JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 29 No. 2, pp. 79-86, February 2024
https://doi.org/10.9708/jksci.2024.29.02.079

Utilizing Context of Object Regions for Robust Visual Tracking

Janghoon Choi*

*Assistant Professor, Graduate School of Data Science, Kyungpook National University, Daegu, Korea

[Abstract]

In this paper, a novel visual tracking method which can utilize the context of object regions is presented. Conventional methods have the inherent problem of treating all candidate regions independently, where the tracker could not successfully discriminate regions with similar appearances. This was due to lack of contextual modeling in a given scene, where all candidate object regions should be taken into consideration when choosing a single region. The goal of the proposed method is to encourage feature exchange between candidate regions to improve the discriminability between similar regions. It improves upon conventional methods that only consider a single region, and is implemented by employing the MLP-Mixer model for enhanced feature exchange between regions. By implementing channel-wise, inter-region interaction operation between candidate features, contextual information of regions can be embedded into the individual feature representations. To evaluate the performance of the proposed tracker, the large-scale LaSOT dataset is used, and the experimental results show a competitive AUC performance of 0.560 while running at a real-time speed of 65 fps.

▸ **Key words**: Visual tracking, Object tracking, Video understanding, Context modeling, Mixer network

[요 약]

본 논문에서는 단일 영역 정보만을 활용하는 기존의 방법론을 개선하기 위해, 물체의 맥락영역에 대한 정보를 함께 물체 추적에 활용하는 새로운 기법을 제시한다. 기존의 방법론들은 모든 후보 영역들을 독립적으로 처리하는 구조로, 비슷한 외양의 영역들이 등장하는 경우 이를 성공적으로 구분하지 못하는 문제점을 보여주었다. 이는 주어진 장면 내에 등장하는 모든 후보 물체 영역들에 대한 맥락 정보를 고려하지 못하여 생기는 문제이다. 제안하는 방법론에서는 비슷한 외양의 영역들 간의 특징점 정보 교환을 보조하고 이들 간의 구별성을 높이는 것을 목표로 하였다. 이를 구현하기 위해 MLP-믹서 (MLP-Mixer) 모델을 활용하여 맥락영역 간의 정보 교환을 모델링하는 모듈을 제시하였다. 이를 통해 구현된 특징점 채널별, 영역간의 상호작용 연산은 영역의 개별 특징점 표현에 대해 장면 맥락 정보가 내장될 수 있도록 보조한다. 제안한 방법론의 성능을 평가하기 위해 대규모 물체 추적 데이터셋인 LaSOT을 사용하였고, 성능 평가 결과 제안한 알고리즘은 AUC 지표 기준 0.560의 높은 성능과 함께 65fps의 실시간 속도로 동작함을 확인하였다.

▸ **주제어**: 물체 추적, 객체 추적, 비디오 이해, 맥락 모델링, 믹서 네트워크

• First Author: Janghoon Choi, Corresponding Author: Janghoon Choi
*Janghoon Choi (jhchoi09@knu.ac.kr), Graduate School of Data Science, Kyungpook National University
• Received: 2024. 01. 26, Revised: 2024. 02. 06, Accepted: 2024. 02. 14.

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

물체 추적 (visual tracking) 문제는 컴퓨터 비전 (computer vision) 연구의 대표적인 하위 연구주제로, 무인 감시, 자율 주행, 로보틱스 등의 여러 응용 분야들을 가지는 중요한 주제이다. 물체 추적 문제는 비디오의 첫 프레임 이미지와 함께 초기 물체 상태 (initial target state)가 경계 상자 (bounding box)와 같은 형태로 주어진다고 가정할 때, 순차적으로 주어지는 비디오 프레임들에 대해 물체의 영역을 성공적으로 검출하는 것을 목표로 한다. 물체 추적을 수행하는 동안, 목표 물체의 변형, 비슷한 물체의 등장, 조명 변화, 화질 저하 등 여러 도전적인 상황들이 발생할 수 있으며, 좋은 물체 추적 알고리즘은 이러한 상황들에서도 성공적으로 목표 물체를 정확히 검출하여야 한다.

최근 딥러닝 (deep learning) 및 합성 곱 신경망 (CNN; convolutional neural networks) 모델의 발전 및 컴퓨터 비전 분야에의 다양한 응용 사례들과 함께, 물체 추적 알고리즘들의 성능 또한 큰 향상을 이루었다[1]. 그 중에서도 샴 네트워크 (Siamese network) 구조를 활용한 SiamFC[2] 방법론은 높은 성능과 함께 빠른 동작 속도로 주목을 받았으며, 많은 후속 연구들이 이루어졌다. 객체 패치 (target patch)와 탐색 패치 (search patch) 이미지들을 동시에 입력으로 받아 특징점을 추출 후, 상호 상관 (cross-correlation) 연산을 기반으로 탐색 패치 안에서 객체 패치의 위치를 검출하는 SiamFC 구조를 기반으로, 장기 추적 (long-term tracking)이 가능하도록 발전시킨 GlobalTrack[3] 방법론은 기존 물체 검출 기법인 Faster R-CNN[4]의 후보 영역 제안 (region proposal) 단계 및 영역 분류 (region classification) 단계의 2단계 구조를 차용하였다.

하지만 이와 같은 기존 SiamFC 기반의 방법론들의 경우, 제안된 모든 후보 영역들에 대해 독립적으로 목표 물체와의 유사도를 산출한다는 한계점이 있어, 유사한 외양의 물체가 등장하는 경우 산출된 유사도 간에 차이가 매우 적어 추적기가 물체영역 간에 혼동을 일으키기 쉽다는 문제점이 있다. 따라서 본 연구에서는 이러한 문제점을 해결하기 위해 여러 개의 맥락영역 (context region)들에 대해 구별성 (discriminability)을 최대화하고 유사도를 분리할 수 있도록 특징점을 학습하는 통합된 모델링 방법론을 제시하며, 이는 MLP-믹서 (MLP-Mixer)[5] 네트워크를 사용하여 구현된다.

제안하는 방법론에서는 GlobalTrack의 후보 영역 제안 네트워크의 출력 영역들을 기반으로 하여, MLP-믹서 모델을 활용하여 구성된 맥락정보 활용 모듈을 통해 해당 장면의 맥락에 특화된 특징점 표현 (feature representation)을 학습하는 것을 목표로 하였다. 제안하는 방법론에 대한 성능 평가를 위해서는 대규모 (large-scale), 장기적 (long-term) 물체추적 데이터셋 중 하나인 LaSOT[6]을 활용하였으며, 제안하는 방법론이 높은 성능과 함께 실시간 속도 (65 fps)로 동작함을 확인하였다. 제안하는 물체 추적 방법론에 대한 개관은 아래 Fig. 1과 같다.

본 논문은 다음과 같이 구성된다. 2장의 관련 연구에서는 딥러닝 기반의 물체 추적 알고리즘들에 대한 기존 방법론들을 소개하고, 각 알고리즘이 어떤 방식으로 물체 추적을 수행하는지에 대해 분석하고 그 한계점을 서술한다. 3장의 제안 방법에서는 맥락 정보를 물체 추적 알고리즘에서 사용하기 위해 제안하는 방법론을 소개하며, 4장에서는 제안된 물체 추적 방법론을 실험적으로 검증하고 타 물체 추적 방법론과의 비교실험을 수행한다. 마지막으로 5장에서는 제안한 방법론에 대한 정리 및 한계점과 함께 향후의 연구 방향성에 대해 제안하도록 한다.

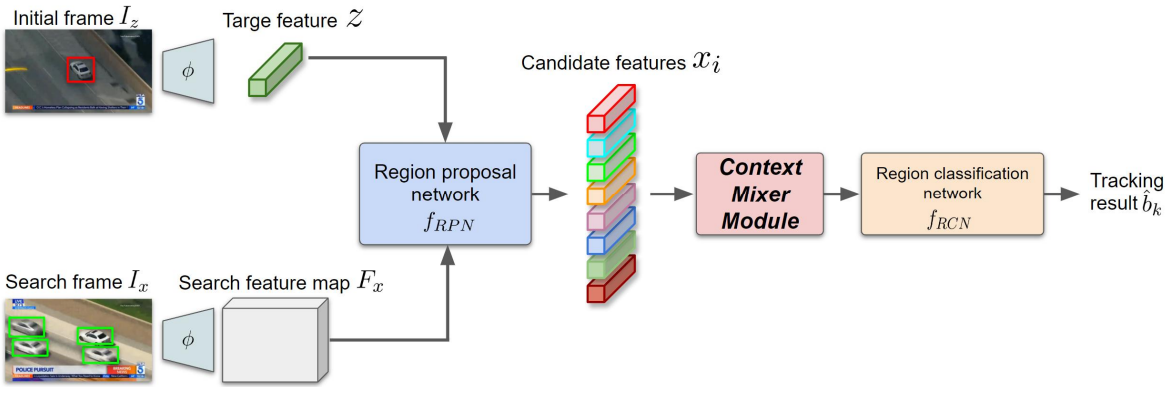

Fig. 1. Overview for the proposed visual tracking framework

II. Related Work

물체 추적 문제는 일반적으로 검출을 통한 추적 (tracking-by-detection) 기반의 접근법을 통해 방법론들이 제안되어 왔으며, 해당 방법론들의 경우 탐색 영역 내에서 목표 물체를 검출하기 위해 이진 분류 (binary classification) 문제로의 모델링을 통해 물체 추적 문제를 해결하고자 하였다. 성공적으로 학습이 완료된 이진 분류기 모델은 탐색 영역에서 목표 물체가 있는 부분을 양 (positive)의 레이블을 가지도록, 배경 영역이 있는 부분을 음 (negative)의 레이블을 가지도록 분류하며, 양의 레이블을 가지면서 가장 신뢰도 점수 (confidence score)가 높은 영역을 목표 물체 영역으로 선택하게 된다.

딥러닝 및 신경망 모델 기반의 강력한 특징점 표현 (feature representation)들을 활용할 수 있게 되면서 물체 추적 방법론들 또한 큰 성능 향상을 이루었다. 대표적인 딥러닝 기반의 방법론으로는 MDNet[1], SiamFC[2], SiamRPN[9]과 같은 방법론이 있다. MDNet[1]의 경우 VGG[10] 네트워크 기반의 특징점 추출 네트워크를 사용하여 여러 도메인의 물체 및 배경 정보가 포함된 학습 데이터셋에 대해 사전학습을 진행한다. 그 결과 여러 도메인에서 공통적으로 사용할 수 있는 특징점을 추출하는 네트워크가 학습되며, 실제 물체 추적 상황에서는 네트워크의 특징점 추출 부분을 사용하고 물체 추적 상황에 따라 경사하강법 (gradient descent)을 사용하여 네트워크를 업데이트한다. MDNet의 경우 기존 CNN 특징점을 사용한 상관 필터 (correlation filter) 기반의 물체 추적 알고리즘 대비 높은 성능을 달성하였으나, 물체의 외양 모델의 업데이트에 소요되는 시간 및 많은 계산량에서 오는 한계점이 있어 실시간 (real-time)처리는 어려운 문제점이 있었다.

이후 SiamFC[2]에서는 기존의 상관 필터 기반 물체 추적 알고리즘이 가지는 간단한 구조의 장점과, CNN 기반의 강력한 특징점 추출 모델의 장점을 융합함으로써 MDNet이 가지는 문제점을 개선하여 많은 주목을 받았다. SiamFC는 파라미터의 숫자 및 계산량의 증가 요인인 완전연결층 (fully-connected layer)을 배제하고 전적으로 합성곱 연산 (fully convolutional)만을 사용한 모델 구조를 차용하였다. 또한 샴 (Siamese) 구조를 통해 객체 패치 (target patch)와 탐색 패치 (search patch) 이미지들을 동시에 입력으로 받아 특징점을 추출하고, 객체 패치에서 얻은 특징점 맵을 필터의 형태로 탐색 패치의 특징점 맵에 적용하여 상관 필터 기반의 구조를 차용하였다. 이후에는 가장 유사도 (similarity score)가 높은 부분을 선택하고 이를 목표 물체의 위치로 출력한다. 실시간 속도를 달성하기 위해 경사하강법 기반의 모델 갱신은 수행하지 않았다.

Table 1. Comparison of target localization schemes

	Region Proposal	Region Classification
GlobalTrack	RPN[4]	RCNN[4]
ATOM	IoUNet (ECCV'18)	-
SiamRPN++	RPN[4]	-
SiamFC	-	-
Proposed Method	RPN[4]	***Context Utilization +*** RCNN[4]

이후에는 SiamFC를 기반으로 한 다양한 물체 추적 방법론들이 많이 제안되었으며, 대표적으로는 SiamRPN[9]과 같이 물체 검출 (object detection) 분야에서 많이 사용되는 후보 영역 제안 (region proposal)[4] 네트워크를 차용하여 물체 영역을 더욱 정확하게 검출할 수 있도록 하는 방법론이 고안되었다. 또한 트랜스포머[13] 기반의 모델 또한 사용되었다[14,15]. 하지만 기존의 샴 구조 및 상호 상관 연산 기반의 물체 추적 알고리즘의 경우, 현재 프레임에서 물체를 검출하고자 할 때 직전 프레임에서 검출된 영역을 기반으로 탐색 패치를 추출하는 한계점이 있다. 따라서 추적에 한 번이라도 실패하는 경우 추적기가 납치 (hijacking) 및 표류 (drifting) 현상으로 인해 올바른 위치로 회복하지 못하는 문제가 발생하여 장기 추적 문제에는 부적합하였다. 이를 해결하기 위해 GlobalTrack[3]에서는 현재 프레임에서 탐색 패치를 추출하는 대신 프레임 전체에 대해 전반적으로 탐색을 진행하고, 후보 영역 제안 단계 및 영역 분류 단계의 2단계 구조[4]로 추적기를 모델링하였으며, 장기 추적 문제에서 향상된 성능을 얻었다.

하지만 GlobalTrack 또한 SiamFC와 같이 모든 후보 영역들에 대해 독립적으로 유사도를 산출한다는 한계점이 있으며, 비슷한 외양의 물체들이 한 장면에 등장하는 경우 유사도 간에 차이가 매우 적어 추적기가 물체영역 간에 혼동을 일으키는 납치 (hijacking) 현상에 취약하다는 문제점이 잔존한다. 따라서 본 연구에서는 이러한 문제점을 해결하기 위해 제안된 후보 영역들 전체를 맥락영역 (context region)으로 취급하여, 이들 간에 구별성 (discriminability)을 최대화하고 유사도의 차이를 높여 올바른 물체에 대해 최대의 유사도를 출력할 수 있는 특징점을 학습하는 모델링 방법론을 제시한다 (. 이를 위해서는 최근에 주목을 받은 MLP-믹서 (MLP-Mixer)[5] 네트워크를 사용하여 후보 영역에서 추출된 특징점들의 채널 간 상호작용 (inter-channel interaction)을 모델링함으로써 구현하였다.

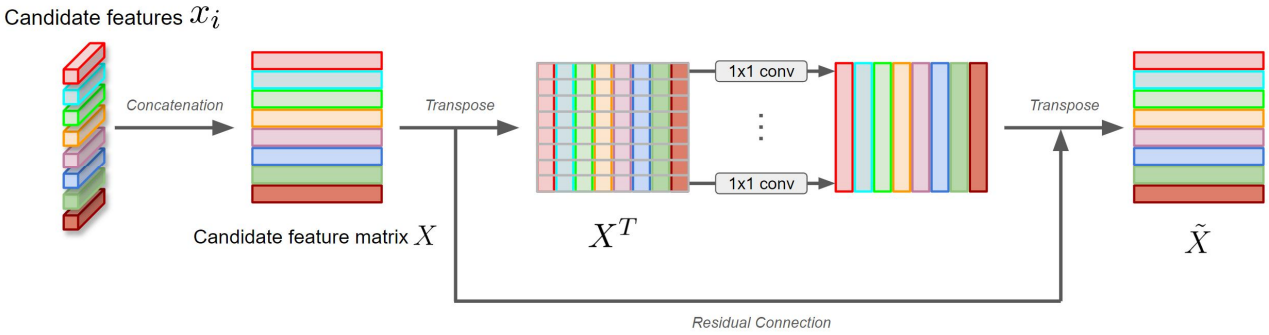

Fig. 2. Detailed view of the context utilization module

III. Proposed Method

제안하는 물체 추적 모델은 비디오의 첫 프레임 I_z 및 물체의 초기 상자 좌표 b_0와 함께, 물체를 찾고자 하는 탐색 프레임 I_x를 입력으로 받는다. 각 프레임에서는 특징점 추출 네트워크 ϕ를 통해 $h \times w \times c$ 크기의 특징점 맵 (feature map) $F_z = \phi(I_z), F_x = \phi(I_x)$를 추출하며, ϕ는 사전 학습된 ResNet-18[7]을 사용하였다. 이후의 물체 검출 과정은 크게 두 단계로 진행된다. 첫 번째는 영역 제안 네트워크 (region proposal network) f_{RPN}을 통해 여러 개의 후보 영역 (candidate region)을 얻는 단계이며, 두 번째 단계는 얻어진 후보 영역의 특징점에 대해 분류 네트워크 f_{RCN}을 활용하여 가장 유사한 후보 영역을 추적 모델의 최종 출력으로 얻는 단계이다. 각 단계별 동작 세부사항과 학습 방법은 다음과 같다.

3.1. Region Proposal Stage

첫 단계인 후보 영역 제안 단계의 경우, 먼저 ROIAlign[4] 방법을 사용하여 첫 프레임 특징점 맵 F_z에서 공간적 풀링 (spatial pooling)된 객체의 특징점 표현 z를 추출한다. 풀링 크기는 $s = 5$로 $s \times s \times c$의 형태로 얻어진 z를 사용하여 탐색 프레임 특징점 맵 F_x와의 상호 상관 (cross-correlation) 연산을 $\widehat{F_x} = F_x * z$와 같이 수행하게 된다. 이와 같이 처리된 특징점 맵 $\widehat{F_x}$를 후보 영역 제안 헤드 (head) 네트워크에 입력으로 전달하고, 출력으로는 영역 제안을 위한 이진 분류 맵 (binary class map) p (크기 $h \times w \times 2$)와 박스 회귀 맵 (bounding box regression map) q (크기 $h \times w \times 4$)를 $p, q = f_{RPN}(\widehat{F_x})$와 같이 얻는다. p, q는 모두 공간적으로 $h \times w$ 크기로, 각각 2,4의 채널 크기를 가진다. 이후, p에서 채널 방향의 softmax 연산을 통해 각 위치에 대해 유사도 점수를 계산하고, 해당 위치들에 해당하는 q의 값을 사용하여 후보 영역의 아래, 위, 왼쪽, 오른쪽 경계에 대한 박스 좌표정보를 얻은 후, NMS 연산을 통해 N개의 유력한 후보 영역 $\{b_i | i = 1, ..., N\}$을 추려내 선발한다. 후보 영역 제안 헤드 (head) 네트워크의 학습을 위한 손실 함수는 아래와 같으며,

$$L_{RPN} = L_{cls}(p, p^*) + \lambda L_{reg}(q, q^*) \quad (1)$$

위의 식에서 p^*, q^*는 각각 모든 위치별 목표 레이블 맵을 나타내며, λ는 두 항의 균형을 위한 변수이다. L_{cls}는 분류를 위한 Focal loss를, L_{reg}는 IoU (intersection-over-union) loss를 사용하였으며 L_{reg}의 경우 $p^* > 0$인 위치, 즉 목표 물체의 상자 영역 내부에 위치한 영역들에 대해서만 양수의 손실 함수 값을 출력하고 나머지 배경 영역에 대해서는 0을 출력한다.

3.2. Region Classification Stage

두 번째 단계인 후보 영역 분류 단계의 경우, 우선 앞 단계에서 얻어진 후보 영역들 b_i들의 좌표 정보로 탐색 프레임 특징점 맵 F_x에 대해 $s = 1$로 ROIAlign 연산을 수행하여 c차원의 특징 벡터 형태의 x'_i를 N개 ($N = 32$ 사용) 추출하고, 같은 과정을 통해 첫 프레임 특징점 맵 F_z에서 얻어진 물체의 특징 벡터 z'와의 벡터 내적 (inner product) 연산을 $x_i = x'_i \cdot z'$와 같이 수행한다. 다음에는 후보 영역들 간의 특징 정보를 공유하고, 맥락정보를 반영하기 위한 모듈 (context mixer module)을 위해 $X = [x_1; ...; x_N]$과 같이 행벡터 x_i들에 대한 연쇄연산 (concatenation)을 통해 하나의 $N \times c$ 크기 텐서를 만들고, 전치 연산을 통해 $c \times N$ 크기의 $X^T = [y_1; ...; y_c]$를 얻는다. 이후 c개의 행 벡터 y_i들은 독립적으로 MLP 네트워크 $\tilde{y_i} = g_{MLP}(y_i)$를 통과시킨 후, 얻어진 텐서 $\tilde{X}^T = [\tilde{y_1}; ...; \tilde{y_c}]$에 대해 다시 전치 연산을 수행하여 맥락

정보가 반영된 $\widetilde{X}$를 얻는다. 이 과정을 $\widetilde{X}= g_{MIX_m}(X)$와 같이 표현할 수 있으며, 이러한 MLP-mixer 연산을 $m=3$번 반복한다. 맥락정보 활용 모듈의 상세는 Fig. 2와 같다. 이후에는 위의 연산을 통해 맥락 정보가 반영된 특징점 $\widetilde{x_i}$들을 사용하여 영역 분류를 수행하며, 각 $\widetilde{x_i}$들은 후보 영역 분류 헤드 (head) 네트워크의 입력으로 전달된다. 출력은 각 영역에 대해 이진 분류 로짓 (logit)값 u와 박스 좌표 개선값 v를 $u,v = f_{RCN}(\widetilde{x_i})$와 같이 얻는다. 여기서 u는 2차원의 크기를, v는 4차원의 크기를 가지는 벡터이다. 마지막으로, u에서 softmax 연산을 통해 가장 큰 분류 점수 u_k를 가지는 k번째 영역 $\widehat{b_k}= b_k + v_k$를 최종 물체추적의 결과로 얻는다. 후보 영역 분류 헤드 (head) 네트워크의 학습을 위한 손실 함수는 아래와 같다.

$$L_{RCN} = L_{cls}(u,u^*) + \lambda L'_{reg}(v,v^*) \quad (2)$$

위의 식에서 u^*, v^*는 각각 목표 분류 레이블과 목표 박스 좌표를 나타내며, λ는 두 항의 균형을 위한 변수이다. L_{cls}는 Focal loss를, L'_{reg}는 L1 loss를 사용하였으며, L_{cls}의 경우 후보 영역과 목표 영역의 IoU 값이 $\tau_p = 0.5$를 넘는 경우에만 양의 레이블을, $\tau_n = 0.4$이하인 경우는 음의 레이블을 학습하도록 하였다. 또한 L'_{reg}의 경우에도 마찬가지로 후보 영역과 목표 영역의 IoU 값에 따라 τ_p를 초과하는 경우 양의 손실함수 값을, τ_n이하인 경우는 0을 출력하여 학습하지 않도록 한다.

3.3. Training and Architecture Details

앞에서 서술된 물체 추적 모델을 학습시키기 위해 아래와 같이 통합된 손실 함수를 정의하였으며,

$$L_T = L_{RPN} + \lambda_1 L_{RCN} \quad (3)$$

위의 식에서 L_{RPN}과 L_{RCN}의 경우 식 (1) 및 식 (2)에서 정의한 손실 함수를 사용하였다. 손실 함수의 최소화를 위해서는 경사하강법 기반의 AdamW[8] 최적화 알고리즘을 사용하였다. 학습률 (learning rate)은 10^{-4}을 적용하였으며, 과적합 방지를 위한 가중치 감쇄 (weight decay) 변수는 10^{-5}로 설정하였다. 학습을 위해 사용한 데이터셋은 LaSOT의 training set이며, 매 학습연산 (iteration)마다 임의의 비디오를 선택 후 해당 비디오의 프레임 두 장을 선택하여 활용하였고, 총 2×10^6회의 학습연산을 수행하면서 중간 지점에 한 번의 학습률 감쇄 (learning rate decay)를 0.5의 가중치로 수행하였다.

Table 2. Detailed architecture of the proposed tracking model

Feature Extraction Stage

Input: I_z, I_x $(3\times480\times720)$	
Feature extractor: ResNet-18 [7]	
CONV: $256\times512\times1\times1$ Stride:1, Padding:0	CONV: $256\times512\times1\times1$ Stride:1, Padding:0
$F_z = \phi(I_z)$ $(256\times30\times45)$	$F_x = \phi(I_x)$ $(256\times30\times45)$

Region Proposal Stage

ROIAlign: z $(256\times5\times5)$		$F_x = \phi(I_x)$ $(256\times30\times45)$	
Cross-Correlation $\widehat{F_x} = F_x * z$, Stride:1, Padding:2			
$\widehat{F_x}$ $(256\times30\times45)$			
CONV: $256\times256\times3\times3$ Stride:1, Padding:1	$\times2$	CONV: $256\times256\times3\times3$ Stride:1, Padding:1	$\times2$
GroupNorm $G=16$		GroupNorm $G=16$	
CONV: $2\times256\times1\times1$ Stride:1, Padding:0		CONV: $4\times256\times1\times1$ Stride:1, Padding:0	
p $(2\times30\times45)$		q $(4\times30\times45)$	

Region Classification Stage

ROIAlign: z $(256\times1\times1)$		ROIAlign: x'_i $(256\times1\times1)$	
Inner Product $x_i = x'_i \cdot z'$ → Concat $X=[x_1;..;x_N]$			
X $(N\times256)$			
Transpose: X^T $(256\times N)$			
MLP: $N\times N$ with Residual Connection			$\times3$
Transpose $\widetilde{X}$ $(N\times256)$ → Row vectors $\widetilde{x_i}$ $(256\times1\times1)$			
CONV: $256\times256\times1\times1$ Stride:1, Padding:0	$\times2$	CONV: $256\times256\times1\times1$ Stride:1, Padding:0	$\times2$
GroupNorm $G=16$		GroupNorm $G=16$	
CONV: $2\times256\times1\times1$ Stride:1, Padding:0		CONV: $4\times256\times1\times1$ Stride:1, Padding:0	
u $(2\times1\times1)$		v $(4\times1\times1)$	

특징점 추출을 위한 네트워크 ϕ는 $3\times480\times720$ 크기의 RGB 이미지를 입력으로 받으며, 모든 입력 이미지에 대해 가장 긴 변의 크기에 맞춰 리사이징 (resizing) 작업을 수행하였다. 출력으로 나오는 특징점 맵 F_z, F_x는 $512\times30\times45$의 크기를 가진다. 후보 영역 제안 및 분류를 위한 헤드 네트워크들의 경우 3개의 합성곱 층 (convolution layer)로 구성되어 있으며, 채널의 크기는 256으로 설정하였다. 또한 ReLU 활성함수 (activation function)와 그룹 정규화 (group normalization) 연산을 사이에 삽입하였다. 네트워크 아키텍처에 대한 상세 구조는 Table 2와 같다. 크게 세 부분으로 나뉘며, 음영 표시된 부분은 중간 출력으로 얻어지는 특징점 맵의 크기이고 이외의 부분은 네트워크의 개별 모듈을 나타낸다.

Table 3. Comparison of performance metrics on the LaSOT test set

	AUC	Precision	Normalized Precision
Proposed Method	0.560	0.578	0.620
GlobalTrack	0.521	0.529	0.599
ATOM	0.518	0.506	0.576
SiamRPN++	0.496	0.491	0.569
SiamFC	0.336	0.339	0.420

Table 4. Performance Comparison for challenge attributes on the LaSOT test set

	Background Clutter	Deformation	Scale Variation
Proposed Method	0.475	0.602	0.562
GlobalTrack	0.436	0.533	0.520
ATOM	0.451	0.514	0.515
SiamRPN++	0.449	0.529	0.494
SiamFC	0.308	0.351	0.332

Table 5. Ablation experiments for the proposed context modeling module

	AUC	Precision	Normalized Precision
Proposed Method	0.560	0.578	0.620
Baseline	0.549	0.565	0.608

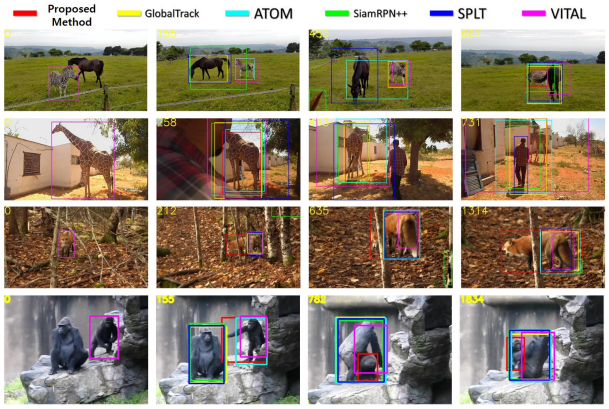

Fig. 3. Qualitative comparison on LaSOT test videos (*zebra-17, giraffe-10, fox-5* & *gorilla-9*)

IV. Experimental Results

4.1. Dataset and Evaluation Metrics

제안하는 물체추적 알고리즘을 학습하기 위해서는 LaSOT 데이터셋의 training 셋을, 평가하기 위해서는 LaSOT 데이터셋의 test 셋을 사용하였다. 데이터셋에 포함된 총 1400개의 영상 중 test를 위해 총 280개 비디오를 대상으로 평가를 진행하였으며, 해당 데이터셋은 30fps 기준 평균 길이가 1분 이상으로, 장기 추적 (long-term tracking) 성능을 평가할 수 있는 데이터셋이다. 또한 모든 프레임에 대해 상자 레이블이 되어 있다. 평가 지표의 산출을 위해서 우선 비디오의 모든 프레임들에 대해 예측된 상자와 레이블 된 상자간의 IoU 또는 거리 오차의 값들을 얻고, 이들에 대한 역치값 (threshold)을 변화시키면서 특정 값보다 높은/낮은 값을 가지는 상자의 비율을 측정하고, 이를 통해 성공 플롯 (success plot)과 정확도 플롯 (precision plot)을 얻는다. 이렇게 얻어진 두 플롯에서 세 가지 평가 지표를 도출하며, 각각은 성공 플롯, 정확도 플롯, 물체 크기에 대해 정규화(normalized)된 정확도 플롯의 AUC (area-under-curve) 값들이다.

4.2. Quantitative Evaluation

표 3은 정량적 성능 평가 및 비교의 결과이다. 제안하는 물체추적 알고리즘 (Proposed method)의 성능이 비교 대상 알고리즘들 대비 AUC 및 정확도 면에서 높은 평가 지표를 달성하는 것을 확인하였다. 특히 본 방법론과 같은 2단계 영역 검출 기반의 GlobalTrack[3] 알고리즘 대비 높은 성능 향상을 확인하였다. 그리고 ResNet-18 및 ResNet-50 특징점 기반의 다른 물체 추적 알고리즘인 GlobalTrack, ATOM[11], SiamRPN++[12]과 비교하였을 때도 가장 적은 계산량을 가지는 ResNet-18을 사용하면서 가장 높은 성능을 달성하는 것을 볼 수 있었다. 또한 표 4에서는 영상의 특성별 세부 성능 지표를 측정하였고 (AUC지표 사용) 배경 물체 (background clutter), 변형 (deformation), 스케일 변화 (scale variation)의 세 가지 특성에 대해 우수한 성능을 확인하였다. 특히, 본 방법론과 같은 2단계 영역 검출 기반의 GlobalTrack의 경우, 프레임 전체에 대해 전반적으로 목표 물체를 찾는 구조적 특성으로 인해 비슷한 물체 간의 혼동이 잦아, 배경 물체 특성에 대한 AUC 성능이 타 물체 추적 알고리즘 대비 낮은 것을 볼 수 있다. 하지만 본 연구에서 제안하는 알고리즘의 경우, 제안하는 맥락영역 활용 모듈을 추가함으로써 유사한 물체들을 성공적으로 분류할 수 있어 가장 높은 성능을 달성할 수 있었다. 추가적으로, 제안하는 물체 맥락영역 정보 활용 모듈의 효과를 더 정확히 검증하기 위하여 물체 추적 알고리즘에서 제안하는 모듈만을 제외한 모델 (표 5에서 Baseline)을 같은 설정에서 학습시키고 이의 성능을 비교 실험을 통해 얻었다. 그 결과, 표 4와 같이 제안하는 모듈의 추적 성능 향상 효과를 확인할 수 있었다.

4.3. Qualitative Evaluation

Fig. 3은 제안하는 물체 추적 알고리즘의 정성적인 비교를 위한 결과를 나타낸다. LaSOT 데이터셋의 test set에서 선별된 4개의 비디오에 대해 제안하는 물체 추적 알고리즘과 4개의 타 알고리즘이 출력하는 상자 (bounding box)를 선택된 프레임들에 대해 서로 다른 색을 사용하여 시각화하였다. 각 행은 서로 다른 비디오를, 각 열은 해당 비디오의 여러 프레임에 대한 시각화를 나타낸다. 시각화 및 비교 결과, 본 논문에서 제안하는 방법론이 타 물체 추적 알고리즘 대비 유사한 종류 또는 비슷한 외양을 가지는 물체들이 한 장면에 같이 등장하더라도 더 높은 정확도로 대상 물체의 영역을 올바르게 추정하는 것을 확인할 수 있었다.

V. Conclusion

본 논문에서는 물체 추적 문제에서 유사한 외양의 물체들이 등장하는 상황에서의 정확도를 향상시키기 위해, 전반적인 물체 맥락 정보를 활용하는 방법론을 제안하였다. 맥락 영역에서의 정보를 추출하고 활용하기 위해 MLP-Mixer 모델을 활용하였으며, 해당 모델에서는 전치 연산을 통해 맥락 영역들의 특징점 벡터의 채널 간의 상호작용을 모델링하여 영역 분류 단계에서의 구별성을 높이고 결론적으로 정확도를 높이고자 하였다. 제안된 알고리즘을 LaSOT 데이터셋의 test set의 비디오에 대해 검증해 본 결과, 높은 성능 지표와 함께 제안된 맥락 정보 모델링 모듈이 물체 추적 성능 향상에 기여함을 확인하였다.

본 연구에서는 비슷한 외양 또는 종류를 가지는 영역들 간의 분류 성능을 높이기 위한 특징점 학습 방법론에 대해 주로 논의하였다. 향후 연구에서는 선별된 영역이 아닌 장면 전체에 대한 특성을 각 객체의 특징점 표현에 반영할 수 있는 학습 방법론을 연구하는 방향성을 생각할 수 있으며, 비디오에서 얻을 수 있는 시계열 정보를 모두 반영할 수 있다면 목표 물체를 검출하기에 더욱 용이한 특징점을 학습할 수 있을 것으로 생각된다. 이를 위해 제안된 방법론을 기반으로 한계점을 고찰해 보면, 연속된 프레임 간에 물체들의 움직임 (motion) 정보를 활용하지 않았다는 점이 있다. 물체들의 공간적 배치 및 이전 프레임들에서의 궤적 (trajectory) 정보 등을 활용한다면 더 정확하게 물체 추적을 수행할 수 있을 것으로 예상된다.

ACKNOWLEDGEMENT

This research was supported by the Republic of Korea Government (Ministry of Science and ICT) through the research fund of the National Research Foundation of Korea (NRF) under grants NRF-2021R1C1C2095450, RS-2023-00242528

REFERENCES

[1] H. Nam and B. Han, “Learning Multi-Domain Convolutional Neural Networks for Visual Tracking”, in Proc. CVPR, pp. 4293-4302, Boston, MA, USA, June. 2015. DOI: 10.1109/CVPR.2016.465

[2] L. Bertinetto, J. Valmadre, J.F. Henriques, A. Vedaldi, and P.H. Torr, “Fully-Convolutional Siamese Networks for Object Tracking”, in Proc. ECCV Workshops, pp. 850-865, Amsterdam, Netherlands, Oct. 2016. DOI: 10.1007/978-3-319-48881-3_56

[3] L. Huang, X. Zhao, and K. Huang, “GlobalTrack: A Simple and Strong Baseline for Long-term Tracking”, in Proc. AAAI, pp. 11037-11044, New York, USA, Feb. 2020. DOI: 10.1609/aaai.v34i07.6758

[4] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 39, No. 6, pp. 1137-1149, June 2016. DOI: 10.1109/tpami.2016.2577031

[5] I. Tolstikhin et al., “MLP-Mixer: An All-MLP Architecture for Vision”, in Proc. NeurIPS, pp. 24261-24272, Virtual, Dec. 2021.

[6] H. Fan et al., “LaSOT: A high-quality benchmark for large-scale single object tracking”, in Proc. CVPR, pp. 5369-5378, Long Beach, USA, June 2019. DOI: 10.1109/CVPR.2019.00552

[7] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition”, in Proc. CVPR, pp. 770-778, Las Vegas, USA, June 2016. DOI: 10.1109/CVPR.2016.90

[8] I. Loshchilo and F. Hutter, “Decoupled Weight Decay Regularization”, in Proc ICLR, pp., New Orleans, USA, May 2019. (https://openreview.net/forum?id=Bkg6RiCqY7)

[9] B. Li et al., “High Performance Visual Tracking with Siamese Region Proposal Network”, in Proc. CVPR, pp. 8971-8980, Salt Lake City, USA, June 2018. DOI: 10.1109/CVPR.2018.00935

[10] K. Simoyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition”, arXiv, 2014, arXiv:1409.1556

[11] M. Danelljan et al., “Atom: Accurate tracking by overlap maximization”, in Proc. CVPR, pp.4655-4664, Long Beach, USA, June 2019. (10.1109/CVPR.2019.00479)

[12] B. Li et al., “SianRPN++: Evolution of siamese visual tracking with very deep networks”, in Proc. CVPR, pp.4655-4664, Long Beach, USA, June 2019. (10.1109/CVPR.2019.00441)

[13] A. Vaswani et al., “Attention is All You Need”, in Proc. NIPS, pp.6000-6010, Long Beach, USA, Dec. 2017.

[14] L. Lin, “SwinTrack: A Simple and Strong Baseline for Transformer Tracking”, in Proc. NeurIPS, pp.16743-16754 New Orleans, USA, Dec. 2022.

[15] B. Yan, “Learning spatio-temporal transformer for visual tracking”, in Proc. ICCV, pp.10428-10437, Virtual, Oct. 2017. (10.1109/ICCV48922.2021.01028)

Author

Janghoon Choi received the B.S. degree in electrical and computer engineering, and Ph.D. degree in electrical engineering and computer science from Seoul National University, Korea, in 2013 and 2021, respectively.

Dr. Choi joined the faculty of the Graduate School of Data Science at Kyungpook National University, Daegu, Korea, in Sep. 2022. He is currently an Assistant Professor in the Graduate School of Data Science at Kyungpook National University. He is interested in computer vision problems including visual tracking, video understanding and image restoration.