InsAT: Instance-aware Semantic Alignment and Transfer from Human–Object Keypoints for Zero-to-Few-shot Action Understanding

Kazuki Tsutsukawa, Konicaminolta, Inc.
kazuki.tsutsukawa@konicaminolta.com

Abstract

Keypoint-based action recognition offers robustness to appearance variations and provides privacy-preserving representation. However, existing zero-shot (ZS) approaches largely emphasize human motion while underutilizing contextual information, particularly human–object interactions. Moreover, extending keypoint-based ZS models to few-shot scenarios remains insufficiently explored. We propose **Ins**tance-aware Semantic **A**lignment and **T**ransfer (**InsAT**), a unified framework for ZS recognition and zero-to-few-shot (Z2F) adaptation that leverages instance-level language descriptions. InsAT aligns textual descriptions of humans, objects, and their interactions with visual representations derived from human and object keypoints, enabling effective transfer of interaction knowledge from seen to unseen action classes. To support Z2F adaptation, we introduce Instance-level Visual Adaptation, a parameter-free mechanism that improves recognition by incorporating instance-level contextual cues without updating model weights. Extensive experiments demonstrate that InsAT substantially outperforms prior keypoint-based ZS methods and achieves competitive performance relative to large vision–language models, while remaining data-efficient and robust.

1 Introduction

Human action recognition is a fundamental problem in video understanding, with applications spanning surveillance (Cheng et al., 2021), sports analytics (Shao et al., 2020), and human–robot interaction (Lee et al., 2019). Recent advances in vision–language models (VLMs) have led to substantial progress in red–green–blue (RGB)-based action recognition, particularly in zero-shot learning (ZSL), where unseen action categories are recognized by exploiting pretrained visual–textual representations. In practice, action recognition rarely operates under a strictly zero-shot (ZS) assumption; instead, limited labeled samples of novel actions are often available. Motivated by this scenario, recent studies have investigated zero-to-few-shot (Z2F) adaptation on top of RGB-based ZSL models, aiming to refine decision boundaries for novel classes while preserving the generalization capabilities of pretrained encoders (Rasheed et al., 2023).

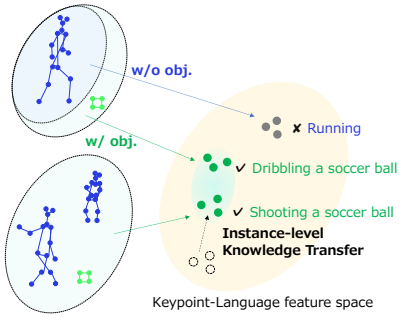

Figure 1: Human and object keypoints are aligned with instance-level descriptions for object-aware recognition and semantic knowledge transfer to unseen actions.

In contrast, keypoint-based action recognition relies on human skeletal representations, offering privacy-preserving and computationally efficient alternatives to RGB-based methods while remaining robust to appearance variations (Yan et al., 2018; Hachiuma et al., 2023). Despite these advantages, most keypoint-based approaches are developed under fully supervised settings, and research addressing ZS scenarios remains limited. Existing keypoint-based ZSL methods predominantly model human motion and largely neglect contextual cues, such as interacting objects or other agents, which are often essential for disambiguating actions.

The motivation for incorporating such contextual information is illustrated in Fig. 1. For example, recognizing *dribbling a soccer ball* requires capturing not only body movements but also their interactions with the ball. Human–object interactions (HOIs) exhibit local similarities across actions; for instance, *dribbling* and *shooting a soccer ball* both involve foot–ball contact on the ground. Lever-

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 36487–36504
July 2-7, 2026 ©2026 Association for Computational Linguistics

aging such shared interaction patterns is crucial for transferring knowledge from seen to unseen actions, as well as for adapting models under few-shot (FS) supervision. While Z2F extensions have been actively explored in RGB-based frameworks, they remain largely unexplored in keypoint-based methods. Existing methods lack explicit mechanisms for modeling transferable HOIs at the instance level, rendering Z2F adaptation particularly challenging.

To address these challenges, we propose, Instance-aware Semantic Alignment and Transfer (InsAT), a unified framework for keypoint-based ZS recognition and Z2F adaptation. InsAT leverages instance-level action descriptions that explicitly characterize humans, relevant objects, and their interaction. By aligning visual representations derived from human and object keypoints with instance-level textual descriptions of HOIs, InsAT enables effective semantic transfer from seen to unseen actions. Furthermore, we focus on Z2F adaptation, rather than episodic FS learning, and introduce Instance-level Visual Adaptation (IVA), which exploits instance-level contextual cues without updating model parameters, ensuring stable performance under extremely limited labeled data. To overcome the category constraints of conventional object detectors, we further integrate an open-vocabulary object detector (OVD), allowing flexible extension to action-relevant objects without retraining.

Our contributions are summarized as follows:
1. Instance-level semantics for keypoint-based ZSL: We propose InsAT, a framework that integrates human and object keypoints and aligns them with instance-level language descriptions for object-aware ZS action recognition.
2. Parameter-free Z2F adaptation: We introduce IVA, enabling effective Z2F adaptation without updating model parameters.
3. OVD-based extensibility: Our framework supports flexible incorporation of scene-specific object categories in both ZS and Z2F settings.
4. Strong empirical performance: InsAT outperforms prior keypoint-based ZSL methods and achieves competitive results relative to RGB-based VLMs across multiple benchmarks.

2 Related Work

2.1 Keypoint-based zero-shot recognition

Keypoint-based action recognition has been extensively studied in supervised settings, where actions are recognized from human skeletal sequences using a variety of spatiotemporal modeling techniques (Yan et al., 2018; Song et al., 2023; Shi et al., 2020). Beyond human motions, several studies have explored incorporating object keypoints to model HOIs, with Hachiuma et al. (2023) demonstrating notable performance gains on standard action recognition benchmarks.

In contrast, ZSL through keypoints sequences has received relatively limited attention compared to RGB-based ZSL approaches (Xu et al., 2021). Most existing methods project visual and textual representations into a shared embedding space, where semantically corresponding actions and descriptions are encouraged to align through cross-modal learning. Early work focused on global visual-semantic alignment (Ali et al., 2019), while subsequent studies incorporated local correspondence modeling to capture finer-grained motion semantics (Gupta et al., 2021). PURLS (Zhu et al., 2024) advanced this direction by aligning spatial–temporal local motion patterns with language representations. More recently, SCoPLe (Zhu et al., 2025a) adopts cross-modal prompt learning to adapt frozen pretrained encoders, and Neuron (Chen et al., 2025) introduces dynamically evolving action prototypes guided by large language model (LLM)–generated semantics for fine-grained ZS recognition.

Despite these advances, existing approaches primarily emphasize human motion representations and lack explicit modeling of object-centric or interaction-level semantics. InsAT addresses this limitation by jointly leveraging human and object keypoints and performing instance-level semantic alignment, enabling more effective transfer of interaction knowledge across action categories.

2.2 Zero-to-few-shot adaptation

FS learning for keypoint-based action recognition has primarily been studied under supervised episodic settings, where models are explicitly trained to recognize novel classes from limited support examples (Ma et al., 2022; Liu et al., 2023). These approaches typically rely solely on visual supervision and are developed independently of ZSL paradigms, limiting their ability to generalize to unseen actions without task-specific training.

In parallel, RGB-based research has made substantial progress in Z2F adaptation by reusing pretrained VLMs and updating either all model parameters, selected subsets, or lightweight components

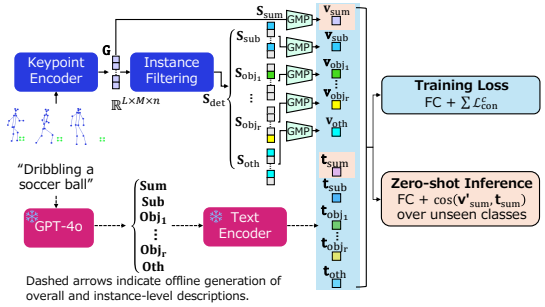

Figure 2: Overview of the training and inference pipeline of InsAT under the ZS setting.

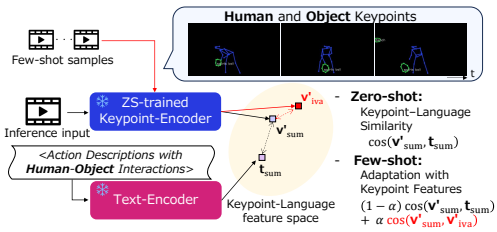

Figure 3: ZS inference and Z2F adaptation with InsAT. The adaptation is achieved via IVA, which integrates visual prototypes with text-based similarity in the keypoint–language embedding space.

such as continuous prompts or adapters (Rasheed et al., 2023; Ahmad et al., 2025). These methods enable efficient adaptation to novel classes while retaining the semantic knowledge encoded in pre-trained models. In contrast, for keypoint-based action understanding, Z2F adaptation remains largely unexplored.

Very recently, Skeleton-Cache (Zhu et al., 2025b) introduced a training-free test-time adaptation approach that enhances ZS skeleton-based action recognition through non-parametric feature caching. While effective, this method focuses on test-time adaptation through non-parametric feature caching and does not model transferable instance-level semantics. In contrast, InsAT supports Z2F adaptation by exploiting a small number of labeled samples at inference time, transferring instance-level semantic knowledge derived from human and object keypoints without updating model parameters. This design enables seamless extension from ZS recognition to Z2F adaptation within a unified framework.

3 Proposed Method

InsAT is a unified framework that is trained under a ZS setting and naturally extends to Z2F adaptation at inference time. Fig. 2 illustrates the ZS training and inference pipeline of InsAT. Given an input video, a keypoint encoder converts extracted human and object keypoints into instance-level visual features (Sec. 3.2). During training, the corresponding action label is provided to an LLM, which generates an action summary along with instance-specific descriptions capturing humans, objects, and their interactions (Sec. 3.3). For each description, the corresponding instance features are retrieved through *instance filtering* for each description (Sec. 3.4), and their textual embeddings, produced by the *text encoder*, are used as supervision signals. Instance-level contrastive learning (Sec. 3.5) aligns the retrieved visual features with their associated textual descriptions, thereby injecting instance-specific and HOI knowledge into the model.

During inference, instance-level visual features are aggregated into a video-level representation. In the ZS setting, this representation is compared against the textual embeddings of unseen action classes to predict the action label. Beyond ZS recognition, InsAT supports Z2F adaptation without retraining through IVA (Sec. 3.6). As illustrated in Fig. 3, IVA leverages a small number of labeled target-domain examples to construct instance-level visual prototypes, which are incorporated into the similarity computation at inference time. This design enables effective Z2F transfer while keeping both the visual encoder and text encoder fully frozen.

3.1 Keypoint detector

InsAT takes as input 2D/3D keypoints of humans and objects detected from visual data. For human 2D keypoints, we employ OpenPose (Cao et al., 2017) or HRNet (Sun et al., 2019), depending on the dataset. For object 2D keypoints, we use standard object detectors trained on the COCO dataset (Lin et al., 2014), namely PPNv2 or D-FINE (Peng et al., 2024). PPNv2 detects 90 categories, while D-FINE detects 80. To extend object coverage beyond COCO categories, we utilize Grounding DINO (Liu et al., 2025b) (Sec. 4.5). Object keypoints are defined as follows: for PPNv2, we use the eight vertices of the convex hull predicted by the model; for other object detectors, we extract eight points from each bounding box, consisting of the four corners and the midpoints of each edge.

For 2D keypoints, we select up to two human instances with the highest detection confidence per frame. When object keypoints are used, we addi-

tionally select up to two objects with the highest confidence scores. These selected keypoints are then used as inputs for action recognition.

3.2 Input and keypoint encoder

The input comprises a set of keypoints $\mathbf{I} \in \mathbb{R}^{L\times K\times M\times channel}$ where L is the number of frames, M is the maximum number of instances (humans and objects) per frame, and K is the maximum number of keypoints per instance. Each *channel* contains attributes such as 2D/3D coordinates, detection confidence, and object category. This unified representation enables flexible integration of human and object keypoints, including those from customized object detectors (*e.g.*, OVD). Because contrastive learning is performed on a per-instance basis, the *keypoint encoder* is designed to preserve instance granularity. Therefore, the instance-preserving architecture of SKP (Hachiuma et al., 2023) is adopted. The encoder f_{key} outputs instance features $\mathbf{G} = f_{\text{key}}(\mathbf{I}) \in \mathbb{R}^{L\times M\times n}$, where n is the feature dimension. During inference, $\mathbf{G}$ is aggregated by global max pooling into a video-level feature. During training, detection-based instance filtering (DIF; Sec. 3.4) extracts global and instance features $\mathbf{S}_{\text{det}}$ corresponding to up to the four description types (Sec. 3.3), which are then pooled to form the video-level feature $\mathbf{v}_c$.

3.3 Text feature generation with context

GPT-4o (OpenAI, 2024a)—an LLM—generates an action summary and instance-focused descriptions under a unified protocol. Using an action label and instruction prompt that requests a comprehensive and detailed account of the action, GPT-4o produces four types of descriptions: **(1) Action summary (Sum)**: an overall description of the action; **(2) subject behavior (Sub)**: the actor's motions and postures; **(3) relevant object (Obj)**: the roles and relationships of pertinent objects with the person; and **(4) relevant others (Oth)**: relationships between the actor and other people. We use *text-embedding-3-large* (OpenAI, 2024b) as the *text encoder*, which transforms these descriptions into embeddings $\left\{\mathbf{t}_{\text{sum}}, \mathbf{t}_{\text{sub}}, \mathbf{t}_{\text{obj}_1}, ..., \mathbf{t}_{\text{obj}_r}, \mathbf{t}_{\text{oth}}\right\}$.

3.4 Instance feature filtering per description

Instance features corresponding to each description (Sec. 3.3) are obtained via DIF. All instance features $\mathbf{G}$ are passed to the DIF function $h_{\text{det}}(\cdot)$, which selects only the instances referenced by a given description. Let $\mathbf{category\_ids} \in \mathbb{Z}^{L\times M}$ denote the matrix of object category IDs assigned to each instance by the detector. For a description with o_{cap} as the target object category (Sec. 3.3), DIF constructs a binary mask $\mathbf{B} \in \mathbb{Z}^{L\times M}$ to identify the matching instances, where each element $b(l, m)$ is given by

$$b(l,m) = \begin{cases} 1, & \text{if } \mathbf{category\_ids}(l,m) = o_{cap} \\ 0, & \text{otherwise} \end{cases}, \tag{1}$$

DIF-filtered instance features are then obtained by the element-wise product of $\mathbf{B}$ and $\mathbf{G}$:

$$\mathbf{S}_{\text{det}} = h_{\text{det}}(\mathbf{G}) = \mathbf{B} \odot \mathbf{G}, \tag{2}$$

3.5 Instance-level cross-modal alignment

For each of the (up to) four description types (Sec. 3.3), we minimize a contrastive loss between the textual features and the corresponding instance features:

$$\mathcal{L}_{\text{total}} = \sum_{c=1}^{N_{\text{cap}}} \beta_c \mathcal{L}^c_{\text{con}}, \tag{3}$$

where $\mathcal{L}^c_{\text{con}}$ denotes the contrastive loss for description type c and N_{cap} is the number of description types. The Kullback–Leibler (KL) divergence is used as the contrastive loss. Given the batched keypoint features $\mathbf{V}'_c$ and textual features $\mathbf{T}_c$ for description c, the loss is

$$\mathcal{L}^c_{\text{con}} = \tfrac{1}{2}\mathbb{E}_{\mathbf{V}'_c,\mathbf{T}_c\sim D}\begin{bmatrix}\text{KL}(\mathbf{P}^{v'_c 2t_c}(\mathbf{V}'_c), \mathbf{Y}^{v'_c 2t_c}) \\ +\,\text{KL}(\mathbf{P}^{t_c 2v'_c}(\mathbf{T}_c), \mathbf{Y}^{t_c 2v'_c})\end{bmatrix}, \tag{4}$$

where $\mathbf{P}(\mathbf{v}'_c) \in \mathbb{R}^{B\times B}$ is the similarity matrix between the two modalities in a batch of size B, converted into log-probabilities by applying *log-softmax* over j for each i. The label matrix $\mathbf{Y} \in \mathbb{R}^{B\times B}$ assigns 1 to positive pairs and 0 otherwise. If the object referenced by description c is not detected, the corresponding entry $\mathbf{Y}(i, j)$ is set to 0. The binary target matrix $\mathbf{Y}$ is then normalized by applying *softmax* over j for each i to form a valid probability distribution. This corresponds to a multi-positive setting in which multiple samples in a batch can share the same action label. Negative pairs are not explicitly penalized (Xiang et al., 2023) to avoid excessive separation of actions that share locally similar motions.

Each keypoint feature $\mathbf{v}_c$ is projected via a fully connected layer to $\mathbf{v}'_c$ so that its dimensionality

matches that of the textual feature $\mathbf{t}_c$ for computing similarity. At inference, we compute the cosine similarity between the video-level feature $\mathbf{v}'_{\text{sum}}$ and textual feature $\mathbf{t}_{\text{sum}}$ describing all candidate classes and predict the class with the highest similarity. In the following section, we show how this ZS inference framework can be seamlessly extended to Z2F adaptation without retraining.

3.6 Instance-level visual adaptation

Although InsAT is trained in a purely ZS manner, practical development scenarios often allow access to a small number of labeled samples from the target domain. To exploit such data without retraining or modifying model parameters, we introduce IVA, an inference-time extension that introduces no additional learnable parameters. Given a small number of labeled samples per class (*i.e.*, K-shot samples, where $K \in \{2, 4, 8, 16\}$), we extract human and object keypoint features using the same frozen keypoint encoder employed during ZS inference. Let $\{\mathbf{v}'^{(c)}_i\}_{i=1}^{K}$ denote the aggregated visual representations obtained from the K samples of class c. We construct a class-specific visual prototype by averaging these representations:

$$\mathbf{v}'^{(c)}_{\text{iva}} = \frac{1}{K} \sum_{i=1}^{K} \mathbf{v}'^{(c)}_i, \tag{5}$$

Although each prototype aggregates multiple instances, it remains grounded in the instance-level semantics learned during ZS training.

For a query input at inference time, we obtain its aggregated visual representation $\mathbf{v}'_{\text{sum}}$ and compute the final similarity score by combining text-based and visual-based similarities:

$$s^{(c)} = (1-\alpha)\cos(\mathbf{v}'_{\text{sum}}, \mathbf{t}^{(c)}_{\text{sum}}) + \alpha \cos(\mathbf{v}'_{\text{sum}}, \mathbf{v}'^{(c)}_{\text{IVA}}), \tag{6}$$

where $\mathbf{t}^{(c)}_{\text{sum}}$ denotes the textual embedding of class c and $\alpha \in [0, 1]$ controls the contribution of FS visual evidence. This formulation preserves language guided semantic grounding while selectively adapting to instance-specific visual patterns in the target domain. We select α via grid search on the validation set for each dataset and shot setting; larger α values are consistently preferred as K increases (Appendix F.1).

Crucially, IVA requires no gradient-based optimization and no parameter updates: the keypoint encoder and the text encoder remain fully frozen, enabling efficient and stable Z2F transfer suitable for low-resource and deployment-oriented settings.

4 Experiments

We evaluated the effectiveness of InsAT through comprehensive comparison with state-of-the-art methods and detailed ablation analyses under ZS and Z2F settings. Top-1 accuracy (Acc.) is used as the primary evaluation metric.

In the ZS setting, each dataset is split into seen and unseen action classes (*e.g.*, 180/20 for Kinetics-200). Models are trained exclusively on the seen classes and evaluated on the unseen classes. In the Z2F setting, we randomly sample K labeled examples per unseen class to construct IVA prototypes (Sec. 3.6) and classify the remaining unseen-class samples. For each value of K, this procedure is repeated five times with different random splits, and the average accuracy is reported.

To analyze the transferability of object-centric knowledge, we conducted experiments on Kinetics-ObjShared (Sec. 4.1.2) under the ZS setting. In addition, to evaluate the extensibility of InsAT beyond the category limitations of conventional object detectors, we assess performance on Kinetics-ObjExt (Sec. 4.1.2) under both ZS and Z2F settings.

4.1 Datasets

4.1.1 Benchmark datasets

We evaluated our method on the following benchmark datasets:

Kinetics-400 (Carreira and Zisserman, 2017) contains 400 classes and 270k 10-second YouTube clips. **Kinetics-200** (Zhu et al., 2024) comprises the first 200 classes of Kinetics-400.

NTU-RGBD 60 (Shahroudy et al., 2016) spans 60 classes and ~57k clips covering daily, interactive, and healthcare actions. **NTU-RGBD 120** (Liu et al., 2020a) is its extended version with 120 classes and ~114k clips. Each frame provides 3D coordinates of 25 joints for up to two people. Although video lengths vary, we clip them to a maximum of 10 seconds. For each frame, 3D human keypoints are used as input features for NTU-RGBD.

HMDB51 (Kuehne et al., 2011) is a relatively small dataset comprising 6.7K videos collected from YouTube, categorized into 51 action classes. For the ZS and Z2F settings, following a previous study (Ahmad et al., 2025), the model is trained on Kinetics-400 and tested on HMDB51. Although video lengths vary, we clip them to a maximum of 10 seconds.

Model	Obj.	Kinetics-200				Kinetics-400				NTU-RGBD 60		NTU-RGBD 120	
		180/20	160/40	140/60	120/80	360/40	320/80	300/100	280/120	48/12	30/30	96/24	60/60
ReViSE (Xian et al., 2017)	✗	24.95	13.28	8.14	6.23	20.84	11.82	9.49	8.23	26.44	14.81	37.96	8.27
DeViSE (Frome et al., 2013)	✗	22.22	12.32	7.97	5.65	18.37	10.23	9.47	8.34	35.80	18.45	40.91	12.19
JPoSE (Wray et al., 2019)	✗	–	–	–	–	–	–	–	–	28.75	12.39	32.44	7.65
SynSE (Gupta et al., 2021)	✗	–	–	–	–	–	–	–	–	33.30	12.00	38.70	7.73
SMIE (Zhou et al., 2023)	✗	–	–	–	–	–	–	–	–	40.18	–	45.30	–
PURLS (Label) (Zhu et al., 2024)	✗	25.96	15.85	10.23	7.77	22.50	15.08	11.44	11.03	35.46	16.29	44.27	14.12
PURLS (Zhu et al., 2024)	✗	32.22	22.56	12.01	11.75	34.51	24.32	16.99	14.28	40.99	23.52	52.01	19.63
SCoPLe (Zhu et al., 2025a)	✗	–	–	–	–	–	–	–	–	52.96	–	52.33	–
Neuron (Chen et al., 2025)	✗	–	–	–	–	–	–	–	–	**62.7**	–	57.1	–
Ours (Label, pose only)	✗	44.68	26.52	20.21	16.76	42.59	31.54	23.53	22.55	48.60	27.54	55.34	22.64
Ours (InsAT, pose only)	✗	46.73	31.25	22.52	19.24	43.35	33.20	24.53	23.96	50.81	**29.31**	**57.29**	**22.82**
Ours (Label, pose + object)	✓	54.48	36.12	27.35	20.51	53.70	39.99	30.92	31.51	–	–	–	–
Ours (InsAT, pose + object)	✓	**55.51**	**38.80**	**28.56**	**24.07**	**56.14**	**41.55**	**32.64**	**32.05**	–	–	–	–

2D skeleton detector: OpenPose (Cao et al., 2017); 2D object detector: PPNv2 (Sekii, 2021).

Table 1: Comparison of ZS accuracy [%] with existing ZSL methods on Kinetics-200, Kinetics-400, NTU-RGBD 60, and NTU-RGBD 120. For NTU-RGBD datasets, only 3D human joint positions are available; therefore, object information is not included in InsAT. For PURLS and InsAT, *Label* denotes the setting in which textual embeddings of action labels are aligned with global keypoint representations.

Model	Modality	Params	Params updated (Z2F)	ZS	$K = 2$	$K = 4$	$K = 8$	$K = 16$
ActionCLIP (Wang et al., 2021)	RGB	168.5M	✓	40.8 ±5.4	47.5	57.9	57.3	59.1
XCLIP (Ni et al., 2022)	RGB	131.5M	✓	44.6 ±5.4	53.0	57.3	62.8	62.4
ViFi CLIP (Rasheed et al., 2023)	RGB	124.7M	✓	51.3 ±0.6	57.2	**62.7**	64.5	66.8
T2L (Ahmad et al., 2025)	RGB	154.2M†	✓	**52.9***	**57.3**	61.1	**65.4**	**67.7**
InsAT with Full-FT	Keypoint	**29.1M**‡	✓	**43.8 ± 0.8**	44.4	51.8	58.4	**66.3**
InsAT with Last-1	Keypoint	**29.1M**‡	✓	**43.8 ± 0.8**	43.1	48.1	53.9	62.0
InsAT with IVA	Keypoint	**29.1M**‡	✗	**43.8 ± 0.8**	**52.7**	**56.9**	**61.1**	**63.5**

† Includes frozen encoders; 5.2M parameters are trainable. ‡ Excludes the text encoder, which is not used during inference. Results are averaged over three splits (* standard deviation not reported). All RGB-based models share a CLIP ViT-B/16 backbone.

Table 2: Comparison of ZS and Z2F adaptation ($K = 2, 4, 8, 16$) accuracy [%] between InsAT and RGB-based VLMs on HMDB51. All models are pretrained on Kinetics-400.

4.1.2 Additional custom subsets

To further investigate the characteristics of InsAT, we constructed several subsets derived from Kinetics-400, as described below. For reproducibility, the class splits for Kinetics-ObjShared and Kinetics-ObjExt are provided in Appendix B.

Kinetics-ObjShared was designed to evaluate object-aware knowledge transfer between actions involving the same object covered by the common COCO (Lin et al., 2014) objects and our default configuration (*e.g.*, *training dog* vs. *grooming dog*). The subset was divided into 12 seen and 10 unseen classes, emphasizing unseen-class recognition when multiple actions involved the same object.

Kinetics-ObjExt evaluated the extensibility of InsAT for actions involving objects beyond COCO and our default configuration, while also assessing object-aware knowledge transfer across actions sharing the same object. Important objects related to target actions but absent from COCO were detected using an OVD. The subset was divided into 10 seen and 6 unseen classes. For example, it includes actions such as *playing piano* and *playing organ*, whose objects are not covered by COCO but are handled using the OVD.

4.2 Comparison with existing ZSL baselines

Tab. 1 shows that InsAT establishes new state-of-the-art performance on both Kinetics-200 and Kinetics-400. Notably, by incorporating object keypoints, InsAT (*pose + object*) consistently outperforms PURLS across all seen/unseen splits, achieving gains of up to +23.2% points on Kinetics-200 (180/20). We further compared InsAT with a *Label* variant, which aligned textual features of action labels with all instance features, following the strategy used in PURLS. Across both *pose only* and *pose + object* configurations, instance-level alignment consistently yields superior performance, confirming the benefit of modeling fine-grained instance semantics.

On NTU-RGBD 60/120, where only 3D human joints are available, InsAT (*pose only*) remains highly competitive and outperforms prior keypoint-based methods such as PURLS and SCoPLe. Although Neuron reports the strongest results overall, InsAT achieves slightly higher accuracy on NTU-RGBD 120 (96/24). Unlike Neuron, which relies

Keypoints	Kinetics180/20				Kinetics120/80			
	$K=2$	$K=4$	$K=8$	$K=16$	$K=2$	$K=4$	$K=8$	$K=16$
pose only	49.50 (↑4.38)	53.46 (↑8.34)	56.89 (↑11.77)	59.10 (↑13.98)	26.61 (↑7.37)	30.99 (↑11.75)	34.89 (↑15.65)	37.16 (↑17.92)
pose + object	**58.46 (↑6.31)**	**62.81 (↑10.66)**	**66.93 (↑14.78)**	**69.17 (↑17.02)**	**32.55 (↑8.48)**	**38.15 (↑14.08)**	**42.83 (↑18.76)**	**45.76 (↑21.69)**

Table 3: Z2F adaptation results on two seen/unseen splits of Kinetics-200 (180/20 and 120/80). Accuracy [%] and improvements over the ZS baseline (in parentheses) are reported. Note that the ZS models for the 180/20 split are obtained from a different training run than that used in Tab. 1.

Strategy	OVD in Pre-training Data	OVD in Z2F Adaptation Data	OVD in Inference Data	Training–Inference Input Mismatch	Additional Pre-training Cost
No-OVD	✗(Common only)	✗(Common only)	✗(Common only)	None	None
Full-OVD	✓(Replace)	✓(Replace)	✓(Replace)	None	**High**
Replace-OVD	✗(Common only)	✓(Replace)	✓(Replace)	**Severe**	None
Add-OVD	✗(Common only)	✓(Add)	✓(Add)	*Partial*	None

Table 4: Comparison of OVD extension strategies, summarized by the stages at which OVD is applied (pretraining, adaptation, inference), the resulting training–inference input mismatch, and additional pretraining cost. "Common only" denotes using detections from a closed-set object detector. "Replace" substitutes a subset of common-detector outputs with OVD detections, while "Add" appends OVD detections to the original outputs.

OVD extension	ZS ($K=0$)	Z2F ($K=16$)
✗(No-OVD)	73.66	75.25
✓(Full-OVD)	**82.15**	**87.53**

Table 5: Effect of OVD extension on Kinetics-ObjExt. Accuracy [%] for ZS and Z2F adaptation. See Tab. 4 for extension definitions.

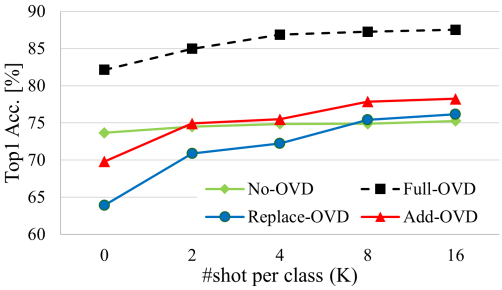

Figure 4: Accuracy [%] under different OVD extension strategies vs. number of shots. Full-OVD denotes an upper-bound reference.

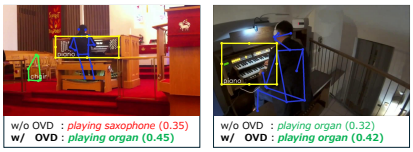

Figure 5: Qualitative results of OVD-based extension on Kinetics-ObjExt (*playing organ*). Common object detections (PPNv2, in green) and OVD detections (in yellow) are visualized; OVD correctly detects the *piano*, corresponding to the *organ*. The Top-1 prediction and similarity score are displayed. Green/red text indicates correct/incorrect predictions.

***Note:* PPNv2 outputs 8-point convex hulls rather than axis-aligned bounding boxes.**

on phase-wise semantics specifically tailored to skeleton benchmarks, InsAT learns instance-level semantic representations that naturally generalize to HOI scenarios.

4.3 Comparison with RGB-based VLMs for ZS recognition and Z2F adaptation

As discussed in Sec. 1, large RGB-based VLMs demonstrate strong performance in Z2F action recognition. However, keypoint-based approaches often outperform RGB-based models on action categories dominated by human motion patterns, such as gestures (Käs et al., 2025). To contextualize InsAT within this broader ZS/Z2F setting, we compared it with representative large RGB-based VLMs (Tab. 2). RGB-based VLMs typically rely on large-scale image–text pretraining (e.g., CLIP (Radford et al., 2021)), whereas an analogous pretraining paradigm based on image-level keypoint–text pairs has not been established. We therefore report cross-paradigm comparisons on HMDB51, a widely used benchmark for RGB-based ZS/Z2F evaluation that also supports keypoint-based methods.

In the ZS setting, InsAT with IVA performs slightly below the latest T2L model but surpasses ActionCLIP (43.8% vs. 40.8%) while using approximately one-fifth of the model parameters. More importantly, InsAT with IVA exhibits consistent and monotonic performance gains as Z2F adaptation is introduced. Accuracy steadily increases from $K=2$ to $K=16$, reaching 63.5% at $K=16$, thereby substantially narrowing the performance gap with large RGB-based VLMs that rely on significantly higher model capacity.

These results indicate that InsAT effectively bridges ZS recognition and Z2F adaptation by leveraging instance-level context encoded in human and object keypoints, without relying on raw RGB appearance information.

4.4 Zero-to-few-shot extension analysis

As discussed in Secs. 1 and 2.2, existing keypoint-based ZSL methods rarely explore extensions from ZS to FS settings. Pose-only inputs may provide limited contextual cues, making adaptation under scarce labeled sampled more challenging.

To examine this limitation, we compare two input configurations: *pose only* and *pose + object*. Tab. 3 reports results on two seen/unseen splits of Kinetics-200 . Incorporating object keypoints consistently improves performance across both splits, with gains increasing as the number of shots K grows (approximately 2%–4% points).

These results suggest that object keypoints provide complementary contextual cues for modeling HOIs, yielding more reliable adaptation signals under limited supervision.

To further analyze the effectiveness of IVA, we compare it with two standard fine-tuning strategies: (i) full fine-tuning of the keypoint encoder (Full-FT) and (ii) updating only the last block and projection layers (Last-1). As shown in Tab. 2, IVA achieves strong Z2F performance without parameter updates, particularly in low-shot settings, while avoiding catastrophic forgetting (Appendix F.2).

4.5 OVD-based extension

InsAT assumes that accurate action depends on identifying objects closely associated with HOIs and scene context. However, conventional closed-set object detectors are limited to predefined categories and often fail to recognize action-relevant objects outside this set. To address this limitation, we integrate an OVD to dynamically extend object categories based on scene content. In Kinetics-ObjExt, we introduce seven additional object categories (*e.g.*, *piano* and *barbell*, see Tab. 10) that are relevant to the target action. Tab. 5 compares InsAT with and without OVD-based extension under both ZS and Z2F ($K = 16$) settings. OVD integration improves ZS accuracy by 8.49% points and yields consistent gains in Z2F, demonstrating that InsAT can flexibly adapt to diverse scenes without retraining the object detector.

Fig. 5 shows qualitative examples, showing that detecting action-related objects increases the confidence of correct predictions. These results illustrate that InsAT effectively leverages object-level cues when recognizing actions that strongly involve object interactions.

Method	Acc. [%]
Label	50.73
InsAT	**56.11**

Table 6: Object-aware transfer. ZS accuracy [%] on Kinetics-ObjShared.

Sum	Sub	Oth	Obj	Acc. [%]
✗	✗	✗	✗	50.73
✓	✗	✗	✗	54.20 (↑ 3.47)
✓	✓	✗	✗	54.90 (↑ 4.17)
✓	✗	✓	✗	55.51 (↑ 4.78)
✓	✗	✗	✓	55.04 (↑ 4.31)
✓	✓	✓	✓	**56.11**(↑ **5.38**)

Table 7: Effect of instance-level contrastive losses. ZS accuracy [%] on Kinetics-ObjShared.

4.6 Ablation studies

4.6.1 Object-centric analysis

OVD extension strategy. Fig. 4 compares different OVD extension strategies defined in Tab. 4 from ZS to FS. Full-OVD consistently achieves the highest accuracy across all shot settings, indicating that object information introduced by OVD remains effective during adaptation. In contrast, No-OVD exhibits limited performance gains, demonstrating that relying solely on common object detectors constrains adaptation effectiveness.

Despite its strong performance, Full-OVD requires running OVD inference over the entire pretraining dataset and retraining InsAT for each customized object set, resulting in substantial computational overhead. To address this limitation, Add-OVD introduces OVD only during Z2F adaptation and inference. This strategy substantially narrows the performance gap with Full-OVD while avoiding additional pretraining, thereby offering a more favorable accuracy–efficiency trade-off.

Object-aware transfer. Object-aware knowledge transfer is expected to be most effective when different actions share the same object. To verify this, we constructed Kinetics-ObjShared, which comprises different actions that share the same object (*e.g.*, *kicking a soccer ball* vs. *shooting a goal [soccer]*). The subset was split into 12 seen and 10 unseen classes. Tab. 6 shows that InsAT's accuracy improves by 5.38% points over the *Label* setting. This improvement exceeds those observed for other Kinetics-400/200 splits, indicating that object-aware knowledge transfer is particularly effective for recognizing actions dependent on shared objects.

4.6.2 Text supervision design

Effect of instance-level alignment. Tab. 7 shows that replacing *Label* (first row) with **Sum** improves accuracy, which is further enhanced by incorporating **Sub**, **Oth**, or **Obj**. The best performance

is achieved using all descriptions, highlighting the benefit of multifaceted instance alignment.

Class-level vs. data-level captions. We generated data-level captions for Kinetics-ObjShared (1/10 training data, 823 videos) using GPT-4o, where each video's representative frame and label are provided to produce a unique caption. Accuracy slightly improves from 51.44% (class-level) to 52.83%. Caption generation is performed offline, and class-level captions are still used at inference time for consistency and efficiency. Overall, class-level captions ensure stable, low-cost training, whereas data-level captions can enhance contextual understanding as generative quality and data diversity increase.

A supplementary comparison with template-based captions shows that although LLM-generated detailed captions perform best, InsAT remains effective with template-based captions, highlighting the importance of instance-level language–visual alignment (Appendix E.3). We also confirmed in a limited setting that an open-source LLM can generate compatible captions, suggesting that InsAT does not rely on a specific proprietary model (Appendix E.4).

4.6.3 Robustness to detection noise

Since InsAT relies on object-aware instance selection via DIF, we evaluate its robustness to noisy object detections at inference time on Kinetics-ObjShared. We corrupt an α fraction of object instances in three ways: *Missing*, which removes the selected instances; *Mislocalized*, which perturbs keypoint coordinates with random noise sampled uniformly from -20% to 20% of the normalized coordinate range; and *Category Switch*, which replaces detected object category IDs with random ones.

Tab. 8 shows that both InsAT and the model without InsAT remain robust under *Missing* and *Mislocalized* noise. InsAT exhibits smaller drops under these structural corruptions, suggesting that DIF-based instance selection mitigates the effects of noisy object cues. Under *Category Switch*, however, InsAT exhibits larger drops, consistent with its reliance on object-category semantics for language–visual alignment.

For completeness, fully supervised results are provided in Appendix D.1, where InsAT achieves the best performance among compared methods.

Method	Missing	Mislocalized	Category Switch
InsAT	-2.36	-0.02	-15.49
w/o InsAT	-6.44	-0.09	-11.51

Table 8: ZS accuracy change (percentage points) on Kinetics-ObjShared under inference time detection noise at $\alpha = 0.8$, relative to $\alpha = 0$.

5 Conclusion

This study introduced InsAT, a unified framework for ZS and Z2F action recognition based on human and object keypoints. By aligning instance-level language descriptions with keypoint-based visual representations, InsAT effectively transfers HOI knowledge to unseen actions. We further proposed IVA, a parameter-free mechanism that enables efficient Z2F adaptation without updating model parameters. Extensive experiments and evaluations demonstrate that InsAT substantially outperforms prior keypoint-based ZS methods and achieves competitive performance relative to large VLMs, while remaining data-efficient and robust. Overall, our work highlights the potential of instance-aware language supervision for scalable and practical action understanding beyond RGB-based paradigms.

6 Limitations

First, InsAT relies on the availability of accurate human and object keypoints, which may limit its robustness in highly complex real-world scenes. Although we provide qualitative examples of HOIs and supplementary results on challenging datasets, keypoint-based representations can become unstable in scenarios involving dense crowds, severe occlusion, or small and visually ambiguous objects. Consequently, our conclusions primarily apply to settings in which reliable keypoint extraction is feasible.

Second, our approach adopts class-level language descriptions to achieve stable and scalable semantic alignment. While data-level captions can be incorporated, our ablation studies indicate only modest performance improvements, suggesting that instance-specific captions may not consistently align with noisy or incomplete keypoint detections.

Ethical Considerations

Our method uses instance-level natural language descriptions generated by an LLM, whose quality and bias may depend on prompt design and model

behavior. To mitigate this risk, we employ a consistent, template-based generation strategy across all action classes. We manually inspected descriptions for 100 randomly sampled classes from Kinetics-400 and did not observe ethically problematic content; however, this evaluation is limited in scale and relies on subjective judgment.

Additionally, although InsAT operates on privacy-preserving human and object keypoints rather than raw images, it may still be misused for surveillance without consent.

References

Shahzad Ahmad, Sukalpa Chanda, and Yogesh S Rawat. 2025. T2L: Efficient zero-shot action recognition with temporal token learning. *Transactions on Machine Learning Research*, 2025.

Salman Ali, Shaheer Athar, Hamed Larijani, Josef Kittler, and Henglin Dai. 2019. Skeleton based zero shot action recognition in joint pose-language semantic space. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1459–1467. IEEE.

Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. 2017. Realtime multi-person 2d pose estimation using part affinity fields. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1302–1310.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-End Object Detection With Transformers. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

João Carreira and Andrew Zisserman. 2017. Quo vadis, action recognition? A new model and the kinetics dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4724–4733. IEEE.

Yang Chen, Jingcai Guo, Song Guo, and Dacheng Tao. 2025. Neuron: Learning context-aware evolving representations for zero-shot skeleton action recognition. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8721–8730.

Ming Cheng, Kunjing Cai, and Ming Li. 2021. Rwf-2000: An open large scale video database for violence detection. In *Proceedings of the 25th International Conference on Pattern Recognition (ICPR)*, pages 4183–4190. IEEE.

Haodong Duan, Yue Zhao, Kai Chen, Dahua Lin, and Bo Dai. 2022. Revisiting skeleton-based action recognition. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2959–2968.

Andrea Frome, Greg S. Corrado, Jon Shlens, Samy Bengio, Jeffrey Dean, Marc'Aurelio Ranzato, and Tomás Mikolov. 2013. DeViSE: A Deep Visual-Semantic Embedding Model. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Pranay Gupta, Divyanshu Sharma, and Ravi Kiran Sarvadevabhatla. 2021. Syntactically Guided Generative Embeddings for Zero-Shot Skeleton Action Recognition. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*.

Ryo Hachiuma, Fumiaki Sato, and Taiki Sekii. 2023. Unified keypoint-based action recognition framework via structured keypoint pooling. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22962–22971.

Tal Hassner, Yossi Itcher, and Orit Kliper-Gross. 2012. Violent flows: Real-time detection of violent crowd behavior. In *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–6.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

Stephanie Käs, Anton Burenko, Louis Markert, Onur Alp Culha, Dennis Mack, Timm Linder, and Bastian Leibe. 2025. How do foundation models compare to skeleton-based approaches for gesture recognition in human-robot interaction? *Preprint*, arXiv:2506.20795.

Hilde Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. 2011. HMDB: A large video database for human motion recognition. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Sang Uk Lee, Andreas Hofmann, and Brian Williams. 2019. A Model-Based Human Activity Recognition for Human–Robot Collaboration. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: Common objects in context. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Hongda Liu, Yunfan Liu, Min Ren, Hao Wang, Yunlong Wang, and Zhenan Sun. 2025a. Revealing key details to see differences: A novel prototypical perspective for skeleton-based action recognition. In

2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 29248–29257.

Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, and Alex C. Kot. 2020a. NTU RGB+D 120: A large-scale benchmark for 3d human activity understanding. *IEEE Trans. Pattern Anal. Mach. Intell.*, 42(10):2684–2701.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. 2025b. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Xingyu Liu, Sanping Zhou, Le Wang, and Gang Hua. 2023. Parallel attention interaction network for few-shot skeleton-based action recognition. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1379–1388.

Ziyu Liu, Hongwen Zhang, Zhenghao Chen, Zhiyong Wang, and Wanli Ouyang. 2020b. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 140–149.

Ning Ma, Hongyi Zhang, Xuhui Li, Sheng Zhou, Zhen Zhang, Jun Wen, Haifeng Li, Jingjun Gu, and Jiajun Bu. 2022. Learning spatial-preserved skeleton representations fornbsp;few-shot action recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Bolin Ni, Houwen Peng, Minghao Chen, Songyang Zhang, Gaofeng Meng, Jianlong Fu, Shiming Xiang, and Haibin Ling. 2022. Expanding language-image pretrained models fornbsp;general video recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

OpenAI. 2024a. Gpt-4o system card. OpenAI Technical Report. Reported: 2024-8-8.

OpenAI. 2024b. New embedding models and API updates. Reported: 2024-02-14.

Yansong Peng, Hebei Li, Peixi Wu, Yueyi Zhang, Xiaoyan Sun, and Feng Wu. 2024. D-FINE: Redefine regression task in detrs as fine-grained distribution refinement. arXiv preprint arXiv:2410.13842.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139 of *PMLR*, pages 8748–8763. PMLR.

Hanoona Rasheed, Muhammad Uzair Khattak, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. 2023. Fine-tuned clip models are efficient video learners. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6545–6554.

Taiki Sekii. 2018. Pose proposal networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Taiki Sekii. 2021. Object Detection Method and Object Detection Device. Patent WO2021/117363.

Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. 2016. NTU RGB+D: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1010–1019. IEEE.

Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. 2020. Finegym: A hierarchical video dataset for fine-grained action understanding. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2613–2622.

Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. 2020. Decoupled spatial-temporal attention network for skeleton-based action-gesture recognition. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, pages 38–53.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. MPNet: Masked and permuted pre-training for language understanding. In *Proceedings of the 34th International Conference on Neural Information Processing Systems (NeurIPS)*.

Yi-Fan Song, Zhang Zhang, Caifeng Shan, and Liang Wang. 2023. Constructing stronger and faster baselines for skeleton-based action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):1474–1488.

Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. 2019. Deep high-resolution representation learning for human pose estimation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5686–5696.

Mengmeng Wang, Jiazheng Xing, and Yong Liu. 2021. Actionclip: A new paradigm for video action recognition. *Preprint*, arXiv:2109.08472.

Michael Wray, Gabriela Csurka, Diane Larlus, and Dima Damen. 2019. Fine-grained action retrieval through multiple parts-of-speech embeddings. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 450–459.

Yongqin Xian, Bernt Schiele, and Zeynep Akata. 2017. Zero-shot learning — the good, the bad and the ugly. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3077–3086.

Wangmeng Xiang, Chao Li, Yuxuan Zhou, Biao Wang, and Lei Zhang. 2023. Generative action description prompts for skeleton-based action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10242–10251.

Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. 2021. VideoCLIP: Contrastive pre-training for zero-shot video-text understanding. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6787–6800. Association for Computational Linguistics.

Sijie Yan, Yuanjun Xiong, and Dahua Lin. 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1).

Yujie Zhou, Wenwen Qiang, Anyi Rao, Ning Lin, Bing Su, and Jiaqi Wang. 2023. Zero-Shot Skeleton-Based Action Recognition via Mutual Information Estimation and Maximization. In *Proceedings of the 31st ACM International Conference on Multimedia*.

Anqi Zhu, Qiuhong Ke, Mingming Gong, and James Bailey. 2024. Part-aware unified representation of language and skeleton for zero-shot action recognition. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18761–18770.

Anqi Zhu, Jingmin Zhu, James Bailey, Mingming Gong, and Qiuhong Ke. 2025a. Semantic-guided cross-modal prompt learning for skeleton-based zero-shot action recognition. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13876–13885.

Jingmin Zhu, Anqi Zhu, Hossein Rahmani, Jun Liu, Mohammed Bennamoun, and Qiuhong Ke. 2025b. Boosting skeleton-based zero-shot action recognition with training-free test-time adaptation. In *The 39th Annual Conference on Neural Information Processing Systems (NeurIPS)*.

Appendix

A Keypoint detector

OpenPose (Cao et al., 2017). For the Kinetics-400 and Kinetics-200, we use human skeletal keypoints detected by OpenPose. OpenPose introduces an intermediate representation known as Part Affinity Fields, which enables the simultaneous estimation of multi-person 2D keypoints and their limb associations. The method is robust in scenes containing multiple individuals and supports real-time processing. We use publicly available keypoints inferred by OpenPose and released by (Duan et al., 2022) under the Apache License Version 2.0.

For the NTU-RGBD 60 and NTU-RGBD 120 datasets, we use publicly available 3D skeletal keypoints provided by (Duan et al., 2022), which are released under the Apache License Version 2.0.

PPNv2 (Sekii, 2018, 2021). For experiments on Kinetics-400 and Kinetics-200, object detection is performed using Pose Proposal Networks v2 (PPNv2) to obtain object keypoints. PPNv2 jointly estimates human keypoints, object keypoints, and object categories from RGB videos in real-time or faster. The model adopts a ResNet-101 backbone (He et al., 2016) and is trained on the MS-COCO dataset (Lin et al., 2014). In ZS settings, although PPNv2 can output both human and object keypoints, we use only its object detections, while human keypoints are unified with those obtained from OpenPose to ensure fair comparison across methods in Tab. 1. In supervised settings on the Kinetics datasets, PPNv2 is additionally used for human keypoint estimation to ensure comparability with prior work.

HRNet (Sun et al., 2019). For experiments on HMDB51 (Kuehne et al., 2011) we adopt HRNet as the human pose detector. HRNet is a top–down approach known for its high pose estimation accuracy and strong performance relative to prior methods. We use publicly available HRNet-inferred keypoints for HMDB51 released by (Duan et al., 2022). The same detectors and configurations are consistently applied across ZS, few-shot (FS), and fully supervised experiments on HMDB51.

D-FINE-X (Peng et al., 2024). To incorporate object keypoints in the HMDB51-based experiment, we employ D-FINE-X. D-FINE-X models bounding box regression within the DETR framework (Carion et al., 2020) by predicting probability distributions rather than fixed coordinates, enabling multi-stage optimization for improved detection accuracy and efficiency.

In these experiments, human joint keypoints detected by OpenPose or HRNet are combined with object detections produced by PPNv2 or D-FINE-X to construct a dataset containing both human and object keypoints. For **PPNv2**, object keypoints are defined as the eight vertices of the convex hull predicted by the model. For **D-FINE-X**, object keypoints are defined by selecting the eight extremal points (four corners and four edge midpoints) from each detected bounding box.

Grounding DINO (Liu et al., 2025b) (OVD). To extend object coverage beyond the predefined COCO categories supported by PPNv2 or D-FINE-X, we employ Grounding DINO as an OVD for the extensibility experiments described in Secs. 4.5 and 4.6.1. Grounding DINO aligns textual prompts with visual features, enabling object detection beyond a fixed category set.

In our experiments, we provide action-relevant names as text prompts and extract corresponding objects' bounding boxes. These detections are then converted into object keypoints using the same eight-point extremal representation as in the closed-set setting. OVD-based detection is applied only in the object extensibility analysis and is not used in the standard evaluation protocol.

B Custom dataset

Kinetics-ObjShared

Kinetics-ObjShared is a custom subset of Kinetics-400 in which each unseen class shares one or more relevant objects with the seen classes (see 4.1.2). This subset is used for the evaluations reported in Tab. 6 and Tab. 7. Tab. 9 lists all 12 seen and 10 unseen action classes along with their associated relevant objects.

Kinetics-ObjExt

Kinetics-ObjExt is a custom subset of Kinetics-400 designed to evaluate the extensibility of InsAT to actions involving objects that fall outside the scope of common closed-set object detectors. Action-relevant important objects that are absent from COCO are detected using an OVD.

Experimental results on this subset are described in Sec. 4.5 and 4.6.1.

The included action classes and their corresponding relevant objects are listed in Tab. 10. These object names are provided to the OVD as text prompts,

and the resulting detections are converted into object keypoints that serve as inputs to InsAT.

Kinetics-1/10 and Kinetics-1/20

Kinetics-1/10 and Kinetics-1/20 were created to assess data efficiency under the supervised setting. They were smaller-scale subsets obtained by randomly sampling the Kinetics-400 training set, reducing the number of videos per class by factors of 1/10 and 1/20. The evaluation set remained the same as Kinetics-400, and three random splits (1–3) were generated to account for sampling variation.

C Implementation details

We use the *text-embedding-3-large* text encoder (OpenAI, 2024b) to convert the captions generated as described in Sec. 3.3 into 3,072-dimensional text embeddings. For comparison, we also evaluate two alternative encoders—CLIP ViT-L/14 (Radford et al., 2021) and MPNet (Song et al., 2020)—whose results are reported in Sec. E.2. Both CLIP ViT-L/14 and MPNet map each caption to a 768-dimensional embedding.

Captions associated with (*relevant object*) (defined in Sec. 3.3) are discarded if the corresponding objects are not detected in the input. To control the contribution of object-related semantics, we include the three most relevant objects per action label in the contrastive loss formulation.

Following (Hachiuma et al., 2023), we adopt a linear learning-rate decay scheduler across all experimental settings. For Kinetics-400, Kinetics-200, NTU-RGBD 60, and NTU-RGBD 120, we first pretrain the keypoint encoder using a supervised classification loss on the seen classes. After this pretraining stage, we perform contrastive learning using the loss defined in eq. (3) for 20 epochs on each split of Kinetics-400 and Kinetics-200, and for 30 epochs on NTU-RGBD 60 and NTU-RGBD 120, with a batch size of 100.

For HMDB51, Kinetics-ObjShared, and Kinetics-ObjExt, the model is trained directly under the ZS training setting, without supervised pretraining on seen classes, to better reflect scenarios where labeled seen-class data are unavailable.

When applied, supervised pretraining is conducted for 150 epochs with a batch size of 120 and a learning rate of 0.12, following the protocol of SKP (Hachiuma et al., 2023). The resulting weights are used to initialize the model before contrastive learning.

All experiments are conducted on a workstation equipped with an Intel Core i7-14700K CPU and two NVIDIA GeForce RTX 3090 GPUs. Training InsAT for 20 epochs on each split of Kinetics-200 using the *pose + object* configuration requires approximately 8–10 hours.

C.1 Hyperparameters

Hyperparameters for ZSL are selected using a coarse-to-fine grid search, followed by step-by-step refinement based on validation performance. The final hyperparameter settings used across experiments are summarized in Tab. 11.

D Experiments in supervised settings

We report additional experiments under fully supervised settings to further examine the effectiveness of InsAT beyond ZS and FS scenarios. All supervised experiments are conducted on standard action recognition benchmarks, using the same keypoint extraction and training configurations described above unless otherwise specified.

D.1 Comparison with keypoint-based baselines

Tab. 12 shows the comparison of the action classification accuracy of InsAT and those of state-of-the-art keypoint-based methods in supervised settings. On Kinetics-400 and HMDB51, integrating InsAT into the previous best-performing approach, SKP (Hachiuma et al., 2023), further improves accuracy by 1.3% and 1.1% points, respectively, achieving new state-of-the-art results.

The magnitude of improvement is relatively modest; however, this is consistent with prior findings in keypoint-based action recognition. For example, a previous study (Xiang et al., 2023) reported gains of 0.4–1.2% points by incorporating contrastive learning with part-level textual descriptions on a different dataset, which is comparable to the improvements observed here.

D.2 Training efficiency

We assess data efficiency using reduced-scale subsets—Kinetics-1/10 and Kinetics-1/20—by randomly downsampling the training data. We train SKP (Hachiuma et al., 2023), with and without InsAT, from scratch on the Kinetics-1/10 and Kinetics-1/20 datasets and average the results over three runs (Tab. 13). InsAT improves accuracy by 2.1% and 1.6% points on Kinetics-1/10 and

relevant objects	seen classes	unseen classes
basket ball	dribbling basketball, dunking basketball	playing basketball
soccer ball	kicking soccer ball, juggling soccer ball	shooting goal (soccer)
dog	training dog, walking the dog	grooming dog, sled dog racing
car	pushing car	checking tires
bike	riding a bike	riding mountain bike
horse	grooming horse	riding or walking with horse
book	reading book	bookbinding
bowl, knife, fork	cooking chicken, cooking egg	cooking on campfire, cooking sausages

Table 9: List of all seen and unseen action classes in Kinetics-ObjShared, grouped by their corresponding relevant objects. Experimental results on this subset are reported in Tabs. 6 and 7.

relevant objects	seen classes	unseen classes
golf club	golf chipping, golf putting	golf driving
barbell	bench pressing, clean and jerk	snatch weight lifting
piano, microphone	playing piano, playing keyboard	playing organ
trumpet	playing trumpet, playing flute	playing saxophone
rope	climbing a rope	skipping rope
shovel	digging	shoveling snow

Table 10: List of all seen and unseen action classes in Kinetics-ObjExt, grouped by relevant objects. These object names are provided as text prompts to an OVD to obtain object detections. Experimental results on this subset are reported in Secs. 4.5 and 4.6.1.

Kinetics-1/20, respectively, demonstrating its robustness and effectiveness even when the training data are limited.

D.3 Evaluation in crowded scenes

The Crowd Violence dataset (Hassner et al., 2012) consists of video clips annotated for violent or non-violent behavior, predominantly captured in public crowd scenes. The dataset presents significant challenges, including heavy occlusions, overlapping individuals, and subtle motion patterns, making it well-suited for evaluating the robustness of action recognition models in dense environments. To specifically assess InsAT's ability to model human–human interactions under crowd conditions, we restrict the input modality to human keypoints only, excluding object information. This design choice isolates the contribution of skeletal motion and interpersonal dynamics, allowing us to examine whether InsAT can capture socially meaningful cues without relying on object context.

For evaluation, we integrate InsAT into the SKP backbone and compare its performance against the same backbone trained without InsAT, ensuring a controlled and fair comparison. Under this setting, InsAT achieves an accuracy of 89.9%, outperforming the baseline SKP model, which attains 87.1% accuracy. All results are obtained from our own re-implementation and experiments, eliminating potential confounds from differences in training protocols or evaluation procedures.

These results demonstrate that InsAT remains effective in densely populated scenes, where action understanding primarily depends on person–person relationships rather than explicit human–object interactions.

E Text supervision

E.1 Caption generation

We use GPT-4o (API version 2024-08-06) (OpenAI, 2024a) to generate four types of textual descriptions for each action label. For each class, we

Evaluation dataset	Kinetics-200/400 (pose only)	Kinetics-200/400 (pose + object)	NTU-RGBD 60/120	HMDB51 (trained on Kinetics-400)	Kenetics-ObjShared	Kenetics-ObjExt
Optimizer	Stochastic Gradient Descent					
Number of epochs	20		30	150	100	
Batch size	100					
Maximum input frames	300		200	300		
Learning rate	0.1	0.07	0.25	0.6	0.1	0.6
Weight decay	0.00001		0.0001	0.00005	0.00001	
LR scheduler	Linear decay					
Keypoint scaling	[0.8, 1.2]					
Keypoint shift	[-0.2, 0.2]					
Keypoint rotate (°)	[-10, 10]					
Keypoint flip ratio	0.5					
Temporal crop window	100					
Temporal FPS drop	5		3	5		

Table 11: Hyperparameter settings for ZS training across different evaluation datasets. For HMDB51, we optimize the sum of the contrastive loss in Eq. 3 and a label cross-entropy loss with a 1:1 weight, which results in a higher learning rate.

Method	Obj.	Kinetics-400	HMDB51
MS-G3D (Liu et al., 2020b)	✗	45.1	-
PoseConv3D (Duan et al., 2022)	✗	47.7	69.7
SKP (Hachiuma et al., 2023)	✗	50.3	70.9
ProtoGCN (Liu et al., 2025a)	✗	51.9	-
SKP w/o InsAT†	✓	58.0[a]	71.2[b]
SKP w/ InsAT	✓	**59.3**[a]	**72.3**[b]

[a] Skeleton: PPNv2 (Sekii, 2021); Object: PPNv2.
[b] Skeleton: HRNet (Sun et al., 2019); Object: D-FINE-X (Peng et al., 2024).
For settings without object keypoints (✗), all models use HRNet as the skeleton detector.

Table 12: Comparison of supervised action recognition accuracy (Top-1 Acc. [%]) between InsAT and state-of-the-art keypoint-based methods. † denotes results obtained from our experiments for fair comparison.

Method	Ratio of scale		
	1/20	1/10	1
w/o InsAT	26.3	34.8	58.0
w/ InsAT	**27.7**	**36.9**	**59.3**

Table 13: Data efficiency analysis. Top-1 Acc. [%] for different dataset scales (Kinetics-400 = 1). SKP (Hachiuma et al., 2023) models are trained with and without InsAT. Results for the 1/20 and 1/10 scales are averaged over three runs.

provide the prompt template described in Sec. 3.3 together with the corresponding action label name as input to the language model. The generated captions are designed to capture complementary aspects of the action, including motion characteristics, involved entities, and interaction context, using four caption types: **Sum** (action summary), **Sub** (subject behavior), **Oth** (relevant others), and **Obj** (relevant object). Below, we present representative examples of the generated captions for selected action classes from Kinetics-400, illustrating the diversity and consistency of the language supervision used during training.

playing tennis

- **Sum**: "The person plays tennis, moving across the court, hitting a ball with a racket, and responding to the opponent's shots."
- **Sub**: "The person grips a tennis racket, swings their arm to hit the ball, shifts weight between feet, and maintains a balanced stance with knees slightly bent."
- **Obj (tennis racket)**: "The person grips the tennis racket in their hand to hit the ball and return it to the opponent."
- **Oth (sports ball)**: "The person aims to hit the sports ball with the racket to send it over the

Text Encoder	Acc. [%]
CLIP ViT-L/14 (Radford et al., 2021)	35.31
MPNet (Song et al., 2020)	36.94
text-embedding-3-large (OpenAI, 2024b)	38.80

Table 14: Text encoder comparison. ZS Acc. [%] on Kinetics-200 (160/40).

net during the game."

- **relevant object (shoe)**: "The person wears shoes on their feet to provide grip and support while moving quickly on the tennis court."
- **Oth**: "Two players interact by hitting a tennis ball back and forth across the net. One player positions themselves closer to the net while the other stays near the baseline."

dribbling basketball

- **Sum**: "The person dribbles a basketball, repeatedly bouncing it on the ground while moving around the court, maintaining control and coordination."
- **Sub**: "The person alternates hands to bounce the basketball, maintaining a low stance with bent knees, while their eyes are forward and feet are positioned for quick movement."
- **relevant object (sports ball)**: "The person dribbles the sports ball with their hand, bouncing it repeatedly on the ground as part of the action."
- **Obj (shoe)**: "The person wears shoes on their feet to provide grip and support while moving and dribbling the basketball."
- **Obj (bench)**: "The bench is situated near the playing area and can be used for resting or observing the game."
- **Oth**: -

shooting basketball

- **Sum**: "The person shoots a basketball towards the hoop, aiming to score, often involving jumping and precise hand-eye coordination."
- **Sub**: "The person bends their knees, aligns their feet shoulder-width apart, raises their arms, and flicks their wrist to release the basketball towards the hoop."
- **Obj (sports ball)**: "The person holds the basketball in their hands, preparing to shoot it towards the hoop."
- **Obj (shoe)**: "The person wears athletic shoes on their feet to provide support and traction while moving on the court."

Method	Acc. (%)
w/o InsAT	34.8
InsAT with template-based captions	36.4
InsAT with GPT-4o-generated captions	**36.9**

Table 15: Comparison of template-based and detailed caption designs under a supervised setting using Kinetics-1/10. Template-based captions use LLM output only for object enumeration and multi-person identification.

- **Obj (bench)**: "The bench is located near the court, providing a place for players t rest between plays."
- **Oth**: -

E.2 Effect of the Text Encoder

Tab. 14 shows that *text-embedding-3-large* (OpenAI, 2024b) achieves the highest accuracy, likely due to its larger embedding dimension (3,072 vs. 768 for others), providing richer textual representations.

E.3 Template-based vs. LLM-generated Detailed Captions

Tab. 15 compares the proposed detailed captions with a simpler template-based caption design under a supervised setting using 1/10 of the Kinetics training data. In the template-based setting, the LLM is used only to determine the involved objects and whether multiple people are involved, and the final captions are constructed from fixed templates. Specifically, the captions are constructed as follows: **Sum:** {*Action Name*}. (e.g., *playing tennis*.) **Sub:** {*Action Name*}. (e.g., *playing tennis*.) **Obj:** {*Action Name*} + Object. (e.g., *playing tennis*, involving the tennis racket.) **Oth:** {*Action Name* + multiple people}. (e.g., *playing tennis*, involving multiple people.)

Template-based captions already improve performance over removing InsAT, and GPT-4o-generated detailed captions further achieve the highest accuracy. This suggests that richer caption content is beneficial, but more importantly, the gains stem from instance-level language–visual alignment in InsAT rather than from simply adding textual information.

E.4 Preliminary verification with an open-source LLM

To verify that InsAT does not rely on a specific proprietary model, we generated instance-level captions using a small open-source LLM (Llama3.2-

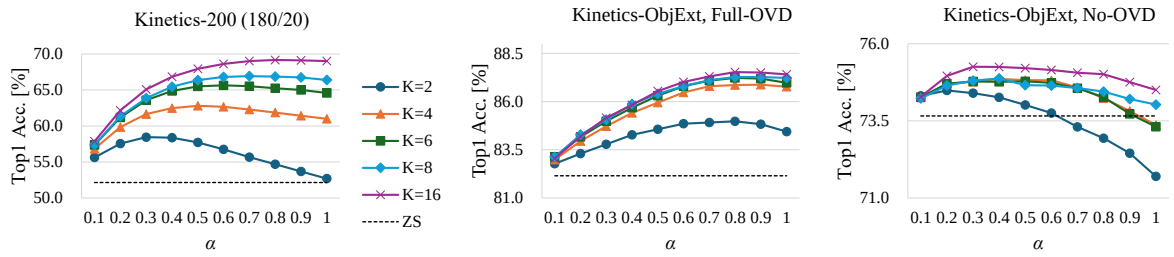

Figure 6: Sensitivity analysis of α in IVA. Left: Kinetics200 (180/20 split). Middle: Kinetics-ObjExt with OVD extension (Full-OVD). Right: Kinetics-ObjExt without OVD extension (No-OVD).
Note that the y-axis ranges differ to better illustrate the effect of α in each setting.

3B-Instruct (Grattafiori et al., 2024)) in a limited setting. The model was able to produce meaningful descriptions for different caption types.

For example, for the action *dribbling basketball*, the model generated instance-level descriptions such as: **Sum:** "The person dribbles a ball on the floor using their fingertips while moving." **Sub:** "The person bends their knees and lowers their center of gravity while bouncing the ball." **Obj (sports ball):** "The ball repeatedly bounces as it is controlled by the hands."

However, the model occasionally generated object tokens outside the predefined detector vocabulary, indicating that additional filtering is required. Overall, this suggests that open-source LLMs can generate meaningful instance-level descriptions without relying on proprietary models.

F Additional Experiments for Z2F Adaptation

F.1 Effect of α in instance-level visual adaptation (IVA)

Fig. 6 analyzes the effect of the weight parameter α (eq. (6)), which controls the contribution of FS visual evidence in IVA. Across all settings, we observe a consistent trend that the optimal value of α shifts toward larger values as the number of FS samples K increases. This trend suggests that visual prototypes constructed from a larger number of labeled samples become more reliable and can be more strongly emphasized during adaptation.

We further analyze this K-dependent trend of α on the Kinetics-ObjExt dataset, which consists of action classes involving objects that are strongly associated with each action. The middle and right plots in Fig. 6 compare the effect of α with and without OVD-based object extension (Full-OVD and No-OVD; see Tab. 4, respectively). When Full-OVD is applied, larger values of α consistently yield better performance, indicating that FS visual evidence becomes more informative when enriched with open-vocabulary object keypoints. In contrast, without OVD extension, the optimal α tends to remain smaller, reflecting the reduced reliability of visual prototypes constructed without object-level information. Overall, these results indicate that detecting action-relevant objects makes FS visual prototypes more reliable, allowing them to be emphasized more strongly during zero-to-few-shot adaptation.

Method	K=2	K=4	K=8	K=16
IVA	0.0	0.0	0.0	0.0
Full-FT	-29.0	-18.4	-16.4	-18.7

Table 16: Change in Top-1 accuracy (%) on the pretraining dataset (Kinetics-400) after few-shot adaptation. Values indicate the difference between accuracy before and after adaptation; negative values denote performance degradation.

F.2 Catastrophic Forgetting Analysis

Since IVA does not update model parameters during adaptation, catastrophic forgetting is not expected to occur. In contrast, fine-tuning-based approaches may suffer from performance degradation on the pretraining data.

To verify this, we evaluate the change in Top-1 accuracy on the pretraining dataset (Kinetics-400) before and after few-shot adaptation. Tab. 16 reports the accuracy differences.

Full-FT shows a substantial performance drop of approximately 16–29 percentage points across shot settings, indicating severe catastrophic forgetting. In contrast, IVA preserves the pretrained knowledge while achieving strong few-shot performance (Tab. 2), demonstrating its robustness against catastrophic forgetting.