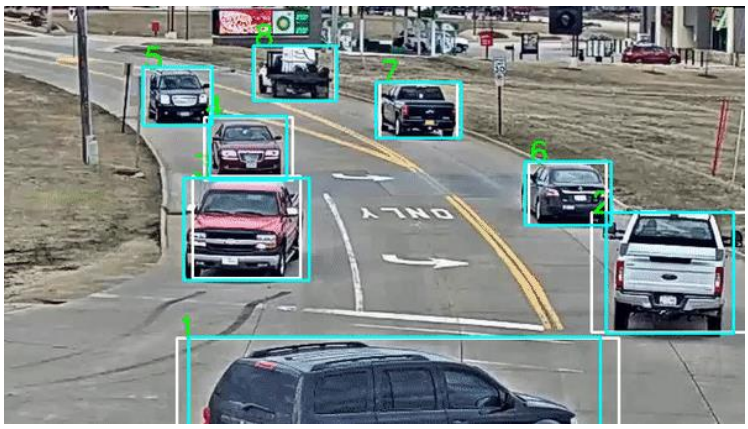
Visual Object Tracking (VOT)

Object Tracking(객체 추적)이란 비디오 영상에서 시간에 따라 움직이는 어떤 물체 또는 여러 개의 물체의 위치를 찾는 과정을 말한다.

객체 추적은 VOT(Visual Object Tracking)과 MOT(Multiple Object Tracking)으로 나뉘어진다.

VOT는 단일 객체를 추적하는 문제이고 Class-Agnostic(물체의 종류는 모름)하다. VOT에서 주어지는 정보는 첫 번째 프레임에서의 객체의 위치(bbox)이고 다른 정보는 없다. 따라서 객체에 대한 자세한 정보는 모르지만 첫 프레임에서 객체가 어디에 있는지 알면 비디오에서 계속 그 객체를 추적해 나가는 문제이다.

MOT는 여러 객체를 추적하는 문제이고 Detection Based Tracking이라고 많이 알려져 있다. 따라서 매 프레임마다 Object Detection을 수행하고 그 결과를 Tracking과 연관짓는다.
Tracking과 연관짓는다는 의미는 현재 프레임에서 Detection한 좌표들이 있을 때, 이 좌표들과 직전 프레임에서 Detection한 좌표들을 연결하는 것을 말한다. 이 과정까지 거쳐야 특정한 여러 객체들을 추적할 수 있다. ex) 자동차들을 구분짓으며 쫓아다님



보통 Object Tracking이라고 하면 VOT를 의미한다.

지금부터 딥러닝 기반 VOT에 대해 알아보겠다.

VOT는 크게 세 가지 특징과 두 가지 제약 조건이 있다.

특징1. Given Arbitrary Target

어떤 물체를 쫓아야 되는지 정해져 있지 않다 ⇔ 어떤 물체든지 쫓을 수 있다.
사람이 처음 지정한 객체 혹은 자동적으로 지정된 객체가 무엇이든지 그것을 추적하려고 시도한다. 처음 정해진 객체가 사람이면 사람을 추적, 차면 차를 추적, 신호등이면 신호등을 추적한다.
다른 기계학습 모델이 좀 specific했다면 VOT는 general하다고 생각하면 된다.

특징2. Localize the Target in Video

추적되는 물체는 비디오에서 Localization된다. (bbox로 나타나진다.)

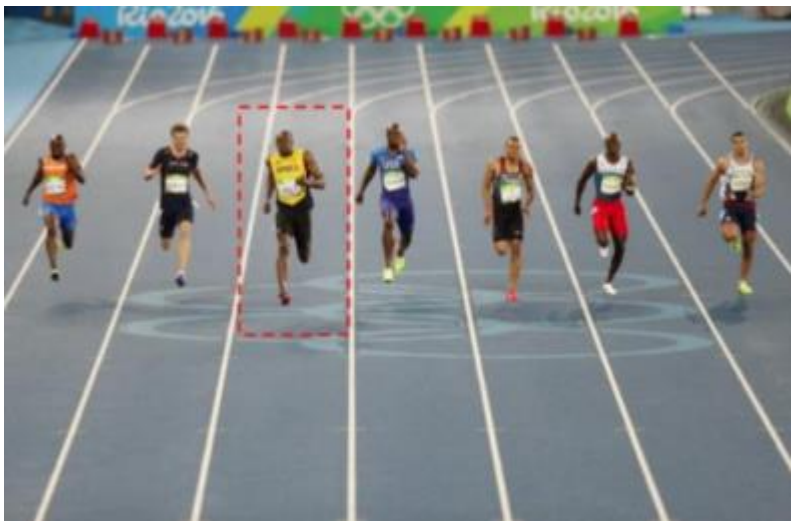
특징3. Class-Agnostic

클래스와 전혀 관계가 없다. 처음 주어진 객체가 무엇인지 모르지만 계속 추적한다.

제약 조건1. Hard Negative

실제로는 negative인데 positive라고 잘못 예측하기 쉽다. 아래 그림에서 빨간 bbox의 사람만 추적한다고 하면 나머지 사람들은 모두 negative가 된다. 그러나 모두 사람이기 때문에 비슷한 Feature를 찾을 테고 프레임마다 다른 사람을 추적할 수도 있다.

즉 Hard Negative한 제약 조건을 가지고 VOT 문제를 풀어야 된다.

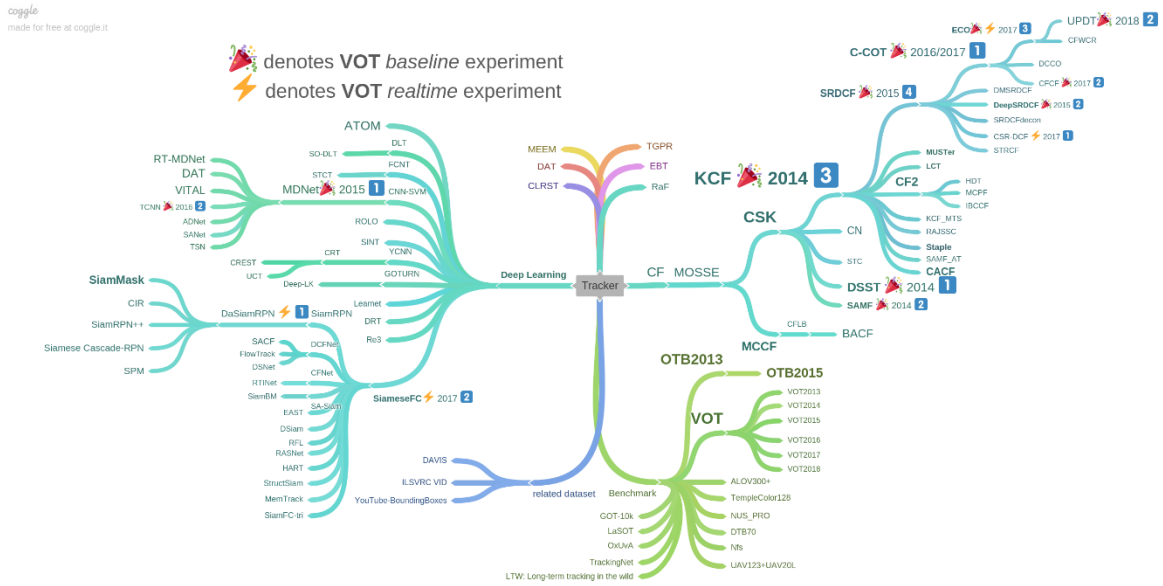


제약 조건2. Object Deformation

추적하려는 객체가 항상 같은 모습이 아니다. 예를 들어서 나비는 날개를 폈다 접었다 하고 사람은 앉았다가 일어날 수도 있다. 이런 조건에 대해서도 VOT 문제를 해결할 수 있어야 한다.

다음은 VOT 논문들을 정리한 그림이다.

https://github.com/foolwood/benchmark_results



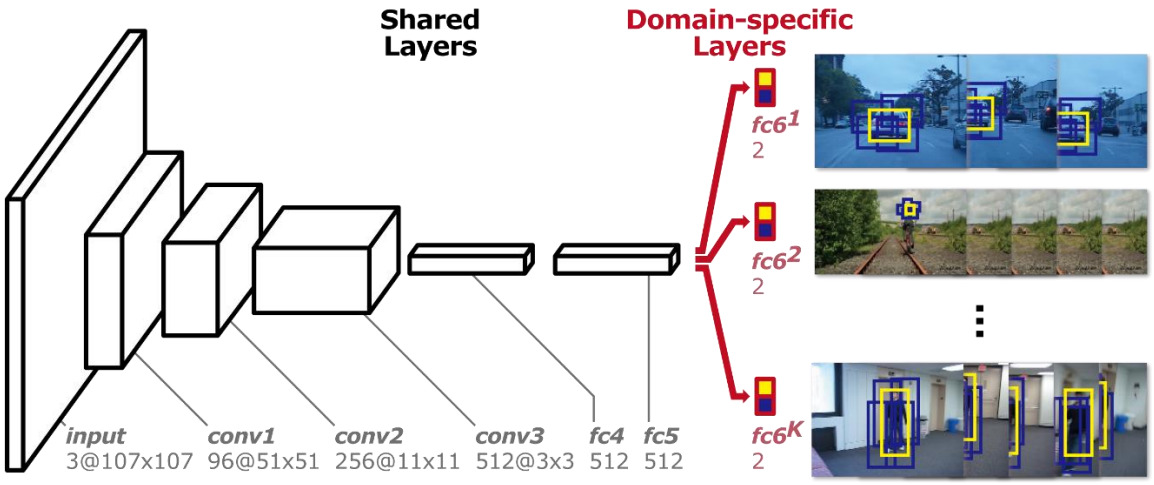
VOT는 크게 세 카테고리로 나뉘어진다.

Correlation Filter 기반(북동) / CNN-based 기반(북서) / Siamese Network 기반(남서)

세 방식 모두 T-1 프레임 타깃을 기준으로 T 프레임의 어느 영역(패치)이 가장 유사한지 찾는다는 관점에서 동일하다.

다만 첫 번째는 Correlation Filter를 이용해서 유사도를 측정하는 고전적인 방법이고 학습도 하지 않는다. CNN 기반과 샴 네트워크 기반이 본격적으로 딥러닝을 이용해서 성능 향상을 이루었다.

1. CNN-Based Tracking (MDNet)



MDNet의 네트워크는 프레임의 패치가 입력으로 들어가 3번의 conv연산과 3개의 fc layer를 거쳐서 1(찾고자 하는 객체, positive)인지 0(상관없는 객체, negative)인지 찾는다.

다만 fc5 layer까지만 학습하고 fc6은 Test하면서 파인튜닝하는 용도로 남겨둔다.

예를 들어 fc5 layer까지는 자동차에 대한 feature를 학습하고 fc6은 실제로 테스트하려는 사람에 대해서 feature를 학습하는 방식이다. (자동차 데이터로 학습, 사람 데이터로 테스트)

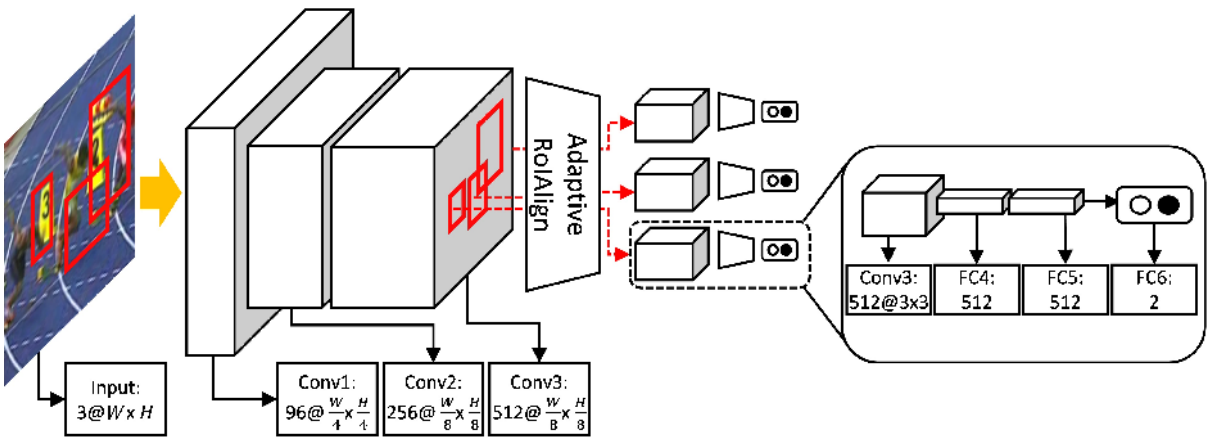
학습 과정에 대해 더 자세히 말하자면, 먼저 자동차 이미지에서 자동차 영역을 Positive 패치로, 주변에 있는 영역을 Negative 패치로 만든다. 이후 패치들을 각각 네트워크에 통과시켜서 positive 패치는 1이 나오고 negative 패치는 0이 나오도록 학습시킨다.

테스트 과정에서는 사람 영상의 첫 프레임에서 지정된 영역을 positive 패치로, 주변에 있는 영역을 negative 패치로 만든다. 이후 네트워크를 통과시키고 fc6 layer에 대해서만 학습한다. 두 번째 프레임부터는 정답을 모르기 때문에 랜덤하게 패치들을 생성하고 네트워크를 통과시킨다. 그러면 fc6 layer에서 사람 패치를 어느 정도 파악할 수 있으므로, 랜덤한 패치들 중 사람에 해당하는 패치가 정답으로 추정된다. 이후 두 번째 프레임에서 정답으로 추정된 패치를 positive로, 다른 패치들을 negative로 만들어서 fc6 layer를 다시 학습시킨다. 이러한 과정을 반복하면 fc6 layer는 사람 패치에 대해서 아주 잘 구분해내는 parameter를 얻게 된다.

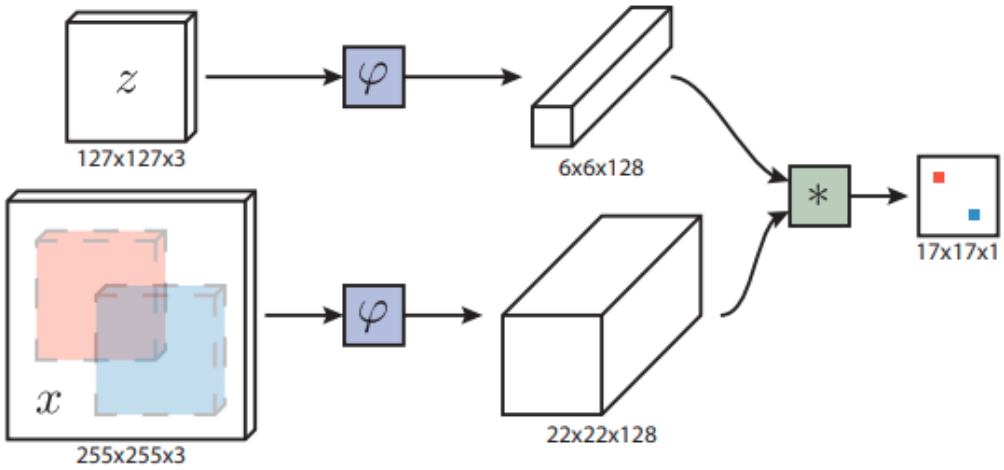
결론적으로 MDNet은 서로 공유되는 레이어들(conv1~fc5)과 Domain-Specific한 Layer(fc6)로 이루어지며 전자는 Feature 추출을 목적으로, 후자는 도메인에 해당하는 정답을 잘 찾으려는 목적으로 사용된다고 보면 된다. (Feature Extractor – Domain Specific Classifier)

다만 학습 과정과 테스트 과정에서 패치들을 전부 네트워크로 통과시킨다는 점이 매우 비효율적이라고 말할 수 있다.

2018년에는 Real-time MDNet이라는 네트워크가 나왔는데, 패치 단위로 네트워크를 통과시키는 것이 아닌 이미지 단위로 네트워크를 통과시키고 feature level에서 Positive Sample, Negative Sample을 만든다. Fast-RCNN의 ROI Pooling처럼 생각하면 될 것 같다. 이 방식으로 정확도와 속도를 모두 높였다고 한다.



2. Siamese Network Based (SiamFC)



샴 네트워크로 객체 추적을 시도한 가장 처음 논문이다. 샴 네트워크는 weight를 공유하는 두 개의 동일한 네트워크를 말한다. 두 네트워크는 서로 다른 입력 값을 받지만 두 결과가 서로 결합되면서 하나의 결과를 도출하는 구조이다.

SiamFC에서 z는 Target으로, 첫 번째 프레임에서 지정된 영역이다. x는 Searching Area로 N번째 프레임이다. z와 x를 각각 CNN에 통과시키고 그 결과를 correlation(상관 분석)해서 x의 어느 위치가 z랑 유사한지 알아낸다. 이 유사도가 나타난 feature map은 score map이라고 한다. 예를 들어 x의 빨간 영역이 z랑 비슷하다면 score map에서 좌측 상단 부분(x의 빨간 영역과 매칭되는 부분)을 1로 나머지는 -1로 예측한다.

아래 예시를 보면 상단 이미지가 z(첫 번째 프레임에서 지정된 영역), 하단 이미지가 x(N번째 프레임)라고 보면 된다. 빨간 박스는 두 이미지를 모두 같은 feature공간으로 매핑해서 correlation했을 때 높은 값이 나오는 영역이다.

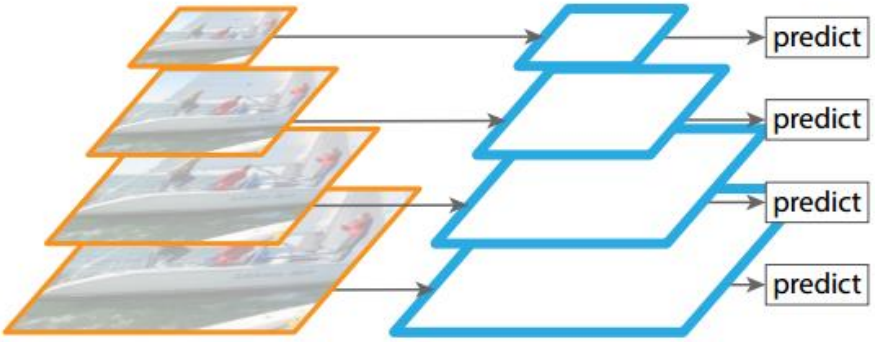
참고로 Correlation 과정은 Convolution 과정과 일치하다.

테스트 과정은 더욱 간단하다.

첫 번째 프레임에서 지정된 영역(z)을 CNN에 통과시켜 필터를 만든다. (Feature Map이라고 표현 안 하고 필터라고 표현한 이유는 x의 feature에서 필터를 통과시켜 유사도를 측정하는 과정이기 때문이다.)

두 번째 프레임부터는 CNN을 통과시켜 Feature Map을 만들고 처음 만든 필터와 Convolution해서 유사도를 측정한다. 이후 유사도가 높게 나오는 영역으로 bbox를 지정한다. 이 과정을 반복한다.

또한 테스트할 때는 이미지를 여러 스케일에 걸쳐서 분석하기 위해 이미지 피라미드 기법을 사용했다고 한다. (3또는 5 레벨) 여러 스케일에 걸쳐서 분석하는 이유는 객체의 크기가 다양할 수 있기 때문이다. 예를 들어 객체가 카메라와 가까이 위치하면 커보일테고 멀리 위치하면 작아 보일텐데, 객체의 크기가 크면 스케일을 크게 봐서(넓은 시야에서 객체를 봐서) 객체를 잘 찾게 하고 객체의 크기가 작으면 스케일을 작게 봐서(좁은 시야에서 객체를 봐서) 객체를 잘 찾게 하는 것이다. 참고로 이미지를 축소시킬수록 detail을 잃어버리기 때문에 이미지 스케일은 큰 값을 가진다. 즉 넓은 영역에 대한 feature를 잘 찾는다.



(a) Featurized image pyramid

SiamFC의 강점은 추적 속도라고 볼 수 있다. SiamFC가 발표되기 전 추적기들은 추적 대상(positive)과 배경(negative)를 분류하기 위한 분류기를 포함하는데, 추적이 진행되는 프레임마다 그 분류기의 학습이 수행된다. (MDNet이 대표적) 이러한 온라인 학습(영상 일부를 이용한 학습)은 추적 속도가 느려지는 문제를 지니고 있다. SiamFC는 이러한 추적 속도의 문제를 해결하기 위해 오프라인 학습(영상 전체를 이용한 학습)을 채택하였다. SiamFC를 학습하기 위해 많은 양의 예시-검색 영상 데이터 쌍을 준비해야 하는 어려움이 있지만 ImageNet Video와 같은 대규모 데이터셋을 활용하여 학습시킨 결과, 표1에서 확인할 수 있듯이 정확도는 온라인 방식의 추적기에 근접하면서도 압도적인 추적 속도를 보이는 것으로 발표되었다.

표 1 VOT2015 벤치마크[4]를 통한 추적기들의 성능 비교

Tracker	정확도	속도(fps)
MDNet	0.5620	1
EBT	0.4481	5
DeepSRDCF	0.5350	〈1
SiamFC–3s	0.5335	86
SiamFC	0.5240	58

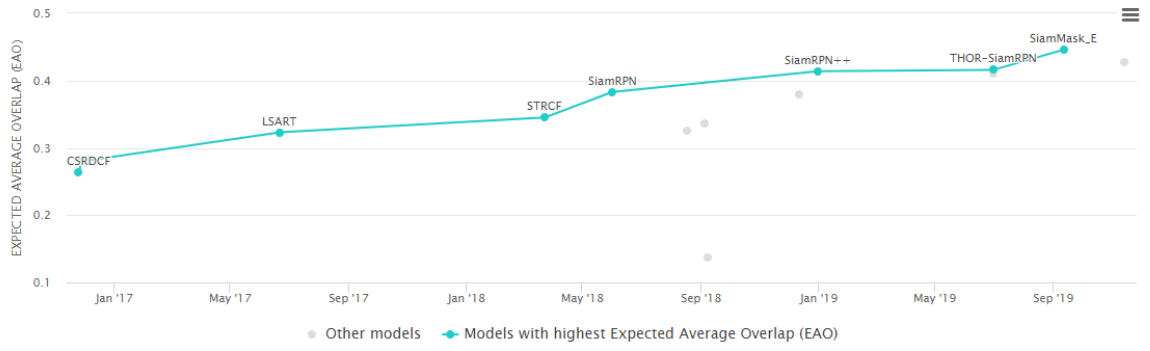
출처 Reproduced from [1].

SiamFC는 이후 SiamRPN, SiamMask 등으로 계속 발전되어 VOT 대회에서 항상 높은 성능을 보인다.

Visual Object Tracking on VOT2017/18

Leaderboard Dataset

View Expected Average Overlap (EAO) by Date for All models



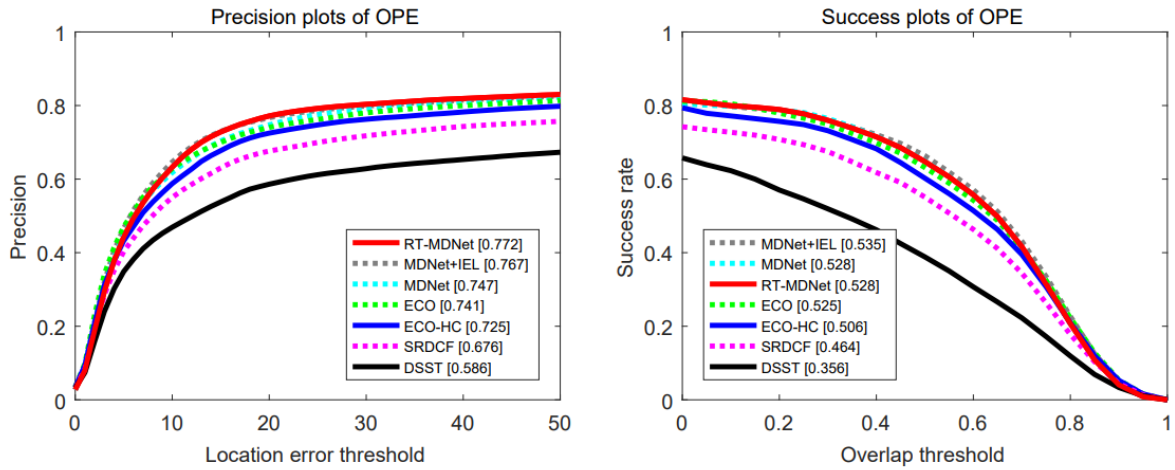
Filter: untagged

Edit Leaderboard

Rank	Model	Expected Average Overlap (EAO)	Extra Training Data	Paper	Code	Result	Year	Tags
1	**SiamMask_E**	0.446	✓	Fast Visual Object Tracking with Rotated Bounding Boxes			2019	
2	**SiamFC++**	0.428	✓	SiamFC++: Towards Robust and Accurate Visual Tracking with Target Estimation Guidelines			2019	
3	**THOR-SiamRPN**	0.4160	×	Tracking Holistic Object Representations			2019	
4	**SiamRPN++**	0.414	×	SiamRPN++: Evolution of Siamese Visual Tracking with Very Deep Networks			2018	

https://paperswithcode.com/sota/visual-object-tracking-on-vot201718

마지막으로 VOT의 벤치마크 보는 법에 대해 소개하겠다.



Location Error는 예측한 bbox와 GT의 중심점 사이의 거리를 의미한다. 가로축 수치는 거리에 대한 threhold로, 만약 50이면 50픽셀까지는 정답으로 인정한다는 의미이다. 그리고 대괄호 안에 적힌 수치 (0.772 등)는 25픽셀만큼 떨어진 정도를 정답으로 인정할 때 precision을 나타낸 것이다.

Overlap은 예측한 bbox와 GT가 겹친 정도(IoU)를 의미한다. 가로축 수치는 겹친 정도에 대한 threshold로, 만약 1이면 전부 다 겹쳐야 정답으로 인정한다는 의미이다. 대괄호 안에 적힌 수치는 AUC(Area Under Curve)를 의미한다.

Trackers	A	R	EAO
Ours	0.507	0.375	0.247
SiamRPN	0.490	0.460	0.244
CSRDCF++	0.459	0.398	0.212
SiamFC	0.502	0.604	0.182
ECO_HC	0.494	0.571	0.177
Staple	0.530	0.688	0.170
KFebT	0.451	0.684	0.169
SSKCF	0.530	0.656	0.164
CSRDCFf	0.475	0.646	0.158
UCT	0.490	0.777	0.145
MOSSEca	0.400	0.810	0.139
SiamDCF	0.503	0.988	0.135

VOT의 평가 지표에는 A(Accuracy), R(Robustness), EAO(Expected Average Overlap)도 있다.

Accuracy는 비디오에서 얼만큼 IoU가 threshold보다 높은지를 나타낸다. (높을수록 좋음)
Robustness는 비디오에서 얼만큼 IoU가 threshold보다 낮은지를 나타낸다. (낮을수록 좋음)
EAO는 Accuracy와 Robustness를 함께 평가하는 지표라고 한다.