비디오에서의 다양한 회전 각도와 회전 속도를 사용한 시 공간 자기 지도학습

*김태훈 황원준
아주대학교
*th951113@ajou.ac.kr wjhwang@ajou.ac.kr

Self-Supervised Spatiotemporal Learning For Video Using Variable Rotate Angle And Speed Prediction

*Taehoon Kim Wonjun Hwang
Ajou University

요약

기존에 지도학습 방법은 성능은 좋지만, 학습할 때 비디오 데이터와 정답 라벨이 있어야 한다. 그러나 이러한 데이터의 라벨을 수동으로 붙여줘야 하는 문제점과 그에 필요한 시간과 돈이 크다는 것이다. 이러한 문제점을 해결하기 위한 다양한 방법 중 자기 지도학습(Self-Supervised Learning) 중 하나인 회전 방법을 비디오 데이터에 적용하여 학습하는 연구를 진행하였다. 본 연구에서는 두가지 방법을 제안한다. 먼저 기존의 비디오 데이터를 입력으로 받으면 단순히 비디오 자체를 회전시키는 것이 아닌 입력으로 들어온 비디오의 각각 프레임이 시간이 지나면서 일정한 속도로 회전을 시킨다. 이때의 회전은 총 네 가지 각도[0, 90, 180, 270]를 분류하도록 하는 방법론이다. 두 번째로 비디오의 프레임이 시간이 지나면서 변할 때 프레임 별로 고정된 각도로 회전시키는데 이때 회전하는 속도 네 가지 [1x, 0.5x, 0.25x, 0.125]를 분류하도록 하는 방법론이다. 이와 같은 제안하는 pretext task들을 통해 네트워크를 학습한 뒤, 학습된 모델을 fine tune 시켜 비디오 분류에 대한 실험을 수행 및 결과를 도출하였다.

1. 서론

영상 처리 분야에서 영상을 이용한 영상 인식, 객체 검출, 영상 분할 등을 딥러닝을 통해 학습하여 수행할 때 좋은 결과를 내기 위해서는 지도학습 방법을 사용해서 많은 데이터와 그에 대한 정답 라벨을 통해 학습하여 성능을 측정한다. 그러나 이런 많은 데이터를 위해서는 수집에 많은 시간과 비용이 필요하지만 수집된 데이터가 있어도 그에 대한 정답 라벨을 만들기에는 어렵다는 한계가 있다.

이와 같은 문제를 해결하려는 방법으로 비지도 학습이 있다. 비지도 학습은 기존에 정답 라벨을 통해 사용하는 지도학습방법과는 다르게 데이터의 정답 라벨이 없어도 영상 자체의 특징들을 학습하여

성능을 내는 방법이 있다. 그 외에도 소수의 정답 라벨을 가진 데이터와 다수의 정답 라벨이 없는 데이터 셋을 가지고 다양한 방법을 적용하여 결과를 도출하는 준 지도학습 등이 있다.

이러한 방법 중 비지도 학습에서 하나의 부류인 자기 지도학습이 최근에 크게 주목을 받고 있다. 자기 지도학습은 비지도 학습의 한 가지 종류이므로 학습에 정답 라벨이 없는 데이터 셋을 이용하여 학습한다. 학습 방법은 연구자 자기만의 문제를 만들어서 그 문제를 푸는데 실제 데이터에 부여되는 정답 라벨이 아니라 임의의 정답을 부여하여 그 정답을 맞히도록 한다. 즉 일종의 자신이 만든 문제를 지도학습 처럼 학습을 하는 것이다. 이와 같이 학습을 위해 연구자 정의하는 임의의 문제를 다른 말로는 pretext task라고 한다. 이러한 자기 지도학습 방법의 예시로 이미지를 9개의 패치로 잘라서 그 패치들을 사용자가 정한 순서대로 섞은 뒤 그 섞은 패치들의 배열을 네트워크의 학습을 통해 맞추는 직소 퍼즐 방법[2]이 있다. 또한, 이미지를 임의의 각도로 돌려 네트워크의 입력으로 넣고 학습을 통해 이미지가 돌린 각도를 맞추는 회전 방법[1] 등등 많은 pretext task들이 존재한다.

그러나 이와 같은 자기 지도학습(Self-Supervised learning)방법은 이미지 데이터 셋에 대해서는 많은 연구가 이루어지고 많은 방법론이 존재하지만, 상대적으로 비디오 데이터에 관해서는 연구가 덜 이루어져 있다.

그림 1 회전을 비디오 데이터에 제안한 방법론

그래서 본 연구는 자기 지도학습 방법을 비디오 데이터에 적용하여 비디오에 알맞은 pretext task에 대해 모색해보고 학습된 네트워크를 통해 비디오 분류(classification) 성능 결과를 도출하였다.

2. 자기 지도학습 방법

자기 지도 학습(Self-Supervised learning)이란 정답이 없는 데이터(영상)들에 대해서 데이터가 표현하는 특성(representation) 등을 통해 임의로 정답 라벨을 부여한 뒤 일반적으로 네트워크를 학습하는 것과 동일하게 수행하는 것이다.

자기 지도학습을 사용하는 경우의 학습방식을 보면 다음과 같다.

1. Pretext Task 수행을 통해 임의의 정답을 부여한 데이터를 입력으로 하여 네트워크를 학습시킨다.

2. 학습된 네트워크의 weight를 전이시켜서 본래 수행하려는 original task(downstream task)를 수행한다.

이와 같은 자기 지도학습 방법은 다양한 pretext task가 존재한다.

그중 본 논문은 이미지 데이터 셋에서 사용하던 회전[1] 방식을 비디오 데이터 셋에 맞게 변형하여 실험을 하였다. 그 이유는 회전 방법이 이미지 데이터 셋에서 다른 방법들 보다도 좋은 결과[3]를 도출하기도 했고 방법 자체가 다른 방식에 비해서 간단하므로 데이터 학습이 다른 방식에 비해 속도가 빠르기 때문에 이미지 데이터 셋보다 상대적으로 데이터 사이즈가 큰 비디오에서 알맞을 것이기 때문에 회전 방법을 사용하였다.

이와 같은 회전 방식은 이미지 데이터 셋에서 사용방법은 이미지 하나에 대해 4개의 이미지로 복제한 뒤 [0, 90, 180, 270]에서 랜덤하게 선택하고 회전을 시킨다. 그 후 하나의 네트워크에서 회전된 각도인 [0, 90, 180, 270]를 4가지의 클래스로 하여 입력 이미지가 어떤 각도로 회전되어 있는지 예측하는 분류를 수행하는 방식으로 사용된다.

위와 같은 이유와 사용하기 쉬운 방식으로 본 연구에서 이미지에서 성능과 속도가 좋은 회전 방법을 사용하였다. 그러나 연구에서는 본래 회전 방법이 이미지 자체를 회전시키는 방법을 사용하여 입력으로 비디오 자체를 회전시키는 것이 아닌 프레임 단위로 회전시켜 전체 비디오 입력에 얼만큼의 회전이 적용되었는지를 네트워크의 학습을 통해 맞추는 방법과 관련한 pretext task를 두 가지 제안한다.

그림 1에서와같이 2가지 방법으로 pretext task를 구성하였는데 각각 회전 방향과 속도에 대한 것이다. (a)에서는 회전 방향을 다르게 주는데 총 네 가지 [0, 90, 180, 270]의 정답을 만들었다. 네 가지로 나눈 이유는 회전 방식을 이미지 데이터에서 사용할 때 이와 같은 4가지의 각도를 주는 것이 가장 좋은 결과를 내었다는 것을 실험적으로 증명되어 있다.[1] 이와 같은 4가지의 각도를 적용할 때 비디오의 프레임 단위로 적용을 하게 되는데 예를 들어 90도의 정답을 데이터 적용하고자 할 때 첫 프레임은 기존의 프레임 방향을 가지고 그다음 프레임에는 90도를 회전한다. 그럼 3번째 프레임은 기존의 90도에서 한 번도 회전한 180도로 회전시키게 되고 나머지 프레임에 대해서도 이와 같은 방법을 사용하여 회전한다. 그러면 회전시킨 video는 입력으로 네트워크에 집어넣어 학습시키게 되는데 매 프레임마다 변화하는 각도가 무엇인지를 맞추는 pretext task이다. 이때 회전 속도는 고정한다.

(b)에서는 회전 속도를 다르게 주는 것이다. 예를 들어 회전은 90도로 고정시켜 놓고 회전 속도를 [1x, 0.5x, 0.25x, 0.125x] 네 가지로 두어 네트워크 학습을 통해 회전되는 속도를 맞추게 하는 방법이다. 그러면 만약 0.5 배속이라면 첫 번째 프레임과 두 번째 프레임은 그대로 기존의 방향으로 유지하고 3번째 프레임과 4번째 프레임은 90도로 회전하고 5번째, 6번째 프레임은 180도로 회전시켜 비디오를 구성하는 것이다. 그러면 속도에 따라 회전시킨 video는 입력으로 네트워크에 집어넣어 학습시키게 되는데 비디오를 global 하게 보면서 회전하는 속도가 무엇인지를 맞추는 pretext task이다. 이때 회전시킬 각도는 고정한다. 기본적으로 입력 데이터로 들어가는 것은 전체 비디오가 아닌 비디오에서 임의의 시작점을 랜덤하게 선택한 뒤 시작점으로부터 연속하는 16 프레임을 추출하여 그 16 프레임에 대해서 랜덤하게 선택된 각도를 적용하게 된다.

3. 실험 설계 및 방법

실험은 UCF 101 dataset[6]에서 수행하였다. UCF 101 데이터 셋은 Center for Research in Computer Vison at University of Central Florida에서 만든 데이터 셋으로 기존의 UCF 50 데이터 셋에서 확장한 것이다. 데이터는 YouTube에서 다운 받은 총 13320개의 비디오를 5개(Human-Object Interaction, Body-Motion Only, Human-Human Interaction, Playing Musical Instruments, Sports.)의 큰 타입으로 분류하고 101가지의 작은 카테고리 behavior

그림 2 실험에 사용한 3D CNN (Convolutional 3D)

들로 정리한 것이다.

네트워크의 경우 3D CNN 모델을 사용하였는데 3D CNN은 2D CNN의 확장으로 2D에서 비디오 데이터와 같이 temporal data를 추가한 것이다. 이러한 3D CNN 중 하나이고 2D CNN 중 VGG 네트워크와 유사한 C3D[4]를 사용하여 실험을 하였다.

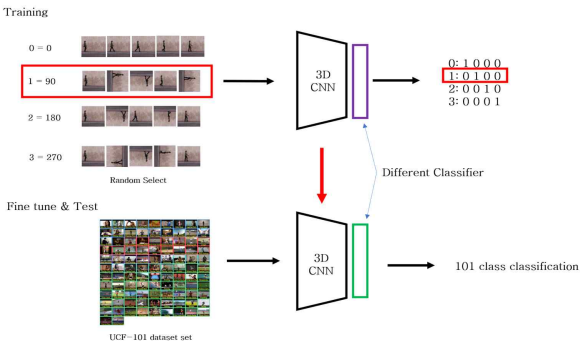

그림 3 전체적인 실험 과정(회전 방향 예시)

실험 방법은 기존의 자기 지도 학습 방식과 유사하게 그림3과 같이 수행하였다.

1) 그림 2의 학습(training) 부분을 보면 먼저 본 논문에서 제시한 pretext task 방법인 회전 방향과 속도 방법을 이용하여 비디오 16 프레임을 바꾼 뒤 입력으로 하여 학습하고 회전 방향 또는 회전 속도를 맞추는 방향으로 역전파한다. 이때의 데이터는 UCF 101의 train set으로

정답 라벨을 사용하지 않고 본 논문에서 pretext task로 사용하는 임의의 답을 부여한다.

2) 그렇게 학습이 끝난 네트워크를 그림2의 fine tune & test 부분과 같이 UCF 101 train set으로 fine tune 시킨 뒤 test set으로 분류 성능을 측정한다.

이때 학습한 3D CNN의 classifier 부분을 떼어 내고 weight를 가져와서 fine tune을 하게 되는데 그 이유는 기존에 학습에서는 pretext task에서 구별해야 하는 클래스가 4개이었는데 (그림3은 회전 방향의 경우임.) fine tune을 할 때에는 101 class를 구별해야하기 때문에 pretext task 학습할 때의 classifier는 fine tune 할 때는 필요없다.

이외의 추가적인 파라미터들의 경우도 기존의 실험[7]의 파라미터를 참고하여 실험하였다.

4. 실험 결과

위의 실험 방법을 통해 나온 실험 결과는 아래의 표들과 같다.

Speed	1x	0.5x	0.25x	0.125x
ACC	64.57	63.7	62.91	**64.76**

표 1 회전 속도에 따른 회전 방향 분류 결과

Angle	90	180	270
ACC	63.47	63.76	**64.71**

표 2 회전 방향에 따른 회전 속도 분류 결과

표 1과 2의 결과는 각각 회전의 방향과 회전의 속도를 맞추는 pretext task를 수행하고 난 뒤 fine tune을 한 결과들이다. 먼저 표 1의 결과는 회전 방향을 맞추는 결과이고 이 회전 속도를 고정으로 방향을 맞추는 방법으로 실험을 하였기 때문에 4가지의 속도 각각에 대해서 정확도를 구한 결과가 된다. 이 결과를 보면 0.125 배속, 즉 8프레임 당 한번 회전하는 각도 [0, 90, 180, 270]을 맞추는 pretext task가 가장 좋은 성능을 보였다. 표 2의 결과는 회전 속도를 맞추는 결과이고 이때 회전 방향을 고정으로 속도를 맞추는 방법으로 실험을 하였기 때문에 4가지 방향에 대해서 정확도를 구한 결과가 된다. 이 결과를 보았을 때는 270도 의 방향, 즉 한번 회전할 때 270도 회전하고 [1x, 0.5x, 0.25x, 0.125x]의 속도를 맞추는 것이 세 가지의 경우 중 가장 높은 결과를 보이는 것을 알 수 있다. 이때 표 2의 고정되는 각도에서 0도가 없는 이유는 0도를 고정하고 회전 속도를 다르게 해도 프레임이 0도로 회전되지 않기 때문에 0도를 고정해서 하는 실험은 수행하지 않았다.

두번째 실험으로 비디오 데이터 셋에서의 자기 지도학습 방법 중에서 하나의 비디오에서 연속되는 프레임으로 이루어진 clip 여러 개를 뽑아내고 그 clip들의 순서를 맞추는 clip ordering [5][7] 방법에 적용해 보는 실험을 하였다. 실험 방법은 먼저 16개의 프레임으로 이루어진 3개의 clip을 전체 비디오의 랜덤한 위치에서 가져온다. 단 동일한 위치에서 시작하는 clip은 없음을 가정하고 먼저 뽑은 clip은 뒤에 뽑은 clip보다 순서가 빠르다. 그 후 본 논문에서 제시한 pretext task인 회전 방향 혹은 속도를 각 clip에 랜덤하게 선택하여 적용한 뒤 세 개의 clip을 섞어서 네트워크를 학습하는 pretext task로 구성하였다. 아래의 표 3과 4는 순서를 맞추는 방법을 적용하여 회전 방향 또는 속도를 맞추는 결과이다.

Speed	1x	0.5x	0.25x	0.125x
ACC	**65.3**	64.18	63.8	65.2

표 3 순서 방법을 추가한 회전 속도에 따른 회전 방향 분류 결과

Angle	90	180	270
ACC	65.48	65.13	**66.02**

표 4 순서 방법을 추가한 회전 방향에 따른 회전 속도 분류 결과

결과를 보면 먼저 표 3의 경우 표 1에서의 결과들과 약 1% 정도의 개선을 보였다. 즉 기존에 회전 속도를 고정하고 회전 방향을 분류하는 것 보다 추가적으로 clip 들 간의 순서를 맞추는 것을 더할 경우 1% 더 잘 답을 맞추도록 학습된다.

표 4의 결과는 표 2의 결과와 비교했을 때 90도와 180도 방향으로 회전을 고정했을 때의 결과는 순서를 맞추는 것(clip ordering method)을 추가했을 때 1% 정도의 차이를 보였다. 270도의 방향으로 회전을 고정했을 경우 기존에 순서를 맞추는 것이 없었을 경우 최대 64.71%이었던 정확도가 순서를 맞추는 방법을 넣었을 때 약 2% 가 증가하는 것을 실험을 통해 결과를 내었다.

Model	C3D	R3D[8]
Random	61.8	54.5
Ordering[5]	65.6	64.9
Ours	**66.02**	**65.43**

표 5 다른 method와 모델별 결과 비교

추가적인 실험에서 제일 결과가 좋았던 270도의 방향으로 회전을 고정했을 경우 기존에 순서를 맞추는 것이 추가된 방법을 다른 모델에서의 결과를 도출하였다. 추가 실험에 사용한 것은 R3D 모델[8]은 2D CNN에서의 Resnet 네트워크를 확장한 것이다. 이에 대한 결과를 Random initialize 한 weight을 가진 3D CNN 모델을 fine tune한 결과와는 5%, Clip Ordering 논문[5]에 있는 결과와 비교했을 때는 1% 정도의 개선을 보인다.

그리고 표 5의 네트워크 학습 결과를 Video Retrieval 결과로 확인하였다. 아래의 그림 4와 표 6은 Video Retrieval 결과를 시각화 및 수치화하여 나타낸 것이다.

그림 4 Video Retrieval Result (Red font = Correct)

Method	Random	Ordering[5]	Ours
Top-1	14.4	12.5	**15.0**
Top-5	24.1	29.0	**29.8**
Top-10	30.7	39.0	**39.3**
Top-20	39.1	**50.6**	50.2
Top-50	51.4	**66.9**	66.2

표 6 Video Retrieval Top -k Accuracy

그림 4와 표 4는 C3D로 학습한 Random weight initialize 모델과 본 논문의 제안한 pretext task 중 가장 성능이 좋은 모델을 가져와서 Video Retrieval을 수행했다. 이때 사용한 데이터 셋은 UCF 101을 사용하였다. Video Retrieval은 dataset에서 test set을 query video로 하여 해당 query 비디오와 가장 유사한 Top -k 개의 비디오를 train set에서 찾는 것이다.

그림4는 query video에 대해 Top - 3에 해당하는 train set에 비디오를 뽑아낸 것이다. 빨간색 글씨로 되어있는 것들은 맞춘 것이고 검은색 글씨로 되어있는 것은 맞추지 못한 것을 의미하며 상대적으로 Random weight initialize 모델보다 본 논문의 pretext task로 Video Retrieval을 했을 때의 결과가 더 좋은 것을 알 수 있다.

표 6은 다양한 Top -k에 대해서 정확도를 나타낸 것인데 Top -1 ~ Top - 50 까지 전반적으로 Random weight initialize 모델보다 최대 15% 차이를 보이는 것을 통해 상대적으로 더 높은 결과를 보이는 것을 알 수 있다. Clip Ordering 방법보다 Top - 1, 5, 10에서는 Proposed Pretext task 가 더 높은 정확도를 보이는데 top -20, 50의 경우는 Ordering 방법을 사용하는 것이 더 높지만 1% 미만의 차이를 보인다.

5. 결론

본 연구에서 자기 지도학습 방법 중 하나인 회전(Rotation)을 기존에 이미지 데이터 셋이 아닌 비디오 데이터 셋에 맞도록 pretext task를 설계하여 UCF 101 비디오 데이터 셋에서 실험하여 결과를 도출하였다.

연구에서는 두 가지의 방법을 제시하였는데 먼저 회전 속도를 고정한 뒤 전체 비디오에서 추출한 임의의 clip (클립 당 16 프레임) 이 [0, 90, 180, 270]의 각도 중 어떤 각도로 프레임 별로 회전하는지를 맞추는 방법의 pretext task를 제안하였고 결과적으로 0.125 배속으로 고정할 때 회전 각도를 맞추는 것이 64.76%으로 가장 좋게 나타났다.

두 번째 방법은 회전 각도를 고정하고 전체 비디오에서 추출한 임의의 clip (clip 당 16 프레임) 이 [1x, 0.5x, 0.25x, 0.125x]의 회전하는 속도 중 어떤 속도로 각도를 회전시키는지 맞추는 방법의 pretext task를 제안하였고 결과적으로 270도의 회전 각도를 고정시키고 회전 속도를 맞추는 것이 64.71%로 가장 좋게 나타났다.

추가적인 실험으로 하나의 비디오에서 시작 지점이 다른 3개의 clip을 뽑아 그 clip 들 간의 순서[5][7]를 맞추는 방법을 본 연구에서 제시하는 pretext task 들과 결합하여 실험 결과를 내보았다. 결과적으로 순서 방법을 사용할 경우, 사용하지 않았을 때보다 1% 정도의 성능 향상이 있었다. 특히 270도의 회전 각도를 고정시키고 회전 속도를 맞추는 경우에는 순서 방법을 적용했을 때 2% 증가한 66.02%의 정확도를 보였다. 마지막으로 다른 3D CNN 모델 중 하나인 R3D에서 결과를 보았을 때 비록 C3D 에서의 성능보다는 낮았지만 pretext task로 학습 하고 fine tune을 했을 때가 Random initialize weight를 사용 했을 때 보다 C3D, R3D 모두 5% 향상되었으며 Clip Ordering[5]의 결과와 비교했을 때는 1% 정도의 개선을 보인다.

향후의 연구에서는 비디오 데이터 셋에서 더 좋은 성능을 낼 수 있는 pretext task에 대하여 연구해 볼 것이다. 추가적으로 UCF 101 이외에 Kinetics 600와 같은 다양한 데이터 셋에서 추가적인 실험을 해볼 예정이다.

감사의 글

"본 연구는 과학기술정보통신부 및 정보통신기획평가원의 SW중심대학사업의 수행결과로 추진되었음"(2015-0-00908)

참고 문헌

[1] S. Gidaris, P. Singh, and N. Komodakis,“Unsupervised representation learning by predictingimage rotations.” In International Conference onLearning Representations (ICLR), 2018.
[2] M. Noroozi and P. Favaro, “Unsupervised learning ofvisual representations by solving jigsaw puzzles.” InEuropean Conference on Computer Vision (ECCV),2016.
[3] Kolesnikov, Alexander, Xiaohua Zhai, and LucasBeyer. "Revisiting self-supervised visualrepresentation learning." arXiv preprintarXiv:1901.09005. 2019.
[4] Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: Proceedings of the IEEE international conference on
computer vision. pp. 4489-4497. 2015.
[5] Xu, D., Xiao, J., Zhao, Z., Shao, J., Xie, D., Zhuang, Y.: Self-supervised spatiotemporal learning via video clip order prediction. In: Proceedings of the IEEE Conference on Computer
Vision and Pattern Recognition. pp. 10334-10343. 2019.
[6] Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from
videos in the wild. arXiv preprint arXiv:1212.0402. 2012.
[7] Cho, Hyeon, et al. "Self-Supervised Spatio-Temporal Representation Learning Using Variable Playback Speed Prediction." *arXiv preprint arXiv:2003.02692* (2020).
[8]Hara, Kensho, Hirokatsu Kataoka, and Yutaka Satoh. "Learning spatio-temporal features with 3D residual networks for action recognition." Proceedings of the IEEE International Conference on Computer Vision Workshops. 2017.