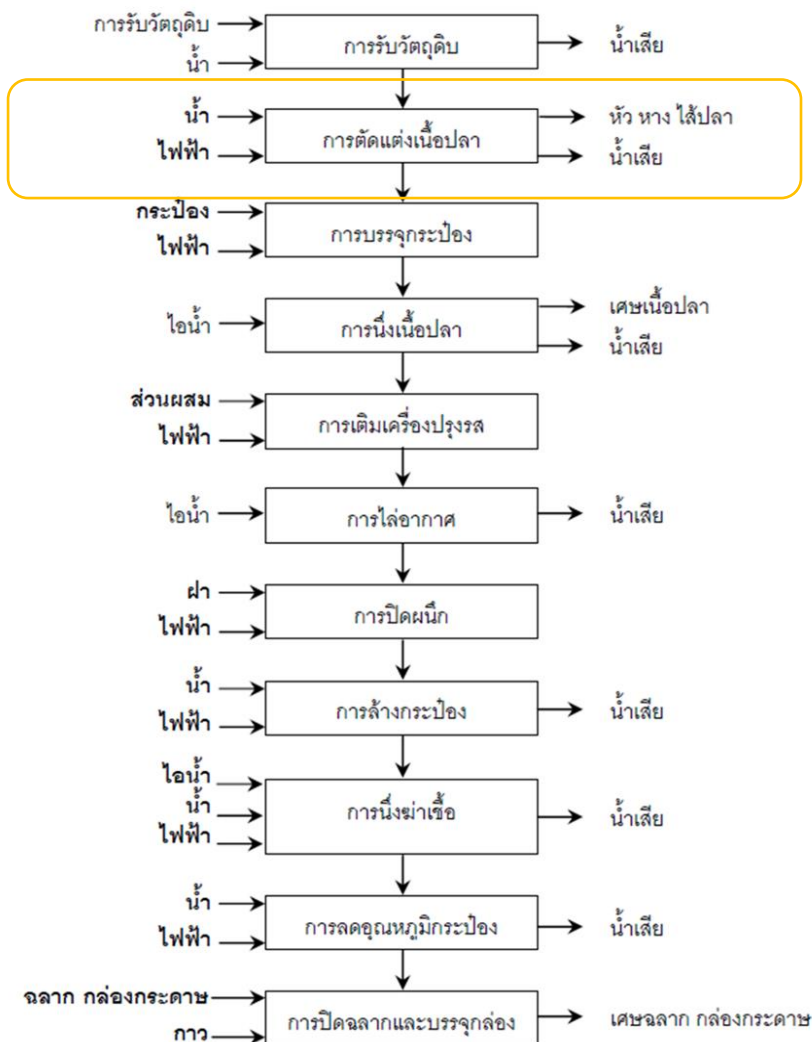
1. ระบุตำแหน่งอวัยวะของปลาได้ คือ บริเวณลูกตา ริมฝีปากด้านบน รวมทั้งเส้นตัดเส้นตัดที่มีระยะห่างจากตาปลาไปทางด้านผังก้างตัว 2 เซนติเมตร และเส้นโค้งในส่วนที่เป็น red meat ที่ติดกับหัวปลาได้
2. มีความแม่นยำในการทำนายบริเวณที่สนใจไม่น้อยกว่า 80 เปอร์เซ็นต์

2 ทฤษฎีที่เกี่ยวข้อง

2.1. ทฤษฎีและหลักการ

2.1.1. ข้อมูลเกี่ยวกับกระบวนการตัดหัวปลา

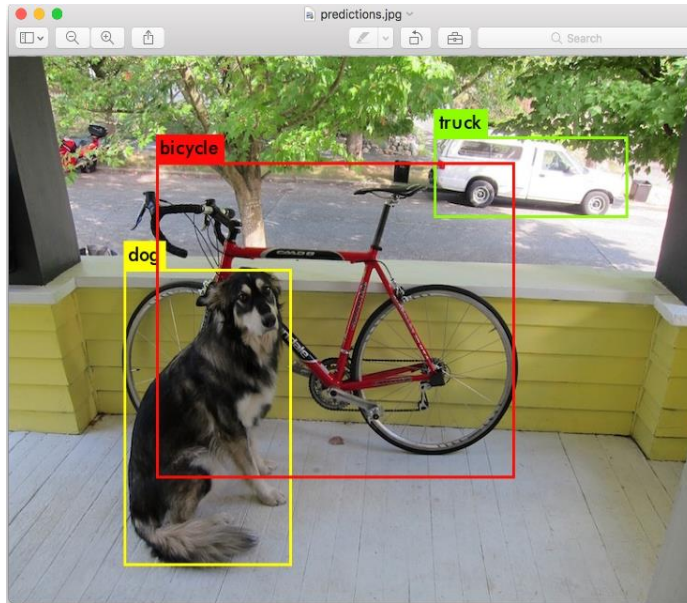
โดยขั้นตอนที่เราสนใจในกระบวนการทำปลากระป๋องคือส่วน การตัดแต่งเนื้อปลา : การตัดแต่งเนื้อปลาโดยตัดส่วนที่ไม่ต้องการออก ได้แก่ หัวปลา หางปลาและไส้ปลา โดยส่วนที่เหลือจากการตัดแต่ง จะถูกแยกไว้เพื่อนำไปขายหรือใช้ประโยชน์ต่อไป



รูปที่ 5 กระบวนการทำปลากระป๋อง

2.1.2. การตรวจจับสิ่งของโดยใช้ YOLO

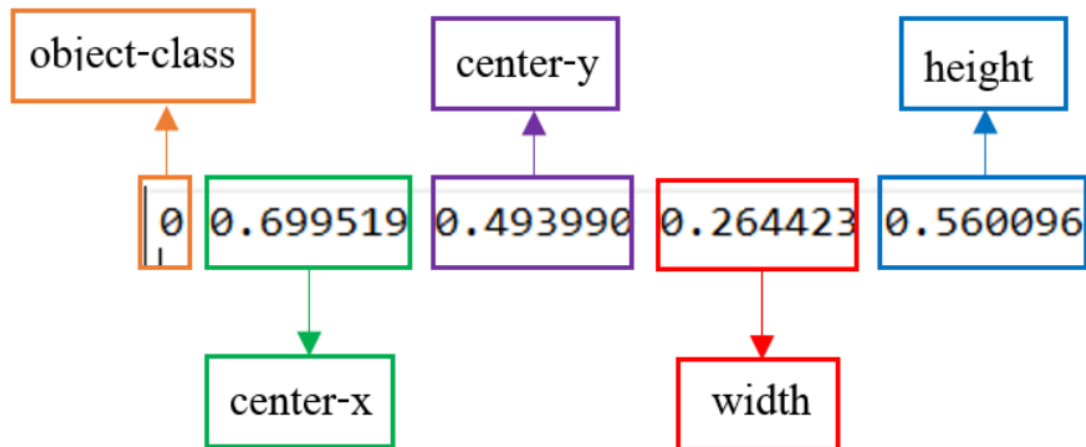
YOLO คือ ระบบตรวจจับวัตถุแบบ real time



รูปที่ 6 ภาพที่ผ่านการตรวจจับวัตถุแบบ real time ด้วย YOLO (Redmon, 2021)

label image คือการกำหนดตำแหน่งของ object ที่เราต้องการ โดยจะเรียงลำดับใน 1 บรรทัด ตามนี้

<object-class> <center-x> <center-y> < width > <height>

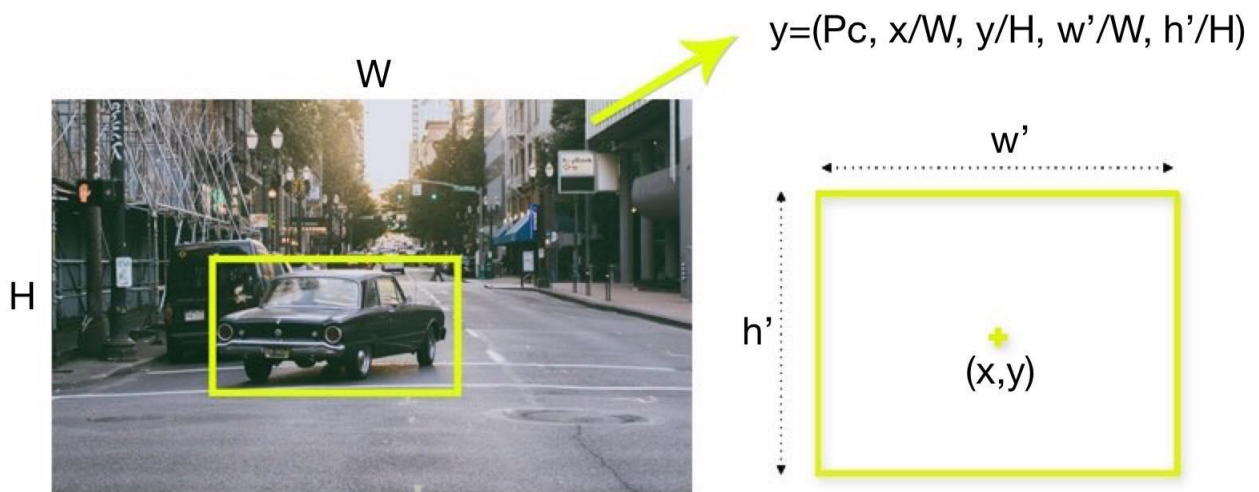


รูปที่ 7 ลำดับการเรียง label image ในไฟล์ .txt

$$\begin{aligned} \text{center} - x &= \frac{x}{W} \\ \text{center} - y &= \frac{y}{H} \\ \text{width} &= \frac{w'}{W} \\ \text{height} &= \frac{h'}{H} \end{aligned}$$

ซึ่ง x เป็นพิกัด x (พิกเซล) ของกึ่งกลางของกล่องขอบเขตวัตถุ
 y เป็นพิกัด y (พิกเซล) ของศูนย์กลางของกล่องขอบเขตวัตถุ
 w' เป็นความกว้าง (พิกเซล) ของกล่องขอบเขตวัตถุ
 h' เป็นความสูง (พิกเซล) ของกล่องขอบเขตวัตถุ
 W เป็นความกว้าง (พิกเซล) ของภาพทั้งหมด
 H เป็นความสูง (พิกเซล) ของภาพทั้งหมด

ดังรูป

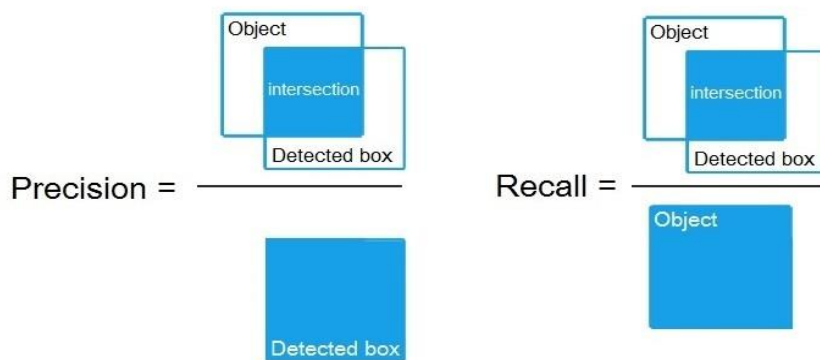


รูปที่ 8 ตัวแปรที่ใช้แทนขอบเขตวัตถุ (Michal Maj, 2021)

2.1.3. การประเมินประสิทธิภาพของโมเดลการตรวจจับวัตถุ (Evaluating performance of an object detection model)

mAP (mean Average Precision) for Object Detection)

- AP (Average precision) เป็น Confusion Matrix ที่ใช้ในการวัดค่าความแม่นยำของเครื่องตรวจจับวัตถุเช่น Faster R-CNN, SSD เป็นต้น ค่า Average precision จะแสดงผลในรูปค่าตัวเลขในช่วง 0 ถึง 1
- Confusion Matrix คือการประเมินผลลัพธ์การทำนาย (หรือผลลัพธ์จากโปรแกรม) เปรียบเทียบกับผลลัพธ์จริงๆ ที่หาโดยมนุษย์
 - True Positive (TP) คือ สิ่งที่โปรแกรมทำนายว่าจริง และมนุษย์ระบุว่าเป็นจริง
 - True Negative (TN) คือ สิ่งที่โปรแกรมทำนายว่าไม่จริง และมนุษย์ระบุว่าเป็นจริง
 - False Positive (FP) คือ สิ่งที่โปรแกรมทำนายว่าจริง แต่มนุษย์ระบุว่าเป็นจริง
 - False Negative (FN) คือ สิ่งที่โปรแกรมทำนายว่าไม่จริง แต่มนุษย์ระบุว่าเป็นจริง
- Precision & recall
 - Precision เป็นตัววัดค่าความแม่นยำของผลการทำนายวัตถุ ยิ่งมีเปอร์เซ็นต์มาก ความถูกต้องในการตรวจจับวัตถุก็ยิ่งมากขึ้นตาม
 - Recall (True Positive Rate) คือ ค่าที่บอกว่าโปรแกรมทำนายได้ว่าจริง เป็นอัตราส่วนกับค่าวัตถุจริงทั้งหมด



รูปที่ 9 อัตราส่วนการคำนวณค่า Precision และ Recall ในรูปแบบ Bounding box (Everingham, et al., 2021)

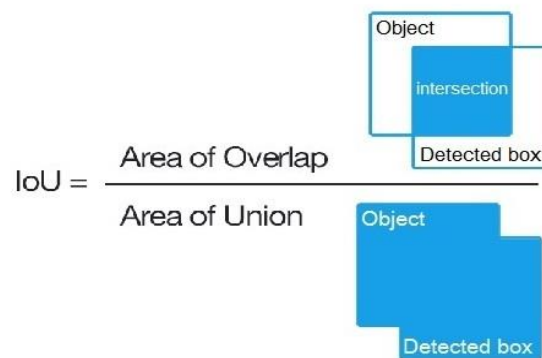
สูตรการคำนวณค่า Precision และ Recall (Hui, 2018)

$$\text{Precision} = \frac{TP}{TP+FP}$$

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

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

- IoU (Intersection over union) เป็นค่าสำหรับวัดความทับซ้อนระหว่าง 2 พื้นที่ล้อมรอบ ใช้เพื่อคาดการณ์ว่าขอบเขตที่ทำนายทับซ้อนกับขอบเขตวัตถุจริงมากแค่ไหน ในชุดข้อมูลบางชุด เรากำหนดเกณฑ์ IoU ไว้ล่วงหน้า (เท่ากับ 0.5) สำหรับจำแนกว่าผลการทำนายนั้นเป็น True positive หรือ False positive.



รูปที่ 10 อัตราส่วนการคำนวณค่า IoU ในรูปแบบ Bounding box (AlexeyAB, 2021)

2.1.4. สมการทางคณิตศาสตร์

1. ทฤษฎีบทพีทาโกรัส

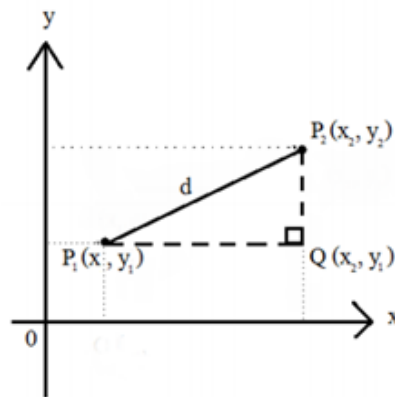
ในสามเหลี่ยมมุมฉากใดๆ พื้นที่ของสี่เหลี่ยมจัตุรัสที่มีด้านเป็นด้านตรงข้ามมุมฉาก เท่ากับผลรวมพื้นที่ของสี่เหลี่ยมจัตุรัสที่มีด้านเป็นด้านประชิดมุมฉากของสามเหลี่ยมมุมฉากนั้น

ทฤษฎีบทดังกล่าวสามารถเขียนเป็นสมการสัมพันธ์กับความยาวของด้าน a , b และ c ได้ ซึ่งมักเรียกว่า สมการพีทาโกรัส ดังสมการด้านล่าง

$$a^2 + b^2 = c^2$$

2.ความรู้เบื้องต้นเกี่ยวกับเรขาคณิตศาสตร์ เรื่อง การหาระยะทางระหว่างจุดสองจุด
กรณีเส้นตรงไม่ขนานกับแกน x และ y

กำหนดจุด $P_1(x_1, y_1)$ และจุด $P_2(x_2, y_2)$ อยู่บนเส้นตรงเดียวกันการหาระยะทางระหว่างจุด P_1 และจุด P_2 สามารถหาได้ตามเงื่อนไขของพิกัดจุด บนเส้นตรงดังนี้ กำหนด d แทน ระยะทางระหว่างจุด P_1 และ จุด P_2 ลากส่วนของเส้นตรงจากจุด P_1 และ P_2 ให้ขนานกับแกน X และแกน y ตามลำดับ มาตัดกันที่จุด Q และจุด Q จะมีพิกัด (x_2, y_1) จะได้รูปสามเหลี่ยมมุมฉาก P_1QP_2 ซึ่งมีมุม P_1QP_2 เป็นมุมฉาก จากทฤษฎีบทพีทาโกรัส จะได้ว่า



รูปที่ 11 กราฟประกอบสมการการหาระยะทางระหว่างจุดสองจุด (สินีนานูและเพ็ญพิชชา, 2008)

$$d^2 = (|x_1 - x_2|)^2 + (|y_1 - y_2|)^2$$

$$d^2 = (|x_1 - x_2|)^2 + (|y_1 + y_2|)^2 ; (|x_1 - x_2|)^2 \text{ มีค่าเท่ากับ } (|x_1 - x_2|)^2$$

จะได้ระยะทางระหว่างจุด $P_1(x_1, y_1)$ และจุด $P_2(x_1, y_2)$ คือ

$$d = \sqrt{(|x_1 - x_2|)^2 + (|y_1 - y_2|)^2}$$

$$\text{จะประยุกต์ได้เป็น } (2cm)^2 = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

.....สมการที่ 1

หมายเหตุ : x_2 และ y_2 เป็นพิกัดตำแหน่งที่ได้จาก YOLO

3.สมการเส้นตรงและความชัน

การสร้างสมการเส้นตรงสามารถสร้างได้จากสมการ $y - y_0 = m(x - x_0)$

โดยต้องมีส่วนประกอบทั้งหมด 2 ส่วน

ความชัน : ความชันสามารถหาได้จาก หัวข้อเรื่อง ความชันของกราฟ (Slope) ที่ผ่านมา ซึ่งหาได้ 2 วิธี

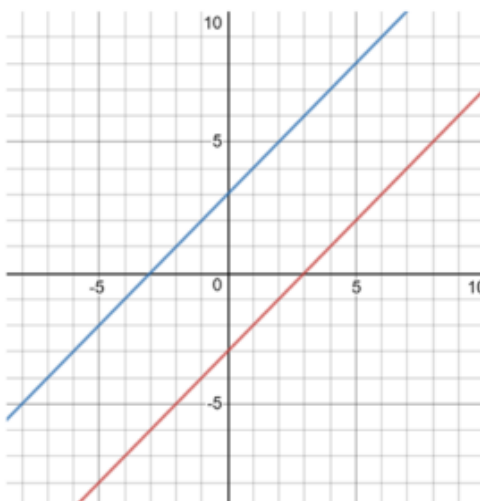
1.จุดที่เส้นตรงผ่าน

2.จุดที่เส้นตรงผ่านสามารถมองได้จากกราฟ หรือ โจทย์กำหนดมาให้

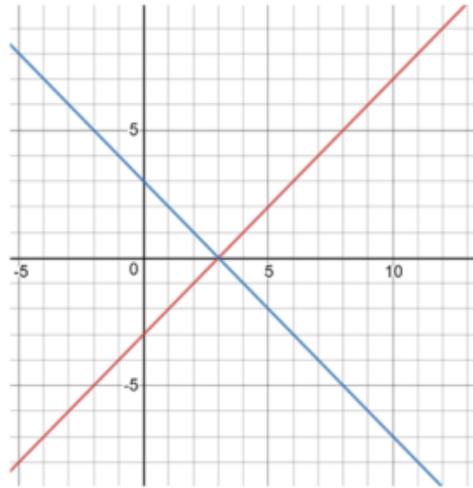
ความสัมพันธ์ของกราฟความชัน

ถ้า $m_1 = m_2$ แล้วเส้นตรงทั้ง 2 จะขนานกัน

ถ้า $m_1 \times m_2 = -1$ แล้วเส้นตรงทั้งสองจะตั้งฉากกัน



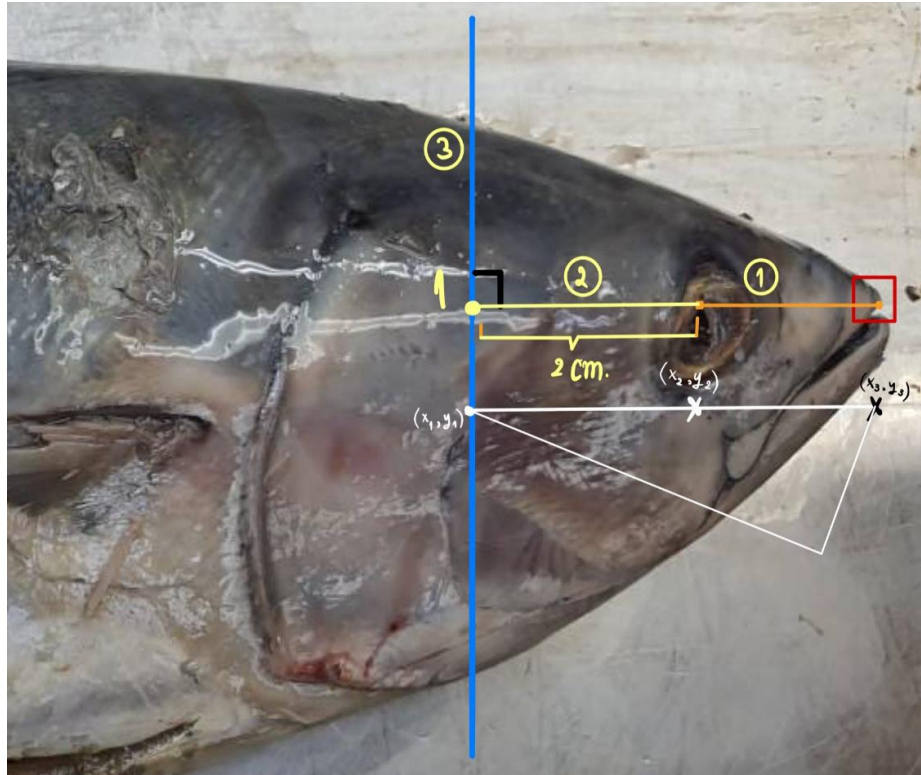
รูปที่ 12 กราฟที่มีความชัน $m_1 = m_2$ แล้วเส้นตรงทั้ง 2 จะขนานกัน (Tuenong Group, 2021)



รูปที่ 13 กราฟที่มีความชัน $m_1 \times m_2$ แล้วเส้นตรงทั้ง 2 จะตั้งฉากกัน (Tuenong Group, 2021)

จะประยุกต์ได้เป็น $m_{32} = m_{31} ; \frac{(y_2 - y_3)}{(x_2 - x_3)} = \frac{(y_1 - y_3)}{(x_1 - x_3)}$
สมการที่ 2

สามารถหาคำตอบของพิกัด x_1 และ y_1 จากจำนวน 2 สมการ (สมการที่ 1 และสมการที่ 2) ซึ่งเป็นจุดที่ระบุตำแหน่งของเส้นตั้งฉากต่อไป จากนั้น $m_1 \times m_2 = -1$ แล้วเส้นตรงทั้งสองจะตั้งฉากกัน ทำให้สามารถหาความชันของเส้นหมายเลข 3 ได้จากสมการ $m_{31} \times m_{\text{เส้นหมายเลข 3}} = -1$

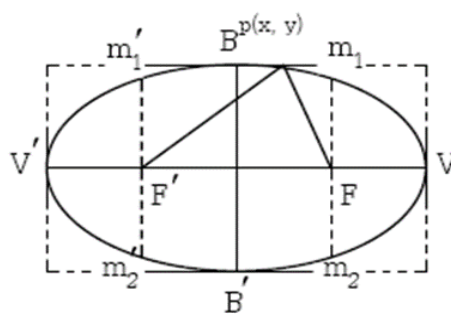


รูปที่ 14 หมายเลขเส้นตัด และพิกัดของจุดต่างๆที่ใช้ในการคำนวณ

4. สมการวงรี

วงรี (Ellipse) คือเซตของจุดทั้งหมดในระนาบซึ่งผลบวกของระยะทางจากจุดใด ๆ จุดหนึ่งในเซตไปยังจุดคงที่ 2 จุดมีค่าคงตัว

ส่วนประกอบของวงรี



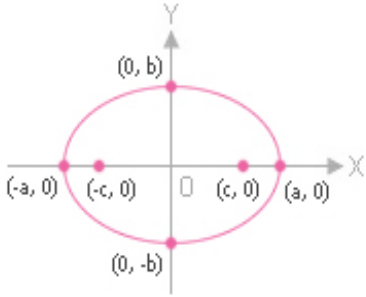
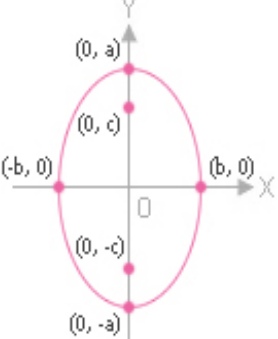
รูปที่ 15 ส่วนประกอบของวงรีและตัวแปรต่างๆ (Tewfree, 2020)

F, F' เป็นจุดคงที่ เรียกว่าจุดโฟกัส (Focus)

v, v' เป็นเส้นตรงที่ผ่านจุดโฟกัส และมีจุดปลายทั้งสองเป็นจุดยอด เรียกว่า แกนเอก

B, B' เป็นเส้นตรงที่ผ่านจุดศูนย์กลางและตั้งฉากกับแกนเอก โดยมีจุดปลายทั้งสองอยู่บนวงรี เรียกว่า แกนโท

m_1, m_2, m'_1, m'_2 เป็นเส้นตรงที่ผ่านจุดโฟกัส และตั้งฉากกันแกนของรูป เรียกว่าเส้นลาตัสเรกตัม

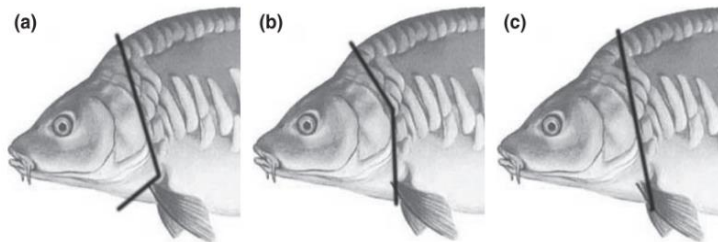
วงรีที่มีจุดศูนย์กลางอยู่ที่จุดกำเนิดและแกนเอกอยู่บนพิกัด		
สมการรูปแบบมาตรฐาน	$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1, a > b > 0$	$\frac{x^2}{b^2} + \frac{y^2}{a^2} = 1, a > b > 0$
จุดยอด	$(-a, 0), (a, 0)$	$(0, -a), (0, a)$
แกนเอก	อยู่บนแกน X มีความยาว $2a$ หน่วย	อยู่บนแกน Y มีความยาว $2a$ หน่วย
แกนโท	อยู่บนแกน Y มีความยาว $2b$ หน่วย	อยู่บนแกน X มีความยาว $2b$ หน่วย
โฟกัส	$(-c, 0), (c, 0); c^2 = a^2 - b^2$	$(0, -c), (0, c); c^2 = a^2 - b^2$
กราฟ		

รูปที่ 16 วงรีที่มีจุดศูนย์กลางอยู่ที่จุดกำเนิดและแกนเอกอยู่บนพิกัด (Tewfree, 2020)

2.2. เอกสารที่เกี่ยวข้อง

2.2.1. The effect of cutting and fish-orientation systems on the deheading yield of carp (Andrzej Dowgiallo, 2008)

งานวิจัยประยุกต์เกี่ยวกับรูปแบบการตัดหัวปลาคาร์ฟแบบต่างๆ โดยการคำนวณหาค่าเฉลี่ยจุดตัดที่เหมาะสม (the mean carp-deheading yield) สำหรับการตัดหัวปลา เพื่อลดการสูญเสียของเนื้อปลาคาร์ฟในกระบวนการตัดหัวปลาให้มากที่สุด



รูปที่ 17 รูปแบบเส้นตัดปลาแบบต่างๆ (a) V-cut with two circular knives. (b) V-cut with one cup-type circular knife. (c) straight cut at an angle to the fish backbone. (Dowgiallo, 2008)

สามารถประยุกต์ใช้ในการวางแนวเส้นในการตัดหัวปลาผลสรุปมีดังนี้คือ

-แบบ (a) V-cut ด้วยใบมีดวงกลมแบบคู่ มีความใกล้เคียงกับจุด yield เฉลี่ยคือ 77.9%

-แบบ (b) V-cut ด้วย one cup-type knife – 75.6%

-แบบ (c) ตัดตรงที่มุม 79 องศา ถึงกระดูกสันหลังของปลา – 77.4%.

สรุปได้ว่าไม่มีความแตกต่างอย่างมีนัยสำคัญทางสถิติกับค่าเฉลี่ยของจุด yield ของการตัดทั้ง 3 แบบ ดังนั้น การตัดตรงจึงเป็นการตัดที่ดีที่สุด เนื่องจากช่วยลดความยุ่งยากกระบวนการตัดได้อีกด้วย

3 เครื่องมือและวิธีการดำเนินโครงการ

3.1. เครื่องมือที่ใช้ในการทำโครงการ

- 1.Laptop core i5 NVIDIA 930M
- 2.Desktop PC DELL
- 3.Samsung Galaxy J7 Pro <ถ่ายรูปและบันทึกภาพ>
- 4.ระบบปฏิบัติการ UBUNTU 18.04 LTS
- 5.ระบบปฏิบัติการ WINDOW 10
- 6.Python 3.9.0
- 7.OpenCV 4.5.1
- 8.LosslessCut <โปรแกรมสำหรับตัดต่อวิดีโอ>
- 9.KRename <โปรแกรมสำหรับเปลี่ยนชื่อไฟล์อัตโนมัติ>
- 10.Labelimg <โปรแกรมสำหรับสร้างกรอบล้อมวัตถุที่สนใจ>
- 11.YOLOv4 by AlexeyAB <โปรแกรมสำหรับการรู้จำวัตถุ>
- 12.Sublime Text

3.2. วิธีการดำเนินโครงการ

3.2.1 ศึกษาข้อมูลเกี่ยวกับการตัดปลา และงานวิจัยที่เกี่ยวข้อง ทำรายการข้อมูลที่สำคัญที่ต้องการรวบรวมก่อนไปสถานที่จริง (โรงงานผลิตปลากระป๋อง) และอุปกรณ์ที่ต้องใช้

3.2.2 ติดตั้งโปรแกรมเกี่ยวกับเทคโนโลยีประมวลผลภาพและศึกษาการใช้โปรแกรมพื้นฐาน

- 1 ระบบปฏิบัติการ UBUNTU
- 2 ติดตั้ง code ของระบบ real-time object detection (YOLO และ Labelimg)
- 3 ติดตั้งฟังก์ชันโปรแกรม OpenCV
- 4 โปรแกรมอื่นๆ: LosslessCut, KRename และ Sublime Text

3.2.3 คุยกับทางโรงงาน เพื่อให้ทางผู้จัดทำทราบความต้องการสำหรับการออกแบบการทำงานของโปรแกรมให้สอดคล้องกับการใช้งานในโรงงานจริง และทำการเก็บข้อมูลโดยการถ่ายภาพนิ่ง และภาพเคลื่อนไหว

3.2.4 ออกแบบวิธีการตรวจจับภาพให้เป็นไปตามวัตถุประสงค์ของโรงงาน โดยอาศัยสมการทางคณิตศาสตร์เข้ามาเกี่ยวข้อง โดยส่วนประกอบสำหรับการลากเส้นตัดทั้งหมด คือ

1. จุดอวยะสำหรับลากเส้นตรงสองจุด: ส่วนปากบน และส่วนตาปลา ตรวจจับโดยYOLO

2. เส้นตรงจากการใช้สมการคณิตศาสตร์ประยุกต์กับการเขียนโปรแกรมใน OpenCV โดยใช้จุดอ้างอิงสองจุด ตามข้อ 1

3. เส้นตรงที่มีความชันเท่ากับเส้นตรงจากข้อ 2 โดยมีการลากเส้นจากจุดอ้างอิงจากส่วนตาปลาออกไปสองเซนติเมตร (ลากเส้นโดยใช้สมการคณิตศาสตร์ประยุกต์กับการเขียนโปรแกรมใน OpenCV)

4. เส้นตรงที่ตั้งฉากกับเส้นตรงในข้อ 3 (ลากเส้นโดยใช้สมการคณิตศาสตร์ประยุกต์กับการเขียนโปรแกรมใน OpenCV)

5. เส้นโค้งวงรี* ลากเส้นโดยใช้สมการคณิตศาสตร์ประยุกต์กับการเขียนโปรแกรมใน OpenCV)

3.2.5 เขียนคำสั่งหาพิกัดจุดในภาพ คำสั่งสำหรับจำลองการลากเส้นทั้งหมดจากข้อ 3.2.4 และคำสั่งสำหรับปรับเทียบ(calibrate) ขนาดของพิกเซลของภาพที่ถ่ายกับมาตรวัดจากวัตถุจริง (หน่วยเซนติเมตร) ของ OpenCV โดยใช้ Sublime text

3.2.6 เขียนโปรแกรมเพิ่มเติม โดยนำคำสั่งตรวจจับพิกัดวัตถุจาก YOLO ไปรวมกับโปรแกรมประยุกต์สมการทางคณิตศาสตร์ใน OpenCV

3.2.7 การทดสอบผลจะแบ่งออกเป็น 2 ช่วง: ช่วงที่ 1 ทดสอบความแม่นยำของการตรวจจับอวัยวะที่สนใจสองจุด ช่วงที่ 2 ทดสอบความแม่นยำและประสิทธิภาพจากการใช้โปรแกรมลากเส้นตัดต่างๆ โดยมีรายละเอียดดังนี้

1) ทดสอบผลการตรวจจับบริเวณจุดที่สนใจ ซึ่งจะจัดทำข้อมูลออกเป็น 2 classes คือ eye และ mouth โดยมีค่า Parameter จำนวน 6 ตัว คือ x, y, w, h, w', h' สำหรับการหา Boundary box เพื่อล้อมรอบวัตถุ 1 ชิ้น ซึ่งค่าความแม่นยำและค่าความผิดพลาดของโมเดลตรวจจับวัตถุ จะได้จากโปรแกรมการคำนวณของ YOLOV3 แสดงผลออกมาในรูปแบบกราฟ Mean average precision: mAP และค่าที่ IoU (อ้างอิงจากหัวข้อที่ 2.1.3 การประเมินประสิทธิภาพของโมเดลการตรวจจับวัตถุ) โดยจะแบ่งอัตราส่วนของข้อมูลที่ใช้สำหรับการสอนโมเดล และทดสอบได้ดังนี้ Training Dataset = 80% ของ Dataset, Test Dataset = 20% ของ Dataset และ Validation Dataset = 20% ของ Training Dataset

2) ทดสอบเพื่อหาค่าความแม่นยำสำหรับการลากเส้นตัดที่เหมาะสม จากการสุ่มตัวอย่างภาพปลาที่มีการเรียงตัวในแนวขวาง จำนวน 3 ภาพ จาก 10 ภาพ เพื่อนำมาคำนวณความชันของเส้นตัดเฉียงเฉลี่ย และองศาเส้นตัดเฉียงที่เหมาะสมในการมูมกับเส้นตัดตั้งฉาก

3.2.8 ปรับปรุงแก้ไขให้โปรแกรมมีความแม่นยำมากขึ้น และเปรียบเทียบค่าความแม่นยำของการทดสอบในแต่ละครั้ง

3.2.9 สรุปผลและเขียนรายงาน

4 ผลการดำเนินโครงการและวิจารณ์

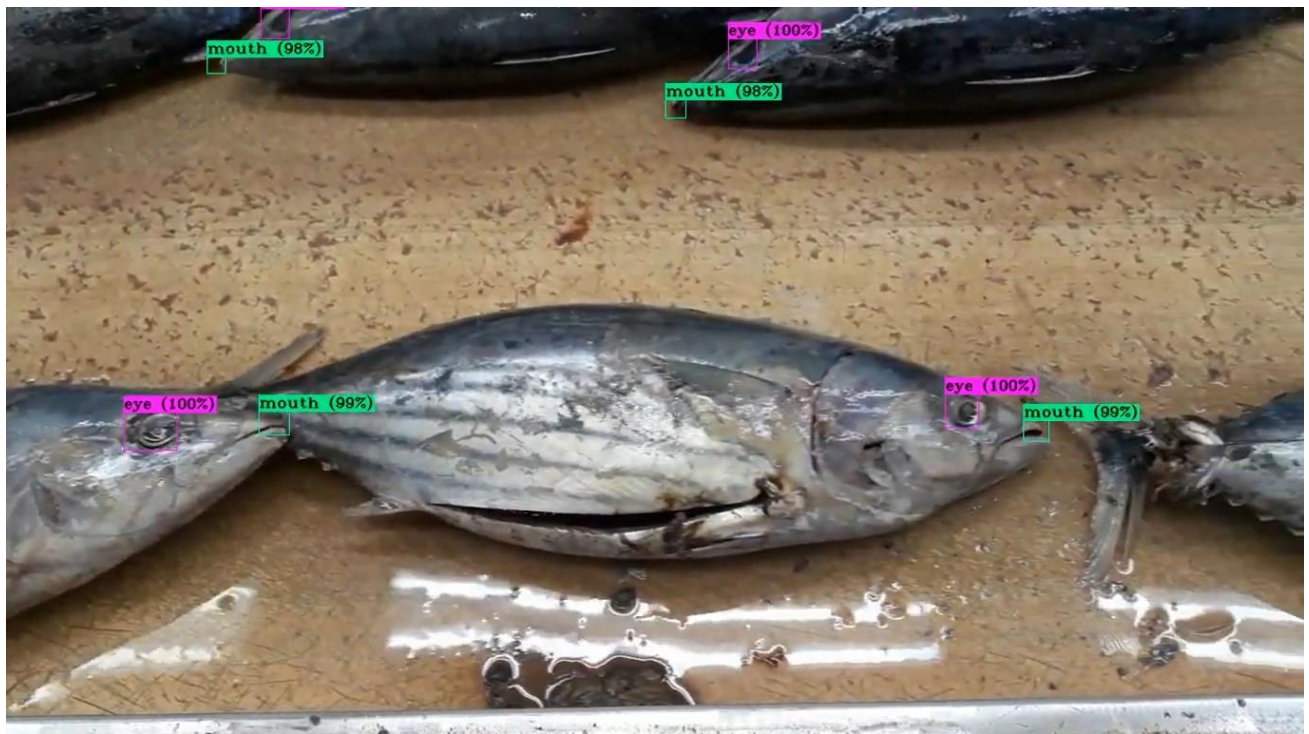
4.1. ผลจากการตรวจจับอวัยวะที่สนใจ (บริเวณปากบนและตาปลา)



รูปที่ 18 ตัวอย่างการทำนายอวัยวะปลา (ตาและปากบน) สำหรับตัวอย่างสุ่มที่ 1



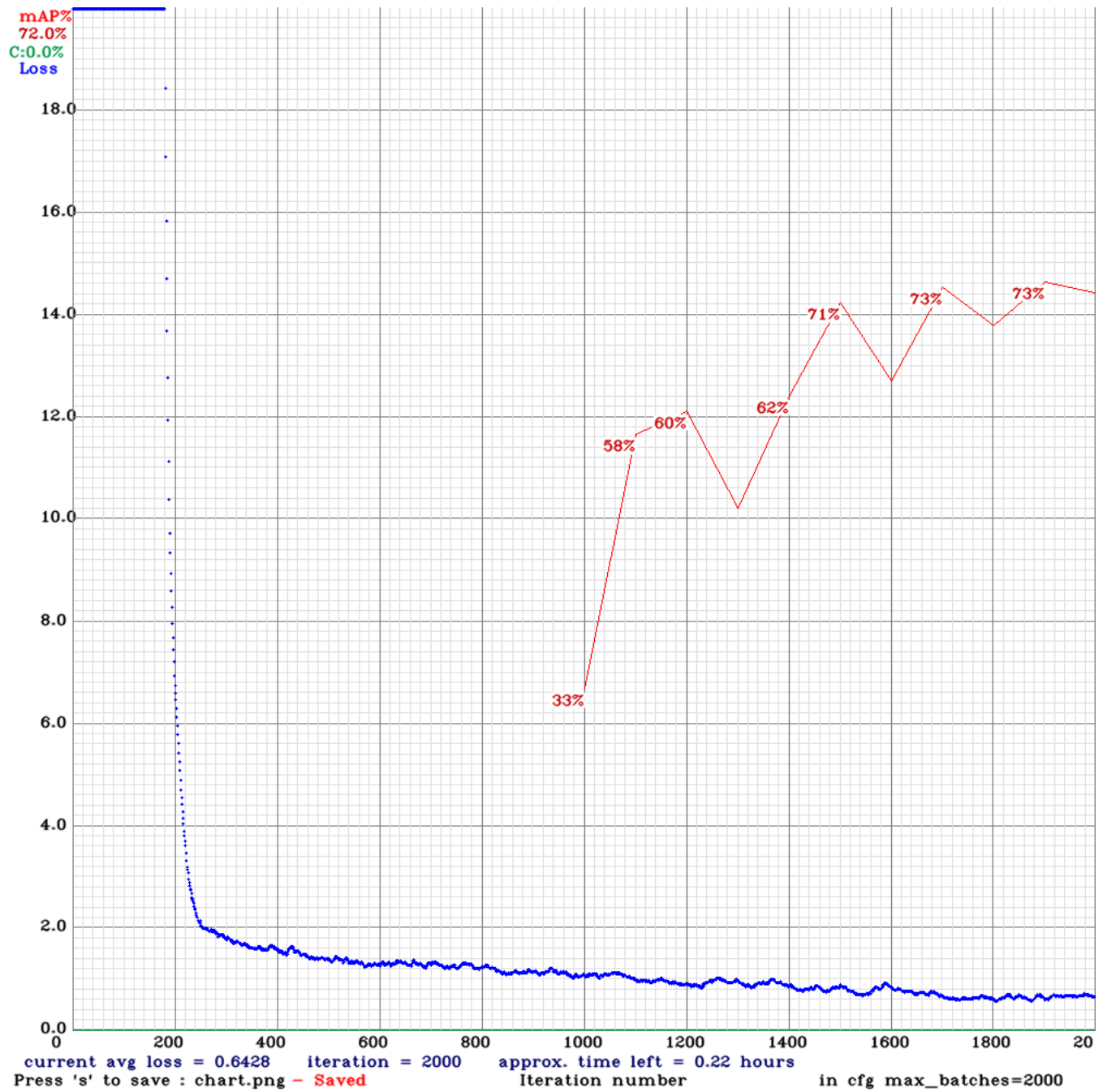
รูปที่ 19 ตัวอย่างการทำนายอวัยวะปลา (ตาและปากบน) สำหรับตัวอย่างสุ่มที่ 2



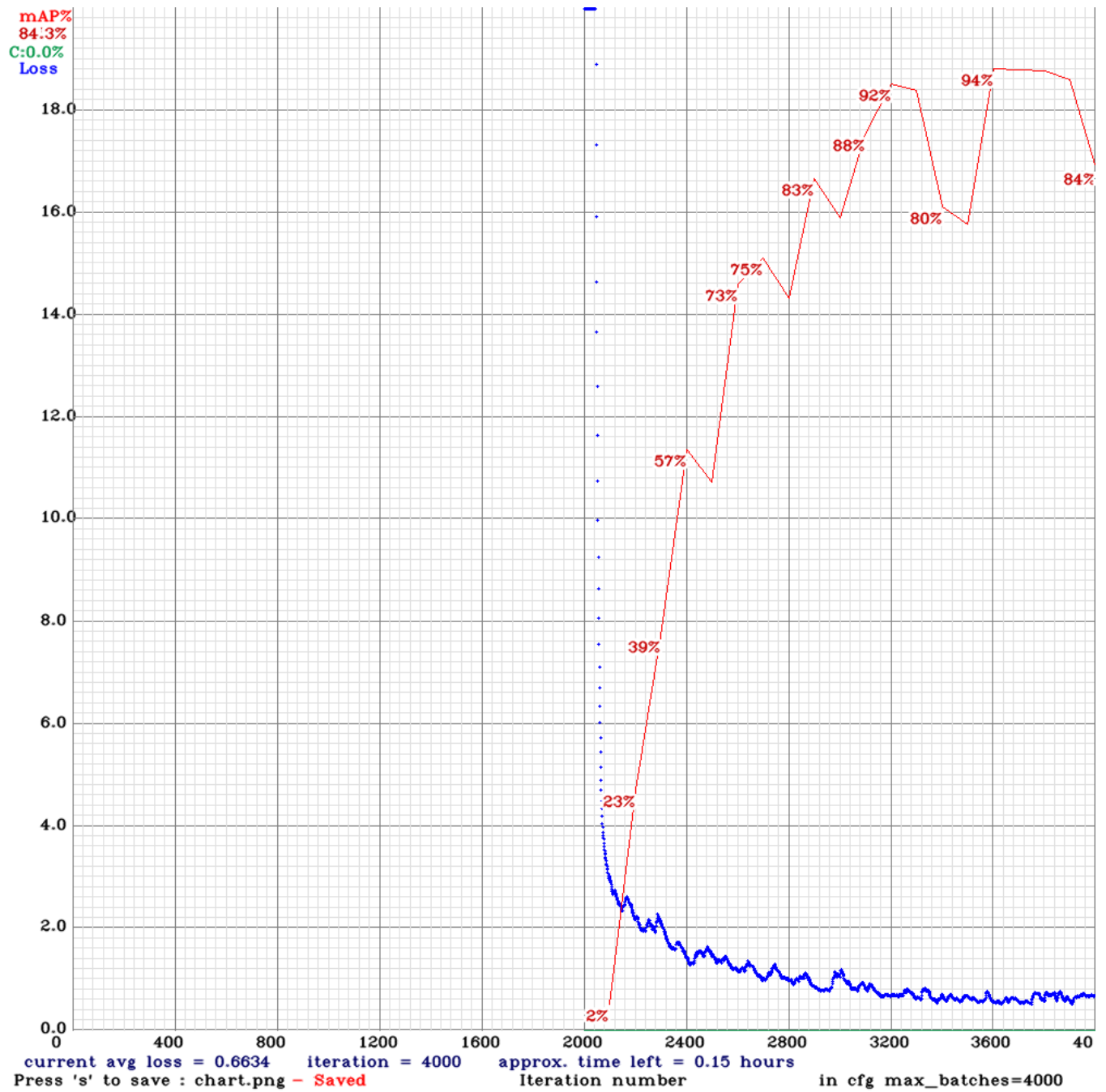
รูปที่ 20 ตัวอย่างการทำนายอวัยวะปลา (ตาและปากบน) สำหรับตัวอย่างสุ่มที่ 3



รูปที่ 21 ตัวอย่างการทำนายอวัยวะปลา (ตาและปากบน) สำหรับตัวอย่างสุมที่ 4



รูปที่ 22 ค่า mAP ช่วงการเทรนที่ 0-2000 รอบ



รูปที่ 23 ค่า mAP ช่วงการเทรนที่ 2000-4000 รอบ (ใช้ไฟล์ weight ที่ 2000 รอบเป็น pretrain-weight)


```

dongyang@dongyang-server: ~/Downloads/Aom_am_pls_dont_delete_this/aom_ws/darknet
File Edit View Search Terminal Help
Loading weights from /home/dongyang/Downloads/Aom_am_pls_dont_delete_this/aom_ws
/Tuna_dehead/model/trained_weights/yolo-obj_best.weights...
  seen 64, trained: 230 K-images (3 Kilo-batches_64)
Done! Loaded 107 layers from weights-file

  calculation mAP (mean average precision)...
  Detection layer: 82 - type = 28
  Detection layer: 94 - type = 28
  Detection layer: 106 - type = 28
12
  detections_count = 127, unique_truth_count = 51
class_id = 0, name = eye, ap = 96.00%          (TP = 24, FP = 1)
class_id = 1, name = mouth, ap = 75.26%       (TP = 20, FP = 7)

  for conf_thresh = 0.25, precision = 0.85, recall = 0.86, F1-score = 0.85
  for conf_thresh = 0.25, TP = 44, FP = 8, FN = 7, average IoU = 63.86 %

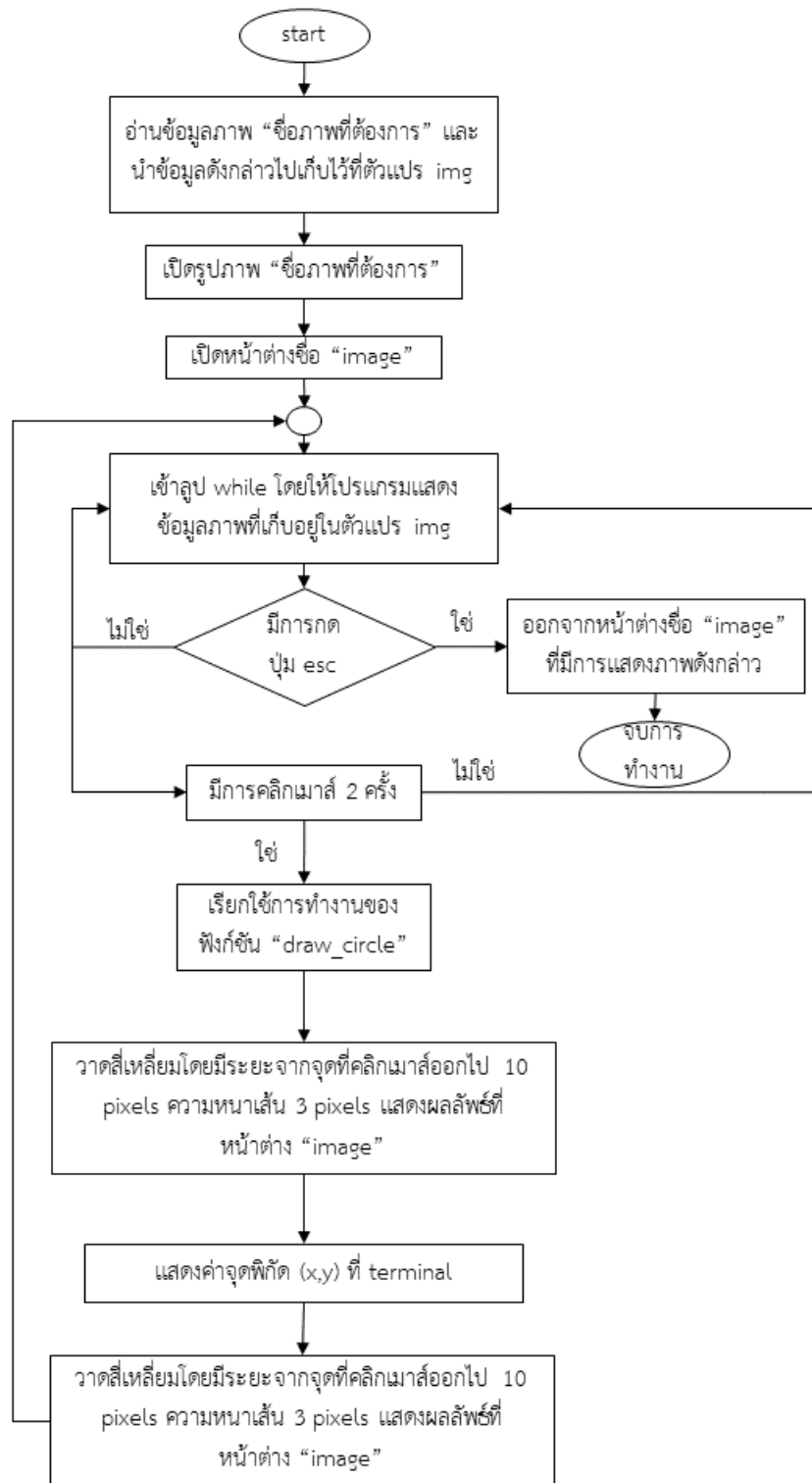
  IoU threshold = 50 %, used Area-Under-Curve for each unique Recall
  mean average precision (mAP@0.50) = 0.856302, or 85.63 %
Total Detection Time: 0 Seconds

Set -points flag:
  -points 101` for MS COCO
  -points 11` for PascalVOC 2007 (uncomment `difficult` in voc.data)

```

รูปที่ 24 ค่าสำหรับประเมินประสิทธิภาพของโมเดลการตรวจจับวัตถุ

4.2. คำสั่งเพื่อหาพิกัดจุดในภาพ เพื่อในการเขียนโปรแกรมจำลอง



```

Double_click_point.py x mouse_advance.py x draw_line.py x distance into eq. of ellipse.py x
1 import cv2
2 import numpy as np
3
4
5
6 # mouse callback function
7 def draw_circle(event,x,y,flags,param):
8     if event == cv2.EVENT_LBUTTONDBLCLK:
9         # cv2.circle(img,(x,y),10,(255,0,0),0)
10        cv2.rectangle(img,(x-10,y-10),(x+10,y+10),(0,255,0),3)
11        print(x,y)
12
13 # Create a black image, a window and bind the function to window
14 # img = np.zeros((512,512,3), np.uint8)
15 img = cv2.imread('TUNAClip_00_00_00_998.jpg',1)
16 cv2.namedWindow('image')
17 cv2.setMouseCallback('image',draw_circle)
18
19 while(1):
20     cv2.imshow('image',img)
21     if cv2.waitKey(20) & 0xFF == 27:
22         break
23 cv2.destroyAllWindows()
24

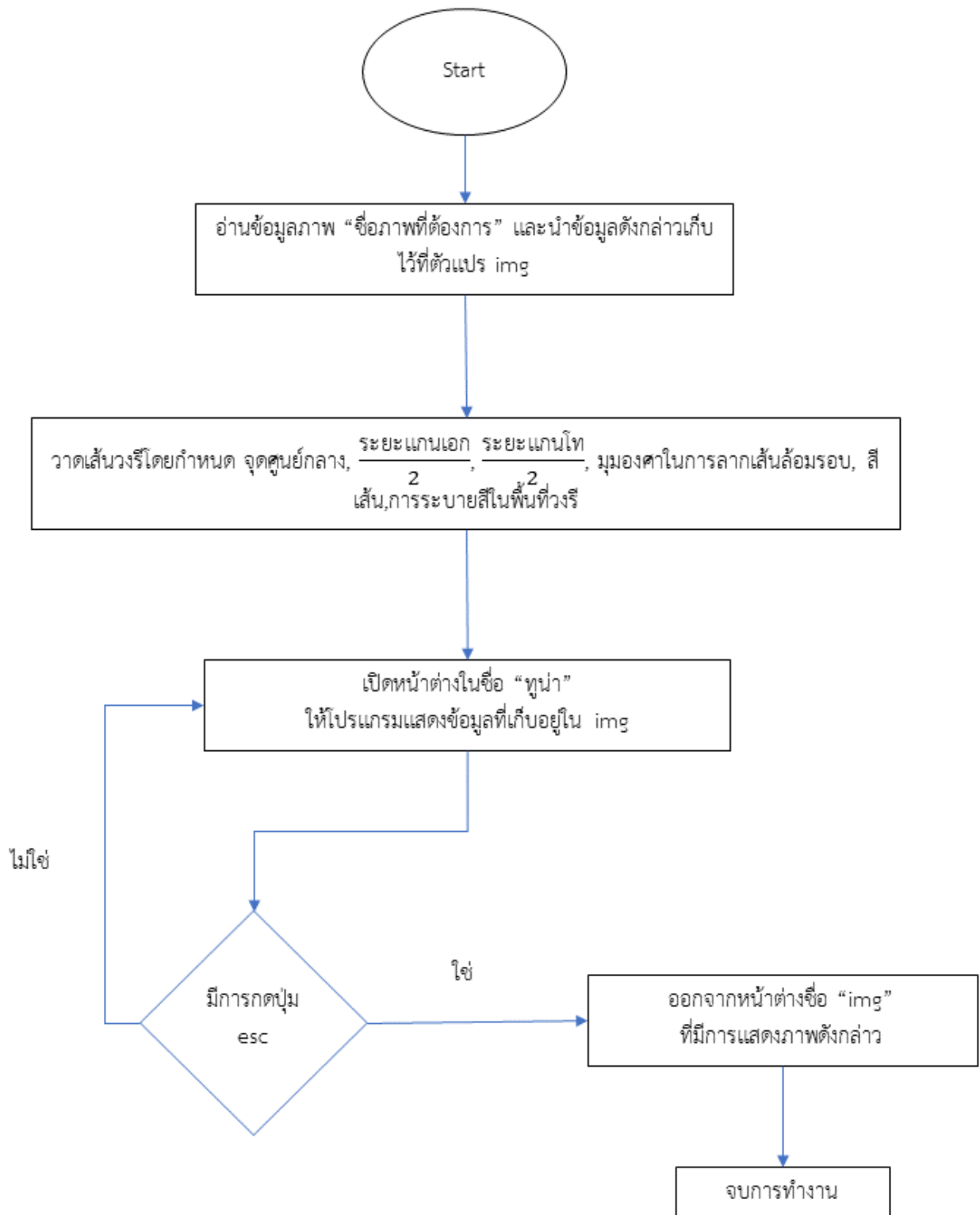
```

รูปที่ 25 คำสั่งใน OpenCV สำหรับหาพิกัดจุดต่างๆในภาพตามที่ต้องการโดยการคลิกเมาส์



รูปที่ 26 ผลลัพธ์จากคำสั่งสำหรับหาพิกัดจุดต่างๆในภาพตามที่ต้องการโดยการคลิกเมาส์

4.3. คำสั่งจำลองการลากเส้นตรงโดยใช้จุดอ้างอิงสองจุด (บริเวณปากบนและตาปลา) และผลลัพธ์



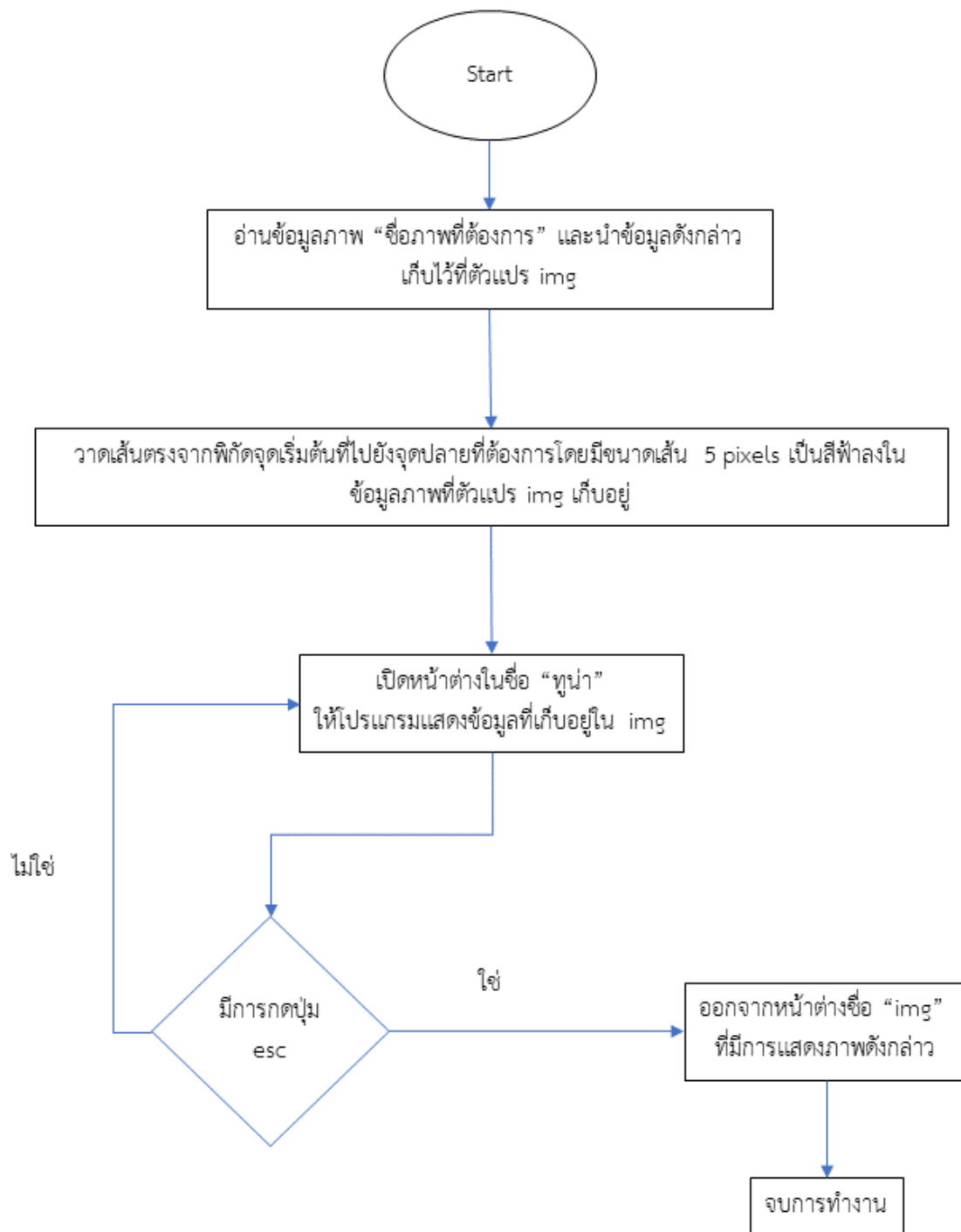
```
Double_click_point.py x mouse_advance.py x draw_line.py ● dis
1 #draw straight line from two double click point
2 # Draw a diagonal blue line with thickness of 5 px
3 #img = cv2.line(img,(x,y),(0,0),(255,0,0),5)
4
5 import numpy as np
6 import cv2
7 img = cv2.imread('Image1.png',1)
8
9 # Draw a diagonal blue line with thickness of 5 px
10 img = cv2.line(img,(742,853),(582,839),(255,0,0),5)
11 cv2.imshow('tuna',img)
12 k = cv2.waitKey(0)
13
14
```

รูปที่ 27 คำสั่งใน OpenCV สำหรับลากเส้นตรงโดยใช้จุดอ้างอิงสองจุด



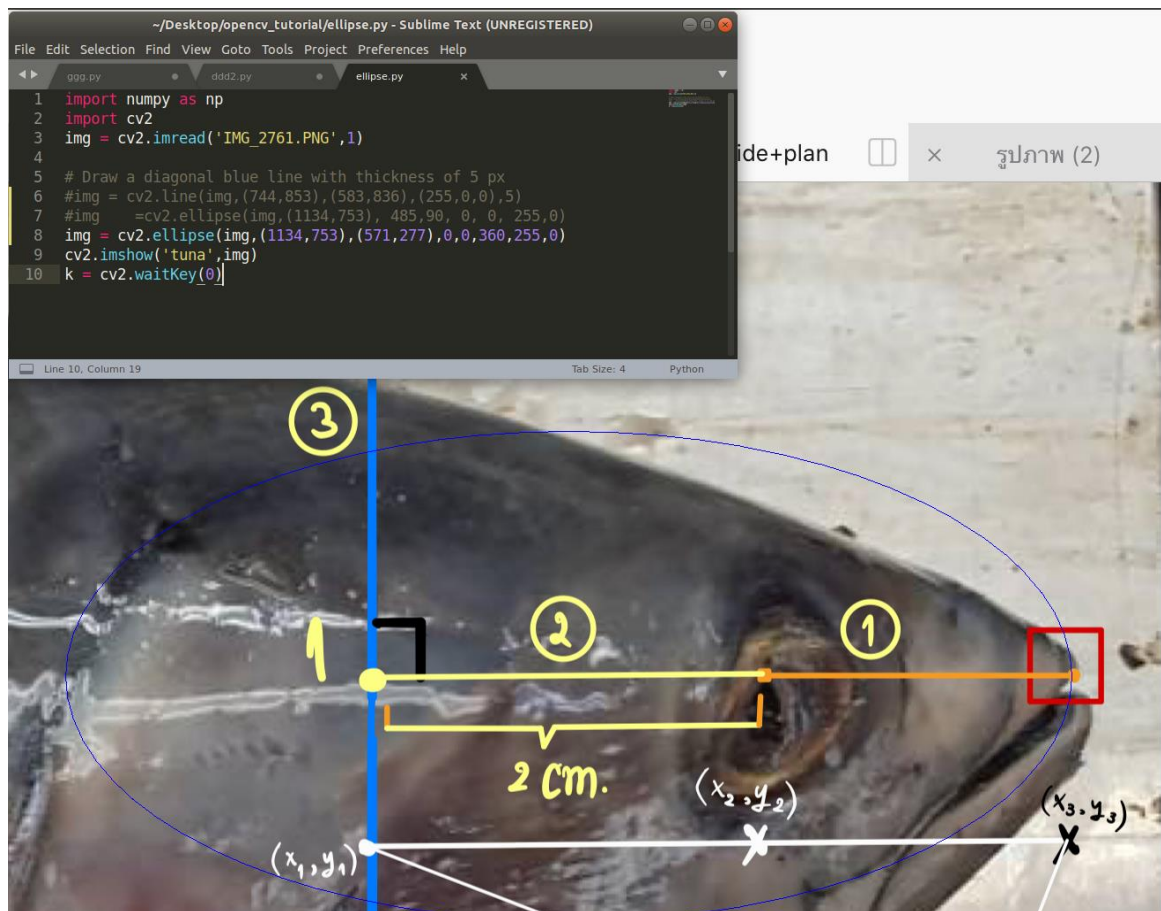
รูปที่ 28 ผลลัพธ์จากคำสั่งสำหรับลากเส้นตรงโดยใช้จุดอ้างอิงสองจุด

4.4. คำสั่งจำลองการลากเส้นวงรี และผลลัพธ์



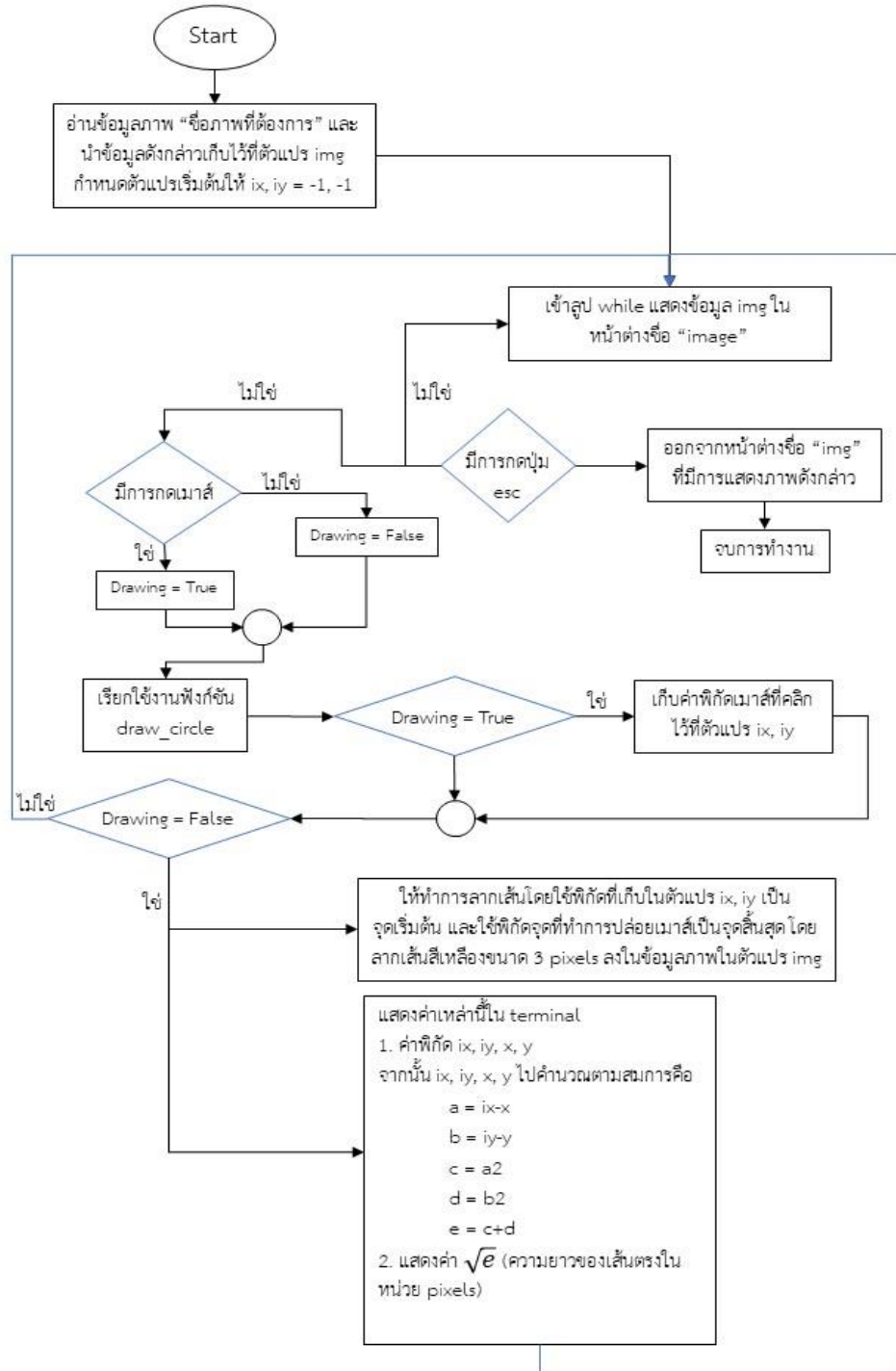

```
Double_click_point.py x ellipse.py x mouse_advance.py x draw_line.py
1 import numpy as np
2 import cv2
3 img = cv2.imread('Image1.png',1)
4
5 # Draw a diagonal blue line with thickness of 5 px
6 #img = cv2.line(img,(744,853),(583,836),(255,0,0),5)
7 #img =cv2.ellipse(img,(1134,753), 485,90, 0, 0, 255,0)
8 img = cv2.ellipse(img,(1134,753),(571,277),0,0,360,255,0)
9 cv2.imshow('tuna',img)
10 k = cv2.waitKey(0)]
```

รูปที่ 29 คำสั่งใน OpenCV สำหรับลากเส้นลากเส้นวงรี



รูปที่ 30 ผลลัพธ์จากคำสั่งสำหรับลากเส้นลากเส้นวงรี

- 4.5. คำสั่งที่ใช้ในการปรับเทียบ(calibrate) ขนาดของพิกเซลของภาพที่ถ่ายกับมาตรวัดจากวัตถุจริง (หน่วยเซนติเมตร) โดยการคลิกเมาส์ลากจากจุดเริ่มต้นไปยังจุดสิ้นสุดที่ต้องการวัด และผลลัพธ์ของความยาวของเส้นที่ลากและพิกัด




```

1 import numpy as np
2 import cv2 as cv
3 import math
4 drawing = False # true if mouse is pressed
5 mode = True # if True, draw rectangle. Press 'm' to toggle to curve
6 ix,iy = -1,-1
7 # mouse callback function
8 def draw_circle(event,x,y,flags,param):
9     global ix,iy,drawing,mode
10    if event == cv.EVENT_LBUTTONDOWN:
11        drawing = True
12        ix,iy = x,y
13    # elif event == cv.EVENT_MOUSEMOVE:
14    #     if drawing == True:
15    #         if mode == True:
16    #             cv.line(img,(x,y),(x,y),(0,255,0),3)
17    #         else:
18    #             cv.circle(img,(x,y),5,(0,0,255),-1)
19    elif event == cv.EVENT_LBUTTONUP:
20        drawing = False
21        if mode == True:
22            cv.line(img,(ix,iy),(x,y),(0,255,0),3)
23            print(ix,iy,x,y)
24            a=ix-x
25            b=iy-y
26            c=math.pow(a, 2)
27            d=math.pow(b, 2)
28            e=c+d
29            print(math.sqrt(e))
30        else:
31            cv.circle(img,(x,y),5,(0,0,255),-1)
32
33
34 #img = np.zeros((512,512,3), np.uint8)
35 img = cv.imread('TUNAcclip_00_01_57_865.jpg',1)
36 cv.namedWindow('image')
37 cv.setMouseCallback('image',draw_circle)
38 while(1):
39     cv.imshow('image',img)
40     k = cv.waitKey(1) & 0xFF
41     if k == ord('m'):
42         mode = not mode
43     elif k == 27:
44         break
45 cv.destroyAllWindows()

```

รูปที่ 31 คำสั่งใน OpenCV สำหรับปรับเทียบ(calibrate) ขนาดของพิกเซลของภาพที่ถ่ายกับมาตรวัดจากวัตถุจริง



รูปที่ 32 ผลลัพธ์ในรูปของเส้นตรงจากจุดเริ่มต้นไปยังจุดสิ้นสุดที่ต้องการวัด

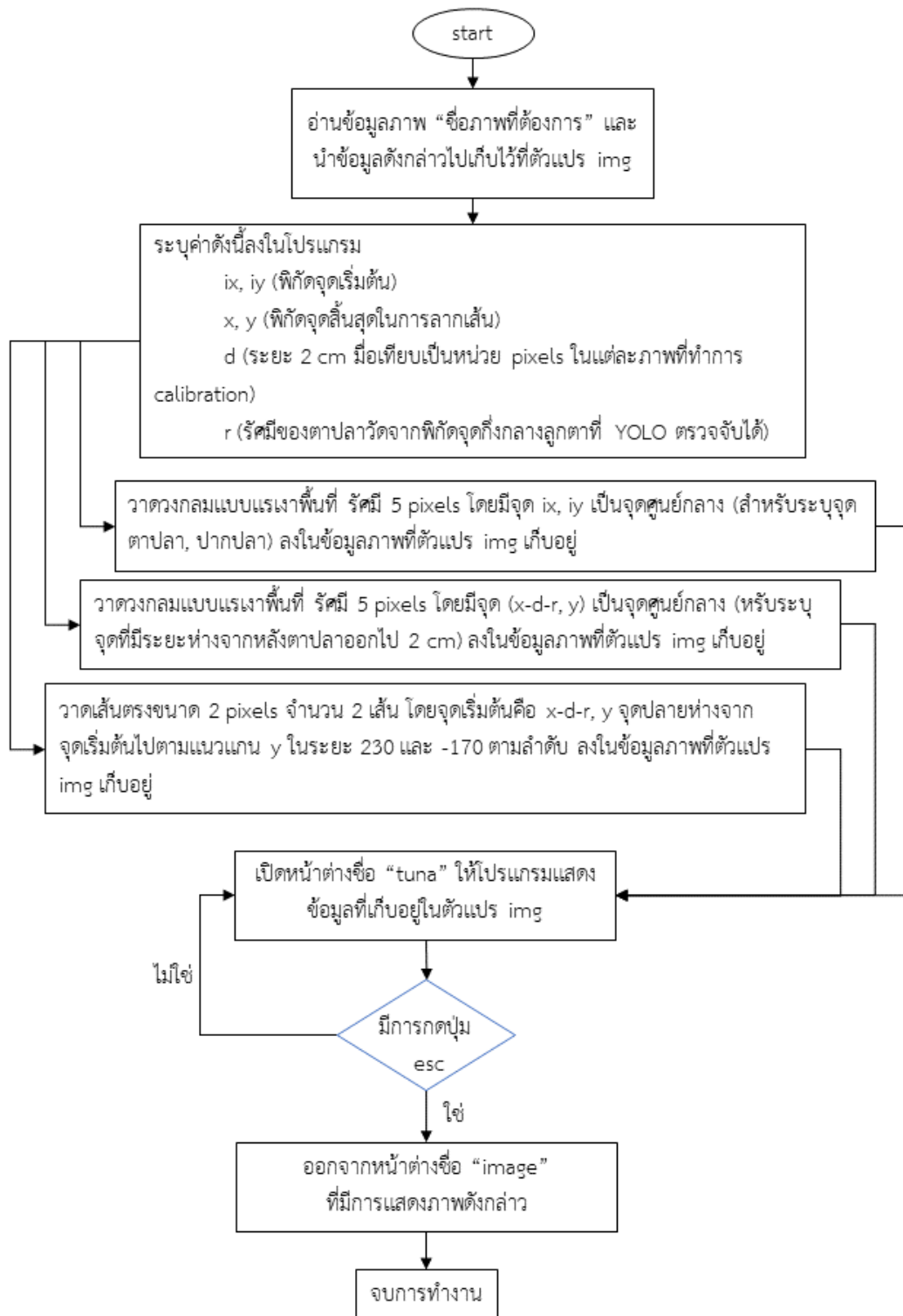
```

thalach@thalach-Vostro-3670: ~/Desktop/opencv_tutorial
File Edit View Search Terminal Help
thalach@thalach-Vostro-3670:~/Desktop/opencv_tutorial$ python mouse_advance.py
(281, 639, 1687, 731)
1409.00674235
(971, 714, 971, 714)
0.0
thalach@thalach-Vostro-3670:~/Desktop/opencv_tutorial$ python mouse_advance.py
(496, 811, 1836, 381)
1407.302384

```

รูปที่ 33 ผลลัพธ์จากคำสั่งที่แสดงพิกัดจุดเริ่มต้น พิกัดจุดสิ้นสุด และความยาวของเส้นตรงในหน่วยพิกเซล

4.6. เส้นตัดสำหรับใช้ตัดหัวปลา



```
~/Desktop/opencv_tutorial/4point_new.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
darknet.py 4point.py 4point_new.py 4_point.py darknet_images_aom.py
1 import numpy as np
2 import cv2
3 img = cv2.imread('TUNAcclip_00_00_26_969.jpg',1)
4
5 # Draw a diagonal blue line with thickness of 5 px
6 #img = cv2.line(img,(744,853),(583,836),(255,0,0),5)
7 #img =cv2.ellipse(img,(1134,753), 485,90, 0, 0, 255,0)
8 #recieve point from yolo
9 ix=1000
10 iy=592
11 x=846
12 y=597
13 d=43
14 r=27
15
16 #mouth and eye point
17 img=cv2.circle(img,(ix,iy), 5, (255,0,0), -1)
18 cv2.circle(img,(x,y), 5, (0,255,0), -1)
19
20 #draw 2cm point
21 cv2.circle(img,(x-d-r,y), 5, (0,0,255), -1)
22
23 #point4 draw line
24 cv2.circle(img,(x-d-r,y+230), 5, (0,255,255), -1)
25 cv2.circle(img,(x-d-r,y-107), 5, (0,255,255), -1)
26
27 #draw line
28 cv2.line(img,(x-d-r,y),(x-d-r,y+230),(255,0,0),2)
29 cv2.line(img,(x-d-r,y),(x-d-r,y-107),(255,0,0),2)
30
31 # #draw Oblique line
32 # cv2.line(img,(x-d-r,y),(1415,665),(0,0,255),2)
33
34
35
36 cv2.imshow('tuna',img)
37 k = cv2.waitKey(0)
```

รูปที่ 34 โค้ด OpenCV สำหรับวาดวงกลมเพื่อใช้แสดงจุดอ้างอิง โค้ด OpenCV สำหรับวาดเส้นตัดตรง



รูปที่ 35 ผลลัพธ์ในที่แสดงในรูปจุดวงกลมที่ใช้ประกอบการลากเส้นตัด

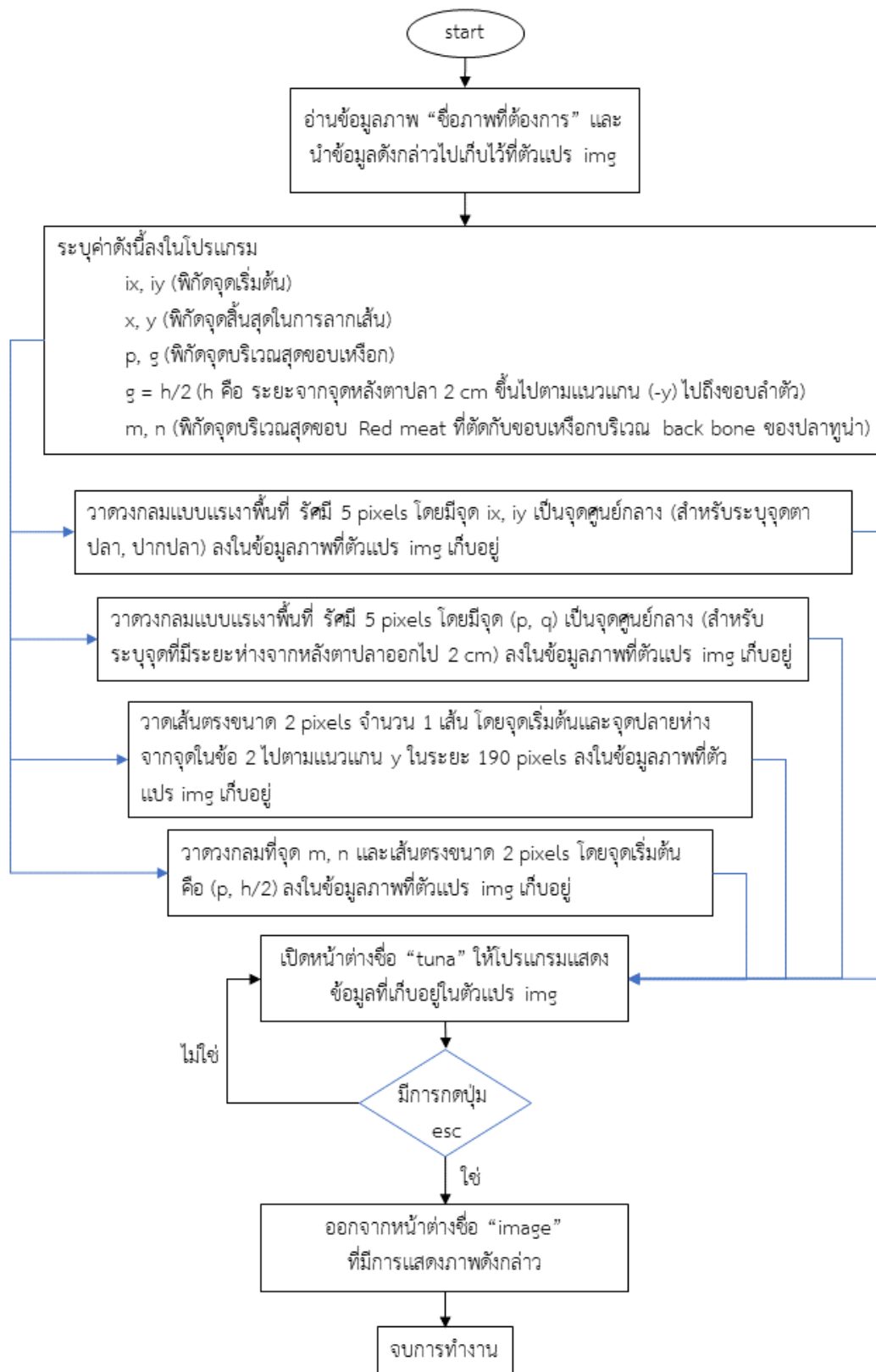


รูปที่ 36 ผลลัพธ์ในที่แสดงในรูปเส้นตรงเพื่อแสดงเส้นทางสำหรับตัดหัวปลาที่ 1



รูปที่ 37 ผลลัพธ์ในที่แสดงในรูปเส้นตรงเพื่อแสดงเส้นทางสำหรับตัดหัวปลาที่ 2

เมื่อลากเส้นตัดตรงซึ่งมีระยะห่างจากหลังตาปลาเป็นระยะ 2 เซนติเมตร จะเห็นว่าเส้นตัดตรงไม่ลากผ่านพื้นที่ในส่วนของ Red meat ซึ่งขัดแย้งกับรูปเงื่อนไขการตัดหัวปลาที่โรงงานवादมาให้ โดยอาจเกิดความผิดพลาดจากการกระยะตัดของผู้กำหนดเงื่อนไข ทั้งนี้ทางผู้จัดทำได้ยึดภาพที่ทางโรงงานवादมาให้เป็นหลัก จึงได้ทำการปรับเปลี่ยนจากเงื่อนไขเดิมคือเส้นตัดตรงห่างจากหลังตาปลาเป็นระยะ 2 เซนติเมตร เป็นเงื่อนไขใหม่คือเส้นตัดตรงบริเวณสุดขอบเหงือก ดังรูปที่ 39



```

~/Desktop/opencv_tutorial/4_point.py - Sublime Text (UNREC
File Edit Selection Find View Goto Tools Project Preferences Help
darknet.py 4_point.py Double_click_point.py darknet_images_aom.i
1 import numpy as np
2 import cv2
3 img = cv2.imread('TUNAClip_00_00_54_937.jpg',1)
4
5 # Draw a diagonal blue line with thickness of 5 px
6 #img = cv2.line(img,(744,853),(583,836),(255,0,0),5)
7 #img = cv2.ellipse(img,(1134,753), 485,90, 0, 0, 255,0)
8 #recieve point from yolo
9 (ix,iy)=(1648, 770)
10 (x,y)=(1536, 764)
11 (p,q)=(1294, 799)
12 (m,n)=(1405, 663)
13
14
15
16 #mouth and eye point
17 img=cv2.circle(img,(ix,iy), 5, (0,255,255), -1)
18 cv2.circle(img,(x,y), 5, (0,255,255), -1)
19 cv2.circle(img,(p,q), 5, (0,255,255), -1)
20
21
22 #draw line
23 cv2.line(img,(p,q-190),(p,q+190),(255,0,0),2)
24
25
26 #draw Oblique line
27 cv2.circle(img,(m,n), 5, (0,255,255), -1)
28 cv2.line(img,(p,q-69),(m,n),(0,0,255),2)
29
30
31 cv2.imshow('tuna',img)
32 k = cv2.waitKey(0)

```

รูปที่ 38 โค้ด OpenCV สำหรับลากเส้นตัดตามเงื่อนไขแบบใหม่



รูปที่ 39 ผลลัพธ์การลากเส้นตัดตามเงื่อนไขแบบใหม่สำหรับตัดหัวปลาตัวที่ 1

ความชันของเส้นตัดเฉียงในรูปที่ 39 คือ -1.687 เส้นตัดเฉียงทำมุมกับเส้นตัดตั้งฉากเท่ากับ 58 องศา



รูปที่ 40 ผลลัพธ์การลากเส้นตัดตามเงื่อนไขแบบใหม่สำหรับตัดหัวปลาตัวที่ 2

ความชันของเส้นตัดเฉียงในรูปที่ 40 คือ -2 เส้นตัดเฉียงทำมุมกับเส้นตัดตั้งฉากเท่ากับ 62 องศา



รูปที่ 41 ผลลัพธ์การลากเส้นตัดตามเงื่อนไขแบบใหม่สำหรับตัดหัวปลาตัวที่ 3

ความชันของเส้นตัดเฉียงในรูปที่ 40 คือ -2 เส้นตัดเฉียงทำมุมกับเส้นตัดตั้งฉากเท่ากับ 63 องศา

โดยตัวอย่างผลลัพธ์การทดสอบตัดตามเงื่อนไขแบบใหม่ ซึ่งจะใช้การเลือกภาพจำนวน 10 ภาพที่เป็นภาพปลาที่มีการเรียงตัวในแนวขวาง และทำการสุ่มภาพเพื่อประเมินความชัน และมุมสำหรับลากเส้นตัดเฉียงที่เหมาะสมมาจำนวน 3 ภาพ

5 สรุปผลการดำเนินงานและข้อเสนอแนะ

ให้นำเสนอการสรุปผลจากการดำเนินงานของโครงการที่ได้ทำไป โดยทำการสรุปเนื้อหาสำคัญในแต่ละประเด็นให้ชัดเจน รวมทั้งข้อเสนอแนะ เพื่อใช้เป็นแนวทาง ในการปรับปรุงโครงการเพิ่มเติมในอนาคต

5.1. ความแม่นยำในการตรวจจับวัตถุ

ผลการประเมินประสิทธิภาพของโมเดลการตรวจจับวัตถุสามารถให้ค่าความแม่นยำมากกว่า 80% ตามที่ตั้งเป้าหมายไว้ ซึ่งสามารถบอกได้จากค่ากราฟ mAP (2000 ถึง 4000 รอบ) คือ 85.63%

โดยค่าสำหรับประเมินประสิทธิภาพของโมเดลการตรวจจับวัตถุมีดังนี้

detections_count = 127 หมายถึง จำนวนภาพทั้งหมดที่ใช้เป็น dataset สำหรับการสร้าง model ตรวจจับวัตถุ จำนวน 127 ภาพ

unique_truth_count = 51 หมายถึง จำนวนภาพสำหรับนำมา train model จำนวน 51 ภาพ

class_id = 0, name = eye, AP = 96.00% (TP = 24, FP = 1) หมายถึง คลาสลำดับที่ 0, ชื่อคลาส คือ eye, มีค่าความแม่นยำเฉลี่ยในการตรวจจับวัตถุในคลาส eye เท่ากับ 96.00% (แสดงว่าโปรแกรมสามารถระบุวัตถุสำหรับคลาส eye ได้ใกล้เคียงกับค่าที่มนุษย์กำหนดไว้คิดเป็น 96.00%) มีค่า True Positive = 24 ภาพ และ False Positive = 1 ภาพ

class_id = 1, name = mouth, AP = 75.26% (TP = 20, FP = 7) หมายถึง คลาสลำดับที่ 1, ชื่อคลาส คือ mouth, มีค่าความแม่นยำเฉลี่ยในการตรวจจับวัตถุในคลาส mouth เท่ากับ 75.26% (แสดงว่าโปรแกรมสามารถระบุวัตถุสำหรับคลาส mouth ได้ใกล้เคียงกับค่าที่มนุษย์กำหนดคิดเป็น 75.26%) มีค่า True Positive = 20 ภาพ และ False Positive = 1 ภาพ (นิยามความหมายของ True Positive และ False Positive อ้างอิงจากหัวข้อที่ 2.1.3 การประเมินประสิทธิภาพของโมเดลการตรวจจับวัตถุ)

for conf_thresh = 0.25 ย่อมาจาก confidence threshold หมายถึง เกณฑ์ค่าความเชื่อมั่นในผลการทำนายวัตถุโดยผู้เขียนโปรแกรมเป็นผู้กำหนดค่า ในที่นี้คือเท่ากับ 0.25 เมื่อกำหนดค่าดังกล่าวแล้วจะได้ค่าต่างๆ เฉพาะสำหรับ confidence threshold ที่ค่านั้นๆ นั่นคือ

1. precision = 0.85 คือ ค่าที่บอกว่าโปรแกรมทำนายว่าเป็นวัตถุชนิดนั้นจริง เป็นค่าความถูกต้อง 85%
2. recall = 0.86 คือ ค่าที่บอกว่าโปรแกรมทำนายได้ว่าเป็นวัตถุจริง เป็นอัตราส่วน 86% ของวัตถุที่ระบุว่าเป็นจริงทั้งหมด
3. TP = 44 คือ ค่า True Positive ของ 2 classes รวมกัน
4. FP = 8 คือ ค่า False Positive ของ 2 classes รวมกัน
5. FN = 7 คือ ค่า False Negative ของ 2 classes รวมกัน

6. average IoU = 63.86 % คือ ค่าเฉลี่ยของอัตราส่วนของพื้นที่ทับซ้อน (ระหว่างพื้นที่ที่โปรแกรมทำนายว่าเป็นวัตถุจริง กับวัตถุที่จริงที่มนุษย์ระบุไว้ใน LabelImage) ส่วนพื้นที่ทั้งหมดที่ถูกระบุว่าเป็นค่าวัตถุจริง ยิ่งค่า average IoU มีเปอร์เซ็นต์มาก หมายถึงโปรแกรมสามารถระบุวัตถุได้ใกล้เคียงกับค่าที่มนุษย์กำหนดไว้ นั่นแสดงถึงความสามารถของโปรแกรมในการทำตามเงื่อนไขที่มนุษย์ระบุ ในที่นี้คือ 63.86%
7. mean average precision (mAP@0.50) = 0.856302, or 85.63 % หมายถึง ค่ากลางของค่าเฉลี่ยความแม่นยำของการเทรนทั้งหมด (ในที่นี้คือ 4000 รอบ) ซึ่งหมายถึงเมื่อคิดเป็นเปอร์เซ็นต์การตรวจจับวัตถุของโปรแกรม 4000 รอบ สามารถระบุค่าวัตถุได้ใกล้เคียงกับค่าที่มนุษย์ระบุไว้มีค่าความถูกต้องคือ 85.63 %

5.2. ตำแหน่ง และค่าความชันของเส้นตัดเฉียงที่เหมาะสม

จากการสุ่มตัวอย่างภาพปลาทุ่นเป็นจำนวน 3 รูป จาก 10 รูป เพื่อนำมาคำนวณหาตำแหน่ง และค่าความชันของเส้นตัดเฉียง จากค่าความชันจำนวน 3 ค่า คือ -1.687, -2, -2 ตามลำดับ จะได้ค่าเฉลี่ย (Mean)= -1.89567, ค่าเบี่ยงเบนมาตรฐาน (SD)= 0.18071, ค่าแปรปรวน (Variance)= 0.03266, ค่าเฉลี่ยของประชากร (Population mean)= 0.14755, ส่วนเบี่ยงเบนมาตรฐานของประชากร (Zigma)= 0.02177 และจากองศาที่เส้นตัดเฉียงทำมุมกับเส้นตัดตั้งฉากจำนวน 3 ค่า คือ 58, 62, 63 ตามลำดับ จะได้ค่าเฉลี่ย (Mean)= 61, ค่าเบี่ยงเบนมาตรฐาน (SD)= 2.64575, ค่าแปรปรวน (Variance)= 7, ค่าเฉลี่ยของประชากร (Population mean)= 2.16025, ส่วนเบี่ยงเบนมาตรฐานของประชากร (Zigma)= 4.66667 จึงสรุปได้ว่าค่าความชันเฉลี่ยที่เหมาะสมสำหรับเส้นตัดคือ -1.89567 และองศาที่เส้นตัดเฉียงทำมุมกับเส้นตัดตั้งฉากที่เหมาะสมคือ 61

5.3. แนวทางในการปรับปรุงโครงงานเพิ่มเติม

- 1) สำหรับเครื่องตัดพลังน้ำแรงดันสูงควรมีการเพิ่มระบบการจัดวางเรียงปลาแนวเดียวกันเพื่อให้สะดวกต่อการตัดและลดขั้นตอนสำหรับการเขียนโปรแกรมที่ซับซ้อน
- 2) ควรวัดขนาดจริงของปลาทุ่น และระยะห่างจากปากถึงตา(สำหรับปลาแต่ละตัว) เพื่อให้ได้สเกลสำหรับการคำนวณปรับเทียบขนาดฟิสิกเซลเป็นเซนติเมตรที่แม่นยำขึ้น
- 3) ปรับปรุงและเพิ่มความซับซ้อนของคำสั่งใน OpenCV เพื่อให้โมเดลสามารถตรวจจับและลากเส้นตัดของปลาแต่ละตัวได้แบบเรียลไทม์

6 บรรณานุกรม

- AlexeyAB. (22 March 2021). *Yolo v4, v3 and v2 for Windows and Linux*. เข้าถึงได้จาก <https://github.com/AlexeyAB/darknet#yolo-v4-v3-and-v2-for-windows-and-linux>
- Andrzej Dowgiallo. (2008). The effect of cutting and fish-orientation systems on the deheading yield of carp. *International journal of food science & technology*, 1688-1692.
- Appsilon DataScience. Michal Maj. (23 march 2021). *Object Detection and Image Classification with YOLO*. เข้าถึงได้จาก kdnuggets: <https://www.kdnuggets.com/2018/09/object-detection-image-classification-yolo.html>
- F Johnsson, R C Zijerveld, J C Schouten, C M van den Bleek, และ B Leckner. (2000). Characterization of fluidization regimes by time-series analysis of pressure fluctuations. *International Journal of Multiphase Flow*, 26, 663-715.
- Jonathan Hui. (7 March 2018). *mAP (mean Average Precision) for Object Detection*. เข้าถึงได้จาก jonathan-hui.medium.com: <https://jonathan-hui.medium.com/map-mean-average-precision-for-object-detection-45c121a31173>
- Joseph Chet Redmon. (22 march 2021). *YOLO: Real-Time Object Detection*. เข้าถึงได้จาก pjreddie: <https://pjreddie.com/darknet/yolo/>
- Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn , และ Andrew Zisserman. (2021). The PASCAL Visual Object Classes (VOC) Challenge. *International Journal of Computer*, 1-34.
- Nikko Co., Ltd. (22 March 2021). *Fish deheading machine NOH-100*. เข้าถึงได้จาก <https://www.directindustry.com/prod/nikko-co-ltd/product-174818-1758360.html>
- Poramet Arromdee, และ Vladimir I Kuprianov. (2012). Combustion of peanut shells in a cone-shaped bubbling fluidized-bed combustor using alumina as the bed material. *Applied Energy*, 97, 470-482.
- Tewfree. (22 April 2020). วงรี (Ellipse). เข้าถึงได้จาก <https://www.tewfree.com/>: <https://www.tewfree.com/วงรี-ellipse-คณิตศาสตร์-ม-4>
- Tuenong Group. (22 March 2021). ความชื้น. เข้าถึงได้จาก Tuenongfree: <https://tuenongfree.xyz/ความชื้น/>
- เกษตรฮับ เซ็นเตอร์. (15 ตุลาคม 2020). *kasethub*. เรียกใช้เมื่อ 22 มีนาคม 2021 จาก <https://www.kasethub.co.th/ปลาทูน่ากระป๋อง.html>
- นิชากร ศรีเพชรดี. (1 สิงหาคม 2018). ไม่หยุดประมงผิดกฎหมายในปี 2048 จะไม่มีปลา (ทูน่า) ให้จับอีกต่อไป. เรียกใช้เมื่อ 22 มีนาคม 2021 จาก <https://greennews.agency/?p=17905>

สินีนากู แซ่โล้ง, และ เพ็ญพิชชา จันทน์เครือวงศ์. (23 March 2008). ความรู้เบื้องต้นเกี่ยวกับเรขาคณิตศาสตร์ > ระยะทางระหว่างจุดสองจุด. เข้าถึงได้จาก <https://sites.google.com/site/mathqueen1043:https://sites.google.com/site/mathqueen1043/khwam-ru-beuxng-tn-keiyw-kab-rekhakhnit-sastr/raya-thang-rahwang-cud-sxng-cud>

สุรพล อุปติสสกุล. (2521). สถิติ: การวางแผนการทดลองเบื้องต้น. กรุงเทพฯ: มหาวิทยาลัยเกษตรศาสตร์.

7 ภาคผนวก

7.1. ภาคผนวก ก

คำสั่งใน OpenCV สำหรับพัฒนาโมเดลให้สามารถตรวจจับและลากเส้นตัดของปลาแต่ละตัวได้แบบเรียลไทม์

```
from ctypes import *
import math
import random
import os
import cv2
import numpy as np
import time
import darknet

def convertBack(x, y, w, h):
    xmin = int(round(x - (w / 2)))
    xmax = int(round(x + (w / 2)))
    ymin = int(round(y - (h / 2)))
    ymax = int(round(y + (h / 2)))
    return xmin, ymin, xmax, ymax

def cvDrawBoxes(detections, img):
    mouth_pose = None
    eye_pose = None

    for detection in detections:
        x, y, w, h = detection[2][0],\
            detection[2][1],\
            detection[2][2],\
            detection[2][3]
        xmin, ymin, xmax, ymax = convertBack(
```

```

        float(x), float(y), float(w), float(h))
pt1 = (xmin, ymin)
pt2 = (xmax, ymax)
#check class
if detection[0].decode() == "eye":
    #do something
    eye_pose = [x, y, w]
    cv2.rectangle(img, pt1, pt2, (0, 255, 0), 1)
    cv2.putText(img,
                detection[0].decode() +
                " [" + str(round(detection[1] * 100, 2)) + "]",
                (pt1[0], pt1[1] - 5), cv2.FONT_HERSHEY_SIMPLEX, 0.5,
                [0, 255, 0], 2)
if detection[0].decode() == "mouth":
    #do something
    mouth_pose = [x, y, w]
    cv2.rectangle(img, pt1, pt2, (255, 255, 0), 1)
    cv2.putText(img,
                detection[0].decode() +
                " [" + str(round(detection[1] * 100, 2)) + "]",
                (pt1[0], pt1[1] - 5), cv2.FONT_HERSHEY_SIMPLEX, 0.5,
                [255, 255, 0], 2)

#math 6 cm.
a=math.pow(eye_pose,2)
b=math.pow(mouth_pose,2)
c=math.sqrt()

#try to draw arc
if mouth_pose == None or eye_pose == None :
    pass
elif c < 105:

```

```

#in case of detects both eye and mouth
#draw arc
#mouth and eye point
#assume that 2cm=d=43pixels in whole Tuna
img=cv2.circle(img,(mouth_pose[0],mouth_pose[1]), 5, (255,0,0), -1)
cv2.circle(img,(eye_pose[0],eye_pose[1]), 5, (0,255,0), -1)

#draw 2cm point
cv2.circle(img,(eye_pose[0]-d-eye_pose[2]/2,eye_pose[1]), 5, (0,255,255), -1)

#point4 draw line
cv2.circle(img,(eye_pose[0]-d-eye_pose[2]/2,eye_pose[1]+230), 5, (0,255,255), -1)
cv2.circle(img,(eye_pose[0]-d-eye_pose[2]/2,eye_pose[1]-107), 5, (0,255,255), -1)

#draw line
cv2.line(img,(eye_pose[0]-d-eye_pose[2]/2,eye_pose[1]),(eye_pose[0]-d-
eye_pose[2]/2,eye_pose[1]+230),(255,0,0),2)
cv2.line(img,(eye_pose[0]-d-eye_pose[2]/2,eye_pose[1]),(eye_pose[0]-d-eye_pose[2]/2,eye_pose[1]-
107),(255,0,0),2)

else:
    pass
return img

netMain = None
metaMain = None
altNames = None

def YOLO():

    global metaMain, netMain, altNames

```



```

configPath = "./cfg/yolov4.cfg"
weightPath = "./yolov4.weights"
metaPath = "./cfg/coco.data"
if not os.path.exists(configPath):
    raise ValueError("Invalid config path `" +
                     os.path.abspath(configPath)+"`)")
if not os.path.exists(weightPath):
    raise ValueError("Invalid weight path `" +
                     os.path.abspath(weightPath)+"`)")
if not os.path.exists(metaPath):
    raise ValueError("Invalid data file path `" +
                     os.path.abspath(metaPath)+"`)")
if netMain is None:
    netMain = darknet.load_net_custom(configPath.encode(
        "ascii"), weightPath.encode("ascii"), 0, 1) # batch size = 1
if metaMain is None:
    metaMain = darknet.load_meta(metaPath.encode("ascii"))
if altNames is None:
    try:
        with open(metaPath) as metaFH:
            metaContents = metaFH.read()
            import re
            match = re.search("names *= *(.*)$", metaContents,
                              re.IGNORECASE | re.MULTILINE)
            if match:
                result = match.group(1)
            else:
                result = None
        try:
            if os.path.exists(result):
                with open(result) as namesFH:

```

```

        namesList = namesFH.read().strip().split("\n")
        altNames = [x.strip() for x in namesList]
    except TypeError:
        pass
    except Exception:
        pass
#cap = cv2.VideoCapture(0)
cap = cv2.VideoCapture("test.mp4")
cap.set(3, 1280)
cap.set(4, 720)
out = cv2.VideoWriter(
    "output.avi", cv2.VideoWriter_fourcc(*"MJPG"), 10.0,
    (darknet.network_width(netMain), darknet.network_height(netMain)))
print("Starting the YOLO loop...")

# Create an image we reuse for each detect
darknet_image = darknet.make_image(darknet.network_width(netMain),
                                   darknet.network_height(netMain),3)

while True:
    prev_time = time.time()
    ret, frame_read = cap.read()
    frame_rgb = cv2.cvtColor(frame_read, cv2.COLOR_BGR2RGB)
    frame_resized = cv2.resize(frame_rgb,
                               (darknet.network_width(netMain),
                                darknet.network_height(netMain)),
                               interpolation=cv2.INTER_LINEAR)

    darknet.copy_image_from_bytes(darknet_image,frame_resized.tobytes())

    detections = darknet.detect_image(netMain, metaMain, darknet_image, thresh=0.25)
    image = cvDrawBoxes(detections, frame_resized)

```

```
    image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
    print(1/(time.time()-prev_time))
    cv2.imshow('Demo', image)
    cv2.waitKey(3)
cap.release()
out.release()

if __name__ == "__main__":
    YOLO()
```

ประวัติผู้จัดทำ

ชื่อ-นามสกุล นางสาว ถลัชนันท์ สลัดทุกข์ เลขประจำตัวนักศึกษา 5910110122

ภาควิชาวิศวกรรมเครื่องกล สาขาวิศวกรรมเมคคาทรอนิกส์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์
ที่อยู่ปัจจุบัน 46/5 หมู่ 1 ตำบลท่าบอง อำเภอรโนด จังหวัดสงขลา

โทรศัพท์ ที่บ้าน – โทรศัพท์เคลื่อนที่ 080-5499896

สถานที่ทำงาน(ถ้ามี) -

โทรศัพท์ที่ทำงาน (ถ้ามี) - E-mail thalachanan.st@gmail.com

ระดับการศึกษา:

คุณวุฒิการศึกษา	จากโรงเรียน/สถาบัน	ปีการศึกษาที่จบ
มัธยมศึกษาตอนปลาย	โรงเรียนธิดานุเคราะห์	2559
มัธยมศึกษาตอนต้น	โรงเรียนธิดานุเคราะห์	2556