Watch Only Once: An End-to-End Video Action Detection Framework

Shoufa Chen, Peize Sun, Enze Xie, Chongjian Ge, Jiannan Wu and et al.

2023.01.21 논문 리뷰

배성훈

Watch Only Once: An End-to-End Video Action Detection Framework

- **Research Background**:
  - 기존의 Video Action Detection은 Actor localization과 Action Classification을 나눠서 Two-stage로 진행하거나, One-stage에 2개의 다른 모델을 적용해 학습하는 방법 제시
  - 기존 방법의 문제
    - Actor localization이 Video clip의 key frame에서 actor bounding boxes를 예측하기 위해 2D detection model에 의존
      - ➢ Clip의 이웃하는 frame까지 고려하면 localization noise, 상당한 계산량, Memory cost 발생
    - Action classification은 video sequence에서 embedded temporal knowledge를 추출하기 위해 3D video model에 의존
      - ➢ Single key frame은 Action classification에 대해 temporal motion representation이 좋지 못함
  - Keyframe은 Actor localization에 positive, Action classification에는 negative
  - Multiple frame는 Actor localization에 negative, Action classification에는 positive

Watch Only Once: An End-to-End Video Action Detection Framework

- **Research Background**:
  - 해결을 위한 2가지 방법

1. Off-the-shelf person detector 사용

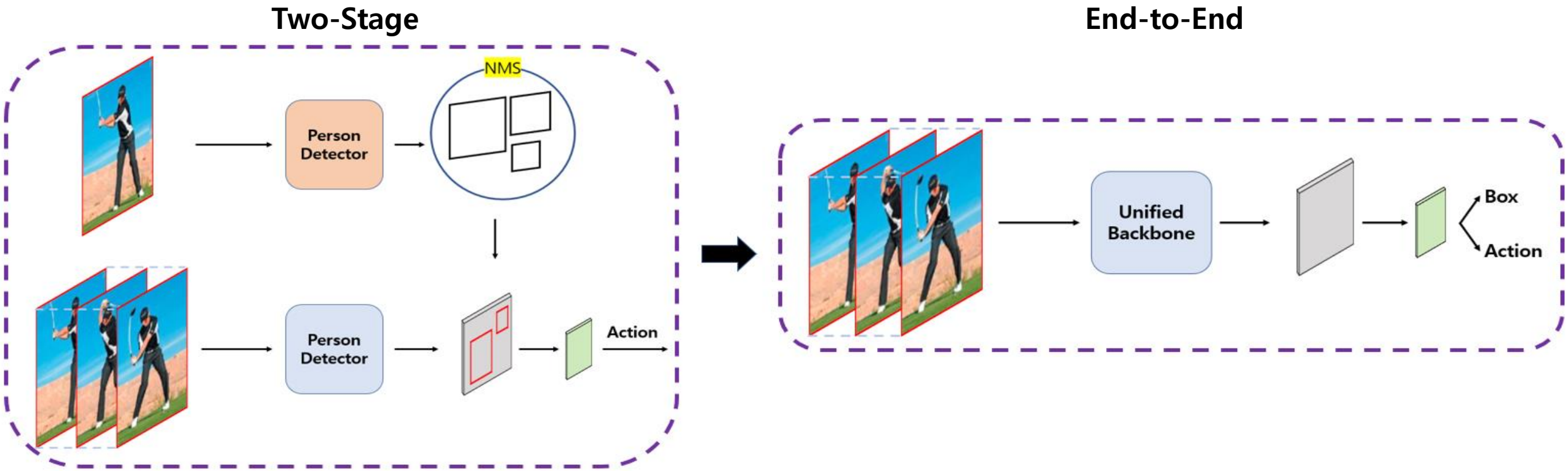
- Action classification 모델과 같이 공동으로 학습 X
- Actor proposals를 만듬
- 독립적인 video model이 action class 예측을 위한 입력으로 actor proposal와 raw frame 사용
- Actor localization을 위한 모델은 ImageNet이나 COCO로 사전학습된 person detector
- Target action detection dataset에 따라 Fine-tuned
  - ➢ 이러한 방법은 복잡하고 무거운 pipeline 생성 (Two backbone, Two stages)
  - ➢ 각 backbone을 optimization 하는 것은 sub-optimal 문제 야기

2. One stage: Actor detection, Action classification 공동 학습

- 학습 pipeline이 단순화
  - ➢ 전체 framework가 **Heavy computation, Memory cost** 문제 가짐

Watch Only Once: An End-to-End Video Action Detection Framework

- **Research Background**:
  - 저자는 기존의 방법에 대한 문제로부터 End-to-End 모델로 actor localization과 action classification을 모두 처리할 수 있는 방법을 제안
  - Single Unified framework인 WOO를 제안, 하나의 End-to-End 모델로 Actor localization, Action Classification
  - Video clip에서 Action classification의 확률, Actor bounding boxes의 좌표 예측
  - Single backbone network가 2D image detection, 3D video classification 모두 처리

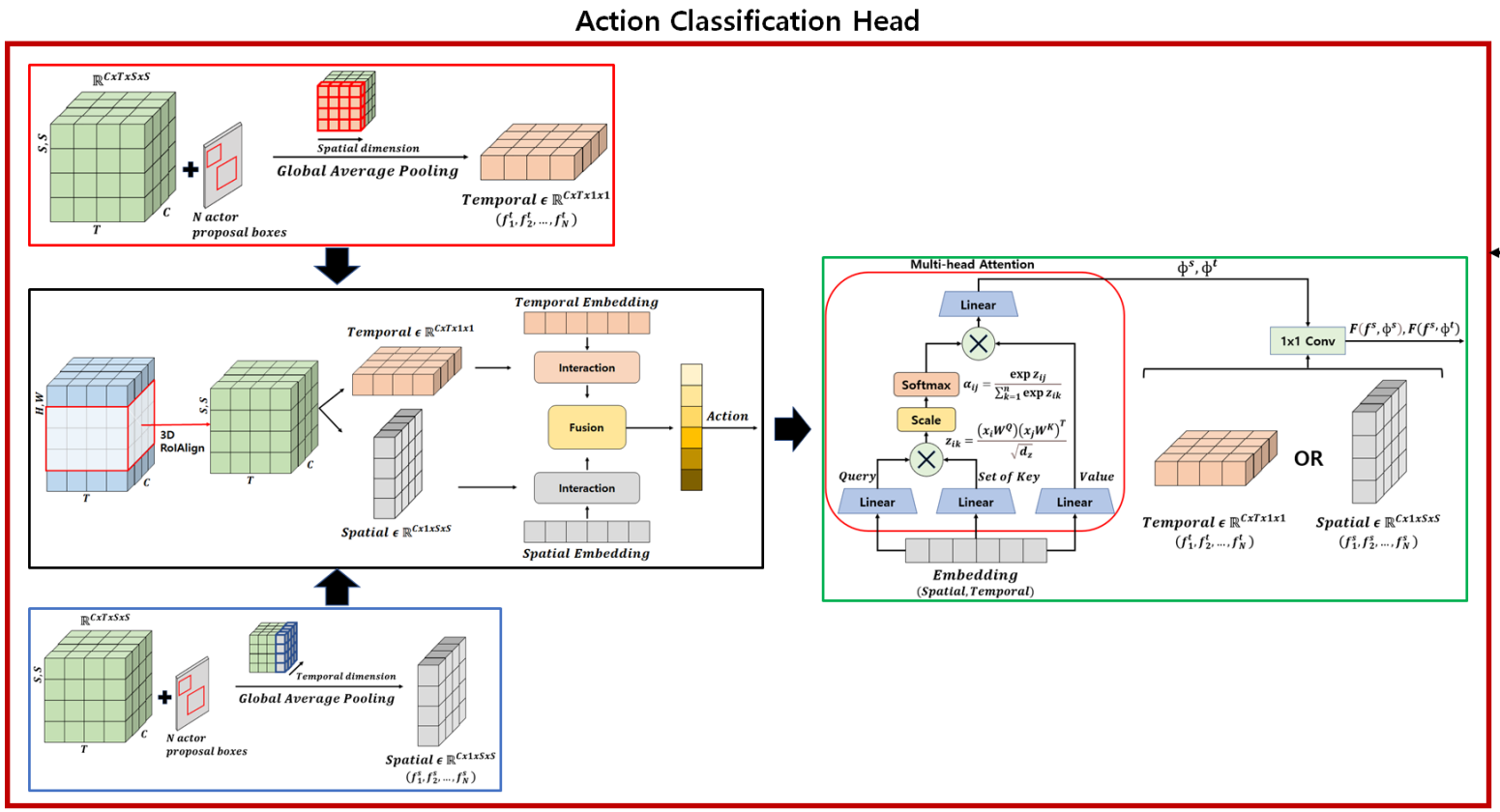


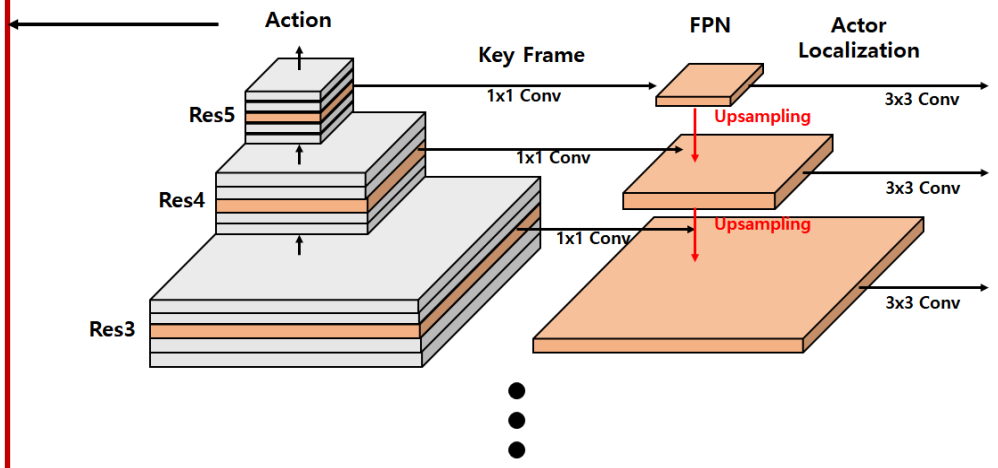
Watch Only Once: An End-to-End Video Action Detection Framework

- **Method**:
  - **WOO (Watch Only Once)**
  - **3 key component**
    - Unified backbone
      - Light weight
      - Backbone network 초기 단계의 모든 frame의 feature에서 key frame feature 분리
      - 모델이 깊어질수록 key frame feature가 이웃하는 frame과의 상호작용이 많아짐
        - ➢ Clip의 이웃하는 frame까지 고려하면 localization noise, 상당한 계산량, Memory cost 발생
      - P. 7
    - Spatial-temporal action embedding
      - Action classificaiton에서 Spatial, Temporal features 간의 뚜렷한 차이를 위해 설계된 방법론
      - P. 13
    - Spatial-temporal knowledge fusion mechanism
      - P. 15

Watch Only Once: An End-to-End Video Action Detection Framework

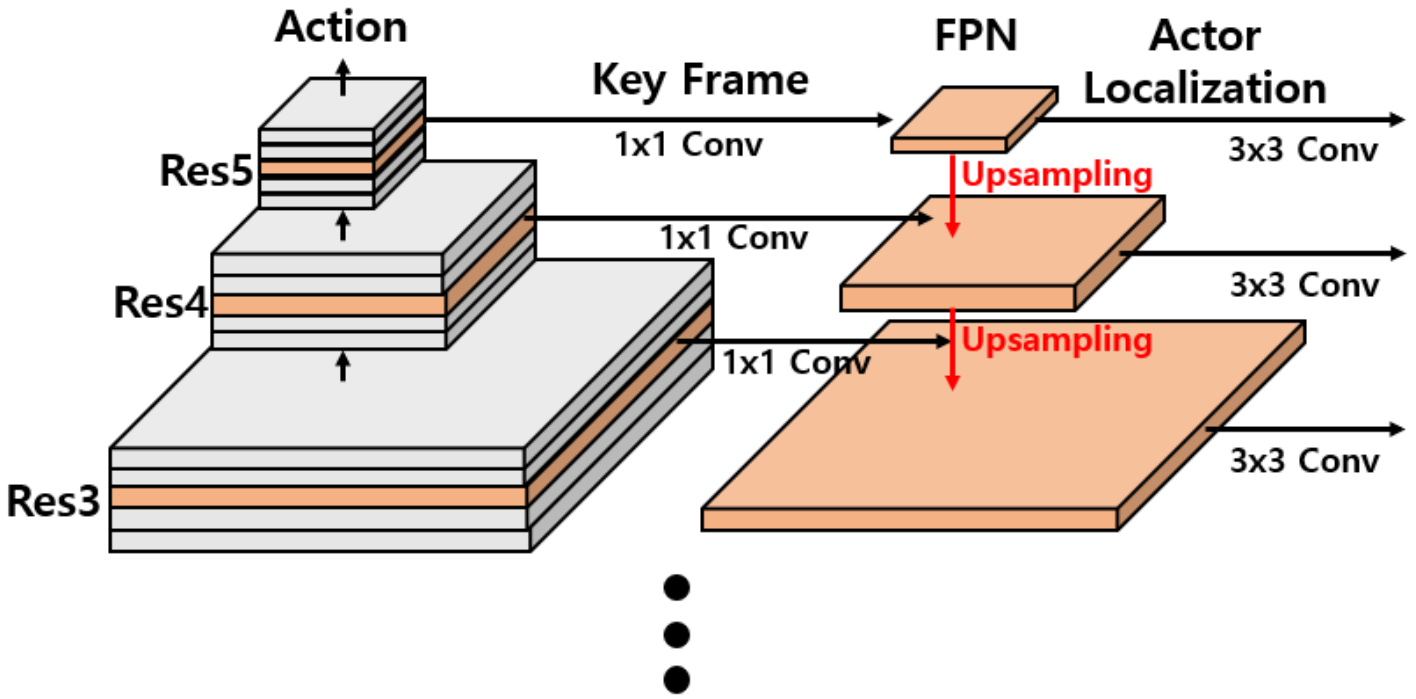
- **Method**:
  - Post processing (NMS) 없이, 주어진 video clip에서 Actor bounding box와 Action classes 출력
  - Input Spatial-temporal feature maps:
    $X \in \mathbb{R}^{CxTxHxW}$, $C: number\ of\ channels$; $T: Time$; $H, W: Spatial\ height\ width$
  - Video clip의 중간에 Key frame 배치: $X_t = [T/2] \in R^{C \times H \times W}$





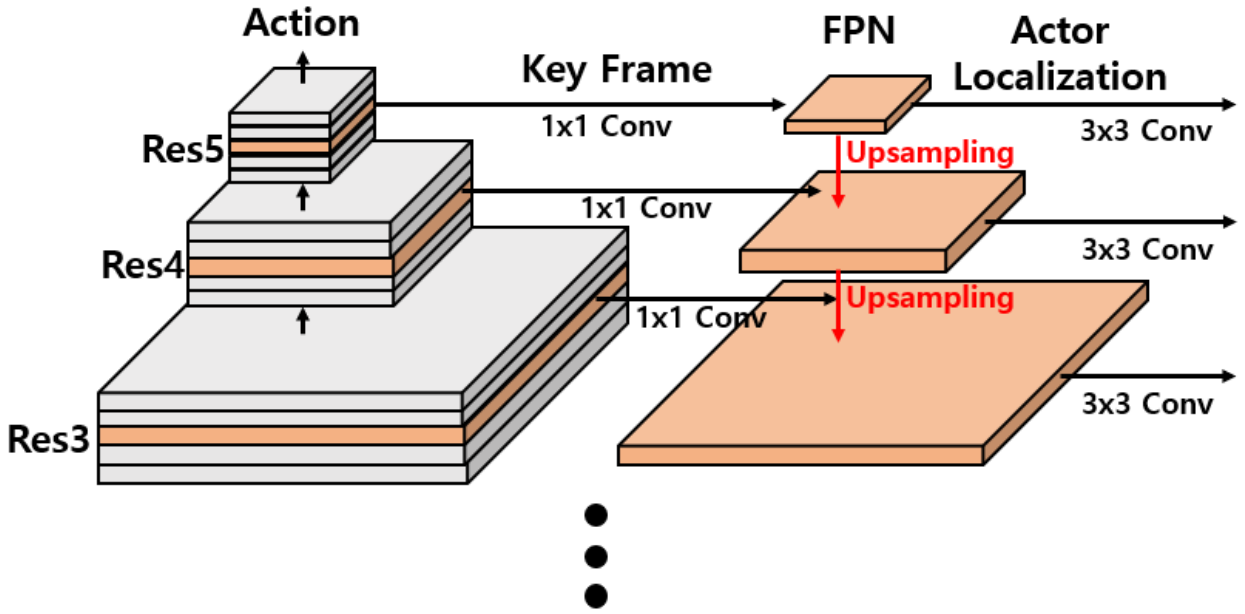
Watch Only Once: An End-to-End Video Action Detection Framework

- **Method**:
  - **Union Backbone**
    - 기존의 방법 (Two-stage)은 Key frame feature가 3D convolution (Temporal pooling, Temporal kernel size > 1)에 의해 인접한 Frame feature와 상호작용이 발생해 원치 않는 문제 야기
    - 이를 해결하기 위해, Temporal interaction이 발생하기 전에 네트워크의 초기 단계에서 Key frame feature 분리
    - 기존 Backbone인 SlowFast의 res5의 dilation을 제거하고 **FPN**을 Key frame feature 추출에 사용



Watch Only Once: An End-to-End Video Action Detection Framework

- **Method**:
  - **Union Backbone**
    - 설계 이점
      1. Actor localization head가 Hierarchical feature representation을 source feature로 사용해 object detection 이점
      2. Key frame feature는 FPN을 통해 backbone 초기 단계부터 video frame의 feature로부터 분리
         -> 모델이 깊어질수록 발생하는 인접한 frame간의 상호작용을 줄임
      3. Image feature를 입력으로 task를 수행하는 경량 FPN 모듈만 Backbone에 추가해 Parameter와 FLOPs 감소
         Video Backbone Architecture와 독립적이기 때문에 다른 Video Backbone 사용 가능

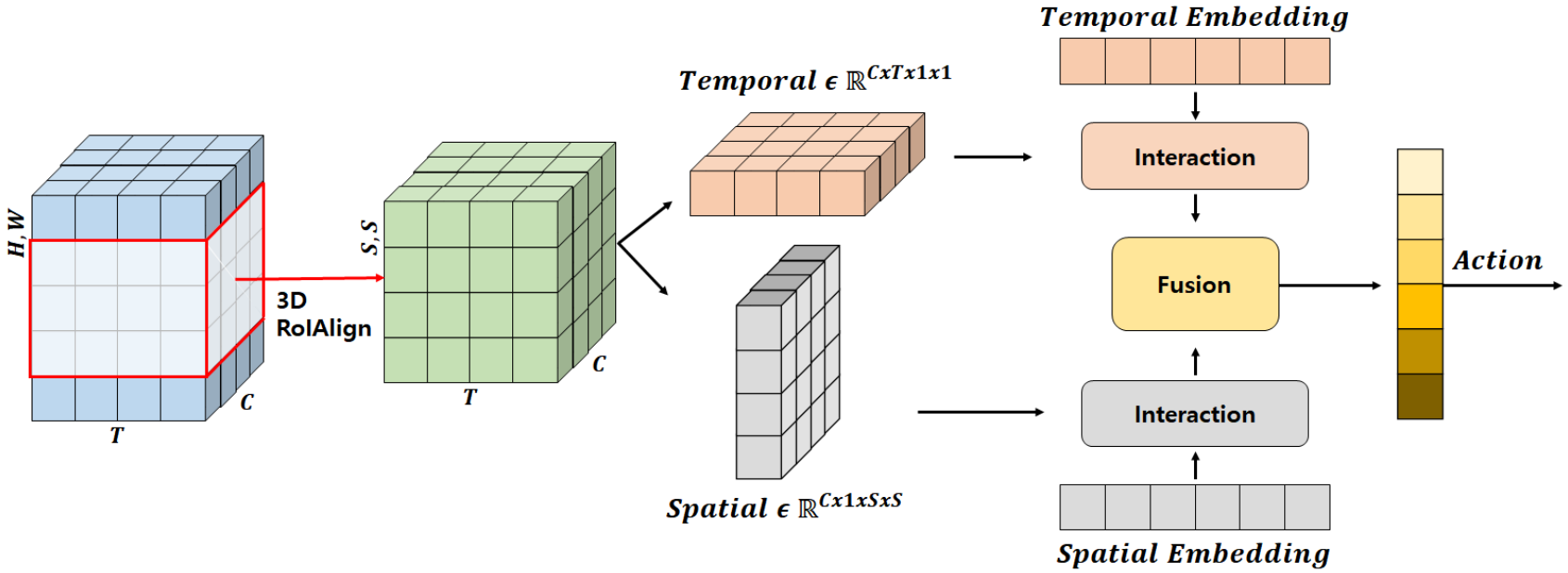


Watch Only Once: An End-to-End Video Action Detection Framework

- **Method**:
  - **Actor Localization Head**
    - End-to-End Actor localization 설계
    - FPN 모듈에 의해 생성되는 Hierarchical features를 입력으로 받음
    - Detection head는 Bounding box 좌표와 box가 actor를 포함하는 모델의 신뢰도를 나타내는 Score 예측
      - **Person Detector**
        - **Training**: Prediction과 Ground truth 간의 optimal bipartite matching을 위한 prediction loss 활용
        - **Evaluation**: Post processing (NMS) 사용 X

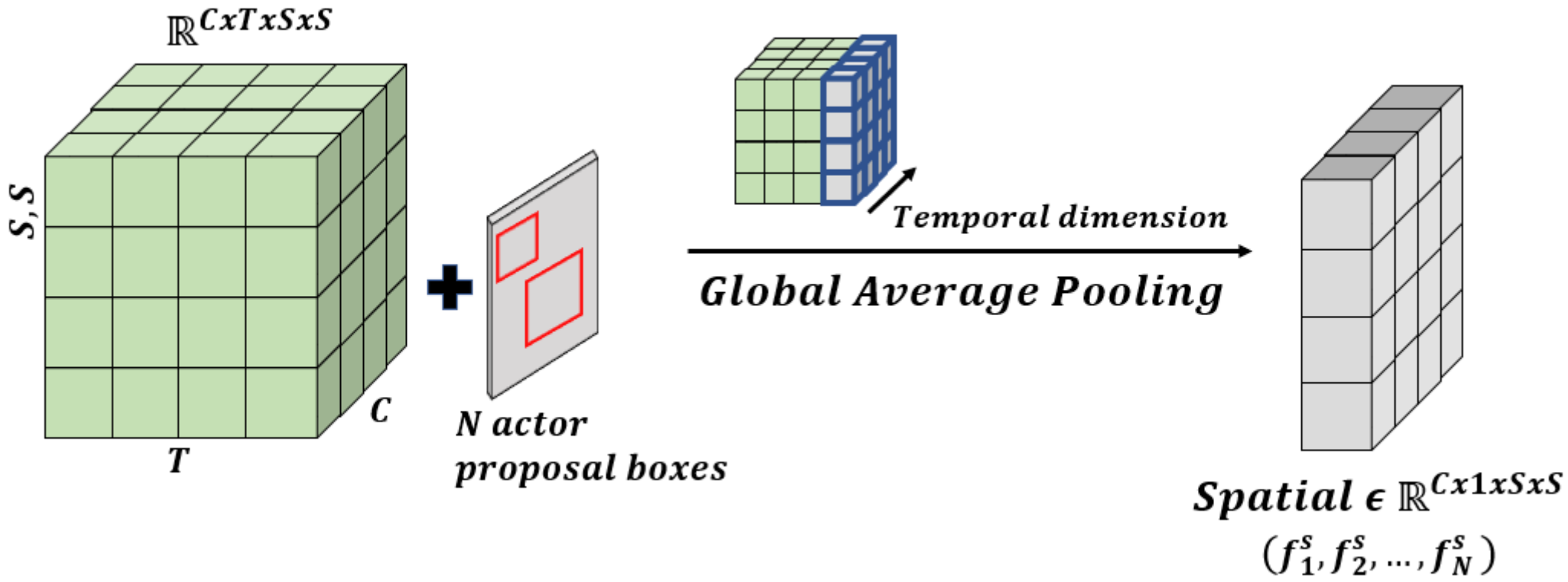
Watch Only Once: An End-to-End Video Action Detection Framework

- **Method**:
  - **Actor Classification Head**
    - **RoIAlign**의 입력: Res5 Output + Actor Localization Head Person Detector의 **N Actor Proposal Boxes**
    - 각 Box에 대해 아래의 과정을 거쳐 Final class prediction 출력
      1. Spatial Action Features
      2. Temporal Action Features
      3. Embedding Interaction
      4. Spatial-temporal knowledge fusion mechanism



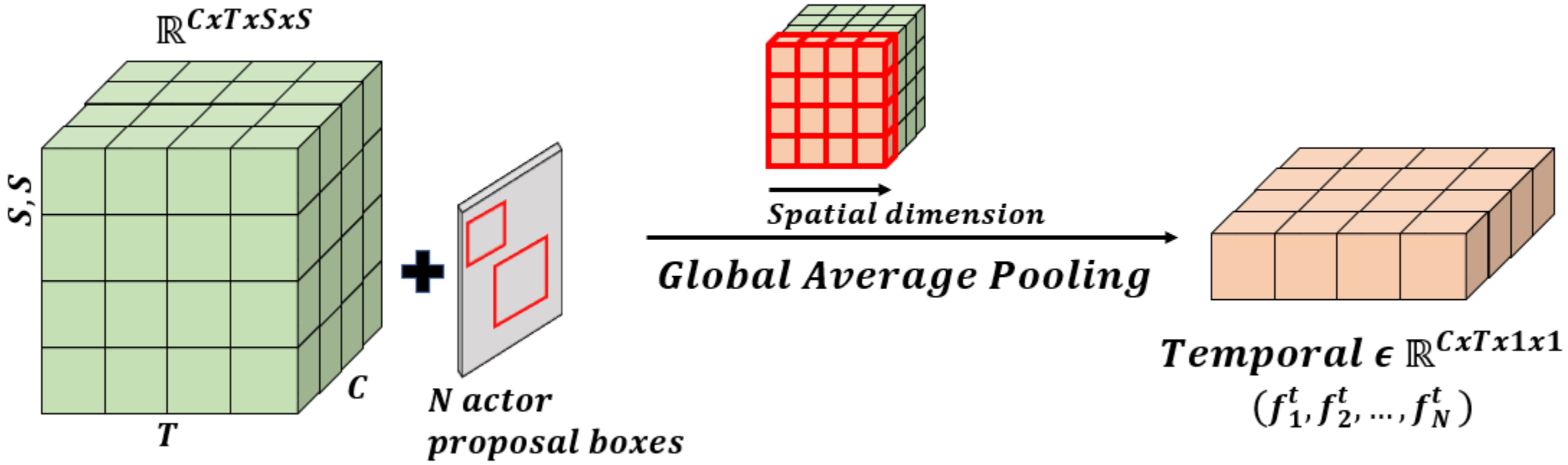
Watch Only Once: An End-to-End Video Action Detection Framework

- **Method**:
  - **Actor Classification Head**
    1. **Spatial Action Features**
       - RoIAlign을 거친 feature $\mathbb{R}^{CxTxSxS}$와 N actor proposal boxes에 Global Average Pooling 수행
       - Spatial Feature map $f_1^s, f_2^s, \dots, f_N^s \in \mathbb{R}^{Cx1xSxS}$ 생성



Watch Only Once: An End-to-End Video Action Detection Framework

- **Method**:
  - **Actor Classification Head**
    - **2. Temporal Action Features**
      - RoIAlign을 거친 feature $\mathbb{R}^{CxTxSxS}$와 N actor proposal boxes에 Global Average Pooling 수행
      - Temporal Feature map $f_1^t, f_2^t, \dots, f_N^t \in \mathbb{R}^{CxTx1x1}$ 생성

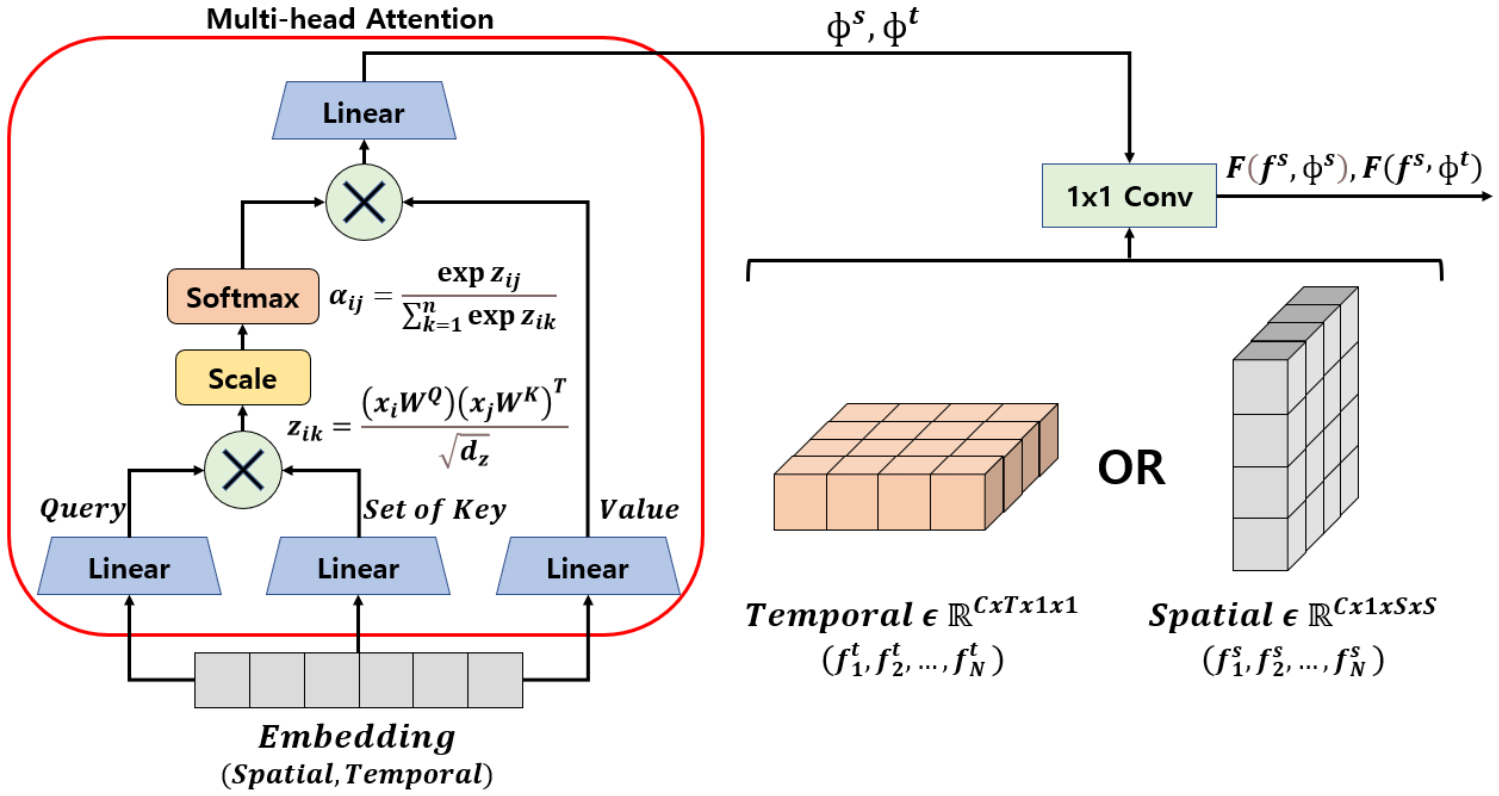


Watch Only Once: An End-to-End Video Action Detection Framework

- **Method**:
  - **Actor Classification Head**
    - **3. Embedding Interaction**
      - Spatial, Temporal 간의 더 뚜렷한 차이를 보이는 feature 추출
      - 풍부한 Instance 특징 추출
        - **Spatial embedding**: Spatial 속성 (Shape, Pose) 압축 : $\boldsymbol{E^s} \in \mathbb{R}^{\boldsymbol{Nxd}}$
        - **Temporal embedding**: Temporal 속성 (Dynamic motions, Action의 Temporal scale) 압축 $\boldsymbol{E^t} \in \mathbb{R}^{\boldsymbol{Nxd}}$
      - N 개의 features의 각각의 feature에 **배타적** Spatial, Temporal embedding 진행

Watch Only Once: An End-to-End Video Action Detection Framework

- **Method**:
  - **Actor Classification Head**
    - **3. Embedding Interaction**
      - 또한, **Actor 간의 Interaction** 포착을 위해, 모든 RoI feature에 **Attention module** 생성
      - **각 Actor RoI**가 **고유의 Spatial, Temporal embedding**을 가지고, Embedding이 Feature map보다 가볍기 때문에 효율성을 위해 Feature map 대신 Embedding에 **Attention mechanism** 채택



Watch Only Once: An End-to-End Video Action Detection Framework

- **Method**:
  - **Actor Classification Head**
    - **4. Spatial-temporal knowledge fusion mechanism**
      - 1x1 convolution의 Spatial, Temporal output을 Fusion 함수를 사용해 합침
      - 다양한 방법 실험: Summation, Concatenation, Cross-attention
      - Cross-Attention을 사용했을 때 다른 방법보다 좋은 성능 달성
      - 마지막으로, FC layer를 사용해 Final class prediction logits 얻음

Spatial-Temporal Fusion

Fusion	AP	AP_{50}	AP_{75}	GFLOPs
sum	14.8	20.5	16.9	68.0
concat	14.7	20.5	17.0	68.1
CA	**15.4**	**21.3**	**17.7**	**68.0**

Watch Only Once: An End-to-End Video Action Detection Framework

- **Method**:
  - **Objective Function**
    - End-to-End Localization, Classification

$$\mathcal{L} = \underbrace{\lambda_{cls} \cdot \mathcal{L}_{cls} + \lambda_{L1} \cdot \mathcal{L}_{L1} + \lambda_{giou} \cdot \mathcal{L}_{giou}}_{①} + \underbrace{\lambda_{act} \cdot \mathcal{L}_{act}}_{②}$$

① **Set prediction loss**
- Prediction과 Ground truth 간의 Optimal bipartite matching 생성
- $\mathcal{L}_{cls}$: 2개의 class 간의 Cross-entropy loss 나타냄 (Actor 포함 vs Actor 미포함)
- $\mathcal{L}_{L1}, \mathcal{L}_{giou}$: **Box loss**
- $\lambda_{cls}, \lambda_{L1}, \lambda_{giou}$: Loss 항 ($\mathcal{L}$)의 기여를 균형 있게 조정하는 **상수 스칼라**

② **Action**
- $\mathcal{L}_{act}$: Action classification을 위해 사용되는 **Binary cross entropy**
- λ_{act}: 가중치

Watch Only Once: An End-to-End Video Action Detection Framework

- **Experiments**:
  - **Implementation details**
    - **Training detail**
      - Optimizer: AdamW (Weight decay: 0.0001)
      - Mini batch: 16 video clip
      - 8 GPUs: 1개의 GPU에 2개의 Clip 할당
      - Training schedule:
        - Number of training iterations: 300
        - Learning rate schedule: Initial LR $2.5\text{x}10^{-5}$, 12 epochs (첫 번째 1000 iteration에서 **linear warm-up (10^{-3})**을 사용) Epoch 6, 10일 때 Decay factor 0.1
        - Batch size: 16
      - Backbone: Kinetics로 사전학습한 가중치 초기화, 새롭게 추가되는 layer는 Xavier로 사전학습된 가중치 초기화
      - Video Frame input 각각에 random scaling 적용, 가장 짧은 면의 범위를 256~320 pixels, 가장 긴 면 < 1333 pixels
      - Person Detector Head의 Loss weight: $\lambda_{cls} = 2, \lambda_{L1} = 5, \lambda_{giou} = 2$
      - Action Classification Loss weight: $\boldsymbol{\lambda_{act} = 4}$
      - Default proposal boxes: 100

Watch Only Once: An End-to-End Video Action Detection Framework

- **Experiments**:
  - **Implementation details**
    - **Inference detail**
      - 제안된 모델이 주어진 Input video clip에서 Actor detection, Action classification scores와 연관된 100개의 bounding boxes 예측
        - Actor Detection Scores: Box가 Actor 포함하는 확률
        - Action Classification Scores: Box에 상응하는 모든 Action class 확률
        - Confidence Score > 0.7 에 해당하는 Detected boxes만 Final output

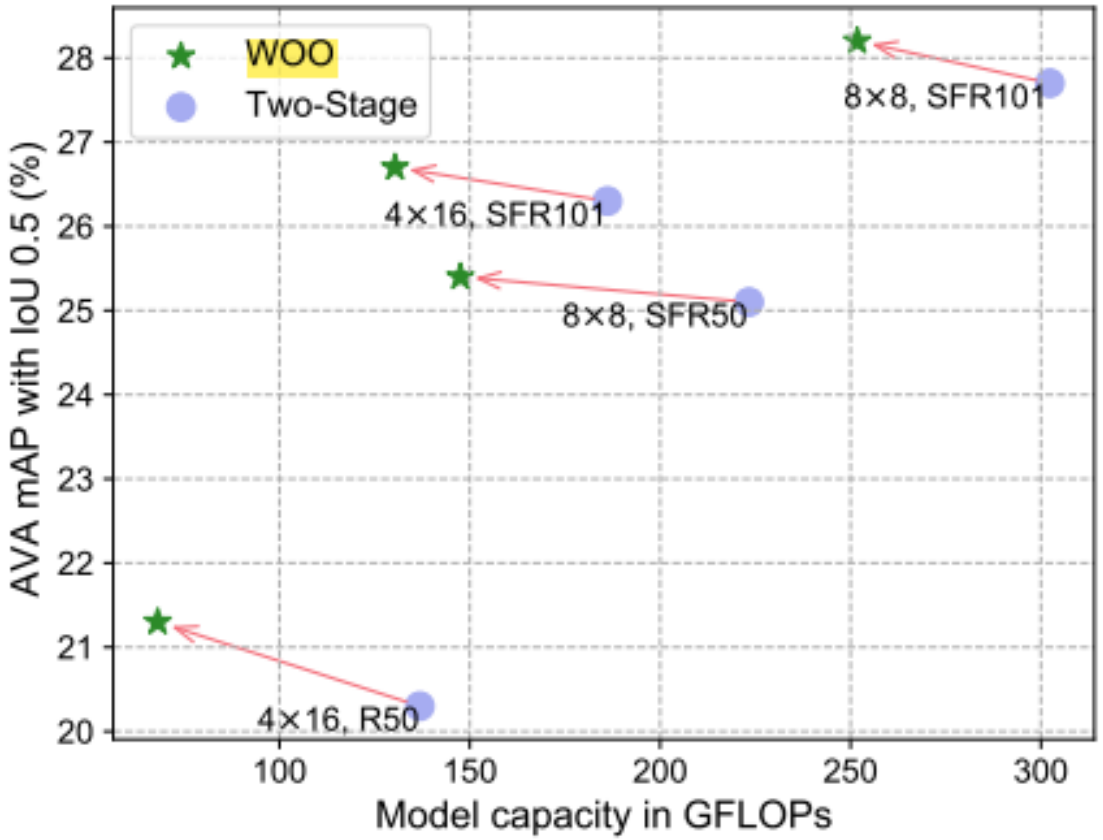
Watch Only Once: An End-to-End Video Action Detection Framework

- **Experiments**:
  - **SOTA와 비교**
    - AVA v2.1, v2.2 (mAP [IoU threshold = 0.5]) 대한 SOTA와 제안된 방법 비교
    - Testing에서 Single model과 Single cropping 을 사용한 방법만 고려
    - Model complexity를 현저히 줄이며, 기존 two-stage, two-backbone 을 뛰어넘는 성능
    - AVA v2.1
      - 기존의 SOTA보다 제안한 방법이
      - **0.5mAP ↑** (27.3 → 28.0), **GFLOPs 56.5 % ↓** (302.3 → 245.8)
    - AVA v2.2
      - 기존의 SOTA보다 제안한 방법이
      - **0.9mAP ↑** (27.4 → 28.3), **GFLOPs 50.6% ↓** (302.3 → 251.7)

model	AVA	E2E	$T \times \tau$	pre	val mAP	GFLOPs
AVA baseline [12]	v2.1	✗	64×1	K400	15.6	
Relation Graph [41]			36×1	K400	22.2	
VAT [10]			64×1	K400	25.0	
ACRN [30]			-	K400	17.4	
ATR [16]			-	K400	21.7	
Context-Aware [38]			32×2	K400	28.0	
LFB [37]			32×2	K400	27.6	
X3D-XL [7],			16×5	K400	26.1	
I3D [9]			64×1	K600	21.9	
SlowFast, R50 [8]			8×8	K400	24.7	223.3
SlowFast, R101 [8]			8×8	K600	27.3	302.3
WOO, SFR50		✓	8×8	K400	25.2	141.6
WOO, SFR101			8×8	K600	**28.0**	245.8
SlowOnly, R50 [8]	v2.2	✗	4×16	K400	20.3	136.8
SlowFast, R50 [8]			8×8	K400	24.7	223.3
SlowFast, R101 [8]			8×8	K600	27.4	302.3
WOO, SR50		✓	4×16	K400	21.3	68.0
WOO, SFR50			8×8	K400	25.4	147.5
WOO, SFR101			8×8	K600	**28.3**	251.7

Watch Only Once: An End-to-End Video Action Detection Framework

- **Experiments**:
  - **Model complexity and Accuracy**
    - 기존의 Two stage 접근법보다 WOO가 더 높은 mAP를 보이며 GFLOPs 감소



Watch Only Once: An End-to-End Video Action Detection Framework

- **Experiments**:
  - **FPN 사용에 따른 결과**
    - Actor localization을 위해 거의 cost-free로 feature 추출
    - Lightweight FPN을 채택했을 때, 그렇지 않은 접근법보다 좋은 성능을 달성하면서 더 낮은 GFLOPs 달성
    - 결과적으로, 제안한 방법으로 성능 향상 및 계산량 감소

Model	FPN	Person Detector			AVA			GFLOPs
		AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}	
WOO	✗	74.4	95.3	85.4	14.8	20.8	16.8	90.3
WOO	key	**75.6**	**95.6**	**87.1**	**15.3**	**21.3**	**17.5**	**68.0**
SlowFast	✗				**14.7**	**20.3**	**16.6**	136.9
SlowFast	TP	-	95.5	-	13.8	19.2	15.6	120.4
SlowFast	key				13.7	19.1	15.6	120.4

Watch Only Once: An End-to-End Video Action Detection Framework

- **Experiments**:
  - **Spatial-Temporal Fusion**
    - Action classification에서 Spatial-Temporal feature를 합치는 방법에 대한 연구 진행
    - Cross-Attention 방식이 가장 높은 성능과 낮은 GFLOPs 달성

Fusion	AP	AP_{50}	AP_{75}	GFLOPs
sum	14.8	20.5	16.9	68.0
concat	14.7	20.5	17.0	68.1
CA	**15.4**	**21.3**	**17.7**	**68.0**

Watch Only Once: An End-to-End Video Action Detection Framework

- **Conclusion**:
  - Video Action Detection을 위한 간단한 End-to-End 방법인 WOO 제안
  - 단일 통합 백본(Single Unified Backbone)을 포함

    → Actor Localization, Action Classification을 위한 Task별 Feature 제공
  - Video clip 주어지면, 모델이 Bounding box와 Action class를 직접 예측
  - 2개의 Video Action Detection Benchmark에 제안된 방법을 검증하고 Higher mAP, Lower GFLOPs 달성

    → SOTA 달성
  - 독립적인 Person Detector Model과 Post-processing을 적용하지 않는다는 점에서 상당히 흥미로운 방법론