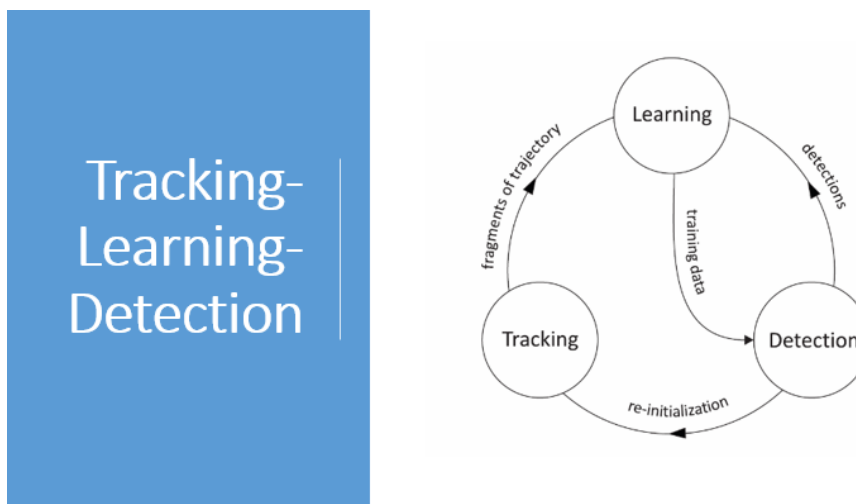


Tracking-Learning-Detection



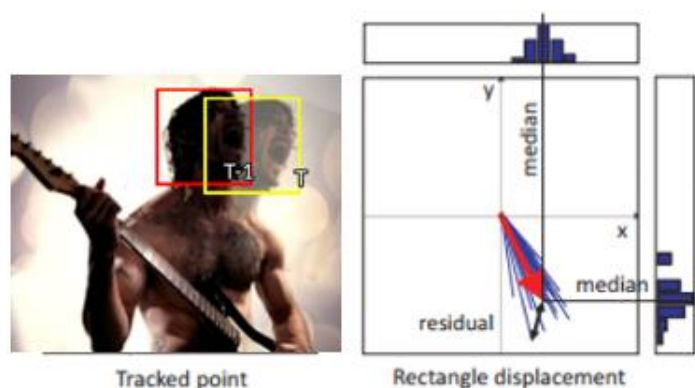
เป็นหนึ่งในกระบวนการที่สามารถ Track วัตถุอะไรก็ได้ ภายในรูปภาพ / วิดีโอ แบบ Long term โดยมีภาพตัวอย่างอย่างน้อย 1 รูป ซึ่งกระบวนการ TLD นี้ จะมีกระบวนการเรียนรู้ด้วยตัวเอง และ ยังมีความแม่นยำและความเร็วที่ดีอีกด้วย ซึ่งในกระบวนการ มีสามส่วนประกอบที่สำคัญคือ

Tracker จะเป็นการประมาณ วิธีของการเคลื่อนที่ของวัตถุในเฟรมที่ติดกัน และตัวTracker นั้นจะล้มเหลวและไม่สามารถกู้คืนได้ ถ้าหาก วัตถุหายไปจากภาพจึงต้องมี Detector เป็นตัวช่วย

Detector จะถือว่าทุกๆเฟรมจะมีความเป็นอิสระต่อกัน และ จะเป็นตัวที่ต้อง Scan ของภาพโดยจะ สังเกต และ เรียนรู้จากภาพที่เคยผ่านมา และ Detector จะทำให้สร้าง error อยู่สองอย่างคือ False positive และ False negative

Learning จากการสังเกตของ Tracker และ Detector จะทำให้ Learning ประมาณค่าต่างๆ และนำไปเรียนรู้ได้ เพื่อ นำไป Update ข้อมูลสำหรับการ Detect ในเฟรมต่อไปได้ ซึ่ง วิธีการ Tracking Detection Learning จะกล่าวโดยสรุปดังนี้

Tracking



รูป 1-1 แสดงตัวอย่างการติดตามด้วย Median-Flow Tracker

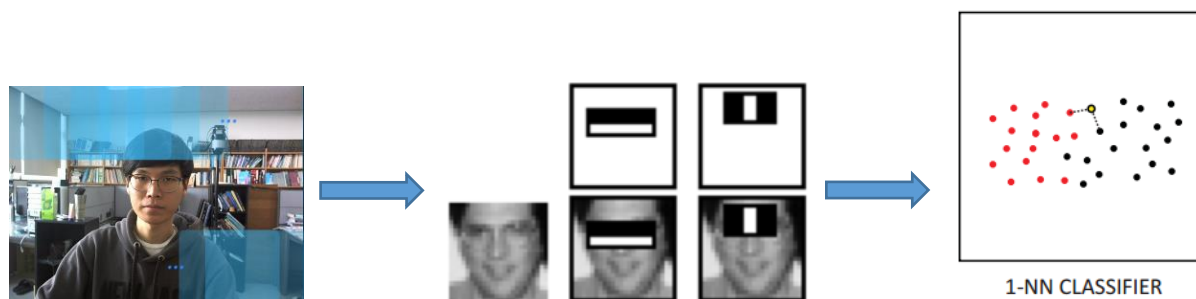
วิธีการ Track นี้ จะมีพื้นมาจากวิธี Median-Flow Tracker ซึ่งจะมีหน้าที่หา Median ของจุดสนใจทั้งหมดของสิ่งที่เราต้องการ Track ที่ได้จาก Bounding Box ของ Detector

ซึ่ง Tracker จะดูการประมาณการเคลื่อนที่ของ Median flow จากทิศทางและขนาด จึงทำให้ Bounding Box เคลื่อนที่ตามวัตถุได้

Median-flow จะต้องพลาดทุกครั้งเมื่อ Object เคลื่อนที่ออกจากกล้อง หรือ เคลื่อนที่ไวเกินไป ดังนั้น วิธีแก้ปัญหาคือ ดูระยะห่างระหว่าง จุดที่ track ล่าสุด กับ จุด median ถ้ามันห่างมากกว่า 10 px จะถือว่า จะไม่ return bounding box

และต้องพึ่งพา Detector เมื่อ Tracker เกิดความผิดพลาด

Detection



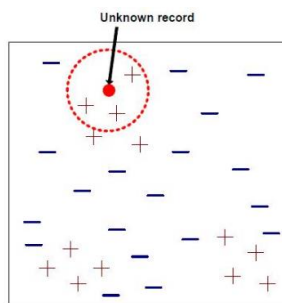
รูป 2-1 แสดงตัวอย่างขั้นตอนกระบวนการ Detection

จะ detect รูปโดยการ Scan รูปที่ได้รับมา โดยใช้วิธีการ Scanning- window grid โดย window size จะเริ่มจากขนาดเริ่มต้นของ Bounding box ของตัวโมเดล และเพิ่มขนาดต่อไปเป็น 1.2 เท่า เลื่อน window ไปทางขวา 10 %ของภาพ และเลื่อนทาง แนวตั้ง 10 %ของภาพ QVGA

ซึ่งแต่ละ windows ก็จะถูกหา คุณลักษณะ(feature) ด้วยวิธีการ Cascaded classifier โดยการใช้ Mask ในรูปแบบต่างๆ

และ ใช้ Integral image ในการหาผลต่างระหว่าง สีขาวกับสีดำ เพื่อ นำไปเป็น feature และส่งค่าไปหา Learning เพื่อ Classified

ส่วนการ Classified นั้น จะใช้วิธีการ Nearest Neighbor (NN) classifier ซึ่งจะดูว่า ข้อมูลที่ได้มา ล่าสุด คล้ายกับ Object ที่ต้องการ หรือไม่ ซึ่ง เราจะสร้างโครงสร้าง การใช้ Classifier 1 Nearest neighbor ว่า ใกล้ข้อมูลที่คล้ายรูปตัวอย่างหรือไม่ ถ้าคล้าย ก็ถือว่าใช่



รูป 2-2 แสดงตัวอย่างของ Nearest neighbor Classifier

Learning

การเรียนรู้ของ Classifier แบ่งออกเป็น 2 แบบ ใหญ่ๆ คือ

- Offline Learning

คือการ เรียนรู้ของตัวแบ่งแยกประเภท(Classifier) จาก เทรนนิ่งเซตที่เตรียมมาก่อนที่จะเริ่มใช้งาน

- Online learning

คือการ เรียนรู้ของตัวแบ่งแยกประเภท(Classifier) ระหว่างการทำงาน โดยขณะที่ทำงาน โปรแกรมก็จะเก็บ ตัวอย่าง Positive example และ Negative example ไปด้วย ซึ่ง ทั้งนี้ TLD จะใช้ การเรียนรู้ประเภท online learning

ซึ่งขั้นตอนกระบวนการเรียนรู้ของ TLD มีดังนี้คือ initialize ให้กับ Detector ใน เฟรมแรก และ คอย Update ตัว Detect ทุกๆเฟรม โดยใช้ P-expert และ N-expert ซึ่งจะกล่าวถึงดังนี้

- Initialization

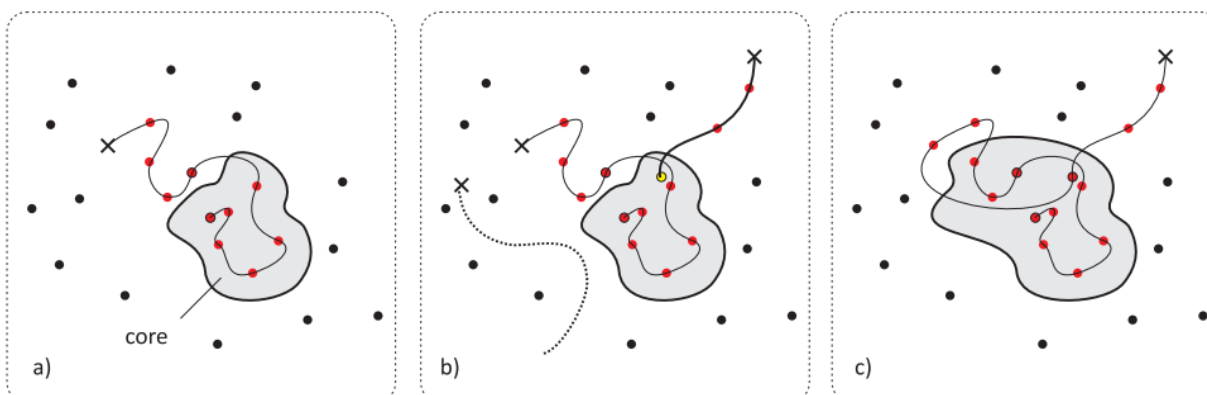
- ในเฟรมแรก เราจะ scan รูป โดยวิธี Scanning grid แล้ว เลือก 10 Bounding Boxes โดย ที่ใกล้กับ รูปตัวอย่างมากที่สุดเริ่มต้นมากที่สุด
- จากนั้น เราจะ Generate Bounding Box 20 แบบโดยการ (shift $\pm 1\%$, scale change $\pm 1\%$, in-plane rotation $\pm 10^\circ$) and add them with Gaussian noise ($\sigma = 5$) on pixels. ซึ่ง จะได้ผลลัพธ์คือ Positive และ Negative patch มาเป็นข้อมูล

- P-expert

Generate positive training example และ ถ้าความเหมือนแบบ ภาพตัวอย่างตอนแรก มากๆ แสดงว่า ใกล้ค่าความเป็นจริงมาก และ ถ้า ค่าความเหมือนแบบ ภาพตัวอย่าง มากกว่า Threshold จะถือว่า P-expert จุดนี้คือ Core ซึ่ง P-expert จะทำการสร้าง positive example การ Update model detection และ ensemble classifier

- N-expert

- มีหน้าที่สร้าง Negative examples ซึ่ง จะถูกเพิ่มพร้อมกับ P-expert



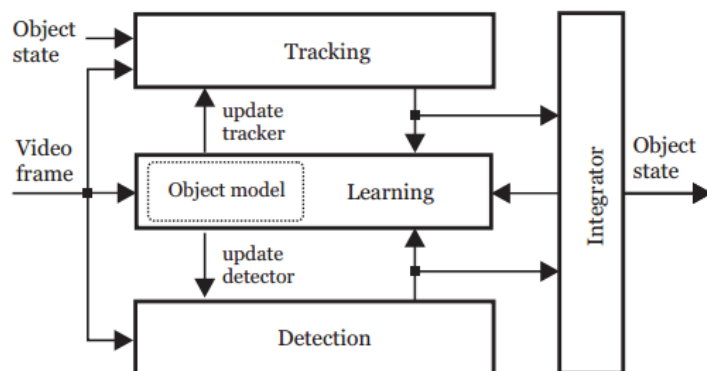
รูปที่ 3-1 แสดงการ Update การเรียนรู้ของวิธี PN-learning โดยจุดสีแดงคือ Positive example จุดสีดำคือ Negative example

Summary

ซึ่งภาพรวมทั้งหมดของ TLD จะเป็นดังนี้ คือ Tracker และ Detector จะส่งข้อมูลให้ Learning มีข้อมูลมากขึ้น และ Learning จะ Update ตัว Classifier ให้ Detection และ Update การ Track ไว้ใช้สำหรับ Frame ถัดไป ซึ่ง Detector กับ Tracker จะทำงานสอดคล้องกัน ถ้า Tracker ติดตามวัตถุผิดพลาด เช่น

ตัวอย่างที่ 1 จุด Median วัตถุในภาพภาพ(T-1) ห่างจาก จุด Median วัตถุในภาพภาพ(T) เกิน Threshold ที่กำหนด Detector ก็จะมาทำหน้าที่ช่วยตัว Tracker และ ส่งกรอบที่ Track ผิดพลาด ไป Update Training Set ได้

ตัวอย่างที่ 2 Detector ตรวจพบ รูปที่คล้ายภาพตัวอย่างเริ่มต้น มากกว่า 1 กรอบ ก็ให้อ้างอิงกับ ตำแหน่ง Tracker ว่า ถ้ากรอบใดใกล้กับ กรอบของ Tracker จะถือว่าเป็น Positive example และอื่นๆที่ไม่อยู่ใกล้ Tracker ก็จะนำไปเป็น Negative example และส่ง feature เหล่านี้ ให้แก่ PN-Learning ในกระบวนการ Learning ต่อไป



รูปที่ 4-1 แสดงขั้นตอนการทำงานของ TLD