Bayesian Uncertainty Attribution-Guided Fine-Tuning for Open-Set Action Recognition

Shehan Senavirathna[1], Hongji Guo[1], and Qiang Ji[1⋆]

Rensselaer Polytechnic Institute, Troy, NY 12180, USA
{senavk,guoh11,jiq}@rpi.edu

Abstract. Open-set action recognition (OSAR) requires a model to classify known actions while rejecting unseen ones. Existing methods typically use predictive uncertainty only at inference time, treating it as a rejection score after representation learning is complete. This leaves the underlying representation unchanged and allows spurious background or context cues to remain embedded in the model, which weakens known–unknown separation. We propose a staged uncertainty-guided fine-tuning framework that uses ensemble-derived epistemic uncertainty not only for rejection, but also as a training signal for representation refinement. The framework proceeds in three stages: motion-guided ensemble training to bias learning toward action-relevant evidence, in-distribution epistemic regularization to improve uncertainty reliability on known samples, and uncertainty attribution mask-based input attention (UAM-IA), which converts uncertainty attribution maps into input attention masks for refinement. To improve deployment efficiency, we further distill the refined ensemble into a single student model trained to approximate both the ensemble predictive distribution and the ensemble-derived epistemic uncertainty. Under a controlled OSAR protocol with UCF101 as the known set and HMDB51 and MiT-v2 as unknown sources after removing overlapping classes, the proposed method improves known–unknown separation across multiple backbones. The distilled student preserves much of this benefit while enabling efficient single-model inference.

Keywords: Bayesian Deep Learning · Open-Set Action Recognition · Uncertainty Quantification · Uncertainty Attribution · Video Understanding

1 Introduction

Recent human action recognition models have achieved strong performance in closed-set benchmarks, where the set of action categories observed at test time is assumed to match the training label space. In realistic deployments, however, this assumption rarely holds. A model trained only on known actions will inevitably encounter novel, out-of-distribution (OOD) actions, and standard classifiers often remain overly confident on such unfamiliar inputs. For example, a model

⋆ Corresponding author.

trained to recognize *playing golf* may confidently misclassify an unseen *mowing the lawn* video if it relies on shared scene context such as grass or outdoor background cues rather than the action itself. This reliance on contextual shortcuts makes confidence-based rejection brittle and limits the reliability of closed-set models in open-world settings.

Open-set action recognition (OSAR) addresses this challenge by jointly requiring accurate recognition of known actions and reliable rejection of unknowns. A central difficulty in OSAR is uncertainty estimation: the model must not only predict a class label, but also indicate when its prediction is unreliable. Recent methods have improved OSAR by introducing uncertainty-aware scoring mechanisms, including evidential formulations and scene-debiasing strategies. However, many of these approaches primarily use uncertainty as an inference-time decision signal. As a result, uncertainty is measured after representation learning, but is not explicitly used to refine the learned features that gave rise to that uncertainty. Consequently, nuisance factors such as background bias, irrelevant motion, and spatiotemporal ambiguity can remain embedded in the representation, degrading known–unknown separation.

In this work, we use uncertainty not only as a diagnostic quantity, but also as a training signal for representation refinement. We propose a staged uncertainty-guided fine-tuning framework for OSAR. First, we train a deep ensemble with motion-prior alignment, using optical flow only as a weak training-time guide to bias the initial representation away from static contextual shortcuts and toward action-relevant evidence. Second, we refine the ensemble by explicitly regularizing epistemic uncertainty on known training samples, which improves the stability of the uncertainty estimates. Third, we introduce **Uncertainty Attribution Mask-based Input Attention (UAM-IA)**, where ensemble-derived epistemic uncertainty is backpropagated to the input to produce attribution maps, which are then converted into attention masks for uncertainty-guided refinement. Because deep ensembles are expensive at deployment, we additionally distill the refined ensemble into a single student model. The student is trained to match both the ensemble mean predictive distribution and the ensemble-derived epistemic uncertainty, allowing deployment with a single forward pass while retaining much of the refined ensemble's predictive and rejection behavior.

We evaluate the proposed method under a controlled OSAR protocol using UCF101 as the known set and HMDB51 and MiT-v2 as unknown sources, with semantically overlapping classes removed. We compare the full refinement pipeline against matched deep-ensemble baselines and compare the distilled student against representative single-model OSAR baselines. Our experiments show that staged uncertainty-guided refinement consistently improves known–unknown separation, with the largest incremental gain obtained after adding the Stage 3 UAM-IA refinement.

Our contributions are summarized as follows:

- We propose a staged uncertainty-guided fine-tuning framework for open-set action recognition that uses predictive uncertainty not only for rejection, but also for representation refinement.

- We introduce UAM-IA, which converts uncertainty attribution maps into input attention masks for uncertainty-guided refinement toward more reliable spatiotemporal evidence.
- We present an efficient deployment strategy that distills the refined ensemble into a single student model trained to approximate both the ensemble predictive distribution and the ensemble-derived epistemic uncertainty.
- We establish a controlled evaluation protocol with matched baselines and non-overlapping open-set splits, and show through ablations that the proposed refinement stages improve open-set recognition performance.

2 Related Work

2.1 Open-set Action Recognition and Open-vocabulary Action Recognition

Existing OSAR methods follow three complementary directions. First, semantic and representation-based methods strengthen class structure or introduce auxiliary knowledge. STECN [13] uses semantic-textual expansion and class normalization, USE [16] explores richer semantic information, and Prototypical Similarity Learning (PSL) [7] preserves instance- and class-specific information for better known–unknown separation. Second, reconstruction and debiasing methods reduce reliance on nuisance context; SOAR [38], for example, combines evidential prediction with scene-debiasing reconstruction. Third, uncertainty-aware methods use predictive uncertainty for unknown rejection. DEAR [2] formulates single-label OSAR through evidential deep learning, while MULE [39] extends evidential modeling to multi-label action recognition. In contrast, our purely visual, single-label framework estimates ensemble-derived epistemic uncertainty and feeds it back into representation learning through explicit regularization and attribution-guided input attention before distilling the refined behavior into an RGB-only student.

OSAR should also be distinguished from open-vocabulary action recognition (OVAR). OVAR methods use large-scale vision-language pretraining and textual semantics to recognize actions beyond the supervised vocabulary. Representative approaches include ActionCLIP [33], Open-VCLIP [34], FROSTER [17], and Open-MeDe [37]. Although both OSAR and OVAR address categories outside the training labels, they operate under different assumptions. OVAR relies on broad external semantic priors learned from web-scale image-text corpora, whereas OSAR focuses on reliable recognition and rejection under a closed-world visual training regime without access to such external priors. Our work, therefore, targets the OSAR setting: we do not assume vision-language supervision at training or test time, and instead study how predictive uncertainty can be estimated and actively exploited within a purely visual action recognition pipeline.

2.2 Uncertainty Quantification, Attribution, and Uncertainty-guided Refinement

Uncertainty quantification (UQ) is central to robust open-set recognition because the model must not only predict a label, but also indicate when that prediction

is unreliable. Deep ensembles [22] provide a strong estimate of epistemic uncertainty through predictive disagreement, while evidential formulations [2, 29] model class evidence and produce uncertainty-aware predictions through evidential objectives. A related line of work studies uncertainty attribution (UA), where the goal is to identify which parts of the input or representation are responsible for uncertain predictions; prior work, such as CLUE [1] and gradient-based uncertainty attribution [32], shows that uncertainty can be localized and interpreted rather than treated only as a scalar output score. Other works further incorporate predictive uncertainty into training through uncertainty-aware reweighting [4, 23] of the supervised loss or as a regularization [10]. Our framework builds on these directions but combines them in a staged manner tailored to OSAR. Specifically, we first use ensemble-derived epistemic uncertainty as an explicit regularization signal to reduce in-distribution epistemic uncertainty during fine-tuning, and then use uncertainty attribution maps to construct input attention masks for uncertainty-guided refinement. Thus, uncertainty is used not only for rejection or interpretation, but also as an explicit signal for representation refinement during training.

2.3 Efficient Approximation of Ensemble-derived Uncertainty

Deep ensembles provide strong epistemic uncertainty estimates, but their inference cost is high for video recognition because each prediction requires multiple backbone forward passes [22]. This motivates an efficient approximation strategy related to knowledge distillation [15, 19], where the goal is to preserve the behavior of a stronger teacher model in a cheaper deployment-time student. Our setting differs from standard distillation in two ways. First, the teacher is a refined uncertainty-aware ensemble rather than a single network. Second, the student is trained to match not only the ensemble predictive distribution over known classes, but also the ensemble-derived epistemic uncertainty through an auxiliary scalar regression objective. This makes the distilled student suitable for efficient open-set deployment, where both classification quality and reliable unknown rejection are required [11, 25].

3 Method

In this section, we first present the overall framework of the proposed method in Sec. 3.1. We then introduce the problem setup and uncertainty formulation in Sec. 3.2. Next, we describe the three main components of our method: motion-guided ensemble training in Sec. 3.3, epistemic regularization in Sec. 3.4, and uncertainty attribution mask-based input attention in Sec. 3.5. Finally, we describe the efficient single-model deployment variant in Sec. 3.6 and summarize the full training and inference procedure in Sec. 3.7.

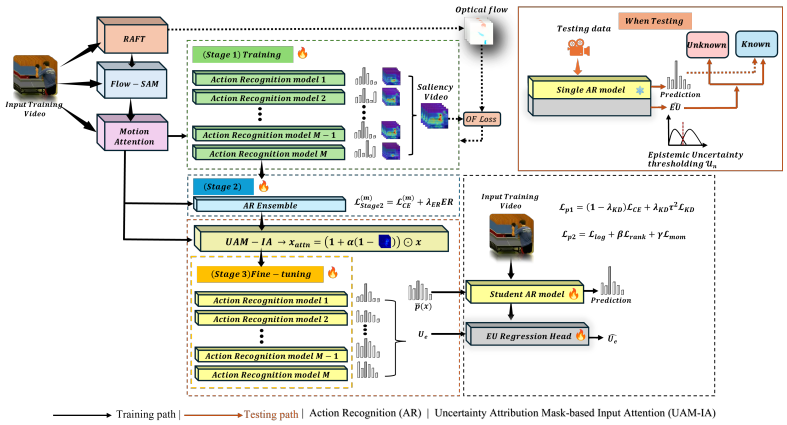


Fig. 1: Overview of the proposed framework. The model is refined sequentially through motion-guided training (Stage 1), in-distribution epistemic regularization (Stage 2), and uncertainty attribution mask-based input attention (Stage 3). The refined ensemble is distilled into a single student, comprising a classification branch and an auxiliary epistemic uncertainty (EU) head for efficient deployment.

3.1 Overall Framework

The overall framework is illustrated in Fig. 1. Our goal is to use predictive uncertainty not only for open-set rejection, but also as a training signal for representation refinement. To this end, we train a deep ensemble of action recognition models and refine it in three sequential stages. In **Stage 1**, we use optical flow only as a weak training-time motion prior to align the RGB classifier's class-decision saliency with motion-relevant regions, thereby providing a motion-aware initialization for the later uncertainty-refinement stages. In **Stage 2**, we refine the ensemble by explicitly regularizing epistemic uncertainty on in-distribution training data. In **Stage 3**, we backpropagate uncertainty to the input space, obtain uncertainty attribution maps, and convert them into input attention masks that guide fine-tuning toward more reliable spatiotemporal evidence.

Although the ensemble provides strong uncertainty estimates, multi-model inference is expensive for video recognition. Therefore, after the three-stage refinement, we distill the refined ensemble into a single student model that approximates both the ensemble predictive distribution and the ensemble-derived epistemic uncertainty. Thus, the ensemble is used during training to produce robust uncertainty supervision, while deployment is performed using a single model.

3.2 Preliminaries

Let $\mathcal{X}$ denote the space of video clips and $\mathcal{Y} = \{1, \ldots, C\}$ the label set of known action classes available during training. In OSAR, the test distribution additionally contains unknown classes $\mathcal{U}$ such that $\mathcal{U} \cap \mathcal{Y} = \emptyset$.

During training, we maintain an ensemble $\{f_m\}_{m=1}^{M}$, where each model $f_m : \mathcal{X} \rightarrow \mathbb{R}^C$ outputs logits over the known label space. For an input clip $x \in \mathcal{X}$,

model m produces logits $\mathbf{z}^{(m)}(x) = f_m(x)$ and class probabilities

$$g_i^{(m)}(x) = \frac{\exp(z_i^{(m)}(x))}{\sum_{j=1}^{C} \exp(z_j^{(m)}(x))}, \tag{1}$$

where $\mathbf{g}^{(m)}(x) = [g_1^{(m)}(x), \ldots, g_C^{(m)}(x)] \in [0,1]^C$.

The ensemble mean predictive distribution is

$$\bar{\mathbf{g}}(x) = \frac{1}{M} \sum_{m=1}^{M} \mathbf{g}^{(m)}(x) \tag{2}$$

Following [32], we decompose predictive uncertainty into total, aleatoric, and epistemic components. For class i, these are defined as

$$U_{t,i}(x) = -\bar{g}_i(x) \log \bar{g}_i(x), \tag{3a}$$

$$U_{a,i}(x) = \frac{1}{M} \sum_{m=1}^{M} \left[-g_i^{(m)}(x) \log g_i^{(m)}(x) \right], \tag{3b}$$

$$U_{e,i}(x) = U_{t,i}(x) - U_{a,i}(x), \tag{3c}$$

where $\bar{g}_i(x)$ is the i-th component of the ensemble mean prediction. The clip-level uncertainties are then

$$U_t(x) = \sum_{i=1}^{C} U_{t,i}(x), \qquad U_a(x) = \sum_{i=1}^{C} U_{a,i}(x), \qquad U_e(x) = \sum_{i=1}^{C} U_{e,i}(x) \tag{4}$$

In our framework, $U_e(x)$ is the primary uncertainty signal used for refinement and for open-set rejection. During the three-stage training procedure, $U_e(x)$ is estimated from the ensemble and used as the uncertainty signal for refinement. In the deployment variant described later, the same ensemble-derived uncertainty is cached and used as a supervision target for distilling a single student model.

3.3 Stage 1: Motion-guided Ensemble Training

Human actions are primarily defined by motion, yet video recognition models often exploit static contextual shortcuts such as background scene or camera bias. To reduce this failure mode, we introduce motion guidance during ensemble training. We first estimate optical flow using RAFT [31]. To obtain stable motion-focused regions, we further use FlowI-SAM [35], which combines optical-flow cues with Segment Anything (SAM) [18] to produce moving-object segmentations. Let Q_t denote the resulting segmentation mask for frame t. We normalize it to obtain a per-frame motion attention map

$$\mathrm{MA}_t = \sigma(Q_t), \tag{5}$$

where $\sigma(\cdot)$ denotes normalization to $[0,1]$. These masks are used only during training to define motion-prior alignment supervision. Stage 1 does not train the classifier to predict optical flow. Raw flow can mark motion, including camera or background motion, but it does not indicate which regions the RGB classifier actually uses for its action decision. Grad-CAM, in contrast, identifies the classifier's class-decision evidence. We therefore use the RAFT/FlowI-SAM output as

a weak motion prior and align the classifier evidence with this prior, rather than treating flow as an additional input modality or prediction target. Let S denote the model saliency map obtained by Grad-CAM [28], and let O denote the normalized motion-prior map obtained from the FlowI-SAM mask and resized to the saliency-map resolution. We impose the alignment loss

$$\mathcal{L}_{\mathrm{OF}} = \mathrm{MSE}(S, O). \tag{6}$$

Each ensemble member is trained independently with different random seeds, classifier-head initializations, minibatch orders, and stochastic augmentations using

$$\mathcal{L}_{\mathrm{S1}}^{(m)} = \mathcal{L}_{\mathrm{CE}}^{(m)} + \lambda_{\mathrm{OF}} \mathcal{L}_{\mathrm{OF}}, \tag{7}$$

where $\mathcal{L}_{\mathrm{CE}}^{(m)}$ is the standard cross-entropy loss for model m. Because strong camera motion can make the prior noisy, optical flow is used only as a weak training-time guide; it is neither the rejection score nor a test-time modality. No optical flow or Grad-CAM computation is used at inference.

3.4 Stage 2: Epistemic Regularization

The training set contains only known classes. Therefore, high epistemic uncertainty on these samples indicates unstable or incomplete in-distribution evidence. In Stage 2, we explicitly reduce this uncertainty by regularizing the current ensemble's epistemic uncertainty during training. For a mini-batch $\{(x_k, y_k)\}_{k=1}^{B}$, we compute the ensemble-based epistemic uncertainty $U_e(x_k)$ using Eq. (4) from the current ensemble predictions. We then define the epistemic regularization term as

$$\mathcal{L}_{\mathrm{ER}} = \frac{1}{B} \sum_{k=1}^{B} U_e(x_k) \tag{8}$$

Each ensemble member is refined using

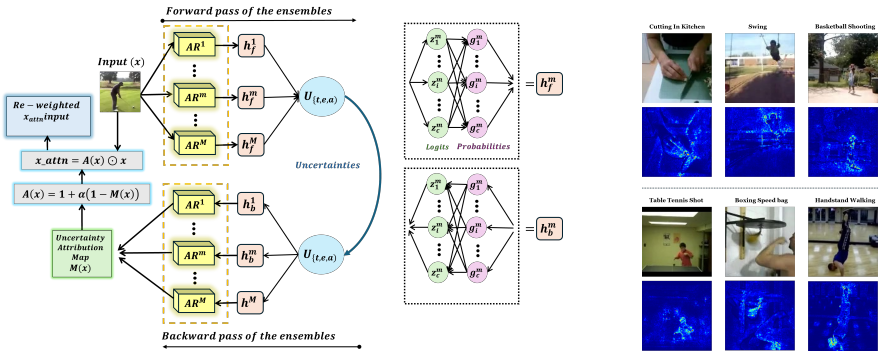
$$\mathcal{L}_{\mathrm{S2}}^{(m)} = \mathcal{L}_{\mathrm{CE}}^{(m)} + \lambda_{\mathrm{ER}} \mathcal{L}_{\mathrm{ER}} \tag{9}$$

This stage encourages lower epistemic uncertainty on known training samples while preserving the standard classification objective. Importantly, Stage 2 uses only in-distribution training data and does not require access to unknown samples during optimization. Section 4.5 compares total-, aleatoric-, and epistemic-uncertainty regularizers and examines how the three uncertainty components change after refinement.

3.5 Stage 3: Uncertainty Attribution Mask-based Input Attention

While Stage 2 reduces epistemic uncertainty at the sample level, it does not identify which input regions are most associated with that uncertainty. In Stage 3, we make uncertainty actionable through uncertainty attribution mask-based input attention (UAM-IA).

At the beginning of each epoch, we run the current ensemble (initialized from the Stage 2 model and updated during Stage 3) over the training set and compute



(a) UAM-IA method **(b)** Qualitative examples

Fig. 2: Overview of the UAM-IA framework and qualitative examples of uncertainty-guided attention, for six action samples from UCF101 [30]. The top row shows representative RGB frames from the original videos, and the bottom row shows the corresponding attention masks produced in Stage-3.

the epistemic uncertainty $U_e(x)$ for each training sample. Following the gradient-based uncertainty attribution formulation of [32], we backpropagate $U_e(x)$ to the input space and use the gradient magnitude as the attribution signal. Let $M(x) \in [0, 1]$ denote the normalized uncertainty-attribution map derived from this signal. We then form a smooth residual attention map

$$A(x) = 1 + \alpha\big(1 - M(x)\big), \tag{10}$$

where $\alpha \geq 0$ controls the strength of the modulation. Since $M(x) \in [0, 1]$, the attention values satisfy $A(x) \in [1, 1 + \alpha]$. Thus, regions with lower uncertainty attribution are softly amplified, while regions with higher uncertainty attribution remain close to their original scale. Highly attributed regions may still contain occluded hands, actor–object interactions, or subtle motion, so suppressing them can remove useful evidence, whereas amplifying them can emphasize unstable evidence. UAM-IA therefore keeps such regions visible while emphasizing lower-attribution evidence. Fig. 2 summarizes the UAM-IA computation and shows representative masks, illustrating how the detached attention map preserves the input while emphasizing lower-uncertainty evidence. Section 4.5 compares this form with high-attribution amplification and suppression, a frame-difference mask, and a sweep over α. Using this detached attention map, the attended input for the current epoch is defined as

$$x_{\text{attn}} = A(x) \odot x, \tag{11}$$

where $\odot$ denotes element-wise multiplication. Each ensemble member is then fine-tuned on the attended input using the standard classification loss:

$$\mathcal{L}_{\text{S3}}^{(m)} = \mathcal{L}_{\text{CE}}^{(m)}(x_{\text{attn}}, y) \tag{12}$$

The attribution maps are recomputed at the start of every epoch and then kept fixed during that epoch. This avoids differentiating through the attribution-generation step during parameter updates while still allowing the attention masks to adapt as training progresses.

3.6 Efficient Single-model Distillation

Although the refined ensemble provides strong open-set performance, its inference cost grows linearly with the number of ensemble members. To obtain an efficient deployment model, we distill the refined ensemble into a single student network.

After Stage 3, we run the refined ensemble on the training split and cache two targets for each sample: the ensemble mean class distribution $\bar{\mathbf{g}}(x)$ and the ensemble epistemic uncertainty $U_e(x)$, computed as in Eq. (4). The student is then trained in two phases.

Phase 1 (classification distillation). The student is initialized with Kinetics-pretrained weights and trained to match both the ground-truth label and the cached ensemble class distribution. Let $\mathcal{L}_{\mathrm{CE}}$ denote the standard classification loss and let $\mathcal{L}_{\mathrm{KD}}$ denote the KL-divergence between the teacher and student class distributions. The phase-1 objective is

$$\mathcal{L}_{\mathrm{P1}} = \mathcal{L}_{\mathrm{CE}} + \lambda_{\mathrm{KD}} \tau^2 \mathcal{L}_{\mathrm{KD}}. \tag{13}$$

This phase learns the student backbone and classification branch.

Phase 2 (epistemic uncertainty regression). We initialize from the Phase 1 student, freeze the Phase-1 backbone and classification branch, and train only an auxiliary uncertainty head. Let $\phi_s(x)$ denote the student feature, and let $\hat{U}_e(x) = r_s([\phi_s(x), \mathbf{g}_s(x)])$ be the predicted epistemic uncertainty, where $[\cdot,\cdot]$ denotes concatenation. We optimize

$$\mathcal{L}_{\mathrm{P2}} = \mathcal{L}_{\log} + \beta \mathcal{L}_{\mathrm{rank}} + \gamma \mathcal{L}_{\mathrm{mom}}, \tag{14}$$

where $\mathcal{L}_{\log} = \|\log(1+\hat{U}_e) - \log(1+U_e)\|_2^2$, $\mathcal{L}_{\mathrm{rank}}$ is a pairwise margin-ranking loss computed within each mini-batch from the cached teacher U_e targets, and $\mathcal{L}_{\mathrm{mom}} = (\mu_B(\hat{U}_e) - \mu_B(U_e))^2 + (\sigma_B(\hat{U}_e) - \sigma_B(U_e))^2$ matches the batch mean and standard deviation of the predicted and teacher uncertainty values.

At inference, only the student is used. The classification branch predicts the posterior over known classes, and the auxiliary branch outputs $\hat{U}_e$, which is thresholded for known/unknown rejection. This provides efficient single-model deployment while preserving much of the refined ensemble's predictive and uncertainty behavior.

3.7 Training and Inference Procedure

Training applies the three refinement stages sequentially: motion-guided ensemble training (Stage 1), epistemic regularization (Stage 2), and epoch-wise attribution-guided input attention (Stage 3). After finalizing the ensemble, we cache its mean class distribution and epistemic uncertainty on the training split. These targets supervise the student in two phases: classification distillation and uncertainty-head regression.

At inference, only the distilled student is used. The classifier predicts class probabilities over the known classes, and the auxiliary head predicts the epistemic uncertainty estimate $\hat{U}_e(x)$. A threshold on $\hat{U}_e(x)$ is used for known–unknown rejection. Optical flow and full ensemble inference are not used at test time.

4 Experiments

4.1 Datasets and protocol

Following prior OSAR works [2, 38], we evaluate on standard action-recognition benchmarks including UCF101 [30], HMDB51 [21], and MiT-v2 [26]. We use split 1 of the UCF101 training partition (9,537 videos from 101 classes) and hold out a stratified subset exclusively for hyperparameter and rejection-threshold selection; the remaining clips are used for optimization. The official UCF101 split-1 test partition is used as the known test set. HMDB51 and MiT-v2 are used only as sources of unknown samples for final reporting. To ensure a strict open-set protocol, we remove semantically overlapping classes between the known and unknown sets before evaluation. After filtering, the HMDB51 unknown split contains 41 non-overlapping classes with 1,230 validation videos, and the MiT-v2 unknown split contains 277 non-overlapping classes with 27,700 validation videos. The overlap-removal class lists are given in the supplementary material.

4.2 Evaluation metrics

For closed-set recognition, we report top-1 accuracy on the held-out known test set. For open-set detection, we report AUROC, FPR@95, and TPR@10. For $(C + 1)$-way open-set recognition, we additionally report the open macro F1 score (open maF1) following DEAR [2], together with its mean and standard deviation across runs. To further analyze known–unknown separation, we report the Hellinger distance [5, 9] between the known and unknown uncertainty distributions, where larger values indicate better separation. We use calibration metrics such as Expected Calibration Error (ECE) and Negative Log-Likelihood (NLL) in the ablation analysis to study the effect of the proposed refinement stages on in-distribution prediction reliability.

4.3 Implementation details

Our framework is implemented in MMAction2 [8] using PyTorch [27]. We train an ensemble of $M = 10$ action-recognition models initialized from Kinetics-400 pretraining [6]. Diversity is induced by different random seeds, classifier-head initialization, minibatch order, and stochastic augmentation. We optimize all stages with Adam using an initial learning rate of 10^{-3}. Training is performed sequentially for 50 epochs in total, with $n_1 = 30$ epochs for Stage 1, $n_2 = 10$ epochs for Stage 2, and $n_3 = 10$ epochs for Stage 3. For motion estimation in Stage 1, we use RAFT [31]. The backbone-specific $(\lambda_{\mathrm{OF}}, \lambda_{\mathrm{ER}}, \alpha)$ settings are $(0.65, 0.55, 1.00)$ for TPN, $(0.60, 0.50, 1.12)$ for TSM, $(0.55, 0.55, 1.14)$ for SlowFast, and $(0.60, 0.45, 1.02)$ for I3D. After ensemble refinement, the student is trained for 15 epochs in each distillation phase. Batch sizes, additional distillation hyperparameters, threshold-selection details, complementary backbone uncertainty distributions, qualitative examples, and failure cases are provided in the supplementary material.

4.4 Main Results

We evaluate the proposed method in two settings: a deployment-time single-model comparison, and a matched deep-ensemble comparison against representative OSAR baselines.

Single-model comparison. Table 1 evaluates the distilled deployment model, which uses one student for classification and epistemic-uncertainty prediction. All methods use RGB-only test-time input; optical flow is offline guidance only in Stage 1. The student is strongest on TPN and remains competitive across the other backbones with a single forward pass.

Matched deep-ensemble comparison. Table 2 compares our method against a vanilla deep ensemble while controlling for backbone, ensemble size, and evaluation protocol. Here, *Vanilla DE* denotes $M = 10$ independently trained members using the same backbone and base training recipe, but without the proposed Stage 1–3 refinements. Across all backbones, the proposed refinement improves open-set metrics (AUROC, FPR@95, TPR@10, and open maF1) while maintaining strong closed-set accuracy. This indicates that the gains arise from the proposed training strategy rather than from ensembling alone.

Table 1: Single-model comparison with representative OSAR baselines across four backbones: TPN [36], TSM [24], SlowFast [12], and I3D [6]. The compared baselines are SoftMax entropy thresholding, OpenMax [3], MC Dropout [14], BNN-SVI [20], DEAR [2], and SOAR [38]. Models are trained on UCF101 [30] and evaluated with UCF101 as the known set and MiT-v2 [26] or HMDB51 [21] as unknown sources under the open-set protocol.

Backbone	Method	UCF101+MiT-v2				UCF101+HMDB51				Acc.
		AUROC↑	FPR@95↓	TPR@10↑	Open maF1↑	AUROC↑	FPR@95↓	TPR@10↑	Open maF1↑	
TPN	SoftMax	43.36	97.82	8.89	55.01 ± 0.32	44.92	97.33	6.42	72.31 ± 0.12	92.00
	OpenMax	60.02	73.93	23.02	65.31 ± 0.19	62.65	64.23	19.30	65.32 ± 0.12	55.37
	MC-Drop	90.86	32.59	72.51	71.96 ± 0.19	84.89	64.76	57.19	77.47 ± 0.14	91.28
	BNN-SVI	90.23	32.23	67.86	69.57 ± 0.19	84.93	66.82	58.82	75.38 ± 0.15	90.11
	DEAR	90.31	33.67	68.32	73.57 ± 0.19	85.16	62.72	57.14	84.82 ± 0.14	92.02
	SOAR	91.45	30.96	74.37	74.48 ± 0.21	86.67	61.02	60.62	85.43 ± 0.14	92.63
	Ours	**93.10**	**24.64**	**83.84**	**78.04 ± 0.18**	**88.50**	**53.85**	**73.51**	**88.97 ± 0.17**	**95.98**
TSM	SoftMax	46.39	94.45	9.35	54.29 ± 0.34	44.58	98.44	9.32	76.29 ± 0.19	92.11
	OpenMax	61.49	58.90	12.49	64.30 ± 0.25	60.97	63.83	10.46	64.39 ± 0.17	53.48
	MC-Drop	87.87	41.69	61.22	65.67 ± 0.26	84.82	63.67	63.53	75.68 ± 0.20	92.15
	BNN-SVI	89.92	40.42	72.66	65.94 ± 0.25	83.28	65.96	54.31	77.63 ± 0.19	91.83
	DEAR	89.12	38.98	68.07	67.33 ± 0.36	84.26	**57.79**	62.12	86.05 ± 0.17	91.94
	SOAR	90.47	37.17	69.69	69.33 ± 0.21	**85.96**	60.62	65.98	87.67 ± 0.17	92.49
	Ours	**92.50**	**35.88**	**79.30**	**73.47 ± 0.24**	84.24	78.29	**78.84**	**88.45 ± 0.21**	**95.55**
SlowFast	SoftMax	56.02	89.33	15.54	61.12 ± 0.26	55.39	91.58	20.57	75.02 ± 0.15	96.17
	OpenMax	68.49	39.38	10.48	69.74 ± 0.17	67.00	77.54	25.35	66.46 ± 0.16	60.33
	MC-Drop	95.01	18.21	88.99	71.12 ± 0.16	89.52	53.95	75.82	73.35 ± 0.15	96.24
	BNN-SVI	94.83	20.51	87.37	68.92 ± 0.19	88.68	60.88	74.05	71.14 ± 0.16	96.01
	DEAR	95.12	20.35	87.63	75.51 ± 0.17	89.33	58.78	75.95	89.71 ± 0.17	96.56
	SOAR	95.72	18.84	90.68	76.47 ± 0.14	**90.72**	**52.32**	**76.93**	90.64 ± 0.19	96.53
	Ours	**95.75**	**17.09**	**90.73**	**80.82 ± 0.21**	88.38	58.68	74.44	**90.77 ± 0.18**	**97.33**
I3D	SoftMax	44.47	96.93	8.85	55.50 ± 0.45	44.34	97.91	3.66	73.13 ± 0.12	94.10
	OpenMax	63.96	45.86	3.78	66.21 ± 0.16	63.67	80.53	6.54	67.81 ± 0.12	56.54
	MC-Drop	93.66	25.43	85.72	68.12 ± 0.20	86.11	77.50	70.13	71.13 ± 0.15	94.13
	BNN-SVI	93.16	25.88	79.36	67.96 ± 0.19	85.63	71.52	66.14	71.57 ± 0.17	93.89
	DEAR	93.52	29.53	84.03	75.12 ± 0.27	87.12	71.32	72.21	88.07 ± 0.20	93.97
	SOAR	**94.60**	25.33	**86.95**	76.22 ± 0.32	**88.10**	**69.57**	**72.75**	**89.55 ± 0.22**	**95.24**
	Ours	93.98	**24.50**	85.75	**76.81 ± 0.32**	86.31	69.65	72.11	89.17 ± 0.19	95.18

Table 2: Matched deep-ensemble comparison across four backbones: TPN [36], TSM [24], SlowFast [12], and I3D [6]. Both methods use the same ensemble size ($M = 10$) and the same open-set protocol. In both cases, epistemic uncertainty is computed using the ensemble. Models are trained on UCF101 [30] and evaluated with UCF101 as the known set and MiT-v2 [26] or HMDB51 [21] as unknown sources.

Backbone	Method	UCF101+MiT-v2				UCF101+HMDB51				Acc.
		AUROC↑	FPR@95↓	TPR@10↑	Open maF1↑	AUROC↑	FPR@95↓	TPR@10↑	Open maF1↑	
TPN	Vanilla DE	89.21	28.14	79.60	73.88 ± 0.15	84.39	62.64	68.75	85.10 ± 0.18	93.24
	Ours (DE)	**93.95**	**23.50**	**85.05**	**79.18 ± 0.20**	**88.62**	**54.16**	**74.31**	**90.57 ± 0.22**	**96.17**
TSM	Vanilla DE	89.72	39.12	75.42	70.47 ± 0.24	81.83	82.02	66.04	84.10 ± 0.25	93.44
	Ours (DE)	**92.52**	**35.69**	**80.59**	**74.23 ± 0.36**	**85.51**	**77.92**	**70.33**	**89.83 ± 0.23**	**95.74**
SlowFast	Vanilla DE	93.38	19.69	85.02	78.04 ± 0.21	86.47	60.13	69.17	88.09 ± 0.17	95.61
	Ours (DE)	**95.87**	**16.20**	**91.03**	**82.66 ± 0.20**	**88.49**	**58.89**	**75.22**	**91.48 ± 0.20**	**97.51**
I3D	Vanilla DE	91.96	28.02	82.44	74.17 ± 0.22	85.10	71.23	68.29	86.02 ± 0.18	93.76
	Ours (DE)	**94.72**	**24.64**	**86.53**	**77.79 ± 0.32**	**87.95**	**69.48**	**72.53**	**89.47 ± 0.19**	**95.28**

4.5 Ablation and Controlled Analyses

All ablation studies use the TPN backbone, with UCF101 as known data and HMDB51 as unknown data. We adopt this fixed setting to analyze the contribution of each component in a controlled, representative configuration.

Controlled comparison with PSL [7]. We rerun PSL (see Table 3), under our overlap-removed UCF101/HMDB51 protocol with the same TPN backbone. PSL is the closest purely visual, single-label OSAR baseline among the recent methods considered; STECN and USE use semantic-textual expansion or richer semantic information, while MULE targets multi-label OSAR. Since both PSL and our method use the same known classes, unknown split, and backbone, the comparison isolates the method difference without mixing protocol or architecture effects.

Table 3: Comparison with PSL on UCF101/HMDB51.

Method	AUROC↑	FPR@95↓	Acc.↑
PSL [7]	88.04	55.82	95.73
Ours (student)	**88.50**	**53.85**	**95.98**

Effect of the Stages. We study the effect of the three stages by selectively enabling or removing Stage 1, Stage 2, and Stage 3 in Table 4. We report the same open-set metrics used in the main results, namely AUROC, FPR@95, TPR@10, open maF1, and closed-set accuracy, together with the Hellinger distance (H) between the known and unknown uncertainty distributions. We also report ECE and NLL on the held-out known set to assess the impact of each stage on the quality of in-distribution confidence. For ablated variants that exclude Stage 1, the model is initialized from the corresponding vanilla deep-ensemble training setup before applying the remaining enabled stages. Table 4 shows that Stage 1 alone provides a modest improvement over the vanilla deep ensemble, whereas Stages 2 and 3 produce the larger gains. The variant with Stage 2 and Stage 3 but without Stage 1 remains strong at 87.76 AUROC, confirming that motion guidance is a useful initialization rather than the primary source of improvement. Adding Stage 3 to Stages 1+2 improves every reported metric. The full model,

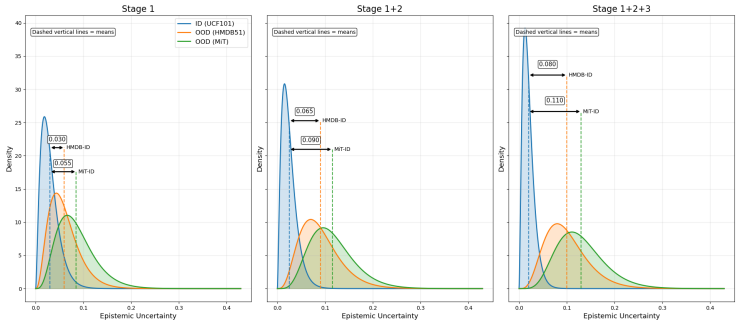


Fig. 3: Epistemic-uncertainty distributions for UCF101 ID and HMDB51 OOD clips on TPN after Stage 1, Stage 1+2, and Stage 1+2+3. Dashed lines mark the distribution means. The progressive increase in the mean gap is consistent with the Hellinger distance and open-set improvements in Table 4.

therefore, gives the best open-set performance, closed-set accuracy, Hellinger distance, ECE, and NLL, indicating that the stages are complementary.

Table 4: Deep-ensemble results on the TPN backbone under the UCF101/HMDB51 open-set setting. Stage 1, Stage 2, and Stage 3 denote motion-guided training, epistemic regularization, and UAM-IA, respectively. A tick (✓) indicates that the corresponding stage is enabled, while a cross (×) indicates that it is removed.

Stage 1	Stage 2	Stage 3	AUROC↑	FPR@95↓	TPR@10↑	maF1↑	Acc.↑	H↑	ECE↓	NLL↓
×	×	×	84.39	62.64	68.75	85.10 ± 0.18	93.24	0.67	0.025	0.19
✓	×	×	84.78	61.44	69.91	86.24 ± 0.21	94.33	0.69	0.023	0.17
×	✓	✓	87.76	55.28	73.47	89.33 ± 0.17	95.69	0.74	0.016	0.12
✓	✓	×	86.21	56.97	71.22	88.01 ± 0.19	95.02	0.77	0.018	0.13
✓	✓	✓	**88.62**	**54.16**	**74.31**	**90.57 ± 0.22**	**96.17**	**0.80**	**0.014**	**0.10**

Evolution of uncertainty separation. Figure 3 shows that refinement shifts the ID and OOD epistemic-uncertainty distributions farther apart across stages. The corresponding mean gap and Hellinger distance increase, supporting the use of epistemic uncertainty as both the training signal and the rejection score. The supplementary material provides the same stage-wise visualization for I3D, TSM, and SlowFast.

Stage 2 objective and component behavior. Because $U_e = U_t - U_a$, minimizing epistemic uncertainty could in principle reduce disagreement by shifting mass into U_a. We therefore compare regularizers based on U_a, U_t, and U_e, while retaining cross-entropy in every row. Table 5(a) shows that U_e gives the strongest Stage 2 result. On held-out ID clips, all mean uncertainty components decrease after Stage 2: $\bar{U}_t$ changes from 0.391 to 0.357, $\bar{U}_a$ from 0.355 to 0.331, and $\bar{U}_e$ from 0.036 to 0.026. Thus, epistemic regularization reduces total uncertainty and ensemble disagreement without increasing aleatoric uncertainty. For difficult ID clips, cross-entropy continues to supervise the class label, while the regularizer avoids treating intrinsic ambiguity as the novelty signal.

Table 5: Controlled analyses on TPN under UCF101/HMDB51. (a) Stage 2 regularizer target before UAM-IA; cross-entropy is retained in every row. (b) Final-model component removals.

(a) Stage 2 target

Variant	AUROC↑	FPR@95↓	Acc.↑
CE only	85.43	58.64	94.61
CE+U_a	85.38	58.87	94.54
CE+U_t	85.94	57.62	94.83
CE+U_e	**86.21**	**56.97**	**95.02**

(b) Component removals

Variant	AUROC↑	FPR@95↓	Acc.↑
$\lambda_{\mathrm{OF}} = 0$	87.76	55.28	95.69
$\lambda_{\mathrm{ER}} = 0$	88.03	54.89	95.84
Full model	**88.62**	**54.16**	**96.17**

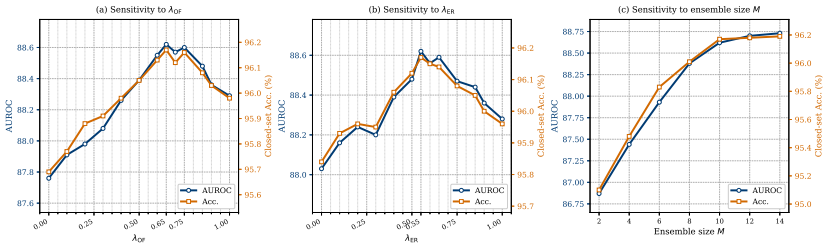


Fig. 4: Sensitivity to λ_{OF}, λ_{ER}, and ensemble size on TPN under UCF101/HMDB51. The selected values are 0.65, 0.55, and $M = 10$; AUROC saturates near $M = 10$ (88.62/88.70/88.73 for $M = 10/12/14$).

The zero-weight controls in Table 5(b) isolate the two added objectives. Removing motion guidance or epistemic regularization weakens the final result, but both variants remain competitive, consistent with their complementary roles.

Sensitivity to λ_{OF}, λ_{ER}, and ensemble size. Fig. 4 shows that the selected λ_{OF}, λ_{ER}, and $M = 10$ lie in a stable operating region, with performance saturating as the ensemble grows beyond ten members.

Stage 3 mask form and attention strength. We compare UAM-IA with ordinary fine-tuning, a cheap frame-difference mask, and $A_\beta = 1 - \beta M$, where $\beta = -0.5$ amplifies high-attribution regions and $\beta = 0.5$ suppresses them. Table 6(a) shows that UAM-IA is strongest, indicating that its gain is not explained by generic foreground or motion emphasis. The α (in Eq. 10) sweep in Table 6(b) has a broad optimum around $\alpha = 1$: weak modulation has limited effect, while excessive amplification begins to reduce known–unknown separation.

Table 6: Stage 3 controls on TPN under UCF101/HMDB51. All variants start from the same Stage 1+2 model.

(a) Attention form

Variant	AUROC↑	FPR@95↓	Acc.↑
No mask	86.21	56.97	95.02
Frame difference	87.18	56.04	95.47
High-UA amp.	87.05	56.31	95.38
High-UA sup.	87.42	55.71	95.61
UAM-IA	**88.62**	**54.16**	**96.17**

(b) Sensitivity to α

α	AUROC↑	FPR@95↓	TPR@10↑	Acc.↑
0.00	86.21	56.97	71.22	95.02
0.25	86.97	58.46	72.15	95.41
0.50	87.84	56.31	73.43	95.83
1.00	**88.62**	**54.16**	**74.31**	**96.17**
1.50	88.11	55.02	73.48	95.94
2.00	87.26	57.14	71.96	95.52

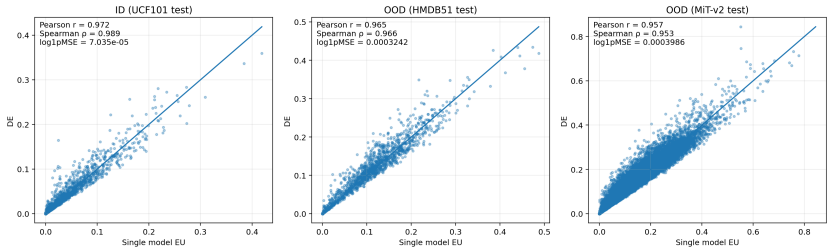


Fig. 5: Student-ensemble fidelity on UCF101 ID clips and HMDB51/MiT-v2 OOD clips. The distilled student closely follows the refined ensemble uncertainty, preserving both the ordering and scale of the ensemble-derived epistemic uncertainty.

Thresholds, fidelity, and cost. The Hellinger distance rises from 0.67 to 0.80 (Table 4). Thresholds are selected only from held-out UCF101 training data and frozen for all unknown sources. For TPN, TSM, SlowFast, and I3D, the thresholds are 0.047, 0.065, 0.060, and 0.055, respectively. Open maF1 uses these fixed thresholds, whereas AUROC, FPR@95, and TPR@10 use the full score curve. Fig. 5 shows that the distilled student closely matches the refined ensemble uncertainty on both ID and OOD clips, supporting single-model deployment. Training uses ensemble, motion/saliency, and attribution overhead, but deployment is single-pass: the 10-member RGB ensemble requires 322.525 ms/505.42 GFLOPs, whereas the RGB student requires 39.240 ms/52.542 GFLOPs. The supplementary material gives threshold-selection details, complementary backbone uncertainty distributions, qualitative examples, and failure cases.

5 Conclusion

We presented a staged uncertainty-guided fine-tuning framework for open-set action recognition, where ensemble-derived epistemic uncertainty is used not only for rejection but also for representation refinement. The framework combines motion-guided training, in-distribution epistemic regularization, and uncertainty attribution mask-based input attention. Across matched ensemble comparisons and controlled ablations, these stages improve known–unknown separation while preserving closed-set accuracy, with the largest gain coming from UAM-IA. We further distill the refined ensemble into a single RGB-only student model, which preserves much of the ensemble behavior with substantially lower inference cost. The main limitation is the offline training cost from ensemble refinement and attribution generation. Future work will explore stronger uncertainty distillation, efficient ensemble approximations, and larger-scale evaluations with transformer and foundation-model video backbones.

Acknowledgment

This work was supported in part by the DARPA project Environment-Driven Conceptual Learning (ECOLE) under Grant No. HR00112390059.

References

1. Antorán, J., Bhatt, U., Adel, T., Weller, A., Hernández-Lobato, J.M.: Getting a clue: A method for explaining uncertainty estimates (2021), https://arxiv.org/abs/2006.06848
2. Bao, W., Yu, Q., Kong, Y.: Evidential deep learning for open set action recognition. In: Proc. IEEE Int. Conf. Comput. Vis. pp. 13349–13358 (2021)
3. Bendale, A., Boult, T.E.: Towards open set deep networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1563–1572 (2016)
4. Bian, C., Yuan, C., Wang, J., Li, M., Yang, X., Yu, S., Ma, K., Yuan, J., Zheng, Y.: Uncertainty-aware domain alignment for anatomical structure segmentation. Medical Image Analysis **64**, 101732 (2020). https://doi.org/10.1016/j.media.2020.101732, https://www.sciencedirect.com/science/article/pii/S1361841520300967
5. Cam, L.L.: Asymptotic Methods in Statistical Decision Theory. Springer (1986)
6. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. pp. 6299–6308 (2017)
7. Cen, J., Zhang, S., Wang, X., Pei, Y., Qing, Z., Zhang, Y., Chen, Q.: Enlarging instance-specific and class-specific information for open-set action recognition. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 15295–15304 (2023). https://doi.org/10.1109/CVPR52729.2023.01468
8. Contributors, M.: Openmmlab's next generation video understanding toolbox and benchmark. https://github.com/open-mmlab/mmaction2 (2020)
9. Cover, T.M., Thomas, J.A.: Elements of Information Theory. Wiley, 2 edn. (2006)
10. Du, X., Wang, X., Gozum, G., Li, Y.: Unknown-aware object detection: Learning what you don't know from videos in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 13678–13688 (June 2022)
11. Fathullah, Y., Gales, M.J.F.: Self-distribution distillation: efficient uncertainty estimation. In: Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence. Proceedings of Machine Learning Research, vol. 180, pp. 663–673 (2022)
12. Feichtenhofer, C., Fan, H., Malik, J., He, K.: Slowfast networks for video recognition. In: Proc. IEEE Int. Conf. Comput. Vis. pp. 6202–6211 (2019)
13. Feng, B., Lu, Q., Zhang, R., Cao, Y., Yao, Y., Wang, Z.: Stecn: Towards open-set action recognition via semantic-textual expansion and class normalization. IEEE Transactions on Multimedia (2023), early Access / 2023
14. Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: Balcan, M.F., Weinberger, K.Q. (eds.) Proceedings of The 33rd International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 48, pp. 1050–1059. PMLR, New York, New York, USA (20–22 Jun 2016), https://proceedings.mlr.press/v48/gal16.html
15. Hinton, G.E., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. ArXiv **abs/1503.02531** (2015), https://api.semanticscholar.org/CorpusID:7200347
16. Hu, X., Zhang, R., Cao, Y., Lu, Q., Yao, Y., Wang, Z.: Exploring rich semantics for open-set action recognition. IEEE Transactions on Multimedia (2024)

17. Huang, X., Li, H., Qiao, Y., Wang, P., Gao, W., Li, H.: Froster: Frozen CLIP is a strong teacher for open-vocabulary action recognition. In: International Conference on Learning Representations (ICLR) (2024)
18. Kirillov, A., et al.: Segment anything. In: Proc. IEEE Int. Conf. Comput. Vis. pp. 4015–4026 (2023)
19. Korattikara, A., Rathod, V., Murphy, K., Welling, M.: Bayesian dark knowledge. In: Advances in Neural Information Processing Systems. vol. 28 (2015)
20. Krishnan, R., Subedar, M., Tickoo, O.: Bar: Bayesian activity recognition using variational inference. ArXiv **abs/1811.03305** (2018), https://api.semanticscholar.org/CorpusID:53210875
21. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: Hmdb: a large video database for human motion recognition. In: IEEE Int. Conf. Comput. Vis. pp. 2556–2563 (2011)
22. Lakshminarayanan, B., Pritzel, A., Blundell, C.: Simple and scalable predictive uncertainty estimation using deep ensembles. In: Proceedings of the 31st International Conference on Neural Information Processing Systems. p. 6405–6416. NIPS'17, Curran Associates Inc., Red Hook, NY, USA (2017)
23. Landgraf, S., Hillemann, M., Wursthorn, K., Ulrich, M.: Uncertainty-aware cross-entropy for semantic segmentation. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences **X-2-2024**, 129–136 (06 2024). https://doi.org/10.5194/isprs-annals-X-2-2024-129-2024
24. Lin, J., Gan, C., Han, S.: Tsm: Temporal shift module for efficient video understanding. In: Proc. IEEE Int. Conf. Comput. Vis. pp. 7083–7093 (2019)
25. Malinin, A., Mlodozeniec, B., Gales, M.: Ensemble distribution distillation. In: International Conference on Learning Representations (2020)
26. Monfort, M., et al.: Multi-moments in time: Learning and interpreting models for multi-action video understanding. IEEE Trans. Pattern Anal. Mach. Intell. **44**(12), 9434–9445 (Nov 2021)
27. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 32. Curran Associates, Inc. (2019), https://proceedings.neurips.cc/paper_files/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf
28. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 618–626 (2017). https://doi.org/10.1109/ICCV.2017.74
29. Sensoy, M., Kaplan, L., Kandemir, M.: Evidential deep learning to quantify classification uncertainty. In: Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 31. Curran Associates, Inc. (2018), https://proceedings.neurips.cc/paper_files/paper/2018/file/a981f2b708044d6fb4a71a1463242520-Paper.pdf
30. Soomro, K., Zamir, A., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. ArXiv **abs/1212.0402** (2012), https://api.semanticscholar.org/CorpusID:7197134
31. Teed, Z., Deng, J.: Raft: Recurrent all-pairs field transforms for optical flow. In: Eur. Conf. Comput. Vis. pp. 402–419 (2020)

32. Wang, H., Joshi, D., Wang, S., Ji, Q.: Gradient-based Uncertainty Attribution for Explainable Bayesian Deep Learning . In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 12044–12053. IEEE Computer Society, Los Alamitos, CA, USA (Jun 2023). https://doi.org/10.1109/CVPR52729.2023.01159, https://doi.ieeecomputersociety.org/10.1109/CVPR52729.2023.01159
33. Wang, M., Xing, J., Liu, Y.: Actionclip: A new paradigm for video action recognition. arXiv preprint arXiv:2109.08472 (2021)
34. Weng, Z., Yang, X., Li, A., Wu, Z., Jiang, Y.G.: Open-vclip: Transforming CLIP to an open-vocabulary video model via interpolated weight optimization. In: Proceedings of the 40th International Conference on Machine Learning (ICML). pp. 35178–35199 (2023)
35. Xie, J., Yang, C., Xie, W., Zisserman, A.: Moving object segmentation: All you need is sam (and flow). In: ACCV (2024)
36. Yang, C., Xu, Y., Shi, J., Dai, B., Zhou, B.: Temporal pyramid network for action recognition. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. pp. 591–600 (2020)
37. Yu, Y., Cao, C., Zhang, Y., Zhang, Y.: Learning to generalize without bias for open-vocabulary action recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2025)
38. Zhai, Y., Liu, Z., Wu, Z., Wu, Y., Zhou, C., Doermann, D., Yuan, J., Hua, G.: Soar: Scene-debiasing open-set action recognition. In: Proc. IEEE Int. Conf. Comput. Vis. pp. 10244–10254 (2023)
39. Zhao, C., Du, D., Hoogs, A., Funk, C.: Open set action recognition via multi-label evidential learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 22981–22990 (2023)