Capturing Temporal Information in a Single Frame: Channel Sampling Strategies for Action Recognition - Supplementary Material

Kiyoon Kim
kiyoon.kim@ed.ac.uk
Shreyank N Gowda
s.narayana-gowda@sms.ed.ac.uk
Oisin Mac Aodha
oisin.macaodha@ed.ac.uk
Laura Sevilla-Lara
lsevilla@ed.ac.uk

School of Informatics
University of Edinburgh
Edinburgh, UK

1 Training Details

On CATER task 2, CATER Camera Motion task 2, and Something-Something datasets, we used 2 NVIDIA RTX 3090 GPUs to train and test. We used 2 NVIDIA RTX 2080 Ti GPUs for the other datasets. For CATER, we used 32 frames due to the need for long-term temporal understanding. We set the total batch size to 24 and the initial learning rate to 0.0024. For Something-Something, we used 8 frames, a total batch size of 64, and an initial learning rate of 0.0064. As an exception, TRN and MTRN models are trained with one RTX 3090 GPU with half the total batch size and quarter the learning rate.

In all of the experiments, we kept the following protocols. The videos were resized so that the shorter side becomes 224 to 336 pixels, and we performed random cropping during training. For testing, we resized the input to have the shorter spatial side resolution of 256 and we used one center crop for CATER and Something-Something, and five crops (center and corners) and their horizontal flips for other datasets. For I3D experiments, we followed the common practice from [5] of densely sampling the video with a sampling stride of eight for RGB, three for GrayST because it samples more frames, and tested using ten evenly sampled clips throughout the video with three spatial crops of each, totaling 30 clips. The learning rate was decayed by 0.1 when validation metrics saturated for ten epochs. We stopped the experiments when the validation metrics saturated for 20 epochs. We used 16-bit precision to save memory, increase the batch size, and to train faster.

Model	Sampling	Top-1	Top-5
TSN	RGB	17.2	42.7
	TC	**36.8**	**65.3**
	TC-RGB	33.6	61.4
	TC-Red	36.0	65.2
	TC-ShortLong	35.2	64.4
TSM	RGB	45.4	74.5
	TC	**45.8**	**74.7**
	TC-RGB	44.9	74.1
	TC-Red	44.7	73.8
	TC-ShortLong	44.8	73.6

Table 1: Ablation experiments of different TC Reordering strategies on Something-Something-V1. RGB refers to standard RGB channel sampling and TC is our proposed sampling strategy.

2 Ablation Studies

TC variants. We explore some variants for ablation experiments: TC-Red, TC-RGB, and TC-ShortLong. TC-Red uses only red channels with the same frame ordering, *i.e.* $\mathbf{x}_i^{\mathrm{TC}} = (x_i^{\mathrm{R}}, x_{i+1}^{\mathrm{R}}, x_{i+2}^{\mathrm{R}})$. This baseline allows us to measure the effect of the diversity of color information using the TC Reordering. TC-RGB uses traditional RGB-like representation with the same temporal frame ordering as the TC Reordering: $\mathbf{x}_i^{\mathrm{TC}} = (x_i^{\mathrm{R}}, x_{i+1}^{\mathrm{G}}, x_{i+2}^{\mathrm{B}})$. Intuitively, this may seem to be the best representation as this is the closest representation to the RGB representation that the model is pre-trained on. However, in our experiments we observe that our TC Reordering actually outperforms TC-RGB. Finally, TC-ShortLong replaces the last two frames that consists of a lot of duplicates, with frames having longer sampling stride instead, *i.e.* $\mathbf{x}_7^{\mathrm{TC}} = (x_3^{\mathrm{R}}, x_5^{\mathrm{R}}, x_7^{\mathrm{R}})$ and $\mathbf{x}_8^{\mathrm{TC}} = (x_4^{\mathrm{G}}, x_6^{\mathrm{G}}, x_8^{\mathrm{G}})$ for the $T = 8$ case.

In Table 1 we contrast the different variants of TC Reordering. In the case of TSN, all variants of TC Reordering result in a significant performance increase compared to standard RGB. However, for TSM, only our proposed TC variant is superior. Perhaps counter-intuitively, TC-RGB, which maintains the RGB channel order but temporally shifts them, actually performs worse than our TC Reordering. TC frames consist of information from the same input color channel, which perhaps better enables it to capture local temporal changes, especially in cases where color is not informative for the task at hand.

Grayscale-only We report a grayscale-only experiment result on Something-Something-V1 in Table 2. We let $\mathbf{X}^{\mathrm{GO}} = \{\mathbf{x}_1^{\mathrm{GO}}, \mathbf{x}_2^{\mathrm{GO}}, \ldots, \mathbf{x}_T^{\mathrm{GO}}\}$ denote a GrayOnly video clip. We sample T grayscale frames following the sparse sampling strategy,

$$\mathbf{X}^g = \{x_1^g, x_2^g, \ldots, x_T^g\}, \tag{1}$$

where x_i^g is a grayscale image. Then, a GrayOnly frame is made by duplicating the same channel three times.

$$\mathbf{x}_i^{\mathrm{GO}} = (x_i^g, x_i^g, x_i^g). \tag{2}$$

We see very little difference in performance compared to RGB.

Model	Sampling	Top-1	Top-5
TSN	GrayOnly	16.1	41.2
	RGB	17.2	42.7
	TC	36.8	65.3
	TC+2	**37.0**	**65.6**
	GrayST	35.5	65.4

Table 2: Grayscale-only result on Something-Something-V1. Surprisingly, the performance is very close to that of RGB, and our methods utilize this fact to make the models further capture temporal information.

Model	Sampling	Top-1	Top-5
TSN	RGB	50.7	84.6
	TC+2	65.9	92.0
	GrayST	**66.4**	**92.7**
TSM	RGB	71.7	94.2
	TC+2	73.3	94.0
	GrayST	**73.7**	**95.0**

Table 3: Evaluation of our methods on the OnlyTimeCanTell-Temporal dataset.

3 Only Time Can Tell dataset

We show results on OnlyTimeCanTell-Temporal dataset [3] in Table 3. The dataset consists of 50 classes which require extensive temporal reasoning, taken from the Kinetics-400 and Something-Something-V1 dataset.

4 Limitations

Despite the positive results, we also find some limitations of the proposed approaches, which we hope can lead to interesting future work.

Datasets Requiring Less Temporal Reasoning. Numerous datasets have significant object and scene bias making even TSN perform very similar to powerful 3D networks. In such cases, we found that the sampling strategies do not result in improved performance. Table 4 and 5 show such cases on Diving-48 [2] and UCF-101 [4] datasets.

Interestingly, Diving-48 V2 shows a decrease in performance with the GrayST input. The implication of this is that color information is important on this dataset, and with grayscale images it is difficult to distinguish divers of interest from the background. Despite the fact that the dataset is said to be "temporally-heavy", our experiment showed that the gap between 2D network and the state-of-the-art 3D network with double the number of frames and backbone depth is marginal.

The UCF-101 labels are biased towards object and scene information. As a result the performance of TSM is almost identical to that of TSN. The result did not show strong pattern but in general RGB seems to be preferable in this case.

Model	Backbone	#frame	Sampling	Top-1
TSN	ResNet50	8	RGB	**75.4**
			TC	75.0
			GrayST	73.5
SlowFast*	ResNet101	16	RGB	77.6

Table 4: Evaluation on Diving-48-V2. The result marked with $*$ is from [1], which uses 30 clips for testing (3 spatial $\times$ 10 temporal).

Model	Backbone	Sampling	Top-1	Top-5
TSN	ResNet50	RGB	83.6	**96.1**
		TC	83.6	96.0
		GrayST	**84.7**	95.9
TRN	ResNet50	RGB	**84.6**	**96.6**
		TC	81.6	95.3
		GrayST	82.2	95.5
TSM	ResNet50	RGB	**83.7**	96.0
		TC	81.4	95.5
		GrayST	83.6	**96.1**

Table 5: 8-frame evaluation on UCF-101 split 1.

Camera Motion. We found that TC Reordering combined with TRN and TSM negatively impacts performance on the CATER Camera Motion dataset. Note that the models make use of the temporal ordering of frames while TSN does not. Again, we still saw improvement with the GrayST method on this dataset. Judging from the fact that TC Reordering only hurts the temporal models, we think that this ordering plays a critical role in heavy camera motion scenarios.

Additionally, this dataset requires 3D-geometric understanding as the camera orbits around the objects substantially, making stationary objects look like they are sliding. The strength of TC Reordering is in cases where the model can analyze the motion information, but the large camera motion likely confuses the model when trying to understand what is the action of interest and what is the camera motion.

References

[1] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? *arXiv preprint arXiv:2102.05095*, 2021.

[2] Yingwei Li, Yi Li, and Nuno Vasconcelos. Resound: Towards action recognition without representation bias. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[3] Laura Sevilla-Lara, Shengxin Zha, Zhicheng Yan, Vedanuj Goswami, Matt Feiszli, and Lorenzo Torresani. Only time can tell: Discovering temporal data for temporal modeling. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 535–544, January 2021.

[4] 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.

[5] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018.