Semantically Guided Representation Learning For Action Anticipation

Anxhelo Diko[1], Danilo Avola*[1], Bardh Prenkaj*[2], Federico Fontana[1], and Luigi Cinque[1]

[1] Sapienza University of Rome, Computer Science Department
{diko,avola,fontana.f,cinque}@di.uniroma1.it
[2] Technical University of Munich, Chair of Responsible Data Science
bardh.prenkaj@tum.de

Abstract. Action anticipation is the task of forecasting future activity from a partially observed sequence of events. However, this task is exposed to intrinsic future uncertainty and the difficulty of reasoning upon interconnected actions. Unlike previous works that focus on extrapolating better visual and temporal information, we learn action representations that are aware of their semantic interconnectivity based on prototypical action patterns and contextual co-occurrences, proposing the novel Semantically Guided Representation Learning (S-GEAR) framework. S-GEAR learns visual action prototypes and leverages language models to structure their relationship, inducing semanticity. To gather insights on S-GEAR's effectiveness, we test it on four action anticipation benchmarks, obtaining improved results compared to previous works: +3.5, +2.7, and +3.5 absolute points on Top-1 Accuracy on Epic-Kitchen 55, EGTEA Gaze+ and 50 Salads, respectively, and +1.4 on Top-5 Recall on Epic-Kitchens 100. We further observe that S-GEAR effectively transfers the geometric associations between actions from language to visual prototypes. Finally, S-GEAR opens new research frontiers in anticipation tasks by demonstrating the intricate impact of action semantic interconnectivity. Code: https://github.com/ADiko1997/S-GEAR.

Keywords: Action Anticipation · Semantic Interconnection · Prototype Learning · Geometric Associations

1 Introduction

Anticipating future actions is a key attribute of human intelligence when navigating the world. This remarkable skill translates directly to advanced computer vision applications such as self-driving cars [13, 27] or wearable assistants [53, 56], enabling safer navigation and better user experience [32]. Recent developments in deep learning techniques have boosted the research on video understanding, reaching remarkable milestones on tasks like action recognition [4, 16, 21, 24, 30, 51]. Models related to action recognition can extrapolate

* Equal second author contribution.

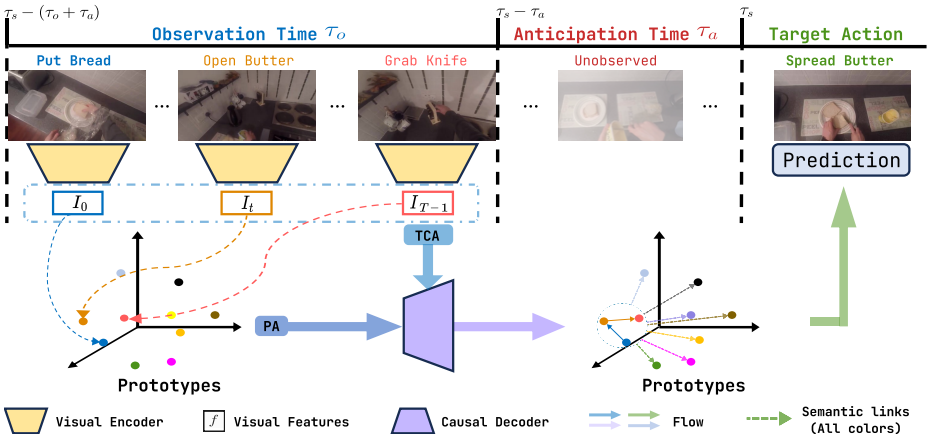

Fig. 1: We propose learning action prototypes that encode typical action representations and meaningful semantic interconnections. The model leverages these prototypes to enhance the network encodings of observed actions and to forecast upcoming ones.

essential spatiotemporal information from videos of isolated actions and correctly classify them. However, real-world applications operate in dynamic environments where actions are interconnected. For instance, imagine a self-driving car observing pedestrians. Predicting their intent to cross the street requires analyzing how observed dynamics relate to likely future events. This temporal misalignment between observation and future target introduces a challenge for recognition models proving them insufficient and shifting the attention towards action anticipation [13, 15, 32, 47, 53, 56]. This emerging research area focuses on enabling vision systems to predict future activity by observing ongoing events.

In trying to deal with the implications of action anticipation, previous methods extended recognition models with sequence units like LSTMs [1, 13, 32, 53] and causal transformers [15, 53, 56]. The success of these approaches relies on the ability of the network to extract and maintain key visual information from videos over time. However, these methods have limitations. They cannot explicitly model the semantic connectivity between actions beyond the immediate video context, which is critical when dealing with co-occurring action sequences. According to cognitive sciences, semantic interconnectivity is fundamental for anticipating the future [11]. It helps structure our knowledge by associating actions with objects, intentions, and likely outcomes. This enables us to draw on past experiences to form reliable predictions even in unseen situations. Inspired by such observations, we raise the question: *Is it possible to encode meaningful semanticity between action representations in a vision model?*

In pursuit of answering our question, we propose the ***S**emantically **G**uided R**E**presentation Le**AR**ning* (**S-GEAR**) framework (see Fig. 1). S-GEAR tackles action anticipation with a novel representation learning approach oriented by two fundamentals of actions semantic connectivity: **(1)** understanding the typical patterns of individual actions, and **(2)** modeling relationships between

actions based on contextual co-occurrences [6,34,50]. For **(1)**, S-GEAR learns a set of visual action prototypes. Each prototype encodes specific action patterns, capturing typical movements or gestures that define and distinguish action categories, reducing reliance on the specific appearance details of individual videos. Conversely, for **(2)**, building semantic relationships between actions solely from videos is challenging. First, it requires processing long action sequences to include enough context and defining co-occurrence relationships. Second, actions are usually not represented equally in videos [7, 9], hardening the modeling of under-represented action relationships. S-GEAR circumnavigates these issues by exploiting language models known to extract inter-concept semantic relationships [35, 42] – i.e., effectively tackle **(2)**. Specifically, S-GEAR creates language prototypes based on action labels and transfers their inherent semantic connectivity to visual prototypes without aligning them directly. To achieve this, S-GEAR uses a new loss function that enables visual prototypes to maintain visual cues, such as object and movement patterns, while encoding semanticity by mimicking the geometric associations between actions from language.

S-GEAR uses an encoder-decoder transformer architecture to learn prototypes and encode semantic relationships between actions. The encoder consists of a standard *Vision Transformer* (ViT) [10, 46] for visual context, while the decoder is a *Causal Transformer* (CT) [46,49] which models temporal causality. These structures are interconnected through two novel computational blocks, namely *Temporal Context Aggregator* (TCA) and *Prototype Attention* (PA) for, respectively, causality enhancement and semanticity promotion. Lastly, S-GEAR appends a classification head that produces future class probabilities based on the decoder's output and geometric association with the visual prototypes.

To assess S-GEAR's performance, we conduct extensive experiments on two egocentric video datasets, Epic-Kitchens [7,9] (both Epic-Kitchens 55 & 100 versions) and EGTEA Gaze+ [55]. Moreover, we evaluate S-GEAR on an exocentric dataset, namely 50 Salads [44], to demonstrate its versatility in long-term dense anticipation. We show that S-GEAR improves over the current state-of-the-art in most scenarios. We also conduct ablation studies highlighting the usefulness of the semantic connectivity between actions that S-GEAR incorporates.

This paper's contributions are fourfold. **(1)** We present S-GEAR, a novel prototype learning framework for action anticipation leveraging action interconnectivity. **(2)** We introduce a novel approach to map semanticity from language to vision without direct alignment between modalities. **(3)** We conduct extensive experiments on two egocentric datasets and an exocentric one to highlight S-GEAR's versatility in different action anticipation scenarios (i.e., egocentric vs. exocentric and short-term vs. long-term). **(4)** We showcase the benefits of S-GEAR w.r.t. its counterparts that do not rely on semantic relationships.

2 Related Work

Action anticipation predicts future actions before they occur in video clips and is well explored both in third-person (exocentric) videos [5, 20, 40, 43, 47],

and first-person (egocentric) videos [8, 12–15, 23, 32, 37, 38, 53, 55], due to its applicability on autonomous agents and wearable assistants [15, 27, 48]. Funari et al. [13] introduce RU-LSTM, a model with two LSTMs and a modality attention component. Osman et al. [31] integrate RU-LSTM into SlowFast. Qi et al. [32] enhance LSTMs with Self-Regulated Learning (SRL). Dessalene et al. [8] use hand-object contact representations for action anticipation. Xu et al. [53] employ curriculum learning. Roy et al. [37] predict final goals for near-future anticipation. Liu et al. [23] store long-term action prototypes for richer short-term representations. Girdhar et al. [15] propose AVT, combining ViT and causal transformer, paving the way for [14,51,56]. Manousaki et al. [26] proposed VLMAH, the first vision-language approach for action anticipation, combining textual features of long-term past textual information and visual short-term past information by concatenating them. It relies on multi-branch LSTMs as their anticipation architecture. Unlike previous works, S-GEAR considers the semantic relationship between action representations [11] by using vision and language prototypes to guide the model's training process semantically.

Vision-Language alignment relies on effectively aligning concepts between vision and language in a unified representation space. Typically achieved through contrastive training of modality encoders [18, 33, 54], these methods use vision-language pairs for encouraging proximity between corresponding visual and text embeddings. Zhai et al. [54] utilize contrastive learning to align text encoder representations with a frozen pre-trained vision model. Radford et al. [33] introduce CLIP, training separate encoders for text and images and aligning representations through contrastive loss. Ma et al. [25] extend CLIP to videos, employing multi-grained contrastive learning. Advancements include cross-modal fusion architectures using a cross-modality encoder for text and visual inputs [3, 17]. Unlike previous works, S-GEAR only translates the geometric association between action prototypes from language to vision without shifting spaces.

Prototype Learning involves creating characteristic "prototypes" of labeled data samples. Initially dominant in few-shot learning for novel class prediction [41, 45], this strategy now successfully encodes spatial and temporal patterns in domains such as video semantic segmentation [22] and action recognition [28].

3 Method

We propose S-GEAR for action anticipation. S-GEAR discerns essential spatiotemporal signals and understands the semantic relationships between actions. It contains a neural network architecture tailored for understanding spatiotemporal video sequences and a learning policy that guides the network semantically to map out the interconnections between actions.

Task Formulation. Action anticipation involves predicting an action category for an event starting at time τ_s, observing a video segment V_o within the interval $[\tau_s-(\tau_o+\tau_a);\tau_s-\tau_a]$ [9]. Here, τ_o and τ_a denote the observation and anticipation periods set specifically to the dataset.

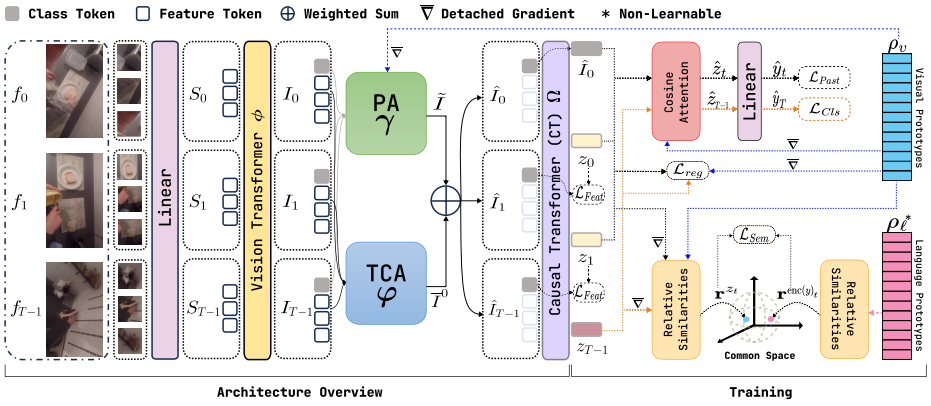

Fig. 2: S-GEAR processes frame sequence patches and creates input token sequences S_t. ViT ϕ encodes S_t into intermediate features I_t. PA γ and TCA φ process I_t, merging outputs into semantically enhanced causal features $\hat{I}_t$. Class tokens pass through the CT decoder Ω, predicting future features z_t. The features z_t and the proposed prototypes are trained for action anticipation ($\mathcal{L}_{Cls}$) and semantic relation encodings ($\mathcal{L}_{Sem}$). The network is also regularized for accurate future representations ($\mathcal{L}_{Feat}$) and correct past action classification ($\mathcal{L}_{Past}$). Finally, a distance loss ($\mathcal{L}_{reg}$) is applied to z_t.

3.1 Proposed Architecture

S-GEAR processes a sequence of video frames and produces a set of features that can accurately describe the subsequent action. To achieve this, as shown in Fig. 2, S-GEAR employs an architecture composed of (1) a visual encoder for extracting feature vectors from the input frames; (2) the Temporal Context Aggregator (TCA) module designed to incorporate detailed temporal context from past to current observations; (3) the Prototype Attention (PA) block, which combines visual features with learned prototypes and (4) the Causal Transformer (CT) decoder responsible for predicting future representations.

Visual Encoder. Upon receiving a video segment $V_o = \{f_0, ..., f_{T-1}\}$ of length T, S-GEAR relies on ViT [10] as the visual encoder ϕ to obtain spatial features from each frame. ViT splits each frame into P non-overlapping patches of equal size, which are then flattened and transformed into a series of feature tokens $S_t \in \mathbb{R}^{P\times d}$ corresponding to frame $f_t \in V_o$. Here, d represents the token dimensionality. Then, to preserve the spatial context, learnable positional encodings are added to S_t. Additionally, the so-called "*class token*" CLS_t, which captures the global context of frame f_t, is prepended to S_t. The transformer blocks then act on S_t, generating visual features $I_t = \phi(S_t)$ with the same dimension as S_t.

Temporal Context Aggregator (TCA) and Prototype Attention (PA). In this stage, I_t passes through two specialized units to enhance temporal causality and semantic interconnections between actions. Inspired by the left-to-right causal transformer [46], we craft TCA φ, the first unit, to effectively transfer comprehensive context from the past to the current frame representation. In TCA, unlike standard causal blocks that mainly rely on the global representa-

tions I_t^0 (class token), we consider all feature patches. Thus, given global and local representations I of the frames, we obtain causal intermediate features $\overline{I} \in \mathbb{R}^{T\times(P+1)\times d}$ where each $\overline{I}_t \in \overline{I}$ is enhanced with detailed contextual information from past frames. Contrarily, the second unit PA, denoted as γ, operates parallel to the TCA on I_t^0 and the visual prototypes. Specifically, PA aggregates information from selected visual prototypes upon feature similarity to I_t^0, promoting semantic relation encoding between actions as inferred from the different prototypes. We rely on the attention mechanism using I_t^0 as queries and the visual prototypes as keys and values to produce semantically enhanced feature sets $\tilde{I} \in \mathbb{R}^{T\times d}$. We then combine $\tilde{I}$ and $\overline{I}^0$ as a weighted sum $\hat{I} = \lambda\overline{I}^0 + (1-\lambda)\tilde{I}$ (λ is learnable). We point the reader to Appendix A.1-2 for details on TCA and PA.

Temporal Decoder. We rely on an autoregressive Causal Transformer (CT) decoder Ω, as presented in [15, 53, 56] to analyze $\hat{I}$ from $t = 0$ to $t = T-1$ and generate a set of features that describes the likely future. Similar to the visual encoder, we add learnable positional encodings to $\hat{I}$ to preserve the temporal context. Afterward, we feed the embedded features with positional encodings to the decoder blocks, built upon the masked multi-head self-attention [15]. Thus, Ω generates a new sequence $\zeta = \Omega(\hat{I})$ s.t. $\forall t$, $z_t \in \zeta$ represents the future features of $\hat{I}_t$ after observing all the past ones including itself. For $t = T-1$, z_t represents the future action happening τ_a seconds after the observed sequences.

3.2 Semantic Guiding Policy

We exploit vision/language prototypes and a common communication space between them to facilitate a semantic-based guiding policy for action anticipation.

Prototypes. We aim to translate semantic relationships from language-based action concepts to the visual domain. Thus, we define two sets of prototypes. The first, defined as the language prototypes $\rho_\ell \in \mathbb{R}^{K\times d}$ (where K is the number of action classes), is extracted by encoding action labels composed of verb and noun combination using the "Sentence Transformer" proposed in [35] as language encoder. These prototypes serve as the reference space for learning actions "semantic connectivity" [42]. The second, defined as the visual prototypes $\rho_\upsilon \in \mathbb{R}^{K\times d}$, ensures that S-GEAR remains in the visual domain and effectively preserves characteristic visual patterns. Such prototypes are learnable and initialized from typical action samples encoded from the proposed architecture trained for action recognition. We exploit ρ_υ to encode visual action representations and inherit the semanticity from ρ_ℓ. Refer to Appendix A.3 for initialization details.

Common Communications Space. To translate action relationships from language to vision without shifting domains, we define a common space where vision and language representations co-exist and are compared via their relative associations w.r.t. the prototypes. In more detail, given an action *visual encoding* $z_t \in \zeta$, we compute its relative representation by comparing it against all elements in ρ_υ using a similarity function: i.e., $\mathbf{r}^{z_t} = \{r_1^{z_t}, \ldots, r_K^{z_t}\}$ s.t. $r_k^{z_t} = cos(z_t, \rho_\upsilon[k])$ for each action class $k \in \{1, \ldots, K\}$. Similarly, we compute the relative representation of a *language encoding* $\text{enc}(y)_t$ – i.e., the language encoding of the action label at time t – against the prototypes in ρ_ℓ as

$\mathbf{r}^{\text{enc}(y)_t} = \{r_1^{\text{enc}(y)_t}, \ldots, r_K^{\text{enc}(y)_t}\}$ s.t. $\mathbf{r}_k^{\text{enc}(y)_t} = cos(\text{enc}(y)_t, \rho_\ell[k])$ for all action classes $k \in \{1, \ldots, K\}$. Now, we ensure that each k-th entry in $\mathbf{r}^{z_t}$ and $\mathbf{r}_k^{\text{enc}(y)_t}$ represents the geometric association with the k-th class prototype in the language/vision domain. Hence, we can directly compare these two representations based on their relative position in their original vector spaces.

3.3 Training

To train the model, for each labeled action segment, we sample a clip preceding it and ending exactly τ_a seconds before the start of the action. We pass the clip through S-GEAR to obtain z_t and then optimize it to learn semantically and visually meaningful prototypes for action anticipation.

Prototype Learning. Learning prototypes aim to establish a visual latent space where predefined semantic connections describe actions by "aligning" the latent space topology defined by ρ_v with ρ_ℓ. To do so, we calculate the relative positions $\mathbf{r}^{z_t}$ and $\mathbf{r}^{\text{enc}(y)_t}$, which we use to define the semantic loss in Eq. 1.

$$\mathcal{L}_{Sem} = \left|\mathbf{r}^{z_t} - \mathbf{r}^{\text{enc}(y)_t}\right|. \tag{1}$$

During optimization, the prototypes in ρ_v will be refined to represent relative relationships between actions akin to those inferred from the language space. Additionally, to guide S-GEAR push the action z_t towards the prototype of the same class k (i.e., $\rho_v[k]$) and avoid divergences, we add a lasso regularization to $\mathcal{L}_{Sem}$ as in Eq. 2.

$$\begin{aligned} \mathcal{L}_{reg} &= \left|\left|z_t - \rho_v[k]\right|\right|_2^2 \\ \mathcal{L}_{Sem} &= \mathcal{L}_{Sem} + \mathcal{L}_{reg}. \end{aligned} \tag{2}$$

Thus, while shaping the visual latent space geometry defined by ρ_v (Eq. 1), we enforce action representations to fall close to their visual prototype (Eq. 2).

Anticipation Training. Besides prototype learning, we train S-GEAR for action anticipation by optimizing the cross-entropy loss between the predicted class label $\hat{y}_T$ and the ground truth y_T. $\hat{y}_T$ is obtained from the encoded action representation and its relative position w.r.t. the visual prototypes. More specifically, for the action representation z_{T-1}, we calculate $\mathbf{r}^{z_{T-1}}$. Since $\mathbf{r}^{z_{T-1}}$ contains values in $[-1, +1]$, we transform them into probabilistic weights using softmax. Now, we aggregate all the prototype vectors into a single representation, $\overline{z}_{T-1} \in \mathbb{R}^d$, according to the obtained weights (see Eq. 3).

$$\overline{z}_{T-1} = \text{softmax}(\mathbf{r}^{z_{T-1}}) \cdot \rho_v. \tag{3}$$

To jointly learn the action representation and its exact collocation in the visual space w.r.t. the prototypes, we perform a weighted sum as in Eq. 4:

$$\hat{z}_{T-1} = \sigma(\alpha) z_{T-1} + (1 - \sigma(\alpha))\overline{z}_{T-1}, \tag{4}$$

where σ is a sigmoid function, and α is a learnable scalar. Such operations are represented as Cosine Attention in Fig. 2. Lastly, we feed $\hat{z}_{T-1}$ through a linear

layer and softmax its output to obtain $\hat{y}_T$. We calculate the cross-entropy loss (Eq. 5) between the ground truth and the predicted action class.

$$\mathcal{L}_{Cls} = -\sum_{i}^{K} y_T^i \log(\hat{y}_T^i). \tag{5}$$

Additionally, inspired by [15, 47], we leverage the causality of the decoder Ω. Here, we use any true class label for the past frames and minimize the cross-entropy on past label predictions (Eq. 6). Notice that the predicted label $\hat{y}_t$ is produced following the same reasoning described above for $\hat{y}_T$ (see Eq. 3, 4).

$$\mathcal{L}_{Past} = -\sum_{t=0}^{T-2}\sum_{i}^{K} y_{t+1}^i \log(\hat{y}_{t+1}^i). \tag{6}$$

To produce faithful future features, we minimize the distance between the predicted future frame features and the actual ones:

$$\mathcal{L}_{Feat} = \sum_{t=0}^{T-2} \big|\big|\hat{I}_{t+1} - z_t\big|\big|. \tag{7}$$

The overall loss function used to train S-GEAR is a weighted sum of all the individual losses: $\mathcal{L}_{tot} = \lambda_1\mathcal{L}_{Sem} + \lambda_2\mathcal{L}_{Cls} + \lambda_3\mathcal{L}_{Past} + \lambda_4\mathcal{L}_{Feat}$.

4 Experiments

4.1 Datasets and Metrics

The EPIC-Kitchens 55 (EK55) dataset [7] is a medium-scale first-person cooking dataset comprising 432 videos from 32 different individuals and approximately 40,000 segments. It encompasses 92 verbs and 272 object classes, resulting in 2,747 action classes. Additionally, we use the train and validation splits provided in [13]. Our model's performance on EK55 is evaluated using Top-1/5 Accuracy at $\tau_a = 1$s, following prior works [13, 15, 32, 53].
The EPIC-Kitchens 100 (EK100) dataset [9] is a substantial extension of EK55, encompassing 700 videos from 37 individuals in 45 diverse kitchens. It comprises ∼90,000 activity segments spanning 495 training, 138 validation, and 67 test videos. EK100 offers a richer representation of cooking activities through its broader range of verbs (97), objects/nouns (300), and action classes (4,053). To assess model performance on EK100, we employ the class aware mean Top-5 Recall [9, 15, 53] metric at $\tau_a = 1$s.
The EGTEA Gaze+ dataset (EG) [55] includes 28 hours of first-person cooking videos from 32 subjects across 86 sessions, covering 7 tasks. The dataset contains 10,325 activity instances, categorized into 19 verbs, 51 objects, and 106 activity classes. To evaluate our model, we employed Top-1 Accuracy on split 1 for $\tau_a = 0.5$s [2, 15, 56] and Top-5 Accuracy averaged across all three splits to evaluate overall performance for $\tau_a = 1$s [13, 23, 32].

The 50 Salads dataset (50S) [44] comprises 50 exocentric videos featuring salad preparation activities performed by 25 different actors and categorized into 17 activity classes. We assess our model using mean Top-1 Accuracy across the 5 official splits following previous works [13, 32]. Unlike other benchmarks, the 50S offers a dense action anticipation challenge with variable observation and anticipation times. Specifically, for a given video segment in input, τ_a goes from 10% to 50% of the video's duration while τ_o is set to 20% or 30%.

4.2 Implementation Settings

Visual Encoder. S-GEAR employs the ViT Base (ViT-B) architecture as its visual encoder with a patch size of 16×16. It comprises 12 transformer blocks, feature dimension 768, and operates with 12 attention heads. We set each frame size for input dimensions to 384×384 for the EK55/100 datasets and 224×224 for the EG and 50S datasets. Besides the default encoder, following prior works [15, 32, 53], we show that S-GEAR can also be used with other backbones like TSN and irCSN using pre-extracted features as in [13] and [15], respectively.
Intermediate Stage. Our intermediate processing stage, crucial for linking the visual encoder's output to the causal transformer decoder, consists of 2 TCA blocks and 1 PA block. Note that when replacing ViT with other backbones, we omit TCA blocks. This is because, without ViT's detailed local patches, the architecture essentially becomes a standard causal transformer.
Causal Transformer Decoder. For EK55/100 datasets, we employ a 6-layer causal transformer decoder with 4 heads and a dimensionality of 2048 to process the observed context and predict future events. For the EG dataset, we reduce the number of layers to 2. Meanwhile, for the 50S dataset, an 8-layer decoder with eight heads and the same dimensionality is used.
Observation. For EK100, we set the observation time, τ_o, to 15s, processing video segments at 1fps. For EK55 and EG, we maintain the same processing rate but reduce τ_o to 10s. In contrast, for the 50S, we align with [1, 19, 32] and adopt observation rates of 20% and 30% for each input sequence, with 0.25fps.
Training Settings. We employ different training strategies for each dataset. For EK100, EK55, and EG, we use an SGD optimizer with a momentum of 0.9 and weight decay of 1e-5, processing mini-batches of 3. The learning rates are 1e-4 for EK55/100 and 4.75e-4 for EG, all with cosine scheduling and warmup of 10, 20, and 5 epochs, respectively. The total training durations are 50 epochs for EK100, 35 for EK55, and 10 for EG. In contrast, for 50S, we opt for AdamW optimizer with parameters β_1, β_2 set to 0.9, 0.999, a weight decay of 1e-4, and a learning rate 5e-6. This setup also includes cosine scheduling and 20 warmup epochs, with the model training for 100 epochs on mini-batches of 2. Finally, we run our experiments on an RTX4090 and 2×V100 GPUs.

4.3 Baselines

We compare against RU-LSTM [13], SRL [32], AVT [15], DCR [53], MeMViT [51], RAFTformer [14], HRO [23], AFFT [56], TempAgg. [39], Imagination [52]

Table 1: Experiments on Epic-Kitchens 55/100 for τ_a=1s.

	Model	Encoder Initialization		Top-1 Acc.	Top-5 Acc.
RGB	RU-LSTM [13]	TSN	IN1K	13.1	30.8
	SRL [32]	TSN	IN1K	/	31.7
	AVT [15]	TSN	IN1K	13.1	28.1
	DCR [53]	TSN	IN1K	13.6	30.8
	S-GEAR (ours)	TSN	IN1K	15.6	32.8
	AVT [15]	irCSN	IG65M	14.4	31.7
	DCR [53]	irCSN	IG65M	14.4	<u>34.0</u>
	S-GEAR (ours)	irCSN	IG65M	**16.2**	33.1
	AVT [15]	ViT-B	IN21K	12.5	30.1
	S-GEAR (ours)	ViT-B	IN21K	15.8	**34.5**
Obj	RU-LSTM [13]	FRCNN	IN1K	10.0	29.8
	DCR	FRCNN	IN1K	<u>11.5</u>	**30.5**
	S-GEAR (ours)	FRCNN	IN1K	**12.45**	<u>30.4</u>
Flow	RULSTM	TSN	IN1K	8.7	21.4
	DCR	TSN	IN1K	<u>8.9</u>	<u>22.7</u>
	S-GEAR (ours)	TSN	IN1K	**10.8**	**25.8**

(a) Unimodal results on EK55 validation set. Models are grouped based on backbone initialization and modality.

	Model	Encoder	Initialization	Verb	Noun	Action
RGB	DCR [53]	TSM	K400	32.6	32.7	16.1
	MeMViT [51]	MViTv2-16	K400	32.8	33.2	15.1
	RAFTformer [14]	MViTv2-16	K400	<u>33.3</u>	35.5	17.6
	AVT [15]	TSN	IN1K	27.2	30.7	13.6
	DCR [53]	TSN	IN1K	31.0	31.1	14.6
	S-GEAR (ours)	TSN	IN1K	25.8	29.8	14.9
	AVT [15]	ViT-B	IN21K	30.2	31.7	14.9
	S-GEAR (ours)	ViT-B	IN21K	31.1	37.3	18.3
	RAFTformer-2B [14]	MViTv2-16&24	K400&700	**33.8**	<u>37.9</u>	<u>19.1</u>
	S-GEAR-2B (ours)	ViT-B×2	IN21K	32.7	**37.9**	**19.6**
Obj	AVT [15]	FRCNN	IN1K	18.0	<u>24.3</u>	8.7
	DCR	FRCNN	IN1K	**22.2**	24.2	<u>9.7</u>
	S-GEAR (ours)	FRCNN	IN1K	<u>20.8</u>	**28.6**	**11.4**
Flow	AVT [15]	TSN	IN1K	20.9	16.9	6.6
	DCR	TSN	IN1K	**25.9**	<u>17.6</u>	**8.4**
	S-GEAR (ours)	TSN	IN1K	<u>21.5</u>	**18.2**	<u>7.9</u>

(b) Unimodal results on EK100 validation set. Models are grouped based on modality and backbone initialization except RaftFromer-2B and S-GEAR-2B, which use multiple backbones.

Model	Modalities	Top-1 Acc.	Top-5 Acc.
RU-LSTM	RGB+Obj+Flow	15.3	35.3
TempAgg.	RGB+Obj+Flow	15.1	35.6
Imagination	RGB+Obj+Flow	15.2	35.4
SRL	RGB+Obj+Flow	/	35.5
AVT+ [15]	RGB+Obj	16.6	37.6
HRO	RGB+Obj+Flow	/	37.4
DCR	RGB+Obj+Flow	<u>19.2</u>	<u>41.2</u>
S-GEAR (ours)	RGB+Obj+Flow	**22.7**	**43.2**

(c) Multimodal results on EK55 validation set.

Model	Modalities	Validation			Test		
		Verb	Noun	Action	Verb	Noun	Action
RU-LSTM	RGB+Obj+Flow	27.8	30.8	14.0	25.3	26.7	11.2
TempAgg.	RGB+Obj+Flow+HOI+Audio	23.2	31.4	14.7	21.8	30.8	12.6
AVT+	RGB+Obj	28.2	32.0	15.9	25.6	28.8	12.6
AVT++	RGB+Obj+Flow	/	/	/	26.7	32.2	16.7
DCR	RGB+Obj+Flow	/	/	18.3	/	/	17.3
AFFT	RGB+Obj+Flow+HOI+Audio	22.8	34.6	18.5	20.7	31.8	14.9
S-GEAR (ours)	RGB+Obj	29.5	<u>37.8</u>	18.9	<u>25.9</u>	32.0	14.7
S-GEAR-2B (ours)	RGB+Obj	**30.5**	**38.4**	<u>19.6</u>	25.5	<u>31.7</u>	<u>15.3</u>
S-GEAR-4B (ours)	RGB+Obj	<u>30.2</u>	37.0	**19.9**	**26.6**	**32.6**	**15.5**

(d) Multimodal results on EK100 validation and test sets. HOI refers to Hand-Object-Interaction. Deemphasized works are ensembles of multiple models.

and more to ensure a fair comparison. Bold and underlined values in the tables illustrate the best and second-best results, respectively.

4.4 Unimodal Comparison

Table 1 (a), (b) provides unimodal results on EK55 and EK100 datasets, ensuring a fair comparison of S-GEAR against baselines. In EK55 (Table 1 (a)), in RGB, S-GEAR demonstrates a point improvement of 1.1 on Top-5 Acc. (vs. the second-best SRL) and 2.0 on Top-1 Acc. (vs. the second-best DCR) for the TSN features. Regarding the irCSN features, S-GEAR surpasses DCR by 1.8 points in Top-1 Acc. while trailing it on Top-5 Acc. by 0.9. Using the ViT-B backbone, S-GEAR surpasses AVT by 3.3 (Top-1) and 6.4 (Top-5). For the object modality, we use Faster R-CNN features for a fair comparison, obtaining 0.9 Top-1 Acc. improvement, yet falling behind on Top-5 by 0.1. Finally, S-GEAR yields 1.9 (Top-1) and 3.1 (Top-5) point gains for the flow modality over prior works.

Table 1 (b) details the results of the EK100 benchmark. S-GEAR competes with MeMViT [51] and RAFTformer [14] with MViTv2-16 backbone for RGB. S-GEAR demonstrates improvements in Top-5 Recall for actions (3.2 over

Table 2: Experiments on the EGTEA Gaze+ and the third-person dataset 50-Salads.

Model	Modalities	Top-1 Acc. ($\tau_o = 0.5$s)	Top-5 Acc. ($\tau_a = 1$s)
RU-LSTM [13]	RGB+Flow	/	66.40
DCR [53]	RGB+Flow	/	67.9
SRL [32]	RGB+Flow	/	70.7
HRO [23]	RGB+Flow+Obj	/	<u>71.5</u>
AVT [15]	RGB	43.0	/
AFFT [56]	RGB+Flow	42.5	/
S-GEAR (ours)	RGB	**45.7**	**71.9**

(a) EG results regarding Top-1 Acc. for $\tau_o = 0.5$s and Top-5 Acc. for $\tau_a = 1.0$s.

$\tau_o \rightarrow$	20%				30%			
$\tau_a \rightarrow$	10%	20%	30%	50%	10%	20%	30%	50%
RU-LSTM [13]	22.2	17.8	12.7	08.3	22.3	15.5	10.8	05.2
CNN model [2]	21.2	19.0	16.0	09.9	29.1	20.1	17.5	10.9
Grammar-based [36]	24.7	22.3	19.8	12.7	29.7	19.2	15.2	13.1
Uncertainty [1]	28.9	22.4	19.9	12.8	29.1	20.5	15.3	12.3
RNN [2]	30.1	25.4	18.7	13.5	30.8	17.2	14.8	09.8
Time-Cond. [19]	32.5	27.6	21.3	**16.0**	35.1	<u>27.1</u>	**22.1**	<u>15.5</u>
SRL [32]	<u>37.9</u>	**28.8**	<u>21.3</u>	11.1	<u>37.5</u>	24.1	17.1	09.1
S-GEAR (ours) [32]	**41.0**	<u>28.5</u>	**21.5**	<u>15.3</u>	**41.0**	**27.8**	<u>21.4</u>	**16.7**

(b) 50S results on dense action anticipation. (*Percentages are w.r.t. the video duration*).

MeMViT, 0.7 over RAFTformer) and nouns (4.1 over MeMViT, 1.8 over RAFTformer). While trailing slightly on verbs, unlike its competitors (Kinetics-400), S-GEAR performs well without spatiotemporal initialization. Additionally, we formed S-GEAR-2B by late-fusing two S-GEAR versions with ViT-B backbones (input 224×224 and 384×384). Despite being a late fusion (compared to RAFTformer-2B's joint architecture), S-GEAR-2B achieves a 0.5 improvement in action — all without spatiotemporal initialization. Furthermore, S-GEAR demonstrates strong performance compared to AVT and DCR across modalities, achieving overall gains of 3.4 and 2.2 for action Top-5 Recall in RGB. Contrarily, S-GEAR shows gains of 1.7 and 4.4 (actions, nouns) on object modality and slightly trails DCR on verbs. Finally, S-GEAR remains competitive even in the flow modality. This comparison verifies S-GEAR's contribution to training effective anticipation models aware of action semantic interconnections.

4.5 Comparison with the SOTA

Epic-Kitchens. Previous approaches often utilize cross-modality ensembling [9, 56] or joint training [13, 56] for multimodal evaluation on these benchmarks. Ensembling S-GEAR across modalities, we observe significant gains. On EK55 (Table 1 (c)), late-fusing our models (RGB+Obj+Flow) yields a boost of 3.5 (Top-1 Acc.) and 2.0 (Top-5 Acc.) over prior work. Similarly, on EK100 (Table 1 (d)), late-fusing RGB modalities with object features leads to 1.4 improvement in action Top-5 Recall. Finally, though we report EK100 test set results (Table 1 (d)) and obtain competitive performances, it is crucial to note that leaderboard rankings often rely on large-scale external data or fusion across diverse models (i.e., the de-emphasized models on Table 1 (d)). This makes the test set less effective for comparing the core strengths of models [53]. We point the reader to Appendix B.2 for details on our specific ensembling weights.

EGTEA Gaze+. We evaluate S-GEAR on two task on EG (Table 2 (a)). The first includes Top-1 Acc. on split-1 for $\tau_a = 0.5$ where we achieve 2.5 point improvement compared to previous work. The second includes the average Top-5 Acc. across the three splits at $\tau_a = 1s$ where we surprisingly improve on HRO with 0.4 points despite using only the RGB modality with our ViT-B backbone.

Table 3: Ablation study (Top-5 Recall) on EK100 validation set.

Settings	TCA	Sem	PA	Verb	Noun	Action
(1) Baseline	-	-	-	30.5	32.6	15.2
(2) Sem	-	✓	-	30.7	35.7	17.8
(3) TCA	✓	-	-	31.0	33.9	16.7
(4) PA + Sem	-	✓	✓	**32.0**	<u>36.2</u>	<u>18.0</u>
(5) TCA + PA (ρ_ℓ)	✓	-	✓	30.6	33.3	17.4
S-GEAR	✓	✓	✓	<u>31.1</u>	**37.3**	**18.3**

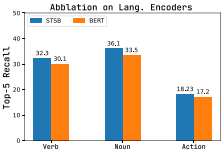

Fig. 3: Ablation on language encoders.

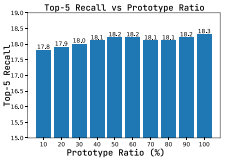

Fig. 4: Performance according to used prototype ratio.

50 Salads. Our dense anticipation experiments on the 50S (Table 2 (b)) show S-GEAR's potential for long-term and exocentric tasks. It outperforms competitors in 5/8 scenarios, with Top-1 Accuracy gains of up to 3.5, despite not being tailored for long-term anticipation like Time-Cond. [19].

4.6 Ablation Study

We analyze the importance of S-GEAR's components to justify our design choices. Specifically, we investigate **(a)** the impact of architectural and training elements, **(b)** the significance of encoding semantic action relationships, **(c)** the number of prototypes for defining relative action positions, and **(d)** S-GEAR's performance for different anticipation time τ_a.

(a) We use EK100 (RGB) to evaluate the impact of architectural components and our prototype learning strategy (see Table 3). We use a baseline (1) comprising a ViT-B encoder, a casual decoder, and a linear classification head similar to AVT [15]. On top of this baseline, we switch on/off each component that comprises S-GEAR: i.e., (2) the prototype learning with semantic guidance, including the cosine attention block on the classification head, (3) the TCA block, and (4) the PA block. Note that the PA block needs prototypes; thus, in the table, we toggle the semantic column as well. While all strategies improve over (1), (2) has the most impact, adding up to 2.6 points on Top-5 Recall for action classes. Such improvements are caused by the ability of the prototype learning strategy to cluster actions that co-occur frequently. The network then uses this proximity to encode action representations aware of their exact collocation through the cosine attention block, taking hints that the next probable action can be found in its proximity in the latent space. Note that the effectiveness of $\mathcal{L}_{Sem}$ is strictly related to the regularization term $\mathcal{L}_{reg}$ since without it the network representations will converge on a different space compared to visual prototypes. Finally, to motivate our choice of learning visual prototypes ρ_v rather than directly using language prototypes ρ_ℓ, we rely on (5), which includes all the architecture components except the prototype learning strategy. Instead, action representations are directly aligned with fixed ρ_ℓ. This resulted in decreased performance compared to S-GEAR. While ρ_ℓ captures semantic structure, we believe it lacks the scene information crucial for accurate anticipation, such as motion and visual

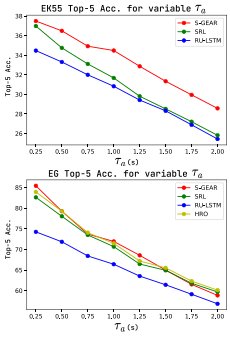

Fig. 5: EK55 (top) and EG (bottom) Top-5 Acc. for variable τ_a.

Fig. 6: Qualitative example of observed actions (Observation), the target activity (Target), S-GEAR's Top-5 predictions, and the Top-5 semantically similar actions with the observed sequence based on language encoding (Semantic).

context. S-GEAR overcomes this limitation by learning its visual prototypes, allowing them to adapt to the specific visual cues relevant to the task.

(b) S-GEAR builds on the principle that semantically similar actions often co-occur, making semantic relationship encoding crucial. To ablate on the importance of such relationships, we leverage two Sentence Transformer variations from HuggingFace: "bert-large-nli-max-token" (BERT) and "stsb-mpnet-base-v2" (STSB). These models share a similar architecture but differ in training data size, with STSB being better at semantic relation extraction. Fig. 3 shows that S-GEAR performs better with STSB-generated prototypes, highlighting that modeling accurate semantic interconnections gives better results.

(c) While prototypes are valuable, they introduce a computational cost due to their large matrix size (e.g., in EK100 with 4053 actions). In this regard, we investigate the possibility of approximating an action's relative position by comparing it to only a subset of prototypes. Experiments on EK100, using varying portions of visual prototypes (see Fig. 4) show that we can achieve good results using only a fraction (i.e., 17.8 Top-5 Recall at 10% vs. 18.3 at 100%) of the prototypes while significantly reducing the number of computations.

(d) Finally, we evaluate the performance of S-GEAR for variable τ_a. We expect the performance to drop as τ_a increases. Hence, we experiment on EK55 and EG training S-GEAR with $\tau_a = 0.25$ and test its autoregressive capabilities by increasing τ_a up to 2s at inference time. We report the results in Fig. 5. While the performance drop is highlighted as $\tau_a \to \infty$, we notice that S-GEAR performs better than previous works on EK55. On the other hand, on EG, S-GEAR remains highly competitive, slightly trailing HRO and SRL with $\tau_a > 1$s.

4.7 Qualitative Results

Fig. 6 demonstrates S-GEAR's ability to anticipate future actions on the EG dataset, using $\tau_a = 1$s and $\tau_o = 32$s. Alongside S-GEAR's Top-5 predictions, we

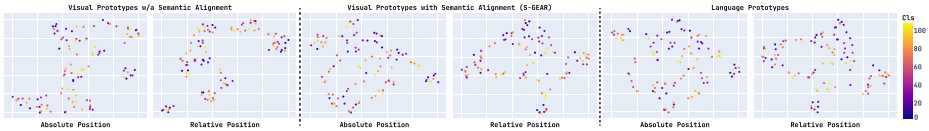

Fig. 7: Illustration (via UMAP [29]) of the absolute and relative position of visual action prototypes w/a semantic alignment (left), after semantic alignment (middle), and language prototypes (right) for EG.

include the Top-5 semantically similar language prototypes given the observed action sequence. These examples reveal the connection between anticipation and semantics, suggesting that the two are aligned. On the other hand, the last row example also highlights divergences emphasizing S-GEAR's room for semantic improvement. To further investigate the semantic alignment between S-GEAR and language prototypes, in Fig. 7, we illustrate the geometric association learned by S-GEAR prototypes (middle) on EG, comparing it with its initial values (left) and the language prototypes (right) both in terms of absolute and relative positions. The latter is determined using cosine similarity to compare each prototype against all others. S-GEAR's prototypes demonstrate a latent space topology closer to the language prototypes than its counterpart w/o semantic alignment in terms of absolute and relative position. Such phenomenon indicates that S-GEAR can reason upon the semantic connectivity between actions, projecting contextually similar ones closer in latent space. However, S-GEAR's topology is slightly different since visual cues influence inter-prototype distances. We point the reader to Appendix B for more experimental details.

5 Conclusion

We presented S-GEAR, a novel framework for action anticipation that leverages semantic interconnectivity between actions. S-GEAR learns visual and language prototypes that encode typical action patterns and their relationships based on contextual co-occurrences. S-GEAR transfers the geometric associations between actions from language to vision without direct alignment, creating a common communication space. S-GEAR employs a transformer-based architecture incorporating temporal context aggregation and prototype attention to enhance the action representations and predict future events. We evaluate S-GEAR on four action anticipation benchmarks, showing improved results compared to previous works. We also demonstrate that we can effectively encode semantic relationships between actions, opening new research frontiers in anticipation tasks. While S-GEAR shows promising results, its limitations include the lack of an in-built multimodal mechanism and semantic interconnections that explicitly account for occurrence order. Accounting for co-occurrence orders can reduce future prediction uncertainty, narrowing the scope of future action to those likely to follow the observed sequence. We will address these limitations in future work.

Acknowledgments

We would like to express our sincere gratitude to Professor Giovanni Maria Farinella and Professor Antonino Furnari from the University of Catania for their invaluable guidance, support, and insightful feedback.

References

1. Abu Farha, Y., Gall, J.: Uncertainty-aware anticipation of activities. In: Proc. of the IEEE/CVF Int. Conf. on Comput. Vis. Workshops. pp. 0–8 (2019)
2. Abu Farha, Y., Richard, A., Gall, J.: When will you do what?-anticipating temporal occurrences of activities. In: Proc. of the IEEE/CVF Conf. on Comput. Vis. and Pattern Recognit. pp. 5343–5352 (2018)
3. Akbari, H., Yuan, L., Qian, R., Chuang, W.H., Chang, S.F., Cui, Y., Gong, B.: Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. Adv. Neural Inf. Process. Syst. **34**, 24206–24221 (2021)
4. Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lučić, M., Schmid, C.: Vivit: A video vision transformer. In: Proc. of the IEEE/CVF Int. Conf. on Comput. Vis. pp. 6836–6846 (2021)
5. Bokhari, S.Z., Kitani, K.M.: Long-term activity forecasting using first-person vision. In: 13th Asian Conf. on Comput. Vis. pp. 346–360 (2017)
6. Bullinaria, J.A., Levy, J.P.: Extracting semantic representations from word co-occurrence statistics: A computational study. Behavior Research Method. **39**, 510–526 (2007)
7. Damen, D., Doughty, H., Farinella, G.M., Fidler, S., Furnari, A., Kazakos, E., Moltisanti, D., Munro, J., Perrett, T., Price, W., et al.: Scaling egocentric vision: The epic-kitchens dataset. In: Proc. of the Eur. Conf. on Comput. Vis. pp. 720–736 (2018)
8. Dessalene, E., Devaraj, C., Maynord, M., Fermüller, C., Aloimonos, Y.: Forecasting action through contact representations from first person video. IEEE Trans. on Pattern Anal. and Mach. Intell. **45**(6), 6703–6714 (2023). https://doi.org/10.1109/TPAMI.2021.3055233
9. Dima, D., Doughty, H., Farinella, G.M., Antonino, F., Evangelos, K., Ma, J., Davide, M., Munro, J., Toby, P., Price, W., et al.: Rescaling egocentric vision: Collection, pipeline and challenges for epic-kitchens-100. Int. J. of Comput. Vis. **130**(1), 33–55 (2022)
10. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: Int. Conf. on Learning Representations. pp. 1–21 (2021)
11. Frederiksen, C.: Propositional representations in psychology. In: Int. Encyclopedia of the Social & Behavioral Sci.s, pp. 12219–12224. Springer (2001). https://doi.org/https://doi.org/10.1016/B0-08-043076-7/01490-X
12. Furnari, A., Battiato, S., Maria Farinella, G.: Leveraging uncertainty to rethink loss functions and evaluation measures for egocentric action anticipation. In: Proc. of the Eur. Conf. on Comput. Vis. Workshops. pp. 0–10 (2018)
13. Furnari, A., Farinella, G.M.: Rolling-unrolling lstms for action anticipation from first-person video. IEEE Trans. on Pattern Anal. and Mach. Intell. **43**(11), 4021–4036 (2021). https://doi.org/10.1109/TPAMI.2020.2992889

14. Girase, H., Agarwal, N., Choi, C., Mangalam, K.: Latency matters: Real-time action forecasting transformer. In: IEEE/CVF Conf. on Comput. Vis. and Pattern Recognit. pp. 18759–18769 (2023). https://doi.org/10.1109/CVPR52729.2023.01799
15. Girdhar, R., Grauman, K.: Anticipative video transformer. In: Proc. of the IEEE/CVF Int. Conf. on Comput. Vis. pp. 13505–13515 (2021)
16. Girdhar, R., Ramanan, D.: Attentional pooling for action recognit. Adv. Neural Inf. Process. Syst. **30**, 1–10 (2017)
17. Huang, J., Li, Y., Feng, J., Wu, X., Sun, X., Ji, R.: Clover: Towards a unified video-language alignment and fusion model. In: Proc. of the IEEE/CVF Conf. on Comput. Vis. and Pattern Recognit. pp. 14856–14866 (2023)
18. Jia, C., Yang, Y., Xia, Y., Chen, Y.T., Parekh, Z., Pham, H., Le, Q., Sung, Y.H., Li, Z., Duerig, T.: Scaling up visual and vision-language representation learning with noisy text supervision. In: Int. Conf. on Mach. Learning. pp. 4904–4916 (2021)
19. Ke, Q., Fritz, M., Schiele, B.: Time-conditioned action anticipation in one shot. In: Proc. of the IEEE/CVF Conf. on Comput. Vis. and Pattern Recognit. pp. 9925–9934 (2019)
20. Koppula, H.S., Saxena, A.: Anticipating human activities using object affordances for reactive robotic response. IEEE Trans. on Pattern Anal. and Mach. Intell. **38**(1), 14–29 (2015)
21. Li, Y., Wu, C.Y., Fan, H., Mangalam, K., Xiong, B., Malik, J., Feichtenhofer, C.: Mvitv2: Improved multiscale vision transformers for classification and detection. In: Proc. of the IEEE/CVF Conf. on Comput. Vis. and Pattern Recognit. pp. 4804–4814 (2022)
22. Lin, F., Qiu, Z., Liu, C., Yao, T., Xie, H., Zhang, Y.: Prototypical matching networks for video object segmentation. IEEE Trans. on Imag. Process. **32**, 5623–5636 (2023). https://doi.org/10.1109/TIP.2023.3321462
23. Liu, T., Lam, K.M.: A hybrid egocentric activity anticipation framework via memory-augmented recurrent and one-shot representation forecasting. In: Proc. of the IEEE/CVF Conf. on Comput. Vis. and Pattern Recognit. pp. 13904–13913 (2022)
24. Liu, Z., Hu, H., Lin, Y., Yao, Z., Xie, Z., Wei, Y., Ning, J., Cao, Y., Zhang, Z., Dong, L., et al.: Swin transformer v2: Scaling up capacity and resolution. In: Proc. of the IEEE/CVF Conf. on Comput. Vis. and pattern Recognit. pp. 12009–12019 (2022)
25. Ma, Y., Xu, G., Sun, X., Yan, M., Zhang, J., Ji, R.: X-clip: End-to-end multi-grained contrastive learning for video-text retrieval. In: Proc. of the 30th ACM Int. Conf. on Multimed. pp. 638–647 (2022)
26. Manousaki, V., Bacharidis, K., Papoutsakis, K., Argyros, A.: Vlmah: Visual-linguistic modeling of action history for effective action anticipation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1917–1927 (2023)
27. Marchetti, F., Becattini, F., Seidenari, L., Bimbo, A.D.: Multiple trajectory prediction of moving agents with memory augmented networks. IEEE Trans. on Pattern Anal. and Mach. Intell. **45**(6), 6688–6702 (2023). https://doi.org/10.1109/TPAMI.2020.3008558
28. Martinez, B., Modolo, D., Xiong, Y., Tighe, J.: Action recognition with spatial-temporal discriminative filter banks. In: Int. Conf. Comp. Vis. pp. 5482–5491 (2019)
29. McInnes, L., Healy, J., Melville, J.: Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:1802.03426 (2018)

30. Miech, A., Laptev, I., Sivic, J.: Learnable pooling with context gating for video classification. In: Proc. of the IEEE/CVF Conf. on Comput. Vis. and pattern Recognit. Workshop. pp. 1–8 (2017)
31. Osman, N., Camporese, G., Coscia, P., Ballan, L.: Slowfast rolling-unrolling lstms for action anticipation in egocentric videos. In: IEEE/CVF Int. Conf. on Comput. Vis. Workshops. pp. 3430–3438 (2021). https://doi.org/10.1109/ICCVW54120.2021.00383
32. Qi, Z., Wang, S., Su, C., Su, L., Huang, Q., Tian, Q.: Self-regulated learning for egocentric video activity anticipation. IEEE Trans. on Pattern Anal. and Mach. Intell. **45**(6), 6715–6730 (2023). https://doi.org/10.1109/TPAMI.2021.3059923
33. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: Int. Conf. on Mach. Learning. pp. 8748–8763 (2021)
34. Ramanathan, V., Li, C., Deng, J., Han, W., Li, Z., Gu, K., Song, Y., Bengio, S., Rosenberg, C., Fei-Fei, L.: Learning semantic relationships for better action retrieval in images. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 1100–1109 (2015)
35. Reimers, N., Gurevych, I.: Sentence-bert: Sentence embeddings using siamese bert-networks. In: Proc. of the 2019 Conf. on Empir. Methods Nat. Lang. Process. and the 9th Int. Joint Conf. on Nat. Lang. Process. pp. 3980–3990 (2019). https://doi.org/{10.18653/V1/D19-1410}
36. Richard, A., Kuehne, H., Gall, J.: Weakly supervised action learning with rnn based fine-to-coarse modeling. In: Proc. of the IEEE/CVF Conf. on Comput. Vis. and Pattern Recognit. pp. 754–763 (2017)
37. Roy, D., Fernando, B.: Action anticipation using latent goal learning. In: Proc. of the IEEE/CVF Winter Conf. on Appl. of Comput. Vis. pp. 2745–2753 (2022)
38. Roy, D., Rajendiran, R., Fernando, B.: Interaction region visual transformer for egocentric action anticipation. In: Winter Conf. on Appl. of Comp. Vis. pp. 6740–6750 (2024)
39. Sener, F., Singhania, D., Yao, A.: Temporal aggregate representations for long-range video understanding. In: Proc. of the Eur. Conf. on Comput. Vis. (ECCV). pp. 154–171. Springer (2020)
40. Singh, K.K., Fatahalian, K., Efros, A.A.: Krishnacam: Using a longitudinal, single-person, egocentric dataset for scene understanding tasks. In: Proc. of the IEEE/CVF Winter Conf. on Appl. of Comput. Vis. pp. 1–9 (2016)
41. Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. Adv. Neural Inf. Process. Syst. **30** (2017)
42. Somin, W., Silvio, A., Byron, C.W.: Revisiting relation extraction in the era of large language models. In: Proc. of the 61st Conf. Assoc. Comput. Linguist. Meet. pp. 15566–15589 (2023). https://doi.org/10.18653/V1/2023.ACL-LONG.868
43. Soran, B., Farhadi, A., Shapiro, L.: Generating notifications for missing actions: Don't forget to turn the lights off! In: Proc. of the IEEE/CVF Int. Conf. on Comput. Vis. pp. 4669–4677 (2015)
44. Stein, S., McKenna, S.J.: Combining embedded accelerometers with computer visio for recognizing food preparation activities. In: Proc. of the 2013 ACM Int. Conf. Ubiquitous Comput. pp. 729–738 (2013)
45. Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P.H., Hospedales, T.M.: Learning to compare: Relation network for few-shot learning. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 1199–1208 (2018)

46. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Adv. Neural. Inf. Process. Syst. vol. 30, pp. 1–11 (2017)
47. Vondrick, C., Pirsiavash, H., Torralba, A.: Anticipating visual representations from unlabeled video. In: Proc. of the IEEE/CVF Conf. on Comput. Vis. and Pattern Recognit. pp. 98–106 (2016)
48. Wang, D., Liu, H., Wang, N., Wang, Y., Wang, H., McLoone, S.: Seem: A sequence entropy energy-based model for pedestrian trajectory all-then-one prediction. IEEE Trans. on Pattern Anal. and Mach. Intell. **45**(1), 1070–1086 (2023). https://doi.org/10.1109/TPAMI.2022.3147639
49. Wang, Q., Li, B., Xiao, T., Zhu, J., Li, C., Wong, D.F., Chao, L.S.: Learning deep transformer models for machine translation. In: Proc. of the 57th Conf. Assoc. Comput. Linguist. Meet. pp. 1–13 (2019)
50. Wilson, S., Mihalcea, R.: Measuring semantic relations between human activities. In: Int. Joint Conf. on Nat. Lang. Process. pp. 664–673 (2017)
51. Wu, C.Y., Li, Y., Mangalam, K., Fan, H., Xiong, B., Malik, J., Feichtenhofer, C.: Memvit: Memory-augmented multiscale vision transformer for efficient long-term video recognition. In: Proc. of the IEEE/CVF Conf. on Comput. Vis. and Pattern Recognit. pp. 13587–13597 (2022)
52. Wu, Y., Zhu, L., Wang, X., Yang, Y., Wu, F.: Learning to anticipate egocentric actions by imagination. IEEE Trans. on Imag. Process. **30**, 1143–1152 (2020)
53. Xu, X., Li, Y.L., Lu, C.: Learning to anticipate future with dynamic context removal. In: Proc. of the IEEE/CVF Conf. on Comput. Vis. and Pattern Recognit. pp. 12724–12734 (2022). https://doi.org/10.1109/CVPR52688.2022.01240
54. Zhai, X., Wang, X., Mustafa, B., Steiner, A., Keysers, D., Kolesnikov, A., Beyer, L.: Lit: Zero-shot transfer with locked-image text tuning. In: Proc. of the IEEE/CVF Conf. on Comput. Vis. and Pattern Recognit. pp. 18123–18133 (2022)
55. Zhang, M., Ma, K.T., Lim, J.H., Zhao, Q., Feng, J.: Deep future gaze: Gaze anticipation on egocentric videos using adversarial networks. In: IEEE Conf. on Comput. Vis. and Pattern Recognit. pp. 3539–3548 (2017). https://doi.org/10.1109/CVPR.2017.377
56. Zhong, Z., Schneider, D., Voit, M., Stiefelhagen, R., Beyerer, J.: Anticipative feature fusion transformer for multi-modal action anticipation. In: Proc. of the IEEE/CVF Winter Conf. on Appl. of Comput. Vis. pp. 6068–6077 (2023)