궤적에서 객체의 중심점과 변위 추정을 통한 영상 객체 탐지 연구

손하영, 이유진, 최계원*

성균관대학교

shyds628@g.skku.edu, leeyoujin225@g.skku.edu,*kaewonchoi@skku.edu

A Study on Video Obejct Detection by Estimation of Center and Movement of the object on the trajectory

Ha Young Son, You Jin Lee, Kae Won Choi*

Sungkyunkwan Univ.

요 약

본 논문에서는 영상 속에서 원거리에 위치하거나 해상도에 비해 매우 작아 형상이 명확하지 않아 탐지가 잘 되지 않는 객체를 탐지하기 위해, Anchor-Free 방식으로 Object Detection을 수행하는 CenterNet을 적용하고, 동시에 객체의 움직임을 추정하는 네트워크인 STTNet(Short Term Tracklet Network)을 만들었다. 이 네트워크에 VisDrone-MOT 데이터셋을 훈련시켜 영상의 일정 시간 간격의 프레임 사이에서 객체의 위치와 변위를 추정하여 영상 객체 탐지를 수행하였다.

Ⅰ. 서 론

최근 다양한 교통 상황과 물류 환경에서, Object Detection을 통한 Visual Surveillance의 적용이 연구되고 있다. 이 중 항만 환경은 안전 사고 방지와 보안을 위해 객체마다 탐지가 필요하지만, 크기가 제각각인 다양한 장애물이 존재하거나 움직이며 사람의 눈으로 파악하기 어려운 넓은 구역을 한꺼번에 감시해야 하는 환경이라는 문제점이 있다.

특히 이러한 환경의 경우 한 번에 넓은 환경을 감시하기 위해 낮은 지점이 아니라 높은 지점에 폐쇄회로 카메라가 설치된다는 특수성이 있다. 따라서 영상 속의 객체는 카메라로부터 상당히 원거리에 위치하며, 이로 인해 객체의 형상이 명확하지 않고 해상도에 비해 매우 작다는 문제점이 발생한다. 이러한 객체를 잘 식별하기 위해서는 전후 프레임을 참고하여 추가적인 시간적 정보를 학습에 활용할 필요가 있다.

이렇게 추가적인 시간적 정보를 학습에 활용하는 방법에는 정지된 단일 이미지 한 장이 아니라 영상을 입력으로 Object Detection을 수행하는 방법이 있다. Video Object Detection과 Object Tracking이 그 예이다.

영상을 입력으로 사용할 경우 단일 정지 이미지에서는 발생하지 않고 영상에서만 발생하는 문제가 있는데, 객체의 움직임에 따라 모션 블러가 발생하거나 포커스를 놓치기도 한다는 점이다. 이를 해결하기 위해 CNN을 거친 Feature 외에도 다양한 Feature을 사용하기도 한다. Optical Flow와 유사하게, 객체의 움직임을 나타내는 Vector를 Flow Network를 통해 Feature로 만든 Flow Field[3]를 인접한 프레임에서 모아 Video Object Detection에 사용한 FGFA[4]가 그 예이다.

영상의 각 프레임마다 Object Detection을 수행할 때 대부분 Faster-RCNN[5], YOLO-v3[6]와 같은 Anchor-Based 방식을 사용하지만, 특수한 상황에 맞추어 Anchor-Free 방식을 사용하기도 한다. 대표적인 Anchor-Free 방식인 CenterNet[1]은 Anchor 없이 각 객체들의 중심점을 추정하여 객체들을 하나의 점으로 표현하는 네트워크이다. 이 CenterNet을 Object Tracking에 적용한 사례는 FairMOT[2]가 있다.

본 논문에서는 영상의 작은 크기의 원거리 객체를 탐지하는 것이 목표로, 시간적 정보를 얻기 위해 여러 장의 이미지를 쌓아 훈련에 사용했다. 또한, 바운딩 박스를 찾는 것이 중점이 아니며 주로 작은 객체를 탐지해야 하기 때문에 높은 해상도를 가진 단일 레이어를 사용하는 것이 유리하다는 점에서 Anchor-Free 방식인 CenterNet을 기반으로 객체를 탐지했다. 또한, 갑자기 방향을 바꾸기도 하는 객체의 움직임을 좀 더 잘 파악하기 위해 객체의 변위 정보도 함께 예측했다.

Ⅱ. 본론

그림 1. VisDrone 데이터셋의 이미지와 그 라벨링

훈련 데이터셋은 VisDrone-MOT2019[7]을 사용했으며, 이 데이터셋은 Multi Object Tracking을 위해 만들어진 데이터셋으로, 드론이 고점에서 천천히 비행하면서 지상을 촬영한 영상에 객체의 클래스, 위치, ID를 프레임마다 라벨링하였다. 그림 1처럼 차량이 많은 교차로, 사람이 많이 지나다니는 공원과 학교 등 여러 환경을 높은 지점의 원거리에서 다양한 시간대에 촬영한 영상들이 우리의 목표와 잘 부합하여 훈련에 사용하였다. 훈련에는 56개의 시퀀스(총 24,201 프레임)를 사용했고, 테스트에는 17개의 시퀀스(총 6,618 프레임)을 사용했다.

입력으로는 그림 2와 같이 시퀀스 상의 연속된 이미지를 8장 단위로 쌓아 하나의 샘플을 만든 뒤, 이 샘플들을 모아 배치를 만들어 훈련에 사용했다. 샘플을 쌓는 단위가 작아질수록 샘플 내부의 프레임끼리의 시간적 인접성이 커지고, 커질수록 그 반대가 된다. 각 샘플의 처음 프레임과 마지막 프레임에서 객체의 중심점과 변위를 목표로 훈련했다.

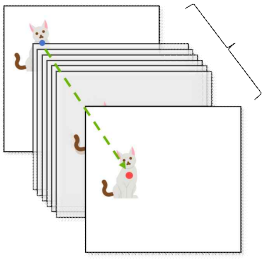

그림 2. 입력 샘플의 구조. 파란색 점은 시작 이미지 물체의 Center, 빨간색 점은 마지막 이미지 물체의 Center이다.

그림 3. 출력 이미지 샘플. 빨간색은 시작 이미지의 확률 분포 히트맵, 파란색은 마지막 이미지의 확률 분포, 초록색은 움직임을 나타내는 화살표.

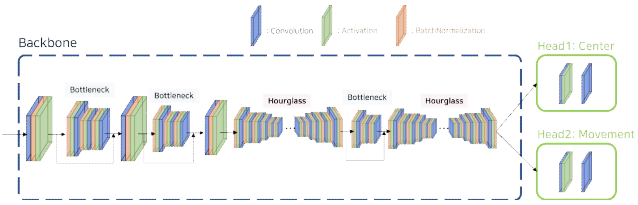

그림 4. STTNet의 훈련 네트워크 구조. 여러 개의 Bottleneck 레이어와 Hourglass 구조 레이어로 이루어진 백본 네트워크와 두 개의 탐지 Head로 이루어져 있다.

훈련한 백본 네트워크는 그림 4처럼 고해상도의 정보를 유지하기 위해 Skip Connection이 존재하는 Bottleneck 구조 레이어와 Hourglass 구조 레이어 여러 층으로 이루어진 깊은 네트워크로, 이를 거친 후 두 개의 Detection Head를 통해 출력을 만들었다. Hourglass 레이어는 Bottleneck 레이어를 재귀적으로 쌓아 만들었다. Head 중 하나는 각 클래스마다 객체의 중심점이 있을 확률의, 즉 Class score의 분포도를 찾는 Head이며, 다른 하나는 각 클래스마다 객체의 변위를 찾는 Head이다.

각 Head마다 훈련에 사용하는 Loss는 다음과 같다. 중심점을 찾는 Loss인 L_c는 CenterNet에서 사용된 Loss와 같이 Focal Loss[8]를 사용하였다.

$$L_c = -\frac{1}{N}\sum \begin{cases} (1-\hat{Y})^{\alpha}\log(\hat{Y}) & \text{if } Y=1 \\ (1-Y)^{\beta}(\hat{Y})^{\alpha}\log(1-\hat{Y}) & otherwise \end{cases} \quad (1)$$

$\hat{Y}$는 Prediction, Y는 Target, N은 이미지의 중심점 개수, α와 β는 Focal Loss의 하이퍼 파라미터로 각각 2와 4를 사용했다. 객체의 변위를 찾는 Loss인 L_m은 smooth L1 loss를 사용하여 훈련했다.

그림 3은 이 훈련을 통해 얻을 수 있는 출력을 보기 쉽도록 샘플의 첫 번째 시작 이미지와 마지막 이미지를 겹쳐 만든 이미지 샘플이다. 출력은 클래스마다 입력 샘플 첫 번째 시작 이미지의 객체 중심점의 확률 분포, 마지막 이미지의 객체 중심점의 확률 분포, 객체의 X축 변위, 객체의 Y축 변위로 이루어져 있다.

Ⅲ. 결론

클래스는 크게 사람, 이륜차(자전거, 오토바이), 사륜차(일반 차량, 트럭, 버스 등)의 총 세 가지로 나누어 학습하였다.

학습 결과, 8장의 이미지를 쌓아 만든 샘플 내에서 객체의 움직임을 따라 변위의 방향이 달라짐을 알 수 있었고, 객체가 위치한 곳에 객체가 위치할 확률이 문턱값보다 높음을 알려주는 키포인트를 볼 수 있었다. 이를 통해 영상 내의 일정 기간 프레임 내에서 객체의 움직임과 위치를 추정할 수 있음을 알 수 있었다.

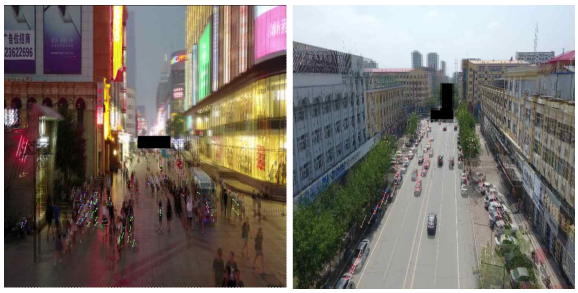

그림 5. 각 클래스 별로 출력을 보기 쉽도록 만든 이미지. 빨간색은 사륜차를, 파란색은 이륜차를, 초록색은 사람의 위치와 변위를 나타낸다.

그러나 샘플을 쌓는 프레임 수가 늘어나 샘플의 시간적 거리가 증가할수록, 카메라가 급격히 흔들리거나 크기가 작은 물체를 탐지하지 못하는 모습을 보였으며 다른 클래스로 추정하는 문제가 발생했다. 따라서 다른 프레임의 시간적 정보를 융합하고 보정하는 Feature Aggregation 방법을 통해 영상 객체 탐지를 개선할 필요가 있다.

ACKNOWLEDGMENT

이 논문은 2021년도 정부(해양수산부)의 재원으로 해양수산과학기술진흥원의 지원("IoT기반 지능형 항만 물류 기술 개발"사업의 "스마트 항만 IoT 융합•운영 기술 개발")을 받아 수행된 연구임.

참 고 문 헌

[1] Zhou, Xingyi, Dequan Wang, and Philipp Krähenbühl. "Objects as points." arXiv preprint arXiv:1904.07850 (2019).

[2] Zhang, Yifu, et al. "Fairmot: On the fairness of detection and re-identification in multiple object tracking." arXiv preprint arXiv:2004.01888 (2020).

[3] Zhu, Xizhou, et al. "Deep feature flow for video recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. (2017).

[4] Zhu, Xizhou, et al. "Flow-guided feature aggregation for video object detection." Proceedings of the IEEE International Conference on Computer Vision. (2017).

[5] Ren, Shaoqing, et al. "Faster R-CNN: towards real-time object detection with region proposal networks." IEEE transactions on pattern analysis and machine intelligence 39.6 (2016): 1137-1149.

[6] Redmon, Joseph, and Ali Farhadi. "Yolov3: An incremental improvement." arXiv preprint arXiv:1804.02767 (2018).

[7] Wen, Longyin, et al. "Visdrone-mot2019: The vision meets drone multiple object tracking challenge results." Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops. (2019).

[8] Lin, Tsung-Yi, et al. "Focal loss for dense object detection." Proceedings of the IEEE international conference on computer vision. (2017).