[CVPR] Removing the Background by Adding the Background : Towards Background Robust Self-Supervised Video Representation Learning

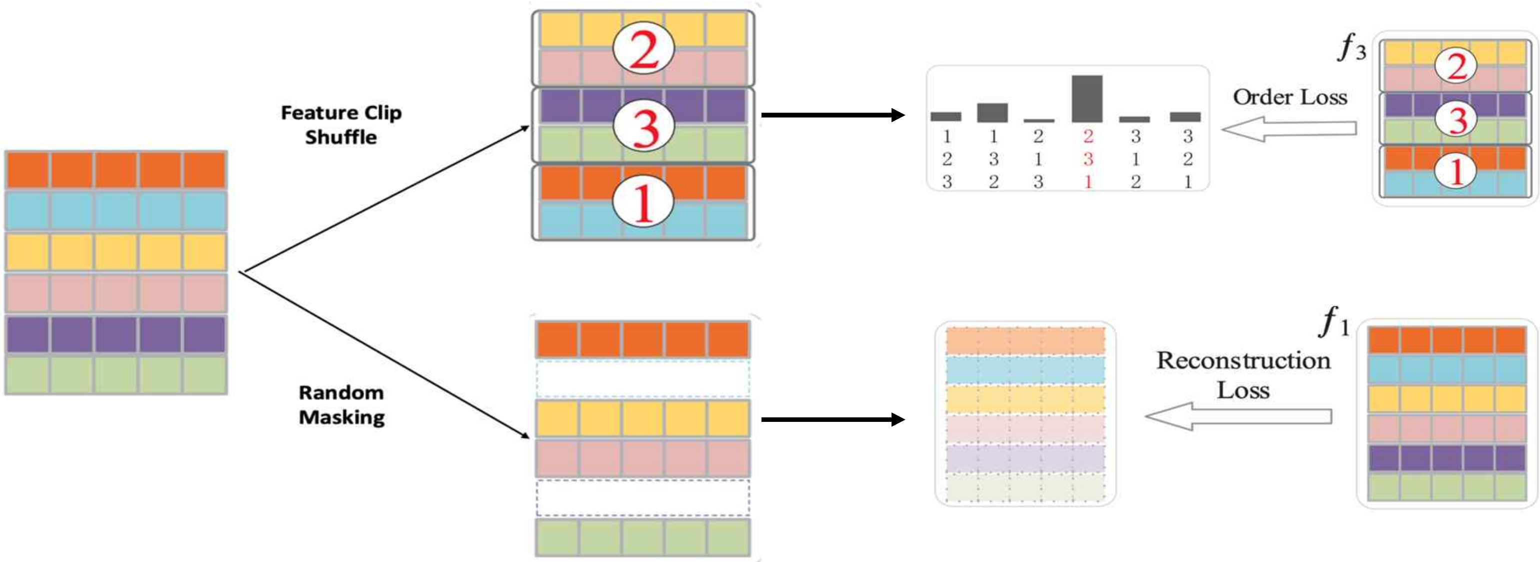
(2021, *Jnpeng Wang et al*)

Sejong RCV – 임근택

Preliminaries

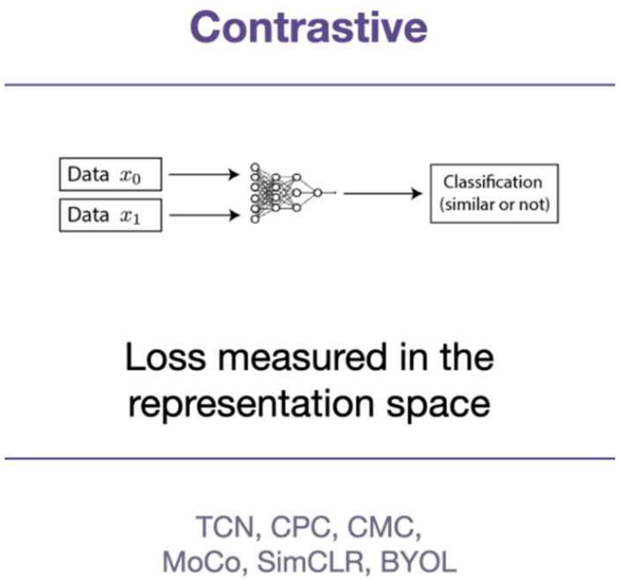
Self Supervised Learning with Pretext task

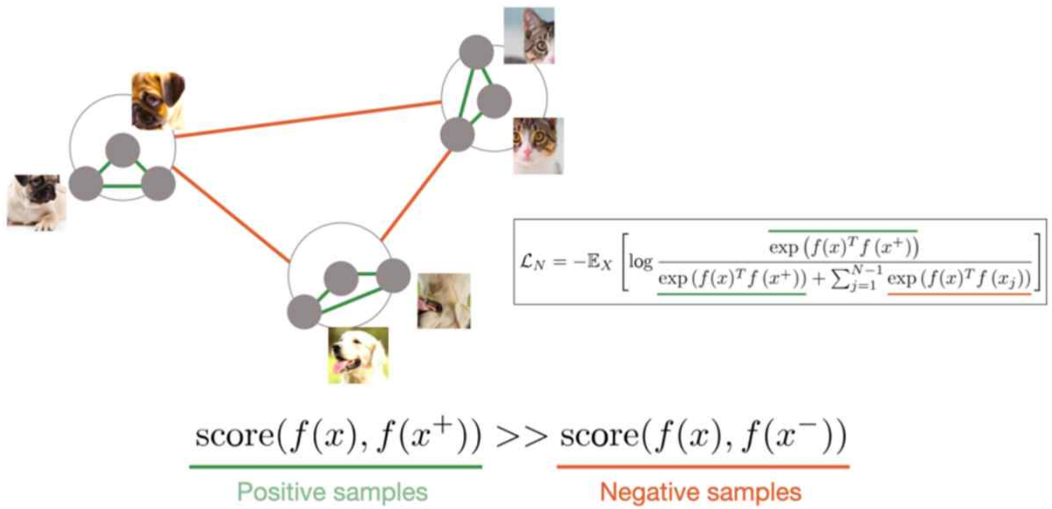
- Pretext tasks are pre-designed tasks for networks to solve, and visual features are learned by learning objective functions of pretext tasks.



Self Supervised Learning with Contrastive Learning

- Another mainstream method is based on contrastive learning, **which regards each instance as a category.**





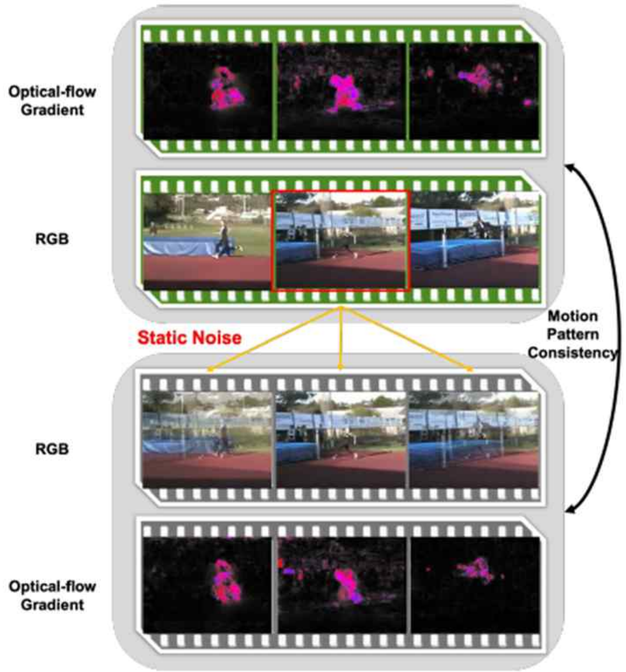
Introduction

Problem Definition

- Video datasets usually exist large **implicit biases** over scene and object structure, **making temporal structure become less important and the prediction tends to have a high dependence on the video background**.

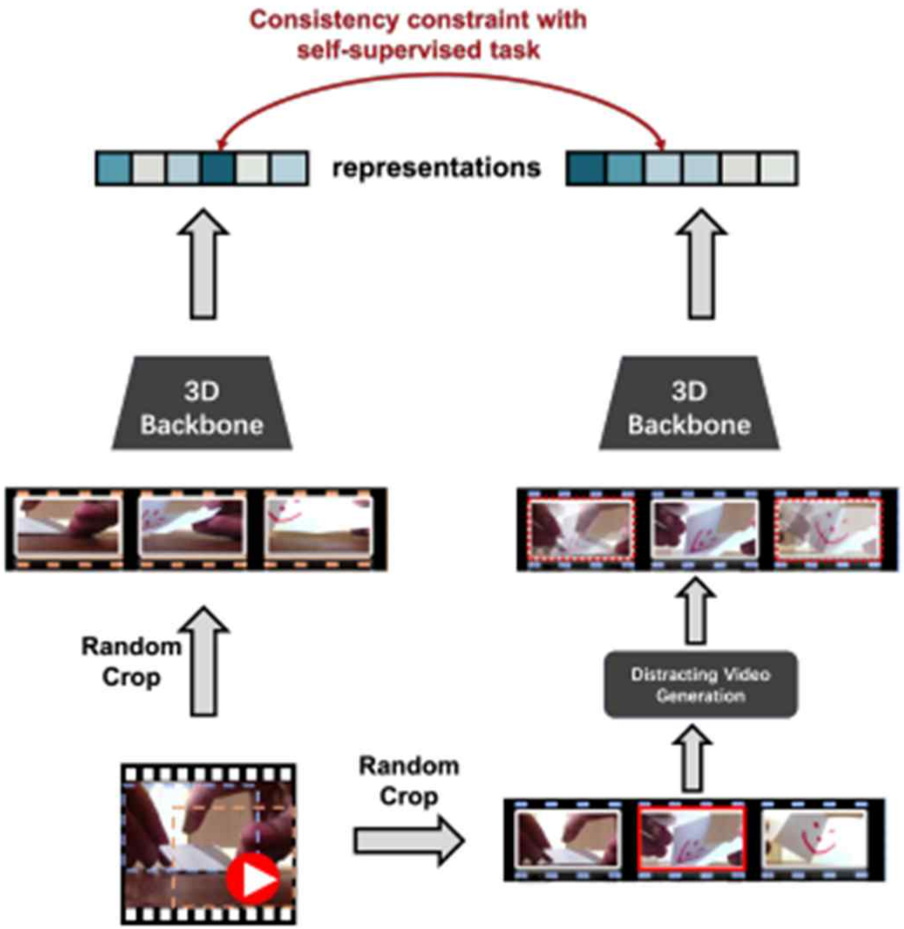
Problem Definition

- One intra-video static frame is randomly selected and added to other frames as Noise.
- However **optical flow gradient is basically not changed, indicating that the motion pattern is retained**



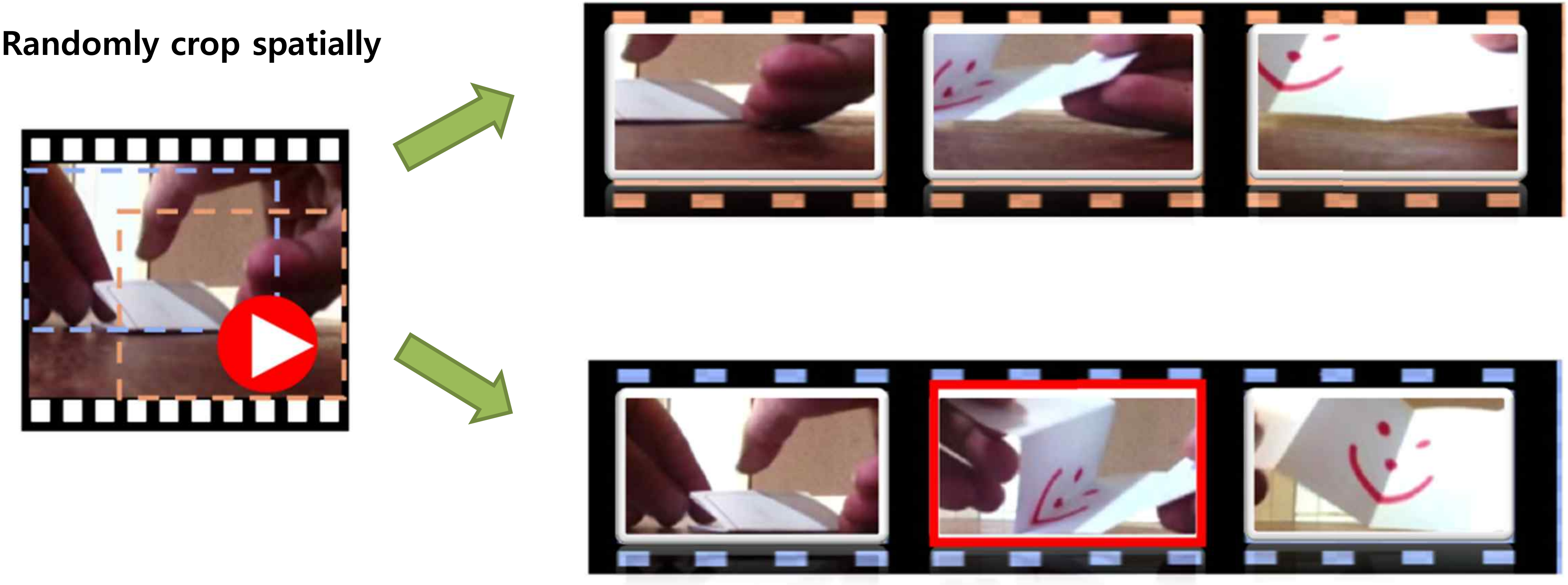
Proposed Method

Architecture



Generating Distracted Video

Randomly crop spatially



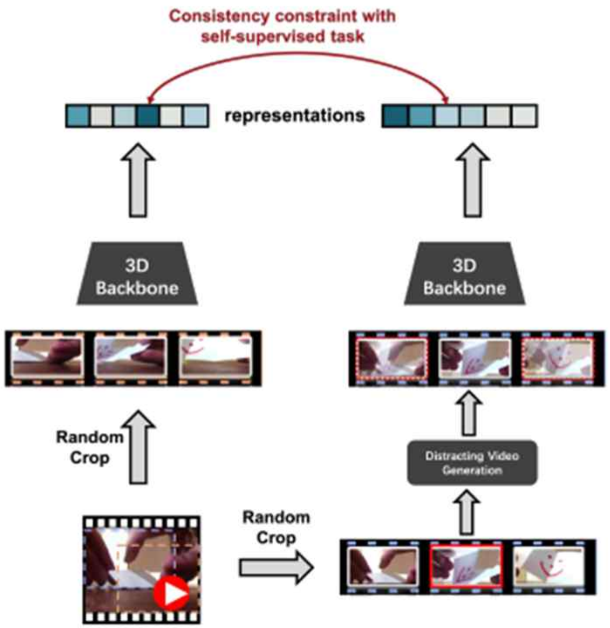
Generating Distracted Video

static frames is added to
other frames as Noise

Architecture

- To mitigate the model reliance towards the background, removing the background impact by adding the background

- The model will be promoted to suppress the background noise, **yielding video representations that are more sensitive to motion changes.**



Plug and Play

Pretext Task

- Most pretext tasks can be formulated as a multi-category classification task and optimized with the cross-entropy loss.

$$\mathcal{L}_p = -\frac{1}{M} \sum_{r \in R} \mathcal{L}_{ce}(F(r(x);\theta), r)$$

$$\mathcal{L}_{be} = ||\psi(f_{x^o}) - \psi(f_{x^d})||^2$$

$$\mathcal{L} = \mathcal{L}_p + \beta \mathcal{L}_{be}$$

Contrastive Learning

- Contrastive learning aims to learn an invariant representation for each sample, which is achieved by maximizing similarity of similar pairs over dissimilar pairs.

  - **Positive Sets** : Same Video, Same Clip
  - **Negative Sets** : Same/Different Video , Different Clip

$$z_x = \phi(f(x))$$

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N} log \frac{exp(z_{x_i^o} \cdot z_{x_i^d})}{exp(z_{x_i^o} \cdot z_{x_i^d}) + \sum_{n \in \mathcal{N}_i} exp(z_{x_i^o} \cdot z_n)}$$

Experiments

Action Recognition

Method			Pretrain					Fine-tune	
Method(year)	**Backbone**	**Depth**	**Dataset(duration)**	**Frame**	**Res**	**Single-Mod**	***C/P***	**UCF101**	**HMDB51**
Supervised									
Random Init	I3D	22	✗	-	224	✓	-	60.5	21.2
ImageNet Supervised	I3D	22	ImageNet	-	224	✓	-	67.1	28.5
K400 Supervised	I3D	22	K400(28d)	-	224	✓	-	96.8	74.5
Self-supervised									
Shuffle [34] [ECCV, 2016]	AlexNet	8	UCF101(1d)	-	112	✓	*P*	50.2	18.1
VGAN [47] [NeurIPS, 2016]	VGAN	22	UCF101(1d)	-	112	✓	*P*	52.1	-
OPN [28] [ICCV, 2017]	Caffe Net	14	UCF101(1d)	-	112	✓	*P*	56.3	22.1
Geometry [12] [CVPR, 2018]	Flow Net	56	UCF101(1d)	16	112	✗	*P*	55.1	23.3
IIC [43] [ACM MM, 2020]	C3D	10	UCF101(1d)	16	112	✗	*C*	72.7	36.8
Pace [50] [ECCV, 2020]	R(2+1)D	23	K400(28d)	16	112	✓	*C*	77.1	36.6
3D RotNet [23] [2018]	C3D	10	K400(28d)	16	112	✓	*P*	62.9	33.7
3D RotNet + BE	C3D	10	K400(28d)	16	112	✓	*P*	**65.4**(2.5↑)	**37.4**(3.7↑)
ST Puzzles [26] [AAAI, 2019]	C3D	10	UCF101(1d)	48	112	✓	*P*	60.6	28.3
ST Puzzles + BE	C3D	10	UCF101(1d)	48	112	✓	*P*	**63.7**(3.1↑)	**30.8**(2.5↑)
Clip Order [57] [CVPR, 2019]	C3D	10	UCF101(1d)	64	112	✓	*P*	65.6	28.4
Clip Order + BE	C3D	10	UCF101(1d)	64	112	✓	*P*	**68.5**(2.9↑)	**32.8**(4.4↑)
MoCo [21] [CVPR, 2020]◊	C3D	10	UCF101(1d)	16	112	✓	*C*	60.5	27.2
MoCo + BE	C3D	10	UCF101(1d)	16	112	✓	*C*	**72.4**(11.9↑)	**42.3**(14.1↑)
CoCLR[19] [NeuIPS, 2020]	R3D	23	K400(28d)	32	128	✗	*C*	87.9	54.6
DPC [17][ICCW, 2019]	R3D	34	K400(28d)	64	224	✓	*P*	75.7	35.7
AoT [54] [CVPR, 2018]	T-CAM	-	K400(28d)	64	224	✓	*P*	79.4	-
Pace [50] [ECCV, 2020]	S3D-G	23	K400(28d)	64	224	✓	*C*	87.1	52.6
SpeedNet [1] [CVPR, 2020]	S3D-G	23	K400(28d)	64	224	✓	*P*	81.1	48.8
SpeedNet [1] [CVPR, 2020]	I3D	22	K400(28d)	64	224	✓	*P*	66.7	43.7
MoCo [21] [CVPR, 2020]◊	I3D	22	K400(28d)	16	224	✓	*C*	70.4	36.3
MoCo + BE	I3D	22	K400(28d)	16	224	✓	*C*	**86.8**(16.4↑)	**55.4**(19.1↑)
MoCo + BE	I3D	22	UCF101(1d)	16	224	✓	*C*	82.4	52.9
MoCo + BE	R3D	34	UCF101(1d)	16	224	✓	*C*	83.4	53.7
MoCo + BE	R3D	34	K400(28d)	16	224	✓	*C*	87.1	56.2

Action Recognition

Method	Pretrain	Single-Mod	Diving48
Supervised Learning			
R(2+1)D [46][CVPR, 2018]	✗	✓	21.4
R(2+1)D [46] [CVPR, 2018]	Sports1M	✓	28.9
I3D[7]◊[CVPR, 2017]	ImageNet	✓	20.5
I3D[7]◊[CVPR, 2017]	K400	✓	27.4
TRN [64] [ECCV, 2018]	ImageNet	✗	22.8
DIMOFS [2] [2018]	K400+Track	✗	31.4
GST [31] [ICCV, 2019]	ImageNet	✓	38.8
Att-LSTM [24] [CVPRW, 2019]	ImageNet	✓	35.6
GSM [42] [CVPR, 2020]	ImageNet	✓	40.3
CorrNet [48] [CVPR, 2020]	Sports1M	✓	44.7
Self-supervised Learning			
MoCo + BE (I3D)	Diving48	✓	58.3
MoCo + BE (R3D-18)	UCF101	✓	46.6
MoCo [21] ◊ (I3D)	UCF101	✓	43.2
MoCo + BE (I3D)	UCF101	✓	**58.8**(15.6↑)
MoCo [21] ◊ (I3D)	K400	✓	47.9
MoCo + BE (I3D)	K400	✓	**62.4**(14.5↑)

Table 2: Top-1 accuracy (%) of integrating BE into MoCo and compared to previous method on Diving48.

Video Retrieval

Method	Net	1	5	10	20	50
Clip Order [57]	C3D	7.4	22.6	34.4	48.5	70.1
Clip Order [57]	R3D	7.6	22.9	34.4	48.8	68.9
VCP [32]	C3D	7.8	23.8	35.3	49.3	71.6
MemDPC [18]	R3D	7.7	25.7	40.6	57.7	-
Pace [50]	R3D	9.6	26.9	41.1	56.1	76.5
MoCo [21] ◊	C3D	9.5	25.4	38.3	52.2	72.4
MoCo + BE	C3D	10.2	27.6	40.5	56.2	76.6
MoCo + BE	I3D	9.3	28.8	41.4	57.9	78.5
MoCo + BE	R3D	**11.9**	**31.3**	**44.5**	**60.5**	**81.4**

Table 3: **Recall-at-topK (%).** Accuracy under different K values on HMDB51.

Variants of Distracting Video Generation

Method	UCF101	HMDB51
baseline	72.7	42.1
Gaussian Noise	73.2(0.5↑)	42.4(0.3↑)
Video Mixup	68.3(4.4↓)	38.1(4.0↓)
Video CutMix	71.2(1.5↓)	40.5(1.6↓)
Inter-Video Frame	77.4(4.7↑)	46.5(4.4↑)
Intra-Video Frame	**82.4**(9.7↑)	**52.9** (10.8↑)

Table 4: Top-1 accuracy (%) of different distracting video generation methods on UCF101 and HMDB51.

Is Background Really Removed?

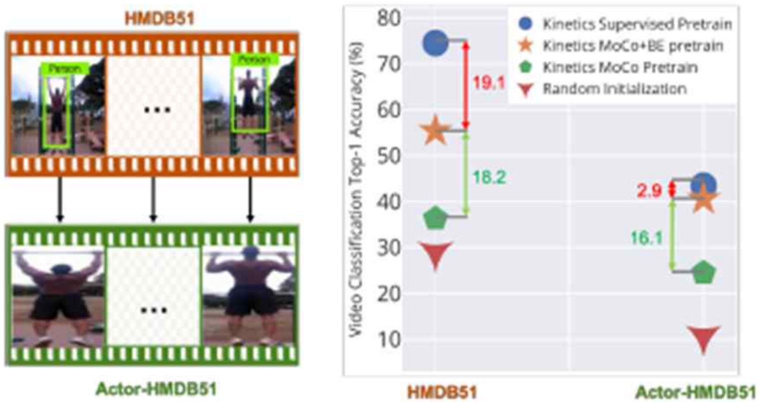


Figure 5: Fine-tuning on the actor dominated dataset actor-HMDB51, our method is very close to the result of Kinetics fully supervised, with only 2.9% difference. Meanwhile the improvement brought by BE over MoCo baseline has only a small drop compared to HMDB51, from 18.2% to 16.1%.

Visualization Analysis

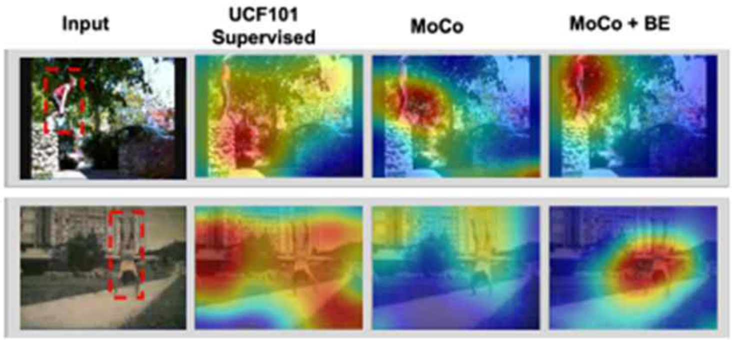


Figure 6: **Generalization ability on novel classes.** Supervised model is severely affected by the scene bias, while after pre-training with MoCo+BE, the model can precisely focus more on moving areas.