Supplementary Materials to End-to-End Learning of Motion Representation for Video Understanding

Lijie Fan*[2], Wenbing Huang*[1], Chuang Gan[3], Stefano Ermon[4], Boqing Gong[1], Junzhou Huang[1]
[1] Tencent AI Lab
[2] Tsinghua University, Beijing, China
[3] MIT-Watson Lab
[4] Department of Computer Science, Stanford University
flj14@mails.tsinghua.edu.cn, hwenbing@126.com, ganchuang1990@gmail.com
ermon@cs.stanford.edu, boqinggo@outlook.com, jzhuang@uta.edu

This supplementary material provides the visualization of the multi-scale TV-L1 in Figure 1 and introduces more details for the gradient computations of Eq. (11). Besides, we present more visualization results as a complement of Figure 4 to illustrate the action features learned by our TVNets. We also provide additional experimental evaluations on the action similarity labeling task to verify the effectiveness of the TVNet.

1. Gradients Computation

We repeat Eq. (11) here for convenience, *i.e.*,

$$\boldsymbol{p'}_d = \frac{\boldsymbol{p}_d + \tau/\theta \nabla \boldsymbol{u}_d}{1 + \tau/\theta \sqrt{\nabla \boldsymbol{u}_{d1}^2 + \nabla \boldsymbol{u}_{d2}^2 + \varepsilon}}, \tag{1}$$

where we have denoted the output as $\boldsymbol{p'}_d$ to distinguish it with the input $\boldsymbol{p}_d$. Calculating the gradient with respect to $\boldsymbol{u}_{d1}$ gives

$$\begin{aligned}
\frac{\partial}{\partial \nabla \boldsymbol{u}_{d1}} \boldsymbol{p'}_d &= \frac{\tau}{\theta} \frac{(1 + \tau/\theta \sqrt{\nabla \boldsymbol{u}_{d1}^2 + \nabla \boldsymbol{u}_{d2}^2 + \varepsilon}) - \nabla \boldsymbol{u}_{d1} (\boldsymbol{p}_d + \tau/\theta \nabla \boldsymbol{u}_d)(\nabla \boldsymbol{u}_{d1}^2 + \nabla \boldsymbol{u}_{d2}^2 + \varepsilon)^{-1/2}}{(1 + \tau/\theta \sqrt{\nabla \boldsymbol{u}_{d1}^2 + \nabla \boldsymbol{u}_{d2}^2 + \varepsilon})^2}, \\
&= \frac{\tau/\theta}{1 + \tau/\theta \sqrt{\nabla \boldsymbol{u}_{d1}^2 + \nabla \boldsymbol{u}_{d2}^2 + \varepsilon}} - \frac{\tau/\theta \nabla \boldsymbol{u}_{d1} (\boldsymbol{p}_d + \tau/\theta \nabla \boldsymbol{u}_d)}{(1 + \tau/\theta \sqrt{\nabla \boldsymbol{u}_{d1}^2 + \nabla \boldsymbol{u}_{d2}^2 + \varepsilon})^2} \frac{1}{\sqrt{\nabla \boldsymbol{u}_{d1}^2 + \nabla \boldsymbol{u}_{d2}^2 + \varepsilon}}, \\
&= \boldsymbol{a} - \frac{\boldsymbol{b}}{\sqrt{\nabla \boldsymbol{u}_{d1}^2 + \nabla \boldsymbol{u}_{d2}^2 + \varepsilon}},
\end{aligned} \tag{2}$$

where $\boldsymbol{a} = \frac{\tau/\theta}{1+\tau/\theta \sqrt{\nabla \boldsymbol{u}_{d1}^2 + \nabla \boldsymbol{u}_{d2}^2 + \varepsilon}}$ and $\boldsymbol{b} = \frac{\tau/\theta \nabla \boldsymbol{u}_{d1} (\boldsymbol{p}_d + \tau/\theta \nabla \boldsymbol{u}_d)}{(1+\tau/\theta \sqrt{\nabla \boldsymbol{u}_{d1}^2 + \nabla \boldsymbol{u}_{d2}^2 + \varepsilon})^2}$. Clearly, both $\boldsymbol{a}$ and $\boldsymbol{b}$ are well-defined in the sense that their denominators are never equal to zeros. Similarly, we can compute the gradients with respect to $\boldsymbol{u}_{d2}$.

2. Visualization Results

In the paper, Figure 4 has visualized the learned action features of several samples on the UCF101 dataset. Here, for better illustration, we provide more results in Figure 2. For the visualization of the optical flows as well as the motion representations obtained by our TVNet, we combine the two flow fields (*i.e.* the x-direction and y-direction flows) of each sample and utilize the optical visualization tool provided by MPI Sintel[1]. It is observed that the feature obtained by the trainable version of TVNet-50 is able to capture the outline of the input image while still retaining the movements of the key parts (see the first sample in Figure 2 for example).

*indicates equal contributions. This work was conducted when Lijie Fan was served as a research intern in Tecent AI Lab.
[1]http://sintel.is.tue.mpg.de/downloads

Table 1. Action similarity labeling result. Our method obtains the best classification accuracy and a comparable AUC value compared to C3D.

Methods	Model	Acc.	AUC
C3D	linear	78.3	86.5
Imagenet	linear	67.5	73.8
STIP	linear	60.9	65.3
STIP	metric	64.3	69.1
MIP	metric	65.5	71.9
MIP+STIP+MBH	metric	66.1	73.2
iDT+FV	metric	68.7	75.4
ours	linear	**79.2**	**84.3**

To further reveal the motion patterns within the action features learned by the trainable version of TVNet-50, the last two columns of Figure 2 also visualize the reconstructed images for the action features by using the DeepDraw tool[2]. To be specific, we use BN-Inception architecture for our model training, taking consecutive motion representations from TVNet-50 as inputs, which has 10 channels. In order to make the tool work properly, we make the DeepDraw tool available to inputs with arbitrary number of channels. After the gradient ascent process, we reconstruct the gray-scale images by averaging the channels in the results. In Figure 2, the fifth column shows the visualizing results by optimizing the "global_pool" layer, while the sixth column presents the result for the "fc" layer.

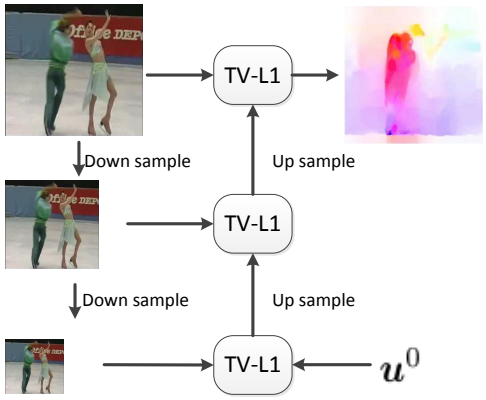

Figure 1. The multi-scale version of the TV-L1 method. Here the scale-size, *i.e.* N_{scales} is 3.

3. Action similarity labeling

Dataset. The ASLAN dataset [1] consists of 3, 631 videos of 432 action classes. We use the prescribed evaluation protocols as suggested by [1]. Different from action recognition, this problem focuses on predicting action similarity not the actual action label. The main challenge is that testing videos contains "never-seen-before" actions.

Implementation details. We use our TVNet-based models trained on UCF101. We split videos into 25 clips with equal length. For each clip, we extract the 4 kinds of features *i.e.*, global_pool, inception_5b_output, inception_5a_output, inception_4e_output from the BN-Inception net concatenated on TVNet-50. We also extract the feature of "global_pool" from the BN-Inception net trained on rgb images. The features for videos are computed by averaging the clip features separately for each type of feature, followed by an L2 normalization. Given a pair of videos, we compute the 12 different distances given by [1]. With 5 types of features, we obtain 60-dimensional ($12 \times 5 = 60$) feature vector for each video pair. We normalize these 60 distances independently such that each dimension has zero mean and unit variance. Finally, a linear SVM is trained to classify video pairs.

Results. Table 1 presents the results of our method compared to the state-of-the-art models. Our method achieves better accuracy and a comparable AUC value compared to C3D. The C3D model is pretrained on a the Sports-1M dataset with one

[2]https://github.com/auduno/deepdraw

million of videos, while our network is trained on UCF101 where only several thousands of videos are available. Even so, applying the features from our network still obtain desired performance, thus verifying the effectiveness of our method on modeling action.

References

[1] O. Kliper-Gross, T. Hassner, and L. Wolf. The action similarity labeling challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(3):615–621, 2012. 2

(c) The multi-scale version of the TV-L1 method. Here the scale-size, *i.e.* N_{scales} is 3.


RGB Frame
TV-L1
TVNet Non-Trained
TVNet
Visualization
ApplyLipStick
Archery
BabyCrawling
BandMarching
BoxingSpeedBag
BrushingTeeth
CliffDiving
Drumming
Fencing

Figure 2. Illustrations of the action patterns learned by TV-L1 and TVNet-50 on the UCF101 dataset. From the first to the fourth column, we display the image-pair (indeed the first image), the optical flow learned by TV-L1, the features leaned by TVNet-50 without and with task-specific training. The last two columns display the visualizations of "global_pool" and "fc" layers.