

# Multimodal Knowledge Distillation for Egocentric Action Recognition Robust to Missing Modalities

Dustin Carrión-Ojeda<sup>\*1,2</sup> Maria Santos-Villafranca<sup>\*3</sup> Alejandro Perez-Yus<sup>3</sup>  
 Jesus Bermudez-Cameo<sup>3</sup> Jose J. Guerrero<sup>3</sup> Simone Schaub-Meyer<sup>1,2</sup>

<sup>1</sup>Technical University of Darmstadt, Department of Computer Science <sup>2</sup>hessian.AI <sup>3</sup>I3A – University of Zaragoza  
 \*Equal contribution

**Abstract**—Egocentric action recognition enables robots to facilitate human-robot interactions and monitor task progress. Existing methods often rely solely on RGB videos, although additional modalities, such as audio, can improve accuracy under challenging conditions. However, most multimodal approaches assume that all modalities are available at inference time, leading to significant accuracy drops, or even failure, when inputs are missing. To address this limitation, we introduce KARMMA, a multimodal Knowledge distillation framework for egocentric Action Recognition robust to Missing Modalities that does not require modality alignment across all samples during training or inference. KARMMA distills knowledge from a multimodal teacher into a multimodal student that leverages all available modalities while remaining robust to missing ones, enabling deployment across diverse sensor configurations without retraining. Our student uses approximately 50 % fewer computational resources than the teacher, resulting in a lightweight and fast model that is well suited for on-robot deployment. Experiments on Epic-Kitchens and Something-Something demonstrate that our student achieves competitive accuracy while significantly reducing performance degradation under missing modality conditions. Project page available at: <https://visinf.github.io/KARMMA/>

## I. INTRODUCTION

Egocentric vision aims to capture and interpret the world from a first-person perspective. Recently, interest in this field has grown due to advances in human-like robots and the increasing popularity of portable cameras. Egocentric vision supports a wide range of applications, including assistive devices [1], human-robot interaction (HRI) [2], and surveillance [3]. The release of large-scale egocentric datasets [4, 5] has further enabled progress in tasks such as gesture/action recognition [6, 7] and object recognition [8].

Compared to the exocentric (third-person) setting, egocentric video is more challenging due to camera motion, which introduces blur and frequent occlusions. To mitigate these issues, recent works have explored multimodal cues [9, 10]. However, most multimodal methods assume that all modalities are available at inference time. This assumption often fails in practice, particularly in robotics, due to privacy constraints, muted microphones, or sensor malfunctions [11] (e.g., a faulty camera). Moreover, multimodal architectures typically suffer significant accuracy drops when the most informative modality is missing [12].

To meet the safety requirements of robotics, especially HRI, perception must remain reliable under missing modality conditions. We address this challenge with a multimodal

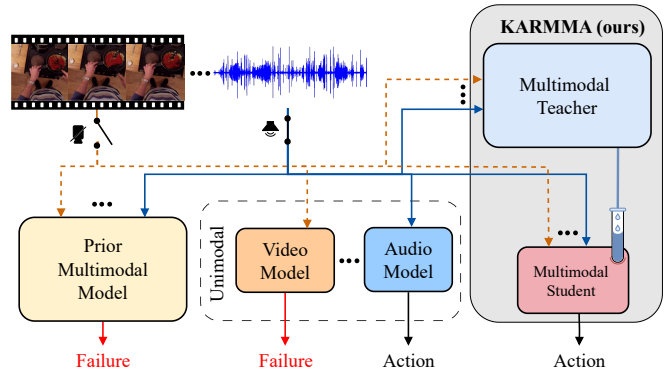


Fig. 1: **KARMMA motivation.** We introduce a novel *multimodal-to-multimodal* framework that leverages all available modalities while remaining robust when any are absent, eliminating the need for modality-aligned data. KARMMA produces a lightweight student that operates on any subset of the trained modalities, providing high flexibility and computational efficiency, making it suitable for edge and on-device deployment. Solid lines indicate available modalities, while dashed lines denote missing ones.

framework that leverages all available modalities while improving robustness to missing ones. However, conventional multimodal models are computationally demanding, and processing all modalities can be slow, limiting reliable operation in dynamic or uncertain environments. Instead, we propose a multimodal-to-multimodal distillation pipeline in which a large teacher, constructed by fusing frozen, pre-trained unimodal encoders, transfers its knowledge to a lightweight student. Notably, both models can operate on any subset of modalities during training and inference, making them suitable for real-world applications where sensor availability is unpredictable (see Fig. 1).

**Contributions.** (1) We propose a novel multimodal-to-multimodal distillation framework for egocentric action recognition that does not require modality alignment across samples during training or inference. (2) Our distillation process explicitly accounts for missing modalities, producing a lightweight, fast, and flexible student suitable for on-robot deployment. (3) Our teacher fuses features from frozen, pre-trained unimodal encoders, eliminating the need to retrain them, which simplifies the integration of newer encoders as they become available. (4) We introduce a fusion block with a simple, parameter-free token reduction strategy that lowers computational cost without sacrificing accuracy.

## II. RELATED WORK

**Egocentric action recognition** has primarily been studied using RGB video [13], which is the most informative modality for this task. However, several works show that incorporating additional modalities, such as gaze [14] or audio [15], improves accuracy. Early multimodal research focused on bimodal setups, but the recent release of multimodal egocentric datasets [4, 5, 16, 17] has driven the development of multimodal approaches beyond two modalities. For instance, Gong et al. [10] investigate generalization to unseen and missing modalities, while Dong et al. [18] study domain generalization under missing modality conditions. Additionally, several works [6, 19] distill a multimodal teacher into a unimodal student to improve its accuracy. In contrast, we distill knowledge from a large multimodal teacher into a compact multimodal student that is explicitly designed to handle missing modalities.

**Modality fusion** is commonly implemented using cross-attention [20], although such pairwise fusion limits scalability beyond two modalities. Contrastive objectives provide an alternative strategy for fusing modality pairs. For example, Lin et al. [21] introduce a contrastive loss tailored to egocentric video-language alignment. Other approaches pre-train modality-specific experts and fuse their frozen features using model-agnostic mechanisms [6], enabling fusion across multiple modalities. However, all these methods assume that all modalities are available at inference time. To address this limitation, Gong et al. [10] apply modality dropout during training to improve robustness to missing inputs, while Ramazanova et al. [22] introduce a learnable modality token activated only when a modality is absent. Ramazanova et al. [23] further propose a test-time adaptation strategy for missing modalities which requires a separate forward pass for each modality combination and is restricted to dropping a fixed modality in a bimodal setting. Our transformer-based fusion extends these ideas by combining modality dropout with two types of learnable tokens. The resulting student model is fast and handles any subset of available modalities without requiring additional forward passes.

**Token reduction** is essential in self-attention-based fusion methods, as computational cost scales quadratically with the number of input tokens. To mitigate this cost, Fayyaz et al. [24] propose a parameter-free module that prunes the attention matrix, while Shang et al. [25] prune tokens via clustering using an interquartile range scoring function [26]. Other approaches merge redundant tokens based on cosine similarity [27] or treat tokens as samples of a continuous signal and subsample them accordingly [28]. In contrast, we show that a simple parameter-free strategy, averaging contiguous tokens within each modality, effectively reduces computation without sacrificing accuracy.

**Knowledge distillation** transfers knowledge from a large teacher model to a smaller student. In multimodal settings, Radevski et al. [6] and Hatano et al. [19] improve unimodal students using multimodal teachers, while Wei et al. [29] propose a multimodal-to-multimodal distillation framework

with modality dropout for classification and segmentation. Building on these works, we introduce a novel multimodal-to-multimodal distillation framework for egocentric action recognition that combines modality dropout with an additional mechanism to enhance robustness to missing inputs.

## III. KARMMA

This work addresses multimodal egocentric action recognition, which aims to classify human actions from multiple egocentric modalities. Fig. 2 summarizes our proposed KARMMA framework, which distills knowledge from a large teacher into a compact student.

### A. Problem Definition

Given an egocentric dataset with  $M$  modalities, let  $\mathbf{x}_i^j \in \mathbb{R}^{T \times D_1^j \times \dots \times D_n^j}$  denote the  $i^{\text{th}}$  action sequence of the  $j^{\text{th}}$  modality, where  $T$  is the number of time steps and  $D_1^j \times \dots \times D_n^j$  are the  $n$  modality-specific spatial or spectral dimensions. For a multimodal action sequence  $\mathbf{X}_i = \{\mathbf{x}_i^1, \mathbf{x}_i^2, \dots, \mathbf{x}_i^M\}$ , the goal is to learn a model  $f$  that predicts the corresponding action  $\mathbf{y}_i$ , formulated as  $\hat{\mathbf{y}}_i = \arg\max \sigma(f(\mathbf{X}_i))$ , where  $\sigma(\cdot)$  denotes the softmax operator. Since  $\mathbf{X}_i$  may contain any subset of the  $M$  modalities,  $f$  must be robust to missing inputs. The representation of actions  $\mathbf{y}_i$  also varies across datasets. For example, in Something-Something [16], an action is represented by a single label, whereas in Epic-Kitchens [5], it consists of two outputs (noun and verb).

### B. Teacher and Student

Both teacher and student consist of three components: modality-specific feature extractors (FEs), a transformer-based fusion block (FB), and a multi-head MLP (MH-MLP). A key advantage is that both models can perform inference on *any* subset of the training modalities, so a single model handles all modality combinations. Additionally, the student uses smaller FEs and a more compact FB, reducing memory usage and inference time.

**Feature Extractors (FEs).** Since  $\mathbf{X}_i$  may include up to  $M$  modalities, we use  $M$  *frozen*, pre-trained *unimodal* FEs for the teacher and train only the remaining components. This design facilitates replacing FEs as newer encoders become available. For each modality, we select an established FE available in multiple sizes so the teacher uses a larger variant, while the student uses a smaller one. When available, student FEs are initialized from pre-trained weights and fine-tuned.

**Fusion Block (FB).** Inspired by previous works [10, 12, 30], we design a transformer-based FB capable of handling an arbitrary number of input tokens and modalities. As shown in Fig. 2 (left), modality-specific tokens are first projected to a shared dimension via a linear embedding layer. The projected tokens, together with a learnable [CLS] token, pass through  $l$  transformer layers. The FB outputs  $M + 1$  tokens: one averaged token per modality and the [CLS] token that aggregates cross-modal information, providing a modality-agnostic representation that remains informative when some modalities are missing.

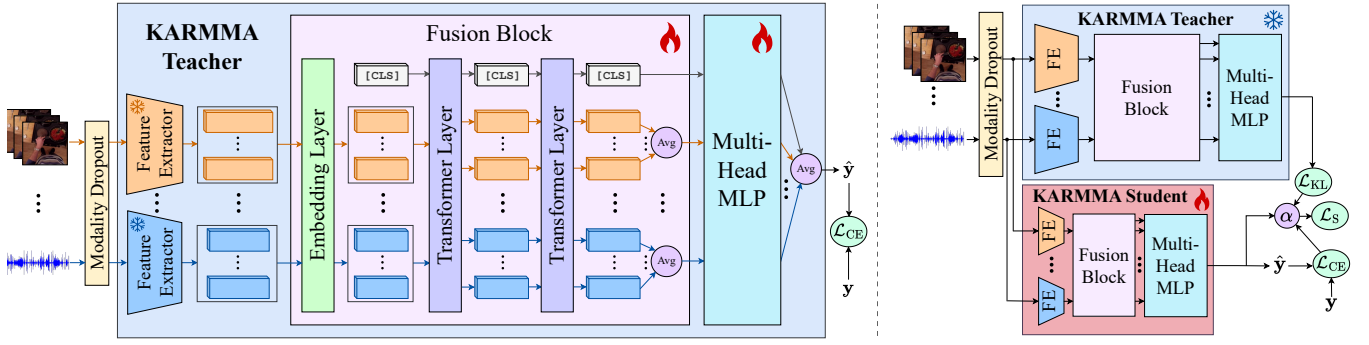


Fig. 2: **KARMMA training.** *Left* (first stage): training the teacher. *Right* (second stage): distilling knowledge from the frozen teacher into the student. Both networks use modality dropout and our student includes our strategy for handling missing modalities (see Sec. III-D) to remain robust when inputs are incomplete.

**Multi-Head MLP (MH-MLP).** The  $M + 1$  tokens produced by the FB are fed into a MH-MLP consisting of a shared fully connected layer followed by independent classification heads. The number of heads matches the number of outputs required to describe an action. As discussed in Sec. III-A, actions may require a single output (*e.g.*, Something-Something [16]) or multiple outputs (*e.g.*, Epic-Kitchens [5]). For each head, we average the  $M + 1$  predictions to obtain the final output.

### C. Token Reduction Strategy

Since the computational cost of our FB grows with the number of modalities, we introduce a simple *parameter-free* token reduction strategy ( $\Theta$ -Average) that caps the number of tokens per modality with a threshold  $\Theta$ . If a modality-specific FE outputs  $k$  tokens with  $k > \Theta$ , we partition them into  $\Theta$  groups of  $\lfloor \frac{k}{\Theta} \rfloor$  tokens, assigning the remaining  $k \bmod \Theta$  tokens to the last group. Averaging tokens within each group yields exactly  $\Theta$  tokens. If  $k \leq \Theta$ , all tokens remain unchanged. Applying our  $\Theta$ -Average to the outputs of both teacher and student FEs limits the number of tokens entering the FB, reducing computational and memory costs without introducing learnable parameters.

### D. KARMMA Enhancements

Our proposed KARMMA framework incorporates three main enhancements: (1) modality dropout applied to both teacher and student, (2) a novel student-side strategy for handling missing modalities, and (3) a multimodal-to-multimodal distillation scheme that improves the accuracy of a flexible, lightweight, and fast student, eliminating the need for using a large, slow model at inference.

**Modality Dropout.** Previous works [10, 22, 29] show that modality dropout improves robustness to missing modalities. Similar to neuron dropout, modality dropout removes entire modalities with probability  $p$  while ensuring that at least one modality remains active. As illustrated in Fig. 2 (right), we apply modality dropout to *both* teacher and student so that neither relies on the full modality set during training. This allows KARMMA to train on datasets where action sequences  $\mathbf{X}_i$  may contain different modality subsets (*i.e.*, no modality alignment is required), making our framework

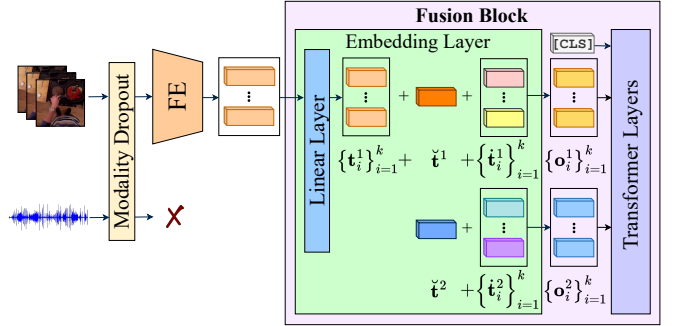


Fig. 3: **Missing modality strategy.** To handle missing modalities, the embedding layer projects the tokens from the feature extractor when available. Then, it adds a learned modality token  $\check{\mathbf{t}}^m$  to all projected tokens. Finally, a learned token  $\mathbf{t}_i^m$  is added to each token.

highly flexible and suitable for practical scenarios, including robotics, where sensor dropouts are common.

**Missing Modality Strategy.** To further enhance the robustness of the student to missing modalities, we propose a simple yet effective strategy illustrated in Fig. 3. Our strategy introduces two types of learnable tokens:

- 1) *Modality-specific token* ( $\check{\mathbf{t}}^m$ ): one token per modality that helps distinguish modalities and acts similar to positional encodings.
- 2) *Token-specific tokens* ( $\mathbf{t}_i^m$ ): a set of  $k$  tokens per modality that compensate when a modality is absent.

Formally, let the FE for modality  $m$  produce  $k$  tokens from the input  $\mathbf{x}_i^m$ . Then the output of the embedding layer  $\mathbf{o}_i^m$  is expressed as

$$\mathbf{o}_i^m = \begin{cases} \{\mathbf{t}_i^m + \check{\mathbf{t}}^m + \mathbf{t}_i^m\}_{i=1}^k & \text{if } m \text{ is present,} \\ \{\check{\mathbf{t}}^m + \mathbf{t}_i^m\}_{i=1}^k & \text{otherwise,} \end{cases} \quad (1)$$

where  $\mathbf{t}_i^m$  is the  $i^{\text{th}}$  projected token for modality  $m$ . This strategy provides rich information by combining both types of learnable tokens while keeping the input size of the FB unchanged, allowing the network to effectively handle any pattern of missing modalities.

**Knowledge Distillation.** As shown in Fig. 2, KARMMA operates in two stages. First, we train a large multimodal

teacher with frozen FEs using the cross-entropy loss:

$$\mathcal{L}_{\text{CE}} = -\frac{1}{N} \sum_{i=1}^N \mathbf{y}_i \cdot \log \hat{\mathbf{y}}_i^{\text{T}}, \quad (2)$$

where  $N$  is the number of training action sequences,  $\mathbf{y}_i$  is the ground-truth action, and  $\hat{\mathbf{y}}_i^{\text{T}}$  is the predicted action.

In the second stage, the trained teacher is frozen and its knowledge is distilled to a lightweight student by aligning the class probability distributions from both models via the Kullback–Leibler (KL) divergence:

$$\mathcal{L}_{\text{KL}} = \frac{1}{N} \sum_{i=1}^N \bar{\mathbf{y}}_i^{\text{T}} \cdot \left( \log \bar{\mathbf{y}}_i^{\text{T}} - \log \bar{\mathbf{y}}_i^{\text{S}} \right), \quad (3)$$

where  $\bar{\mathbf{y}}_i^{\text{T}} = \sigma(f^{\text{T}}(\mathbf{X}_i))$  and  $\bar{\mathbf{y}}_i^{\text{S}} = \sigma(f^{\text{S}}(\mathbf{X}_i))$  are the class probability distributions from the teacher and student, respectively. To prevent the student from just copying the teacher, we optimize a combination of the cross-entropy and distillation losses:

$$\mathcal{L}_{\text{S}} = \alpha \mathcal{L}_{\text{CE}} + (1 - \alpha) \mathcal{L}_{\text{KL}}, \quad (4)$$

where  $\alpha$  balances the loss terms. The full student, including its FEs, is trained with this objective, resulting in a model with improved accuracy while requiring substantially less memory and offering faster inference than the teacher.

#### IV. EXPERIMENTS

**Datasets.** We evaluate our KARMMA framework on two standard egocentric action recognition datasets: Epic-Kitchens-100 [5] and Something-Something V2 [16]. Epic-Kitchens contains 300 nouns and 97 verbs, with each action defined as a verb-noun pair (e.g., “pick up + knife”). For this dataset, we consider three modalities: RGB video (V), optical flow (F), and audio (A). In contrast, Something-Something contains 174 object-agnostic actions (e.g., “moving [something] up”) and provides three modalities: V, F, and object detection annotations (D).

As in most previous works, we report results on the validation splits of both datasets because the official test sets are not publicly available. However, we observed that the Epic-Kitchens validation split contains four nouns absent from the training set; we therefore remove the corresponding samples and evaluate on this *pruned* validation split. Moreover, to avoid overestimating the generalization capabilities of our method by validating and testing on the same data, we further define *Epic-Kitchens\**: a 90% / 10% train/validation split of the original training set, reserving the pruned validation split exclusively for testing.<sup>1</sup>

**Implementation Details.** For the *teacher*, we use Swin-B [31] pre-trained on Kinetics-400 [32] as the feature extractor for V, A, and F, and a 12-layer STLT [33] pre-trained on Action Genome [34] for D. The fusion block uses an embedding dimension of 768, 2 transformer layers ( $l=2$ ), 8 attention heads, 30% attention dropout, and 50% modality dropout. We apply  $\Theta$ -Average token reduction with  $\Theta=300$

TABLE I: **Analysis of KARMMA under different modality combinations.** “KARMMA<sub>T</sub>” and “KARMMA<sub>S</sub>” denote our teacher and student, respectively (see Sec. III-B). “Baseline” uses the same architecture as “KARMMA<sub>S</sub>” but is trained end-to-end with cross-entropy loss and without the KARMMA enhancements (see Sec. III-D), whereas “Baseline w/  $\delta$ ” incorporates modality dropout and our missing modality strategy. “V,” “F,” “A,” and “D” denote RGB video, optical flow, audio, and object detection annotations, respectively, and “[A/D]” indicates that either audio or object detection is used. We report top-1 action recognition accuracy (%) on all datasets. The “Epic-Kitchens\*” column reports results on our custom split (see Sec. IV). Blue highlights our final teacher and student. **Bold** and underlined values indicate the best and second-best results.

Method	Inference Modalities	Epic-Kitchens	Epic-Kitchens*	Something-Something
Baseline		40.00	39.26	57.31
Baseline w/ $\delta$	V+F+[A/D]	41.49	40.05	60.31
KARMMA <sub>S</sub>		<b>43.00</b>	<b>41.98</b>	<b>62.88</b>
KARMMA <sub>T</sub>		36.44	35.84	53.51
Baseline		36.80	36.07	56.98
Baseline w/ $\delta$	V+F	39.69	39.00	<b>60.38</b>
KARMMA <sub>S</sub>		<b>41.29</b>	<b>40.42</b>	<u>58.34</u>
KARMMA <sub>T</sub>		32.96	31.99	43.14
Baseline		37.48	36.60	40.03
Baseline w/ $\delta$	V+[A/D]	39.36	37.76	53.04
KARMMA <sub>S</sub>		<b>40.84</b>	<b>40.10</b>	<b>59.35</b>
KARMMA <sub>T</sub>		33.79	33.44	49.89
Baseline		5.49	6.84	37.35
Baseline w/ $\delta$	F+[A/D]	27.09	27.21	50.78
KARMMA <sub>S</sub>		<b>30.36</b>	<b>28.24</b>	<b>53.96</b>
KARMMA <sub>T</sub>		20.44	19.95	40.62
Baseline		32.10	32.50	39.91
Baseline w/ $\delta$	V	37.84	36.48	<b>53.02</b>
KARMMA <sub>S</sub>		<b>38.85</b>	<b>38.37</b>	51.97
KARMMA <sub>T</sub>		29.44	29.34	33.80
Baseline		2.58	4.73	36.50
Baseline w/ $\delta$	F	25.64	26.01	<b>50.97</b>
KARMMA <sub>S</sub>		<b>27.28</b>	<b>26.06</b>	47.24
KARMMA <sub>T</sub>		15.51	15.70	23.82
Baseline		2.05	2.25	0.07
Baseline w/ $\delta$	[A/D]	6.23	6.44	1.21
KARMMA <sub>S</sub>		8.17	7.53	<b>37.95</b>
KARMMA <sub>T</sub>		<b>9.38</b>	<b>9.11</b>	<u>29.40</u>

(see Sec. III-C). The teacher is trained using AdamW [35] for 100 epochs using a batch size of 32, weight decay of 0.05, and gradient clipping at 1.0. The learning rate starts at  $1e^{-5}$ , warms up linearly to  $5e^{-4}$  over 10 epochs, and then follows cosine decay back to the initial value.

Unless stated otherwise, the *student* inherits the hyperparameters of the teacher. It uses Swin-T [31] pre-trained on Kinetics-400 [32] for V and F, AST-T [36] for A, and a 9-layer STLT [33] for D. Since the latter two lack publicly available pre-trained weights, they are trained from scratch. The fusion block is reduced to an embedding dimension of 384, with  $l=1$ , and no attention dropout. The student is trained with a batch size of 6, a weight decay of 0.01, gradient clipping at 2.0, and a peak learning rate of  $1e^{-4}$ . In Eq. (4), we set  $\alpha=0.7$ .

##### A. Analysis of KARMMA

To evaluate the effectiveness of KARMMA, Tab. I reports the top-1 action recognition accuracy for the teacher (KARMMA<sub>T</sub>) and student (KARMMA<sub>S</sub>) introduced

<sup>1</sup>The Epic-Kitchens\* split is available in the released codebase.

in Sec. III-B. KARMMA<sub>S</sub> includes the enhancements described in Sec. III-D; thus, to isolate their contributions, we compare against two baselines: (1) “Baseline” uses the same architecture as KARMMA<sub>S</sub> but is trained end-to-end with cross-entropy and without KARMMA enhancements; (2) “Baseline w/  $\delta$ ” adds modality dropout and our missing modality strategy but excludes knowledge distillation.

The results in Tab. I show that both baselines and the student outperform the teacher across most modality combinations. While knowledge distillation typically produces a stronger teacher than student, in our setting the student benefits from fully trainable feature extractors, whereas the teacher’s remain frozen. Thus, although KARMMA<sub>S</sub> uses smaller feature extractors, they are fine-tuned for multimodal egocentric action recognition. In contrast, KARMMA<sub>T</sub> focuses solely on learning effective feature fusion under missing modality conditions. Consequently, transferring this knowledge to KARMMA<sub>S</sub> yields absolute gains (relative gains in parentheses) of 3.00% (7.5%), 2.72% (6.9%), and 5.57% (9.7%) over the Baseline when all modalities are present on Epic-Kitchens, Epic-Kitchens\*, and Something-Something, respectively. Additionally, the results on Epic-Kitchens\* closely match those on the original split, indicating robust generalization across data splits.

Tab. I also shows that the highest accuracy is achieved when all modalities are available, confirming that our model effectively leverages multimodal information. Moreover, the results confirm that RGB video is the most informative modality, while audio and object detection annotations contribute the least. Consequently, replacing V+[A/D] with F+[A/D] leads to a significant accuracy drop across all models, and unimodal [A/D] settings degrade even further. However, these drops are more severe for the Baseline model, showing that modality dropout and our missing modality strategy help mitigate over-reliance on dominant modalities.

Since KARMMA<sub>S</sub> outperforms both baselines across most modality combinations and datasets, it demonstrates that while a model can be trained to handle missing modalities (Baseline w/  $\delta$ ), incorporating our knowledge distillation further improves both accuracy and robustness to missing modalities. This improvement is particularly strong on Something-Something, where KARMMA<sub>S</sub> achieves an absolute accuracy gain of 36.74% ( $\approx 3000\%$  relative) over Baseline w/  $\delta$  when only object detection annotations (D) are used. On Epic-Kitchens, the gain is smaller at 1.94% (31% relative), but it still represents a substantial improvement over Baseline w/  $\delta$  when only audio (A) is used. The large difference in gains between the two datasets may be due to audio being less informative than object detection annotations, which provide spatial and dimensional information about the objects involved in the action.

Although Tab. I demonstrates that our student can effectively handle missing modalities, it only considers scenarios where entire modalities are absent at inference time. In contrast, robotics deployments often face run-time *sensor dropouts* (e.g., occluded cameras or muted microphones). To simulate such conditions, Fig. 4 reports the action recogni-

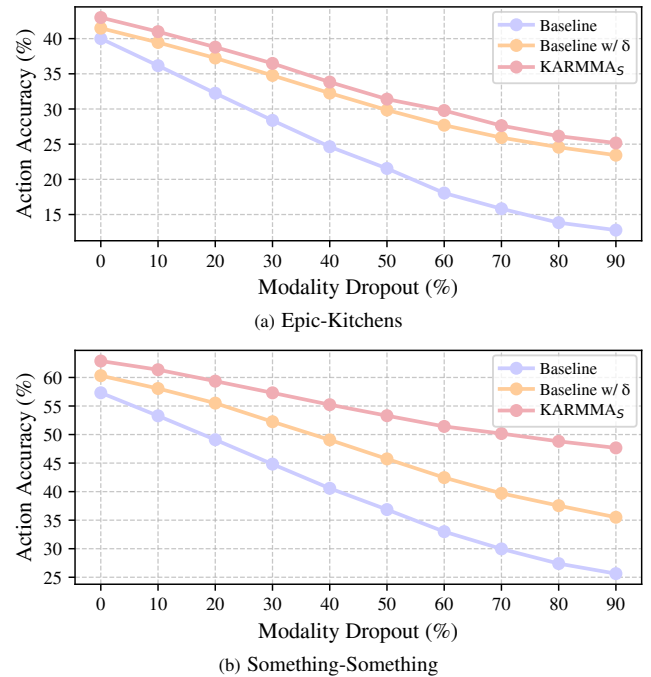


Fig. 4: **Impact of run-time sensor dropouts.** To emulate robotics deployments, we vary the modality dropout probability at inference from 0% to 90% and report action recognition accuracy. “Baseline” uses the same architecture as our student “KARMMA<sub>S</sub>” (see Sec. III-B) but is trained end-to-end with cross-entropy loss and without the KARMMA enhancements (see Sec. III-D), whereas “Baseline w/  $\delta$ ” incorporates modality dropout and our missing modality strategy.

tion accuracy of the two baselines and our student as the probability of dropping each modality increases from 0% to 90% during inference. These results show that the Baseline model is highly sensitive: missing one or more modalities in only 10% of the samples already reduces accuracy by roughly 5% on both datasets, and increasing this probability to 90% leads to absolute drops of about 27% and 32% for Epic-Kitchens and Something-Something, respectively. Incorporating modality dropout and our missing modality strategy enhances robustness, as Baseline w/  $\delta$  exhibits a significantly smaller accuracy drop than the Baseline. Moreover, our distillation pipeline consistently improves the accuracy of KARMMA<sub>S</sub>, yielding absolute gains (relative gains in parentheses) of approximately 2% (9%) and 12% (34%) over Baseline w/  $\delta$  in the 90% missing modality setting for Epic-Kitchens and Something-Something, respectively.

### B. Comparison With the State of the Art

Since KARMMA is inspired by Radevski et al. [6], which is the current state-of-the-art (SOTA) method in multimodal-to-unimodal distillation for egocentric action recognition, Tab. II compares both methods on the Epic-Kitchens dataset. For a fair comparison, we report two versions of our model: a unimodal (RGB video only) student (KARMMA<sub>V</sub>), and our full multimodal student (KARMMA<sub>S</sub>).

The top part of Tab. II shows that KARMMA<sub>V</sub> achieves slightly lower accuracy than the student of Radevski et al. [6]. However, KARMMA<sub>V</sub> uses a teacher with approximately 7.5

TABLE II: Comparison with the SOTA multimodal-to-unimodal distillation approach for egocentric action recognition. All methods distill knowledge from a multimodal teacher trained on RGB video (V), optical flow (F), and audio (A). “KARMMA<sub>V</sub>” denotes a unimodal version of our multimodal student (“KARMMA<sub>S</sub>”). Results for Radevski et al. [6] are computed using their publicly available student checkpoint. Blue highlights our final student. **Bold** and underline values indicate the best and second-best results.

Method	Student Train. Mods.	Student Infer. Mods.	Flexible Mod. Inference	Teacher Train. Params. (M)	Student Train. Params. (M)	Action Acc. (%)	GFLOPs
Radevski et al. [6]	V	V	✗	84.65	<b>28.21</b>	41.81	<b>175</b>
KARMMA <sub>V</sub>			✗	<u>11.25</u>	<u>29.64</u>	40.84	<u>176</u>
KARMMA <sub>S</sub>	V+F+A	V	✓	<b>11.25</b>	65.15	38.85	176
KARMMA <sub>S</sub>		V+F+A	✓	<b>11.25</b>	65.15	<b>43.00</b>	<u>358</u>

times fewer trainable parameters, as our teacher does not fine-tune its feature extractors. Moreover, both models are limited to performing inference with RGB video only. In robotics, this is problematic as camera occlusions or privacy settings can degrade the RGB stream while other sensors remain available, and training a separate student for each modality combination is impractical on embedded platforms.

The bottom part of Tab. II shows how our KARMMA<sub>S</sub> overcomes this limitation by supporting inference with various modality combinations, unlike the unimodal students. Since KARMMA<sub>S</sub> is designed to handle and benefit from multiple modalities, its accuracy on RGB video only is slightly lower than that of the unimodal models, which are optimized specifically for this modality. However, when all modalities are available, KARMMA<sub>S</sub> outperforms the student of Radevski et al. [6] by 1.19% (2.9% relative). Additionally, when performing inference with a single modality, all models have a similar computational cost, making our multimodal student equally efficient while offering the flexibility to exploit additional sensors when available. Therefore, KARMMA<sub>S</sub> is applicable across diverse multimodal scenarios, including on-robot deployment, leveraging all available modalities while remaining robust to missing ones.

### C. Ablation Studies

All ablations are conducted on the Epic-Kitchens-100 [5] dataset with models trained for 50 epochs (cf., 100 in the main experiments).

**KARMMA Enhancements.** The results in Tab. III show that applying modality dropout improves the noun, verb and action accuracy of the Baseline. Adding our proposed strategy for handling missing modalities further improves verb and action accuracy. Knowledge distillation yields the largest gains across all metrics, and extending training to 100 epochs provides an additional boost. These results demonstrate that our KARMMA enhancements not only increase the robustness to missing modalities (see Tab. I and Fig. 4) but also enhance the overall accuracy.

**Knowledge Distillation.** The balance between teacher supervision (knowledge distillation) and task supervision (cross-entropy) is controlled by the coefficient  $\alpha$  in Eq. (4). Therefore, Tab. IV reports the student noun, verb and action accuracy for different  $\alpha$  values. The results highlight the importance of properly balancing teacher and task supervision: no reliance on the teacher ( $\alpha=1.0$ ) fails to leverage the additional

TABLE III: Analysis of the KARMMA enhancements. “Baseline” uses the same architecture as our student (see Sec. III-B) but is trained end-to-end with cross-entropy loss and without the KARMMA enhancements (see Sec. III-D). “Longer training” denotes 100 training epochs. Blue highlights our final student. **Bold** and underline values indicate the best and second-best results.

Enhancement	Noun Acc. (%)	Verb Acc. (%)	Action Acc. (%)
Baseline	50.08	64.74	38.94
+ Modality dropout	51.30	65.22	39.82
+ Missing modality strategy	50.60	65.61	39.87
+ Knowledge distillation	<u>53.05</u>	67.24	41.68
+ Longer training	<b>54.80</b>	<b>67.65</b>	<b>43.00</b>

TABLE IV: Impact of knowledge distillation. The “ $\alpha$ ” column corresponds to the value used in Eq. (4) that balances the cross-entropy and distillation losses. Blue highlights the value used in our final student. **Bold** and underline values indicate the best and second-best results.

$\alpha$	Noun Acc. (%)	Verb Acc. (%)	Action Acc. (%)
1.0	50.60	65.61	39.87
0.7	<b>53.05</b>	<b>67.24</b>	<b>41.68</b>
0.4	<u>52.03</u>	<u>66.07</u>	<u>40.17</u>
0.1	49.98	64.51	37.82

TABLE V: Effect of token reduction. “None” keeps all tokens, while “ $\Theta$ -Average” applies our proposed strategy (see Sec. III-C). Memory usage refers to the average GPU memory consumed by the fusion block during training. The top-1 action recognition accuracy is computed using our teacher, “KARMMA<sub>T</sub>”. Blue highlights the configuration used in our final teacher and student. **Bold** and underline values indicate the best and second-best results.

Token Reduction	# of Tokens per Mod.	GFLOPs	Mem. Usage of FB (GB)	Action Acc. (%)
None	785	1759	14.28	<b>35.26</b>
$\Theta$ -Average	500	1725	6.35	34.76
$\Theta$ -Average	<u>300</u>	<u>1707</u>	<u>2.65</u>	<u>34.99</u>
$\Theta$ -Average	<b>100</b>	<b>1693</b>	<b>0.50</b>	32.62

guidance it provides, while too much reliance ( $\alpha=0.1$ ) leads to overfitting to its outputs and a drop in accuracy. The optimal balance is achieved with  $\alpha=0.7$ , yielding an absolute action accuracy gain of 1.81% (4.5% relative) over training without distillation ( $\alpha=1.0$ ).

**Token Reduction.** Tab. V shows the impact of token reduction in the fusion block of our teacher. Using all tokens yields the highest accuracy, but our parameter-free strategy ( $\Theta$ -Average) with  $\Theta=300$  offers the best accuracy–efficiency

trade-off: it reduces memory usage by 11.63 GB (81.45 % relative) and lowers GFLOPs, with only a 0.27 % absolute drop in accuracy. In more constrained scenarios, a smaller threshold ( $\Theta=100$ ) further reduces computational cost while maintaining competitive accuracy.

**Modality Dropout.** During training we apply different modality dropout rates and evaluate their impact at inference. Fig. 5 reports the resulting action accuracy across various modality combinations, and Tab. VI summarizes these results using three metrics: average action accuracy, average relative drop compared to the full-modality setting (V+F+A), and average rank. A 50 % dropout rate achieves the best trade-off, yielding an absolute accuracy gain of 8.15 % (50.7 % relative) and a 25.42 % absolute reduction in relative drop (41.4 % relative) compared to no dropout. These results highlight the importance of selecting an appropriate dropout rate: too high rates ( $\geq 75\%$ ) degrade overall accuracy, while too low rates ( $\leq 25\%$ ) fail to build sufficient robustness, resulting in larger accuracy drops when modalities are missing.

**Resource Efficiency.** Fig. 6 compares the average memory usage and GFLOPs of the teacher and student at inference time across different modality combinations. The results show that, in all cases, our KARMMA student reduces memory consumption by approximately 50 % and significantly lowers GFLOPs, resulting in a lightweight model with fast inference. The largest savings occur when only audio is used, as the student processes it with an AST-T [36], which is significantly smaller than the Swin-T [31] used for RGB video and optical flow. These savings are especially important for edge and on-robot deployment, where memory budgets are tight and fast inference is desired.

## V. CONCLUSIONS

In this work, we introduced *KARMMA*, a novel multimodal-to-multimodal distillation framework for egocentric action recognition that does not require modality alignment across samples during training or inference. The resulting student is lightweight, fast, and benefits from the availability of multiple modalities while remaining robust to missing ones. Unlike unimodal approaches, our KARMMA student supports *any* combination of the training modalities without retraining, making it well-suited to real-world multimodal scenarios, including edge and on-robot deployment for applications such as human-robot interaction.

Our framework also reduces training costs by using frozen pre-trained unimodal feature extractors for the teacher, simplifying the integration of newer encoders as they become available. Additionally, our simple parameter-free token reduction strategy improves computational efficiency without sacrificing accuracy. Overall, KARMMA outperforms the state-of-the-art multimodal-to-unimodal distillation method for egocentric action recognition, achieving higher accuracy while reducing parameter count and GFLOPs. Future work will focus on integration into robotic systems and on extending KARMMA to additional modalities to broaden its applicability across platforms and tasks.

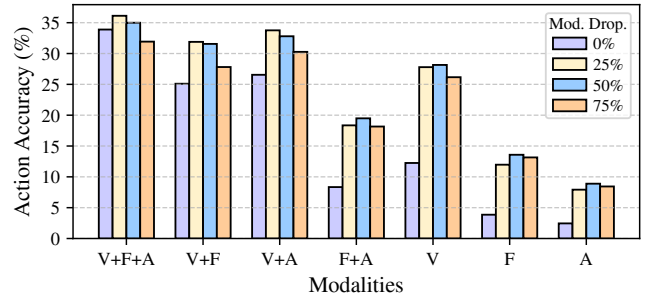


Fig. 5: **Impact of modality dropout rate during training.** Each bar shows the top-1 action recognition accuracy of our teacher (see Sec. III-B) trained with a specific modality dropout rate and evaluated across different modality combinations. “V,” “F,” and “A” denote RGB video, optical flow, and audio, respectively.

TABLE VI: **Summary of the impact of modality dropout rate during training.** “Average Accuracy” refers to the mean action accuracy across all modality combinations. “Average Relative Drop” denotes the mean relative accuracy drop compared to the full-modality setting (V+F+A). “Average Rank” is the mean rank across modality combinations. All values are derived from Fig. 5. Blue highlights the modality dropout rate used in our final teacher and student. **Bold** and underline values indicate the best and second-best results.

Modality Dropout (%)	Average Accuracy (%)	Average Relative Drop (%)	Average Rank
0	16.06	61.37	3.86
25	<u>23.97</u>	39.24	<u>1.86</u>
50	<b>24.21</b>	<u>35.95</u>	<b>1.43</b>
75	22.27	<b>35.28</b>	2.86

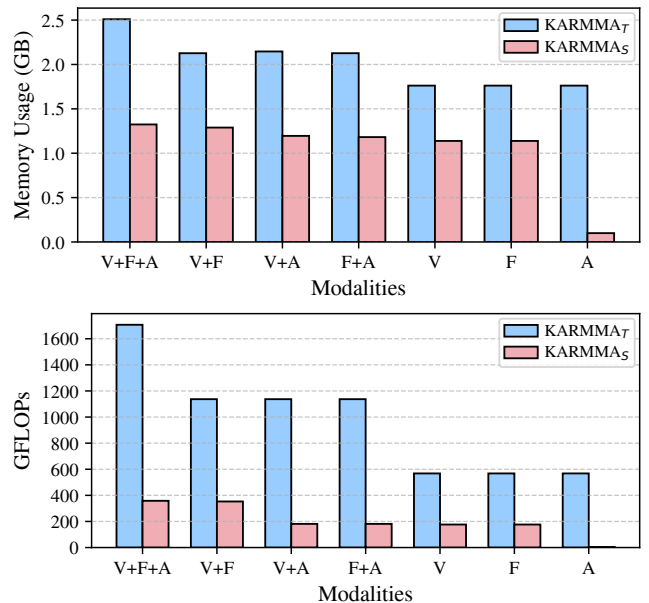


Fig. 6: **Resource efficiency of KARMMA student.** “KARMMA<sub>T</sub>” and “KARMMA<sub>S</sub>” denote our teacher and student, respectively (see Sec. III-B). The top figure compares their average GPU memory usage for different modality combinations, while the bottom figure compares their GFLOPs, both measured at inference. “V,” “F,” and “A” denote RGB video, optical flow, and audio, respectively.

## ACKNOWLEDGMENTS

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany’s Excellence Strategy (EXC-3057/1 “Reasonable Artificial Intelligence”, Project No. 533677015). We gratefully acknowledge support from the hessian.AI Service Center (funded by the Federal Ministry of Research, Technology and Space, BMFTR, grant No. 16IS22091) and the hessian.AI Innovation Lab (funded by the Hessian Ministry for Digital Strategy and Innovation, grant No. S-DIW04/0013/003). SSM has been funded by the DFG – 529680848. MSV, APY, JBC, and JGG further acknowledge support by projects PID2024-158322OB-I00 and PID2021-125209OB-I00, (MCIN/AEI/10.13039/501100011033/ FEDER, UE), project JIUZ2024-IyA-07, DGA 2022-2026 scholarship, and Grant SMT Erasmus+, project 2022-1-ES01-KA131-HED-000065592 funded by Campus Iberus.

## REFERENCES

- [1] K. Lee, A. Shrivastava, and H. Kacorri, “Hand-priming in object localization for assistive egocentric vision,” in *WACV*, 2020, pp. 3411–3421.
- [2] I. Fang, Y. Chen, Y. Wang, J. Zhang, Q. Zhang, J. Xu, X. He, W. Gao, H. Su, Y. Li, and C. Feng, “EgoPAT3Dv2: Predicting 3D action target from 2D egocentric vision for human-robot interaction,” in *ICRA*, 2024.
- [3] S. Ardeshir and A. Borji, “Integrating egocentric videos in top-view surveillance videos: Joint identification and temporal alignment,” in *ECCV*, 2018, pp. 300–317.
- [4] K. Grauman, A. Westbury, L. Torresani, *et al.*, “Ego-Exo4D: Understanding skilled human activity from first-and third-person perspectives,” in *CVPR*, 2024, pp. 19383–19400.
- [5] D. Damen, H. Doughty, G. M. Farinella, *et al.*, “Scaling egocentric vision: The EPIC-KITCHENS dataset,” in *ECCV*, 2018, pp. 720–736.
- [6] G. Radevski, D. Grujicic, M. Blaschko, M.-F. Moens, and T. Tuytelaars, “Multimodal distillation for egocentric action recognition,” in *ICCV*, 2023, pp. 5213–5224.
- [7] D. Papanagiotou, G. Senterri, and S. Manitsaris, “Egocentric gesture recognition using 3D convolutional neural networks for the spatiotemporal adaptation of collaborative robots,” *Frontiers in neurorobotics*, vol. 15, p. 703545, 2021.
- [8] P. Akiva, J. Huang, K. J. Liang, R. Kovvuri, X. Chen, M. Feiszli, K. Dana, and T. Hassner, “Self-supervised object detection from egocentric videos,” in *ICCV*, 2023, pp. 5225–5237.
- [9] A. Radford, J. W. Kim, C. Hallacy, *et al.*, “Learning transferable visual models from natural language supervision,” in *ICML*, 2021, pp. 8748–8763.
- [10] X. Gong, S. Mohan, N. Dhir, J.-C. Bazin, Y. Li, Z. Wang, and R. Ranjan, “MMG-Ego4D: Multimodal generalization in egocentric action recognition,” in *CVPR*, 2023, pp. 6481–6491.
- [11] R. Wu, H. Wang, H.-T. Chen, and G. Carneiro, “Deep multimodal learning with missing modality: A survey,” *Trans. Mach. Learn. Res.*, 2026.
- [12] M. Ma, J. Ren, L. Zhao, D. Testuggine, and X. Peng, “Are multimodal transformers robust to missing modality?” in *CVPR*, 2022, pp. 18177–18186.
- [13] H. Wang, M. K. Singh, and L. Torresani, “Ego-Only: Egocentric action detection without exocentric transferring,” in *ICCV*, 2023, pp. 5250–5261.
- [14] K. Min and J. J. Corso, “Integrating human gaze into attention for egocentric activity recognition,” in *WACV*, 2021, pp. 1069–1078.
- [15] E. Kazakos, J. Huh, A. Nagrani, A. Zisserman, and D. Damen, “With a little help from my temporal context: Multimodal egocentric action recognition,” in *BMVC*, 2021, p. 268.
- [16] R. Goyal, S. E. Kahou, V. Michalski, *et al.*, “The “something something” video database for learning and evaluating visual common sense,” in *ICCV*, 2017, pp. 5842–5850.
- [17] K. Grauman, A. Westbury, E. Byrne, *et al.*, “Ego4d: Around the world in 3,000 hours of egocentric video,” in *CVPR*, 2022, pp. 18995–19012.
- [18] H. Dong, I. Nejjar, H. Sun, E. N. Chatzi, and O. Fink, “SimMMDG: A simple and effective framework for multimodal domain generalization,” in *NeurIPS*, 2023.
- [19] M. Hatano, R. Hachiuma, R. Fujii, and H. Saito, “Multimodal cross-domain few-shot learning for egocentric action recognition,” in *ECCV*, 2024, pp. 182–199.
- [20] S. Pramanick, Y. Song, S. Nag, K. Q. Lin, H. Shah, M. Z. Shou, R. Chellappa, and P. Zhang, “EgoVLPv2: Egocentric video-language pre-training with fusion in the backbone,” in *ICCV*, 2023, pp. 5285–5297.
- [21] K. Q. Lin, J. Wang, M. Soldan, *et al.*, “Egocentric video-language pretraining,” in *NeurIPS*, 2022.
- [22] M. Ramazanova, A. Pardo, H. Alwassel, and B. Ghanem, “Exploring missing modality in multimodal egocentric datasets,” in *CVPRW*, 2025, pp. 75–85.
- [23] M. Ramazanova, A. Pardo, B. Ghanem, and M. Alfarra, “Test-time adaptation for combating missing modalities in egocentric videos,” in *ICLR*, 2025.
- [24] M. Fayyaz, S. A. Koohpayegani, F. R. Jafari, S. Sengupta, H. R. V. Joze, E. Sommerlade, H. Pirsiavash, and J. Gall, “Adaptive token sampling for efficient vision transformers,” in *ECCV*, 2022, pp. 396–414.
- [25] Y. Shang, M. Cai, B. Xu, Y. J. Lee, and Y. Yan, “LLava-PruMerge: Adaptive token reduction for efficient large multimodal models,” in *CVPR*, 2025, pp. 22857–22867.
- [26] A. Boukerche, L. Zheng, and O. Alfandi, “Outlier detection: Methods, models, and classification,” *ACM Comput. Surv.*, vol. 53, no. 3, pp. 1–37, 2020.
- [27] D. Bolya, C.-Y. Fu, X. Dai, P. Zhang, C. Feichtenhofer, and J. Hoffman, “Token merging: Your ViT but faster,” in *ICLR*, 2023.
- [28] D. Marin, J.-H. R. Chang, A. Ranjan, A. Prabhu, M. Rastegari, and O. Tuzel, “Token pooling in vision transformers for image classification,” in *WACV*, 2023, pp. 12–21.
- [29] S. Wei, C. Luo, and Y. Luo, “MMANet: Margin-aware distillation and modality-aware regularization for incomplete multimodal learning,” in *CVPR*, 2023, pp. 20039–20049.
- [30] A. Nagrani, S. Yang, A. Arnab, A. Jansen, C. Schmid, and C. Sun, “Attention bottlenecks for multimodal fusion,” in *NeurIPS*, 2021, pp. 14200–14213.
- [31] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, “Swin transformer: Hierarchical vision transformer using shifted windows,” in *ICCV*, 2021, pp. 10012–10022.
- [32] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman, “The kinetics human action video dataset,” *arXiv:1705.06950 [cs.CV]*, 2017.
- [33] G. Radevski, M.-F. Moens, and T. Tuytelaars, “Revisiting spatio-temporal layouts for compositional action recognition,” in *BMVC*, 2021.
- [34] J. Ji, R. Krishna, L. Fei-Fei, and J. C. Niebles, “Action genome: Actions as compositions of spatio-temporal scene graphs,” in *CVPR*, 2020, pp. 10236–10247.
- [35] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” in *ICLR*, 2019.
- [36] Y. Gong, Y.-A. Chung, and J. Glass, “AST: Audio spectrogram transformer,” in *Interspeech*, 2021, pp. 571–575.