Supplementary for NSNet: Non-saliency Suppression Sampler for Efficient Video Recognition

In this supplementary, we provide more implementation details and experimental results of our proposed NSNet. Accordingly, we organize the supplementary materials as follows.

- In Section A, we present more implementation details for training and inference of our method.
- In Section B, we provide more ablation studies to further analyze the capability of our proposed NSNet.
- In Section C, we present predicted saliency score distributions to qualitatively analyze the capability of proposed NSNet.

A Implementation Details

A.1 Training

Pre-processing. Following previous works, the frames fed to recognizer are rescaled with a shorter side of 256 and center cropped to 224×224 for all datasets. The resolution of frames fed into Feature Embedding module is 224×224 for ActivityNet, FCVID and UCF101, and 112×112 for Mini-Kinetics. Note that we **only** use the RGB frames of these datasets for experiments. Following [6], before adaptive sampling by samplers, T frames are uniformly pre-sampled from frame sequence. For those videos whose lengths are shorter than T, we repeat multiple times and splice them to T frames.

Model training details and hyper-parameters. For transformer encoder, the hidden dimensions of query, key and value is set to the ratio between the number of input feature channels and the number of heads. The hidden dimension of FFN is set to be equal to the input feature channel number. Dropout [4] is used to reduce over-fitting. In Video Glimpse module, dropout layers are placed before classification fully-connected layer with ratio of 0.9 and after temporal attention layer with 0.2, respectively. In transformer encoder, the dropout rate after the positional encoding layer is set to 0.2. Temporal random shift is adopted as data augmentation strategy. The model is trained using SGD optimizer with momentum of 0.9 and batch size of 64 for 120 epochs. The learning rate is set to starting at 10^{-2}, decaying by the factor of 0.1 at the 50^{th} and 75^{th} epoch.

Prototype generation. For a video x_v, we first apply the recognizer for each frame and obtain the predictions $\{\hat{y}_i \in \mathbb{R}^C\}_{i=1}^T$ of all frames. Then we collect the correctly predicted frames set from each video $X_g = \{x_i|_{i\in[1,T],\underset{j}{\text{argmax}}\hat{y}_{i,j}=c}\}$, where c is the ground truth category. We further select the top ϵ percent frames with highest confidence on the c-th category $\hat{y}_{i,c}$ from X_g and average pool the frame features of them as the guiding video feature $\tilde{x}_v^g$. Then, for the c-th category, the prototype feature p_c can be computed by average pooling the all the guiding video features belonging to the c-th category. We use $\epsilon = 30$ in all experiments.

A.2 Inference

We describe the combination strategies in detail here.

A.3 Score Combination.

We consider 3 types of fusion operations, which includes *addition*, *multiplication* and *maximization*. For *addition*, we fuse the saliency scores of two branches in convex combination $\alpha s_i^f + (1-\alpha)s_i^v$, where α is a combination ratio parameter. For *multiplication* and *maximization*, we fuse the saliency scores of two branches in element-wise muliplication $s_i^f * s_i^v$ and element-wise maximization $\max(s_i^f, s_i^v)$, respectively.

Index Combination. We consider three strategies, which involves *intersection*, *union* and *join*. We firstly get frame index lists $\{\pi_i^f\}_{i=1}^T$ and $\{\pi_i^v\}_{i=1}^T$ by sorting $\{s_i^f\}_{i=1}^T$ and $\{s_i^v\}_{i=1}^T$ in descending order, respectively. In *intersection*, given a budget of K salient frames at most, we firstly take top K frames from index lists, $\{\pi_i^f\}_{i=1}^K$ and $\{\pi_i^v\}_{i=1}^K$ respectively and get the intersection of them $I(K) = \{\pi_i^f\}_{i=1}^K \cap \{\pi_i^f\}_{i=1}^K$. When there exist coincident frames, we expand I with one element from either $\{\pi_i^f\}_{i=K+1}^T$ or $\{\pi_i^v\}_{i=K+1}^T$ by turns for i' steps, until $|I(K+i')| = K$. For *union*, following [1], we try to obtain a set of salient frames whose length is represented by $\alpha|\pi^f| + (1-\alpha)|\pi^v|$. We firstly get the union of top saliency frames from two lists $U(K) = \{\pi_i^f\}_{i=1}^{\lceil K*\alpha \rceil} \cup \{\pi_i^v\}_{i=1}^{\lceil K*(1-\alpha) \rceil}$. We expand $U(K)$ with one element from π^f at a time for i' steps until $|U(K+i')| = K$. For *join*, we concatenate $\{s_i^f\}_{i=1}^T$ and $\{s_i^v\}_{i=1}^T$ to a list with a length of $2T$, from which K non-overlap top saliency score frames are selected as final salient frames set.

We use $\alpha = 0.6$ for score addition and index union in the ablation studies of fusion strategies. Union fusion are used in all other experiments.

B Additional Ablation Studies

B.1 Different guiding saliency score.

In Table 1, we compare our prototype based guiding saliency score with an alternative choice, where we use the classification response of the ground truth

category produced by the recognizer to generate the NS pseudo labels, namely *response-based* guiding saliency score. It is shown that the prototype based score achieves better performance than response based one, which demonstrates that the prototype distance in feature space can offer more robust saliency cues.

Table 1. Performance of different guiding saliency score in FS module.

Guiding Saliency Score	mAP(%)
Response-based	74.1
Prototype-based	**74.7**

B.2 Different fusion strategies of two modules.

In Table 2, we show the impacts of different fusion strategies which are described in Section A.2. We can observe that various fusion strategies consistently improve the performance of single modules. The 'index union' fusion gets slightly higher performance than others thus we choose it in all our experiments.

Table 2. Comparison of various fusion strategies.

	Max	Mul	Add
Score	75.1	75.2	**75.3**
	Join	Inter	Union
Index	75.1	74.9	**75.5**

B.3 Different lightweight Feature Extractor in FEM.

In Table 3 we compare various backbones for lightweight feature extractor in FEM. As expected, the lightweight backbone with better performance is complementary to our method. Comparing with the ShuffleNetv2 [2] and MobileNetv2 [3] counterparts, our NSNet gets additional improvement on EfficientNet-b0 [5] with extra computation overhead. For fair comparisons with previous works, we use the MobileNetv2 as the lightweight feature extractor by default.

B.4 Different recognizer.

In Table 4 we investigate the impacts of various backbones of the recognizer, where we also report the training strategy of recognizer, *i.e.,* with TSN or without TSN. The model without TSN is trained by sampling one frame from each video. It is shown that our method is complementary to more advanced recognizers.

Table 3. Study on different backbones for lightweight extractor in FEM. FLOPs/f means FLOPs for each frame processed by the backbone.

Backbone	mAP	FLOPs/f
ShuffleNetv2	70.8	0.15G
MobileNetv2	75.5	0.31G
EfficientNet-b0	76.0	0.39G

Table 4. Study on different backbones for the recognizer. "Train" refers to training strategy, *viz.,* with TSN style training or without TSN style Training.

Backbone	Train	mAP(%)
ResNet-101	w/o TSN	75.5
ResNet-101	w/ TSN	80.8
ResNet-152	w/ TSN	83.0

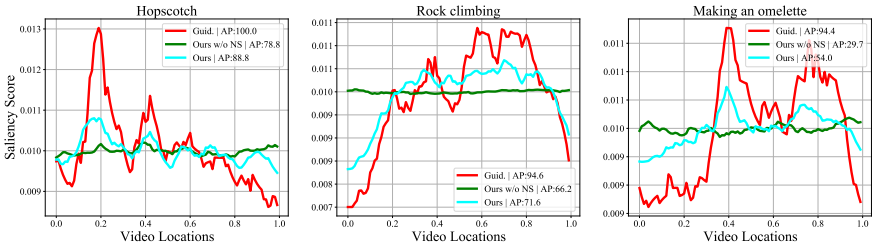

Fig. 1. Measured saliency distribution by different variants of our approach. We show the average value over all samples for a category for 3 variants, *viz.,* guiding saliency score (Guid.), our approach without non-saliency suppression (Ours w/o NS) and our approach(Ours). Our approach can generate saliency measurements close to Guid. for all 3 categories. However, Ours w/o NS only produce a relatively flat line on difficult categories like *rock climbing* and *making an omelette*, which shows it cannot handle saliency measurement on difficult categories without NS mechansim. Predicted saliency distributions are smoothed by Exponential Moving Average with weight of 0.8 for a better sense of trend.

C Qualitative Analysis

To take a closer look to how NS mechanism benefits saliency measurement, we present temporal saliency distributions produced by variants of our approach on the validation set of ActivityNet in Figure 1, which is computed by averaging the temporal saliency distribution of all samples within a given class. We adopt guiding saliency score (Guid.) as an alternative of saliency "ground truth", for it exploits labels of validation set and represents a upper bound of any sampler, achieving mAP of 96.3 (*v.s.* 75.3 achieved by NSNet). We can see that the saliency distribution of NSNet is more close to that of Guid. and achieving much higher AP than that of NSNet w/o NS on all three categories. As the discrimina-

tion difficulty increases, AP decreases dramatically from left sub-figures to right ones in Figure 1. In the easiest category *hopscotch*, both NSNet and NSNet w/o NS show similar saliency trends to Guid. to varying degrees. However, in much more difficult categories with low AP, like *rocking climbing* and *making a omelette*, the saliency scores measured by NSNet w/o NS tend to generate temporal uniform distributions and NSNet still shows highly similar trends to Guid., which demonstrates that proposed NS based supervisions can enhance robustness of saliency measurements in many scenarios.

References

1. Korbar, B., Tran, D., Torresani, L.: Scsampler: Sampling salient clips from video for efficient action recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (October 2019)
2. Ma, N., Zhang, X., Zheng, H.T., Sun, J.: Shufflenet v2: Practical guidelines for efficient cnn architecture design. In: Proceedings of the European conference on computer vision (ECCV). pp. 116–131 (2018)
3. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4510–4520 (2018)
4. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research **15**(56), 1929–1958 (2014), http://jmlr.org/papers/v15/srivastava14a.html
5. Tan, M., Le, Q.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: International Conference on Machine Learning. pp. 6105–6114. PMLR (2019)
6. Wu, W., He, D., Tan, X., Chen, S., Wen, S.: Multi-agent reinforcement learning based frame sampling for effective untrimmed video recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6222–6231 (2019)