

Supplementary Material for Unsupervised Learning of Video Representations via Dense Trajectory Clustering

Pavel Tokmakov¹, Martial Hebert¹, and Cordelia Schmid²

¹ Carnegie Mellon University

² Inria

In this supplementary material, we provided additional analysis of our method, that was not included in the main manuscript due to space limitations. We begin with reporting few-shot results for various self-supervised objectives in Section 1. We then discuss some additional ablations of our approach in Section 2, and study the effect of the hyper-parameters of LA objective function on the final performance in Section 3. We also evaluate the impact of the network depth and input resolution in Section 4. We conclude with reporting the remaining implementation details of our method in Section 5.

1 Few-shot evaluation

When finetuning a model, even on a datasets of modest size, like UCF101, the effect of self-supervised pretraining is confounded by the effectiveness of the adaptation strategy itself. Indeed, it has been shown recently that, on several tasks that were traditionally used to measure the effectiveness of image-based unsupervised learning approaches, fully supervised performance can be achieved with no pretraining at all, by simply better utilizing the existing data [1]. Thus, to gain more insight into our objectives, we propose to use pretrained models as feature extractors, and learn linear classifiers in a few-shot regime. The results on UCF101 are reported in Table 1.

The most important observation here is that the gap between fully-supervised and unsupervised representations increases as the data becomes scarcer. This shows that, despite being useful in practice, unsupervised pretraining is still far

Table 1. Comparison between variants of unsupervised learning objective on the first split of UCF101 in a few-shot regime, using classification accuracy. The networks are fully frozen, and a linear classifier is learned, gradually decreasing the amount of training data. The gap between unsupervised and supervised representations increases, but our full method (‘Video LA + IDT’) still outperforms other variants across the board.

Method	1-shot	5-shot	10-shot	20-shot	All
Scratch	1.7	7.5	10.6	17.2	38.2
Video IR	13.4	27.7	35.2	42.4	56.5
Video LA	15.6	30.6	36.4	44.2	58.6
Video LA + IDT prior	17.8	31.5	38.4	45.5	58.8
Supervised	46.4	62.0	67.7	73.3	81.8

from making large datasets obsolete. Among the objectives studied in our work, however, Video LA with IDT prior shows the strongest performance across the board, and is especially effective in the low-data regime.

2 Additional ablations

In the main paper, we capitalized on the version of IDT which uses human detections to suppress optical flow in background regions. To validate the importance of this component, we have recomputed IDTs without human detections, and report the results in Table 2 (denoted as Ours\Det). Removing this step from the IDT pipeline indeed decreases the performance of our approach both on UCF101 and HMDB51, confirming the observations in [2] that suppressing background motion improves the descriptors’ quality.

Next, we evaluate the final tuning step in our approach. Recall that after training the network with the clusters obtained in the IDT space, we construct a joint space of IDT and 3D ConvNet representations, and further tune the network in this space using the iterative Local Aggregation objective. A variant without this tuning step, in reported in Table 2 as Ours

Table 2. Additional ablations on the first split of UCF101 and HMDB51 using classification accuracy. All the models use a 3D ResNet18 backbone and take 16 frames of resolution 112×112 as input.

Model	UCF101	HMDB51
Ours\Det	72.6	43.1
Ours\Tune	72.6	43.1
Ours full	72.8	44.0

\Tune, indeed achieves lower performance (coincidentally, it is exactly the same as the performance of the variant without person detections). This demonstrates that, although IDT descriptors already capture appearance information, using the more expressive 3D ConvNet representation provides further benefits.

3 Effect of the objective parameters

Finally, we ablate the hyper-parameters of the Local Aggregation objective function (number of clusters K , and number of runs of K-mean m) in Table 3. The results in the main paper were obtained with $K = 6000$, and $m = 3$, which roughly correspond to the parameters used in [3] adjusted for the size of Kinetics dataset. As can be seen from the table, increasing the values of these hyper-parameters improves the performance on UCF101,

Table 3. Effect of the hyper-parameters of the LA objective function on the first split of UCF101 and HMDB51 using classification accuracy. All the models use a 3D ResNet18 backbone and take 16 frames of resolution 112×112 as input.

K	m	UCF101	HMDB51
30000	10	73.0	42.4
12000	6	73.7	42.2
6000	3	72.8	44.0
3000	2	72.8	43.9
1500	1	72.2	41.6

but hurts on HMDB51. Decreasing these values, in contrast, hurts the performance on both datasets. Overall, the values we used in the main paper strike a good balance for the two benchmarks.

Table 4. Evaluation of the effect of the network depth and input resolution on the first split of UCF101 and HMDB51 using classification accuracy.

Network	Frame resolution	UCF101	HMDB51
3D ResNet18	112×112	72.8	44.0
3D ResNet18	224×224	74.6	43.8
3D ResNet34	112×112	73.6	42.2
3D ResNet34	224×224	77.1	45.8

4 Effect of the network depth and resolution

In this section, we evaluate how the performance of our approach changes with the network depth and input resolution. To this end, we first independently increase the resolution to 224×224 , and network depth to 34, compared to 112×112 and 18 used in the rest of the paper, and then report a combined variant (3D ResNet34 with 224×224 inputs) in Table 4. All the models are learned on the training set of Kinetics-400 with 16-frame long clips, using our final objective (Video LA with IDT prior), and tuned on UCF101 and HMDB51.

Firstly, we observe that increasing the input resolution indeed results in a significant performance improvement on UCF101, whereas on HMDB51 accuracy remains almost unchanged. This is in line with to our intuition that appearance information is more important for UCF101. Curiously, increasing the network depth while keeping the original input resolution decreases the performance on HMDB51, while providing a modest improvement on UCF101. We hypothesize that the model capacity is limited by the small resolution. Indeed, the final variant, which combines larger inputs with a deeper network, shows significant improvements over the baseline on both UCF101 and HMDB51. Even higher accuracy could be obtained by training this configuration with longer clips.

5 Implementation details

For experiments with with IDT priors we use exactly the same hyper-parameters for the LA objective as described above. We use the original implementation of [2] to extract IDT descriptors. Human detections are computed with ResNet101 variant of Mask-RCNN [4] model pretrained on MS COCO [5]. We evaluate the importance of human detections for the final performance of our approach in the supplementary material. When computing Fisher vector encoding, we generally follow the setting of [6]. In particular, we set the feature importance to 90% when computing PCA, and the number of components in GMM to 256. When fitting the PCA and GMM models we randomly choose 3500 videos from Kinetics and 500 IDT descriptors from each video, to get a representative sample. Note that extracting IDTs and encoding them into Fisher vectors does not require GPUs, and thus the code can be efficiently run in parallel on a CPU cluster. As a result, we were able to compute the descriptors for Kinetics in just 5 days.

When finetuning on UCF101 and HMDB51, we set the learning rate to 0.1 and momentum to 0.9, using batch size 128. We drop the learning rate by a

factor of 0.1 when the validation performance stops improving. Following [7], we freeze the first ResNet block when finetuning on UCF101, and the first two blocks on HMDB51 to avoid overfitting. During inference, for every video we sample five clips at random, using the center crop. The final prediction is obtained by averaging softmax scores over the five clips. For few-shot experiments, we use the protocol of [8] and freeze the entire network, only learning a linear classifier.

References

1. He, K., Girshick, R., Dollár, P.: Rethinking ImageNet pre-training. In: ICCV. (2019) [1](#)
2. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: ICCV. (2013) [2](#), [3](#)
3. Zhuang, C., Zhai, A.L., Yamins, D.: Local aggregation for unsupervised learning of visual embeddings. In: ICVV. (2019) [2](#)
4. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: ICCV. (2017) [3](#)
5. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common objects in context. In: ECCV. (2014) [3](#)
6. Wang, L., Koniusz, P., Huynh, D.Q.: Hallucinating IDT descriptors and I3D optical flow features for action recognition with CNNs. In: ICCV. (2019) [3](#)
7. Jing, L., Yang, X., Liu, J., Tian, Y.: Self-supervised spatiotemporal feature learning via video rotation prediction. arXiv preprint arXiv:1811.11387 (2018) [4](#)
8. Chen, W.Y., Liu, Y.C., Kira, Z., Wang, Y.C.F., Huang, J.B.: A closer look at few-shot classification. In: ICLR. (2019) [4](#)