X-MIC: Cross-Modal Instance Conditioning for Egocentric Action Generalization

Anna Kukleva[1,2*] Fadime Sener[1] Edoardo Remelli[1] Bugra Tekin[1] Eric Sauser[1]
Bernt Schiele[2] Shugao Ma[1]
[1]Meta Reality Labs; [2]Max Planck Institute for Informatics, Saarland Informatics Campus
{annakukleva, famesener}@meta.com

Abstract

Lately, there has been growing interest in adapting vision-language models (VLMs) to image and third-person video classification due to their success in zero-shot recognition. However, the adaptation of these models to egocentric videos has been largely unexplored. To address this gap, we propose a simple yet effective cross-modal adaptation framework, which we call X-MIC. Using a video adapter, our pipeline learns to align frozen text embeddings to each egocentric video directly in the shared embedding space. Our novel adapter architecture retains and improves generalization of the pre-trained VLMs by disentangling learnable temporal modeling and frozen visual encoder. This results in an enhanced alignment of text embeddings to each egocentric video, leading to a significant improvement in cross-dataset generalization. We evaluate our approach on the Epic-Kitchens, Ego4D, and EGTEA datasets for fine-grained cross-dataset action generalization, demonstrating the effectiveness of our method.[1]

1. Introduction

Egocentric action recognition has recently become a popular research topic due to the rising interest in augmented reality and robotics. Recently, two large-scale egocentric datasets Epic-Kitchens [6] and Ego4D [10], capturing the daily activities of users have been introduced. While there is a growing interest in studying action recognition on egocentric datasets, evaluations primarily occur within the same dataset; lacking cross-dataset evaluations that is crucial for real-world deployment of recognition models. Testing models on different datasets presents several challenges, such as encountering unfamiliar environments, different users, and previously unseen objects and their corresponding actions, all of which can significantly impact performance. Recently, vision-language models [13, 27, 39] such as CLIP [27] have demonstrated remarkable performance across diverse third-persons datasets like Kinetics-600 [16] and ImageNet [7], showcasing their ability to generalize effectively and achieving zero-shot performance of 59.8% and 76.2%, respectively. However, their zero-shot performance drops significantly when applied to egocentric datasets like Epic-Kitchens, with noun and verb recognition reaching only 8.8% and 5.9%, respectively; highlighting the domain gap between third-person and egocentric data.

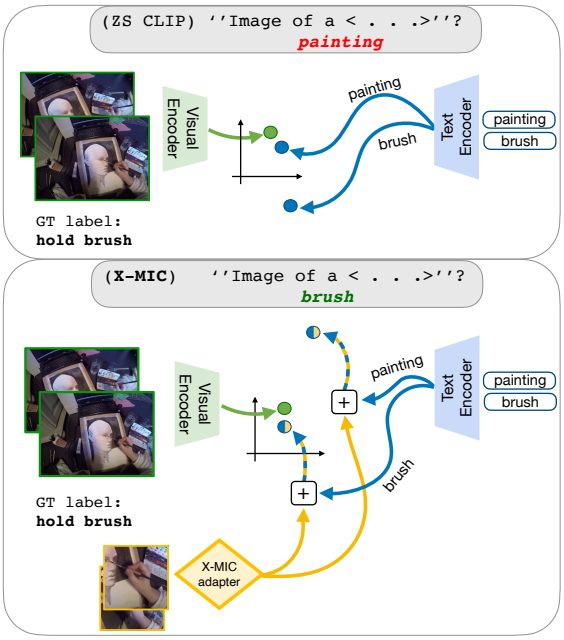

Figure 1. **Egocentric video classification with VL models.** **Top:** Standard zero-shot CLIP. As the dominant object in the scene is painting, the model predicts class "painting" while the object of interest is "brush". **Bottom:** CLIP model with our X-MIC adaptation directly in the shared VL embedding. X-MIC vectors adapt focus of the CLIP model to the hand area, guiding text modality to capture egocentric domain-specific information.

CLIP's zero-shot generalization to new datasets lever-

[1]https://github.com/annusha/xmic
*work is done during internship at Meta

ages learning a shared embedding space for text and visual modalities. To enhance generalization to new domains, a prominent research direction [44] explores adapting the text encoder by appending trainable *prompt* tokens to class tokens, modifying the class-text input from "a photo of an apple" to "<learnable prompt> apple". As an alternative approach, recent work has proposed to train feature *adapters* on both the visual and textual domains [5, 8], drawing insights from the NLP works [12, 34]. Despite their promising results, these methods overlook the inherent characteristics of the egocentric video domain. To overcome this, we propose a simple yet effective adapter architecture, injecting egocentric video-specific knowledge into a frozen VL embedding space, depicted in Fig. 1. Our method transforms each video through an adapter into a vector for cross-modal instance conditioning of text — referred to as X-MIC-vector. Our cross-modal adaptation performed directly in the embedding space results in significantly improved efficiency during training and testing. Moreover, our new adapter module disentangles frozen visual encoder from the visual temporal modeling through cross-modal adaptation. Each X-MIC-vector is video-specific, therefore, allowing us to align any frozen text to each input video individually. Finally, to align the text embedding to the video, we simply add the X-MIC-vector to the text embedding vectors.

We extensively evaluate our approach on Epic-Kitchens [6], Ego4D [10] and EGTEA [20] datasets, demonstrating superior generalization compared to SOTA VL-adaptation methods.

Our contributions can thus be summarized as:

- addressing the task of egocentric cross-dataset and zero-shot action recognition with VLMs that is designed for real-world applications, e.g. AR, addressing the impracticality of collecting data from every new environment,
- a simple yet effective framework, referred to as X-MIC, for cross-modal adaption of VL models directly in the pre-trained VL embedding space; our module disentangles temporal modeling from the frozen visual encoder,
- a new egocentric spatial-temporal attention module enhances information around hands, thereby improving egocentric action recognition performance,
- thorough comparisons with respect to image and video state-of-the-art VL adaptation methods which demonstrate the effectiveness of our approach.

2. Related Work

Egocentric Action Generalization. While egocentric vision gained attention with datasets like Epic-Kitchens [6] and Ego4D [10], current state-of-the-art [17, 25, 29, 30, 36–38, 40, 45] primarily focus on intra-dataset evaluation, which limits their applicability to real-world scenarios. Several methods fine-tuned CLIP on egocentric datasets [21, 26, 41], yet generalization on fine-grained verbs and nouns recognition remains underexplored. Our work comprehensively investigates both intra-dataset and inter-dataset generalization on both verbs and nouns.

Prompt Learning and Adapters. Prompt learning in NLP [9, 14, 19, 32, 42] adapts frozen text models by appending task-specific information. Extending this to image recognition, CoOp [44] learns appendable vectors in the text token space. CoCoOp [43] introduces image-conditioned prompt learning, boosting performance but with high computational costs. MaPLe [18] leverages shared deep prompts for text and visual encoders, while PromptSRC [28] suggests regularizing constraints for frozen encoders. Chen *et al.* [4] found that prompting boosts model transferability in tasks with fewer number of visual tokens, like image classification, but has limited impact in tasks with more tokens, such as video understanding. Thus, an alternative research direction explores adapting vision-language models with feature adapters [12]. Clip-adapter [8] learns new features through an additional bottleneck layer and blends them with original pre-trained features in a residual style. Our approach falls under the adapter category. Unlike previous visual adapters, we introduce cross-modal instance conditioning specifically designed for egocentric video recognition.

Adapting VLMs to Videos. Recent advancements in prompt learning extend to third-person videos. A5/A6 [15] introduces a temporal module atop visual encoder, keeping both encoders frozen. EVL [22] discards the text encoder, relying solely on temporally encoded frame features by visual encoder. Vita-CLIP [1] uses shallow prompts on the text encoder, similar to [44], and introduces deep temporal prompts for the visual encoder. Recently, OAP [3] generalizes the verbs observed during training to an open vocabulary of objects with a prompt-based object encoder on egocentric videos. Our method builds on existing work while introducing an adapter architecture specifically tailored to egocentric domain, resulting in superior performance.

3. X-MIC Adaptation Approach

We begin by introducing the preliminaries such as classification with VLMs like CLIP and different types of VL adaptations in Sec. 3.1. Then, in Sec. 3.2, we give an overview of our adapter method for text conditioning and present our egocentric-spatio-temporal attention module.

3.1. Preliminaries and Baselines on VL Adaptation

Vision-language models (VLMs), such as CLIP, demonstrate effective zero-shot generalization across various downstream tasks for image recognition and third-person video recognition. However, certain domains, like egocentric videos, still face challenges due to a significant gap between web-collected and egocentric data.

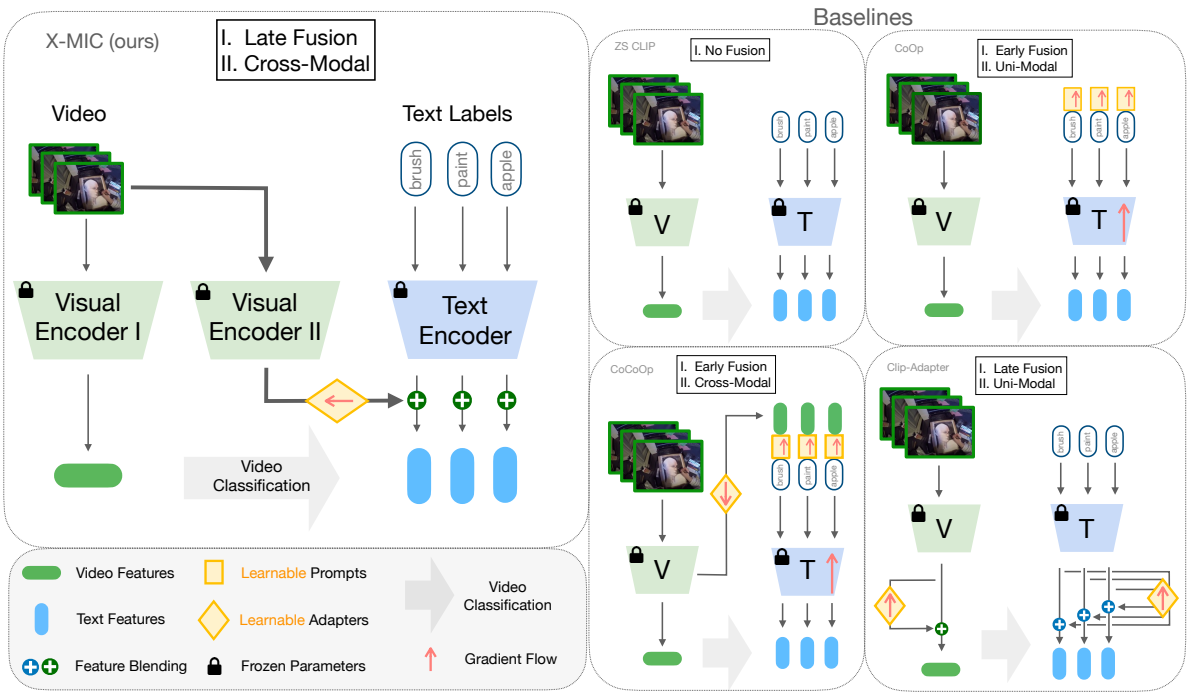

Figure 2. **Overview of our X-MIC method and previous adaptation methods of VLMs.** Baselines: **No Fusion** is a standard zero-shot video classification method. The average of the frame representations is compared to text representations in the shared VL embedding space. **Early Fusion & Uni-Modal** is a prompt learning method, where the learnable parameters are concatenated to text tokens and optimized through the text encoder. Subsequently, the text encoder is adapted to the new domain. **Early Fusion & Cross-Modal** is an extension of Early Fusion & Uni-Modal method, where additional learnable parameters are introduced in the form of an adapter. This adapter maps video representations to embedding space of text tokens, which are then concatenated to learnable prompts and text tokens. Memory consumption, required for forward-backwards pass through the text encoder, expands with respect to all combinations of all text-labels and videos in the batch. **Late Fusion & Uni-Modal** is a method, where adaptation of both encoders is based on the feature blending of original text and video representations with the adapted corresponding representations. Ours: **X-MIC** adaptation method falls in *Late Fusion & Cross-Modal* category. Adapted video features are blended with the original text features. Simple adaptation of text modality to each individual video is efficient as it does not require gradient propagation through text or video encoders. Additionally, we propose to employ Visual Encoder II, offering flexibility in utilizing various types of visual features for conditioning. Note that Visual Encoder I and II can be represented by a single visual encoder, such as the CLIP visual encoder.

Below, we provide an overview existing prompt learning and adapter-based methods.

Video Classification with VL Dual Encoders. Trained on hundreds of millions of text and visual data pairs, VL dual encoders bring the two modalities together in a shared embedding space. When evaluating models pre-trained on extensive web data, a crucial metric is their ability to transfer to other downstream tasks without additional fine-tuning, a process commonly known as zero-shot evaluation. To perform zero-shot classification, one needs to propagate a set of C predefined classes in the form of text, denoted as $t =$ "Image of a <class>" through a pre-trained text encoder $T(\cdot)$. This process extracts individual text embeddings, represented as $e_t = T(t) \in \mathcal{R}^{1\times D}$ for each class. Subsequently, these vectors undergo $l2$ normalization, resulting in $\bar{e}_t = \frac{e_t}{||e_t||}$ (hereafter, the overline symbol $\bar{e}$ indicates $l2$ normalization of vector e). Then a matrix $\bar{E}_T \in \mathcal{R}^{C\times D}$ is constructed, representing a simple linear classifier, and is thus referred to as the text-based classifier. To classify an input video v, we sample N frames, denoting the sampled frames as $v' = \{z_i\}^N$, where z_i represents a frame from the video v. Subsequently, all sampled frames are mapped to the shared VL embedding space, using the frozen visual encoder $V(\cdot)$. Applying average pooling over the embeddings of the frames yields a single-vector video representation: $\bar{e}_{v'} = avg\_pool(\{\overline{V(z_i)}\}^N) \in \mathcal{R}^{1\times D}$. The video vector $\bar{e}_{v'}$ is then classified using the text-based classifier $\bar{E}_T$.

No Fusion. We refer to frozen dual encoders $T(\cdot)$ and $V(\cdot)$ without additional adaptation as "No Fusion" baseline.

Early Fusion and Uni-Modal Adaptation. A prompt learning-based method, CoOp [44], introduces P learnable vectors appended to all C input text classes in the token embeddings of the textual encoder (see Fig. 2). To optimize these prompts, gradients are propagated through the frozen

text encoder for $C \times P \times D$ adaptable parameters, where D is the dimensionality of the tokens. This optimization remains independent of the batch size of the visual input.

Early Fusion and Cross-Modal Adaptation. A follow-up work, CoCoOp [43], extends learnable text prompts to cross-modal prompts by introducing an adapter module of the frozen visual encoder to the token embedding space (see Fig. 2). In this architecture, each of the C class-tokens are appended not only with P learnable text prompts but also with individual input-conditioned prompts generated by the adapter. Optimizing these prompts for a batch of size B involves propagating $B \times P \times D \times C$ gradients, making training inefficient and slow as shown in [43].

Late Fusion and Uni-Modal Adaptation. CLIP-Adapter [8] adopts a late fusion approach as an alternative to early fusion adaptation. The text and visual encoders are followed by uni-modal adapter modules that generate adapted uni-modal feature vectors. These adapted features are then fused with the corresponding original features in the VL embedding space, subsequently optimized with the standard classification loss. This optimization is efficient due to the lightweight nature of adapters, eliminating the need for heavy text-encoder gradient propagation.

3.2. X-MIC Adaptation

Overview. We aim at achieving generalization in egocentric action recognition across domains and to novel action classes. Our X-MIC-adaptation framework is designed to improve the alignment between frozen text representations and the egocentric visual domain directly within the VL embedding space. To adapt the text modality to the egocentric domain, we introduce a simple cross-modal text conditioning operation based on the input videos. Specifically, each X-MIC-vector serves as an adapted video representation. We align any frozen text representation to each individual input video by a simple addition operation with the X-MIC-vector. Consequently, text representations are adapted to individual input videos, and these adapted text embeddings are further utilized for the classification of corresponding videos into fine-grained noun and verb classes. Moreover, by introducing an egocentric-spatio-temporal attention module, we aggregate temporal information between video frames and emphasize areas around hands to enhance hand-object interactions. X-MIC-vectors offer dual benefits: a simple and efficient cross-modal conditioning approach, and the decoupling of domain-specific knowledge from the frozen VL embedding, resulting in improved generalization on egocentric videos.

X-MIC Adaptation. Our adaptation method, X-MIC, aligns frozen text class embeddings directly to the new domain in the shared VL embedding space. During training and inference, our approach resembles zero-shot classification, as we classify frozen video representations from the original visual backbone $V(\cdot)$ using an adapted text-based classifier $\bar{E}_T$ tailored to each input video v. This enables efficient domain adaptation without the need for fine-tuning the entire model, categorizing our method as late fusion with cross-modal adaptation.

Specifically, for an input video v, we sample N frames to form a sparse video sequence $v' = \{z_i\}^N$. The video sequence v' is then decoded using the original visual encoder $V(\cdot)$, resulting in a single vector $\bar{e}_v$. Additionally, we encode the C classes into the text-based classifier $\bar{E}_T$ as detailed in Sec. 3.1.

To generate the X-MIC-vector, we introduce a second frozen visual encoder, denoted as $V_{II}(\cdot)$. This secondary encoder can either be an identical copy of the original encoder $V(\cdot)$ or a distinct pre-trained encoder. In Sec. 4.1, we demonstrate that incorporating a different type of $V_{II}(\cdot)$ can result in significant generalization improvements. For instance, DINO [23], which is uni-modal, captures distinct characteristics [24] of the visual input compared to multi-modal CLIP like models that focus solely on main objects.

We employ the second encoder $V_{II}(\cdot)$ to produce an intermediate representation of frames, denoted as $x_v = \{V_{II}(z_i)\}^N$. Before adapting the intermediate representation, we apply $l2$-normalization to the vector. See Sec. 4.4 for a detailed analysis of the impact of this normalization. Our video adapter $A(\cdot)$ incorporates a temporal aggregation module. By feeding these intermediate representations into this module, we obtain the final X-MIC-vector for adaptation, represented as $a_v = A(\bar{x}_v)$.

Finally, to adapt the frozen text-based classifier $\bar{E}_T$ to the video v, we simply sum X-MIC-vector with each class representation in the embedding space: $\overline{e_t + a_v} \in \mathcal{R}^D$, and when combined, these updated vectors form an adapted text-based classifier $\overline{E}_T^{a_v}$. Subsequently, we classify the video representation $\bar{e}_v$ with the adapted text-based classifier $\overline{E}_T^{a_v}$. The process of classification with X-MIC-adaptation can be summarized as follows:

$$c = \text{argmax}_t < \overline{e_t + A(\overline{V_{II}(x_v)})}, \bar{e}_v >, \qquad (1)$$

where c represents the class with the highest similarity between the adapted text-based classifier $\overline{E}_T^{a_v}$ and the video v, and $< \cdot, \cdot >$ denotes dot product.

Ego-Spatio-Temporal Attention Module. Our adaptation module consist of two transformer blocks $b_S(\cdot)$ and $b_T(\cdot)$ designed to aggregate different types of information. This module not only adapts each video to the shared VL embedding space but also captures egocentric video-specific spatial and temporal information, through $b_S(\cdot)$ and $b_T(\cdot)$ respectively. To capture hand-object interactions better, we introduce an attention block that focuses on regions around

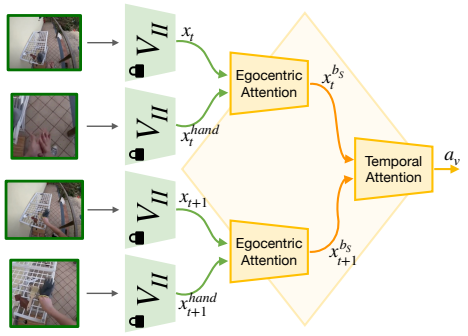

Figure 3. **Ego-Spatio-Temporal Attention Module.** It takes a sequence of full frames interleaved with hand crops as input, and outputs X-MIC vector a_v, representing video v as a single vector for text conditioning in the shared VL embedding space.

hands, see Fig. 3. This involves applying self-attention between hand crops and full frames, guiding the model to emphasize information around hands, a crucial region of interest in egocentric videos. To aggregate temporal information from the video, we employ a temporal self-attention block, similar to [15] that updates the frame representations. Our X-MIC-vector for adaptation is derived by applying average pooling over all updated frame representations.

Egocentric videos may include diverse backgrounds and involve significant camera motion. By focusing on the region around hands through cropping and applying self-attention to both the full frame and the cropped region, we guide our model to prioritize attention on the hands. More specifically, for each frame, we use the full frame z_i and the cropped hand region z_i^{hand} from the same frame. We get the intermediate representations through the video encoder, for the full frame $x_i = V_{II}(z_i)$ and the hand region $x_i^{hand} = V_{II}(z_i^{hand})$. We concatenate the two to obtain an intra-frame sequence $[x_i; x_i^{hand}] \in \mathcal{R}^{2\times D}$. We derive the intra-frame representation $x_i^{b_S}$ by averaging the updated representations from both the full and cropped frames:

$$x_i^{b_S} = \text{avg\_pool}(b_S([x_i; x_i^{hand}])). \quad (2)$$

To capture the temporal relations across frames, we apply self-attention between all frames of the video. Specifically, we use the second transformer $b_T(\cdot)$ block to update $x_i^{b_S}$ frame representations, which we aggregate with average pooling into our X-MIC-vector:

$$a_v = \text{avg\_pool}(b_T([x_1^{b_S}, x_2^{b_S}, \cdots, x_N^{b_S}])). \quad (3)$$

In this way, our adaptation module effectively incorporates egocentric video-specific spatial and temporal information into the frozen vision-language embedding space, enhancing generalization to novel classes.

4. Experiments

X-MIC is mainly evaluated on the cross-dataset setting between two large-scale egocentric datasets: Ego4D [10] and Epic-Kitchens [6]. We also evaluate generalization performance on the small-scale EGTEA [20] dataset.

4.1. Datasets

Ego4D [10]. We a subset of Ego4D [10] annotated with fine-grained noun and verb labels, specifically from the FHO benchmark which contains 521 noun and 117 verb classes. The training set consists of 64K video clips, while the testing set comprises 33K clips. The average clip duration is 8 seconds, resulting in a total of approximately 215 hours of videos, excluding irrelevant background clips.

Epic-Kitchens [6]. We use the Epic-Kitchens100, comprising 67K video clips for training and 10K video clips for testing. The average clip length is 3.5 seconds, totaling about 70 hours, excluding irrelevant background clips. The dataset features annotations for 300 noun classes and 97 verb classes, focusing on kitchen-related topics.

EGTEA [20] We use this dataset solely for model testing, given its training set of 8,000 video clips with clip length of 3.2 seconds. During inference, we combine three test splits resulting in 6K video clips in total. The dataset is annotated with fine-grained 20 verb and 54 noun classes.

4.2. Implementation Details

We evaluate the generalization performance based on the adaptation of the pre-trained CLIP ViT-B/16 model, unless otherwise specified. For model training, we use AdamW with(β_1, β_2) = (0.9, 0.999) and weight decay of 0.01 for 15 epochs with a fixed learning rate of 1e-6. The Transformer module b_1 contains 1 self-attention layer, whereas temporal attention module b_2 includes 2 self-attention layers. During training, we sample 16 random frames, and during evaluation we sample frames uniformly. For detecting the hand regions, we use the 100DOH [31] detector, extracting bounding boxes for each frame. Further implementation details are provided in the supplementary.

Cross-Datasets Evaluation. In our work, we investigate the generalization performance of fine-grained noun and verb recognition across egocentric datasets. Our objective is two-fold: achieving strong performance within the dataset on the corresponding test set and demonstrating robust generalization to a shared dataset (Table 1). We compute the harmonic mean between these two to gauge the balance of different types of generalization. For instance, we train our models on Ego4D and subsequently evaluate on both the Ego4D and Epic-Kitchens test sets. We further analyse zero-shot generalization by identifying disjoint subsets of shared and novel classes across datasets (Table 2). See supplementary for corresponding classes.

		Trained on Ego4D (E4D)						Trained on Epic-Kitchens (EK)					
	ta	Nouns			Verbs			Nouns			Verbs		
Evaluation dataset		E4D	EK	hm	E4D	EK	hm	EK	E4D	hm	EK	E4D	hm
ZS CLIP	-	5.89	8.74	7.03	2.18	4.25	2.88	8.74	5.89	7.03	4.25	2.18	2.88
CoOp	-	28.22	10.87	15.70	22.57	20.42	21.44	21.56	9.37	13.06	30.91	13.35	18.64
Co-CoOp	-	30.00	9.51	14.44	21.31	12.99	16.14	24.23	9.27	13.41	34.16	14.17	20.03
CLIP-Adapter	-	30.00	8.95	13.78	22.82	19.94	21.28	33.21	5.73	9.77	36.70	16.09	22.37
CLIP-Adapter*	✓	31.26	10.00	15.16	27.32	22.28	24.54	34.40	4.67	8.22	48.69	15.52	23.54
A5	✓	31.39	7.84	12.55	26.31	22.77	24.41	32.04	3.31	5.99	46.05	17.93	25.81
Vita-CLIP	✓	33.52	10.61	16.11	22.66	25.81	24.13	34.41	9.52	14.91	48.78	13.47	21.11
X-MIC	✓	33.54	15.35	21.06	28.93	26.48	27.65	30.64	12.32	17.57	50.01	18.10	**26.58**
X-MIC+DINO	✓	35.85	18.96	**24.80**	28.27	29.49	**28.86**	44.07	11.45	**18.17**	53.02	16.01	24.60

Table 1. **SOTA comparison on within- and cross-dataset evaluation on Ego4d and Epic Kitchens datasets.** Left: Models trained on Ego4D. Right: Models trained no Epic-Kitchens. Evaluation is on noun and verb classes. ta denotes temporal attention in the corresponding method, other methods apply simple average of the frames. hm stands for harmonic mean evaluation. X-MIC+DINO denotes our model with DINO [23] as Visual Encoder II.

		Trained on E4D, Evaluated on EK						Trained on EK, Evaluated on E4D					
	ta	Nouns			Verbs			Nouns			Verbs		
Eval. subset		shared	novel	hm	shared	novel	hm	shared	novel	hm	shared	novel	hm
ZS CLIP	-	10.38	13.58	11.77	12.32	4.32	6.40	11.38	10.49	10.92	2.73	9.84	4.27
CoOp	-	16.86	16.02	16.43	25.03	5.97	9.64	15.77	10.11	12.32	20.22	5.69	8.89
CoCoOp	-	16.35	11.51	13.51	24.34	0.00	0.00	17.31	11.46	13.79	15.32	6.46	9.09
CLIP-Adapter	-	12.46	5.99	8.09	21.48	3.09	5.40	8.72	7.42	8.02	19.60	3.00	5.20
CLIP-Adapter*	✓	16.24	12.22	13.95	25.29	1.23	2.35	14.67	7.68	10.08	24.17	4.50	7.59
A5	✓	15.25	5.24	7.80	27.90	3.09	5.56	13.54	5.71	8.03	24.29	0.55	1.07
Vita-CLIP	✓	15.84	6.15	8.86	27.22	4.11	7.14	14.60	9.76	11.69	16.46	6.58	9.40
X-MIC	✓	20.04	21.51	20.75	29.01	7.00	**11.27**	19.66	12.24	**15.09**	23.00	7.16	**10.92**
X-MIC+DINO	✓	25.56	20.52	**22.76**	31.92	6.38	10.63	18.91	10.67	13.65	20.55	6.48	9.85

Table 2. **Zero-shot action generalization**. Left: The models are trained on Ego4D (E4D) and subsequently evaluated on Epic-Kitchnes (EK) using disjoint subsets of classes (shared and novel). Right: The models are trained on Epic-Kitchens (EK) and then evaluated in a cross-dataset manner on subsets of classes within Ego4D.

4.3. X-MIC Comparison to SOTA

In Tables 1 and 2, we start with comparing our method to CLIP. Notably, on both the Ego4D and Epic-Kitchens datasets, CLIP yields surprisingly low results for both verbs and nouns, in contrast to its strong performance on third-person datasets [16, 33]. Next, we compare our method to other adaptation methods of VLMs which have shown improvements on image and video recognition benchmarks.

Image-based Adaptation Methods. First, we present a comparison to image-based adaptation models, including CoOp [44], CoCoOp [43], and CLIP-Adapter [8] which do not contain a temporal component. Our analysis in Table 1 shows that early fusion-based models like CoOp and Co-CoOp exhibit limited learning capacity, resulting in poorer performance compared to other models for both nouns and verbs when evaluated within-dataset, especially on Epic-Kitchens. This aligns with earlier findings shown in [4]. A late fusion-based framework, CLIP-Adapter, improves the within-dataset scores but demonstrates weaker generalization on nouns for cross dataset evaluation. However, Table 2 reveals that CoOp [44] demonstrates robustness when novel nouns and verbs are encountered even in the absence of any temporal attention module. We hypothesize that other models may be more prone to overfitting on the shared classes due to a larger number of parameters.

Video-based Adaptation Methods. In Table 1, we evaluate the performance of recent third-person video adaptation models, specifically A5 [15] and Vita-CLIP [1], in an egocentric scenario. Additionally, we enhance the CLIP-Adapter model by incorporating temporal attention and evaluate its effectiveness as a video model. We notice that the inclusion or exclusion of a temporal component, beyond simple averaging, has a relatively minor impact on noun recognition using CLIP-Adapter. To illustrate, when trained on the Epic-Kitchens dataset, CLIP-Adapter, with (denoted as CLIP-Adapter*) and without a temporal attention module, exhibits comparable performance in noun recognition within the dataset (EK), with scores of 33.21% and 34.40%, respectively. However, the role of temporal at-

	Nouns			Verbs		
	E4D	EGTEA	hm	E4D	EGTEA	hm
ZS CLIP	5.89	19.70	9.07	2.18	18.71	3.51
CoOp	29.23	23.90	26.29	22.57	26.45	24.35
Co-CoOp	29.85	27.90	28.84	21.31	27.74	24.10
CLIP-Adapter	30.00	21.41	24.98	22.82	26.51	24.52
CLIP-Adapter *	29.18	22.40	25.34	27.32	26.57	26.93
A5	33.50	23.70	27.76	26.31	28.03	27.14
Vita-CLIP	33.52	17.24	22.76	22.66	27.63	24.89
X-MIC	33.54	29.21	**31.21**	28.93	31.41	**30.12**

Table 3. **SOTA comparison on EGTEA.** The model is trained on Ego4D dataset and evaluated in a zero-shot manner on EGTEA.

	Nouns			Verbs		
	E4D	EK	hm	E4D	EK	hm
F	31.68	14.20	19.61	27.19	24.02	25.51
H	31.35	14.02	19.37	26.32	26.59	26.46
F+H	33.54	15.35	**21.06**	28.93	26.48	**27.65**

Table 4. **Influence of Ego-Spatial-Temporal attention.** F denotes full frames, H denotes hand crops. F+H correspond to our proposed attention module. All models share the same architecture of the temporal attention module.

tention becomes crucial in enhancing verb recognition performance, as evidenced by consistent improvements across both datasets and all models. A5 [15], which combines both early fusion and temporal attention, shows poor cross-dataset generalization on nouns for both datasets, aligning with the findings reported by its authors [15] in the context of cross-dataset third-person video generalization. The recent SOTA model on third-person video generalization, Vita-CLIP [1], demonstrates enhanced noun recognition on both datasets but exhibits lower verb recognition on Ego4D. In contrast to other video adaptation models, we decouple temporal attention from the frozen backbone and introduce X-MIC-vector, encapsulating all temporal information. Moreover, employing cross-modal adaptation, we introduce video-specific classifiers. For each video, we create an individual text-based classifier, which is adapted with our X-MIC-vector. Our approach demonstrates state-of-the-art generalization performance while maintaining high performance on within-dataset evaluation. Moreover, by leveraging DINO pre-trained model [23] as visual encoder V_{II}, we observe significant improvements on within-dataset evaluation. In Table 3, we present our evaluation on EGTEA. Overall, we note consistent trends across all methods.

In Table 2, we observe that video-based models perform poorly, likely due to overfitting on shared classes. Models like A5 [15] and Vita-CLIP [1], with a larger number of parameters, may be more susceptible to this issue. In contrast, our X-MIC framework decouples the adapter module from the frozen VL embedding space, enabling enhanced generalization. Furthermore, we observe that models struggle more with generalizing on verbs than nouns, likely due to the object-centric pre-training data of the backbone models, *e.g.* CLIP is pre-trained solely on image-text pairs.

4.4. Ablations

In this section, we evaluate the effectiveness of our design choices. For all ablations, we train models on Ego4D and evaluate on Ego4D and Epic-Kitchens. As backbone, we use CLIP ViT-B/16, unless otherwise specified.

Ego-Spatial-Temporal Attention. In Table 4, we demonstrate the impact of utilizing full frames, that usually includes scene context, and hand crops on the performance of egocentric videos. We observe that concentrating solely on hand regions enhances verb generalization, whereas the utilization of full images proves marginally more advantageous for noun generalization. When employing our proposed ego-spatial-temporal attention mechanism, we achieve a notable improvement in the harmonic mean. Specifically, there is a 1.45% increase for nouns and a 2.14% boost for verbs compared to using full frames. By guiding the model to consider context in relation to hand areas, our attention approach not only enhances performance within the dataset but also showcases improved cross-dataset performance.

Larger backbone. In Table 5, we assess the effectiveness of our method using a bigger CLIP model, specifically comparing the performance of CLIP ViT-L/14 with ViT-L/16. While we do not observe performance gains for within the dataset evaluations, a compelling trend emerges in cross-dataset generalization, particularly on nouns. Notably, employing the larger model ViT-L/14 results in a significant improvement of over 7% in noun and 2.45% in verb generalization on Epic. This encouraging outcome underscores the potential of vision transformers and suggests that further exploration and refinement of these models could yield even more substantial gains in cross-dataset generalization.

Egocentric VL backbone. Table 5 presents an evaluation on X-MIC-model performance using backbones CLIP and Lavila [41], which is pre-trained on text-video pairs from the Ego4D dataset in a contrastive manner. Note that the Lavila backbone initializes its model from CLIP pre-trained models. We first compare the zero-shot results from the original CLIP backbone and Lavila. Lavila demonstrates a significant improvement in noun recognition by 16.59% on the Ego4D dataset and noun generalization to Epic by 17.18%. While Lavila shows a decrease in verb recognition accuracy within the dataset by 2.38% , its generalization to Epic verbs increases by 6.04%. This outcome is surprising, as we initially expected Lavila to generalize better on verbs due to its training on an egocentric dataset, indicating a strong bias toward object-oriented pre-training strategies. We observe similar trends when our model utilizes CLIP

Evaluation dataset			Nouns			Verbs		
			E4D	EK	hm	E4D	EK	hm
CLIP	ViT-B/16	ZS	5.89	8.74	7.03	2.18	4.25	2.88
		X-MIC	33.54	15.35	21.06	28.93	26.48	27.65
	ViT-L/14	ZS	8.40	13.88	10.46	8.57	9.70	9.10
		X-MIC	33.75	22.46	26.97	28.13	28.93	28.52
Lavila	ViT-L/14	ZS	24.99	31.06	27.69	6.19	15.74	8.88
		X-MIC	35.18	34.97	35.08	12.28	24.66	16.37

Table 5. **Influence of different backbones.** We compare the performance of CLIP ViT-L/14 with ViT-B/16. Additionally, we provide a comparison of CLIP backbone, pre-trained on text-image pairs, to Lavila backbone, pre-trained on pairs of egocentric videos and narrations from full Ego4D. ZS denotes zero-shot evaluation. X-MIC denotes the evaluation of our method with the corresponding backbones (CLIP or Lavila) *without* additional DINO backbone.

norm	Nouns		
	E4D	EK	hm
n1	33.54	15.35	**21.06**
none	32.64	14.34	19.92
n2,n3	32.74	14.59	20.19
n1,n2,n3	31.99	14.49	19.95
n1,n2	15.81	12.3	13.83
n1,n3	12.12	11.34	11.71

Table 6. **Influence of feature normalization**. [n1] stands for $l2$-norm of features after V_{II} encoder and before the adapter (our default). [n2] denotes $l2$-norm of X-MIC vector before sum, [n3] denotes $l2$-norm of text features before sum.

	Nouns					Verbs				
	ZS		X-MIC			ZS		X-MIC		
prompts	E4D	EK	E4D	EK	hm	E4D	EK	E4D	EK	hm
<class>	5.89	**8.74**	33.54	15.35	21.06	2.18	4.25	28.93	26.48	**27.65**
Image of a <class>	**10.52**	6.75	32.31	14.81	20.31	3.28	5.40	28.56	25.98	27.21
Video of a <class>	10.32	6.80	32.62	14.77	20.33	2.93	5.97	28.14	22.70	25.13
Egocentric image a <class>	9.61	7.11	32.09	15.65	21.04	2.98	3.83	28.58	24.02	26.10
Image of a hand holding a <class>	10.09	6.32	32.92	14.28	19.92	**3.29**	**9.87**	27.53	19.33	22.71
Egocentric image of a hand holding <class>	9.23	6.86	33.29	15.83	**21.45**	2.41	6.24	27.66	16.94	21.01

Table 7. **Influence of prompting the frozen text model with additional context.** ZS denotes zero-shot CLIP evaluation. Noun recognition is robust to contextual variations, while verb recognition performs best without additional context.

versus Lavila as a backbone, where noun generalization increases significantly, while verb generalization slightly decreases.

Prompts for text encoder. In Table 7, we evaluate the performance of zero-shot CLIP and our model by prompting the frozen text model for classification with additional context. Our experiments include specific details like "Video of a" or indications of hands and an egocentric view. We find that zero-shot noun performance is the best with the standard "Image of a " context for Ego4D and without context for Epic-Kitchens. However, zero-shot verb recognition benefits from an additional context, achieving 3.29% and 9.87% on Ego4d and Epic-Kitchens, respectively. With our X-MIC adaptation, we observe that noun recognition remains robust to these changes, while verb recognition is sensitive and performs best when no additional context is provided highlighting the complexity of incorporating contextual information in egocentric scenarios.

Importance of normalization. We investigate the significance of feature normalization in the embedding space in Table 6. `n1` represents our default choice, involving the normalization of visual features after the V_{II} encoder and before the adapter. `n2` indicates the normalization of X-MIC-vector, i.e., visual features after our video-adapter module, prior to summation with frozen text features. Lastly, `n3` denotes the normalization of frozen text features before summation with X-MIC-vector. The 'none' corresponds to no normalization. `[n1]` demonstrates the optimal balance between regularization and no regularization. Configurations `[n1]`, `[n2,n3]`, `[n1,n2,n3]`, and `'none'` all yield symmetric feature magnitudes before the summation of frozen text features and X-MIC-vector and marginally change the harmonic mean. Conversely, variations such as `[n1,n2]` and `[n1,n3]` result in imbalances during the summation of different modalities, leading to suboptimal performance.

5. Conclusions & Limitations

We have introduced X-MIC, a simple yet effective cross-modal adaptation framework for VLMs, that injects egocentric video information into the frozen VL embedding, achieving significant improvements in fine-grained cross-dataset egocentric recognition of nouns and verbs. Moreover, X-MIC vectors offer decoupling of the domain-specific knowledge from the frozen VL embedding. This allows to explore different visual backbones for text conditioning directly in the embedding space, showing improved generalization. It is important to note that our method focuses solely on video classification and does not encompass text-vision tasks like text-to-video retrieval, which would necessitate using text-conditioned videos instead of our video-conditioned text representations. We plan to explore this direction in future work.

References

[1] Vita-clip: Video and text adaptive clip via multimodal prompting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23034–23044, 2023. 2, 6, 7

[2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 2

[3] Dibyadip Chatterjee, Fadime Sener, Shugao Ma, and Angela Yao. Opening the vocabulary of egocentric actions. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[4] Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. Adaptformer: Adapting vision transformers for scalable visual recognition. *Advances in Neural Information Processing Systems*, 35:16664–16678, 2022. 2, 6

[5] Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Qiao. Vision transformer adapter for dense predictions. *arXiv preprint arXiv:2205.08534*, 2022. 2

[6] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Jian Ma, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Rescaling egocentric vision: Collection, pipeline and challenges for epic-kitchens-100. *International Journal of Computer Vision (IJCV)*, 130:33–55, 2022. 1, 2, 5

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009. 1

[8] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision*, pages 1–15, 2023. 2, 4, 6

[9] Tianyu Gao, Adam Fisch, and Danqi Chen. Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*, 2020. 2

[10] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Martin, Tushar Nagarajan, Ilija Radosavovic, Santhosh Kumar Ramakrishnan, Fiona Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Zhongcong Xu, Chen Zhao, Siddhant Bansal, Dhruv Batra, Vincent Cartillier, Sean Crane, Tien Do, Morrie Doulaty, Akshay Erapalli, Christoph Feichtenhofer, Adriano Fragomeni, Qichen Fu, Christian Fuegen, Abrham Gebreselasie, Cristina Gonzalez, James Hillis, Xuhua Huang, Yifei Huang, Wenqi Jia, Weslie Khoo, Jachym Kolar, Satwik Kottur, Anurag Kumar, Federico Landini, Chao Li, Yanghao Li, Zhenqiang Li, Karttikeya Mangalam, Raghava Modhugu, Jonathan Munro, Tullie Murrell, Takumi Nishiyasu, Will Price, Paola Ruiz Puentes, Merey Ramazanova, Leda Sari, Kiran Somasundaram, Audrey Southerland, Yusuke Sugano, Ruijie Tao, Minh Vo, Yuchen Wang, Xindi Wu, Takuma Yagi, Yunyi Zhu, Pablo Arbelaez, David Crandall, Dima Damen, Giovanni Maria Farinella, Bernard Ghanem, Vamsi Krishna Ithapu, C. V. Jawahar, Hanbyul Joo, Kris Kitani, Haizhou Li, Richard Newcombe, Aude Oliva, Hyun Soo Park, James M. Rehg, Yoichi Sato, Jianbo Shi, Mike Zheng Shou, Antonio Torralba, Lorenzo Torresani, Mingfei Yan, and Jitendra Malik. Ego4d: Around the World in 3,000 Hours of Egocentric Video. In *IEEE/CVF Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2, 5

[11] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 2

[12] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR, 2019. 2

[13] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021. 1

[14] Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8: 423–438, 2020. 2

[15] Chen Ju, Tengda Han, Kunhao Zheng, Ya Zhang, and Weidi Xie. Prompting visual-language models for efficient video understanding. In *European Conference on Computer Vision*, pages 105–124. Springer, 2022. 2, 5, 6, 7

[16] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 1, 6

[17] Evangelos Kazakos, Arsha Nagrani, Andrew Zisserman, and Dima Damen. Epic-fusion: Audio-visual temporal binding for egocentric action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5492–5501, 2019. 2

[18] Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19113–19122, 2023. 2

[19] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021. 2

[20] Yin Li, Miao Liu, and James M Rehg. In the eye of beholder: Joint learning of gaze and actions in first person video. In *Proceedings of the European conference on computer vision (ECCV)*, 2018. 2, 5

[21] Kevin Qinghong Lin, Jinpeng Wang, Mattia Soldan, Michael Wray, Rui Yan, Eric Z XU, Difei Gao, Rong-Cheng Tu, Wenzhe Zhao, Weijie Kong, et al. Egocentric video-language pretraining. *Advances in Neural Information Processing Systems*, 35:7575–7586, 2022. 2

[22] Ziyi Lin, Shijie Geng, Renrui Zhang, Peng Gao, Gerard de Melo, Xiaogang Wang, Jifeng Dai, Yu Qiao, and Hongsheng Li. Frozen clip models are efficient video learners. In *European Conference on Computer Vision*, pages 388–404. Springer, 2022. 2

[23] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 4, 6, 7

[24] Namuk Park, Wonjae Kim, Byeongho Heo, Taekyung Kim, and Sangdoo Yun. What do self-supervised vision transformers learn? *arXiv preprint arXiv:2305.00729*, 2023. 4

[25] Mandela Patrick, Dylan Campbell, Yuki M. Asano, Ishan Misra Florian Metze, Christoph Feichtenhofer, Andrea Vedaldi, and Joao F. Henriques. Keeping your eye on the ball: Trajectory attention in video transformers. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 2

[26] Shraman Pramanick, Yale Song, Sayan Nag, Kevin Qinghong Lin, Hardik Shah, Mike Zheng Shou, Rama Chellappa, and Pengchuan Zhang. Egovlpv2: Egocentric video-language pre-training with fusion in the backbone. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5285–5297, 2023. 2

[27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1

[28] Hanoona Rasheed, Muhammad Uzair Khattak, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Fine-tuned clip models are efficient video learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6545–6554, 2023. 2

[29] Fadime Sener, Dibyadip Chatterjee, and Angela Yao. Technical report: Temporal aggregate representations. *arXiv preprint arXiv:2106.03152*, 2021. 2

[30] Fadime Sener, Dibyadip Chatterjee, Daniel Shelepov, Kun He, Dipika Singhania, Robert Wang, and Angela Yao. Assembly101: A large-scale multi-view video dataset for understanding procedural activities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21096–21106, 2022. 2

[31] Dandan Shan, Jiaqi Geng, Michelle Shu, and David F Fouhey. Understanding human hands in contact at internet scale. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9869–9878, 2020. 5

[32] Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. *arXiv preprint arXiv:2010.15980*, 2020. 2

[33] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 6

[34] Asa Cooper Stickland and Iain Murray. Bert and pals: Projected attention layers for efficient adaptation in multi-task learning. In *International Conference on Machine Learning*, pages 5986–5995. PMLR, 2019. 2

[35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 2

[36] Chao-Yuan Wu, Yanghao Li, Karttikeya Mangalam, Haoqi Fan, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Memvit: Memory-augmented multiscale vision transformer for efficient long-term video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13587–13597, 2022. 2

[37] Fanyi Xiao, Yong Jae Lee, Kristen Grauman, Jitendra Malik, and Christoph Feichtenhofer. Audiovisual slowfast networks for video recognition. *arXiv preprint arXiv:2001.08740*, 2020.

[38] Xuehan Xiong, Anurag Arnab, Arsha Nagrani, and Cordelia Schmid. M&m mix: A multimodal multiview transformer ensemble. *arXiv preprint arXiv:2206.09852*, 2022. 2

[39] Yuhao Zhang, Hang Jiang, Yasuhide Miura, Christopher D Manning, and Curtis P Langlotz. Contrastive learning of medical visual representations from paired images and text. In *Machine Learning for Healthcare Conference*, pages 2–25. PMLR, 2022. 1

[40] Yue Zhao and Philipp Krähenbühl. Real-time online video detection with temporal smoothing transformers. In *European Conference on Computer Vision*, pages 485–502. Springer, 2022. 2

[41] Yue Zhao, Ishan Misra, Philipp Krähenbühl, and Rohit Girdhar. Learning video representations from large language models. In *CVPR*, 2023. 2, 7

[42] Zexuan Zhong, Dan Friedman, and Danqi Chen. Factual probing is [mask]: Learning vs. learning to recall. *arXiv preprint arXiv:2104.05240*, 2021. 2

[43] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16816–16825, 2022. 2, 4, 6

[44] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022. 2, 3, 6

[45] Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. Cross-task weakly supervised learning from instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3537–3545, 2019. 2

X-MIC: Cross-Modal Instance Conditioning for Egocentric Action Generalization

Supplementary Material

In this supplementary, we provide additional implementation details in Sec. 6, extend the discussion on generalization performance to shared and novel action classes in Sec. 7, explore the synergies between various adaptation techniques and X-MIC framework in Sec. 8, and present additional ablation experiments of our X-MIC framework in Sec. 9.

6. Implementation Details

Transformer Block. In our Ego-spatio-temporal attention module, we utilize a sequence of transformer blocks b_S and b_T to capture spatial and temporal dependencies, respectively. The general structure of the transformer block given input tensors x of dimensionality D is depicted in Alg. 1. LN denotes LayerNorm [2], MHA denotes multi-head attention with 8 heads [35], and MLP denotes 2-layer multi-layer perception with the bottleneck $D/4$ and QuickGELU [11] as an activation function.

Algorithm 1 Transformer Block

Require: $x \in \mathcal{R}^D$
$x \leftarrow x + MHA(LN(x))$
$x \leftarrow x + MLP(LN(x))$

We note that b_S consist of one transformer block and b_T includes two sequential transformer blocks.

Data. For augmentations, we exclusively employ frame flipping during training. Furthermore, for the CLIP backbone, we resize frames so that the shortest side is 224, followed by a center crop of 224x224 and normalization. For the Lavila backbone, frames are directly resized to 224x224.

For Epic-Kitchens, we use the provided annotation boundaries to define action clips. Conversely, in the Ego4D FHO challenge, the boundaries for each clip are initially set as 8 seconds. However, upon observation, we note that actions typically span a duration shorter than 8 seconds, prompting us to uniformly shorten all clips to 4 seconds.

7. Zero-Shot Generalization Discussion

In Table 2, we presented comprehensive cross-dataset results showcasing the generalization performance on both the Epic-Kitchens and Ego4D datasets. Notably, the Epic-Kitchens dataset is exclusive to kitchen-related scenes and actions, while the Ego4D dataset encompasses a diverse range of daily activities. We distinguish between subsets of shared and novel classes and provide additional details in the following paragraph.

Shared-Novel Classes. We categorize classes as "shared" when there is an exact match in their names across datasets. For noun classes in Ego4D and Epic Kitchens, there exist 163 such shared classes, including examples like "apple", "toaster" or "washing machine". Consequently, the set of "novel" noun classes for Ego4D comprises 358 classes, encompassing items such as "transistor", "ambulance" and "stroller". In contrast, the "novel" noun classes for Epic-Kitchens total 137 and predominantly represent more detailed kitchen-related categories such as "mint", "onion ring", "scale". In the domain of verb classes for both Ego4D and Epic Kitchens, we identify 51 *shared* classes, including actions such as "hold", "hang" or "attach". This results in 66 *novel* verb classes for Ego4D with examples like "park", "repair" and "wave". On the other hand, *novel* verb classes for Epic-Kitchens amount to 46, primarily encompassing more detailed kitchen-related actions like "slide", "stab", "unfreeze". For a comprehensive list of classes, refer to the detailed class separation in Sec. 10.

8. Complementary of X-MIC

In Table 12, we illustrate the compatibility of our framework with other adaptation methods. Early fusion methods, where the uni-modal (U) method corresponds to CoOp [44] and the cross-modal (X) method corresponds to CoCoOp [43], demonstrate enhanced performance in both within- and cross-dataset evaluations. However, the integration of our framework with late fusion uni-modal adapters (Tt and Vv) does not further enhance the generalization while maintaining the overall high performance.

9. Ablations of X-MIC

In this section, we conduct additional ablation experiments to validate the efficacy of our design choices.

Temporal Attention. In Table 14, we present an ablation where we keep our ego-spatial attention but replace the temporal module with a simple average over frames. The results demonstrate significant improvements in verb recognition on Ego4D by 6.28%, and in verb generalization to Epic by 6.29% when employing temporal modeling. This aligns with expectations, as the temporal component encodes movements. We also observe improvements in noun recognition and generalization, indicating that recognizing

	Nouns			Verbs		
α	E4D	EK	hm	E4D	EK	hm
0.1	31.23	14.46	19.77	22.85	24.91	23.83
0.5	32.43	14.40	19.94	26.81	23.80	25.21
1.0	33.54	15.35	**21.06**	28.93	26.48	**27.65**
2.0	33.20	14.73	20.40	28.14	26.39	27.24
5.0	32.29	14.24	19.77	27.50	27.00	27.25

Table 8. **Influence of scale of X-MIC vector.** Trained on Ego4D.

	Nouns			Verbs		
	EK	E4D	hm	EK	E4D	hm
F	24.89	12.48	16.62	42.10	18.86	26.05
H	31.13	11.58	16.88	44.20	18.49	26.07
F+H	30.64	12.32	**17.57**	50.01	18.10	**26.58**

Table 9. **Influence of Ego-Spatial-Temporal attention.** F denotes full frames, H denotes hand crops. F+H correspond to our proposed attention module. All models share the same architecture of the temporal attention module. Trained on Epic-Kitchens.

			Nouns			Verbs		
Evaluation dataset			EK	E4D	hm	EK	E4D	hm
CLIP	ViT-L/16	Zero-Shot	8.74	5.89	7.03	4.25	2.18	2.88
		X-MIC	30.64	12.32	17.57	50.01	18.10	26.58
	ViT-L/14	Zero-Shot	13.88	8.40	10.46	9.70	8.57	9.10
		X-MIC	39.02	14.24	20.86	48.12	18.83	27.07
Lavila	ViT-L/14	Zero-Shot	31.06	24.99	27.69	15.74	6.19	8.88
		X-MIC	41.78	29.62	34.67	46.14	9.35	15.54

Table 10. **Influence of different backbones.** We compare the performance of CLIP ViT-L/14 with ViT-L/16. Additionally, we provide a comparison of CLIP backbone, pretrainied on text-image pairs, to Lavila backbone, pretrained on pairs of egocentric videos and narrations from full Ego4D. Trained on Epic-Kitchens (EK).

norm	Trained on Ego4D (E4D)						Trained on Epic-Kitchens (EK)					
	Nouns			Verbs			Nouns			Verbs		
	E4D	EK	hm	E4D	EK	hm	EK	E4D	hm	EK	E4D	hm
n1	33.54	15.35	**21.06**	28.93	26.48	**27.65**	30.64	12.32	**17.57**	50.01	18.10	26.58
none	32.64	14.34	19.92	27.79	25.76	26.74	32.54	10.34	15.69	43.25	19.53	**26.91**
n2,n3	32.74	14.59	20.19	27.55	24.49	25.93	34.08	10.48	16.03	49.32	16.49	24.71
n1,n2,n3	31.99	14.49	19.95	24.30	22.69	23.47	32.46	10.21	15.54	49.02	18.05	26.39
n1,n2	15.81	12.3	13.83	24.3	22.69	23.47	19.88	8.14	11.55	19.72	8.41	11.79
n1,n3	12.12	11.34	11.71	22.88	18.49	20.46	16.16	8.72	11.33	23.68	17.68	20.24

Table 11. **Influence of feature normalization**. Extended Table 6(main). [n1] corresponds to the normalization of visual features after the V_{II} encoder and before the adapter and demonstrates an optimal balance between normalization and no normalization. [n2] corresponds to the normalization of the X-MIC vector before summation with text representation. [n3] corresponds to the normalization of text representation before summation with the X-MIC vector.

	U/X	Nouns			Verbs		
	-Modal	E4D	EK	hm	E4D	EK	hm
Early Fusion	U	28.22	10.87	15.70	22.57	20.42	21.44
+X-MIC	U+X	29.83	11.94	17.05	24.99	22.72	23.80
Early Fusion	X	30.00	9.51	14.44	21.31	12.99	16.14
+X-MIC	X+X	30.33	10.34	15.42	25.53	25.06	25.29
X-MIC	X	33.54	15.35	21.06	28.93	26.48	27.65
+ Tt	X+U	33.66	14.82	20.58	28.41	26.69	27.52
+ Vv	X+U	32.75	15.20	20.77	28.36	25.85	27.05
+ Tt + Vv	X+U	33.23	15.13	20.79	28.20	26.29	27.21

Table 12. **X-MIC framework with other adaptation methods.** U denotes uni-modal methods, X denotes cross-modal methods. Tt denotes text uni-modal adapter for late fusion, Vv similarly denotes video uni-modal adapter for late fusion. Trained on Ego4D (E4D).

# frames	Nouns			Verbs		
	E4D	EK	hm	E4D	EK	hm
32	33.02	14.77	20.41	28.56	25.34	26.85
16	33.54	15.35	**21.06**	28.93	26.48	**27.65**
8	32.06	14.82	20.27	27.30	26.90	27.10
4	31.74	14.99	20.36	26.41	26.17	26.29
2	29.95	12.77	17.91	23.85	25.73	24.75

Table 13. **Influence of number of frames**. Trained on Ego4D.

	Nouns			Verbs		
	E4D	EK	hm	E4D	EK	hm
w/o	31.41	13.31	18.69	22.65	20.19	21.34
w/	33.54	15.35	**21.06**	28.93	26.48	**27.65**

Table 14. **Influence of temporal attention.** Replacing the temporal module with a simple average decreases verb and noun recognition. The models share the same architecture for Ego-Spatial attention module.

nouns in an egocentric dataset requires more than just appearance; motion encoding is also beneficial *e.g.* for recognizing "slicing an apple" action.

Number of frames. In Table 13, we showcase the influence of the number of frames sampled during both training and evaluation. Notably, the performance peaks with 16 and 8 sampled frames. Conversely, sampling only two frames per clip significantly diminishes performance across all classes.

Scale of X-MIC-vector. In Eq. 1 in our main paper, we extend our analysis to validate the importance of the scale of the X-MIC-vector. To be specific, according to Eq. 1 in our main paper is the following:

$$c = \text{argmax}_t < \overline{e_t + \alpha A(\overline{V_{II}(x_v)})}, \bar{e}_v >, \qquad (4)$$

where α is a scale factor. In Table 8, we vary the scale factor α from 0.1 to 5 and observe that higher values result in improved performance, particularly in the evaluation of verbs both within and across datasets.

Ego-Spatial-Temporal Attention. In Table 9, we provide supplementary results to highlight the impact of our ego-spatial-temporal attention module. We note consistent performance across models trained on Epic-Kitchens and Ego4D, see Table 3 in our main paper.

Different backbones. Table 10 presents a comparison between CLIP ViT-B/16 and ViT-L/14 models when trained on Epic-Kitchens. Furthermore, we evaluate the performance of image-text pre-training (CLIP model) and video egocentric pre-training (Lavila). Our findings f those outlined in Table 5 in our main paper.

Importance of normalization. Table 11 offers supplementary results to complement those in Table 6 our main paper. The outcomes closely align with the findings presented our main paper.

10. List of Classes

Shared Nouns:

spoon; plate; knife; pan; lid; bowl; drawer; sponge; glass; hand; fridge; cup; fork; bottle; onion; cloth; chopping board; bag; spatula; container; dough; water; meat; pot; potato; oil; cheese; bread; food; tray; pepper; colander; carrot; tomato; kettle; pasta; oven; sauce; paper; garlic; towel; egg; rice; mushroom; chicken; coffee; glove; leaf; sink; milk; jug; salad; dishwasher; cucumber; peach; flour; courgette; filter; butter; scissors; chopstick; blender; mat; spice; sausage; napkin; microwave; pizza; button; stock; grater; ladle; yoghurt; cereal; broccoli; brush; lemon; juicer; light; squash; leek; fish; lettuce; seed; foil; washing machine; corn; soup; clip; lighter; ginger; tea; nut; vinegar; rolling pin; pie; burger; book; tongs; cream; banana; paste; plug; teapot; floor; lime; bacon; sandwich; phone; thermometer; orange; basket; tablet; cake; avocado; chair; pancake; toaster; apple; chocolate; ice; handle; pea; yeast; coconut; spinach; apron; grape; kale; wire; asparagus; mango; kiwi; bean; whisk; remote control; label; celery; cabbage; ladder; battery; pear; funnel; wall; strawberry; shelf; straw; cork; window; bar; heater; watch; melon; popcorn; candle; balloon; computer; key; pillow; pen; plum; tape; camera;

Novel Nouns Ego4D:

arm; artwork; awl; axe; baby; baking soda; ball; ball bearing; baseboard; bat; bat; bathtub; batter; bead; beaker; bed; belt; bench; berry; beverage; bicycle; blanket; block; blower; bolt extractor; bookcase; bracelet; brake; brake pad; branch; brick; broom; bubble gum; bucket; buckle; butterfly; cabinet; calculator; caliper; can opener; canvas; car; card; cardboard; carpet; cart; cat; ceiling; cello; cement; chaff; chain; chalk; chip; chip; chip; chisel; cigarette; circuit; clamp; clay; clock; coaster; coffee machine; comb; cooker; cookie; corner; countertop; crab; cracker; crayon; crochet; crowbar; curtain; cushion; cutter; decoration; derailleur; detergent; dice; dog; door; doorbell; dough mixer; doughnut; dress; drill; drill bit; drum; dumbbell; dust; duster; dustpan; eggplant; engine; envelope; eraser; facemask; fan; faucet; fence; file; filler; fishing rod; flash drive; flower; foam; foot; fries; fuel; game controller; garbage can; gasket; gate; gauge; gauze; gear; generator; glasses; glue; glue gun; golf club; gourd; grain; grapefruit; grass; grill; grinder; guava; guitar; hair; hammer; hanger; hat; hay; haystack; head; headphones; helmet; hinge; hole; horse; hose; house; ice cream; ink; iron; jack;

jacket; ketchup; keyboard; leash; leg; lever; lock; lubricant; magnet; manure; mask; matchstick; medicine; metal; microscope; mirror; mixer; mold; money; mop; motorcycle; mouse; mouthmower; multimeter; nail cutter; nail gun; nail polish; necklace; needle; net; nozzle; nut; okra; paddle; paint; paint roller; paintbrush; palette; panel; pantspapaya; pastry; peanut; pedal; peel; peeler; peg; pencil; photo; piano; pickle; picture; pilot jet; pin; pipe; planer; plant; playing cards; plier; pole; pot; pump; pumpkin; purse; puzzle or game piece; rack; radio; rail; rake; razor blade; ring; rod; root; rope; router; rubber band; ruler; sand; sander; sandpaper; saw; scarf; scoopscraper; screw; screwdriver; sculpture; seasoning; set square; sewing machine; sharpener; shears; sheet; shell; shirt; shoe; shovel; shower head; sickle; sieve; sketch pad; skirt; slab; snorkel; soap; sock; socket; sofa; soil; solder iron; spacer; speaker; sphygmomanometer; spirit level; spray; spring; squeezer; stairs; stamp; stapler; steamer; steering wheel; stick; sticker; stone; stool; stove; strap; string; stroller; switch; syringe; table; taco; tape measure; television; tent; test tube; tie; tile; timer; toilet; toilet paper; toolbox; toothbrush; toothpick; torch; toy; tractor; trash; treadmill; tree; trimmer; trowel; truck; tweezer; umbrella; undergarment; vacuum; vacuum cleaner; valve; vase; video game; violin; wallet; wallpaper; watermelon; weighing scale; welding torch; wheat; wheel; wheelbarrow; windshield; wiper; wood; worm; wrapper; wrench; yam; zipper; zucchini; ambulance; back; bamboo; bandage; baton; bird; brownie; cash register; cassava; cocoa; cow; cupcake; drone; earplug; hotdog; marble; person; pipette; plunger; printer; putty; racket; ratchet; road; scaffold; stereo; transistor;

Novel Nouns Epic-Kitchens:

tap; cupboard; washing liquid; box; hob; package; bin; salt; jar; top; skin; coffee maker; rubbish; cutlery; can; heat; aubergine; chilli; mixture; clothes; tofu; olive; potato peeler; cover; kitchen towel; vegetable; plastic wrap; sugar; biscuit; wrap; scale; rest; drying rack; alarm; salmon; freezer; spreads; cap; curry; oatmeal; spring onion; holder; powder; egg shell; pork; oregano; food processor; recipe; liquid; pak choi; slow cooker; utensil; noodle; salami; kitchen; tuna; omelette; parsley; salad spinner; presser; coriander; bottle opener; lentil; blueberry; extractor fan; salt cellar; hummus; juice; green bean; knob; wine; pith; fishcakes; raisin; basil; paprika; caper; drink; stalk; turmeric; whetstone; thyme; lady finger; beef; blackberry; slicer; hoover; breadstick; roll; cocktail; crisp; beer; dust pan; washing powder; backpack; cumin; pizza cutter; air; quorn; almond; tv; egg scotch; stand; vide sous machine; masher; hand guard; shrimp; fruit; artichoke; cherry; sprout; sushi mat; crab stick; onion ring; pestle; gin; mint; lemon grass; rubber; gherkin; breadcrumb; cinnamon; dumpling; rosemary; power; syrup; pineapple; sheets; soda; raspberry; airer; turkey; face; whiskey; kitchen door; cd; vanilla extract;

Shared Verbs:

take; put; wash; open; close; insert; turn on; cut; turn off; pour; mix; move; remove; throw; shake; scoop; adjust; squeeze; peel; press; turn; scrape; fill;apply; fold; break; pull; lift; hold; unroll; hang; sprinkle; spray; roll; search; stretch; knead; divide; sharpen; water; attach; wear; measure; unscrew; grate; screw; serve; uncover; lock; carry; mark;

Novel Verbs Ego4D:

arrange; blow; catch; clap; clean; climb; consume; count; cover; crochet; detach; dig; dip; draw; drill; drive; enter; feed; file; fry; give; grind; hit; inspect; iron; kick; knit; loosen; mold; operate; pack; paint; park; pet; plant; play; point; pump; push; read; repair; sand; scroll; sew; shuffle; sieve; sit; smooth; stand; step; stick; swing; talk; tie; tighten; tilt; touch; unfold; untie; walk; weld; wipe; write; zip; watch; wave;

Novel Verbs Epic-Kitchens:

dry; empty; flip; check; scrub; pat; eat; wrap; filter; look; sort; rip; cook; add; crush; set; feel; rub; soak; brush; drop; drink; slide; gather; turn down; coat; transition; increase; wait; lower; form; smell; use; let go; finish; stab; unwrap; choose; flatten; switch; season; unlock; prepare; bake; bend; unfreeze;