Supplementary Material for
Spatio-temporal Filter Analysis Improves 3D-CNN For Action Classification

Takumi Kobayashi, Jiaxing Ye
National Institute of Advanced Industrial Science and Technology
1-1-1 Umezono, Tsukuba, Japan
{takumi.kobayashi,jiaxing.you}@aist.go.jp

A. Analyzing temporal filters

In Sec. 2.1, we show the distribution of temporal filters which are $L_2$-normalized 3-d vectors thus being distributed on a sphere. For ease of analyzing the distribution, we construct Cartesian coordinates composed of physically interpretable axes as shown in Figure A: the average vector $\alpha [1,1,1]^\top$, the 1st differential vector $\alpha [-1,0,+1]^\top$ and the 2nd differential vector $\alpha [-1,+2,-1]^\top$. In the main manuscript, Figure 1 shows the distribution by projecting the temporal-filter samples from a sphere into a plane depicted by gray color in Figure A. In order to visualize further details of the temporal filter distributions, Figure B shows the distributions on two types of planes which are perpendicular to each other; one is spanned by the average and 1st-differential vectors, and the other is by the 1st- and 2nd-differential vectors. Note that as the signs of temporal filters can be arbitrarily given in SVD (1), we plot both the temporal filter $u_i$ and its opposite one $-u_i$ for describing the distribution in Figure 1&B. The visualization in Figure B further supports our findings discussed in Sec. 2.1.

B. Detailed procedure to train models

We detail the training and evaluation procedure on respective datasets in Table A. To train 3D-CNNs, we apply SGD optimizer with momentum of 0.9, weight decay of 0.0005 and the other parameters shown in Table A to video sub-clips of $32 \times 224 \times 224$ sampled by random cropping in a spatio-temporal domain. For evaluation, we extract several clips from an input video sequence at fixed positions. Video sub-clips of $32 \times 256 \times 256$ are uniformly sampled in the spatio-temporal domain with the numbers of clips shown in Table A to cover whole a video volume. The classification scores are summed up across those clips to produce the final classification.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Training samples</td>
<td>168,913</td>
<td>81,663</td>
<td>241,181</td>
<td>121,802</td>
</tr>
<tr>
<td>Test samples</td>
<td>24,777</td>
<td>11,799</td>
<td>19,877</td>
<td>9,934</td>
</tr>
<tr>
<td>Classes</td>
<td>174</td>
<td>87</td>
<td>400</td>
<td>200</td>
</tr>
<tr>
<td>batch size</td>
<td>32</td>
<td>24</td>
<td>32</td>
<td>32</td>
</tr>
<tr>
<td>initial learning rate</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>learning rate schedule</td>
<td>cosine decay</td>
<td>×0.1 at 15, 30-epochs</td>
<td>cosine decay</td>
<td>×0.1 at 15, 30-epochs</td>
</tr>
<tr>
<td>training epochs</td>
<td>100</td>
<td>35</td>
<td>100</td>
<td>45</td>
</tr>
</tbody>
</table>

Evaluation clips: $3^{\text{(spatial)}} \times 3^{\text{(temporal)}} = 9$ clips

$3^{\text{(spatial)}} \times 10^{\text{(temporal)}} = 30$ clips
Figure B. Distributions of the primary temporal filters embedded in I3D-ResNet-50 which is pretrained on (a) SSv2 [1] and (b) K-400 [2] datasets. The temporal filters are normalized in unit $L_2$ norm to distribute on a sphere (Figure A).
C. Effective receptive field

Following [3], we measure the effective receptive field of a 3D-CNN as follows.

1. Randomly draw an input volume by $I = \{I_{cthw} \sim N(0, 1)\}_{c,t,h,w}^{3,32,224,224} \in \mathbb{R}^{3\times32\times224\times224}$.

2. Inject gradients to the center neuron on the last feature map. Let $X \in \mathbb{R}^{2048\times32\times7\times7}$ be the last feature map produced by I3D-ResNet-50 and $W \in \mathbb{R}^{2048\times32\times7\times7}$ be a binary map which activates at the center neuron; $W_{cthw} = 1$ for $(t, h, w) = (17, 4, 4)$, $\forall c$ and $W_{cthw} = 0$ for the others. Thereby, we can design a loss of $\ell = \langle W, X \rangle$, the element-wise multiplication and summation, i.e., inner-product between tensors.

3. The gradients $G_I$ on an input $I$ are computed through back-propagation based on the loss $\ell$.

4. Repeat the above three steps 10 times and average the gradient values to measure the effective receptive field $\bar{G} = \left[\mathbb{E}_I(G_I^2)\right]^{\frac{1}{2}}$ where square and square-root operates on tensors in an element-wise manner.

References

