This supplementary material is organized as follows.

- Sec. 1 provides additional experiments. We highlight the conclusions here:
  - The scene bias problem is a general problem that exists in four different backbones (i.e., I3D [3], SlowFast [5], TSM [9], and TPN [12]).
  - Our SOAR successfully mitigates the scene bias, and achieves state-of-the-art OSAR performance with three additional backbones (i.e., SlowFast [5], TSM [9], TPN [12]), showing the generalization ability and debias ability of our method.

- Sec. 2 introduces additional implementation details.

1. Additional experimental results

1.1. Comparison with the state-of-the-art

In the main paper, we have shown that our SOAR outperforms previous methods in terms of lower scene bias and higher OSAR performance with the I3D backbone [3]. To validate that the superiority of SOAR is not tied to a specific backbone, we carry out experiments with different backbones, i.e., SlowFast [5], TSM [9], and TPN [12]. Furthermore, to analyze how the scene distance affects the overall OSAR and closed-set classification performance, we carry out the scene bias analysis experiments using Open maF1 as the metric.

Scene bias analysis with different backbones. Fig. 1, Fig. 2, and Fig. 3 shows the scene bias analysis experiments with TPN [12], TSM [9], and SlowFast [5] backbones, respectively. We make the following observations. (1) All figures show the same tendency: known actions with unfamiliar scenes (right part of the left figures) and unknown actions with familiar scenes (left part of the right figures) are hard to recognize. This conclusion holds for all backbones, indicating that it is a general problem rather than a backbone-specific problem. (2) Our SOAR achieves lower scene bias in both scenarios with all backbones, showing the generalization ability and debias ability of our method. Especially, our SOAR achieves better OSAR performance when the closed-set testing set exhibits dissimilar scene to the training set (i.e., the right part of Fig. 1a, Fig. 2a, and Fig. 3a), and when the open-set testing set exhibits similar scene to the training set (i.e., the left part of Fig. 1b, Fig. 2b, and Fig. 3b). Such a performance advantage shows that our method successfully avoids the misleading of scene information, and further demonstrates its debias ability. We note that this ability is critical when the testing environment is different from the training environment.


To demonstrate the generalization ability of our proposed AdRecon and AdaScls, we further show the results of the
(a) Analysis on the known action in unfamiliar scene scenario.
Figure 1. Quantitative scene bias analysis using UCF101 [11] as known and MiTv2 [10] as unknown with the TPN backbone [12].

(b) Analysis on the unknown action in familiar scene scenario.
Figure 1. Quantitative scene bias analysis using UCF101 [11] as known and MiTv2 [10] as unknown with the TPN backbone [12].

(a) Analysis on the known action in unfamiliar scene scenario.
Figure 2. Quantitative scene bias analysis using UCF101 [11] as known and MiTv2 [10] as unknown with the TSM backbone [9].

(b) Analysis on the unknown action in familiar scene scenario.
Figure 2. Quantitative scene bias analysis using UCF101 [11] as known and MiTv2 [10] as unknown with the TSM backbone [9].

(a) Analysis on the known action in unfamiliar scene scenario.
Figure 3. Quantitative scene bias analysis using UCF101 [11] as known and MiTv2 [10] as unknown with the SlowFast backbone [5].

(b) Analysis on the unknown action in familiar scene scenario.
Figure 3. Quantitative scene bias analysis using UCF101 [11] as known and MiTv2 [10] as unknown with the SlowFast backbone [5].
Table 1. Comparison with state-of-the-art methods with different backbones. All methods are trained on UCF101 [11], and evaluated on two different open sets where unknown samples are from HMDB51 [8] and MiTv2 [10], respectively.


2. Additional implementation details

For the TPN backbone [12], we follow DEAR [1] to use the slow-only version for feature extraction. For the TSM backbone [9], we use the default setting in MMAction2 [4] for feature extraction following DEAR [1]. For the SlowFast backbone [5], we interpolate the extracted spatio-temporal features from the slow and fast pathways to the same size, and concatenate them in the channel dimension as the final output.

spatio-temporal feature $F$.

We note that our reported results are different from those reported in DEAR [1] as they use binarized prediction for the AUC prediction, which only has one operating point on the ROC curve, while we use the raw prediction for the AUC computation.

References