Cross-Model Pseudo-Labeling for Semi-Supervised Action Recognition
Supplementary Material

Yinghao Xu[1] Fangyun Wei[3] Xiao Sun[3] Ceyuan Yang[1]
Yujun Shen[1] Bo Dai[2] Bolei Zhou[1] Stephen Lin[3]
[1]The Chinese University of Hong Kong [2]S-Lab, Nanyang Technological University [3]Microsoft Research Asia
{xy119,yc019,sy116,bzhou}@ie.cuhk.edu.hk bo.dai@ntu.edu.sg {fawe,xias,stevelin}@microsoft.com

1. Pseudo-Labeling Schemes

How to generate pseudo-labels for unlabeled data based on the model outputs is also an important question for CMPL. Many pseudo-labeling strategies become possible with the introduction of the auxiliary network. Besides the one described in our method, we list some other options: 1) *Self-First*: Each network first checks whether its own prediction is confident enough, and if it is not, then the label is obtained from its sibling. 2) *Opposite-First*: Each network instead prioritizes its companion over itself. 3) *Maximum*: The most confident prediction from the two networks is taken as the pseudo-label. 4) *Average*: The predictions from the two networks are averaged before deriving the pseudo-label.

Let the pseudo-label confidence produced by F and A be l_F and l_A. The pseudo-label confidences for a video u_i are thus $l_F(u_i) = p_i^A, l_A(u_i) = p_i^F$. And now the corresponding mathematical formulations of different pseudo-labeling schemes are presented as following, where u_i is removed for clarity.

1. *Self-First*:

$$l_F = \mathbb{1}(\max(p^F) \geq \tau)p^F + (1 - \mathbb{1}(\max(p^F) \geq \tau))p^A,$$
$$l_A = \mathbb{1}(\max(p^A) \geq \tau)p^A + (1 - \mathbb{1}(\max(p^A) \geq \tau))p^F.$$

2. *Opposite-First*:

$$l_F = \mathbb{1}(\max(p^A) \geq \tau)p^A + (1 - \mathbb{1}(\max(p^A) \geq \tau))p^F,$$
$$l_A = \mathbb{1}(\max(p^F) \geq \tau)p^F + (1 - \mathbb{1}(\max(p^F) \geq \tau))p^A.$$

3. *Maximum*:

$$l_F = l_A = \mathbb{1}(\max(p^F) \geq \max(p^A))p^F + (1 - \mathbb{1}(\max(p^F) \geq \max(p^A)))p^A.$$

4. *Average*:

$$l_F = l_A = \frac{p^F + p^A}{2}.$$

Table 1. **Comparison across different pseudo-labeling schemes.** As there is no auxiliary network, FixMatch can only use the pseudo-labels generated by itself.

Pseudo-Labeling	Top-1
FixMatch	6.78
Self-Confident	10.80
Opposite-Confident	11.13
Maximum	11.85
Average	12.16
Cross	12.90

Experimental Results. Tab. 1 presents the results of different pseudo-labeling schemes. The baseline strategy (FixMatch [8]) performs the worst. Due to the lack of the auxiliary networks, the unlabeled data mainly distinguishable beyond the representation of the primary backbone rarely gets paired with confident pseudo labels since the scores of those unseen videos are easily below the threshold. After introducing the temporal information derived from the auxiliary network, clear improvements are observed. For the remaining strategies except the proposed cross-model scheme, there is a chance that the primary network will dominate the pseudo labeling decisions, leading to the wrong decisions for unlabeled samples. In contrast, for our cross-model strategy, each network always receives pseudo labels from its companion and never from itself, and this is shown to be more effective.

2. Comparison to Self-Supervised Methods

In this section, we compare CMPL with state-of-the-art self-supervised learning approaches. We use UCF-101 as the labeled data and Kinetics-400 as the unlabeled data. A described in the main paper, CMPL jointly use labeled and unlabeled data in a semi-supervised manner. As for self-supervised learning methods, we follow the standard

Table 2. **Comparison with other state-of-the-art self-supervised learning methods on UCF-101.** We use UCF-101 as the labeled data and Kinetics-400 as the unlabeled data. The other self-supervised methods are pretrained on Kinetics-400 and fine-tuned on UCF-101. Our model are trained from scratch.

Method	Architecture	#Frames	UCF-101 [9]
Random Init	3D-ResNet50	8	61.1
ImageNet Init	3D-ResNet50	8	86.2
MotionPred [12]	C3D [10]	16	61.2
RotNet3D [6]	3D-ResNet18	16	62.9
ST-Puzzle [7]	3D-ResNet18	16	65.8
ClipOrder [16]	R(2+1)D-18 [15].	-	72.4
DPC [3]	3D-ResNet34	-	75.7
AoT [14]	T-CAM	-	79.4
SpeedNet [1]	I3D [2]	64	81.1
VTHCL [17]	3D-ResNet50	8	82.1
PacePrediction [13]	S3D-G [11]	64	87.1
CoCLR [4]	S3D-G [11]	32	87.6
Ours	3D-ResNet50	8	**88.9**

protocol to use unlabeled data in Kinetics-400 for pre-training, followed by a fine-tuning on the labeled data in UCF-101.

As shown in Tab. 2, in comparison to the CoCLR [4], our model provides a performance gain of 1.3% only with 8 frames input, indicating the effectiveness of CMPL. It is a very encouraging result, suggesting that semi-supervised learning is a promising solution for action recognition with limited labeled data. We hope that our result can provide a strong baseline for comparison with more self-supervised learning methods.

3. 3D-ResNet Network Structure

Tab. 3 shows the architecture of 3D-ResNet50. It inherits the 2D-ResNet [5] and inflates the 2D kernel at $conv1$ across all stages. The other convolution blocks are still in 2D format, focusing on the spatial semantics. Moreover, there exist no temporal downsampling layers, in order to maintain long-temporal fidelity. Notably, we shrink the width of 3D-ResNet to a factor of 1/4 to use the 3D-ResNet50×1/4 as the default auxiliary pathway.

4. Effects of Sampling Schemes

As illustrated in Section 4.1 of the main paper, the number of sampled videos is the same across different categories. However, different from UCF-101, the distribution of videos across different categories is not balanced in Kinetics-400. We re-sample a new video subset under the Kinetics-400 distribution, called 'category-wise sampling scheme'. To be specific, we first compute the number of each category and next randomly sample the videos from each category with the corresponding ratio and the total number. Tab. 4 presents the results of different sampling schemes under the same setting of ablation study in Section 4.3 of the main paper. Even with the unbalanced distribution, CMPL obtains nearly the same performance with the 'uniform sampling' scheme, suggesting the robustness and generality of our approach.

Table 3. **3D-ResNet50 Network Structure**.The dimensions of convolution kernels are denoted by $\{K_T \times K_H \times K_W, K_C\}$ for temporal, height, width and channels sizes. The output size is in $\{C \times T \times S^2\}$ format denoting channel, temporal and spatial size. We take input size of $3 \times 8 \times 224^2$ which utilizes 8 frames with 224 spatial resolution as an example.

Stage	Block	Output Size
input	–	$3 \times 8 \times 224^2$
conv_1	5×7×7, 64 stride 1, 2, 2	$64 \times 8 \times 112^2$
pool_1	1×3×3, max stride 1, 2, 2	$64 \times 8 \times 56^2$
res_2	$\begin{bmatrix} 3\times1\times1, 64 \\ 1\times3\times3, 64 \\ 1\times1\times1, 256 \end{bmatrix} \times 3$	$256 \times 8 \times 56^2$
res_3	$\begin{bmatrix} 3\times1\times1, 128 \\ 1\times3\times3, 128 \\ 1\times1\times1, 512 \end{bmatrix} \times 4$	$512 \times 8 \times 28^2$
res_4	$\begin{bmatrix} 3\times1\times1, 256 \\ 1\times3\times3, 256 \\ 1\times1\times1, 1024 \end{bmatrix} \times 6$	$1024 \times 8 \times 14^2$
res_5	$\begin{bmatrix} 3\times1\times1, 512 \\ 1\times3\times3, 512 \\ 1\times1\times1, 2048 \end{bmatrix} \times 3$	$2048 \times 8 \times 7^2$

Table 4. Study on sampling Scheme.

	Uniform(Default)	Category-Wise
Top-1	12.90	12.68

References

[1] Sagie Benaim, Ariel Ephrat, Oran Lang, Inbar Mosseri, William T Freeman, Michael Rubinstein, Michal Irani, and Tali Dekel. Speednet: Learning the speediness in videos. *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020.

[2] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017.

[3] Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019.

[4] Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. *arXiv preprint arXiv:2010.09709*, 2020.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016.

[6] Longlong Jing and Yingli Tian. Self-supervised spatiotemporal feature learning by video geometric transformations. *arXiv preprint arXiv:1811.11387*, 2018.

[7] Dahun Kim, Donghyeon Cho, and In So Kweon. Self-supervised video representation learning with space-time cubic puzzles. In *Assoc. Adv. Artif. Intell.*, 2019.

[8] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *arXiv preprint arXiv:2001.07685*, 2020.

[9] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[10] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Int. Conf. Comput. Vis.*, 2015.

[11] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018.

[12] Jiangliu Wang, Jianbo Jiao, Linchao Bao, Shengfeng He, Yunhui Liu, and Wei Liu. Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[13] Jiangliu Wang, Jianbo Jiao, and Yun-Hui Liu. Self-supervised video representation learning by pace prediction. *Eur. Conf. Comput. Vis.*, 2020.

[14] Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. Learning and using the arrow of time. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018.

[15] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *Eur. Conf. Comput. Vis.*, 2018.

[16] Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-supervised spatiotemporal learning via video clip order prediction. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[17] Ceyuan Yang, Yinghao Xu, Bo Dai, and Bolei Zhou. Video representation learning with visual tempo consistency. *arXiv preprint arXiv:2006.15489*, 2020.