Learning reusable concepts across different video understanding tasks

Simone Alberto Peirone Francesca Pistilli Antonio Alliegro Tatiana Tommasi Giuseppe Averta
Politecnico di Torino
simone.peirone@polito.it

Abstract

Our comprehension of video streams depicting human activities is naturally multifaceted: in just a few moments, we can grasp what is happening, identify the relevance and interactions of objects in the scene, and forecast what will happen soon, everything all at once. To endow autonomous systems with such holistic perception, learning how to correlate concepts, abstract knowledge across diverse tasks, and leverage tasks synergies when learning novel skills is essential. In this paper, we introduce Hier-EgoPack, a unified framework able to create a collection of task perspectives that can be carried across downstream tasks and used as a potential source of additional insights, as a backpack of skills that a robot can carry around and use when needed. Project Page: sapeirone.github.io/hier-egopack.

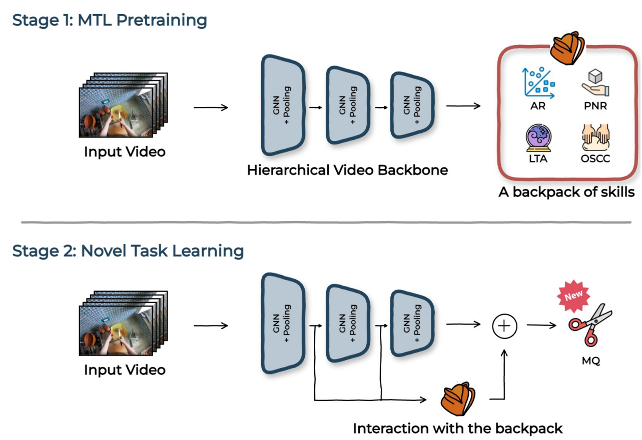

Figure 1. **Novel task learning in egocentric vision.** In the MTL Pretraining stage, Hier-EgoPack learns to solve a set of video understanding *support tasks*. Then, the knowledge from these tasks is collected in the form of prototypes (backpack of skills) and reused to foster the learning process of a *novel task*.

1. Introduction

Our daily activities are extremely complex and diverse, yet humans have the extraordinary ability to perceive, reason, and plan their actions almost entirely from visual inputs. For instance, when observing someone at a kitchen counter with a pack of flour and a jug of water, we can infer they are kneading dough (*reasoning about current activity*). We might predict that their next step will involve mixing flour with water (*reasoning about the future*) to obtain the dough (*reasoning about implications*), maybe with the ultimate goal of preparing some bread (*reasoning about long-range activities*). Although natural for humans, replicating this holistic understanding in artificial intelligence remains a major challenge. Most existing work tackles human activity understanding via task-specific models [14, 16, 17], neglecting shared reasoning patterns across tasks. Although multitask learning (MTL) offers some synergy, it struggles with negative task interference and lacks flexibility for novel tasks [6]. We propose a paradigm shift: rather than just sharing information, systems should abstract and reuse task-specific knowledge to foster future skill learning. EgoPack [9] demonstrated this idea for egocentric videos by learning a set of reusable concepts from multiple *support* tasks to enhance *novel* ones. However, egocentric videos cover a wide range of tasks spanning diverse temporal scales, from actions lasting a few seconds to long-range activities.

In this paper, we introduce Hier-EgoPack, an enhanced version of EgoPack, specifically designed to maximize positive interaction across tasks with different temporal granularity, while still using a unified architecture and minimizing task-specific weights and tuning for novel task learning (Fig. 1). Our hierarchical model captures both fine and coarse temporal patterns and introduces a novel Temporal Distance Gated Convolution (TDGC) layer to reason over temporal dependencies. We validate our approach on the large-scale Ego4D [4] dataset, showing improved performance and positive interaction between tasks knowledge.

2. Related works

Concepts Learning. Concepts Learning covers a broad range of methods that learn an information bottleneck between the input data and the output of a desired task. Concept Bottleneck Models (CBM) [5] learn individual units that represent the activation of specific concepts present in the input. The concepts taxonomy may come from do-

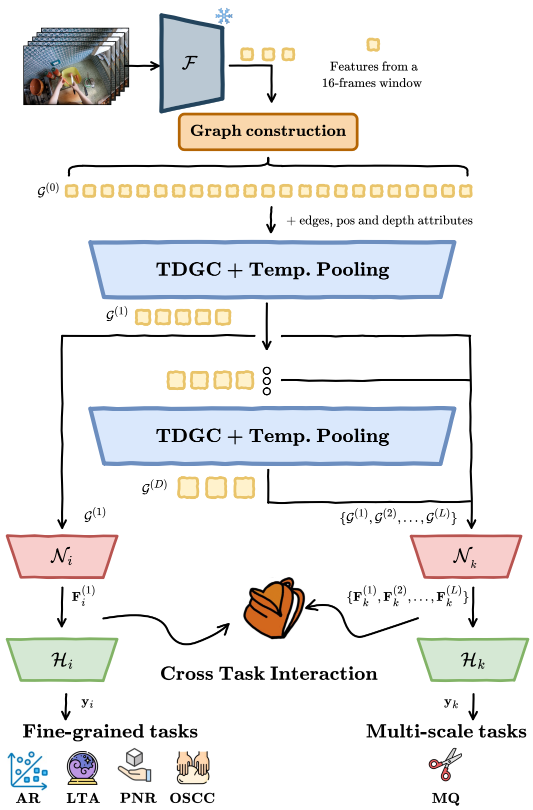

Figure 2. **Overview of Hier-EgoPack.** The video is processed as a graph by the *hierarchical temporal backbone* $\mathcal{M}_t$, shared by all the tasks. The node embeddings from different tasks are collected in the backpack for cross-task interaction.

main knowledge [5], language models [8, 15] or obtained without any supervision [11]. Learning in the high-level concepts space may improve generalization across tasks and domains [1, 3] and produce more interpretable models [12, 15]. In video understanding, few works explored post-hoc concepts-based interpretability [2] or used concepts learning to disentangle static and dynamic features in action recognition models [10]. EgoPack [9] extends concepts-based learning across a diverse set of video understanding tasks that require different reasoning skills, collecting a set of concepts that encode how each task would "perceive" the same action from its specific "perspective".

3. Method

We address a cross-task interaction setting, in which an egocentric vision model is trained to reuse previously acquired knowledge from a set of *support tasks* to foster the learning process of any *novel task*. This work introduces Hier-EgoPack, a unified temporal architecture able to model tasks with different temporal granularity and strong *sense of time*, *i.e.* reasoning on the order of events in a video.

Novel task learning with previous knowledge. A task $\mathcal{T}$ in egocentric vision is defined as a mapping between a video $\mathcal{V}$ and an output space $\mathcal{Y}$. In classifications tasks, such as Action Recognition, this mapping assigns a trimmed video segment v_i to its corresponding label $y_i \in \mathcal{Y}$. Differently, action localization tasks process the entire video $\mathcal{V}$ and predict a set of temporally grounded activities, each described by its start and end timestamps and the corresponding action label: $\mathcal{T} : \mathcal{V} \rightarrow \{(t_i^s, t_i^e, y_i)\}_i$.

Our approach streamlines the processing for different tasks by feeding the temporal backbone $\mathcal{M}_t$ with untrimmed input videos and aligning the output to the downstream task at a later stage, which is a crucial design choice to enable knowledge sharing across different tasks. We follow a two-stages training process:

- **Stage 1: multi-task pretraining** on a set of K support tasks to learn generalizable representations;
- **Stage 2: novel task learning**, in which the model adapts to a *novel task* $\mathcal{T}_{K+1}$ without access to support task labels.

The key idea is to capture and reuse semantic cues shared across tasks. For example, recognizing object state changes can inform action recognition, as actions like *cutting* imply change, while others like *moving* may not.

3.1. A unified architecture for Video Understanding

We represent a video $\mathcal{V}$ as a sequence of N fixed-length segments with associated features $\mathbf{x} = \mathbf{x}_1, \ldots, \mathbf{x}_N$, extracted via a pretrained video encoder $\mathcal{F}$ (e.g., EgoVLP [7]). The video can be interpreted as a temporal graph $\mathcal{G} = (\mathbf{X}, \mathcal{E}, \mathbf{pe})$, where $\mathbf{X} \in \mathbb{R}^{N \times D}$ is a matrix of features of the graph nodes, edge $e_{ij} \in \mathcal{E}$ connects nodes i and j if their temporal distance is below τ and the attribute $\mathbf{pe} \in \mathbb{R}^N$ encodes the *timestamp* (in seconds). Modeling videos as graphs enables reasoning over temporal relations via message passing and to frame multiple egocentric tasks under the unified architecture of Hier-EgoPack. The proposed architecture is built on three components:

1. a *shared temporal backbone* $\mathcal{M}_t$, built with TDGC layers and subsampling for hierarchical temporal reasoning;
2. *task-specific projection necks* $\mathcal{N}_k$ to map node embeddings to the features space of task $\mathcal{T}_k$;
3. *task-specific heads* $\mathcal{H}_k$ for task-specific outputs.

Let $\mathcal{G}^{(0)}$ represent the initial graph of the input video $\mathcal{V}$, where each node's position $\mathbf{pe}$ is initialized to the midpoint of the corresponding video segment. Starting from $\mathcal{G}^{(0)}$, the backbone iteratively updates the graph through L stages:

$$\mathcal{M}_t : \mathcal{G}^{(0)} \rightarrow \{\mathcal{G}^{(1)}, \mathcal{G}^{(2)}, \ldots, \mathcal{G}^{(L)}\},$$

Each stage applies TDGC layers and temporal subsampling (mean/max pooling) to progressively enlarge the temporal extent of the nodes. Edge connections are updated based on scaled node timestamps ($\times 2^l$ at stage l). The number of stages L is task-dependent: single-stage for fine-grained tasks (e.g., AR, OSCC), and multi-stage for long-range temporal tasks. The architecture is shown in Fig. 2.

Temporal Distance Gated Convolution (TDGC). Each stage of the *temporal* backbone $\mathcal{M}_t$ is built as a stack of N_l GNN layers, which we call Temporal Distance Gated Convolution (TDGC). These layers are designed to preserve and encode the temporal sequence of information, capturing the relative past and future dependencies between nodes. More specifically, given two nodes i and j at layer l, we compute s_{ij} and $\mathbf{w}_{ij}$ as:

$$s_{ij} = \mathtt{sign}(\mathbf{pe}^{(l)}_{[i]} - \mathbf{pe}^{(l)}_{[j]}), \quad \mathbf{w}_{ij} = \mathtt{MLP}(|\mathbf{pe}^{(l)}_{[i]} - \mathbf{pe}^{(l)}_{[j]}|).$$

These two factors are used to re-weight the contribution of each neighbor node j in the aggregation step, as follows:

$$\mathbf{x}^{'}_j = \mathtt{MLP}\left(\mathbf{x}^{(l)}_j\right) = \phi(\mathbf{W}^T_n \mathbf{x}^{(l)}_j + \mathbf{b}_n),$$
$$\mathbf{x}^{(l+1)}_i = \mathbf{W}^T_r \mathbf{x}^{(l)}_i + \underset{j \in \bar{\mathcal{N}}(i)}{\mathtt{mean}} \left(s_{ij}(\mathbf{w}_{ij} \odot \mathbf{x}^{'}_j)\right) + \mathbf{b}_r,$$

where $\mathbf{x}^{(l)}_i$ are the features of the node i at layer l, $\bar{\mathcal{N}}(i)$ is the set of neighbors of node i, $\mathbf{W}_n$, $\mathbf{W}_r$ and $\mathbf{b}_n$, $\mathbf{b}_r$ are learnable weights and biases respectively.

3.2. Task-specific components

The temporal backbone $\mathcal{M}_t$ is shared across all downstream tasks and provides task-agnostic temporal reasoning over streams of fixed-length video segments. Each task $\mathcal{T}_k$ has its own projection neck $\mathcal{N}_k$, a two-layer MLP that maps backbone outputs to the task's feature space: $\mathbf{X}^{(l)}_k = \mathcal{N}_k\left(\mathbf{X}^{(l)}\right)$ with $\mathcal{N}_k : \mathbb{R}^D \to \mathbb{R}^D$. For tasks with known temporal boundaries (*e.g.*, AR), we align node embeddings to task annotations:

$$\mathbf{F}^{(l)}_{k,[i]} = \mathtt{align}(\mathbf{X}^{(l)}_k, s_i, e_i) = \underset{j:\ s_i < \mathbf{p}^{(l)}_{[j]} < e_i}{\mathtt{mean}} \mathbf{X}^{(l)}_{k,[j]},$$

where $\mathbf{F}^{(l)}_{k,[i]}$ are the task-specific features of segment v_i for task $\mathcal{T}_k$. For action localization, which operate over the full video, alignment is unnecessary and we directly use $\mathbf{X}^{(l)}_k$.

3.3. Building a backpack of reusable skills

To solve the *novel task* $\mathcal{T}_{K+1}$, the naive approach would be to finetune the model, adding new task-specific neck $\mathcal{N}_{K+1}$ and head $\mathcal{H}_{K+1}$ and possibly updating the temporal backbone $\mathcal{M}_t$. However, this approach may forget previously acquired knowledge. Instead, we explicitly model the perspectives of the *support tasks*, learned during the MTL pretraining step, as a set of task-specific prototypes that can be accessed by the novel task (Fig. 3). We collect these task-specific prototypes from videos annotated for action recognition, as human actions can be seen as the common thread behind the different tasks. We forward these action samples through the temporal backbone, align them based on the AR annotations and project their features using the task-specific necks $\mathcal{N}_k$ of each task to obtain the task-specific features $\mathbf{F}_k$, each row capturing that task's "perspective" on a given segment. We then aggregate features by action label (a unique verb-noun pair) to form the prototypes set $\mathbf{P}^k = \{\mathbf{p}^k_0, \mathbf{p}^k_1, \ldots, \mathbf{p}^k_P\} \in \mathbb{R}^{P \times D}$ for each task $\mathcal{T}_k$, where P is the number of unique verb-noun pairs and D is the feature dimension. These prototypes are frozen and serve as a compact summary of each task's learned representation—a reusable knowledge base for novel tasks, like a *backpack of skills* that the model can carry over.

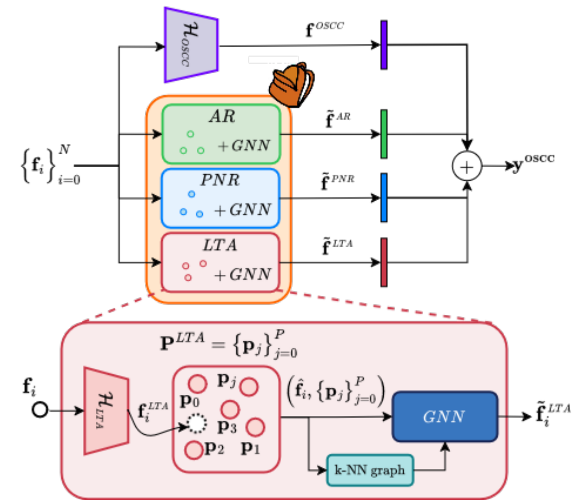

Figure 3. **Learning a novel task with a backpack.** In this *Cross-Tasks Interaction* phase, the network can peek at these different task-perspective to enrich the learning of the novel task.

3.4. Learning a novel task with a backpack

To solve a novel task $\mathcal{T}_{K+1}$, we pass the output graphs from the temporal backbone through all the task-specific necks to obtain features $\mathbf{X}_k$. These features act as queries to retrieve the closest task prototypes in $\mathbf{P}^k$ via k-NN search in the feature space. Each query and its neighboring prototypes form a graph-like structure, where message passing is applied using M layers of SAGE convolution to iteratively refine the task-specific features. At each layer m, we update the features $\mathbf{X}^{(m)}_{k,[i]}$:

$$\mathbf{X}^{(m+1)}_{k,[i]} = \mathbf{W}^{(m)}_r \mathbf{X}^{(m)}_{k,[i]} + \mathbf{W}^{(m)} \cdot \underset{\mathbf{p}^k_j \in \bar{\mathcal{N}}(i)}{\mathtt{mean}} \mathbf{p}^k_j,$$

where $\mathbf{p}^k_j \in \bar{\mathcal{N}}(i)$ is the set of *activated prototypes* for the given task, and $\mathbf{W}^{(m)}_r$, $\mathbf{W}^{(m)}$ are learnable projections. This refinement is applied at each backbone stage l, updating only the task features—not the prototypes—to preserve the original learned perspectives. The final refined features

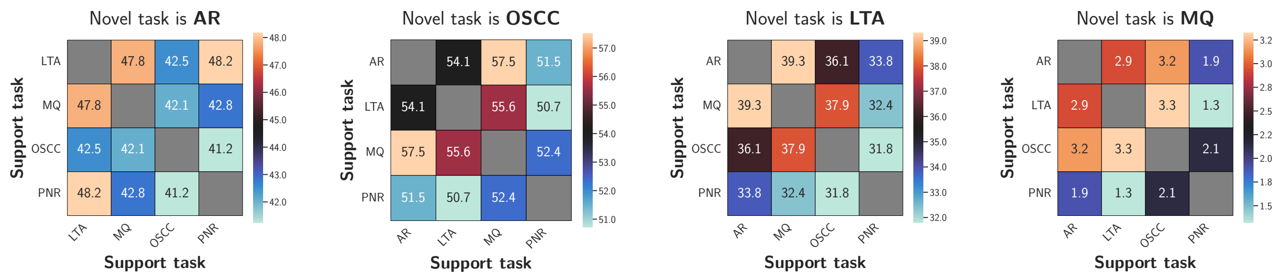

Figure 4. **Activations consensus for different *novel tasks*.** Activations consensus between two *support tasks* is defined as the percentage of their prototypes corresponding to the same label activated by the two tasks.

Table 1. Hier-EgoPack on Ego4D Human-Object Interaction (HOI) and Moment Queries (MQ) tasks.

	AR Top-1 (%)		OSCC Acc.	LTA ED (↓)		PNR Err.	MQ mAP
Method	Verb	Noun	(%)	Verb	Noun	(↓)	(%)
Ego4D Baselines [4]	22.18	21.55	68.22	0.746	0.789	0.62	6.03
EgoT2s [13]	23.04	23.28	72.69	0.731	0.769	**0.61**	N/A
EgoPack [9]	25.10	31.10	71.83	0.728	0.752	**0.61**	N/A
Single Task	26.93	33.50	75.22	0.728	0.752	0.62	20.2
MTL	26.31	33.90	74.79	0.730	0.754	0.62	18.5
Hier-EgoPack	**27.30**	**34.65**	**75.60**	**0.725**	**0.741**	**0.61**	**21.0**

Single Task uses the same hierarchical GNN-based architecture to model all tasks. *Multi-Task Learning (MTL)* uses hard parameter sharing to jointly learn all tasks.

$\tilde{\mathbf{X}}_k^{(l)}$ are aligned, if needed, to produce $\tilde{\mathbf{F}}_k^{(l)}$. We evaluate different fusion strategies to integrate the novel task with the perspectives gained from the previous tasks. In *features-level* fusion, we average the task-specific features for the novel task $\mathbf{F}_{K+1}$ with the *refined* perspectives from the previous tasks $\tilde{\mathbf{F}}_k$. In *logits-level* fusion, we keep a set of separate heads, one for each support task, feed the features $\tilde{\mathbf{F}}_k$ to each head separately and sum their outputs. Intuitively, this approach allows each task to cast a vote on the final prediction, based on its perspective on the same video segment.

4. Experiments

We validate our approach on Ego4D [4], a large scale dataset with 3.6k hours of egocentric videos capturing unscripted daily-life human activities, focusing on five Ego4D benchmarks that cover different temporal granularities. *Fine-grained tasks* focus on short-term understanding of the video, usually a few seconds long, and include: *Action Recognition (AR)*, *Object State Change Classification (OSCC)*, *Point of No Return (PNR)*, *Long Term Anticipation (LTA)*. Other tasks may require both short and long term understanding of the input video. Among these, we analyze an action localization task, *i.e.*, *Moment Queries (MQ)*.

Quantitative results. We show the main results in Table 1, comparing our approach with the Ego4D baselines [4], the task-translation framework EgoT2 [13] and EgoPack [9]. We observe that the task prototypes in Hier-EgoPack provide a comprehensive and easy-to-access abstraction of the model's learned knowledge, enabling the extraction of relevant insights tailored to the specific sample and task, exhibiting superior performance.

Activation consensus across tasks. In this section, we analyze how Hier-EgoPack leverages knowledge abstractions from the *support tasks* (collected in the form of prototypes) to aid the learning of a *novel task*. Specifically, we visualize the *activated prototypes* (*i.e*. the set of prototypes each *support task* looks at) during the interaction process of Hier-EgoPack across different novel tasks and quantify task activation consensus, a measure of the complementarity among support tasks in aiding the learning of a novel task. We define the *activations consensus* as the degree to which different tasks activate prototypes corresponding to the same label for a given sample of the *novel task*. A low consensus suggests that the support tasks capture more diverse cues, *i.e*. different tasks activate different prototypes, whereas a high consensus indicates that activations are more coherent across tasks. We show in Fig. 4 the average activation consensus for different novel tasks. Fine-grained tasks, *e.g*. AR, LTA and OSCC, have higher average consensus compared to MQ. We attribute this difference to the implementation of the interaction process for these two groups of tasks. In fine-grained tasks, the interaction process is applied on the sample-level aligned features, while we use node-level features in MQ which may correspond to background or poorly discriminating regions of the video. However, the low average activations consensus and high diversity in prototypes' activations across tasks shows how Hier-EgoPack is effectively integrating different perspectives for the MQ task.

5. Conclusions

We present Hier-EgoPack, an holistic video understanding model that enables knowledge sharing between egocentric vision tasks with different temporal granularity. Our work emphasizes the importance of prior knowledge and task perspectives in learning novel tasks, focusing on how task-specific knowledge is represented and utilized. Moreover, through our proposed unified architecture, we demonstrate that leveraging diverse task perspectives in egocentric vision, even across varying temporal granularity, leads to more comprehensive and human-like video understanding.

Acknowledgments

This study was carried out within the FAIR - Future Artificial Intelligence Research and received funding from the European Union Next-GenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR) – MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.3 – D.D. 1555 11/10/2022, PE00000013). This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them. Antonio Alliegro and Tatiana Tommasi also acknowledge the EU project ELSA - European Lighthouse on Secure and Safe AI (grant number 101070617).

References

[1] Kaidi Cao, Maria Brbic, and Jure Leskovec. Concept learners for few-shot learning. In *ICLR*, 2021. 2

[2] Kowal et al. Understanding video transformers via universal concept discovery. *arXiv preprint arXiv:2401.10831*, 2024. 2

[3] Marco Fumero, Florian Wenzel, Luca Zancato, Alessandro Achille, Emanuele Rodolà, Stefano Soatto, Bernhard Schölkopf, and Francesco Locatello. Leveraging sparse and shared feature activations for disentangled representation learning. *NeurIPS*, 2023. 2

[4] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *CVPR*, 2022. 1, 4

[5] Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *ICML*, 2020. 1, 2

[6] Iasonas Kokkinos. Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In *CVPR*, 2017. 1

[7] 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. In *NeurIPS*, 2022. 2

[8] Tuomas Oikarinen, Subhro Das, Lam M Nguyen, and Tsui-Wei Weng. Label-free concept bottleneck models. In *ICLR*, 2023. 2

[9] Simone Alberto Peirone, Francesca Pistilli, Antonio Alliegro, and Giuseppe Averta. A backpack full of skills: Egocentric video understanding with diverse task perspectives. In *CVPR*, 2024. 1, 2, 4

[10] Rui Qian, Shuangrui Ding, Xian Liu, and Dahua Lin. Static and dynamic concepts for self-supervised video representation learning. In *ECCV*, 2022. 2

[11] Simon Schrodi, Julian Schur, Max Argus, and Thomas Brox. Concept bottleneck models without predefined concepts. *arXiv preprint arXiv:2407.03921*, 2024. 2

[12] Andong Tan, Fengtao Zhou, and Hao Chen. Explain via any concept: Concept bottleneck model with open vocabulary concepts. In *ECCV*, 2024. 2

[13] Zihui Xue, Yale Song, Kristen Grauman, and Lorenzo Torresani. Egocentric video task translation. In *CVPR*, 2023. 4

[14] Shen Yan, Xuehan Xiong, Anurag Arnab, Zhichao Lu, Mi Zhang, Chen Sun, and Cordelia Schmid. Multiview transformers for video recognition. In *CVPR*, 2022. 1

[15] Yue Yang, Artemis Panagopoulou, Shenghao Zhou, Daniel Jin, Chris Callison-Burch, and Mark Yatskar. Language in a bottle: Language model guided concept bottlenecks for interpretable image classification. In *CVPR*, 2023. 2

[16] Chen-Lin Zhang, Jianxin Wu, and Yin Li. Actionformer: Localizing moments of actions with transformers. In *ECCV*, 2022. 1

[17] Zeyun Zhong, David Schneider, Michael Voit, Rainer Stiefelhagen, and Jürgen Beyerer. Anticipative feature fusion transformer for multi-modal action anticipation. In *WACV*, 2023. 1