LEARNING SPATIO-TEMPORAL FEATURES WITH **3D RESIDUAL NETWORKS** FOR **ACTION RECOGNITION**

Kensho Hara, Hirokatsu Kataoka, Yutaka Satoh | National Institute of Advanced Industrial Science and Technology (AIST)

INTRODUCTION

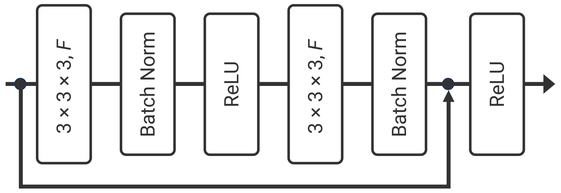
- Performance of 3D CNNs for action recognition is greatly improved using large-scale video datasets.
- Conventional architectures of 3D CNNs are relatively shallow compared with the 2D ones.
- 2D Residual Networks (ResNets) are successful architectures in various tasks.

Exploring 3D ResNets for action recognition

METHOD

Extending 2D conv filters to 3D ones

- Residual Block $B(F) = \begin{bmatrix} 3 \times 3 \times 3, F \\ 3 \times 3 \times 3, F \end{bmatrix}$



Training models using SGD

- Initial learning rate: 0.1
- Momentum: 0.9
- Spatial cropping from 4 corners and 1 center
- Temporal random cropping

Layer	3D ResNet-18	3D ResNet-34
conv1	7 × 7 × 7, 64, stride 1 (T), 2 (XY)	
conv2_x	B (64) × 2	B (64) × 3
conv3_x	B (128) × 2	B (128) × 4
conv4_x	B (256) × 2	B (256) × 6
conv5_x	B (512) × 2	B (512) × 3
	average pool, 400-d fc, softmax	

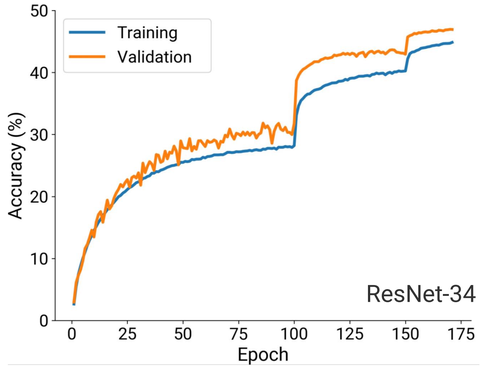
EXPERIMENTS

- Using the Kinetics dataset
  - 400 classes
  - 300k action instances
    - Train, val, test: 240k, 20k, 40k
- Training the 3D ResNets for 1.5 weeks using NVIDIA TITAN X × 2
- Testing on the validation set

Method	Input Size (XY, T)	Accuracy [%] Top-1	Top-5	Avg
3D ResNet-18	112, 16	54.6	78.8	66.7
3D ResNet-34	112, 16	59.9	82.6	**71.3**
C3D*	112, 16	55.6	79.1	67.4
RGB-I3D w/o ImageNet**	224, 64	68.4	88.0	78.2

* pre-trained on the Sports-1M dataset ** testing on the test set

Outperforming C3D without pretraining



Achieving good accuracy without overfitting

CONCLUSIONS & FUTURE WORK

- **Deep 3D ResNets** trained on the **Kinetics** dataset achieved **better performance** compared with shallow ones.
- Future work is **exploration of deeper architectures**.

GITHUB | CODE & PRETRAINED MODELS

https://goo.gl/tBn5yM
PyTorch ver.

https://goo.gl/kA71M8
Torch ver.

- Training and testing 3D ResNets and C3D
- **Classifying videos** and **extracting features** of them using **pretrained models**