Frame Mix-Up for Long-Term Temporal Context in Video Action Recognition

Dongho LEE *Jinwoo CHOI
Kyunghee University

요 약

현재 Action classification model은 computational resources의 제약으로 인해 video전체의 frame으로 학습하지 못한다. Model에 따라 다르지만, 대부분의 경우 하나의 action을 학습시키기 위해 보통 많게는 32frame, 적게는 8frame으로 model을 학습시킨다. 본 논문에서는 이 한계를 극복하기 위해 하나의 video의 많은 frame들을 mix-up과정을 거쳐 한장의 frame에 여러장의 frame 정보를 담고자 한다. 이 과정에서 video의 시간에 따른 변화(temporal- dynamics)를 손상시키지 않기 위해 linear mix-up이라는 방법을 제안하고 그 성능을 증명하며, 여러장의 frame을 mix-up시켜 모델의 성능을 향상시키는 가능성에 대해 논하고자 한다.

1. 서론

최근 video action classification model들은 많은 발전을 이뤄왔다. UCF101 challenge의 SOTA model인 SMART는 98.64%의 top 1 accuracy를 달성 하였고, Kinetics400 challenge의 SOTA model인 MTV-H는 89.9% top 1 accuracy를 달성하며 action recognition task의 일정 부분 정복한것처럼 보인다. 하지만 위 challenge의 결과를 real world에 바로 적용시키기는 어렵다. 가장 큰 이유는 위 challenge의 dataset frame수가 real world와 달리 매우 적은 frame을 갖는다는 차이 때문이다. UCF101 dataset은 video당 평균 7 초 정도의 길이이고 180 frame정도를 갖고 있으며, Kinetics dataset의 경우 video당 10 초 정도의 길이이다. 하지만 real world의 경우 더 다양한 변수가 존재한다. 첫번째로 위 dataset과 다르게 무용과 같이 더 긴 시간을 갖는 복합적인 action이 있을 수 있다. 두번째로 action에 해당되는 구간만을 잘라내는 과정이 없는 real world dataset은 action과 상관없는 frame들이 존재 할 수 있기 때문에 더 긴 시간을 갖는다.

위에서 말한 한계점을 생각해 봤을때, 현재의 action classification model에 더 많은 frame을 학습하는 방법은 real world에 적용시키기 위한 필수 과제라고 볼 수 있다. 하지만 현재 대부분의 model은 training시 computational resource의 한계로 한정된 frame만을 학습에 사용 할 수 있다. 따라서 sub-sampling 방법을 사용하는데, 이 경우 학습에 사용되는 frame의 수는 많게는 32frame, 적게는 8frame 정도를 학습에 사용한다. 이러한 sub-sampling method는 frame을 random 하게 pick하기 때문에 action을 이해하는데 중요한 frame이 소실될 수 있다는 가능성을 갖는다. 예를들어 3 단 점프 action을 model에게 학습시키는 상황을 가정해보자. 3 단 점프 video가 갖는 frame이 200frame 정도라고 할때, 16 frame을 학습에 사용한다면 전체 frame의 8%만을 사용하는 것이다. 이는 중요한 frame을 놓칠 확률이 매우 높다는 얘기와도 같다. 3 단 점프라는 action을 학습하기 위해 시전자가 3 번을 뛰는 frame은 반드시 필요하다.

본 논문에서는 이러한 한계를 극복하기 위해 Frame mix-up이라는 새로운 방법을 제안한다. Frame mix-up은 TSN[1] model을 base로 하여 dataset을 load 하는 pipe line 과정에서 여러장의 frame tensor를 element-wise addition하여 mix-up[2] 해준다. TSN network에 사용할 training set을 data load pipeline을 거치면 아래와 같은 dimension을 갖는다. $\boldsymbol{X} \in R^{!"\#\$"\%"\&}$ 여기서 N은 batch size, T는 segment(frame)의 수, C는 channel, H, W는 image의 resolution을 나타낸다. 여기서 우리가 제안하는 방법은 data load pipeline 과정 중, 모든 frame을 load한 후 segment의 수 만큼 frame을 element-wise addition 한다. (See Figure 1).

$$\boldsymbol{X} \in R^{!"\#\$\%\&'\#()*\%"+","-} \text{ Þ } \boldsymbol{X} \in R^{!"./01/23.\%"+","-}$$

Network를 통과하기 전 frame element-wise addition 해주는 연산은 network parameter를 필요로 하지 않기 때문에 모델의 복잡도를 증가시키지 않으며, 한장의 frame에 더 많은 정보를 담을 수 있다. 하지만, 모든 frame을 element-wise addition 하게 된다면 RGB값의 범위(0~255)를 초과하게 된다. 이를 해결하기 위해 각 frame에 linearly coefficient를 곱해주어 RGB 범위를 맞춤과 동시에 frame 순서에 맞춰 coefficient 값을 증가시켜 모델에게 frame 순서를 학습시켜 성능을 향상시키고자 했고 실험을 통해 이를 증명했다.

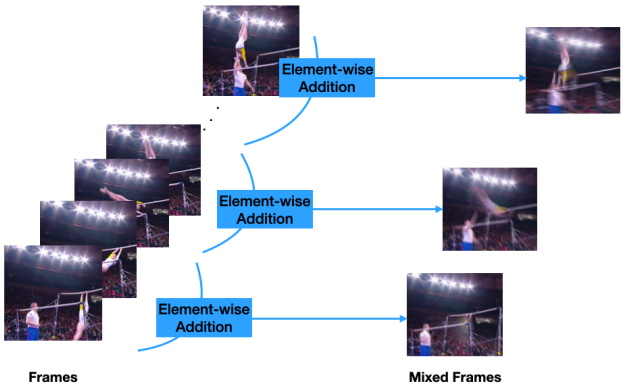

Figure1 : Video가 갖는 총 frame의 수를 F라 할 때, network에 통과 시켜줄 segment의 수를 목표로 각각의 frame을 element-wise addition 해준다. 그림처럼 mixed된 frame은 움직임이 있는 부분은 blur처리한 잔상처럼 mix되는 모습을 볼 수 있다.

2. Frame Mix-up 방법

이번 절에서는 Frame Mix-up method에 대해 다룬다. 앞서 서론에서 표현한 notation은 복잡하여 이해를 방해할 수 있으므로 더 간략하게 표현 하면, $V = \{v_1, v_2, \cdots v_F\}$ 라고 표현하겠다. V 는 하나의 video를 뜻하고, F 는 total frame 수를 표현한다. S 는 network를 통과할 segment수를 의미하고 $S = \{s_1, s_2, \cdots . . s_{++}\}$ 각 segment는 이렇게 표현한다. Total frame number를 segment수로 나눈 값을 anchor value a = F/s 라고 정의한다. Frame을 mix하여 segment를 만드는 과정을 수식으로 표현하면,

$$\text{mixedframe } s_j = \sum_{j=0}^{s} \sum_{i=1+j*a}^{a+(a*j)} v_i \qquad (1)$$

각 frame을 위 수식과 같이 element-wise addition 하게 된다면 frame의 pixel값이 RGB값의 범위를 벗어나게 된다. 따라서 우리는 단순히 frame들을 더해 나가는 방법만으로는 frame을 mix 할 수 없다. 이런 문제를 해결하기 위해 우리는 2 가지 접근법을 제안한다. 첫번째로는 frame의 수 만큼 나눠주는 방법을 average mix method라 부르고 아래 수식과 같이 제안한다.

$$\text{mixedframe } s_j = \frac{1}{s} \times \sum_{j=0}^{s} \sum_{i=1+j*a}^{a+(a*j)} v_i \qquad (2)$$

이러한 방법을 사용하게 된다면 RGB range는 만족하게 되지만, 각 frame이 갖고있는 순서는 모두 무시된다. Video action recognition task에서 중요한 시간적 변화를 model에 전달하지 못한다. 이런 한계를 극복하기 위해 우리가 최종적으로 제안하는 방법은 linear mix method라 부르고 아래와 같다.

$$\text{mixed frame } s_j = \sum_{j=0}^{s} \sum_{i=0+j*a}^{a+(a*j)} \frac{2j}{a(a-1)} \times v_i \qquad (3)$$

Total mixed frame의 합이 1 이 되도록 하며, 동시에 frame의 순서에 따라 계수가 증가하여 model에게 frame의 순서 정보를 전달 해준다. (Figure 2.)

각 linear method와 average method를 비교하여 frame에 linearly order coefficient를 부여 함으로써 모델에게 순서를 학습시킬 수 있다는 것을 성능 향상으로 증명한다.(Table 2.)

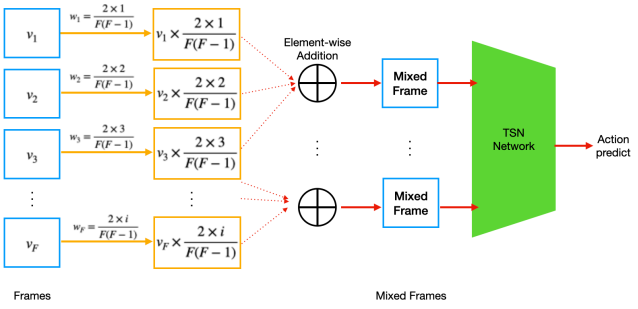

Figure 2. frame을 mix하여 network를 통과시켜 action predict하는 과정을 도식화 하였다.

3. 실험 내용

이번 절에서는 Frame mix 방법에 대한 성능을 검증하는 실험을 진행한다. 본 논문에서 제안한 linear method는 각 frame의 순서를 고려한 방법이다. 따라서 우리가 제안한 방법이 실제 frame의 순서를 제대로 학습한지 검증하기 위해 frame의 순서가 중요한 Something-Something-V1(이하 sthv1) dataset을 통해 검증한다. Sthv1 dataset의 label은 어떤 물체를 왼쪽에서 오른쪽으로 민다. 와 같이 video frame의 순서가 매우 중요한 dataset이다 따라서 본 논문에서 제안한 방법이 실제 모델에게 순서를 학습시키는데 도움이 되었는가에 대해 판단하는 dataset으로 적합하다. 우리가 검증하고자 하는 결과는 base model인 TSN과 비교하여 본 논문에서 제안한 linear method를 적용한 TSN model과 비교하여 sthv1 dataset에서의 성능 향상이 일어나는가를 보고자 한다.

공정한 비교를 위해 조작변인인 frame mix method 적용여부를 제외한 나머지 training 조건은 동일하게 비교한다. Training parameter는 모두 50epoch, 0.02 learning rate, 32batch size, 0.5 dropout ratio를 적용하였고 ImageNet pre-trained weight를 사용하였다.

Table 1. TSN base model과 linear method 적용시킨 TSM model의 sthv1 dataset에서 test 결과이다. Base line model과 Ours model 모두 동일한 8segment로 학습시켰으며, Linear method, average method 모두 all frame을 사용하는 조건으로 test한 결과이다.

	Base model (our impl.)	Ours (linear)	Ours (average)
Top 1 acc (%)	17.29	19.17	19.45 (+ 2.16)
Top 5 acc (%)	43.11	47.98	47.02 (+ 3.89)

Table 2. 한장의 frame당 mix되는 frame 수의 변화에 따른 모델의 성능 변화를 나타낸 표 이다.

Model	Segment number	Mixed frame number	Top 1 acc (%)	Top 5 acc (%)
TSN base (our impl.)	8	-	17.29	43.11
Ours (average mix method)	8	all	19.17	47.98
	8	5	19.5	47.27
	8	3	18.76	46.36
Ours (linear mix method)	8	all	19.45	47.02
	8	5	19.99	48.13
	8	3	19.55	47.42

Table 1.의 실험결과를 보면 기존 base model 대비 linear method model이 top 1 accuracy 기준 2.16%정도의 성능 향상이 이뤄졌다. 마찬가지로 average method또한 1.88%의 성능 향상이 이뤄진 것을 확인 할 수 있다. 이 실험 결과를 통해 model에게 mix된 frame을 feed하는 것 만으로도 성능 향상이 이뤄진다는 사실을 알 수 있다. 이 실험 결과를 통해 본 논문에서 제안한 linearly order coefficient는 효과는 0.28%정도의 미미한 성능향상을 가져왔다. 예상과 다른 결과가 발생한 원인을 분석한 결과, 이러한 결과가 발생 원인은 많은 frame을 mix 할 수록 우리가 제시한 (3) 수식의 계수의 차이는 극히 적어지게 되고 model에게 frame 순서를 학습시키기 어려워진다. Sthv1 dataset의 평균 frame수는 약 48frame정도로 본 논문에서 제안한 method를 적용하면 각 frame 마다 계수의 차는 $\frac{2}{48*49}$ (0.00085)의 차이를 갖는다. 이는 각 frame의 순서를 학습시키기에는 극히 작은 값이기 때문에 모델에게 frame의 순서를 제대로 학습시키기 어려웠을 것이라 예상된다.

앞선 Table 1.의 결과를 토대로 linear method의 유효성을 확인하기 위해 각 segment당 mix되는 frame의 수를 변화시켜 실험을 진행하였다. (Table 2.) 각 segment당 mix되는 frame의 수가 줄어들게 되면 각 frame의 linearly order coefficient는 더 큰 차를 갖게 된다. Table 2 의 8segment, 5frame을 기준으로 계수의 차를 살펴보면 $\frac{2}{97}$ (0.0667)의 차를 갖게 된다. 이는 각 frame의 순서를 학습 가능한 수준의 계수로 판단된다. 이는 실험 결과로도 증명 되었다. 8 segment, 5 mixed frame number case를 보자. Top 1 acc의 차이는 약 0.5% 차이가 나타났다 이는 all frame의 0.28%에 비해 증가한 수치이다. 이로 유추해 볼 때 학습 가능한 수준의 유의미한 계수의 차이는 model에게 frame의 순서를 학습시킬 수 있다는 결론으로 해석된다.

두가지 실험을 통해 알 수 있는 사실은 다음과 같다. 첫번째, video action classification model은 더욱 더 많은 frame 정보를 필요로 한다는 것. 두번째, 많은 frame정보를 단순히 늘리는 것이 아닌, 각 frame의 순서의 정보를 model에게 같이 전달해 주는 과정이 필요하다는 것이다.

4. 결론

본 논문에서는 지금껏 시도하지 않은 새로운 방법으로 model에게 더욱 더 많은 frame 정보를 제공하는 방법을 제안했고, 이를 증명하기 위한 실험에서 유의미한 결론 2 가지를 유추해 냈다. 첫번째로 action classification model은 더 많은 frame정보를 필요로 한다는 것과 단순히 많은 frame정보가 아닌 frame의 순서를 포함한 많은 정보를 필요로 한다는 것이다. 본 논문에서 제시한 linear mix method는 단순히 frame의 순서에 따른 계수값의 차이로 model에게 frame의 순서를 학습시키려 했지만, 더 효과적인 방법으로 model에게 순서를 주입 할 수 있다면 더 높은 성능향상이 이뤄질 것이다. 추가로 본 논문에서는 TSN model 만을 base로 실험했지만 최근 base model로 많이 사용되는 TSM[3] model로도 추가 실험이 필요해 보인다. 또한 3D CNN 기반의 I3D[4] model에서도 frame mix 방법이 유효한지에 대한 검증도 필요로 해 보인다.

5. 감사의 글

이 성과는 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No.2022R1F1A1070997)

참고문헌

[1] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In European Conference on Computer Vision(ECCV), 2016.

[2] Hungyi Zhang, Moustapha Cisse, Yann N. Dauphin, David Lopez-Paz. MIT, FAIR. Mixup : BEYOND EMPIRICAL RISK MINIMIZATION. The International Conference on Learning Representations(ICLR), 2018

[3] Ji Lin, Chuang Gan, Song Han. MIT. TSM: Temporal Shift Module for Efficient Video Understanding. IEEE International Conference on Computer Vision(ICCV), 2017

[4] Joao Carreira, Andrew Zisserman. DeepMind, University of Oxford. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In Proceedings of Computer Vision and Pattern Recognition Conference(CVPR), 2017