Semi-Supervised Action Recognition with Temporal Contrastive Learning (Supplementary Material)

Ankit Singh[1*] Omprakash Chakraborty[2*] Ashutosh Varshney[2] Rameswar Panda[3]
Rogerio Feris[3] Kate Saenko[3,4] Abir Das[2]
[1] IIT Madras, [2] IIT Kharagpur, [3] MIT-IBM Watson AI Lab, [4] Boston University
Project page: https://cvir.github.io/TCL/

This supplementary material contains the following.

- Section 1: Dataset details used in our experiments.
- Section 2: Implementation details of our **TCL** framework.
- Section 3: Implementation details of the video extensions of the image-based baselines.
- Section 4: Additional classwise improvements over S4L for 1% labeled data in Jester.
- Section 5: Effect of group contrastive loss on image datasets.
- Section 6: Additional qualitative examples from different datasets.

1. Dataset-Details

Mini-Something-V2. The Mini-Something-V2 dataset [2] is a subset of Something-Something V2 dataset [5]. It contains a total of 81663 training videos and 11799 validation videos. The resolution of each video is set to a height of 240px and has an average duration of 4.03 seconds. There are a total of 87 action classes related to basic object interactions such as 'Putting something into something', 'Showing something behind something', 'Squeezing something' and 'Showing that something is inside something'.

Jester. The jester dataset consists of a total of 148,092 videos spread across 27 classes with an average of 4391 per class samples. The classes belong to a series of hand gestures such as 'Sliding Two Fingers Up', 'Turning Hand Clockwise' and 'Swiping Down'. Specifically, the training set contains a total of 118,562 clips and 14,787 clips are provided for validation. The average duration of the videos are 3 seconds. The frames are extracted from these videos with 12 fps and maintain a fixed height of 100px but with variable width. The dataset is publicly available at https://20bn.com/datasets/jester/v1.

*The first two authors contributed equally.

Kinetics-400. The Kinetics-400 is a benchmark dataset containing YouTube videos of diverse human-action classes. It consists of around 300K videos spread across 400 classes with each class containing atleast 400 clips. The classes range across a broad spectrum of actions such as shaking hands, hugging and playing instruments. This dataset can be obtained from the link, https://deepmind.com/research/open-source/kinetics.

Charades-Ego. The Charades-Ego dataset is one of the largest datasets comprising of both first-person and third-person views of videos collected across a diverse set of 112 actors. The total 7,860 samples consist of around 4000 such pairs, each spanning around 31.2 seconds on average at 24 fps. The videos in this dataset have multiple activity classes which often overlap, making the dataset particularly challenging. The training set is divided into two separate lists, 'CharadesEgo_v1_train_only3rd' and 'CharadesEgo_v1_train_only1st', which contain the videos corresonding to the third-person and first-person perespectives respectively. Each file lists the video ids with their corresponding activity classes. Following the standard practice [12], we first trim the multi-class 3082 videos of 'CharadesEgo_v1_train_only3rd' and 3085 videos of 'CharadesEgo_v1_train_only1st' to obtain 34254 and 33081 single-class clips respectively. We select the 10% labeled videos class-wise from the 34254 trimmed clips distributed over 157 activity classes. The mAP metric is evaluated over the full 'CharadesEgo_v1_test_only3rd' video set. The dataset is publicly available at https://github.com/gsig/actor-observer.

2. Implementation Details

In this section, we provide additional implementation details (refer Section 4.1 of the main paper) of our **TCL** framework. For the basic convolution operation over the videos, we use the approach identical to that of Temporal Segment Network (TSM) [7]. We utilize the 2D CNNs for their lesser computational complexity over the 3D counterparts

and implement the bi-directional temporal shift module to move the feature channels along the temporal dimension to capture the temporal modeling of the samples efficiently. All hyperparameters related to TSM module has been taken from [7]. As shown in [7], this approach achieves the state-of-art performances while significantly reducing the computational complexity. We have considered 2D ResNet-18 model as our primary backbone and have incorporated the temporal shift module after every residual branch of the model to avoid the interference with the spatial feature learning capability. In our experiments, one epoch has been defined as one pass through all the labeled data. We have used learning rate of 0.002 during the finetuning stage.

3. Image-based Baseline Details

This section provides implementations details of different baselines used in the paper. We have adhered to the base approach proposed in the original works of the respective baselines for all our experiments. Note that, for a given video, same set of augmentations have been applied to all frames of the video so that all frames in a video go through the same set of transformations and do not loose the temporal consistency between the them. Also, following TSM [7], for the high spatially-sensitive datasets like Mini-Something-V2 [5] and Jester [8], we refrain from using the *Random Horizontal Flip* as it may effect the spatial semantics of the frames. The initial lr is set to 0.02 with cosine learning decay in all our baseline experiments unless stated otherwise. All the baselines models are trained for 350 epochs unless otherwise specified.

Supervised We have used the code made public by the authors in [7] for the supervised baseline. It is trained using $\mathcal{L}_{sup}$ for 200 epochs and the initial learning rate is kept same as in TCL. Other hyperparameters are kept same as the ones used for the respective datasets in [7].

MixMatch We followed the approach in [1] to train our MixMatch baseline approach. We applied 2 different augmentations to unlabeled videos set (U) and then computed the average of the predictions across these augmentations. We have used cropping and flipping as the two augmentations in our experiments. The sharpened versions of the average predictions of K different augmentations are used as labels for the unlabeled videos. Then, labeled (V) and unlabeled videos with their targets and predicted labels are shuffled and concatenated to form another set W which serves as a source for modified MixUp algorithm defined in [1]. Then for each i^{th} labeled video we compute $\mathit{MixUp}(V_i, W_i)$ and add the result to a set V'. It contains the *MixUp* of labeled videos with W. Similarly for each j^{th} unlabeled video, we compute $\mathit{MixUp}(U_i, W_{i+|V|})$ and add the result to another set U'. It contains the *MixUp* of unlabeled videos with rest of W. A cross-entropy loss between labels and model predictions from V' and MSE loss between the predictions and guessed labels from U' are used for training. The temperature is set to 0.5 and both μ and γ are set to 1.

S4L: S4L [11] is a self-supervised semi-supervised baseline used in our work. The self-supervision is done by rotating the input videos. Videos are rotated by $\{0, 90, 180, 270\}$ degrees and the model is trained to predict these rotations of the videos. The corresponding rotation loss [11] is used for both labeled and unlabeled videos. The μ and γ are set to 5 in this baseline experiment. The S4L model is trained using rotation loss apart from the $\mathcal{L}_{sup}$ for labeled videos. The initial learning rate is set to 0.1.

Pseudo-Label Pseudo-label [6] leverages the idea that in absence of huge amount of labeled data, artificial labels or pseudo-labels for unlabeled data should be obtained using the model itself. Following this basic intuition, we first train our model using $\mathcal{L}_{sup}$ for 50 epochs to get a reasonably trained model. The next 300 epochs are run using both labeled and unlabeled videos. Consistency is ensured between the pseudo-labels of the unlabeled video with the logits predicted for them by the model. The class for which an unlabeled video gets the highest activation from the model is taken as the pseudo-label of it. Only videos which have highest activation greater than 0.95 are assigned pseudo-labels. Both μ and γ are set to 3 in this set of experiments.

MeanTeacher : The model is trained using the philosophy described in [10]. In this scenario, we have two models, one is the *student* network and the other is the *teacher* network. The teacher network has the same backbone architecture as the student. The weights of the teacher network are exponential moving average weights of the student network. Consistency is ensured between the logits predicted by the teacher and the student for the unlabeled videos. The labeled data, in addition, is trained using $\mathcal{L}_{sup}$. Both μ and γ are set to 1 in this set of experiments. γ is increased from 0 to 1 using sigmoid function over 50 epochs as in [10].

FixMatch. For extending the FixMatch baseline to video domain, we primarily follow the same augmentation and consistency regularization policies laid out in [9]. The videos are passed through two different pathways. In the first pathway, the video frames are weakly augmented and used to obtain the pseudo-labels. In the second pathway, the strongly augmented versions of the same video frames are trained for their representations to be consistent with the corresponding pseudo-labels. Specifically, in the case of weak augmentations, we use *Random Horizontal Flip* followed by *Random Vertical and Horizontal shifts*. For the strong augmentations we use the *RandAugment* [4] augmentation policy followed by *CutOut* augmentation. The experiments are carried out for 350 epochs with a batch size of 8 and considering the μ and γ values as 3 and 9 respectively.

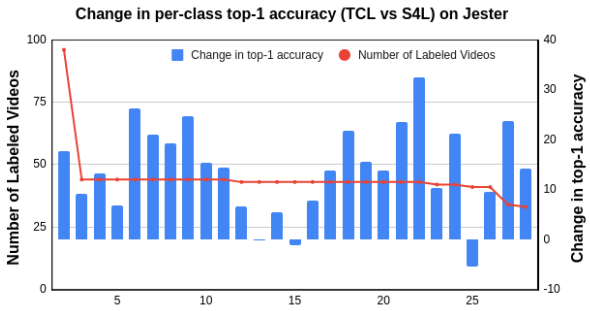

Figure 1: **Change in classwise top-1 accuracy of `TCL` over S4L on Jester.** Blue bars show the change in accuracy on 1% labeled scenario of Jester dataset. The red line depicts the number of labeled videos per class in a sorted manner. Compared to S4L, **`TCL`** improves the performance of most classes including those with less labeled data. (Best viewed in color.)

4. Classwise Improvements

In the main paper, we have presented the change in top-1 accuracy per class of **`TCL`** over FixMatch on 5% Mini-Something V2. Here, we have included the change in top-1 accuracy per class of **`TCL`** over S4L (next best) on Jester dataset using only 1% labeled data in Figure 1. We can observe in Figure 1 that only 2 classes in Jester have less improvement over S4L for this 1% labeled data scenario.

5. Group Contrastive Loss on Image Dataset

We analyze the effect of group contrastive loss on CIFAR10 (using SimCLR [3] with WideResNet-28-2 and 4 labeled samples per class) and observe that it improves performance by 3.15% (84.11% vs 87.26%), showing the effectiveness of group contrastive loss in semi-supervised classification on image datasets too besides the video datasets.

6. Qualitative Examples

In the Main paper, we provided qualitative examples from Jester and kinetics-400 dataset. Here we have included some more samples from all four datasets to show the superiority of our methods over the competing baseline methods. Figure 2, 3, 4 and 5 contain the example frames and their predictions for Mini-Something V2, Jester, Kinetics-400 and Charades-ego respectively.

References

[1] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A Holistic Approach to Semi-Supervised Learning. In *Neural Information Processing Systems*, pages 5050–5060, 2019. 2

[2] Chun-Fu Chen, Rameswar Panda, Kandan Ramakrishnan, Rogerio Feris, John Cohn, Aude Oliva, and Quanfu Fan. Deep Analysis of CNN-based Spatio-temporal Representations for Action Recognition. *ArXiv preprint ArXiv:2010.11757*, 2020. 1

[3] Ting Chen, Simon Kornblith, M. Norouzi, and Geoffrey E. Hinton. A Simple Framework for Contrastive Learning of Visual Representations. *ArXiv preprint ArXiv:2002.05709*, 2020. 3

[4] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical Automated Data Augmentation with a Reduced Search Space. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 702–703, 2020. 2

[5] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, Florian Hoppe, Christian Thurau, Ingo Bax, and Roland Memisevic. The "Something Something" Video Database for Learning and Evaluating Visual Common Sense. In *IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 1, 2

[6] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *International Conference on Machine Learning Workshop*, volume 3, page 2, 2013. 2

[7] Ji Lin, Chuang Gan, and Song Han. TSM: Temporal Shift Module for Efficient Video Understanding. In *IEEE International Conference on Computer Vision*, pages 7083–7093, 2019. 1, 2

[8] Joanna Materzynska, Guillaume Berger, Ingo Bax, and Roland Memisevic. The Jester Dataset: A Large-Scale Video Dataset of Human Gestures. In *IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019. 2

[9] 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. In *Neural Information Processing Systems*, 2020. 2, 4, 5

[10] Antti Tarvainen and Harri Valpola. Mean Teachers are Better Role Models: Weight-Averaged Consistency Targets Improve Semi-Supervised Deep Learning Results. In *Neural Information Processing Systems*, pages 1195–1204, 2017. 2

[11] Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4L: Self-Supervised Semi-Supervised Learning. In *IEEE International Conference on Computer Vision*, pages 1476–1485, 2019. 2, 4

[12] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal Relational Reasoning in Videos. In *European Conference on Computer Vision (ECCV)*, pages 803–818, 2018. 1

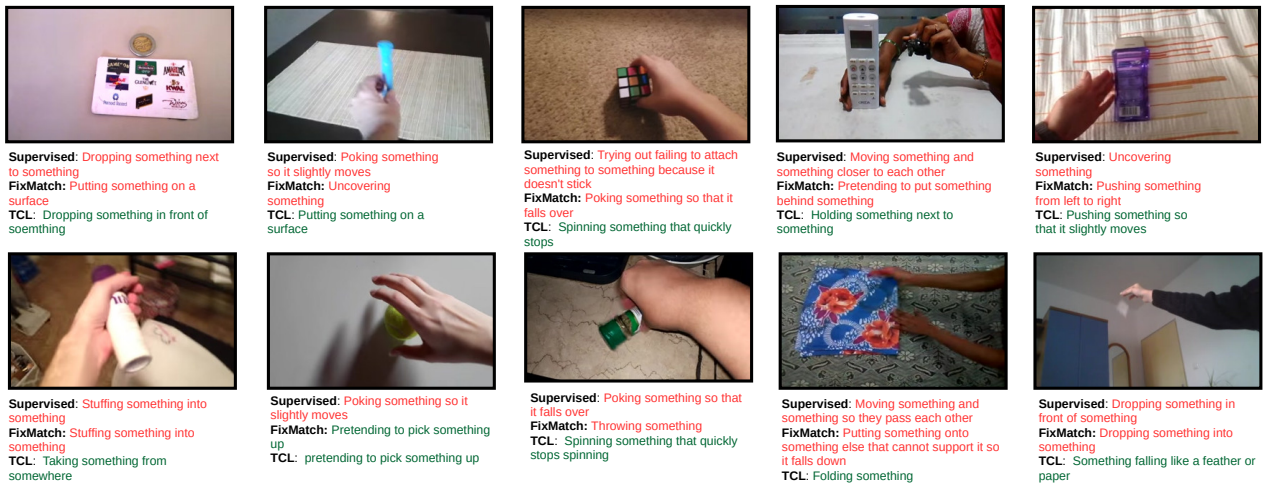

Figure 2: **Qualitative examples comparing `TCL` with supervised baseline and FixMatch [9] on Mini-Something V2 trained using** 5% **labeled data with ResNet-18**. Both rows provide top-1 predictions using supervised baseline, FixMatch and proposed **`TCL`** approach respectively from top to bottom. As observed, the supervised baseline trained using only the labeled data predicts wrong actions. While the competing methods fail to classify the correct actions in most cases **`TCL`** is able to correctly recognize different actions in this dataset. The predictions marked in green match the ground truth labels, whereas the red marked predictions are wrong. (Best viewed in color.)

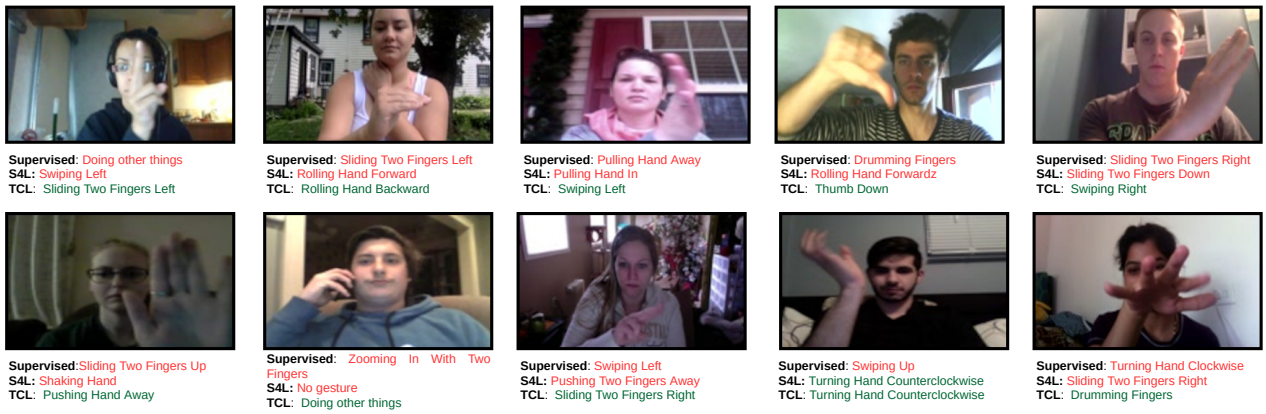

Figure 3: **Qualitative examples comparing `TCL` with supervised baseline and S4L [11] on Jester dataset trained using 1% labeled data with ResNet-18.** Both rows provide top-1 predictions using supervised baseline, S4L and **`TCL`** respectively from top to bottom. As observed, the supervised baseline trained using only the labeled data predicts wrong actions. While the competing methods fail to classify the correct actions in most cases, our proposed approach, **`TCL`** is able to correctly recognize different hand gestures in this dataset. The predictions marked in green match the ground truth labels, whereas the red marked predictions are wrong. (Best viewed in color.)

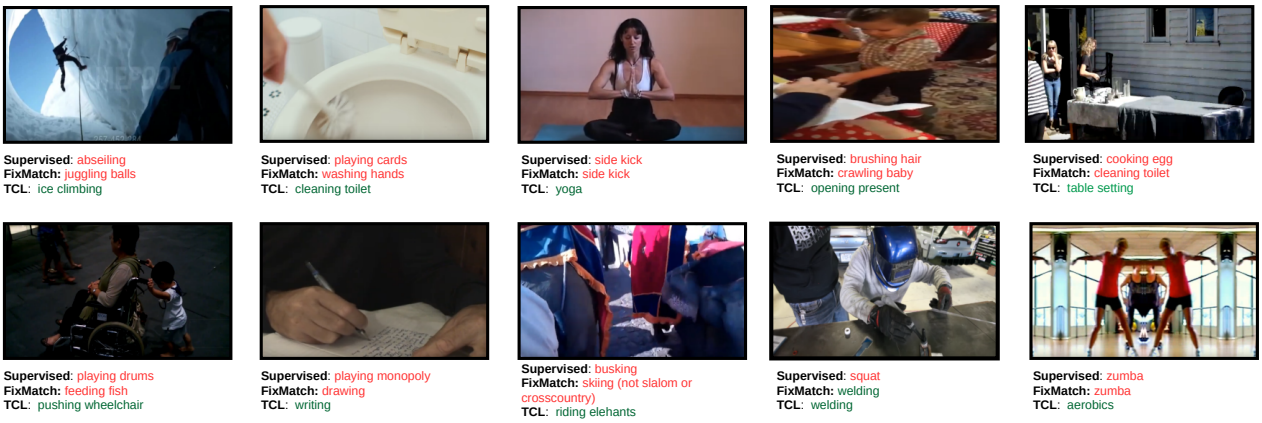

Figure 4: **Qualitative examples comparing `TCL` with supervised baseline and FixMatch [9] on Kinetics-400 trained using** 5% **labeled data with ResNet-18**. Both rows provide top-1 predictions using supervised baseline, FixMatch and **`TCL`** respectively from top to bottom. As observed, the supervised baseline trained using only the labeled data predicts wrong actions. While the competing methods fail to classify the correct actions in most cases our proposed approach, **`TCL`** is able to correctly recognize different actions in this dataset. The predictions marked in green match the ground truth labels, whereas the red marked predictions are wrong. (Best viewed in color.)

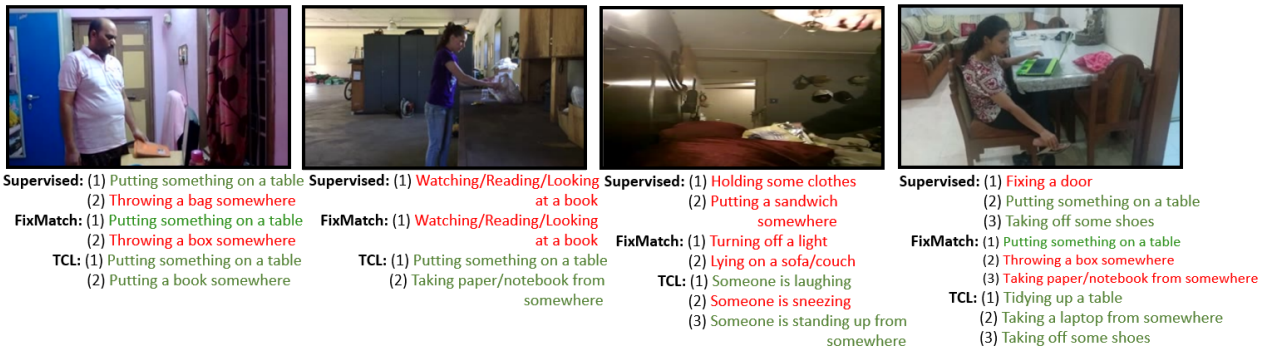

Figure 5: **Qualitative examples comparing `TCL` with supervised baseline and FixMatch [9] on Charades-Ego**. As each of the video samples have multiple actions, we show random frames from different videos of the dataset and compare the Top-K predictions for those frames. Here, 'K' denotes the number of ground-truth classes associated with the respective samples. While the supervised and competing methods fail to classify all the correct actions in most cases, **`TCL`** is able to correctly recognize most of the relevant actions in these videos. The predictions marked in green match ground truth labels, whereas red marked predictions are wrong. (Best viewed in color.)