Video Attribute Prototype Network: A New Perspective for Zero-Shot Video Classification

Bo Wang[1] Kaili Zhao[1] Hongyang Zhao[1] Shi Pu[2] Bo Xiao[1] Jun Guo[1]
[1]Beijing University of Posts and Telecommunications. [2]Indepent Researcher.
{bobo, kailizhao, hooyoung_zhao, xiaobo, guojun}@bupt.edu.cn pushiucm@gmail.com

Abstract

Video attributes, which leverage video contents to instantiate class semantics, play a critical role in diversifying semantics in zero-shot video classification, thereby facilitating semantic transfer from seen to unseen classes. However, few presences discuss video attributes, and most methods consider class names as class semantics that tend to be loosely defined. In this paper, we propose a Video Attribute Prototype Network (VAPNet) to generate video attributes that learns in-context semantics between video captions and class semantics. Specifically, we introduce a cross-attention module in the Transformer decoder by considering video captions as queries to probe and pool semantic-associated class-wise features. To alleviate noises in pre-extracted captions, we learn caption features through a stochastic representation derived from a Gaussian representation where the variance encodes uncertainties. We utilize a joint video-to-attribute and video-to-video contrastive loss to calibrate visual and semantic features. Experiments show that VAPNet significantly outperforms SoTA by relative improvements of 14.3% on UCF101 and 8.8% on HMDB51, and further surpasses the pre-trained vision-language SoTA by 4.1% and 17.2%. Code is available[1].

1. Introduction

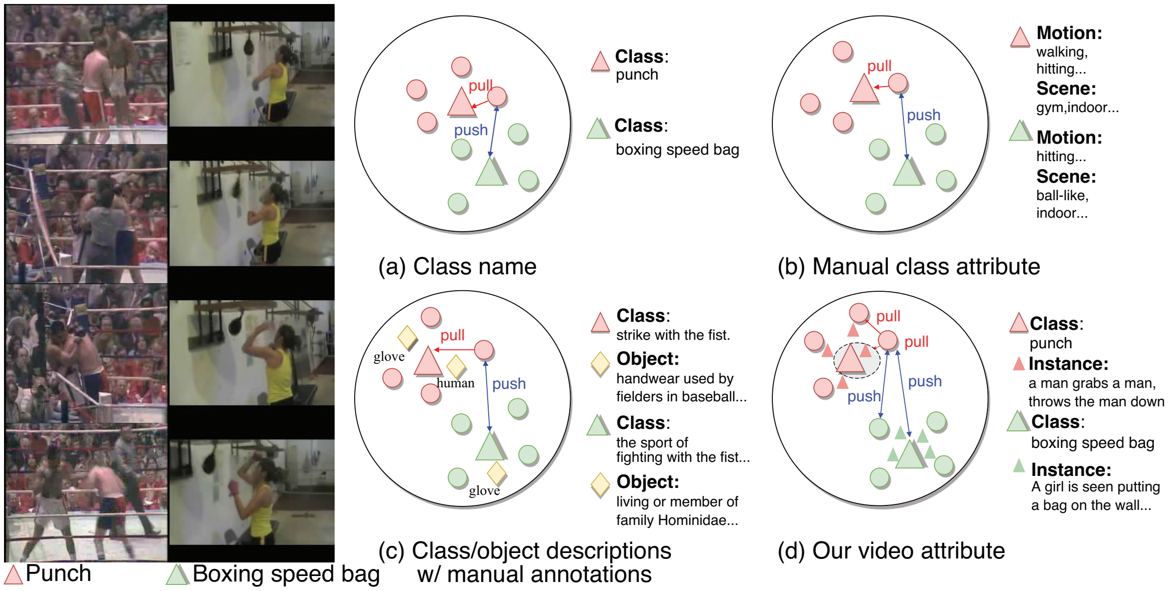
The underlying principle of recent advancements in zero-shot learning [54, 42, 5] is to acquire a latent space bridging the gap between vision and language modalities. This space is trained on diverse semantics from seen classes and subsequently utilized for predicting unseen classes. In zero-shot image classification [30, 58, 4], image attribute describes discriminative visual concepts of objects shared between seen and unseen classes, and has shown its potential. However, video attributes are still undervalued in zero-shot video classification (ZSVC). Tasks such as video captioning [40], video question answering [6], and video understanding [19] in computer vision emphasize the significance of objects and their relationships in time and space when modeling video semantics. We conjecture that video attributes in ZSVC should encompass concepts of objects and their temporal/spatial contexts to effectively describe video contents. Video attributes with video captions enrich the class semantics, resulting in improved model generalizability compared to commonly-used class names.

Reviewing the literature [10, 33, 5, 3, 42], we observe that few presences clearly discuss video attributes. Here, we illustrate three typical semantics of ZSVC — class names, manual class descriptions, and class/object expanded descriptions, along with our video attributes in Fig. 1. Class names are simple but widely-used semantics [44, 59, 3, 42] and are encoded by language models (Word2Vec [38], Glove [41], BERT [8]). Moreover, some classes are too loosely defined (*e.g.*, a simple word "Punch") and some are too close to discriminate, *e.g.*, "Punch" *vs.* "Boxing speed bag" or "Horse Race" *vs.* "Horse Riding". The loosely-defined and close classes will hurt semantic diversity, thus existing methods make efforts to enrich semantics.

For instance, [11, 29, 33] explore class attributes to expand class-wise descriptions by manually defining a complete set of atomic attributes, such as scenes and motions as shown in Fig. 1(b). Class-based expansions still suffer from similar semantics of close classes ("hitting" in both "Punch" and "Boxing speed bag"). To further supplement semantics, [15, 21, 36, 5] apply object-wise attributes appearing in videos, *e.g.*, adding "human" or "glove" in "Punch". The latest ER [5] expands class and object descriptions via web-crawled re-explanations and extra human annotations as illustrated in Fig. 1(c). However, two aspects of the above methods can be improved. First, apart from the tedious annotation, manual attributes in Fig. 1(b)(c) will cause data bias by different annotators. Second, all the above methods only focus on static class/object names or predefined motions but neglect instance-wise objects and their spatial/temporal info which are important for video semantics.

To diversify such video semantics naturally, we utilize video captions that are compatible to comprehend instance-wise semantics of objects and their spatial/temporal con-

[1] https://github.com/bobo199830/VAPNet



Punch
Boxing speed bag
pull
push
Class: punch
Class: boxing speed bag
pull
push
Motion: walking, hitting...
Scene: gym,indoor...
Motion: hitting...
Scene: ball-like, indoor...
glove
pull
human
push
glove
Class: strike with the fist.
Object: handwear used by fielders in baseball...
Class: the sport of fighting with the fist...
Object: living or member of family Hominidae...
pull
pull
push
push
Class: punch
Instance: a man grabs a man, throws the man down
Class: boxing speed bag
Instance: A girl is seen putting a bag on the wall...

(a) Class name

(b) Manual class attribute

(c) Class/object descriptions w/ manual annotations

(d) Our video attribute

Figure 1: Illustrations of visual and semantic embeddings with three typical semantics and our video attribute. ○ and △ (◊) represent visual and semantic features individually. Existing semantics neglect spatial and temporal contexts – class names (a), class attributes with motion/scene descriptions (b), static class/object descriptions (c). Instead, our video attributes (dotted circles in (d)) associate instance-wise video captions and class-wise semantics to enrich semantics, enabling a better visual-semantic calibration and model generalization.

texts. Since existing video captioning model [27] captions a video by understanding actions and events automatically, freeing from manual annotation for ZSVC task. As shown in Fig. 1(d), video captions of class "Boxing speed bag" — "a girl is seen putting a bag on the wall" comprise of objects and their spatial (girl, bag, wall) and temporal (putting) contexts, which will be easily separated from "a man grabs a man, throws the man down" of the close class "Punch".

In this paper, we present Video Attribute Prototype Network (VAPNet), a vision-language model to generate video attributes that instantiate class names by video captions. In specific, to obtain learnable and discriminative video attributes, we introduce a cross-attention module that considers pre-extracted video captions as queries to probe and pool correlated semantics of each class. Besides, to mitigate inaccurate captions, we apply a caption uncertainty module with learning feature and uncertainty simultaneously. At last, we exploit a dual contrastive loss that contrasts video-video and video-attribute jointly so that close classes can be further separated based on additional visual contrasts, as blue/red arrows shown in Fig. 1(d). We conduct extensive ablations and test VAPNet under complete ZSVC protocols. Experiments show that VAPNet significantly outperforms SoTA by relative improvements of 14.3% and 8.8% on UCF101 and HMDB51, and surpasses large-scale pre-training SoTA by 4.1% and 17.2% separately.

2. Related Work

Zero-shot video classification: Existing methods in ZSVC improve model generalization by adding additional semantics to diversify the semantics of seen classes. Additional semantics include class-wise attributes, concepts of objects, and expanded descriptions of class/object names. First, [33, 61, 14] rely on manually annotated category attributes in particular datasets (*e.g.*, UCF101 [47]), which is high-cost and hard to migrate to arbitrary categories in real-world scenarios. Then, methods [21, 37, 36, 12, 25] utilizing object concepts borrow the idea of image attributes adopted in zero-shot image classification, where objects in videos are detected, and similarities between names of objects and categories are computed during the test. To learn discriminant objects concepts, diverse regularization are constrained on objects such as relations in spatial [36], object/scenes [55], intra-class [13], and action-object [15]. The SoTA methods, ER [5], JigsawNet [43], and CLASTER [17] manually expand descriptions of object or class names and optimize a joint visual-semantic embedding used in E2E [3], AURL [42], and ResT [31]. Compared to static objects detected in single frames or pre-defined class descriptions, our video attributes make use of easily attainable video captions that contain both spatial and temporal info, explicitly diversifying semantics at minimal cost.

Vision-language models: Large-scale vision-language pre-trained models (*e.g.*, CLIP [45], ALIGN [22], METER [9], BeiT-3 [53], CoCa [60]) have achieved inspiring progress in recent years. Closest to our VAPNet, contrastive model — CLIP [45] shows strong general transfer ability in down-stream video classification (78.9% *vs*. 58.7% of SoTA method [31] in ZSVC). In this paper, we leverage the pre-trained CLIP as our extra backbone to show that our VAP-

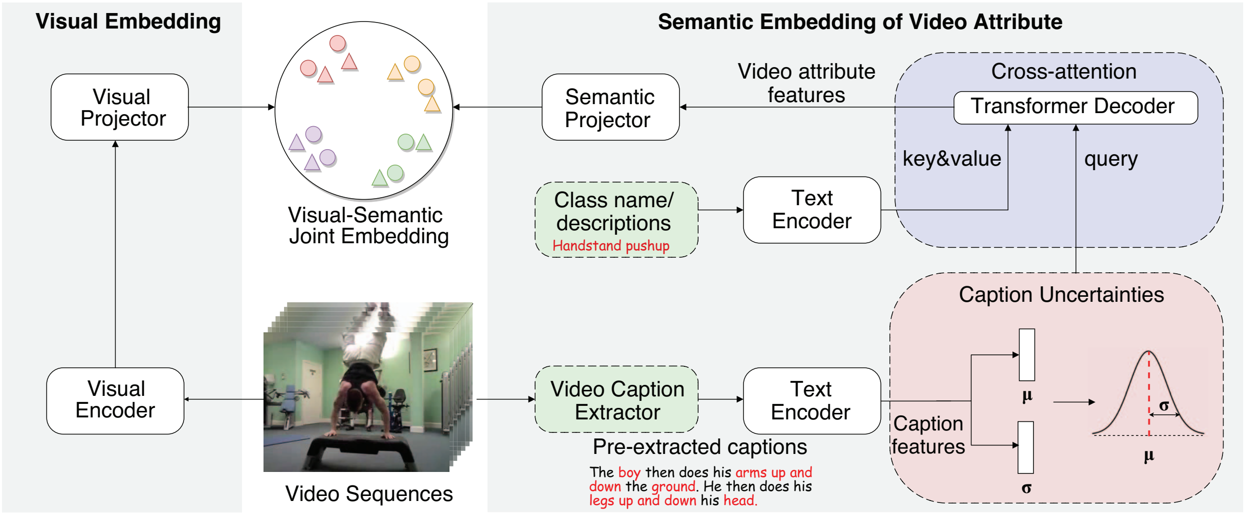


Figure 2: Architecture of video attribute prototype network (VAPNet). VAPNet optimizes a joint visual-semantic embedding on features of video sequences and the proposed video attributes. The semantic inputs are pre-extracted video captions and class name/descriptions. Video attributes are end-to-end learnable through text encoder, caption uncertainty module, transformer decoder, and semantic projector.

Net also can help improve the performance of well-trained large-scale vision-langugage model.

3. Video Attribute Prototype Network

3.1. Architecture

Fig. 2 illustrates the architecture of VAPNet, a vision-language model that optimizes a joint visual and semantic space during training. The inputs include video sequences, the corresponding video captions, and class names/descriptions. With the inputs, we exploit R(2+1)D [48] and SBERT[46] as the visual and text encoders separately. At the end of VAPNet, we apply a 3-layer MLP in our visual and semantic projector to map features of visual and video attributes to the joint embedding space. In specific, we design two critical components to generate video attributes: cross-attention module to instantiate class semantics and caption uncertainty module to alleviate inaccurate captions. At last, we propose a dual contrastive loss by contrasting video-video and video-attribute so that visual-semantic features within the same class tend to be calibrated. Below, we will elaborate on each component in turn, followed by training and inference of VAPNet.

3.2. Video Attribute Generation

The ultimate goal of ZSVC is to assign the most-related category semantics to each video feature. However, we observe that loosely-defined category names bring challenges to perform discrimination. For example, class *mixing* can be understood in multiple ways, since *mixing* can be "spend time together" or "blend music". However, the class *mixing* denotes "combine food" in the UCF101 dataset [47]. Thus, we explicitly enrich the category names using both video captions (instance-wise spatial/temporal contexts) and more structured category-related descriptions (class-wise).

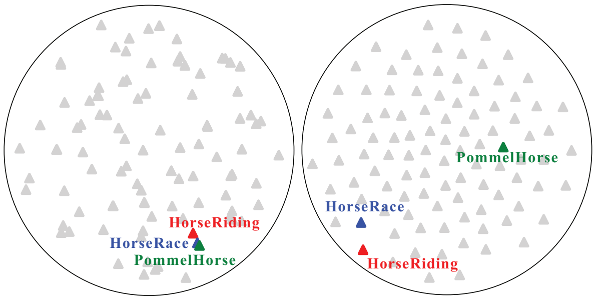
Instance-wise video caption: Video captions are easily attainable, such as live captioning services or pre-trained video captioning models in video captioning research field. In this paper, we choose a pre-trained SoTA video caption extractor (*e.g.*, PDVC [52]) to obtain video captions that capture detailed visual contents and coherent descriptions. Given a video clip x_n, we leverage the caption extractor to generate caption CAP_n with L sentences $\mathrm{CAP}_n = \{\mathrm{CAP}_{n_1}, ..., \mathrm{CAP}_{n_L}\}$. We show examples of caption results in Table 1, *e.g.*, captions of *punch* involve instance/video wise spatial (man, ring, room, bag) contexts, and temporal (grab, throw...down, fight, put) contexts. These captions elaborate semantics, thus facilitating to distinguish similar actions. To exploit the prior knowledge in language models, we introduce a text encoder (*e.g.*, BERT [46]) to encode captions. Here, we average encoding features of all caption sentences given n-th video, summarizing an instance-wise caption feature $\mathrm{cap}_n \in \mathbb{R}^d$.

$$\mathrm{cap}_n = \frac{1}{L}\sum_{i=1}^{L} \mathrm{BERT}(\mathrm{CAP}_{n_i}) \tag{1}$$

Class-wise semantics: To expand class-wise descriptions, we consider class names as queries to crawl alternative descriptions $\mathrm{DES}_m = \{\mathrm{DES}_{m_1}, ..., \mathrm{DES}_{m_K}\}$ from Wikipedia and dictionaries, where K is the number of descriptions for m-th class. Different from manually choosing class descriptions in ER [5], we automatically filter out irrelevant descriptions by calculating the similarity sim_k between embedding of the class name $\mathrm{cat}_m = \mathrm{BERT}(\mathrm{class\,name})$ and its corresponding alter-

Table 1: Examples of instance-wise video captions.

Category	Video	Video Captions
punch	video1	The man then grabs a man and the man throws the man down. The man continue fighting another man.
	video2	Two men are seen standing in a ring. Two men then begin fighting each other.
boxing speed bag	video1	A girl is seen putting a bag on the wall. The girl begins playing a bag.
	video2	A boy is seen standing in a room. The boy then kicks the bag.
playing daf	video1	A woman is sitting on chair with a towel. She then puts her hands on her face. A woman is playing a drum.



(a) Class-wise semantics **(b) Our Video Attribute**

Figure 3: Comparisons of concatenation op and our cross-attention op in video attributes, which facilitates more discrimination.

native descriptions $\text{des}_{m_k} = \text{BERT}(\text{DES}_{m_k})$ with $\text{sim}_k = \frac{\text{des}_{m_k} \cdot \text{cat}_m}{\|\text{des}_{m_k}\| \cdot \|\text{cat}_m\|}$. Thus, the des_{m_k} will be ranked and we empirically select top-K' ($K' = 20$) descriptions to get our final fused class descriptions des_m.

$$\text{des}_m = \frac{1}{K'} \sum_{k=1}^{K'} \text{des}_{m_k} \tag{2}$$

Cross-attention between video captions and class semantics: Existing methods enrich semantics by learning fixed class-wise and additional class/object-wise descriptions in parallel branches [5], sequential optimization [21, 37, 36], or in a concatenation way [20]. Instead, we generate learnable semantics by learning attentive semantics of instance-wise video captions from class-wise descriptions. Utilizing the Transformer decoder with built-in cross-attention proves to be a perfect fit for in-context learning [50, 34], allowing us to focus on relevant parts of the captions and align them with the class name features. This enables us to effectively handle the variability and complexity inherent in video data. Here, we consider features of video captions cap_n as query to probe and pool both features of category names cat_m and expanded descriptions des_m. We follow operations in the standard transformer [50] to obtain video attribute $\text{att}_n^m \in \mathbb{R}^d$ that instantiates m-th class semantics by involving n-th video captions.

$$\begin{aligned} &Q = W_q \text{cap}_n, K^{cat} = W_k^{cat} \text{cat}_m, V^{cat} = W_v^{cat} \text{cat}_m \\ &\qquad K^{des} = W_k^{des} \text{des}_m, V^{des} = W_v^{des} \text{des}_m \\ &\text{att}_n^m = \text{softmax}(\frac{QK^{cat^T}}{\sqrt{d}})V^{cat} + \text{softmax}(\frac{QK^{des^T}}{\sqrt{d}})V^{des} \end{aligned} \tag{3}$$

Where cap_n, cat_m, des_m are all d-dimensional semantic features; $W_q, W_k^{cat}, W_k^{des}, W_v^{cat}, W_v^{des}$ are $d \times d$ learnable parameters. Here Q, K, V are the d-dimensional transformed features. Similar to a transformer operation, we learn a $d \times d$ correlation matrix between Q and K with softmax and update the final class-wise semantic features with V. With the cross-attention mechanism on features of video captions and class-wise semantics, salient features are amplified and will contribute to discrimination.

To illustrate the effectiveness of cross-attention, in Fig. 3, we compare the representations using t-SNE [49] visualization, which are learned by the architecture of Fig. 2 with our video attributes and the concatenation [20] of category names and their expanded descriptions on UCF101 dataset [47]. We show three representative classes *Horse Race*, *Horse Riding*, and *Pommel Horse* with color triangles and other semantics with grey triangles for better visualization. We generate video attributes of the three classes using video captions of one video from *Horse Race*. We conclude that our video attributes help push apart confusing classes (*Horse Race vs. Horse Riding*) and enlarge distances between different categories but be assigned with similar semantics in class names. Besides, the overall semantics with our video attributes distribute more uniformly on the embedding compared to class-wise semantics, leading to the effectiveness of the model's generalizability [42]. More quantitative results will be illustrated in Sec. 4.2.

3.3. Caption Uncertainty Module

Besides advantageous clues, video captions also involve less relevant contents since the caption extractor is pretrained on limited videos. As illustrated in Table 1, the captions of *playing daf* include meaningful descriptions "playing drum" but also bring some imprecise contents (*e.g.*, put hand on face, sit with towel) at the same time. We propose caption uncertainty module to address the challenge of inherent noise by encoding uncertainty in the variance of a Gaussian distribution while utilizing its mean for representation. Unlike existing work that omit uncertainty [25] or rely on human annotators to clean uncertain data [5], our method presents a novel solution for handling uncertainty in video captions. Specifically, we define a stochastic representation z_n sampled from a Gaussian distribution with learnable mean μ_n and variance σ_n^2:

$$p(z_n|\text{cap}_n) = \mathcal{N}(z_n; \mu_n, \sigma_n^2 I) \tag{4}$$

Here, the mean $\mu_n = U(\text{cap}_n)$ represents the intrinsic semantics while the variance $\sigma_n^2 = \Lambda(\text{cap}_n)$ describes the uncertainty of predicted mean, where U and $\Lambda \in \mathbb{R}^{d\times d}$ are learnable parameters. Since sampling operation is not differentiable, we apply the classic re-parameterization trick [26] to enable back-propagation:

$$\text{cap}'_n = \mu_n + \sigma_n \epsilon, \epsilon \sim \mathcal{N}(0, I) \tag{5}$$

The cap'_n is considered as the equivalent sampling representation z_n and will replace cap_n in Eq. 3 to generate more reliable video attributes. During caption uncertainty learning, small variance may be predicted for all samples and no uncertainty could be learnt. To alleviate such model collapse, we encourage the learnt distribution close to the normal distribution by introducing a Kullback-Leibler divergence [1]:

$$\mathcal{L}_{\text{kl}} = KL[\mathcal{N}(z_n|\mu_n, \sigma_n^2)||\mathcal{N}(\epsilon|0, I)] \tag{6}$$

3.4. Training & Inference

Loss function: During training, most existing vision-language models [42, 5] exploit a standard contrastive loss to calibrate visual and semantic features:

$$\mathcal{L}_{\text{vs}} = -\log \frac{e^{(v_n \cdot \text{att}_n/\tau)}}{\sum_{k=1}^{N} \mathbf{1}_{k\neq n} e^{(v_n \cdot \text{att}_k/\tau)}} \tag{7}$$

τ is the temperature parameter and $\mathbf{1}$ is an indicator function; att_n is class semantics (*e.g.*, class names or our video attributes) and v_n is visual features. While Eq. 7 exploits across-modal (*i.e.*, video-to-semantic) information, it neglects rich within-modal information (*i.e.*, video-to-video), which contains instantiated video features to further assist alignment within classes and separation between classes. We maintain videos in the same class as positive pairs while others in batch as negative pairs and perform a joint video-to-attribute and video-to-video contrastive loss as follows:

$$\mathcal{L}_{\text{vs}-\text{vv}} = -\log \frac{e^{(v_n \cdot \text{att}_n/\tau)} + \sum_{k=1}^{N} \mathbf{1}_{y_k = y_n} e^{(v_n \cdot v_k/\tau)}}{\sum_{k=1}^{N} \mathbf{1}_{k\neq n} [e^{(v_n \cdot \text{att}_k/\tau)} + e^{(v_n \cdot v_k/\tau)}]} \tag{8}$$

During training VAPNet, our overall loss function is:

$$\mathcal{L}_{\text{VAPNet}} = \mathcal{L}_{\text{vs}-\text{vv}} + \mathcal{L}_{\text{kl}} \tag{9}$$

Training: We train our VAPNet only on source dataset $\mathcal{D}_s$ with seen classes $\mathcal{S}$ and test on target dataset $\mathcal{D}_u$ with unseen classes $\mathcal{T}$. We follow the strict setting in E2E [3], which requires $\mathcal{S}$ has no overlap with $\mathcal{T}$. Specifically, the requirement is as follows, here θ is the distance threshold:

$$\forall i \in \mathcal{S}, \quad \min_{j\in\mathcal{T}} \cos(i, j) > \theta \tag{10}$$

Inference: For a video clip $x_n \in \mathcal{D}_u$, we extract its visual feature v_n and generate T candidate video attributes att_n^m in Eq. 3, where $m = 1, ..., T$. We perform the Nearest Neighbor Search (NNS) to obtain the prediction result:

$$\arg\max_{m\in\mathcal{T}} \cos(v_n, \text{att}_n^m) \tag{11}$$

4. Experiments

4.1. Settings

Datasets: We train our VAPNet on Kinetics700 [24] and test on UCF101 [47] and HMDB51 [28]. Kinetics700 [24] is one of the large-scale video benchmarks for action recognition where 536489 videos with 700 various human actions are collected. UCF101 [47] contains 13320 YouTube videos with 101 action names of sports. HMDB51 [28] has 51 action classes of sports and daily activities and 6767 YouTube videos and commercial videos. For captioning model, we use Activity Captions [27] that contains 20k long untrimmed videos of various human activities.

Training protocol: VAPNet follows the rigorous setting outlined in E2E [3] and adheres to the conditions specified in Eq.10 to generate Kinetics662, which comprises of 501614 videos. To ensure that our pre-trained video captioning model is not exposed to any test classes, we adopt the same setting and use $\theta = 0.05$ to select 177 classes from the ActivityNet Captions dataset, then retrain our video captioning model. Furthermore, it has been noted in previous studies [3, 42, 31] that learning visual features from scratch is crucial in achieving rigorous results in ZSVC. This is because pre-trained backbones (*e.g.*, R(2+1)D on SUN [57]) learned from overlapping classes result in info leakage. Here, we conduct all ablations and comparisons in the setting of learning visual encoders from scratch.

Evaluation protocol: For fair comparisons with previous works, we present two evaluation protocols. (1) **Protocol 1**: Following E2E [3, 42, 31], we train our VAPNet on Kinetics662 and test on full UCF101 and HMDB51 (*i.e.*, 100% Split), which avoids the random selection of categories and returns more convincing results. (2) **Protocol 2**: Following previous works [2, 3, 5, 18, 35], they randomly choose half of the target dataset's classes (*i.e.*, 50% Split), 50 for UCF101 and 25 for HMDB51 to evaluate the performance. Here we follow E2E [3] to repeat 10 independent runs, reporting the average results for each run.

Implementation details: We include 16 frames of each video to create one video clip following the standard protocol in [51]. For video attribute generation, we retrain the SoTA video caption model PDVC [52] on the filtered ActivityNet Caption to obtain caption results. During training, we adopt the number of video clip C as 1 and set it as 1 or 25 during inference. We utilize R(2+1)D (512D) and SBERT (384D) to extract visual feature and semantic features separately. The frames of a video clip are resized into

Table 2: Ablations of our VAPNet under **Protocol 1** with 1 clip. CAT: category name, CAP: instance-wise video caption, DES: category descriptions obtained by web search engine based on similarity filtering, CA: cross-attention module, UM: caption uncertainty module, Pre: pre-trained backbone. Top-1 accuracy for UCF and HMDB is reported. The higher, the better. Red numbers indicate the best result.

(a) Ablations for different semantics

Method	**CAT**	**CAP**	**DES**	**UM**	**UCF top-1**	**HMDB top-1**
Base	✓				42.8	25.9
Base w/ DES	✓		✓		43.9	26.3
Base w/ CAP	✓	✓			44.2	26.7
VAPNet	✓	✓	✓	✓	48.9	29.3
VAPNet w/o UM	✓	✓	✓		48.4	28.4

(b) Ablations for cross-attention module

Method	**CAT**	**CAP**	**DES**	**CA**	**UCF top-1**
Base w/o CA	✓				42.8
	✓		✓		38.2
	✓	✓			31.9
	✓	✓	✓		24.1
Base w/ CA	✓		✓	✓	43.9
	✓	✓		✓	44.2
	✓	✓	✓	✓	48.4

(c) Ablations for different loss

Method	**CAT**	**CAP**	**DES**	**UM**	**UCF top-1**
VAPNet w/ Eq. 7	✓	✓	✓		46.9
VAPNet w/ Eq. 8	✓	✓	✓		48.4

(d) Ablations for training protocol

Method	**CAT**	**CAP**	**DES**	**Pre**	**UCF top-1**
VAPNet	✓	✓	✓		48.9
VAPNet w/ Pre	✓	✓	✓	✓	51.1

$16 \times 112 \times 112$. For visual projector and attribute projector, we build a 3-layer MLP (2 linear+bn+ReLU and linear+bn). The dimension of the visual-semantic joint space is 2048. The τ in Eq. 8 is 0.1, and the batch size is 256. We use Adam with weight decay for optimization. The initial learning rate is 1e-5 for cross-attention module and 1e-3 for other modules. All experiments are done on 8 RTX 3090.

4.2. Ablation Study

To demonstrate the effectiveness of each component in VAPNet, we conduct extensive ablations, accompanied by Q&A analyses. All ablations were evaluated using UCF or HMDB top-1 accuracy under **Protocol 1** with 1 clip.

Is it more rigorous with learning visual features from scratch? Recent studies, E2E [3] and ResT [31], suggest that pretrained visual backbones used in prior research [5, 17, 43] may lead to info leakage from training to test classes. To rigorously test VAPNet, we compare the performance of its visual encoder when trained from scratch on Kinetics662 without overlap classes *vs.* using a pretrained backbone on Kinetics400. As shown in Table 2d, there is a large improvement of 4.5% on the UCF dataset with the pre-trained backbone. We infer that Kinetics400 may contain similar semantic information to UCF, which could compromise fair comparison of model generalization in ZSVC. Therefore, learning visual features from scratch is a more rigorous setting, and we will conduct all ablation studies and comparisons under this setting.

Is our video attribute prototype beneficial? To substantiate that our video attribute surpasses commonly-used class-wise semantics, we create one variant of **Base** model, **Base w/ DES** which considers both category names (CAT) and dedicated class descriptions (DES). Here, **Base** is built only with CAT as text, then trained with the visual encoder + projector and text encoder + semantic projector in Fig. 2 tailored with Eq. 8 loss. For fair comparisons with VAPNet, we also apply the proposed cross-attention module to fuse CAT and DES in **Base w/ DES**. Table 2a clearly shows that video attribute (**VAPNet**) largely outperforms **Base** and **Base w/ DES** with top-1 accuracy of (48.9, 29.3) on (UCF, HMDB). Moreover, to show that video caption is a better choice as auxiliary semantics compared to extended class descriptions DES, we bring video captions to CAT alone and get **Base w/ CAP**, which increases the top-1 accuracy from (43.9, 26.3) to (44.2, 26.7) compared to **Base w/ DES**.

Below, we will qualitatively show that our video attributes enhance performance by diversifying class semantics and improving class discrimination. First, we intentionally select the unseen class *Nunchucks* which is distant from seen class names, then compare the closest seen classes captured by different representations learned from **VAPNet** w/ video attributes and **Base** w/ class names. Our VAPNet is able to retrieve *tai chi* which is semantically closer than *making snowman* by **Base**. This is because video attributes can bridge the semantic gaps between *tai chi* and *Nunchucks* by providing common video contexts such as "standing in the yard" and "holding/throwing a stick". Then, we elaborate ablations in view of discrimination of confusing classes. From Fig. 4, we observe our **VAPNet** with video attributes consistently outperforms **Base** with conventional class names. For instance, our **VAPNet** significantly improves the indistinguishable performance between *punch* and *boxing speed bag* from (2.5, 40.2) to (59.4, 68.2). To sum up, video attribute prototype benefits ZSVC in model generalization and discrimination.

Is the cross-attention module necessary? It is nontrivial to fuse semantics from different sources especially when category names and their web-crawled descriptions tend to be loosely defined and video captions may contain inaccurate semantics. To justify the non-triviality, we simply average features of class-wise semantics and video captions, obtaining a subset of models shown in **Base w/o CA** of Table 2b. As can be seen, there are dramatically decreases while comparing **Base** and models of **Base w/o CA** with 42.8 vs 38.2/31.9/24.1. In contrast, our cross-attention (CA) module identifies adaptive semantic features by learning attentions between instance-wise captions and class-wise semantics. As shown in Table 2b, our CA evidently improves the performance under any combinations of semantics, *e.g.*, top-1 acc of CAT+CAP+DES with **Base**

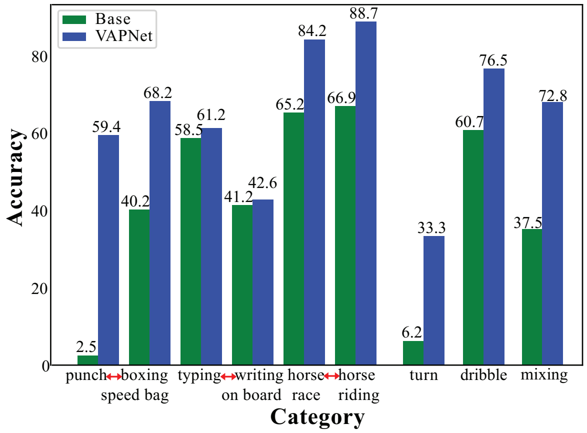


Figure 4: Improvements on close or loose-defined categories for VAPNet using video attributes *vs*. Base with commonly-used category names. ↔ represents similar class pairs.

w/o CA *vs*. **Base w CA** is increased from 24.1 to 48.4.

Is the caption uncertainty module helpful? Since video captions are extracted from a pretrained captioning model, inaccurate semantics are inevitably introduced. To reduce adverse effects of noisy semantics, we introduce the caption uncertainty module (UM) in Sec. 3.3 to learn adaptive semantic representations sampled from a Gaussian distribution. Here, we remove UM from VAPNet to show that UM plays an important role to consistently help improve the performance on both UCF and HMDB datasets. As shown in Table 2a, **VAPNet** increases (0.5, 0.9) points compared with **VAPNet w/o UM**. Apart from learning effective features, UM generates learnable variances that can be considered as an "uncertainty" indicator of caption quality, offering additional interpretability in future optimization efforts.

Is the joint contrastive loss more advantageous? Existing work [3, 42] only exploit video-to-attribute contrastive loss as in Eq. 7 that enables calibration and separation of vision and semantic features. Furthermore, in Eq. 8, VAPNet introduces an additional video-to-video contrast that fully utilizes visual contexts to further improve the discrimination. By contrast, as shown in Table 2c, **VAPNet w/ Eq.** 8 gains a relative 3.2% improvement (46.9→48.4). To deeply understand the improvement of **VAPNet w/ Eq.** 8, we study the distribution of visual features on the space, and compute the degree of feature separation by averaging cosine distance among different class centers (*i.e*., the mean visual features). Our new loss Eq. 8 increases the separation degree from 0.873 to 0.881, the larger number indicates more uniformity and a better generalization [42].

4.3. Comparisons with SoTA under Protocol 1

E2E [3] proposed rigorous evaluation protocol (**Protocol 1**) as well as training protocol which requires non-overlapping classes between source and target datasets and adheres to learn visual features from scratch. Recent methods, AURL [42] and ResT [31] and our VAPNet, all follow the setting. Here, for fair comparisons, we set the same $\theta = 0.05$ in Eq. 10 as used in E2E [3]. In addition, we report numbers of alternative methods by replicating author's released codes or published results. In Table 3, we compare top-1 and top-5 accuracy on UCF [47] and HMDB [28] with 1 and 25 video clips. From the comparisons, we observe our VAPNet consistently surpasses existing SoTA alternatives. Specifically, the largest improvements happen at 25 clips by (14.3%, 8.8%) on UCF top-1 and HMDB top-5, while the smallest comes at 25 clips by (8.5%, 1.2%) on UCF top-5 and HMDB top-1. In conclusion, our VAPNet performs the best under the rigours training/evaluation setting.

Table 3: Comparisons on 100% classes of UCF and HMDB datasets (**Protocol 1**). E2E (aug) used the R(2+1)D pre-trained on SUN [57] dataset. Red/blue numbers are the best/second best.

Method	Video clips	SI	UCF top-1	UCF top-5	HMDB top-1	HMDB top-5
E2E [3]	1	W	35.1	56.4	21.3	42.2
E2E (aug) [3]		W	36.8	61.7	23.0	41.3
AURL [42]		W	44.4	70.0	27.4	53.2
E2E [3]		VA	42.6	66.7	24.7	45.4
VAPNet		VA	48.9	76.4	29.3	57.2
E2E [3]	25	W	37.6	62.5	26.9	49.8
E2E (aug) [3]		W	39.8	65.6	27.2	47.4
AURL [42]		W	46.8	73.1	31.7	58.9
ResT [31]		W	46.7	-	34.4	-
E2E [3]		VA	46.3	70.8	27.9	51.0
VAPNet		VA	53.5	79.3	34.8	64.1

4.4. Comparisons with SoTA under Protocol 2

There are other recent SoTA works (*e.g*., ER, JigsawNet, CLASTER) do not follow the rigours training protocol whose visual encoders are pre-trained on large-scale image datasets (*e.g*., ImageNet [7], SUN [57], MS-COCO [32]) or related action datasets (*e.g*., Kinetics400 [24], Sports-1M [23]). For comprehensive comparisons, we show results using **Protocol 2** in Table 4 where models of our VAPNet, E2E, AURL, and ResT are still trained with the rigours training protocol. First, we find that our VAPNet surpasses all the alternatives on both UCF and HMDB datasets, even may enduring unfair training protocols. Specifically, VAPNet improves the SoTA results by 7.0% (58.7→62.8) on UCF top-1 while 0.4% (43.2→43.4) on HMDB top-1. Second, our VAPNet with the proposed video attributes outperforms alternatives with various types of semantics. For instance, OD [35], E2E [3], AURL [42] and ResT [31] exploit conventional class names while DASZL [25] uses manual attributes and ER [5], JigsawNet [43] leverage extended object or class descriptions. To summarize, by explicitly introducing video attributes to enrich semantics, our VAPNet achieves the SoTA performance in ZSVC. Last but not least, our video attributes also can be considered as an advanced module which helps boost the performance of SoTA alter-

Table 4: Comparisons on 50% classes of UCF and HMDB datasets (**Protocol 2**). Semantic information (SI): manual attribute (MA), word embedding of category names (W), elaborative description (ED), our learnable and discriminative video attribute (VA), and visual embedding from pretrained model (Pre VE).

Method	SI	Pre VE	UCF top-1	HMDB top-1
WGAN [56]	MA	II	37.5	-
OD [35]	MA	II	38.3	-
DASZL [25]	MA	I, II	48.9	-
Act2Vec [18]	W	II	22.1	23.5
TARN [2]	W	II	23.2	19.5
WGAN [56]	W	II	25.8	29.1
OD [35]	W	II	26.9	30.2
Obj2act [21]	W	I	30.3	15.6
TS-GCN [15]	W	I	34.2	23.2
SAOE [36]	W	I	40.4	-
PSGNN [16]	W	I	43.0	32.6
E2E [3]	W	N/A	48.0	31.2
E2E (aug) [3]	W	I	49.2	32.6
AURL [42]	W	N/A	58.0	39.0
ResT [31]	W	N/A	58.7	41.1
ER [5]	ED	I, II	51.8	35.3
JigsawNet [43]	ED	I, II	56.0	38.7
CLASTER [17]	ED	II	53.9	43.2
E2E [3]	VA	N/A	55.1	36.1
VAPNet	VA	N/A	62.8	43.4

I: visual embedding/object scores from model pretrained on image dataset (*e.g.*, ImageNet [7], SUN [57], MS-COCO [32])
II: visual features from model pretrained on action dataset (*e.g.*, Kinetics400 [24], Sports-1M [23])

natives. For instance, we modify E2E [3] using our video attributes (VA) shown in Tables 3, 4 and obtain ∼15% gains.

4.5. Comparisons with large-scale pretrained SoTA

Closest to our VAPNet, contrastive model — CLIP [45] shows strong zero-shot transfer and generalization abilities thanks to pretraining on large-scale vision-language pairs. Despite potential overlaps between the CLIP datasets and our test data, we explore a modified version of VAPNet based on the CLIP benchmark and justify that our video attribute could further improve the performance of CLIP pretrained on vast corpus. Here we replace the visual and semantic encoders with pretrained couterparts from CLIP and obtain the modified VAPNet. For visual features, we resize frames to 224 × 224 following standard CLIP's augmentation, and average all frames' features in a video clip.

As illustrated in Table 5, we report the performance of origin CLIP, XCLIP which adjusts CLIP for videos and our VAPNet with various pretrained backbones. The CLIP and our VAPNet with $\star$ mean adopting prompt engineering on the abstract category names. From the comparisons, we observe our VAPNet consistently surpasses others under the same setting. Specifically, the largest improvements happen with (no prompts, ViT-L/14) by (4.1%, 17.2%) on UCF and HMDB under **Protocol 1**. We also notice that prompts, which requires careful design and exploration, are vital to the generalization of CLIP, *e.g.*, with ViT-L/14, CLIP *vs*. CLIP* corresponds to 76.1 *vs*. 80.5 on UCF while 43.1 *vs*. 54.3 on HMDB. We find our video attribute with no prompts could achieve comparable results to prompts-applied method, *e.g.*, (71.8, 45.3) for VAPNet *vs*. (73.7, 46.4) for CLIP* on (UCF, HMDB) with ViT-B/16. To sum up, our elaborate video attribute can enhance the transfer capability of large-scale pre-trained models on ZSVC.

Table 5: Comparisons with large-scale pre-training methods. Results under **Protocol 1/Protocol 2** of UCF and HMDB datasets are reported. $\star$ means using prompt engineering in the class name.

Method	Backbone	UCF top-1	HMDB top-1
CLIP [45]	ViT-B/16	68.7/78.9	41.1/51.9
X-CLIP [39]	ViT-B/16	- /72.0	- /44.6
VAPNet	ViT-B/16	71.8/81.0	45.3/56.5
CLIP* [45]	ViT-B/16	73.7/81.3	46.4/57.8
VAPNet*	ViT-B/16	74.2/83.5	48.3/58.8
CLIP [45]	ViT-L/14	76.1/84.8	43.1/53.9
VAPNet	ViT-L/14	79.2/87.2	50.5/61.6
CLIP* [45]	ViT-L/14	80.5/88.1	54.3/64.5
VAPNet*	ViT-L/14	82.0/88.7	55.0/65.0

Limitations and possible solutions. Even though our VAPNet achieves remarkable results, our method relies on video caption quality. Uncorrelated captions may harm the video attribute, where pre-processing or introducing text summarization tasks could help generate more reliable semantics. Besides, the general caption model may fail to describe fine-grained categories (*e.g.*, *playing tabla*). A stronger caption model with more knowledge may be helpful. It is also interesting to study how caption models affect the performance (*e.g.*, using different datasets or models).

5. Conclusion

We present a Video Attribute Prototype Network (VAPNet) to generate video attributes that enrich semantics by associating video captions with class-wise semantics. Besides, we propose two critical components: cross-attention module to learn shared attentions between instance-wise and class-wise semantics; caption uncertainty module to alleviate inaccurate captions from final video attributes. We conduct extensive ablations to justify the effectiveness of each module qualitatively and quantitatively. For comparisons with SoTA alternatives, we make comprehensive comparisons under complete training/evaluation protocols applied in ZSVC. What's more, we modify our VAPNet with the large-scale pre-trained CLIP backbone, showing the superiority of our proposed module on different benchmarks.

Acknowledgement. Research reported in this paper was supported by the Natural Science Foundation of China under grants 62076036 and 62076031.

References

[1] Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. *arXiv:1612.00410*, 2016.

[2] Mina Bishay, Georgios Zoumpourlis, and Ioannis Patras. Tarn: Temporal attentive relation network for few-shot and zero-shot action recognition. In *BMVC*, 2019.

[3] Biagio Brattoli, Joseph Tighe, Fedor Zhdanov, Pietro Perona, and Krzysztof Chalupka. Rethinking zero-shot video classification: End-to-end training for realistic applications. In *CVPR*, 2020.

[4] Shiming Chen, Ziming Hong, Yang Liu, Guo-Sen Xie, Baigui Sun, Hao Li, Qinmu Peng, Ke Lu, and Xinge You. Transzero: Attribute-guided transformer for zero-shot learning. In *AAAI*, 2022.

[5] Shizhe Chen and Dong Huang. Elaborative rehearsal for zero-shot action recognition. In *ICCV*, 2021.

[6] Anoop Cherian, Chiori Hori, Tim K Marks, and Jonathan Le Roux. (2.5+1) d spatio-temporal scene graphs for video question answering. In *AAAI*, 2022.

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2018.

[9] Zi-Yi Dou, Yichong Xu, Zhe Gan, Jianfeng Wang, Shuohang Wang, Lijuan Wang, Chenguang Zhu, Pengchuan Zhang, Lu Yuan, Nanyun Peng, et al. An empirical study of training end-to-end vision-and-language transformers. In *CVPR*, 2022.

[10] Valter Estevam, Helio Pedrini, and David Menotti. Zero-shot action recognition in videos: A survey. *Neurocomputing*, pages 159–175, 2021.

[11] Yanwei Fu, Timothy M. Hospedales, Tao Xiang, and Shaogang Gong. Learning multimodal latent attributes. *PAMI*, pages 303–316, 2014.

[12] Chuang Gan, Ming Lin, Yi Yang, Gerard de Melo, and Alexander G. Hauptmann. Concepts not alone: exploring pairwise relationships for zero-shot video activity recognition. In *AAAI*, 2016.

[13] Chuang Gan, Ming Lin, Yi Yang, Yueting Zhuang, and Alexander G. Hauptmann. Exploring semantic inter-class relationships (sir) for zero-shot action recognition. In *AAAI*, 2015.

[14] Chuang Gan, Tianbao Yang, and Boqing Gong. Learning attributes equals multi-source domain generalization. In *CVPR*, 2016.

[15] Junyu Gao, Tianzhu Zhang, and Changsheng Xu. I know the relationships: Zero-shot action recognition via two-stream graph convolutional networks and knowledge graphs. In *AAAI*, 2019.

[16] Junyu Gao, Tianzhu Zhang, and Changsheng Xu. Learning to model relationships for zero-shot video classification. *PAMI*, pages 3476–3491, 2021.

[17] Shreyank N Gowda, Laura Sevilla-Lara, Frank Keller, and Marcus Rohrbach. Claster: clustering with reinforcement learning for zero-shot action recognition. In *ECCV*, 2022.

[18] Meera Hahn, Andrew Silva, and James M. Rehg. Action2vec: A crossmodal embedding approach to action learning. *arXiv:1901.00484*, 2019.

[19] De-An Huang, Vignesh Ramanathan, Dhruv Mahajan, Lorenzo Torresani, Manohar Paluri, Li Fei-Fei, and Juan Carlos Niebles. What makes a video a video: Analyzing temporal information in video understanding models and datasets. In *CVPR*, 2018.

[20] Nazli Ikizler-Cinbis and Stan Sclaroff. Object, scene and actions: Combining multiple features for human action recognition. In *ECCV*, 2010.

[21] Mihir Jain, Jan C. van Gemert, Thomas Mensink, and Cees G. M. Snoek. Objects2action: Classifying and localizing actions without any video example. In *ICCV*, 2015.

[22] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021.

[23] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014.

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

[25] Tae Soo Kim, Jonathan D Jones, Michael Peven, Zihao Xiao, Jin Bai, Yi Zhang, Weichao Qiu, Alan Yuille, and Gregory D Hager. Daszl: Dynamic action signatures for zero-shot learning. In *AAAI*, 2021.

[26] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv:1312.6114*, 2013.

[27] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *ICCV*, 2017.

[28] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *ICCV*, 2011.

[29] Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, 2009.

[30] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *PAMI*, pages 453–465, 2013.

[31] Chung-Ching Lin, Kevin Lin, Lijuan Wang, Zicheng Liu, and Linjie Li. Cross-modal representation learning for zero-shot action recognition. In *CVPR*, 2022.

[32] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[33] Jingen Liu, Benjamin Kuipers, and Silvio Savarese. Recognizing human actions by attributes. In *CVPR*, 2011.

[34] Shilong Liu, Lei Zhang, Xiao Yang, Hang Su, and Jun Zhu. Query2label: A simple transformer way to multi-label classification. *arXiv:2107.10834*, 2021.

[35] Devraj Mandal, Sanath Narayan, Sai Kumar Dwivedi, Vikram Gupta, Shuaib Ahmed, Fahad Shahbaz Khan, and Ling Shao. Out-of-distribution detection for generalized zero-shot action recognition. In *CVPR*, 2019.

[36] Pascal Mettes and Cees G. M. Snoek. Spatial-aware object embeddings for zero-shot localization and classification of actions. *ICCV*, 2017.

[37] Pascal Mettes, William Thong, and Cees GM Snoek. Object priors for classifying and localizing unseen actions. *IJCV*, pages 1954–1971, 2021.

[38] Tomas Mikolov, Kai Chen, Greg S. Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *ICLR*, 2013.

[39] Bolin Ni, Houwen Peng, Minghao Chen, Songyang Zhang, Gaofeng Meng, Jianlong Fu, Shiming Xiang, and Haibin Ling. Expanding language-image pretrained models for general video recognition. In *ECCV*, 2022.

[40] Boxiao Pan, Haoye Cai, De-An Huang, Kuan-Hui Lee, Adrien Gaidon, Ehsan Adeli, and Juan Carlos Niebles. Spatio-temporal graph for video captioning with knowledge distillation. In *CVPR*, 2020.

[41] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014.

[42] Shi Pu, Kaili Zhao, and Mao Zheng. Alignment-uniformity aware representation learning for zero-shot video classification. In *CVPR*, 2022.

[43] Yijun Qian, Lijun Yu, Wenhe Liu, and Alexander G Hauptmann. Rethinking zero-shot action recognition: Learning from latent atomic actions. In *ECCV*, 2022.

[44] Jie Qin, Li Liu, Ling Shao, Fumin Shen, Bingbing Ni, Jiaxin Chen, and Yunhong Wang. Zero-shot action recognition with error-correcting output codes. In *CVPR*, 2017.

[45] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[46] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *EMNLP*, 2019.

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

[48] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, 2018.

[49] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, pages 2579–2605, 2008.

[50] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NIPS*, 2017.

[51] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016.

[52] Teng Wang, Ruimao Zhang, Zhichao Lu, Feng Zheng, Ran Cheng, and Ping Luo. End-to-end dense video captioning with parallel decoding. In *ICCV*, 2021.

[53] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, et al. Image as a foreign language: Beit pretraining for all vision and vision-language tasks. *arXiv:2208.10442*, 2022.

[54] Wei Wang, Vincent W Zheng, Han Yu, and Chunyan Miao. A survey of zero-shot learning: Settings, methods, and applications. *TIST*, pages 1–37, 2019.

[55] Zuxuan Wu, Yanwei Fu, Yu-Gang Jiang, and Leonid Sigal. Harnessing object and scene semantics for large-scale video understanding. In *CVPR*, 2016.

[56] Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. Feature generating networks for zero-shot learning. In *CVPR*, 2018.

[57] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, 2010.

[58] Wenjia Xu, Yongqin Xian, Jiuniu Wang, Bernt Schiele, and Zeynep Akata. Attribute prototype network for zero-shot learning. In *NIPS*, 2020.

[59] Xun Xu, Timothy M. Hospedales, and Shaogang Gong. Transductive zero-shot action recognition by word-vector embedding. *IJCV*, pages 309–333, 2017.

[60] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv:2205.01917*, 2022.

[61] Rowan Zellers and Yejin Choi. Zero-shot activity recognition with verb attribute induction. *arXiv:1707.09468*, 2017.