A Baseline Study and Benchmark for Few-Shot Open-Set Action Recognition with Feature Residual Discrimination

Stefano Berti[0009−0002−0906−2251], Giulia Pasquale[0000−0002−7221−3553], and Lorenzo Natale[0000−0002−8777−5233]

Humanoid Sensing and Perception, Istituto Italiano di Tecnologia, Genoa, Italy
{stefano.berti, giulia.pasquale, lorenzo.natale}@iit.it

Supplementary Material

1 Implementation Details

SAFSAR 5-way 5-shot Since the authors [36] did not release the code, we implemented this method from scratch. We distributed the VideoMAE model over 4 GPUs. We used the pretrained weights of VideoMAE on the Kinetics-400 dataset provided by [38] for all experiments. We used 5 queries for each episodic task T_i. We fixed the weight λ for the $\mathcal{L}_2$ loss of SAFSAR to 0.1 . The values for α_{EOS} and α_{Disc} are fixed to 0.1 and 1000 respectively.
SAFSAR 5-way 1-shot In this case we did parallel training over 4 GPUs, so we consider 20 queries for each episodic task T_i. All parameters are set as in the 5-shot case, except for the learning rates and the number of iterations.
STRM 5-way 5-shot As for SAFSAR 5-way 5-shot, we trained the STRM model by distributing the feature extractor over 4 GPUs. The value of δ_{Disc} is fixed to 10. As in [38], for each episode we consider $4 \times K$ queries (4 queries for each class) and we accumulate gradients for 16 iterations before updating the parameters. For standard datasets –SSv2, HMDB, UCF– we used the same learning rates and number of iterations reported in the original paper [38].

Table 1: Training iterations and learning rates across datasets for SAFSAR 5-way 5-shot

	SSv2	Diving	NTU	HMDB	UCF
Iterations	50K	30K	15K	10K	10K
LR	$4e^{-6}$	$4e^{-6}$	$4e^{-6}$	$4e^{-6}$	$4e^{-6}$

Fig. 1: Qualitative comparison between the baseline SAFSAR and its adaptation to the open-set scenario with FR-Disc on the NTURGBD dataset (not cherry picked). We represent 4 equidistant frame among the 8 used for inference. We assume that a query is classified as *known* if the known score is above 50% and as *unknown* otherwise. A red label means that the prediction is incorrect, while a green label means that the prediction is correct.

Table 2: Training iterations and learning rates across datasets for SAFSAR 5-way 1-shot

	SSv2	Diving	NTU	HMDB	UCF
Iterations	50K	30K	15K	10K	10K
LR	$4e^{-7}$	$4e^{-7}$	$4e^{-7}$	$4e^{-8}$	$4e^{-8}$

1.1 MLS vs MSS

The importance of not-normalized logits. In Table 4, we report the performance gain of SAFSAR for open-set metrics using MLS instead of MSS, where the MLS values are reported in the paper. We note that the MLS score consistently outperforms the MSS score for implicit methods. This confirms the result described in [3, 39] for SAFSAR on spatio-temporal data. Indeed, the

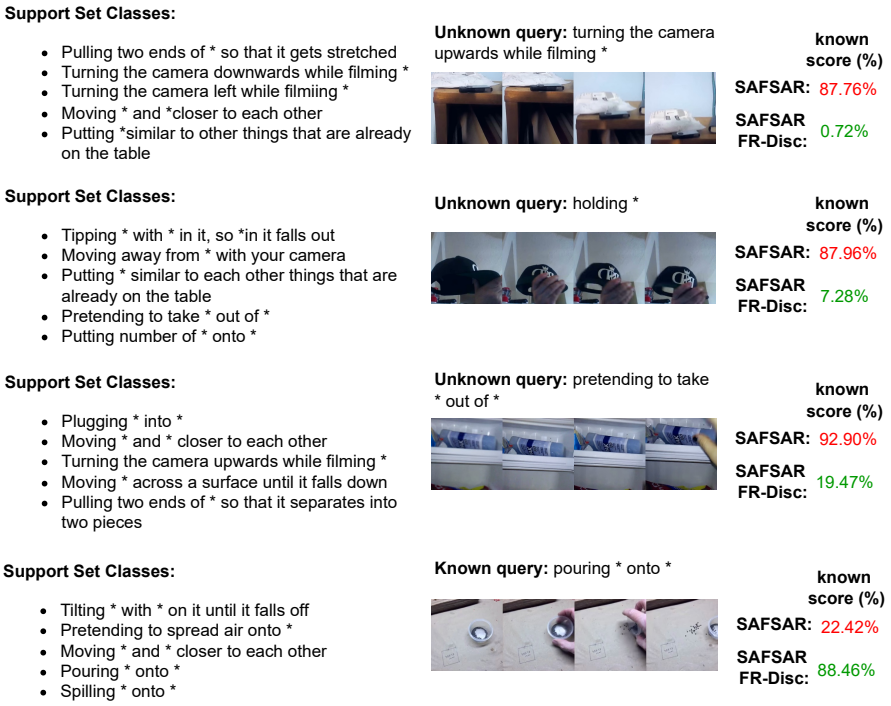


Fig. 2: Qualitative comparison between the baseline SAFSAR and its adaptation to the open-set scenario with FR-Disc on the SSv2 dataset (not cherry picked). We represent 4 equidistant frame among the 8 used for inference. We assume that a query is classified as *known* if the known score is above 50% and as *unknown* otherwise. A red label means that the prediction is incorrect, while a green label means that the prediction is correct.

Table 3: Training iterations and learning rates across datasets for STRM 5-way 5-shot

	SSv2	Diving	NTU	HMDB	UCF
Iterations	75K	75K	40K	20K	20K
LR	$1e^{-3}$	$1e^{-3}$	$1e^{-3}$	$1e^{-4}$	$1e^{-4}$

SoftMax operation discards the magnitude information that is contained in the logits $\{\hat{s}_{ij}\}_{j=1}^{K}$.

The value $\hat{u}_i$ can be derived from the logits $\{\hat{s}_i\}_j$ for SAFSAR because this model uses the cosine similarity as similarity function, whose range is in $[-1, 1]$ and can be interpreted directly as probability. We cannot apply the same ablation to STRM because it uses the negative norm of the difference between the query

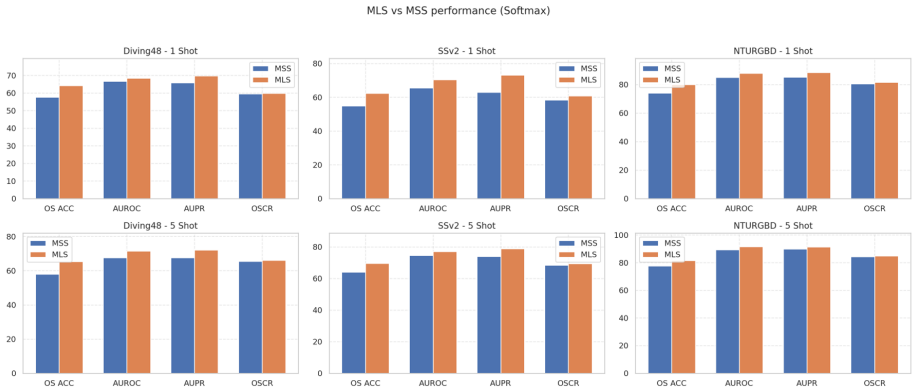


Fig. 3: Open-set performances comparison between MLS and MSS for the SAFSAR model.

Table 4: The gain for open set metrics when using MLS instead of MSS for implicit methods on SAFSAR. Gains are colored: green for positive and red for negative. "*" means that we trained that model for 1K iterations to prevent overfitting.

Dataset	OS-Method	OS ACC 1-shot	OS ACC 5-shot	AUROC 1-shot	AUROC 5-shot	AuPR 1-shot	AuPR 5-shot	OSCR 1-shot	OSCR 5-shot
Diving48	Softmax	+6.57	+7.44	+1.80	+3.93	+3.86	+4.54	+0.34	+0.71
	EOS	+3.37	+3.09	+0.04	-0.01	+1.96	+0.96	+0.04	-0.10
SSv2	Softmax	+7.43	+5.57	+4.90	+2.48	+10.05	+4.93	+2.32	+0.90
	EOS	+4.65	+2.28	+0.69	-0.08	+4.62	+2.89	+0.42	+0.30
NTURGBD	Softmax	+5.99	+3.93	+2.83	+2.15	+3.25	+1.40	+0.95	+0.50
	EOS	+3.59	+5.19	+0.58	+2.15	+1.59	+2.40	+0.23	+0.61
HMDB51	Softmax*	+14.44	+22.40	+4.02	+4.10	+3.34	+8.73	+0.44	+1.84
	EOS*	+16.23	+22.48	+5.71	+3.23	+4.67	+8.16	+1.15	+1.57
UCF101	Softmax*	+30.59	+41.25	+1.09	+0.46	+0.79	+1.27	+0.38	+0.34
	EOS*	+31.30	+39.32	+1.29	+0.32	+1.09	+0.70	+0.39	+0.16

features and the support set classes features, whose rage is in $[-\infty, 0]$ and thus can not be interpreted directly as probability.

2 Metrics Definition

2.1 AUROC

AUROC (Area Under the ROC Curve) is defined on T^{test} as:

$$\text{AUROC} = \int_0^1 \text{TPR}\, d\text{FPR},$$

where TPR (True Positive Rate) and FPR (False Positive Rate) are computed for the considered methods by varying the threshold $\tau \in (0, 1)$ (except for the Garbage Class that does not use τ) as:

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}.$$

and true positives TP are known queries classified as known, false positives FP are unknown queries classified as known, true negatives TN are unknown queries classified as unknown and FN are known queries classified as unknown.

2.2 AUPR

AUPR (Area Under the Precision-Recall Curve) is defined on T^{test} as:

$$\text{AUPR} = \int_0^1 \text{Precision}(\text{Recall})\, d\text{Recall},$$

where Precision and Recall are computed for the considered methods by varying the threshold $\tau \in (0, 1)$ (except for the Garbage Class that does not use τ) as:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{Recall} = \text{TPR}$$

2.3 FS ACC

FS ACC (Few-Shot Accuracy) or closed-set accuracy is defined on $T^{\text{test}}_{\text{known}}$ as:

$$\text{FS ACC} = \frac{1}{|T^{\text{test}}_{\text{known}}|} \sum_{(SS_i, Q_i^{\text{known}}) \in T^{\text{test}}_{\text{known}}} \mathbb{I}\left[\hat{y}_i = y_i\right] \tag{1}$$

where $\hat{y}_i$ is predicted class by $f(SS_i, x_i)$.

2.4 OS ACC

OS ACC (Open-Set Accuracy) is defined on T^{test}as:

$$\text{OS ACC} = \frac{1}{|T^{\text{test}}|} \sum_{(SS_i, Q_i) \in T^{\text{test}}} \mathbb{I}\left[\hat{a}_i = a_i\right] \tag{2}$$

where a_i is the true known/unknown class and $\hat{a}_i$ is the predicted known/unknown class for that sample. For the considered methods (except for GC) the threshold was fixed at $\tau = 0.5$.

2.5 OSCR

OSCR (Open-Set Classification Rate) is defined on $T^{\text{test}}_{\text{known}}$ as:

$$\text{CCR} = \frac{1}{|T^{\text{test}}_{\text{known}}|} \sum_{(SS_i, Q_i) \in T^{\text{test}}_{\text{known}}} \mathbb{I}\left[\hat{y}_i = y_i\right] \mathbb{I}\left[\hat{a}_i = a_i\right] \tag{3}$$

$$\text{OSCR} = \int_0^1 \text{CCR}\, d\text{FPR} \tag{4}$$

where CCR is the Correct Classification Rate on the P_{FS} problem. The OSCR thus represents the area under the CCR-FPR curve, by varying thethreshold $\tau \in (0, 1)$ for the considered methods (except for GC). This dual evaluation is particularly important for few-shot, where correctly identifying known actions and detecting unknown ones are both crucial requirements.

The integral of the metrics that are computed for all threshold values –AUROC, AUPR, OSCR– is approximated using the trapezoidal rule.

3 Discriminator Architecture

3.1 SAFSAR

Input shape: 1×768

- Linear Layer $768 \to 1536$
- ReLU
- Linear Layer $1536 \to 768$
- ReLU
- Linear Layer $768 \to 64$
- ReLU
- Linear Layer $64 \to 1$
- Sigmoid

3.2 STRM

Input shape: 28×1152

- Linear Layer $1152 \to 512$
- BatchNorm1d
- ReLU
- Dropout(0.3)
- Linear Layer $512 \to 256$
- BatchNorm1d
- ReLU
- Dropout(0.3)
- Linear Layer $256 \to 128$
- BatchNorm1d

- ReLU
- Dropout(0.3)
- Linear Layer $128 \rightarrow 64$
- ReLU
- Reshape $28 \times 64 \rightarrow 1792$
- Linear Layer $1792 \rightarrow 1$
- Sigmoid

4 Splits

4.1 Diving48

Training split
Forward_15som_NoTwis_PIKE, Inward_15som_NoTwis_TUCK,
Forward_25som_1Twis_PIKE, Back_3som_NoTwis_PIKE,
Reverse_35som_NoTwis_TUCK, Forward_25som_NoTwis_TUCK,
Forward_45som_NoTwis_TUCK, Reverse_25som_NoTwis_PIKE,
Forward_15som_2Twis_FREE, Back_15som_NoTwis_TUCK,
Forward_25som_NoTwis_PIKE, Back_35som_NoTwis_TUCK,
Reverse_25som_15Twis_PIKE, Reverse_15som_35Twis_FREE,
Forward_35som_NoTwis_PIKE, Back_25som_15Twis_PIKE,
Inward_35som_NoTwis_TUCK, Inward_15som_NoTwis_PIKE,
Back_2som_25Twis_FREE, Back_Dive_NoTwis_PIKE,
Back_25som_NoTwis_PIKE, Back_25som_25Twis_PIKE,
Inward_25som_NoTwis_PIKE, Reverse_15som_NoTwis_PIKE,
Back_15som_15Twis_FREE, Back_25som_NoTwis_TUCK,
Forward_1som_NoTwis_PIKE, Back_35som_NoTwis_PIKE,
Reverse_15som_05Twis_FREE, Back_3som_NoTwis_TUCK,
Back_15som_25Twis_FREE, Back_15som_NoTwis_PIKE
Testing split
Reverse_15som_15Twis_FREE, Back_Dive_NoTwis_TUCK,
Reverse_25som_NoTwis_TUCK, Forward_15som_1Twis_FREE,
Reverse_Dive_NoTwis_PIKE, Back_15som_05Twis_FREE,
Forward_25som_3Twis_PIKE, Reverse_Dive_NoTwis_TUCK,
Back_2som_15Twis_FREE, Inward_25som_NoTwis_TUCK,
Reverse_15som_25Twis_FREE, Forward_25som_2Twis_PIKE,
Inward_Dive_NoTwis_PIKE, Forward_35som_NoTwis_TUCK,
Forward_Dive_NoTwis_PIKE

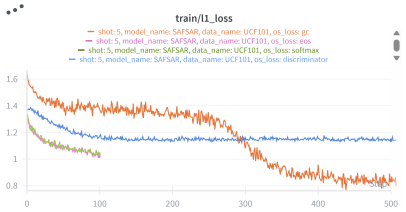
4.2 NTURGBD120

Training split
thumb_down, reach_into_pocket, taking_a_selfie, take_off_glasses,
wipe_face, cutting_nails, bounce_ball, phone_call,
apply_cream_on_hand, move_heavy_objects, writing,
snap_fingers, stand_up, jump_up, thumb_up, side_kick,
type_on_a_keyboard, open_a_box, play_magic_cube, reading,
juggle_table_tennis_ball, tear_up_paper, put_on_bag, throw,
put_on_glasses, arm_swings, shake_head, salute, take_off_bag,
kicking_something, throw_up_cap_hat, put_object_into_bag,
hush, rub_two_hands, toss_a_coin, apply_cream_on_face,
open_bottle, put_on_jacket, cross_toe_touch, ball_up_paper,
counting_money, cross_arms, flick_hair, nod_head, make_OK_sign,
cross_hands_in_front, shake_fist, take_off_a_shoe, sniff, pick_up,

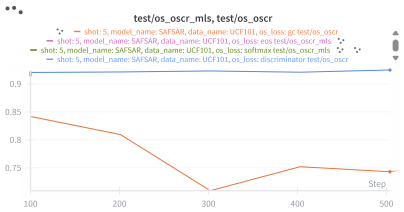
take_object_out_of_bag, drink_water, put_on_headphone, fold_paper, drop, squat_down, take_off_a_hatE

Testing split

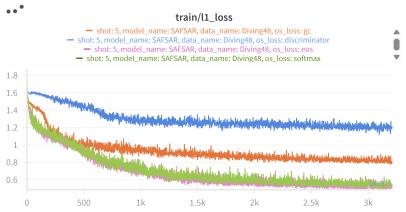
hopping, play_with_phone, make_victory_sign, brush_teeth, check_time_from_watch, arm_circles, clapping, point_to_something, put_palms_together, take_off_jacket, take_off_headphone, staple_book, sit_down, run_on_the_spot, shoot_at_basket, put_on_a_hat, brush_hair, butt_kicks, hand_waving, capitulate, put_on_a_shoe, cutting_paper, tennis_bat_swing, eat_meal, cheer_up



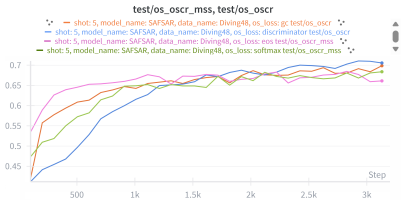
(a) 5-shot SAFSAR on UCF101 - training closed-set loss



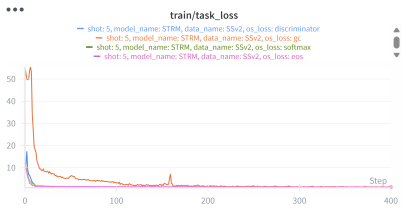
(b) 5-shot SAFSAR on UCF101 - test metric OSCR



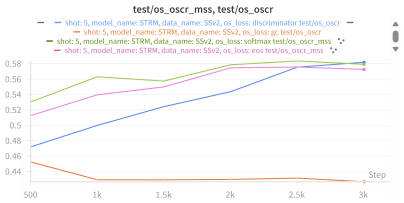
(c) 5-shot SAFSAR on Diving48 - training closed-set loss



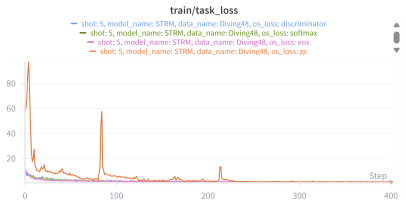
(d) 5-shot SAFSAR on Diving48 - test metric OSCR



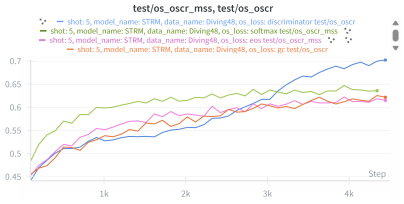
(e) 5-shot STRM on SSv2 - training closed-set loss



(f) 5-shot STRM on SSv2 - test metric OSCR



(g) 5-shot STRM on Diving48 - training closed-set loss



(h) 5-shot STRM on Diving48 - test metric OSCR

Fig. 4: Performances of SAFSAR and STRM on easy spatial dataset (UCF101) and fine-grained temporal datasets (SSv2, Diving48), showing both training closed-set loss and test OSCR metrics. We shot that in some cases (Subfigures (a) and (e)) the training Garbage Class loss converges, but the testing OSCR metric decreases (Subfigures (b) and (f)), indicating an overfitting. Instead, in other cases (Subfigures(c) and (g)) the training Garbage Class loss decreases, but the testing OSCR metric increase indicating a correct training. We argue that the Garbage prototype initialization plays an important role in the learning phase, and that this experiment would need different initialization or normalization techniques.